Advanced Visual Analytics Approaches for the Integrative Study of Genomic and Transcriptomic Data

Dissertation

der Mathematisch-Naturwissenschaftlichen Fakultät
der Eberhard Karls Universität Tübingen
des Wilhelm-Schickard-Institut für Informatik
zur Erlangung des Grades eines
Doktors der Naturwissenschaften
(Dr. rer. nat.)

vorgelegt von

M. Sc. Günter Jäger

aus Arad
(Rumänien)

Tübingen
2015

Tag der mündlichen Qualifikation: 20. April 2016

Dekan:	Prof. Dr. Wolfgang Rosenstiel
1. Berichterstatterin:	Apl. Prof. Dr. Kay Nieselt
2. Berichterstatter:	Prof. Dr. Oliver Kohlbacher

Dedicated to my parents
Maria & Walter

Zusammenfassung

Die Fortschritte der Next-Generation Sequencing (NGS) Technologie ermöglichten es ganze Genome schnell und kosteneffektiv zu sequenzieren. Heute weiß man, dass individuelle Organismen einzigartige Genomsequenzen haben und dass Unterschiede zwischen diesen Sequenzen der Grund für die genetische Vielfalt sind. Zudem werden die biologischen Prozesse lebender Organismen durch Gene und dem Zusammenspiel ihrer Produkte gesteuert. Störungen in diesen Systemen führen oft zu Krankheiten. Daher ist eine der wichtigsten Fragen der biomedizinischen Forschung, wie genetische Varianten Genfunktionen beeinflussen und wie diese auf zugrundeliegende Stoffwechselwege und Geninteraktionsnetzwerke einwirken. Eine der häufigsten Ursachen für genetische Variabilität sind Einzelnukleotidvarianten (SNVs). So genannte genomweite Assoziationsstudien (GWAS), wie auch expression Quantitative Trait Locus (eQTL) Studien, beabsichtigen SNVs mit z. B. krankheitsbezogenen, binären, oder quantitativen, phänotypischen Merkmalen zu assoziieren. Jedoch sind vorhandene Verfahren zumeist eingeschränkt auf statistische Methoden und bisherige Ansätze zur besseren Interpretation der entsprechenden Ergebnisse reichen oft nicht aus.

Das Ziel dieser Dissertation war es neue visuell analytische Verfahren zu entwickeln, um somit rein statistische Methoden in der Identifikation, Charakterisierung und Interpretation von SNVs zu unterstützen. Zu diesem Zweck wurde MAYDAY, ein Programm zur Expressionsanalyse, durch innovative, visuell analytische Methoden erweitert, um integrative Analysen im Bezug auf Varianten- und Genexpressionsdaten zu ermöglichen. Für GWAS basierte Analysen wurde inPHAP entwickelt, welches die visuelle Bewertung von Genotypen und Haplotypen mit chromosomaler Zuordnung der SNVs zwischen Populationen und Untergruppen von Populationen erlaubt. inPHAP bietet eine interaktive, Matrix-ähnliche Visualisierung und fortschrittliche Methoden zur Identifikation von SNV Mustern, insbesondere Aggregationen. Zusätzlich wurde REVEAL für die Analyse von eQTL Daten mit Fokus auf die integrative Analyse von SNV und der Kombination aus SNV und Genexpressionsdaten entwickelt. REVEAL bietet interaktive Netzwerk-, Matrix- und Tabellenvisualisierungen, die auf SNV und Genebene miteinander verknüpft werden können, um SNV–Gen Assoziationen effizient zu untersuchen. Schließlich ist mit Hilfe von GENOMERING die Analyse und Visualisierung von SNVs, sowie struktureller Varianten im Rahmen eines Alignments vollständiger Genome möglich. In dieser Arbeit wurden Erweiterungs- und Optimierungsstrategien zur Verbesserungen der Visualisierung struktureller Gemeinsamkeiten und Unterschiede in GENOMERING, sowie zur Reduktion visueller Störfaktoren im Allgemeinen, entwickelt.

Genetische Varianten, vor allem SNVs, spielen auch im stark wachsenden Gebiet der Paleogenetik eine wichtige Rolle, wo DNS altertümlicher Herkunft mit moderner DNS verglichen wird, um daraus Erkenntnisse zur evolutionären Geschichte zu gewinnen. In dieser Dissertation wurde eine computergestütze Pipeline für die vergleichende NGS Analyse altertümlicher und moderner DNS Proben beschrieben. Besondere Aufmerksamkeit galt dem Read-Vereinigungsschritt, der benötigt wird, um die Qualitätseinschränkungen altertümlicher DNS (aDNS), insbesondere DNS Fragmentierung und den Fehleinbau von Nukleotiden, zu meistern. Des Weiteren ist aDNS normalerweise nur in geringen Mengen zu gewinnen und oft mit DNS moderner Mikroorganismen verunreinigt. Um dieses Problem zu lösen, wurde eine hoch wirtschaftliche, Microarray basierte DNS Isolationsstrategie für die parallele Detektion und Anreicherung von aDNS aus über 100 verschiedenen menschlichen Pathogenen entwickelt.

Auf Grund des stetigen Rückgangs der Sequenzierungskosten, sowie durch die Verfügbarkeit robuster statistischer Methoden, wurde die Zeit, die ein Biologe, Kliniker, Wissenschaftler oder Bioinformatiker benötigt, sowie die Effizienz verschiedener Verfahren um Ergebnisse zu verknüpfen und zu interpretieren zum limitierenden Faktor. Folglich leisten die Programme, die in dieser Arbeit beschrieben wurden, einen außerordentlich wertvollen Beitrag zum Erfolg aktueller und zukünftiger Studien über genetische Varianten.

Abstract

The advances in next-generation sequencing (NGS) technology enabled rapid and cost-effective whole genome analyses. Nowadays, it is known that individual organisms have unique genome sequences and that differences between these sequences are the reason for genetic diversity. Furthermore, the biomolecular processes of living organisms are steered by genes and the interplay of their products. Perturbations in these systems often lead to disease. Thus, one of the major question in biomedical research is how genetic variations influence gene function, and how these affect underlying biological pathways and gene interaction networks. One of the most common sources of genetic diversity are single nucleotide variations (SNVs). So-called Genome Wide Association Studies (GWAS) as well as expression Quantitative Trait Locus (eQTL) studies intend to associate SNVs with e.g. disease related binary or quantitative traits. However, available methods are usually limited to statistical analyses and previous approaches to improve the interpretation of the respective results are often insufficient.

The goal of this dissertation was the development of new visual analytical approaches to assist purely statistical methods in the identification, characterization and interpretation of SNVs. For this purpose, MAYDAY, an expression analysis workbench, has been extended with innovative visual analytical methods to allow for integrative analyses with respect to variation and the combination of variation and gene expression data. For GWAS based analyses, INPHAP has been developed, which allows for the visual assessment of genotype and phased haplotype data between populations or subgroups of populations. It offers an interactive matrix-like visualization and advanced methods for SNV pattern identification, in particular aggregation. In addition, REVEAL was developed for the analysis of eQTL data with a focus on the integrative analysis of SNV and gene expression data. It offers interactive network, matrix and table visualizations to study SNV–gene associations, which can be linked to each other on the SNV and gene level. Finally, the analysis and visualization of SNVs as well as structural variations is possible in the context of whole genome alignments with the tool GENOMERING. In this work, enhancement and optimization strategies have been developed to improve visualization of structural similarities and dissimilarities in GENOMERING, as well as to reduce visual clutter in general.

Genomic variations, especially SNVs, also play an important role in the immensely growing field of paleogenetics, where DNA of ancient origin is compared to modern DNA with the intention to gain insights into evolutionary history. In this dissertation, a computational pipeline for comparative NGS analyses of ancient and modern DNA samples has been described. Special

attention was given to the read merging step, which is required to cope with the quality limitations inherent to ancient DNA (aDNA), in particular DNA fragmentation and nucleotide misincorporation. In addition, aDNA is usually only retrievable in low amounts and it is often contaminated with DNA of modern microorganisms. To solve this issue, a highly economical microarray-based DNA capturing strategy has been developed for the parallel detection and enrichment of aDNA from up to 100 different human pathogens.

With the costs for sequencing in steady decline and with robust statistical methods available, the time spent on and the efficiency of the integration and interpretation of results by biologists, clinicians, researchers and bioinformaticians has become the limiting factor in the field. Therefore, the tools described in this thesis make an invaluable contribution to the success of current and future studies on genetic variations.

Acknowledgements

First and foremost I would like to thank my advisor, PD Dr. Kay Nieselt, leader of the Integrative Transcriptomics group at the University of Tübingen. Kay has not just offered me the possibility to work on many exciting projects, but most importantly provided me the freedom to implement the various ideas described in this work and helped me to improve them through numerous inspiring and encouraging discussions. Especially, during the most difficult times when writing this thesis, Kay gave me moral support and guidance to improve the overall quality of my thesis. I am also very thankful for Prof. Oliver Kohlbacher's support and for being the co-supervisor of my dissertation.

This work would not have been possible without the remarkable individuals with whom I had the honor to work with and share ideas about specific projects and science in general. First of all, I want to express my appreciation to Dr. Stephan Symons. Although Stephan left shortly after I started my dissertation, he was the one who encouraged me to continue my studies on visual analytics applications. I would also like to thank Dr. Florian Battke with whom I had the pleasure to share a room with for over two years. Florian has been a great colleague and I enjoyed our productive and motivating talks very much. Special thanks go to Dr. Alexander Herbig for the countless inspiring discussions and the highly effective and productive collaborations in multiple research projects.

Furthermore, I owe gratitude to all other present and past colleagues at the Integrative Transcriptomics group for all the occasional talks about science and beyond, in particular Aydın Can Polatkan, Stefan Raue, Andreas Friedrich, Alexander Peltzer, André Hennig, Sven Fillinger and Alexander Seitz.

I am thankful for the help of Sabine Gebert-Rudolph with administrative tasks, such as paper submissions, event and trip organizations, as well as the bureaucratic challenges I had to face.

Of course, all of the accomplishments in this work would not have been possible without the many bright scientists with whom I was able to share ideas, especially during conferences. For assistance on the work on genotype and phased haplotype visualization I want to thank Dr. Julian Heinrich, Corina Vehlow and Prof. Daniel Weiskopf. For enlightening discussions about visual analytics for eQTL data I'd like to thank Dr. Jan Aerts, Ryo Sakai, Dr. Christopher W. Bartlett, Dr. William C. Ray, Prof. Jessie Kennedy, and Dr. Jim Procter. In addition, I am very grateful to Prof. Peter Wills for sharing his experiences and conceptions about science.

For the exciting work on ancient DNA and especially human pathogens, I owe gratitude to the people of the medieval leprosy consortium and the paleogentics group at the University of Tübingen. In particular, I want to thank Prof. Johannes Krause, Dr. Verena J. W. Schuenemann, Dr. Kirstin Bos, Alissa Mittnik, Dr. Ben Krause-Kyora, Prof. Steward Cole, Dr. Pushpendra Singh, and Dr. Andrej Banjak. Special thanks go to Prof. Johannes Krause who offered me the opportunity to learn about and work within the field of ancient DNA data analysis.

Furthermore, I want to express gratitude to Dr. Natasha Arora and Dr. Michal Strouhal for the interesting work on *Treponema pallidum*.

This work was also greatly improved by the many brilliant students I had the pleasure to supervise during their final theses. I am especially grateful for the work done by Simon Heumos on NGS data handling and manipulation. Furthermore, I would like to thank all the other students for their work on and improvement of Mayday, namely Sebastian Nagel, Jennifer Erfkämper, Eugen Netz, Ina Spirer, Alicia Owen, and Natalya Sabirova.

I want to thank my friends and family who supported me morally through all the rough times, in particular my brother Rainer Jäger for always being there for me, my cousin Melitta Bickel for her highly motivating nature and her husband Jan Bickel for the amazing mountain bike trips, Daniela Schwenk for all the dancing lessons, Alexander Peltzer, André Hennig, and Simon Heumos for the tough work-outs, and Sven Fillinger for the awesome motorbike tours. I would like to address my special appreciation to my very good friend Markus List. Although, Markus left Germany to continue his studies in Denmark, we never felt apart from each other. I am very grateful for his comments and the great discussions we had, which helped me a lot to improve the quality of this work.

And most of all, I want to thank my loving, encouraging and highly supportive partner Annika Wagner. Thanking can never express my feelings for you. You are the sunshine of my life. Thank you for brightening my world with the warmth of your love.

Lastly, but most importantly, I would like to express my deepest gratitude to my parents Maria and Walter Jäger, for their overwhelming love and support throughout my whole studies and my whole life. They always believed in me and have only the best wishes for me.

I dedicate this thesis to my parents, Maria and Walter Jäger.

Contents

List of Figures

List of Tables

1. Introduction

With the completion of the human genome project in 2003 [163], a major milestone in genetic research had been reached and similar projects for other organisms were initiated. The human genome project made use of Sanger sequencing technology, which is costly and time consuming. Thus, only large laboratories were able to afford large scale applications of Sanger sequencing for the analysis of whole genomes. However, the increased demand for more efficient sequencing techniques led to the development of next-generation sequencing (NGS), with the first commercially available sequencing platform presented in 2005 by Roche 454 Life Sciences [99]. With this, deep sequencing of large genomes became possible, leading to an exponential increase in the number of whole genome sequencing projects started during the last decade. As of today (03/11/2015), *The Genomes OnLine Database* (GOLD) [133] reports over 7,633 completed whole genome sequences and 27,298 permanent drafts.

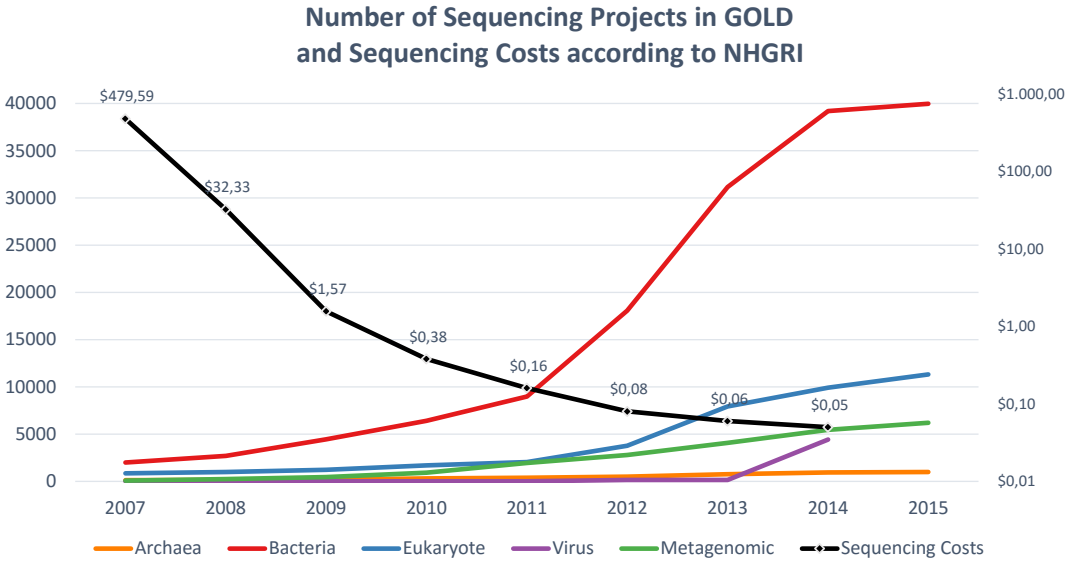

Figure 1.1: The primary y-axis (on the left) shows the number of sequencing projects per year grouped by organism domain (according to the GOLD database). On the log-scaled secondary y-axis (on the right) the sequencing costs per mega-base are shown. Cost data has been taken from the National Human Genome Research Institute (NHGRI)[1].

Together with projects currently in progress, a total of 65,737 different organisms have been or are still studied. The majority (74, 5%) are bacterial genome projects. However, the immense progress in the number of sequencing projects per year (see Figure 1.1) would not have been possible without an

exponential decline of the sequencing costs per mega-base.

NGS technology can be applied to various research areas. One of the major applications, however, is the identification of sequence variations between sequenced genomes. Using information about similarities and differences between genomes, one can address a multitude of different questions. In addition to structural variations, such as, for instance, DNA deletions or insertions, single nucleotide variations (SNVs) play an important role in explaining disease susceptibility and phenotypic differences between populations. So-called genome-wide associations studies (GWAS) aim at finding correlations between phenotypic traits and SNVs on a population scale. This provides the possibility to distinguish between common variations that are shared by many individuals and rare ones that are observed with much lower frequencies. Based on the distributions of variations between different subgroups of a population, SNVs of interest can be identified. If there is a significant association between a SNV and a phenotypic trait, this can often be traced back to individual genes. In many cases, the SNV has a direct influence on the gene product. For protein coding genes, for instance, SNVs can lead to changes in the amino-acid sequence and consequently to a modified three dimensional structure of the respective protein. Such modifications can lead to a reduced functionality, or in the worst case to a complete loss of functionality. In more complex scenarios, however, often not just a single SNV, but rather the combination of several SNVs has an effect on specific genes. Moreover, for diploid organisms the identification of clusters of SNVs located on the same chromosome, so-called haplotypes, is imperative to draw meaningful conclusions. Although, software solutions for the construction of haplotypes exist, there is a lack of tools for their interpretation and the determination of their effects.

In this dissertation, the interactive visual analytics tool inPHAP is described, which was published in 2014 [72]. Its main component is a matrix-like view that allows for the visualization of genotype as well as phased haplotype information. In fact, by the time of writing, inPHAP was the first and only interactive application for the visualization of phased haplotype data. It offers different visual representations to concentrate on specific data characteristics. In particular, differences between cohorts can be assessed on the nucleotide level or, more globally, by comparing frequencies of SNVs based on subgroups of individuals. The latter has been realized by the introduction of aggregation techniques that can be used to emphasize rare or common genotypes / haplotypes for user-defined sub-populations. Furthermore, interactive filtering can be applied to reduce the amount of SNVs that have to be investigated in parallel. inPHAP has successfully been applied to data from the 1000 Genomes project and showed great potential in re-

vealing population specific SNV patterns for rare as well as common variations.

However, SNV effects can also manifest indirectly. Instead of being located inside a gene, which may lead to modifications in gene function, SNVs can also be located far away from a gene and, for example, alter its expression. The consequence is often a disturbance of the underlying biological pathways, in which the affected genes are involved. Consequently, this leads to phenotypic alterations, as for instance, susceptibility to disease. Thus, combining the information gained from SNV based analyses and gene expression analyses can provide valuable insight into the characteristics of specific diseases. A typical technique for the measurement of gene regulation are expression microarrays. These are slides containing probes for thousands of different genes, which allows for the parallel and efficient assessment of gene expression. However, expression microarrays are slowly replaced by the application of NGS technology to isolated mRNA (so-called *RNA-seq*), which allows for the assessment of molecule abundances without a restriction on the dynamic expression range, as it is the case with microarrays. Although, attempts have been taken, to analyze these different types of data together, as for instance in so-called *expression Quantitative Trait Locus* (eQTL) studies, there is still a lack of software solutions that provide methods for an improved interpretability of the results.

To fill this gap, Reveal has been developed offering innovative and interactive visualizations and well established statistics to study SNV associated gene expression changes. To allow for an integrative analysis that covers SNV as well as gene expression data equally well, Reveal has been integrated into the gene expression analysis software Mayday. To this end, a general extension approach has been defined that allows to integrate not just SNV data, but also other data types in the future. Due to the tight interaction of Reveal and Mayday resulting from this extension approach, comprehensive analyses can be performed, since all visualizations in Reveal are linked to each other on the SNV, gene as well as subject level. In particular, selections of these data objects are synchronized between the different visualizations, which enables the user to apply the methods implemented more efficiently. Visualization approaches included in Reveal comprise graph- and matrix-based visualizations for single-locus and two-locus gene expression association data as well as established visualizations, such as Manhattan or LD plots. Reveal has successfully been applied to data from the BioVis eQTL data analysis challenge, where it was selected as the visualization experts' favorite. This led to its publication in 2012, shortly after the conference [71].

In contrast to single nucleotide changes, larger variations, such as structural rearrangements, are much harder to detect with NGS applications. However,

for bacterial genomes such variations are very common. This is due to a higher mutation rate of the genomes as well as *horizontal gene transfer*, where whole or large fragments of DNA sequences are exchanged between different bacterial organisms. These DNA fragments usually contain specific genes related to pathogenicity or drug resistance. Thus, a comparative study of small variations as well as of the structural differences between genome sequences from bacterial strains can provide valuable information. For a meaningful analysis often so-called *multiple whole genome alignments* are computed after the reconstruction of the original genomes either by *de novo*, or by mapping assembly approaches. This allows for the assessment of similarities and difference in the genomic composition. However, due to the complexity of the underlying variations, as well as the size of the genomes, gaining meaningful insights remains a challenge. Furthermore, the number of different organisms that are incorporated in the study design increases data complexity. Visualization approaches are suitable to address this hindrance, such as MAUVE's multiple alignment viewer [33], or genome browsers in general. However, all of them fall short when it comes to the characterization of sequence similarities and especially dissimilarities for comparative studies on multiple whole genome sequences.

To overcome the hindrances of previous whole genome alignment visualizations, GenomeRing has been developed. GenomeRing is a circular plot that is based on Alexander Herbig's SuperGenome concept. This allows for a convenient representation of similarities and dissimilarities between aligned genomes in a common coordinate system. GenomeRing was published together with the SuperGenome concept in 2012 [58]. In order to improve the visual experience with GenomeRing, an optimization algorithm has been developed in this dissertation that allows for the rearrangement of blocks from the SuperGenome based on three different criteria. Each of these criteria can be used to reduce visual clutter with respect to specific data characteristics. To demonstrate how choosing the right optimization strategy can lead to more visually appealing representations, GenomeRing has been applied to different bacterial multiple whole genome alignments with and without block order optimization. Furthermore, the possibility to add SNV and gene information to GenomeRing enables a convenient comparison of genome sequences on the basis of structural as well as single nucleotide variations.

The challenges in the detection and analysis of sequence variations mentioned above become even more difficult when samples with DNA quality issues are analyzed. Such samples are usually collected from sources exposed to extreme environmental conditions, leading to modifications of the DNA molecules. In living organisms, such DNA damage occurs naturally and can in general be repaired. However, for samples collected from dead organisms, DNA

damage can accumulate over time, resulting in fragmented, demethylated and sometimes mutated DNA material. Especially DNA from very old sources (e.g. hundreds or thousands of years old), so-called *ancient DNA* (aDNA), has to be treated carefully. Besides the quality issues resulting from sequencing aDNA, samples are usually contaminated with microorganisms, which requires additional cleaning procedures after sequencing. Thus, the processing of aDNA substantially differs from modern DNA, since the specific characteristics mentioned above have to be addressed adequately. This often requires several different cleaning and quality enrichment procedures, as well as specialized read processing before read mapping and subsequent data analysis. Performing each step in this processing chain manually, however, is time consuming and requires additional effort to adjust the necessary parameters at each individual step. Thus, processing pipelines can be of great assistance, allowing for the application of all the necessary methods in a single command using predefined parameter sets for each intermediate step. Due to the high level of automation, repetition of whole analyses for different biological samples is easily possible. Despite of the various advantages of existing pipelines, an appropriate aDNA specific pipeline that fulfills all of the requirements is currently missing. The only available pipeline for NGS based aDNA analyses, the so-called *Kircher pipeline* [80], has major drawbacks in its practicability in general, as well as its runtime complexity for large scale projects. Moreover, features for post-processing of the reconstructed mapping assemblies, such as variation detection and analysis, are missing.

To address the need for more appropriate analyses of sequence variations contained in aDNA samples, a new aDNA based analysis pipeline has been developed together with Alexander Herbig and published in 2013 [144]. To adequately address the quality issues with ancient DNA fragments, paired-end sequencing is usually applied resulting in overlapping read pairs. This overlap can be used to increase base quality in the overlap region. Although, methods to perform this read merging step exist [92, 96, 134, 150], none of them provides a fast and accurate read merging for aDNA samples. The ClipAndMerge tool developed for the pipeline described in this dissertation outperforms all other read merging solutions with respect to runtime, merging rate and the ability to produce merged high quality reads for subsequent mapping. In fact, read merging is the most crucial step in this pipeline, since all subsequent processes rely on high quality sequence information. In particular, read mapping efficiency and the ability to call variants with low false positive rates can be increased significantly with an accurate read merging. The pipeline has successfully been applied to ancient and modern *Mycobacterium leprae* strains in previous work [57]. In this dissertation, further applications are shown, to demonstrate the pipelines efficiency for comparative analyses of modern bacteria, in particular various *Treponema*

pallidum strains collected from all over the world.

Besides the challenges one has to face regarding quality constraints of aDNA, the amount of DNA contained in typical aDNA samples is also quite low. Thus, additional hindrances in the identification, as well as the extraction of aDNA from environmental samples can render the subsequent analysis difficult. Hence, suitable methods for the identification and enrichment of aDNA molecules of interest are required. To this end, capture microarrays are applied frequently, but the solutions available so far only address single organisms at once. In fact, existing array designs are only available for very few organisms. Moreover, capture techniques that allow for the detection and enrichment of dozens of different organisms in parallel are missing completely.

To provide a more economical capturing technique, the ancient pathogens screening array (APSA) has been developed in this dissertation together with an appropriate analysis tool that simplifies the evaluation of captured DNA fragments. The APSA, which was published in 2014 [17], allows for the parallel detection of almost 100 different human pathogens. This has been achieved by applying an oligo selection technique based on taxonomic relationships between the microorganisms of interest. Furthermore, analysis of the APSA captured reads is conducted by applying the aDNA analysis pipeline described above and mapped reads are further evaluated using the APSA analysis toolkit. This software calculates normalized read count data to allow for an unbiased comparison of the detected pathogens. The APSA has successfully been applied to a *Mycobacterium leprae* positive control and various negative controls, showing true positive enrichment rates of > 460 fold as well as negligible false positive rates.

In summary, this thesis presents multiple software applications to detect, analyze and visually assess single nucleotide variations, which address various research topics. The tools themselves, as well as their application provide a valuable contribution to the fields of visual analytics, genomics and transcriptomics. Due to the design choices made in this work, a foundation for future development and highly integrative analysis platforms has been established.

1.1 Outline

This dissertation is structured into three main parts. In the first part, special attention is drawn towards single nucleotide variation (SNV) analyses and the study of genotype and haplotype data. In chapter 2, an explanation of the theoretical and biological background information is given that builds the basis for understanding the developed methods and visualizations. In particular, detailed information is provided on the principles of visual analytics as well as expression and variation data analysis. The visual analytical approaches developed in this dissertation are based on MAYDAY, a toolkit for expression analysis with a powerful visualization framework. Chapter 3 provides detailed information on MAYDAY and the extensions that have been introduced to allow for an integrative study of expression and variation data. In chapter 4, the INPHAP application is described, which has been developed for the interactive exploration of genotype and phased haplotype data. Subsequently, REVEAL, a software solution for the comprehensive integration of expression and variations data in MAYDAY, is described in chapter 5. REVEAL offers new solutions in the form of interactive visualizations for eQTL data.

In chapter 6, which comprises the second part of this thesis, GENOMERING, which allows for the visualization of structural genomic alterations based on a multiple whole genome alignment, is described. GENOMERING makes use of the SuperGenome concept developed by Alexander Herbig, which defines a common coordinate system for aligned genomes. To reduce visual clutter and to improve the visual experience with GENOMERING, various optimization strategies have been developed and are described in this part of the thesis. Furthermore, additional information is given on how SNV data is visualized in the context of the SuperGenome.

The last part, comprising chapters 7 and 8, focuses on the processing, detection and analysis of variation data from DNA samples of ancient origin. There, new approaches are shown that address the issue of often unpreserved aDNA fragments, as well as low amounts of collectible aDNA. In chapter 7, a pipeline for the processing of sequencing reads resulting from aDNA samples is described and special attention is drawn towards the read merging step used to improve overall read quality. Chapter 8 focuses on the identification and enrichment of aDNA fragments from contaminated sources. This is achieved by applying a newly designed microarray-based DNA capturing approach called APSA.

This work on visualization and analysis approaches for variation data of various kinds is concluded in chapter 9 with a discussion of the presented work and an outlook into future directions.

2. Background

This chapter summarizes general information on the three main research areas addressed in this thesis, namely visual analytics, transcriptomics and genomics. Firstly, an introduction to data visualization and its application to large and heterogeneous data, with focus on visual analytical techniques, is given. Then information on expression and variation data analysis is provided, including state of the art data analysis approaches. This also includes strategies for the processing of old DNA (e.g. DNA of ancient origin). Last but not least, approaches towards the integration of expression and variation data are presented, such as expression quantitative trait locus data, which allow for deeper insights into the phenotypic implications of both *omics* fields.

2.1 The Preference of Data Visualization

The term *data visualization* describes the process of representing data as visual elements. By combining these elements, an image is constructed that carves out specific aspects of the information, while others are omitted. Choosing a suitable visual representation may provide important insights into the data and may help to answer questions related to the data. In this, tasks can be carried out more efficiently. One of the most prominent examples, where the preference of data visualization stands out clearly is shown by the *Anscombe's quartet* [5]. This example shows that pure statistical analysis is not sufficient to gain valuable insight into the data and that visualization can help to overcome this hindrance. *Anscombe's quartet* consists of four different data sets with two variables each. Figure 2.1 shows a visual representation of the data using point diagrams. All four data sets show identical statistical properties, with respect to mean, standard deviation, or the linear regression of the two variables (can be seen via red lines in figure 2.1). However, the visualizations clearly show that the data sets are very different in their nature. *Anscombe's quartet*, therefore prominently demonstrates the importance of data visualization for any analysis.

2.1.1 The Data Visualization Concept

The process of data visualization consists of several different steps that need to be taken, in order to transform data into a suitable visual representation. Ben Fry describes a series of seven steps [47] that are needed in data visualization. These guidelines have become the *gold standard* for any data exploration approach that tries to answer questions using visualizations. In the following, a description of these seven steps is given.

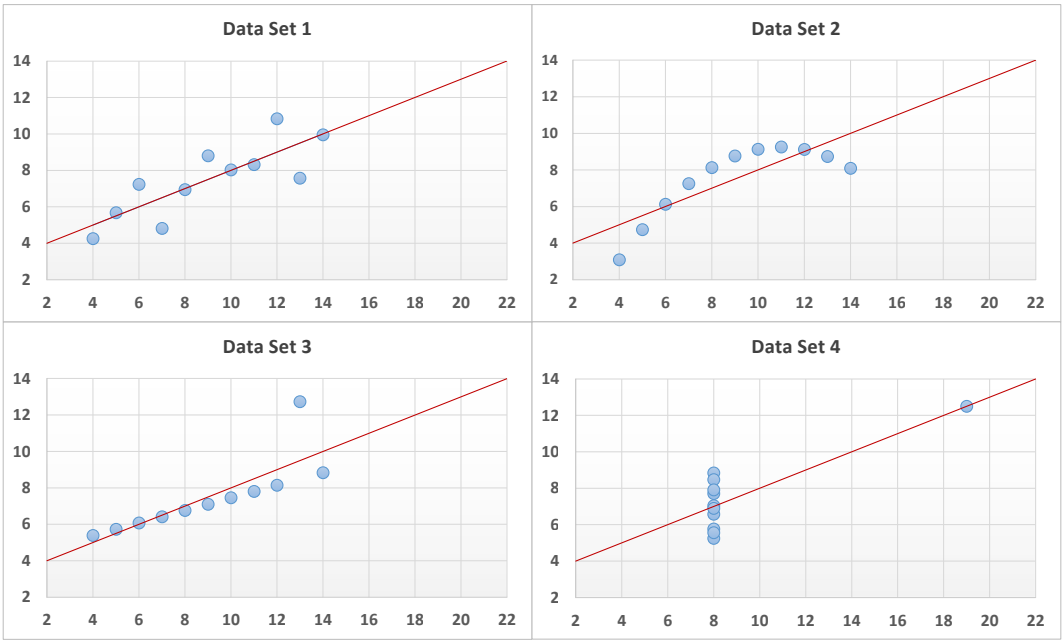

Figure 2.1: Point diagram visualization of *Anscombe's quartet*, showing four data sets with different characteristics, but equal statistical properties, such as equal mean. Although, totally diverse in their nature, statistical methods are not able to reveal the shape of the data as clearly as a visual approach.

(1) Acquire This step describes the process of obtaining data from a specific source. These sources can, however, be any kind of data storage device, such as a file from a disk, or a digital source obtained over a network.

(2) Parse The second step includes the transformation of the obtained data into a suitable structure. This especially means that an ordering of the data into different categories has to take place, which in consequence allows for an easy access and simple data handling.

(3) Filter Clearly, not all of the data that have been collected is usually needed to answer a specific question. Moreover, depending on the research topic different aspects are more important than others. This requires a pre-processing of the data, in order to remove all, but the data of interest.

(4) Mine With this step the focus is drawn to the application of methods from statistics or other data mining approaches, in order to discern patterns or to describe the data in a mathematical way.

(5) Represent As soon as the data is ready for visualization, a suitable visual model has to be chosen. For this, various different fundamental rep-

resentations can be used, such as graphs, lists or trees consisting of different visual elements. A more detailed description of the available visual elements is given in section 2.1.2.

(6) Refine Refine means that an improvement of the basic representation has to take place, in order to provide a clearer and more visually appealing view on the data.

(7) Interact The last, and one of the most important steps, describes the application of methods for manipulating the data. Thus, the user can control which features are visible and which are hidden. Furthermore, specific data aspects can be emphasized by interaction possibilities, such as zooming, panning, or selection.

2.1.2 Data Visualization Primitives

For the construction of powerful visualizations many different visual primitives can be used. These are basic geometrical elements, such as dots or lines, that are often related to specific data types. Moreover, a single visual primitive is usually not practical for every data type. In fact, preferred combinations of data types and geometric primitives used for visual representation exist. Tamara Munzner defines three different basic data categories in her book on visualization principals [109]. These are relational data, spatial data, and tabular data, where the latter can further be divided into categorical data and ordered data. Furthermore, ordered data can either be ordinal or quantitative. For each of these data types, one can define geometrical elements that best represent the data. An overview of the visual elements, grouped by data types, is provided in figure 2.2.

Besides the major visualization primitives, such as *points*, *lines*, *shapes*, or their *planar position*, *color* is one of the most often used tools in many visualizations. Although it could be shown that encoding information via different color hues is not effective in conveying information [26], it is still a highly valuable tool when applied with caution. In fact, most human individuals are not capable of differentiating between more than 12-15 different color values. Thus, color maps are usually introduced that provide a specific ordering, such that in visualizations only those colors are applied in close proximity to each other that can easily be distinguished. In addition, also single or multiple color gradients are used, since varying luminescence allows the identification of small differences. Nevertheless, which color map or gradient one should use largely depends on the type of data, as well as the question one wants to answer with the respective visualization. Thus, choosing an appropriate color encoding can become very difficult. In this, web-applications, such as ColorBrewer [52], are

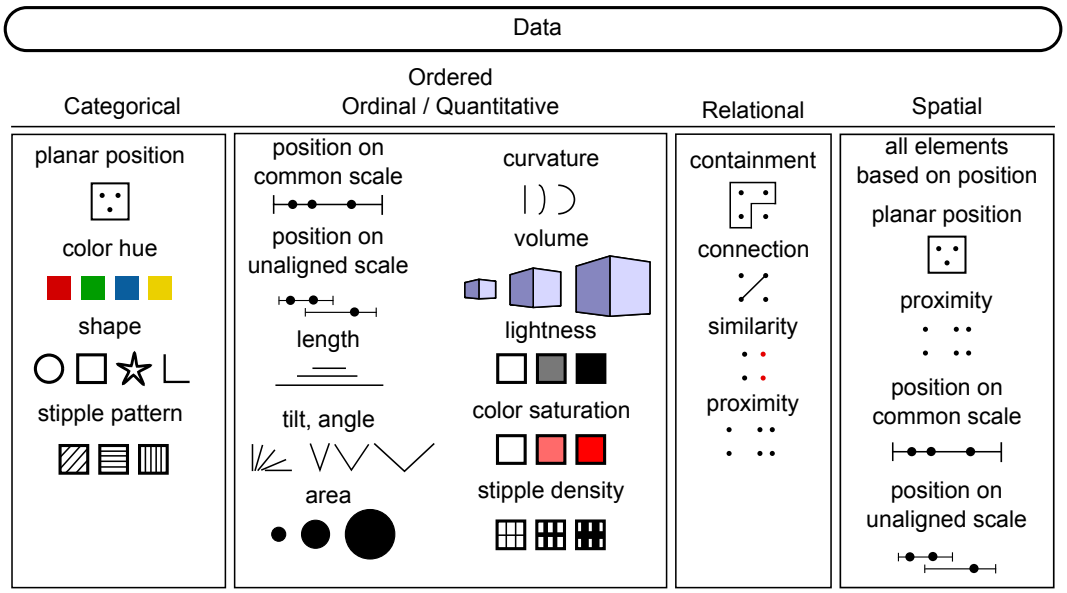

Figure 2.2: Overview of the most important visual primitives used for data visualization. The different elements are grouped based on the data types they suit best. This figure is based on a representation of data primitives by Tamara Munzner [109].

of assistance. This tool provides predefined color maps, as well as visually appealing and informative color gradients for various purposes.

2.2 Visual Analytics

Visual analytics describes the process of combining automated analysis techniques with interactive visualizations. The intention is to improve decision making processes for large and complex data sets. Therefore, visual analytics is especially useful in areas of research, where large information spaces have to be processed and interpreted. Although, the term visual analytics was invented based on the advances in computer science, where the generation of powerful visualizations could be achieved with minimal effort, its roots lie in a more general field. In 1977, John Tukey coined the term *exploratory data analysis*, which is also the title of his famous first book [159]. The common goal is to gain insights into data sets by envisioning their structures. A typical application is quality control of raw data, but also the visualization of results from statistical testing can be extremely useful. For these, various different visualizations have been introduced, such as scatter plots, pie charts, histograms, or box plots, just to name some of the most common ones. With all these visualizations, the main intention is to make use of human cognition, in order to detect patterns in the data and to analyze and interpret these.

Nevertheless, human interaction with a visualization is usually necessary. In this, the user can make refinements to the plots, such as zooming to get a more detailed view on the data, or filtering to concentrate on the data of interest. Consequently, this implies that the different steps of the visualization concept described in section 2.1.1 may have to be traversed iteratively to gain a final interpretation of the visualized data.

Consequently, visual analytical approaches to gain insights into data are best described by multiple iterations of data manipulation, data analysis, data visualization and refinement as well as data interpretation. This process is assisted by human interaction and automatic methods, as for instance data mining techniques that allow for the generation of data models. An alternation between automatic methods and visualization of the resulting data, in order to gain knowledge from the data, is characteristic for visual analytics. This strategy does not just allow for the detection of misleading steps early in the process, but in addition leads to results of higher confidence. Moreover, the insights gained from visualizations can be used to steer model building and to improve the automatic analysis. A schematic description of the visual analytics process in shown in figure 2.3.

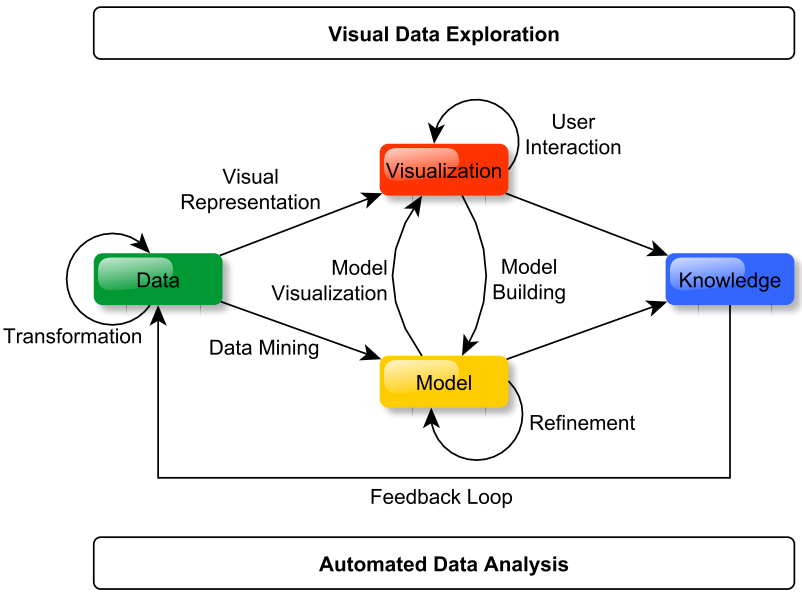

Figure 2.3: Schematic representation of the visual analytics process, combining automatic and visualization methods for an exhaustive data exploration. This figure is based on [78], page 10.

Nowadays, this process is applied in various different "omics" fields, including

genomics, transcriptomics, proteomics, metabolomics and many more. This thesis, however, concentrates on genomics and transcriptomics, i.e. the application of visual analytics to the study of variation and gene expression data, as well as the combination of both.

2.3 Gene Expression

2.3.1 Definition

Gene expression is the process in which a specific nucleotide sequence, called gene, is used as a template to synthesize a functional gene product. Thereby, genes can code for many different structures in a cell. Those that code for amino acid sequences are often called *structural genes*. Moreover, non-protein coding genes exist, such as those that guide transfer RNA (tRNA) or small nuclear RNA (snRNA) production. The nucleotide sequence of a gene is also known as the genetic code. This is the most fundamental level at which the phenotype of an organism can be influenced. Based on the *central dogma of molecular biology* defined by Francis Crick in 1970, gene expression involves two main steps [30]. In the *transcription* phase, messenger RNA (mRNA) is produced based on the gene sequence by an enzyme called RNA polymerase. Subsequently, these molecules are used to guide protein synthesis in the *translation* phase. The gene products are responsible for an organism's phenotype, since they either directly control the organism's shape, or they are involved in complex metabolic pathways, through which an organism is defined.

2.3.2 Gene Regulation

In order to control the rates at which genes are expressed, mechanisms that increase or decrease the production of gene products are needed. These processes are summarized under the term *gene regulation*. Thereby, a defined system of when and where genes get activated is built, and the amount of protein or RNA production is finely controlled. This is achieved by the interaction of genes, RNA molecules, proteins (e.g. transcription factors, which are initiators of gene transcription), or other mechanisms, such as post-translational modification. The regulation of genes was first discovered by Jacques Monod in 1961, who showed that enzymes involved in the lactose metabolism in the bacterium *Escherichia coli* can be activated and deactivated by increasing or decreasing the amount of lactose and glucose [69]. Gene regulation is an essential process in all living organisms, since it does not just allow for the adaption to environmental conditions, but also enables the expression of proteins when needed, e.g. during the development of the organisms phenotypic structures, such as cellular differentiation or morphogenesis.

2.3.3 Measuring Gene Expression

The measurement of gene expression is ideally suited to understand the ongoing mechanisms, networks and reactions in a cell. Nevertheless, there are a couple of assumptions that have to be made, which include that (1) more abundant genes/transcripts are more important, (2) gene expression levels correspond to protein levels, (3) a normal cell has a standard expression profile and changes to that profile indicate changes in a cell's biological processes, and (4) a cell's expression profile represents a snapshot of the cellular metabolism. Thus, the measurement of gene expression can provide valuable information on a cell's activity on the molecular (mRNA) level. Various different techniques exist to measure mRNA abundances, such as *Serial Analysis of Gene Expression* (SAGE) [162], *quantitative real-time Polymerase Chain Reaction* (qRT-PCR) [53], or *Southern Blotting* [149]. However, one of the most widely used tools in the past years and still nowadays, are *DNA microarrays* or *Gene Chips*, which will be explained in the following section. Today, this technique gets slowly replaced by the RNA-seq method, which applies NGS technology to assess mRNA abundances (for details see section 2.4.2).

2.3.4 Microarrays

Microarrays are based on the natural behavior of nucleotide sequences to hybridize into double stranded formations. Usually short DNA sequences of known nucleotide composition, called *probes* are immobilized on a solid surface, such as a glass slide and arranged in a well-defined order. These probes can either be *spotted* onto the slide (*spotted* arrays [142]) or synthesized directly on it (*in situ* arrays [56]). With both techniques, up to millions of different probes can be placed on a single slide, allowing for the measurement of thousands of different gene expression levels simultaneously using fluorescence labeling. While some arrays use only a single dye for labeling, resulting in *absolute expression levels* for each gene, others apply two different dyes. With the latter approach, a comparative analysis of gene expression levels from, for example, two different samples on the same array is possible. By using only a single dye, two arrays are needed to gain the same result. In any case, the measurement and comparison of gene expression levels allows for the detection of differentially expressed genes between two or more sample groups. A prominent example for *in situ* arrays are Affymetrix GeneChips®covering various different organisms, including human, mouse, rat as well as many other eukaryotic and bacterial species.

2.3.5 Gene Expression Analysis

Differential gene expression analysis is usually applied to experimental studies, where two or more conditions are compared with each other. There, differences

in the abundances of transcripts between the conditions are of interest, because of the assumption that these changes in expression may be the cause of the observed phenotype differences. That is, cells under different experimental conditions may react differently on the molecular level, since they try to cope with the respective treatment. Usually, such experiments are performed with three or more biological replicates, which allows for the assessment of the natural biological variance. In some cases, also technical replicates are produced to assess the technical variability between different microarrays. However, before differentially expressed genes can be detected, several pre-processing steps have to be taken, to assure comparability between different microarray raw intensity values. Generally, these steps include *background correction*, *intra-array normalization*, *inter-array normalization*, *feature summarization* and *base 2 log transformation*. These steps will be explained in more detail.

Background Correction Background noise is the result of non-specific hybridization or non-complete removal of unbound fragments during the washing phase. A common strategy to address this issue is to correct the foreground intensities based on the measured or estimated background. The simplest way to achieve this, is by subtracting the background intensity value for each probe from the respective foreground value. However, various different methods exist to perform a more sophisticated background correction. A popular example is the *Robust Multi-Array Analysis* (RMA) algorithm that was introduced for Affymetrix based microarrays [68]. This method uses the convolution of signal and noise distributions to estimate the background noise.

Normalization After the data has been corrected for background noise, the next step is to adjust for technical differences within a single array (intra-array normalization) and/or between different arrays (inter-array normalization). While intra-array normalization methods account for differences between probes on a single array, inter-array normalization tries to compensate for discrepancies between the hybridization processes for each array. Such discrepancies usually lead to scaling differences in the overall fluorescence intensities. Possible reasons are, for instance, differences in the amount of RNA in a sample, or differences in the time that was given for the hybridization phase. Normalization ensures comparability of microarrays, by compensating for technical effects. A typical method for intra-array normalization for printed arrays is the *Print-Tip Loess-normalization* [148], where differences in the amount of RNA for each print-tip are adjusted by a linear smoothing technique. For intra-array normalization, the *Quantile normalization* is usually applied [16]. There, the expression value distribution of each array is adjusted, such that specific statistical properties are similar between all the arrays that need to be compared during subsequent analyses.

Base 2 Logarithm Transformation Distributions of microarray expression values are often skewed, which means that most genes are expressed at very low expression levels and only a few show high expression values. A typical approach to deal with such kind of data is to transform the data on a logarithmic scale [130]. This leads to more symmetric and often Gaussian-like distributions. A further advantage is that fold-changes can be interpreted in terms of the chosen base of the logarithm. A common practice is to use the base 2 logarithm, since it allows for a convenient interpretation of the fold-change values.

Summarization Microarrays usually contain several different probes for a single gene (in particular those of Affymetrix). To obtain a single expression value on the gene level, the normalized intensity values for each corresponding probe have be summarized. The simplest summarization approach is to take the mean over all probes corresponding to the same gene. However, more complicated and statistically motivated methods exist, as for instance, the *Median polish* method [62] included in the RMA normalization.

Statistical Testing In order to identify differentially expressed genes, statistical testing is commonly applied. An example would be the Student's t-test [154], where the within group variance is evaluated and compared to the between group variance, rather than making decisions on a single difference threshold only (e.g. difference in the $\log_2$ fold-change, which is also a common strategy, if statistical testing cannot be applied). The result of such a statistical test is a p-value that describes how probable it is to obtain the observed difference in expression levels when drawing both expression values from the same normal distribution, rather than from two different ones. Genes with a p-value smaller than a predefined significance threshold (typical thresholds are ≤ 0.05, ≤ 0.01, or ≤ 0.001 after correcting the p-values for multiple testing) are defined as differentially expressed with statistical significance.

2.4 Next-Generation Sequencing

The term *nucleic acid sequencing* describes the process of determining the exact sequence of nucleotides in DNA or RNA molecules. This technique has become popular with the completion of the Human Genome Project in 2003 [86, 163]. This project, which lasted for about 13 years and cost around USD 2.7 billion [67] laid a milestone in the progression of sequencing projects in the following years. For the sequencing itself, technology from the so-called first generation has been used, which is known as *Sanger sequencing* [138]. This method (the chain termination method) was developed in 1975 by Frederick Sanger and is considered the gold-standard for now more than three

decades [51]. However, this method is extremely expensive when sequencing whole genomes (e.g. high consumption of reagents, expensive equipment, personnel-intensive, time consuming, etc.). Thus, there was a huge demand for cheaper alternatives, leading to the development of the second-generation sequencing methods, also termed *next-generation sequencing* (NGS). With this technology massive parallel sequencing has been made possible, where millions of DNA fragments can be sequenced in a union, opening new ways for high-throughput data generation. For example, today NGS allows the sequencing of an entire human genome in only a few hours with sequencing costs between USD 1000-2000 [51].

A variety of different NGS platforms have been introduced with the most commonly used systems nowadays being the Illumina MiSeq, the Illumina HiSeq, as well as the Illumina NextSeq. Although, most research facilities rely on the Illumina sequencing technology, also other platforms exit, such as the PacBio RS from Pacific Biosciences, which allows for single molecule, real time (SMRT®) sequencing. With these machines, sequencing became feasible for smaller labs, which led to a massive increase in the amount of sequencing projects in the last years. Furthermore, also clinical diagnostics rely more and more on sequencing to make well informed decisions on genetically related disease states.

Since all the available sequencing platforms are more or less unique in the way the sequencing itself is accomplished, we will mainly concentrate on technologies that rely on *sequencing by synthesis*, which is true for all Illumina platforms. For these platforms, the general steps include *template preparation*, *sequencing and imaging*, as well as *data analysis* [102]. The last step largely varies between the different NGS applications. Thus, we will mainly concentrate on the main processing steps required by most of them, which are *de-multiplexing*, *quality filtering*, and *read mapping* to a reference genome.

2.4.1 NGS Methodology

The application of NGS technology implies several different steps, starting with DNA or RNA samples from arbitrary sources. These steps, together with a very general description of the preprocessing of the resulting sequencing data, are described in the following.

Library Preparation The first step after DNA extraction from a sample involves building a library of nucleic acids (here either DNA or complementary DNA (cDNA) can be used). This is achieved by fragmenting the original DNA/cDNA sequence into smaller pieces and ligating an adapter sequence (synthetic oligo-nucleotides of known sequence) to the 5' and 3' ends. These

preprocessed DNA fragments are then PCR amplified followed by gel purification. This constitutes the final so-called library, which is ready to be loaded onto a sequencing flowcell to conduct the actual sequencing procedure. The flowcell itself is a glass slide containing oligo sequences that are complementary to the ligated adapter sequence, such that DNA fragments can be immobilized.

Cluster Generation Due to the synthetic adapter sequences, fragments from the library are captured on the lawn of complementary oligos that are bound to the surface of the flowcell. By a process called bridge amplification, each fragment is then amplified once again into distinct, clonal clusters, which are subsequently sequenced.

Sequencing and Imaging The bound and amplified library fragments act as a template in the following process. With sequencing by synthesis, a new DNA fragment is synthesized directly on the flowcell using the library fragment as a template. When flooding the flowcell with a known fluorescently labeled nucleotide (for example with adenine), it is incorporated into the growing DNA strand, and can be recorded digitally [131]. Before recording, the remaining unbound nucleotides are washed off. This process is repeated for a predefined number of cycles. Thereby, one cycle comprises the iterative flooding and washing with all four possible nucleotides. As a result, nucleotide sequences (so-called reads) are produced, whose length corresponds to the number of different cycles.

Paired-End Sequencing Paired-end sequencing describes the process of sequencing both ends of a sequencing library, rather than only one end. This strategy results in two different reads, a forward and a reverse read. These are then combined into a read pair for further processing. Paired-end sequencing has a couple of advantages over single-end sequencing, most notably the reduction of time and costs. Paired-end sequencing produces twice as much sequencing information in a single run as single-end sequencing and in addition offers possibilities for an advanced data analysis. First of all, read pairs offer a more accurate read alignment and a more reliable detection of single nucleotide variations, insertions and deletions (*indels*). Since different read pairs usually share approximately the same read-pair spacing (*insert size*), a differential analysis of read pair insert sizes allows for a more sophisticated detection of PCR duplicates and their subsequent removal [51]. PCR duplicates are a common artifact during library preparation and often lead to coverage biases and false positives in the subsequent analyses. Furthermore, paired-end sequencing is better suited to deal with repetitive genomic regions, because larger DNA fragments can be spanned. Although most researches currently follow the paired-end sequencing approach, there are still NGS applications

that are better suited for single-end sequencing, as for instance small RNA sequencing.

2.4.2 NGS Applications

NGS applications are versatile. The extreme reduction of sequencing costs in the past years led to an immense increase of whole genome sequencing projects of a wide variety of different organisms. Projects, such as the 1000 Genomes project [28] for human individuals, or equivalent projects for prokaryotes, as for example the Human Microbiome Project [124], prove the wealth of knowledge one can gain by applying NGS technology. Further fields of application are the sequencing of bacterial strains, viruses or other human pathogens to facilitate the identification of virulence factors leading to disease. Another more specialized application is targeted sequencing, where specific DNA fragments of interest are captured preliminary to the actual sequencing process. This is usually applied in diagnostic settings, where human individuals are screened for known disease causing alterations in the genome. In addition to the classical sequencing of DNA material, one can also start with mRNA material. This so-called RNA-seq approach offers the possibility to measure transcript abundances with a much larger dynamic range of expression level detectability as this is the case for microarrays. Furthermore, results from an RNA-seq experiment can be visualized on the sequence level, providing additional insights into the data. Another advantage of RNA-seq is the possibility to detect alternative splicing events, gene fusions, as well as single nucleotide variations from a single RNA-seq experiment, without additional sequencing costs. Due to these advantages over micorarrays and especially due to the shrinking sequencing costs, RNA-seq is slowly replacing traditional microarray analyses [51].

NGS Applications for old DNA Despite the many fields of research involving the analysis of modern DNA, mentioned above, the study of DNA from human, animal and plant remains offers further possibilities. First of all, it provides insights into evolution. An example is the identification of about 1-3% of the modern human genome sequence that originated from the Neanderthales [165]. However, not only prehistoric legacies are of interest, but also the identification of genetic links between people living today and those who lived in the past [50]. Furthermore, the study of human pathogens, for which DNA can still be present in bones, or if available, tissue remains, is a valid source of information about ancient diseases. This could, for instance, be shown to work well for cases of plague from the middle ages [18]. Another issue is the identification of sex based on human remains. Considering the age and condition of ancient samples, an identification based purely on visual inspection of the remains may be impossible. However, sequencing the ancient DNA (aDNA) obtained from such a sample, offers the possibility for accurate

sex determination. Last but not least, the collection of samples from ancestral remains from all over the world and comparing the DNA information content of these to each other as well as to people living in those regions today, may provide information about migration patterns. Hence, a deeper understanding of how modern humans migrated from Africa into Europe, Asia, and also America can be obtained.

Nevertheless, the analysis of aDNA is not as straightforward as for modern DNA. This is due to some special characteristics of aDNA, that are shaped over time. Despite the fact that usually only a very little amount of aDNA can be extracted, two major biochemical modifications that take place naturally, render the retrieval of the information contained in the DNA difficult. Firstly, DNA gets fragmented over time resulting in sequence lengths as short as 100-500 base pairs [63]. Secondly, aDNA is damaged. It can often be observed, that a large number of sites are modified, as for instance by oxidization. This primarily involves pyrimidine bases (i.e. Cytosine or Thymine) [63]. Mis-incorporations resulting from these modifications, can render the analysis difficult due to, for instance, base specific differences of the sequenced reads to a chosen reference genome. To address these issues, specialized bioinformatic approaches are needed. Some of which will be introduced in more detail in chapters 7 and 8.

2.4.3 NGS Data Analysis

Sequencing results in raw intensity values for each cycle and nucleotide. Thus, the first step in each data analysis procedure is the transformation of these intensity values into readable sequence information. This process is called de-multiplexing. The result is a text-based file (a so-called `FASTQ` file) that contains four lines for each sequenced read. Two of these lines represent the sequence information and the respective quality values for each of the nucleotides in a read. The other two lines are used as headers for the sequence and the quality line and contain among others, information about the instrument, the flowcell, and the cluster tile on the flowcell, as well as information about the the sequencing mode (either single-end or paired-end sequencing). Based on the `FASTQ` files, various different analyses can be performed. However, most NGS applications share some general analysis steps involving the preprocessing of the sequenced reads. As described in section 2.4.1 artificial adapter sequences of known nucleotide composition have been added to the DNA fragments, in order to enable binding to the flowcell. During sequencing parts of these adapter sequences may also be sequenced, especially if the DNA fragments are short and the number of cycles during sequencing is larger than the fragment length. Since the sequence information from the adapters would fudge analyses based on the raw reads, *adapter*

clipping is performed at first. Thereby, semi-local alignment algorithms are typically applied for each read and the respective adapter sequence to detect the artificial nucleotides, which are then subsequently clipped off. Furthermore, all sequencing machines tend to show a decrease in sequence quality with increasing number of cycles. Thus, it is common practice to apply a quality trimming from the 3' ends of the sequenced reads, with a predefined quality threshold. This is necessary to increase the average quality of the reads and in consequence the probability for an alignment with the reference genome in the next step.

Based on the preprocessed high quality reads, two different approaches are usually followed. The reads are either aligned to a user-defined reference genome (*mapping*), or a *de novo* alignment of the reads is calculated (*assembly*), which, besides other applications, offers the possibility to discover previously unavailable genome sequences. When the mapping approach has been followed, several different possibilities exist for further analysis. Typical tasks are the detection of small variants in comparison to the reference (*variant calling*), including *single nucleotide variations* and *indels*. The latter are variations such as insertions and deletions that are no longer than a few nucleotides. But also larger structural variations can be detected. Depending on the respective NGS application (see section 2.4.2) also the detection of novel genes or regulatory elements, as well as the assessment of transcript expression levels is possible. This, however, implies a need of bioinformatic expertise for the development, application and maintenance of appropriate software solutions. Fortunately, many different open-source tools already exist for the main analysis tasks, as for instance, quality assessment tools (e.g. FastQC [4]), read mappers (e.g. BWA [90]), or programs for the assembly of new sequences (e.g. SOAPdenovo2 [94]).

2.5 Genome Wide Association Studies

With advances of the NGS technology and consequently the initiation of projects, such as the 1000 Genomes project [28], studies on a population scale have been made possible. Of special interest are thereby phenotypic differences between cohorts of people, as for instance susceptibility to certain diseases. Understanding the genetic mechanisms leading to disease provides the possibility for better treatment and more specialized medication. In this, single nucleotide variations (SNVs) have become of major interest, since these can have huge effects on gene function. Studies that involve the investigation of hundreds of thousands of SNVs with the intent to link genetic markers, or risk factors, to phenotypic characteristics have shown huge potential [98]. The ultimate goal of these so called *Genome Wide Association Studies*

(GWAS) is to identify genotypic patterns, in order to make predictions about who is at risk and what treatment strategies are needed. An example would be the work by the Wellcome Trust Case Control Consortium[1], which has identified variation-associated phenotypes for various different diseases, including malaria [73], or myocardial infarction [76]. Such studies, however, led to a massive increase in detected variations and variation-phenotype associations. Therefore, the International HapMap Project has been created, which tries to catalog all known genetic variations in the human genome [29]. As of today, a total of 23.6 million different variants are listed. Another example, would be the *Collaborative Oncological Gene environment Study* (COGS) that focuses on genetic susceptibility of hormone-related cancers [15, 40, 48, 103, 125]. Besides SNVs also other genetic variations have been associated with disease, such as small insertions, deletions or repeated DNA fragments. A very prominent example are chromosomal aberrations in cancer, or trinucleotide repeats, which can have severe effects on human health. One of the most prominent trinucleotide related diseases is Huntington's disease, for which its severity is directly correlated with the trinucleotide repetition rate [166]. However, in this thesis the main focus lies on the development of new visualization strategies for single nucleotide variation based data.

In the following, detailed information on variations, especially SNVs, is given and the principles of GWAS are introduced, followed by typical statistical approaches that are widely used in most of today's GWAS. These build the basis for the data used in the visual analytics approaches developed in this work.

2.5.1 Single Nucleotide Variations

Single nucleotide variations (SNVs) are modifications in the DNA that manifest as a substitution of single base pairs. Often the term *single nucleotide polymorphism* (SNP) is used equivalently to describe such single nucleotide changes, disregarding that a polymorphism only describes changes that occur with high frequency (above 1%) on a population scale [27]. SNVs are the most abundant variations in the human genome. Abecasis *et al.* showed that each human individual carries on average about four million variants [28], but most of them do not have any impact. However, depending on the location of a SNV and the nature of the respective nucleotide substitution, functional consequences can be observed. SNVs can lead to amino acid changes, when located in coding regions of proteins. Such substitutions are called *non-synonymous*, whereas substitution that do not change the respective amino acid are termed *synonymous*. Amino-acid changes usually affect protein formation and consequently lead to reduced or even complete

[1] `http://www.wtccc.org.uk` (01/06/2015)

loss of functionality [106]. Furthermore, effects of SNVs outside of coding regions can also be observed, such as reduced affinity of transcription factors to their respective binding cites [106].

Humans, as well as most other mammals, have two copies (alleles) of each genetic locus. Thus, each nucleotide is represented twice in the genome and SNVs can occur in different forms, namely heterozygous variations, where only one allele is mutated and homozygous variations, where both alleles are changed. Looking at whole populations, one can define base pair frequencies in terms of the minor (the less common) allele. Within a population, a SNV with, for example, a minor allele (a) frequency of 0.3, means that 30% of the population has the a allele, whereas 70% have the more common allele (the major allele A).

Genotype and Haplotype In diploid organisms, combinations of base pairs can be defined at each genetic locus. These combinations are called genotypes. This definition, however, does not provide any hint about the parentage of the respective allele. In general, for diploid organisms, one copy of each chromosome is inherited from the mother and the other copy from the father. In this context, one also speaks of homologous chromosomes. However, such homologous chromosomes can be genetically very different. For some diseases, as for instance, cystic fibrosis, not a single variant causes the disease, but a combination of different variants within the same gene [79]. However, with genotype information only, a diagnosis is not possible, since the phase of the mutations is not known. Combinations of genetic variants that are located on the same DNA molecule are called (phased) haplotypes. This term was first used for the Human Leukocyte Antigen (HLA) in 1967, which consists of a set of genes located in close proximity to each other on chromosome 6. For phased haplotypes, the origin of the alleles (either maternal or paternal) is known. There are basically two ways to obtain this information. One can either directly infer the phase, if no other allele combination is possible, or apply haplotype phasing tools that make predictions on the phase of each SNV. Examples for such tools are SHAPTEIT2 [34, 35] and BEAGLE [20, 21], which both require family based data in order to make predictions on the phase. These data usually contain trios, which are combinations of three subjects, a child, a father and a mother. Based on these data, haplotypes can be inferred from combinations of SNVs following the Mendelian inheritance rules.

Linkage Disequilibrium Variations that are in close proximity to each other, i.e. located on the same chromosome, are likely to be passed on to the next generation in combination. As defined above, these combinations of SNVs are also called haplotypes. In contrast to that, SNVs located on different chromosomes may be separated during meiosis. The circumstance of non-random

association between two or more genetic loci is known as chromosomal linkage. However, linkage can also occur for smaller parts of a chromosome and also on a population scale, leading to the more general term *linkage disequilibrium* (LD). LD is the result of shared ancestry between genetic loci. Nevertheless, there are many factors that influence linkage leading to its decay. Crossing-overs during meiosis are the most prominent example for the loss of linkage, which are influenced by a number of other factors, namely population size, the number of founding chromosomes in the population, as well as the number of generations for which the population has existed [36]. Consequently, different human subpopulations show different patterns of LD and different rates of linkage decay. In order to assess LD in a mathematical way, various measures have been proposed. However, all these measures share the same principle, namely comparing the observed frequency of co-occurrence for two alleles to the expected frequency under the assumption of independence. The two most common measures of LD are D' and r^2 [36]. The equations for these measures are:

$$D = \pi_{AB}\pi_{ab} - \pi_{Ab}\pi_{aB} \tag{2.1}$$

$$D' = \begin{cases} \frac{D}{\min(\pi_A\pi_b, \pi_a\pi_B)} & \text{if } D \geq 0 \\ \frac{D}{\min(\pi_A\pi_B, \pi_a\pi_b)} & \text{if } D < 0 \end{cases} \tag{2.2}$$

$$r^2 = \frac{D^2}{\pi_A\pi_B\pi_a\pi_b} \tag{2.3}$$

In these equations, π_A is the frequency of the major allele A and π_B the frequency of the major allele B. Consequently, π_{AB} is the frequency of the AB haplotype of the two major alleles A and B. The frequencies for the minor alleles a and b are defined equivalently. D' takes values between 0 and 1. Thereby, a value of 0 indicates complete linkage equilibrium, meaning that the two alleles are statistically independent under the principles of the *Hardy-Weinberg Equilibrium* (HWE, for details see section 2.5.2), which implies frequent recombination events between them. If $D' = 1$, then complete linkage disequilibrium is given. No recombination between the two alleles can be observed within the population and HWE does not provide statistical significance.

The second parameter, r^2, is a statistical measure of correlation. Hence, the interpretation of its values slightly differs from the D' measure. Large correlation values indicate a similar information content of the two alleles, since one allele of the first SNV is often observed with one allele of the second SNV.

Although these measure are interpreted differently, they are closely related to each other. Due to the sensitivity of the r^2 measure to the respective allele frequencies of the two SNVs, large correlation values are only seen in regions, where also D' indicates strong LD.

2.5.2 Single-Locus Association Testing

One of the major challenges in GWAS is the identification of those SNVs that show a significant association with a specific phenotypic trait. The investigation of data from single human individuals does often not provide the necessary certainty, which is why population based approaches are usually chosen. This allows for statistical testing on a sufficient amount of people, in order to draw well informed conclusions. If a binary trait is to be investigated, so-called genetic association case-control studies are applied. There, human individuals are separated into two different groups, a case group containing all those individuals that carry the trait, and, respectively, a control group that does not show any phenotypic characteristics of the trait. Genetic variations are then tested for an association with the trait on the basis of these two cohorts. Statistical tests, however, are usually performed separately for each individual SNV under the assumption of a specific genotypic model. As shown earlier, human individuals have two alleles of each SNV, thus leading to three possible combinations of the minor allele a and the major allele A. Based on the chosen genotypic model, genotype counts are summarized in so-called contingency tables, which are then used to assess significant differences between the case and control group. Under the null hypothesis of no association of the SNV and the phenotypic trait, the allele frequencies of the case group and the control group are expected to be highly similar. A statistical test, therefore needs to assess whether there is a significant difference between the genotype frequencies of the two groups. A typical example for such a statistical test is given by the χ^2-test. Provided with a contingency table of allele frequencies, this statistic looks for independence of rows and columns from the table. In the following, detailed information on the different genotypic models and the composition of contingency tables is given. Additionally, statistical tests are described, including the χ^2-test as well as other possible approaches that are capable of finding significant genetic associations with binary traits.

Genotypic Models As it is the case for diploid organisms, two alleles for each SNV give rise to three different genotype possibilities, namely AA, Aa, and aa. Based on these combinations, genotypic models can be build, which define the impact of a heterozygous or homozygous variation. The four most commonly described models are the dominant model, the recessive model, the multiplicative as well as the additive model [88]. Under the dominant model, it is assumed that there is a higher risk when having one or more copies of

the major allele A in comparison to the allele a. Here individuals with the genotypes AA or Aa are compared to people with the aa genotype. In contrast to that, the recessive model assumes that two copies of the major allele are required, in order to increase the risk. This means that only people with a homozygous AA genotype are affected. The multiplicative model assumes that risk increases exponentially with each additional risk allele. If, for example, having one copy of the A allele increases the risk by a factor of k, then having two copies of A leads to a k^2 times higher risk. The additive model is related to the multiplicative model, however, here a linear relationship between the risk alleles is assumed. This means that if having one copy of the allele A increases the risk by a factor of k, then the risk of having the genotype AA is $2k$. These models are used to conduct statistical tests for association. However, in most studies not a single model is chosen *a priori*. In fact, multiple models are evaluated in parallel, coupled with an appropriate method for multiple testing correction (see section 2.5.3).

Case-Control Study The *genetic association case control study* is the simplest form of genetic study design, where a binary phenotypic trait is assumed. In this design, a series of cases, carrying the specific phenotypic trait (usually an affection with a disease) are compared to a series of control individuals. To conduct such a study, a 2×3, or 2×2 contingency table is needed that summarizes the genotype or the allele frequencies, respectively, within each group.

Contingency Tables A contingency table summarizes genotype or allele frequencies for different cohorts. Thereby, it is differentiated between the major allele A and the minor allele a. A 2×3 contingency table can then be constructed by simply counting the number of occurrences of the three different genotypes in the two cohorts. An example of a 2×3 contingency table is shown in table 2.1. If one is rather interested in the alleles than in the genotypes, this 2×3 contingency table can easily be converted into a 2×2 contingency table by summing up the frequencies of the major and minor alleles individually (see table 2.2).

Contingency tables are used to decide if effects between different cohorts are present. These effects are reflected by significant similarities or differences between the row and column variables. In the following, statistical methods based on contingency tables are introduced, which allow for the detection of effects if present. For the description of these methods the same variable nomenclature as in the tables 2.1 and 2.2 is used.

χ^2-Test The χ^2-test can be used to compare for two variables x and y, if their probability distributions are equal [121]. Thereby, x and y are called

Table 2.1: Generic example of a 2×3 contingency table.

	AA	Aa	aa	Total
Control	n_{00}	n_{01}	n_{02}	$N_{00} = n_{00} + n_{01} + n_{02}$
Case	n_{10}	n_{11}	n_{12}	$N_{01} = n_{10} + n_{11} + n_{12}$
Total	$N_{10} = n_{00} + n_{10}$	$N_{11} = n_{01} + n_{11}$	$N_{12} = n_{02} + n_{12}$	$N = N_{00} + N_{01} = N_{10}+N_{11}+N_{12}$

Table 2.2: Generic example of a 2×2 contingency table that has been constructed from a 2×3 contingency table as shown in table 2.1.

	A	a	Total
Control	$\tilde{n}_{00} = 2n_{00} + n_{01}$	$\tilde{n}_{01} = 2n_{02} + n_{01}$	$\tilde{N}_{00} = 2N_{00}$
Case	$\tilde{n}_{10} = 2n_{10} + n_{11}$	$\tilde{n}_{11} = 2n_{12} + n_{11}$	$\tilde{N}_{01} = 2N_{01}$
Total	$\tilde{N}_{10} = 2N_{10} + N_{11}$	$\tilde{N}_{11} = 2N_{12} + N_{11}$	$\tilde{N} = 2N$

independent, if the probability distribution of one variable is not affected by the other. For the application to contingency tables (i.e. categorical data), an observed distribution of counts is compared to an expected distribution. In case-control studies the distribution of allele frequencies in the case group is expected to be the same as for the control group under the null hypothesis H_0 of independence. To test for H_0, the sum of the squared difference between the observed counts and the expected counts (cases vs. controls) is calculated and divided by the expected counts. For n classes the corresponding Pearson's χ^2-test statistic with $n - 1$ degrees of freedom is defined as follows:

$$\chi^2 = \sum_{i=1}^{n} \frac{(O_i - E_i)^2}{E_i} \tag{2.4}$$

In this formula, O_i corresponds to the observed allele counts in the cases group and E_i to the expected allele counts from the controls group for the class i. The χ^2 test can be applied to 2×2 as well as 2×3 contingency tables. Under the null hypothesis of independence the test statistic follows the χ^2-distribution with $n - 1$ degrees of freedom, from which a p-value is computed.

Fisher's Exact Test The assumption that under H_0 the test statistic follows the χ^2-distribution only holds for large sample sizes. This means that the calculation of the p-value becomes exact only when the group sizes grow to infinity. Thus, when group sizes are low, an appropriate exact test should be used, such as Fisher's exact test [42]. Although it is often used when frequency counts are small, it is also valid for large values. According to Fisher's exact test for 2×2 contingency tables, the probability of observing the frequencies $\tilde{n}_{00}$, $\tilde{n}_{01}$, $\tilde{n}_{10}$, and $\tilde{n}_{11}$, as well as a total sample size of $\tilde{N}$ is given by the hypergeometric distribution:

$$p = \frac{\binom{\tilde{n}_{00}+\tilde{n}_{01}}{\tilde{n}_{00}}\binom{\tilde{n}_{10}+\tilde{n}_{11}}{\tilde{n}_{10}}}{\binom{\tilde{N}}{\tilde{n}_{00}+\tilde{n}_{10}}} \tag{2.5}$$

$$= \frac{\tilde{N}_{00}!\tilde{N}_{01}!\tilde{N}_{10}!\tilde{N}_{11}!}{\tilde{n}_{00}!\tilde{n}_{01}!\tilde{n}_{10}!\tilde{n}_{11}!\tilde{N}!} \tag{2.6}$$

With this formula, Fisher's exact test calculates the probability of obtaining the frequencies observed in the contingency table, as well as any configuration with a smaller probability of occurrence in the same direction (one-sided test) or in both directions (two-sided test).

Difference of Proportions The difference of proportions test allows one to test if there is a significant difference between two independent proportions. Thus, it is applicable to 2×2 contingency tables. For our application case, let P_1 denote the population proportion of the minor allele a in the case group and P_0 the proportion of a in the control group. The difference $D = P_1 - P_0$ then compares the two proportions. Clearly, if $D = 0$, then there is no difference between the case and the control group and consequently no association of the tested variation with the phenotype separating cases and controls. If however $P_1 > P_0$, then there is a positive association with the phenotype. D can be estimated by the differences in group proportions $\hat{d} = \hat{p_1} - \hat{p_0}$. For sufficiently large sample sizes, the sampling distribution of $\hat{d}$ is approximately normal with mean $D = P_1 - P_0$ and standard deviation (SD):

$$SD(D) = \sqrt{\frac{P(1-P)}{\tilde{N}}} \tag{2.7}$$

SD can be estimated by the pooled standard error:

$$SE(\hat{d}) = \sqrt{\frac{\hat{p}(1-\hat{p})}{\tilde{N}}} \tag{2.8}$$

In the formula above, $\hat{p} = \frac{\tilde{N}_{10}}{\tilde{N}}$ is the pooled sample proportion of the minor allele. The corresponding statistic to test the null hypothesis, can then be formulated as:

$$Z = \frac{\hat{p_1} - \hat{p_0}}{SE(\hat{d})} \tag{2.9}$$

For the one-sided test ($P_1 > P_0$), which looks for a positive association of the minor allele with the phenotype, the test statistic asymptotically follows a standard normal distribution and the p-value can be calculated as $p = Prob(N(0,1) \geq Z)$ [2]. For the two-sided test, which checks for an association of the genotype with the phenotype ($P_1 \neq P_0$) the test based on Z is equivalent to a test based on Z^2. Z^2, however, asymptotically follows a χ^2-distribution with one degree of freedom. Thus the p-value is given by $Prob(\chi_1^2 \geq Z^2)$ [2].

Relative Risk The test statistic based on the differences of proportions becomes inaccurate if the proportions are very small (close to zero). Thus, the relative risk method has been developed, which addresses this issue [2]. In contrast to the difference of proportions, the relative risk is defined as the ratio of the population proportions $P = \frac{P_1}{P_0}$. This can be estimated by the sample relative risk $\hat{p} = \frac{\hat{p}_1}{\hat{p}_0}$. As can be seen immediately, there is no association between the SNV and the phenotype if $p_1 = p_0$, i.e. $\hat{p} = 1$.

Since the distribution of $\hat{p}$ can become skewed for extreme values, usually the $\log \hat{p}$ is calculated instead. Under the null hypothesis that there is no association of the SNV with the phenotype ($P_1 = P_0$), one can define the estimated standard error SE:

$$SE(\log \hat{p}) = \sqrt{\frac{1-\hat{p}_1}{\tilde{N}_{01}\tilde{p}_1} + \frac{1-\hat{p}_0}{\tilde{N}_{00}\tilde{p}_0}} \tag{2.10}$$

The corresponding test statistic is given by:

$$Z = \frac{\log \hat{p}_1 - \log \hat{p}_0}{SE(\log \hat{p})} \tag{2.11}$$

As for the difference of proportions test under the null hypothesis, Z asymptotically follows a standard normal distribution. Hence, the p-value for the one-sided test of positive association of the minor allele with the phenotype ($P_1 > P_0$) can be calculated as $p = Prob(N(0,1) \geq Z)$. In case of the two-sided test, which looks for an association of the genotype with the phenotype ($P_1 \neq P_0$), the test based on Z is equivalent to a test based on Z^2, which asymptotically follows a χ^2 distribution with one degree of freedom [2]. Consequently the p-value for the two-sided test is given by $p = Prob(\chi_1^2 \geq Z^2)$.

Odds Ratio An odds ratio for case-control studies is another possible measure of association between genotypes and a specific phenotype of interest. This measure is very similar to the relative risk ratio, especially when the phenotype under consideration is rare in the population studied.
The odds ratio OR is defined as the ratio of the odds of the minor allele in the case group ($odds_{case}$) with the odds of the minor allele in the control group ($odds_{control}$):

$$odds_{case} = \frac{P_1}{1 - P_1} \tag{2.12}$$

$$odds_{control} = \frac{P_0}{1 - P_0} \tag{2.13}$$

$$OR = \frac{odds_{case}}{odds_{control}} = \frac{\frac{P_1}{1-P_1}}{\frac{P_0}{1-P_0}} \tag{2.14}$$

In the formulas above, P_1 is the population based frequency of the minor allele in the case group and P_0 the population based frequency of the minor allele in the control group. These can be estimated by the sample frequencies $\hat{p}_1$ and $\hat{p}_0$, respectively. According to the definition of the OR, a value of 1 means that there is no association of the minor allele with the phenotype and for $OR > 1$ there is a positive association. As for the relative risk, usually the $\log OR$ is considered, to avoid skewness of the respective distribution [2]. Under the null hypothesis of no association with the phenotype ($P_1 = P_0$) the estimated standard error can be defined as:

$$SE(\log OR) = \sqrt{\frac{1}{\tilde{n}_{00}} + \frac{1}{\tilde{n}_{01}} + \frac{1}{\tilde{n}_{10}} + \frac{1}{\tilde{n}_{11}}} \tag{2.15}$$

The corresponding test statistic is then given by:

$$Z = \frac{\log OR}{SE(\log OR)} \tag{2.16}$$

As for the *Difference of Proportions* and *Relative Risk* statistic, the p-values for the one-sided ($P_1 > P_0$) and two-sided ($P_1 \neq P_0$) alternatives are given by $Prob(N(0,1) \geq Z)$ and $Prob(\chi_1^2 \geq Z^2)$, respectively [2].

Although the interpretation of the odds ratio is much more difficult than for the relative risk ratio, it is still common practice to calculate the odds ratio in typical case-control studies. The reason is that it has the advantage that it can be calculated even when the number of cases and controls is fixed by the study design. In such studies, the calculation of the relative risk is not meaningful, since changing the ratio of cases to controls, would also change the relative risk. Furthermore, for rare phenotypes, the odds ratio is a good approximation of the relative risk.

Population Stratification Although case-control studies have the potential to reveal associations between SNVs and phenotypic traits, they rely strongly on the composition of the case and control group. If there are large variations in the population on the genetic level, one speaks of population stratification [122]. In such cases, subgroups of genotypes with different allele frequencies within a group can be build. These subgroups usually also differ from the rest of the population with respect to the phenotype investigated in the case-control study. The main reason for population stratification is migration, but also other alternative explanations exist. It has often been observed that a migrated group, which is, for example, susceptible to a particular disease, has become part of a larger population [158]. Since the detection of disease related associations in case-control studies largely depends on the homogeneity of the case and control group with respect to the underlying allele frequencies, falsely deduced associations can result from unrelated subgroups within one of the groups. Further reasons for population stratification are related to the nature of the study. There may be participants with unknown ethical backgrounds, which, when unintentionally pooled together, may induce an association with a disease. However, when the participants are split up with respect to their backgrounds, the association may no longer be present. In such cases, methods are needed that are able to cope with population stratification. Armitage's trend test, which will be described in the following, is one example of a statistical test that is able to address population stratification.

Armitage's Trend Test In case-control studies, the Armitage trend test [6, 139, 141] is used to assess disease related association based on 2×3 contingency tables. In contrast to the χ^2-test of independence, the Armitage trend test is able to address the problem of population stratification. It is a modification of the χ^2-test, where assumptions on the genotypes, for example, based on a specific genotypic model, can be introduced. For biallelic organisms the test itself works as follows. Based on the respective genotype of an individual and its affiliation with the case or control group, a linear regression model can be defined. Let y be the variable for the individual's allelic combination. Thereby, $y = 2$ if the individual is homozygous (AA), $y = 1$, if the individual is heterozygous (Aa), and $y = 0$ if he is homozygous with respect to the minor allele (aa). Furthermore, let x be the phenotype variable that defines whether the individual belongs to the case group $(x = 1)$, or to the control group $(x = 0)$. The corresponding linear regression model is then described as:

$$y = \beta_0 + \beta_1 \cdot x + \epsilon \tag{2.17}$$

Based on the null hypothesis $H_0 : \beta_1 = 0$ and the respective alternative hypothesis $H_1 : \beta_1 \neq 0$ the Armitage trend test statistic is defined as follows:

$$A_r = \frac{\hat{\beta}_1^2}{VAR(\hat{\beta}_1)} = Nr_{xy}^2 \tag{2.18}$$

In this formula, r_{xy}^2 is the squared correlation between the genotype variable y and the phenotype variable x. For the variance estimation for y the sum of the squared deviations of y from the fitted values is calculated. Under the null hypothesis, the Armitage trend statistic A_r will approximately follow a χ^2 distribution with one degree of freedom. Thus, the corresponding p-value is given by $Prob(\chi_1^2 \geq Nr_{xy}^2)$.

Hardy-Weinberg Equilibrium In 1908 G. H. Hardy and W. Weinberg independently proposed a theory of how genotype and allele frequencies for diploid organisms behave across different generations in a population. This principle later became known as the Hardy-Weinberg principal [152]. It is based on seven assumptions:

1. the organisms in the population are diploid
2. only sexual reproduction occurs
3. the different generations within a population do not overlap
4. mating is completely random

5. the population is infinitely large
6. the allele frequencies are equal between the sexes
7. there is no significant external impact acting on the population, such as migration, mutation or selection

Under these constraints, the principle describes that genotype and allele frequencies do not change with different generations in the population. Moreover, they remain constant, if the constraints are not violated by any significant external impact. If we consider a SNV for a diploid organism at a given locus with the alleles A and a, and allele frequencies p and q, respectively, then one can easily compute the frequencies for each possible genotype as:

$$f_{AA} = p^2 \tag{2.19}$$
$$f_{Aa} = 2pq \tag{2.20}$$
$$f_{aa} = q^2 \tag{2.21}$$

Clearly, the sum of the frequencies of the two alleles A and a must be 1, i.e. $p+q = 1$. With this, a relation between the three different genotype frequencies can be described, which leads to the Hardy-Weinberg-Equilibrium (HWE):

$$p^2 + 2pq + q^2 = (p+q)^2 = 1 \tag{2.22}$$

This equation can be used to measure, if observed genotype frequencies in a population differ from the theoretical frequency distribution. In real populations, however, the HWE will not be followed strictly, but the model is robust to deviations to some extent. Thus, one can apply statistical tests, in order to identify significant deviations from the predicted allele frequencies. Any statistically significant deviation is a strong indication that either genotyping errors exist, or, if these can be excluded, that some biologically relevant factor acts on the population. A common application of the HWE in GWAS is to check for deviations from HWE in a control group, where no biological effect is expected. There, deviations from HWE are most likely due to genotyping errors. Consequently, such SNVs are usually removed prior to subsequent analyses.

2.5.3 Multiple Testing Correction

For GWAS, thousands of different variants are studied in parallel and statistical testing is applied to each of the variants separately. Thus, it is crucial to correct for multiple testing, in order to control the type I error, also called the *false positive rate* (FPR). The FPR corresponds to the probability of rejecting

the null hypothesis when it is true. Usually, a significance level α is provided in statistical testing, which indicates the proportion of false positives an investigator is willing to tolerate [173]. Thus, when applying multiple simultaneous statistical tests, it is important to restrict the *family-wise error rate* (FWER), which is the probability of observing one or more type I errors. Restriction of the FWER reduces the number of type I errors, but consequently increases the type II error rate (also known as *false negative rate* (FNR)), which is the probability of maintaining the null hypothesis when it should have been rejected. Thus, an increased FNR reduces the power to detect significant associations. Consequently, a suitable trade-off between FPR and FNR has to be specified in each study involving multiple statistical testing [173]. In order to control the FWER, it is important to keep track of the number of different statistical tests and, in addition, to correct the SNV specific significance thresholds based on the number of tests performed. A very simple correction method is given by the Bonferroni correction [39] (see equation 2.23), or the Sidak correction [145] (see equation 2.24), which adjust the significance level α by the total number of tests n.

$$\alpha^* = \frac{\alpha}{n} \tag{2.23}$$

$$\alpha^* = 1 - (1 - \alpha)^{\frac{1}{n}} \tag{2.24}$$

Thresholds based on the adjusted significance level α^* are then applied to identify significant differences in SNV allele frequencies between subject cohorts. These correction methods work well, if the underlying statistical tests are independent. However, in typical GWAS, where SNVs are often located in close proximity to each other, independence can rarely be presumed. In these cases both corrections are very conservative. An alternative approach that is not based on the FWER, is the *false discovery rate* (FDR) [12], which controls for the proportion of expected false-positives. However, this approach also suffers from SNV dependency issues and is therefore not recommended for GWAS. Currently, the *gold standard* for multiple testing correction are permutation based approaches [82], where original p-values are compared to empirical ones, obtained by randomization of the observed case-control labels and repetition of the respective statistical test. Nevertheless, permutation based methods share the disadvantage of being computationally very expensive, requiring either powerful workstations for the calculations, or a preliminary filtering of the SNVs before statistical testing to reduce the total number of tests and consequently the overall number of permutations for p-value correction.

2.5.4 Multi-Locus Analysis

Besides the analysis of single-locus associations, such as the association of a SNV with a disease phenotype, so-called multi-locus association analyses focus on the identification of interactions between genetic variants. The phenomenon in which an interaction between two and more SNVs is needed in order to induce a phenotypic modification, is called *epistasis*. In contrast to single-locus association analyses, the identification and analysis of multi-locus associations bears a couple of challenges. In a typical GWAS, several thousand SNVs are analyzed in parallel. If we only concentrate on pair-wise SNV interactions, we already end up with approximately $10^{10} - 10^{14}$ SNV pairs that have to be considered in a typical study. This leads to enormous computational hindrances as well as to increased multiple hypothesis testing problems [107]. Approaches that address these issues mostly concentrate on a reduction of the overall comparisons that have to be made. Filtering is a common strategy, where SNVs are, for example, selected based on results from a prior single-locus analysis. Interactions are then only evaluated within the subset of selected SNVs. However, this disregards purely epistatic interactions, where single SNVs do not have any statistically detectable effect, and only the combination of the SNVs causes a significant change in the phenotype. With the development of more and more powerful graphics cards that contain several hundred *graphics processing units* (GPUs), attempts have been taken in the direction of parallelization rather than SNV filtering. Algorithms that purely operate on the GPUs have shown to be very effective in reducing the overall runtime and are currently the method of choice for epistatic analyses. Computational cluster systems could also be used, however, due to the relative low computational effort needed to conduct a single test, GPUs are usually preferred. Nevertheless, the systematic analysis of SNV triplets or even larger units still remains computationally infeasible. Consequently, most GWA studies today concentrate on two-locus associations only.

A popular method for the analysis of SNV pair associations is multiple logistic regression. This method is an adaption of linear regression, where in addition a logit transformation is incorporated in the procedure. This allows for the analysis of binary information, as it is the case in standard case-control studies. The following formula shows a typical logit function. In this function, p corresponds to the probability of having a disease. Furthermore, β coefficients are used to describe the individual SNV effect as well as the effect introduced by their interaction. β_0 represents the intercept of the underlying regression model, β_1 and β_2 correspond to the main effect of each of the SNVs from the SNV pair, and β_3 represents the interaction term. The genotype information for the two SNVs is described by the variables x_1 and x_2, respectively. These can be encoded in a number of different ways. In this case, the numbers -1,

0, and 1 are used with respect to an underlying genotypic model (see section 2.5.2). Furthermore, the interaction term $x_1 \times x_2$ can also be encoded in various ways depending again on the genotpic model.

$$logit(p) = \ln \frac{p}{p-1} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 (x_1 \times x_2) \tag{2.25}$$

Stepwise logistic regression further allows to investigate whether two SNVs have independent effects on a trait, or if they are in linkage disequilibrium with each other. The latter would for example be the case when there is evidence of an association, but a missing improvement in the model fit when both SNVs are included [89]. An alternative to this approach is phasing the genotypes into haplotypes, if informative relationship data for the individuals in the study are available. The haplotypes are then used as a unit for the analysis, rather then testing each individual SNV separately. This method is especially useful, because a haplotype represents a functional unit of a gene (see section 2.5.1 for a detailed explanation of genotypes and haplotypes).

2.6 Quantitative Trait Locus

The methods described so far are suitable for the analysis of binary traits. However, more complex phenotypes, such as blood pressure, height, or obesity cannot be characterized sufficiently using a binary description. Thus, quantitative measures are needed to extensively characterize these traits, rather than qualitative ones. Such traits are usually affected by many different genes and environmental conditions. Regions on chromosomes that show a significant association with a quantitative trait have become known as *Quantitative Trait Loci* (QTL). These regions can vary largely in size, ranging from a single marker up to one or more different genes. Thus, a QTL does not necessarily have to contain a SNV, it is simply defined as the region responsible for the respective phenotype. The identification of these regions (also known as *QTL mapping*) is thus of great importance.

2.6.1 QTL Mapping

The simplest method for QTL mapping is the analysis of variance (ANOVA) for each SNV marker. This method is also known as *marker regression* [19]. To perform such an analysis for a specific SNV, individuals are grouped according to their genotype at the SNV locus. Based on these groups and the distribution of the quantitative phenotype values within each group a traditional t-test, or an ANOVA can be performed. The goal is to find significant differences between the groups under the null hypothesis of no difference in

trait means for any of the genotype groups. This step is then repeated for each SNV in the data set. However, in order to perform an ANOVA, several different assumptions have to be made. The underlying test statistic of the ANOVA requires the quantitative trait to be normally distributed and the variance of the trait values within each genotype group to be the same, i.e. the groups have to be homoskedastic. Furthermore, the groups have to be independent. These requirements and especially the latter one can, however, not always be fulfilled. In addition, individuals whose genotype is missing at a specific locus have to be excluded. Furthermore, the power of the QTL detection largely depends on the density of the SNVs. If these are widely spread, then the QTL may be far apart from them, which leads to a decrease in power.

To overcome the mentioned hindrances, Lander and Botstein developed the *interval mapping* (IM) approach [85]. This approach can be applied for the identification of any QTL that is flanked by two different markers. Consequently, each position between the markers in the genome is scanned consecutively. The IM algorithm assumes a mixture model of normal distributions of the quantitative phenotype at the putative QTL locus. Thereby, the distribution of the phenotype at the QTL is constituted by the distributions of respective flanking SNVs assuming equal variance. Variables for the different distributions are estimated using a maximum likelihood approach applying an EM (expectation maximization) algorithm. The strength of evidence is then measured with the so-called LOD (logarithm of the odds) score (see equation 2.26). This is the $\log_{10}$ likelihood ratio of the alternative hypothesis of QTL presence at a position γ versus the null hypothesis of no QTL at that position:

$$LOD(\gamma) = \log_{10} \frac{Pr(y \mid QTL \text{ at } \gamma, \hat{\mu_{0\gamma}}, \hat{\mu_{1\gamma}}, \hat{\sigma_\gamma})}{Pr(y \mid no\ QTL, \hat{\mu}, \hat{\sigma})} \tag{2.26}$$

The values $\hat{\mu_{0\gamma}}$, $\hat{\mu_{1\gamma}}$, and $\hat{\sigma_\gamma}$ are the maximum likelihood estimates of the respective SNV distributions for the putative QTL position γ. y corresponds to the observed phenotype data with the assumption that $y \sim N(\mu, \sigma)$. Under the *no QTL* model, the phenotypes are independent and identically distributed. The higher the LOD score for a particular genome position, the more likely is the presence of a quantitative trait. The null hypothesis is thus rejected if $LOD(\gamma)$ exceeds a predefined threshold. Furthermore, due to the maximum likelihood estimation procedure, the LOD score is asymptotically distributed as $\frac{1}{2}(\log_{10} e)\chi^2$, with χ^2 being the χ^2-distribution with 1 degree of freedom [85]. The corresponding test statistic is then given as:

$$T = 2 \ln \frac{Pr(y \mid QTL \text{ at } \gamma, \hat{\mu_{0\gamma}}, \hat{\mu_{1\gamma}}, \hat{\sigma_\gamma})}{Pr(y \mid no\ QTL, \hat{\mu}, \hat{\sigma})} \tag{2.27}$$

Thus, statistical testing can be applied, in order to provide additional statistical significance of QTL presence. This procedure is known as the *likelihood ratio test*. Together with the *Wald test*, which is a less computationally intense approximation of the *likelihood ratio test* and explained in detail in [164], this procedure is currently considered the *gold standard* for QTL mapping.

2.6.2 Expression Quantitative Trait Locus

Complex human diseases are the result of the interplay of various different genes. As described above, variations within these genes or in close proximity to them (*cis*-acting SNVs), but also variations at greater distances (*trans*-acting SNVs), can have severe effects. However, a direct modification of the gene is not always required. Also changes in the expression levels of genes can have severe effects. These can, for example, be caused by mutations in molecules interacting with the affected genes. Furthermore, the expression level of a gene can be interpreted as a quantitative trait, allowing the application of methods from traditional QTL association analyses. In these so-called eQTL (expression Quantitative Trait Locus) studies not a single quantitative trait is investigated, but the expression levels of several hundred or thousand genes are studied at the same time. Thus, associations have to be made between all pairwise combinations of SNVs and genes in the study, leading to a much more complex and intense study design. In addition, clinical phenotypes, such as susceptibility to a disease, are often included. Despite the computational and statistical challenges, eQTL studies provide a much more comprehensive view of the underlying effects of variations. Thus, they are perfectly suited for integrative investigations of differential expression as well as variation analyses. Software solutions such as PLINK [129], an open-source whole genome association analysis toolkit, allow investigators to perform the necessary statistics for the identification of single- and multi-locus associations within a GWAS as well as eQTL study. However, visual assessment of the results is usually not provided, which is needed to increase interpretability of the results. Furthermore, well designed visualizations allow for a quick detection of interesting patterns in the data. Therefore, in this work, powerful and comprehensive visual analytics methods have been developed to tackle the need for improved ways of interpreting GWAS and eQTL data (see chapters 4 and 5).

2.7 Structural Variations

Structural variations lead to changes in the chromosomes of an organism on a large scale. These variations can be separated from small variations, such as SNVs, as well as small insertions and deletions, due to the number of bases that are affected. Typical sequence lengths range from about 1 kilobase up to 3 megabases. However, these definitions are rather ambiguous and sometimes much smaller or larger changes are also included.

Structural variations can be separated into five different classes. Insertions are modifications, where additional sequence information is incorporated into the genome. Possible sources are, for instance, transposable elements, which are commonly used by viruses, in order to survive in the host cells. Deletions on the other hand lead to a loss of sequence information and are often the result of chromosomal recombination or separation defects during meiosis. Duplications lead to an increased amount of sequence information for the duplicated region. These can also be the result of chromosomal recombination and are therefore considered as the counter part of deletions. The fourth class of structural variations are inversions. Here, sequence information is flipped within the genome, such that sequence parts of a chromosome, for example, are in inverse order in comparison to a reference. These can occur during replication of the chromosomes during cell division by random formation of DNA loops. Last but not least, translocations lead to an exchange of sequence information within a genome. One distinguishes between inter- and intra-chromosomal translocations, whether only a single chromosome, or different chromosomes are involved. Reasons are, for example, crossing-overs during cell division. Furthermore, inversions and translocations can be distinguished from the other classes, since these two do not change the information content, i.e. there is neither a gain, nor a loss of sequence information.

Although the definition of structural variations does not imply any phenotypic effect, many are associated with genetic disease in humans and other animals. Prominent examples can be found in clinical diagnostics, where for instance deletions of large parts of the genome lead to mental retardation or cancer [132].

In contrast to small variations, where the application of NGS can easily detect changes in the nucleotide sequence, since these are smaller than typical sequencing read lengths, structural variations are much harder to identify. While some structural variations, such as deletions or duplications, can be identified due to a change in the mean read coverage of the affected chromosomal region, others such as translocations can only be identified, if enough reads cover the initial sites of fracture. Nevertheless, attempts have been taken to simplify the process of structural variation detection. Using

paired-end sequencing, for example, allows for the analysis of abnormally mapped read pairs. For these read pairs, the respective forward and reverse read could not be mapped with the overall average insert size to a predefined reference genome. This can have several reasons, most of them linked to structural variation, if sequence quality issues can be excluded. For instance, if either the forward read, or the reverse read could not be mapped at all, this is an indication of a potential deletion in the region, where the respective read would have been expected to map. Read pairs showing large deviations with respect to the average insert size, on the other hand, indicate an insertion, duplication or inter-chromosomal translocation event. Intra-chromosomal events further imply that read pairs map to different chromosomes. Coverage analysis and detailed determination of the mapping position further helps to differentiate between these variations. When both read pairs map in the same direction, instead of opposite to each other, this indicates an inversion. Although the consideration of the different arrangements of read pairs allows for the detection of structural variations, this approach is usually limited by the size of the respective variations. Variations that are larger than the average insert size of read pairs by several orders of magnitude can usually not be detected sufficiently. Mate-pair sequencing [66], for instance, displays an attempt to address this issue, since read-pairs with very large insert sizes can be created. However, due to the complex library preparation process and the increased sequencing costs, this approach is not feasible in most cases.

However, for the study of small genomes, as for example bacteria, paired-end sequencing is very effective. These genomes usually experience various different structural rearrangements. This can be explained by the usually high mutation rates, as well as the process of horizontal gene transfer, where parts of the bacterial genomes are exchanged between different bacterial strains or species. Often such elements contain genes that are responsible for pathogenicity or drug resistance, which makes them an attractive research topic. Thus, methods for the analysis and visualization of structural variations are an important instrument to firstly unravel structural changes and secondly to help in the interpretation of their impact. In this thesis, a comparative visual analytics approach for the identification and characterization of structural variations has been described. Details on this method can be found in chapter 6.

3. MAYDAY - An Interactive Visual Analytics Workbench

In this thesis, visual analytics approaches for variation data and the connection with expression data are introduced. For this purpose MAYDAY [11] was chosen as a general framework for the development of new analysis and visualization approaches for variation and expression data, as well as the combination of both. MAYDAY, short for Microarray Data Analysis, is an expression analysis software that focuses on visual data exploration. Its development started in 2004 by Kay Nieselt, Janko Dietsch, and Nils Gehlenborg to address the need for a freely available application to study microarray expression data. With the contributions of Florian Battke during his PhD thesis [9], MAYDAY became a platform for the integrative study of transcriptomics data with respect to the implemented processing methods, the underlying data structures, as well as visualizations. A major contribution was the MAYDAY Seasight extension [10], which enables the integration of RNA-seq expression data and provides appropriate methods to conduct a complete RNA-seq data analysis workflow starting from mapped sequencing reads (see section 2.4.3). With this extensions, a first step was taken towards the integration of data from next-generation sequencing systems. Due to the generic design of the underlying data structures, storage of expression data is independent of the respective source (i.e. microarrays or RNA-seq). This allows for the application of analysis methods without the need to distinguish between microarray and RNA-seq expression values. In addition, MAYDAY possesses a powerful visualization framework, offering different plots for visual data exploration. The visualizations are thereby highly interactive and can be linked to each other. As a result, MAYDAY enables the realization of visual analytical techniques (see section 2.2) to gain better insights into data. Thus, it is perfectly suited for the development of new visualization and analysis strategies including other kinds of data, that have not been addressed so far, in particular single nucleotide variations.

However, until the beginning of this work, no features for the integration of other kinds of data, except expression data, were available. For the extension of MAYDAY with data analysis methods for RNA-seq introduced with the MAYDAY Seasight plugin, data structures available for microarray gene expression data could be used without larger modifications. This means that read counts are interpreted as gene expression values. For RNA-seq data, this was possible, since there are only minor differences in the way RNA-seq expression data is processed in comparison to microarray expression data. This does,

however, not apply for other kinds of data, such as single nucleotide variation data. Studies linking single nucleotide variations to disease (so-called genome wide association (GWA) studies), or expression data to variations (known as eQTL - expression Quantitative Trait Loci - studies, that will be addressed in more detail in chapter 5) offer new insights into gene regulatory mechanisms and are therefore highly useful in making well informed interpretations.

Thus, this chapter provides details on the existing data structures in MAYDAY and information about modifications necessary to extend MAYDAY for the visual analytical study of GWA and eQTL data. This includes a description of how the data model has been extended to allow for the integration of appropriate data structures for non-expression data, as well as details on the changes made to the visualization framework, in order to allow for new visualization approaches that are capable to display expression data together with variation data. This chapter is therefore structured into two parts. In the first part, the main components of the MAYDAY software are introduced, including the MAYDAY data model with its various data structures for expression and meta-data. Furthermore, details are provided for the MAYDAY visualization framework and on the plot generation system, introducing the elements needed to create new visualizations in MAYDAY. The second part focuses on the modifications that were necessary to include other kinds of data, specifically variation data, into the existing MAYDAY data structures. Here, a general approach is described that allows the integration of variation data in the form of new plugins, such that the original data structures do not have to be changed.

3.1 The Basic Data Structures in MAYDAY

Gene expression data is usually organized as an expression matrix, where columns represent different experiments or conditions and rows represent genes. Each cell of this matrix consequently contains an expression value of a specific pair of gene and experiment. In a microarray study, columns usually show different microarray experiments and rows display the probes used on the microarray. Thereby, probes are small DNA, cDNA, or oligonucleotide molecules with known identity that are used to identify genes of interest. Clearly, a single gene can be represented by one or more probes and summarization methods, such as taking the mean expression value of the probes, are usually used to calculate the expression value for each respective gene (see section 2.3.5).

In MAYDAY, expression data is organized in a similar way. There are three major data types used to organize expression values. The `MasterTable` is a

matrix containing expression values for pairs of `Experiments` and `Probes`. As in an ordinary expression matrix, `Experiments` are represented by columns in the `MasterTable`, whereas rows represent `Probes`. `Probes`, as well as `Experiments` hold references to their respective expression values in the `MasterTable`, such that access to its entries is either possible directly from the `MasterTable`, or indirectly using `Probe` or `Experiment` instances. Besides these three data types, further data structures are needed that allow for convenient data handling. An often required feature is the possibility to concentrate only on a subset of the probes in the data set. This allows for the application of analysis methods or visualizations to a set of probes that are of specific interest. An example would be the selection of a subset of probes based on a statistical test, where only those considered significant are used for further processing. MAYDAY offers the possibility to organize probes in groups, called `ProbeLists`. These `ProbeLists` contain references to `Probe` instances. Clearly, a single `Probe` can be contained in one or more `ProbeLists`.

In addition to these basic data structures, meta-information can be included, such as gene annotations (gene names, KEGG pathways, GO terms, and many more). This meta-information is represented by so-called meta-information objects (`MIOs`), which are associated with a specific MAYDAY data type (e.g. a `Probe`, an `Experiment`, a `ProbeList`, etc.). Such `MIOs` can either represent strings, integers, doubles, or even more complex data types, such as genomic locations. `MIOs` that are associated with the same MAYDAY data type can further be grouped into so-called `MIOGroups` that allow for a hierarchical structuring. All these data structures are contained in a `DataSet`, which is the main data container for expression data. Figure 3.1 shows these described relationships.

For some of the basic data structures extended versions exist that provide additional functionality. For instance, one such functionality is filtering. When studying high-throughput data, the number of probes can become large (up to ten thousands). Filtering can help to reduce data complexity, either for visualization purposes or for the analysis of the respective features. A typical scenario is the selection of only those probes that are differentially expressed between two or more conditions. To address this, an extended version of the `ProbeList`, a so-called `DynamicProbeList`, is available. Within this class, specific filter objects can be defined and structured in a hierarchical way. This allows to combine simple filters in order to construct complex ones. The filters themselves are implemented as `DataProcessors`, which can be concatenated to build filter chains. Each `DataProcessor` converts a specific input data type into an output data type that is defined by the respective `DataProcessor`. However, the final `DataProcessor` in such a filter chain, has to transform its input data type into a boolean value, which can be evaluated. An example

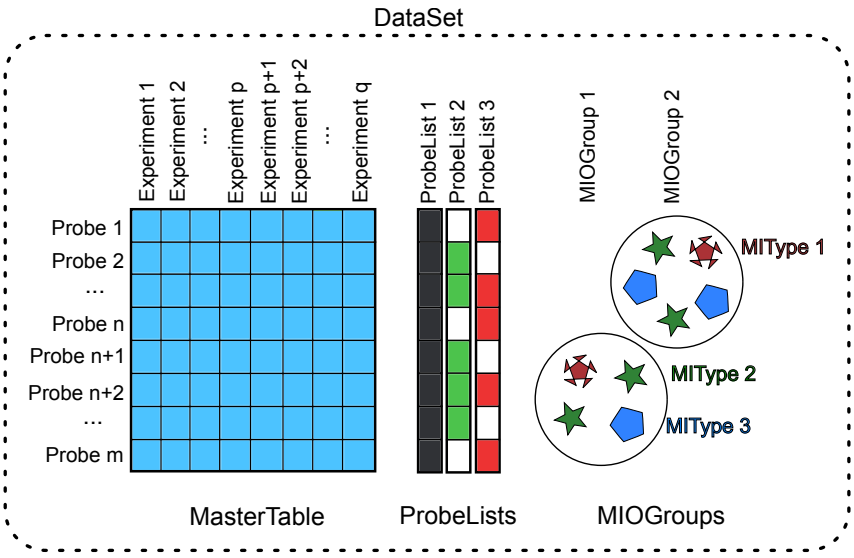

Figure 3.1: Overview of MAYDAY's basic data structures, namely the `MasterTable` containing the expression values, `ProbeLists` representing subsets of probes, as well as `MIOGroups` that can contain meta-information objects of different types.

would be a fold-change filter that allows to select all probes with an absolute fold-change larger than 1. To construct such a filter, first a meta-information value `DataProcessor` has to be chosen, which selects the respective fold-change meta-information values. Thus, this `DataProcessor` internally replaces the `Probe` objects with the corresponding fold-change values, which are then passed on to the next `DataProcessor`. Since absolute fold-change values should be used for filtering, a second `DataProcessor` has to transform the fold-change values into absolute values, before a third `DataProcessor` can apply a simple comparison with the value 1 resulting in a boolean value that is returned. Based on this value, the `Probe` is inserted into the `DynamicProbeList`, or rejected. Due to its dynamic design, a `DynamicProbeList` can rely on other `ProbeLists` or even other `DynamicProbeLists`. This is realized by `DataProcessors` that evaluate if a `Probe` is contained in a specific user-defined `ProbeList`. Consequently, removing a `Probe` from that `ProbeList` automatically leads to the removal of the respective `Probe` from the `DynamicProbeList`.

3.2 MAYDAY's Visualization Framework

As previously described (see section 2.1), visualizations are extremely useful to understand complex data. MAYDAY has therefore been built with a large focus on data visualization and visual data exploration. The basis is a powerful visualization framework. In the following, the data structures of this

framework as of MAYDAY 2.12 are described.

The `ViewModel` is the major interface between the data contained in the `DataSet` and the visualizations. It holds references to the `ProbeLists` it was created from, as well as to so-called *top priority* `ProbeLists`. These are disjoint representations of the original `ProbeLists`. If a `Probe` is, for example, contained in more than one `ProbeList` this would consequently lead to a multiple rendering of the respective `Probe` for data visualization. *Top priority* `ProbeLists` provide an optimization of this circumstance, since each `Probe` is assigned only to the first `ProbeLists` it is contained in. Thus, when rendering the *top priority* `ProbeLists`, each `Probe` is rendered exactly once.

Moreover, all plots in MAYDAY need access to the `DataSet`, in order to request the values needed for visualization. However, the data is not passed directly from the `DataSet` to the visualization, but access is granted via the `ViewModel`. This additional layer allows for live data transformations, just right before visualization. These data transformations are implemented using so-called `DataManipulators`. If a plot requests a data value from the `ViewModel` for visualization, this value is first passed through a `DataManipulator`. There transformations, such as z-score transformation, centering, scaling, or logarithmic transformation to a user-defined base, are performed and the manipulated data value is then passed on to the requesting plot. This strategy leaves the data in the `DataSet` unchanged, offering the possibility to apply different manipulations in separate visualizations simultaneously. Meta-information is handled similarly. Since meta-information can be of various different data types, `ValueProviders` are used that transform the data into a format that can be interpreted by the respective plot. For instance, a `ColorProvider` is a subtype of the `ValueProvider` and is used to transform meta-information values into color values, which can then be used to enhance a visualization by adding color to the data points.

Furthermore, data manipulations, as well as element selections can be shared between different plots. This is realized by `Visualizers` that are affiliated with the plot. All plots that are opened with the same `Visualizer` are synchronized with respect to data manipulations and selections. In fact, changes in the selection state of specific elements in one plot are displayed in all of the linked plots simultaneously. This is achieved through the registration of each plot to a specific `Visualizer` as a so-called `VisualizerMember`. This is an interface that is used by the observing `Visualizer` to organize the connected plots. For a more detailed description of how this connection is implemented see [9].

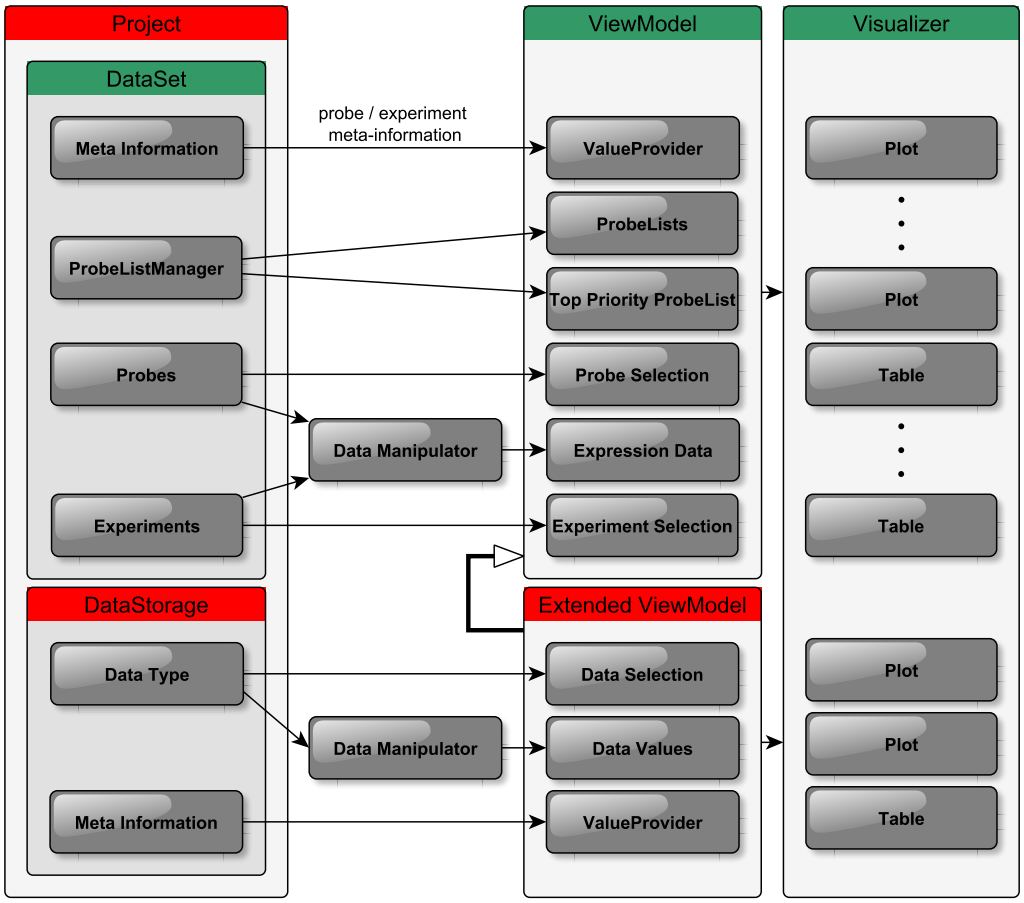

Figure 3.2: Overview of the main components of the MAYDAY visualization framework. Green components correspond to original MAYDAY structures as of version 2.12. Red components have been added in this work to enable visualization of non-expression data. Arrows highlight the possible ways of data being transfered from the data level on the left to the visualization level on the right. Parts of this image are based on Florian Battke's original `ViewModel` illustration in [9], page 44.

Since MAYDAY has been developed with a main focus on expression data analysis, the `ViewModel` in MAYDAY version 2.12 could only handle data types necessary for expression data visualization, including `Probes`, `ProbeLists`, `Experiments`, as well as meta-information objects (`MIOs`). This implies that the available manipulations can only be applied to these data types. Therefore, an adjustment strategy is presented in the following section that addresses the need for an advanced data integration approach. Figure 3.2 gives an overview of the main components of MAYDAY's visualization framework and their relations. Furthermore, red boxes highlight additional elements required for the integration of data types different from expression data.

3.3 Extension of MAYDAY's Visualization Framework beyond Expression Data

Due to its powerful visualization framework, MAYDAY provides a reliable basis for the development of new visualization approaches. In this work, the initial purpose of expression data visualization and analysis has been extended to variation data. However, due to their diverse properties, the integration into MAYDAY could not be performed without larger extensions. Especially, new data structures as well as mechanisms to link variation to expression data had to be developed. The necessary modifications involved MAYDAY's visualization framework, as well as the `DataSet` as the general data container. As mentioned above, the exchange of data was handled mainly by the `ViewModel`. However, in MAYDAY version 2.12 only data structures relevant for gene expression data were supported. In particular, data values from `Probes`, `Experiments`, and associated meta-information objects (`MIOs`) could be processed, as shown in figure 3.2. These circumstances implied an extension of the current `ViewModel` to allow for the integration of other kinds of data and for connecting these with the already existing ones. Although, in this work the main focus lies on the integration and visualization of variation data, a general concept has been devised that can easily be adapted to other data sources.

First of all, a new data container has been defined that is capable of handling new data objects (organized in a `DataStorage`), as well as the original MAYDAY `DataSet`. For this purpose, the data container `Project` has been defined. A `Project` holds a reference to the `DataSet` and in addition can contain one or more `DataStorage` instances for different non-expression data sources. To connect the `DataStorage` and the `ViewModel`, a new abstract class `ExtendedViewModel` has been implemented, that inherits all functionality from the original `ViewModel`. This strategy offers a couple of advantages over a direct modification of the original `ViewModel`. Firstly, all functionality that is provided by the original `ViewModel` remains unchanged, which ensures the applicability of the `ViewModel` to all available MAYDAY plugins. In addition, there is a clear separation between the original functionality and the new features offered by a concrete implementation of the `ExtendedViewModel`. An example would be the handling of selections of non-expression data objects, which is not required by the original `ViewModel`. If, however, code of the original `ViewModel` needs to be changed, it can simply be overwritten in the `ExtendedViewModel`, which does not affect the functionality of the source code in the original `ViewModel` implementation. Thus, plugins relying on the functionality of the original `ViewModel` implementation remain intact at all times. Last but not least, if changes in the original `ViewModel` have to be made, these are also inherited by the `ExtendedViewModel`, such that

no programming efforts are required to add the same functionality to the `ExtendedViewModel`. Furthermore, visualizations relying on a specific type of `ViewModel`, either the original implementation or an extended version, simply need to request the correct type from the `Project` container during their initialization. In order to handle concrete data types with an abstract `ExtendedViewModel`, a defined implementation of the `ExtendedViewModel` is required. An example is provided by the REVEAL application developed in this thesis (see chapter 5), which uses a `RevealViewModel` that inherits the `ExtendedViewModel` and provides concrete implementations of the abstract methods suggested for non-expression data management.

Lastly, interactions with the new data types have to be handled. In Java, this is usually done using so-called `Events`. Special `Listener` classes are able to process `Event` objects and to react accordingly. In the original `ViewModel` implementation so-called `ViewModelEvents` have been used to exchange information about user interaction, as for example element selection in the visualizations. To be able to exchange events that are specific to the `ExtendedViewModel`, `ExtendedViewModelEvents` have been introduced. Similar to the `ExtendedViewModel`, the `ExtendedViewModelEvents` inherit the original functionality from the `ViewModelEvent` class and add additional information types needed only by the `ExtendedViewModel`. With this strategy, applications that work with the `ExtendedViewModel` are also able to react properly to events that are the result of the original `ViewModel`, but not the other way round. This means that the `ExtendedViewModel` is able to handle `ViewModelEvents` as well as `ExtendedViewModelEvents`, while the original `ViewModel` still only handles `ViewModelEvents`.

By following these simple guidelines for the enhancement of the original visualization framework with new data types, extensions in the form of new plugins can easily be added without the need to change any of the existing functionality. Hence, compatibility to already existing MAYDAY plugins is ensured without the need to introduce changes in these. An overview of the general design of this extension strategy is provided in figure 3.2 highlighted by red boxes. One additional class that has to be mentioned in this context, is the `Visualizer`. As described above, `Visualizer` objects are used to enable the synchronization between different plots. Since the original `ViewModel` stays intact, the interaction with the `Visualizer` is not influenced by extended versions of the `ViewModel`. Moreover, the functionality required to interact with a `Visualizer` instance is inherited and can be used readily within the `ExtendedViewModel`. Thus, no further changes to the visualization framework had to be made, since all existing communication features remain intact.

The new strategy described above, has been implemented in the variation and expression data analysis software REVEAL, which was developed in this thesis as a plugin for MAYDAY. REVEAL makes use of concrete implementations of the `ExtendedViewModel` and the `ExtendedViewModelEvent` classes, which are the `RevealViewModel` and the `RevealViewModelEvent`, to integrate expression and SNV data. This allowed the generation of visualizations that can display both expression values as well as SNVs. REVEAL is described in detail in chapter 5.

3.3.1 Visualization Generation in MAYDAY

The Java Swing and Java AWT libraries offer a variety of different highly functional mechanisms, such as listeners for keyboard and mouse events or panes providing scroll functionality. These features offer interaction possibilities with any resulting visualization implemented in Java AWT/Swing. MAYDAY therefore relies mostly on the available Java features. Based on these, functionality that is often needed has been implemented and encapsulated in so-called *helper classes.* One example is the `ZoomController` that offers zoom functionality with associated plots by using the mouse wheel. This component solely consists of Java Swing components. Hence, it can be used independently of the application or required visualization. Zoom functionality is provided by adding a respective plot to the `ZoomController`. In addition to the `ZoomController` that is usually applied to every plot in MAYDAY, MAYDAY's export functionality is another example that relies solely on Java Swing components, but uses additional plugins assisting the export. These plugins provide export functionality to different user-defined file formats, including raster image formats such as `PNG`, `JPG`, or `TIFF`, as well as vector based file formats, including `SVG` and `PDF`. Export functionality is available to every plot in MAYDAY that renders visualization objects by overwriting Java's `paint(Graphics g)` method.

In order to make the implementation of new visualizations as easy as possible, plots are realized in MAYDAY by extending a common abstract `PlotComponent` class. This class provides the graphics canvas for visualization. Each `PlotComponent` is contained in a `PlotContainer`, which represents the respective Graphical User Interface (GUI) window used to make the plot visible to the user. In addition, an interface called `VisualizerMember` is available that enables the communication between plots and the `ViewModel` by using `Visualizer` objects. The interface is implemented by the `PlotContainer` and information is passed on to the `PlotComponent`. With these available structures, the implementation of a new plot only requires adding the respective plot panel to the `PlotComponent`, in order to make it usable in MAYDAY. As a starting point for new plot implementations an abstract `AntialiasPlotPanel`

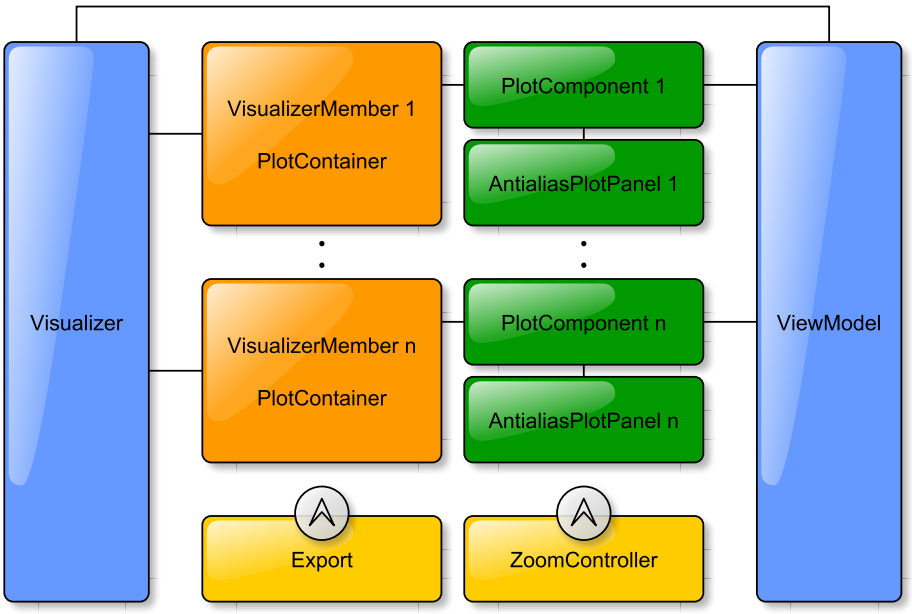

Figure 3.3: Overview of the available data structures for the generation of new visualizations in MAYDAY. Elements of the view model level are colored blue, elements relevant for the GUI representation of new plots are colored orange and elements relevant for the implementation of rendering functions are colored green. The export functionality can be added to any `PlotContainer`, and the `ZoomController` to any `PlotComponent`. These are colored yellow, since they only rely on Java Swing/AWT structures. Lines between the different components highlight their relationships.

is provided, which offers double buffering functionality as well as zooming and image export. With double buffering, the calculation of user interactions is separated from the visualization level. This strategy allows the visualization to perform necessary calculations in the background, while the user is presented the buffered image. As soon as the calculations are finished the image is replaced with the updated version that was calculated in the background. This results in a smooth interaction with plots using the double buffer technique, since costly calculations are hidden from the user. New plots that extend `AntialiasPlotPanel` automatically inherit these features and developers can concentrate on the actual rendering functions needed for the new visualization. The concept of implementing new visualizations in MAYDAY is summarized in figure 3.3.

3.4 Availability and Automated Deployment

Up to MAYDAY version 2.12, the source code was built under Java 6 using the Hudson automated build system [115]. This system was installed by Florian Battke in 2010 during his PhD thesis. The source code was thereby managed using a CVS (Concurrent Versions System). However, this system only allowed access with specific user accounts provided only by the University of Tübingen. The drawback of this approach was that there was no possibility to grant read access to the MAYDAY repository for non-university members. Furthermore, it was necessary to distribute snapshots of the source code by the Hudson system on the MAYDAY website, which could lead to unsynchronized versions between the automatically built Java Webstart version and the manually deployed source code package, since external developers were not able to synchronize their code with the repository. Instead, they had to ask one of the MAYDAY core developers to distribute their code for them, which in addition led to an increased logistical effort.

Within this work the old build system, which had the advantage of monitoring changes in the CVS and providing real-time updates to the Java Webstart version, was modified to overcome the hindrances discussed above. Since the Hudson build system only supported automatic source code builds up to Java version 6, the system was replaced by the Jenkins build system [77]. Jenkins is based on Hudson, but offers a variety of new features, such as building the source code with the new Java version 8, or the integration of git repositories. The latter was especially useful to address the mentioned availability issue. Git is a free and open source distributed version control system that can either be used online or installed as an individual GitLab server. For MAYDAY the latter was chosen[1]. This has the advantage of a secure communication of the Jenkins system[2] with the Git repository as well as the MAYDAY web server, without the need of special security protocols. Furthermore, the Git system has the advantage that developers who would like to join the MAYDAY project can be granted write access, while all other users are restricted to read-only access. Hence, everyone who is solely interested in the source code, without the need to modify it, can access the repository without requiring a personalized user account.

Another important issue that had to be addressed by the new build system are the new security restrictions introduced by Java 7 that had been further increased with Java 8. These new Java versions did no longer permit the usage of self-signed `JAR` files that were previously created by the Hudson build system.

[1] `https://lambda.informatik.uni-tuebingen.de/gitlab/explore/projects` (29/10/2015)

[2] `https://lambda.informatik.uni-tuebingen.de/jenkins/` (29/10/2015)

The new security restrictions required a complete redesign of the MAYDAY build processes used by the Jenkins system for source code deployment. Most notably, the `JAR` signing procedure had to be redesigned. To be precise, in order to continue the deployment of MAYDAY and its plugins as a Java Webstart version, the necessary `JAR` files had to be signed using a certificate from a publicly trusted signing agency. This, however, would have required giving up the MAYDAY webstart version as a free of charge solution. Nevertheless, certificates from non-trusted agencies can still be used, but have to be accepted manually by the respective users. To continue the free webstart deployment of MAYDAY, a self-signing system has been established that signs the necessary `JAR` files with a self-generated certificate that has to be accepted by the MAYDAY users. Certificate generation and `JAR` file signing have thereby been realized using OpenSSL [23]. The required certificate has been made available to the users on the MAYDAY website[3]. With these modifications, state of the art software deployment, as well as a secure distribution of the MAYDAY source code has been made possible.

[3] `http://it.inf.uni-tuebingen.de/?page\_id=248` (29/10/2015)

4. Interactive Visualization of Single Nucleotide Variation Data

Identification and interpretation of genomic variations is important to understand their impact on phenotype and possibly disease. Especially single nucleotide variations (SNVs) are of major interest, since these are the most common genetic variations in humans and can have severe impact on gene product and expression. Many studies have explored the possibilities to manifest the information contained in genotype data. Among these, genome-wide associations studies (GWAS) try to identify SNVs that are associated with a specific phenotype, typically a disease state. In addition, associations with quantitative traits (QTL studies) or gene expression (eQTL studies) can provide deeper insight into the mechanism that lead to a specific phenotype. Moreover, data visualization is a key feature in the process of detecting and interpreting significant variants. Often, Manhattan plots are used to visualize associations between SNVs and phenotype. Manhattan plots are a type of scatterplot, where genomic coordinates are displayed along the x-axis and the y-axis shows the negative logarithm of the associated p-value for each variant in the data set. A typical example of such a plot is shown in figure 4.1. Strong associations have very small p-values that lead to large negative logarithmic values, which then appear most prominent in the Manhattan plot [49]. Although studies based on genotype information have found many interesting variants and associations linked to disease [60], genotype information alone is often not sufficient and additional knowledge on the phase of a variant is needed. Most notably, in cases where phenotypic changes are the result of an interaction between several different variants, phase information becomes crucial. This is because it enables one to link a variant to its respective chromosome and in this making correct interpretations. When dealing with data containing phase information, the term haplotype is used, which refers to a cluster of variants located on the same chromosome, often in close proximity to each other.

Many tools have been designed to study variants statistically or visually, either separately or in their haplotype context. One of these tools is the SNP&Variation Suite by Golden Helix [55]. This is a collection of analytical tools for managing, analyzing and visualizing genomic together with phenotypic data. It provides many well established visual tools, but most of them do not scale well for large data. In contrast to that, Flapjack [104] is another tool that is designed for the visualization of large-scale genotype data with focus on plant data. In addition, it also offers haplotype visualization. Savant [43] offers visualization of multi-individual genotype data by agglom-

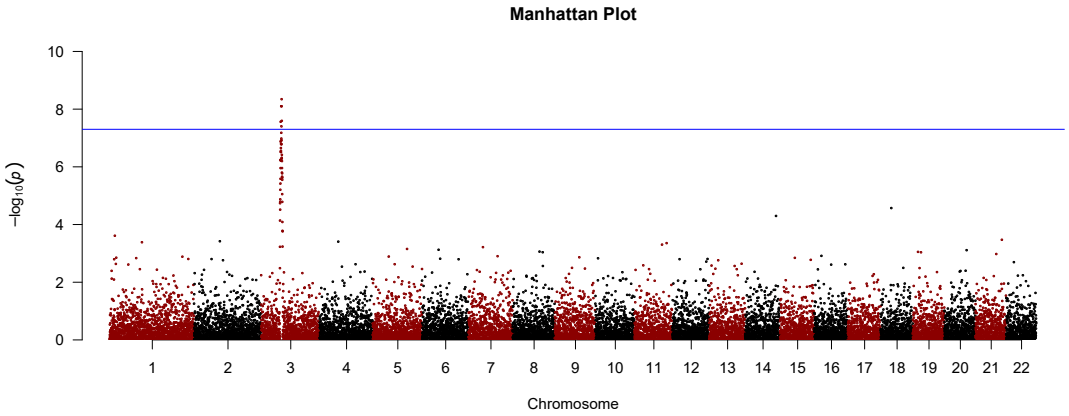

Figure 4.1: Example of a typical Manhattan plot. The plot was created in the R programming environment [157], using the package `qqman` [160]. It shows simulated human GWAS data, as obtained using the software PLINK [129]. The horizontal blue line indicates the level of significance $\alpha \leq 5 \times 10^{-8}$, resulting from a Bonferroni correction for multiple testing, where the number of SNVs is about 10^6. SNVs located above that line are considered to be significantly associated with a respective phenotype.

erating SNVs from larger genomic regions and linking them to each other. Visualization is thereby realized using a linkage disequilibrium (LD) plot as originally described by Haploview [7]. Furthermore, some genome browsers also allow visualizing genotype cohort data using specific visualization modes. Although all the tools described so far are highly useful for the visualization and exploration of complex genotype and haplotype data, they are limited to showing raw data. In contrast, Haploscope [137] visualizes haplotype cluster frequencies that are estimated using statistical models. The iXora framework [161] offers inferring haplotypes from genotype population data and association of observed phenotypes with these. It provides statistical tests such as Fisher's exact test and visualization methods such as line charts for parental haplotype distributions or bar plots for haplotype raw data. All of the described tools aid in gaining a better understanding of the underlying data. However, most of them only focus on single aspects of the visualized data. Statistical visualizations are furthermore often insufficient, since such complex data have to be addressed on many different levels and in particular interactivity is of utmost importance. This task gets even more challenging when it comes to the analysis of phased haplotype data, that is for example derived by projects such as the 1000 Genomes project [64].

Until today, an interactive tool for the analysis and visualization of phased haplotype data has been missing. However, with the iHAT (interactive Hierarchical Aggregation Table) project [54] a step in the right direction has been taken. The general idea followed by iHAT is the reduction of data

complexity in order to visually reveal structural patterns. This reduction of data complexity is achieved by applying the concept of aggregation, which allows the user to concentrate on specific data aspects. Although, iHAT can basically be used for all kinds of tabular data, one of its main applications is the visualization of genome-wide associations, allowing the user to make connections between genotype and meta-information, as for example phenotype descriptions. Thereby, qualitative as well as quantitative meta-information can be processed and visualized along with the raw genotype information. In iHAT genotype data is represented as a table containing single nucleotide variations in columns, whereas rows display individuals. For biallelic individuals only three different genotype states can occur, namely a heterozygous or homozygous allele combination, or both alleles are equal to the reference. Thus, single cells colored with one of three different color values (one for each possible state) have been used to represent genotype data. To gain a deeper understanding of underlying structures in the data, an aggregation technique and the visualization of the respective results have been described. For genotype data individuals can be grouped based on a user-defined selection, a hierarchical clustering or based on meta-information. Such groups can then be used to summarize the genotypes of the contained individuals using aggregation. The main idea is to concentrate only on the allele combination that appears most often within the selected group. Consequently, only the color value of the respective most frequent allele combination is necessary to represent the whole group. Furthermore, to show the confidence in the group genotypes, the underlying allele frequencies are used and each cell is visually encoded with a respective saturation value in addition to its hue.

However, in iHAT's first implementation, which was conducted during this thesis in cooperation with the VISUS, the visualization institute at the University of Stuttgart, it was not possible to visualize phased haplotype data. To fill this gap and to address the need for an interactive phased haplotype visualization tool, a reimplementation of the iHAT tool was necessary. This chapter concentrates on the resulting software, which is called inPHAP, short for INteractive genotype and Phased HAPlotype visualization. inPHAP is strongly based on the general concepts of iHAT, but extends its functionalities in various different ways.

In the following, a detailed description of the inPHAP tool is given and changes made in comparison to iHAT are stated. This is followed by a proof of concept application to data from Phase I of the 1000 Genomes project. Text and figures in this chapter were adapted from our previous work on inPHAP published in [72].

4.1 inPHAP - Interactive Genotype and Phased Haplotype Visualization

To address the need for an interactive phased haplotype and genotype visualization tool, inPHAP has been developed. Based on the interactive aggregation table iHAT, it follows the principal of aggregating raw data, but is a fully reimplemented version that addresses most of iHAT's drawbacks, namely the missing possibility to process whole chromosomes, up to eukaryotic genomes, as well as the lack of phased haplotype data integration and a missing advanced meta-information visualization. The new inPHAP tool is therefore not just an extension of iHAT, but a software solution with new and more advanced features and visualization concepts. It has been implemented in the Java programming language and exists in two versions, a stand-alone, as well as an integrated version in the Mayday software framework. In the following, detailed information about the design and implementation as well as the application of inPHAP to real data sets is given.

4.1.1 Application Design

In order to serve the needs of genome-wide visualizations, inPHAP has been designed as a GUI-based tool with a main focus on interactivity and the possibility to view data in various ways. For this purpose, the tool is structured into six different components that allow for the exertion of specific visualization and interaction features. An overview of the graphical user interface, highlighting all of its components is given in figure 4.2. The main component, which handles the most important part of the software, the visualization of genotype and phased haplotype data, is constructed as a matrix. Here SNVs are located in columns and subjects in rows. Different color encodings can be applied to the cells, that represent combinations of SNVs and subjects. Little white cell corners assist the user in keeping track of the respective row and column of interest. This main component is accompanied by two different components for meta-information visualization, one for subject specific and one for SNV specific meta-data. The subject meta-information component represents each data element as a separate column providing one cell for each subject. Cells are colored using user-defined color maps or gradients to represent actual data values. In contrast, the meta-information component for SNV specific meta-data contains additional rows for each data element, where colored cells encode SNV-based meta-information. These three components make up the main visualization of the application. In order to improve interactivity with inPHAP, three further components have been designed. An overview panel displays the viewer's current focus and location in the main visualization. It provides a zoomed-out view of the complete data set with a small red rectangle giving

an estimate of the proportion of the currently visible sub-matrix in comparison to the whole matrix. Furthermore, a settings panel has been added to the application, that allows the user to quickly switch between different visual representations, to adjust colors and fonts, as well as to change cell sizes if needed. The last component is an information panel, that provides basic statistics for the currently loaded data set, such as the total number of subjects and SNVs, as well as the number of meta-information columns and rows. In this panel, "MI columns" stands for subject specific meta-information and "MI rows" for SNV specific meta-information. The graphical user interface is completed by a menu bar and a button bar on top of as well as a status bar underneath these six components. The menu bar and button bar offer various functions, such as data import and export, sorting, filtering, aggregation or image export. The status bar informs the user about changes made to the data and to the visualization of the data by giving details about what has been changed and how this change affected the underlying data. However, only the latest change is displayed there. For a complete summary of all interactions with the applications, as well as potential error messages, an additional log window is accessible through the help menu in the menu bar.

Genotype Visualization Human genotype data usually consist of two character symbols standing for the respective nucleotides on the maternal and paternal allele. However, it is unknown which of the two bases originates from the father and which from the mother. Furthermore, only those bases are of interest to researchers that show some difference to a defined reference. When comparing the genotype of an individual at a specific genetic locus to a reference, there are only three states that can occur:

1. one of the two alleles differs from the reference nucleotide, which means that there is a heterozygous SNV at that position;

2. both alleles differ from the reference nucleotide at the specified position, leading to a homozygous variation;

3. none of the two alleles differs from the reference.

Based on these observations, a visual encoding can be chosen that represents one of these three states at every genetic locus of interest. For the genotype visual encoding in the inPHAP tool, each cell in the matrix is colored with respect to the genotype state defined by the respective SNV/subject pair for that cell. By default, red color is used for homozygous SNVs, yellow color for heterozygous SNVs and green color if there is no change in comparison to the reference base. The specific default color values have been chosen based on ColorBrewer color maps [52], in order to maximize the ability to distinguish cells from each other. These default color values can, however, easily be

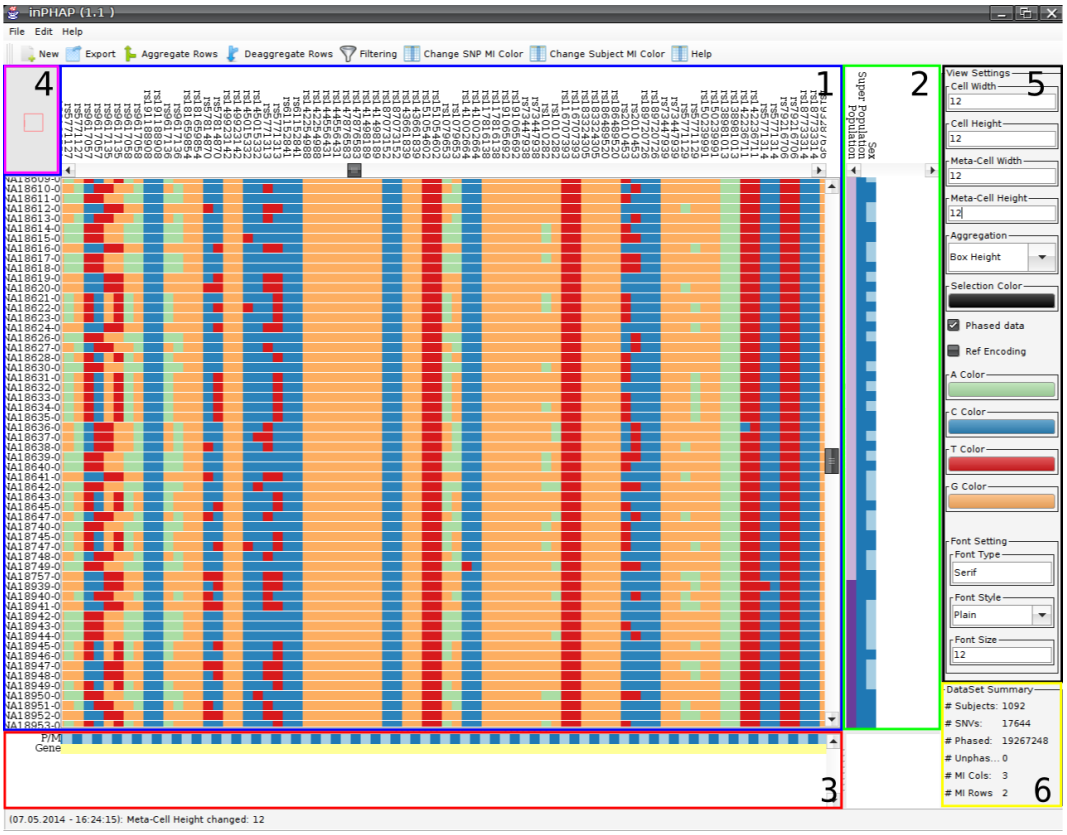

Figure 4.2: The inPHAP graphical user interface. It consists of six components, which are highlighted with boxes of different color. **Blue (1)**: The genotype/haplotype visualization component providing color-encoded base information for phased haplotype or unphased genotype data, **green (2)**: the subject meta-information component, **red (3)**: the SNV meta-information component, **purple (4)**: the overview component, displaying the viewers current focus in the main visualization, **black (5)**: the settings component, which allows the user to quickly change the visual representation of the data, **yellow (6)**: the data set summary component, providing general information for the currently loaded data set.

modified by the user to fulfill specific needs or to enhance visual separation of the three different states for color blind people. In figure 4.3 an example of a typical genotype visualization is shown using the default color encoding for homozygous and heterozygous SNVs as well as alleles that show no difference to the reference.

Phased Haplotype Visualization In contrast to genotype data, where the phase of a respective allele combination is unknown, haplotype data offers the possibility to arrange single nucleotide variations into groups based on their

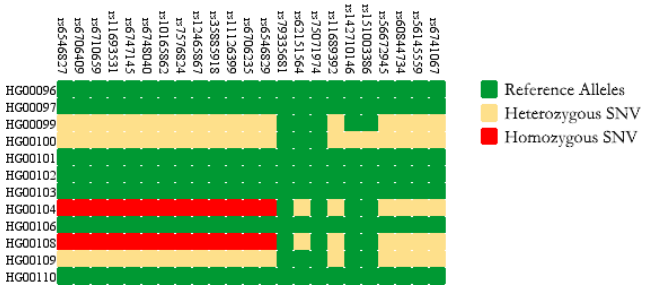

Figure 4.3: Genotype visualization with inPHAP. A randomly chosen region on chromosome 2 of the human genome is shown. Rows represent individuals and SNVs are shown in columns. Colored cells encode for an individuals allele combinations with respect to the shown SNVs.

chromosomal origin. This requires analysis tools to handle phase information by respective visualization approaches that have to take the origin of a variation into account. Consequently, a single column for a SNV is not sufficient for a reasonable visual representation of phase information with the matrix approach introduced in the inPHAP tool. Therefore, the concept of having a single column for a SNV was extended to one column for each allele from a SNV locus. This means that each cell now represents one specific allele for a single subject. This design choice is motivated by data from Abecasis *et al.* who used two rows for each allele [28]. The decision to use two columns rather than two rows offers the possibility to assign meta-information to alleles only, which can be useful in some cases. As for genotype data a reference based visual encoding is available for the haplotype visualization, but here only two states are possible. Either there is a difference to the reference base for the specific allele or the reference base and the base for the respective allele are the same. By default yellow color is used to highlight a difference from the reference base and blue color represents similar bases. Although this visual representation easily allows the user to spot differences to a given reference, it does not allow to spot differences between the paternal and maternal allele, if both are equal to or if both differ from the reference. To address this issue, a second visual encoding is introduced that is based on the nucleotides themselves rather than their similarity to the reference base. For each of the four bases Adenine (A), Guanine (G), Cytosine (C), and Thymine (T) a unique color is used. By default green color is used for A, blue for C, red for T, and yellow for G. Again, the colors have been selected based on ColorBrewer color maps. Cells representing missing nucleotides, as for example in males when comparing the X and Y chromosome, are colored white. With this second color encoding differences between the maternal and paternal allele can be investigated more easily. To make optimal use of both visual representations, inPHAP allows the user to interactively switch between them with a click of a button using the

settings component. Figure 4.4 shows both visual encodings of the haplotype visualization for the same chromosomal region.

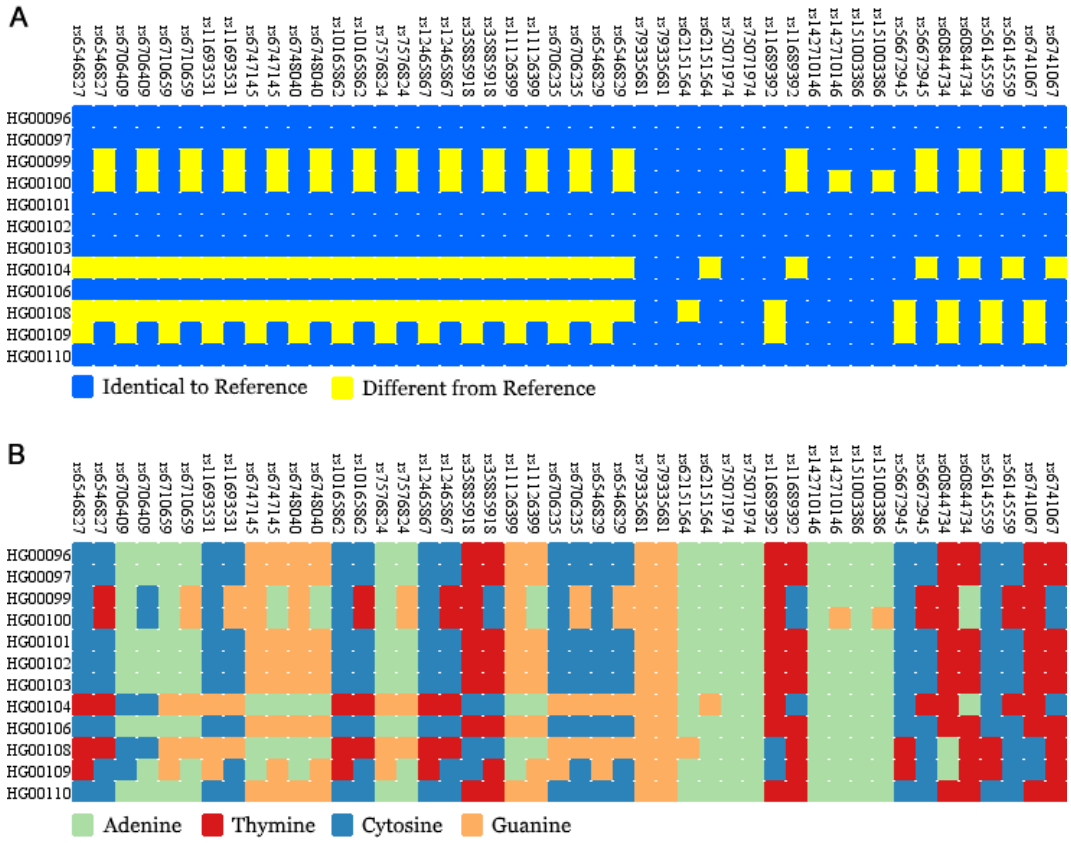

Figure 4.4: Phased haplotype visualization with INPHAP. (A) shows the reference based visual encoding on a randomly chosen genomic region on chromosome 2 of the human genome. (B) shows the same genomic region as (A) using the nucleotide based visual encoding. In both visualizations individuals are shown in rows and SNVs in columns.

Meta-Information Visualization The INPHAP tool offers the import of two different kinds of meta-information, namely SNV based as well as subject based meta-data. Although these types are distinguished during the import of the data, they are treated equally on the visualization level. Here, the only distinction is made between numerical meta-data and categorical meta-data, no matter if they belong to subjects or SNVs. Usually, for numerical meta-data, color gradients are used and the numerical values from the respective meta-information group are mapped to color values from the gradient. For categorical meta-data, each category is assigned a unique number, which is then mapped to a unique color from a user selected color map. Color gradients

as well as color maps are mainly taken from ColorBrewer, but also standard color gradients are available and user-defined gradients can be generated. By default, color gradients are used for numerical meta-data and color maps for categorical meta-data, but the user is not restricted to this practice and can for example also use color gradients for categorical meta-data or color maps for numerical meta-data if needed.

4.1.2 Interaction Possibilities

To provide comprehensive insights into the visualized data, INPHAP offers a variety of different interaction possibilities explained in the following.

Navigation Navigation is possible along the subject axis as well as along the SNV axis using interactive sliders. Furthermore, the user can navigate using the overview component. There, the current view is indicated by a red rectangle, which can be dragged to the desired location inducing a change of the current view in the main visualization.

Zooming INPHAP offers zooming in two different dimensions, i.e. the width and height of each cell in the matrix visualization can be adjusted separately. In addition, semi-independent zooming is possible for meta-information cells. These are linked to the main matrix cells, which means that changing the height of a cell in the matrix also changes the height of cells for subject meta-information columns, while their width can be adjusted individually. Analogously, the width of a cell in a meta-information row is linked to the matrix while its height can be adjusted individually. This strategy enables a clear visualization of meta-information even on a zoomed out overview of the main visualization component.

Selection Subjects as well as SNVs can be selected by either clicking on the respective identifiers or by dragging over a region of subject or SNV identifiers. Selected cells are then highlighted with a colored border drawn around the selected cells and the subject or SNV identifiers are overlaid with a colored box. Selection of subjects or SNVs also affects the respective meta-information cells, which are highlighted with colored borders, too. The default selection color is black, but a different color can easily be chosen via the settings component.

Sorting INPHAP allows the user to sort both, subjects and SNVs. The sorting process itself is guided by meta-information. With a double-click on a meta-information identifier, the respective rows or columns are sorted according to the order given by the meta-information values. Sorting is possible in ascending or descending order using a stable sort approach. This means

that the order of elements that belong to the same sub-group according to the selected meta-information is not changed. With this it is possible to sort consecutively based on different types of meta-information. In addition, sorting of subjects can be performed based on a previously calculated hierarchical clustering, which can be imported in the `NEWICK` format [113].

Filtering Since the number of SNVs in a typical genotype or haplotype data set is usually very large, offering the user the ability to concentrate only on those SNVs that are of special interest can be beneficial. To do so, the data have to be reduced to only those SNVs that pass a user-defined filter. For this, inPHAP offers a variety of different filtering methods:

- *Chromosomal Location*: only those SNVs are shown that are located in a specific region on a chromosome, such as a gene or promoter region or any other regions specified by the user.
- *SNV List*: if a predefined list of interesting SNVs is available the user can reduce the current view to only those SNVs that are contained in that list.
- *Regular Expression*: SNVs can be filtered using a regular expression for SNV identifiers.
- *Frequency*: only those SNVs are shown whose genotype frequency lies below or above a user-defined threshold.

All these filters only affect the current view of the main visualization and not the underlying data. This allows for a quick response to the user, since the underlying data structure does not have to be changed. Furthermore, several filters can be combined to build more powerful filtering rules.

Aggregation Aggregation has proven to be a powerful method to reveal hidden structures in the data by reducing the overall complexity [54]. The implementation of aggregation techniques in the inPHAP tool is based on the general concept introduced in iHAT, but with extended functionality regarding the methods for aggregating SNV data as well as meta-data. In fact, the original implementation in the iHAT software has been reimplemented to enhance performance of the overall process as well as to allow the user to aggregate phased haplotype data. Thereby, aggregation is only possible on a per SNV level summarizing genotype or haplotype constellations of different subjects, that share some common characteristics. The selection of subject groups for aggregation can be based on a user-defined selection of rows, or guided by meta-information for the subjects. Such meta-information can, for example, be the affiliation of a subject to a specific population or

sub-groups of populations. In such cases, aggregation makes it easier to spot similarities shared by specific subject groups as well as to spot differences between those groups. For the aggregation process itself, several different methods have been implemented for the variation based data as well as for the meta-data. Consequently, the aggregation of subjects not only affects the underlying data in the main visualization, but also the corresponding subject meta-information. For variation data a maximum and minimum aggregation method is available. If for example the maximum aggregation method has been selected, then for each SNV the base with the highest frequency given the selected subjects is chosen as the consensus base and the respective frequency is stored as an indication of how representative this base is given the underlying base distribution. This method is similar to the aggregation strategy introduced in iHAT [54]. The minimum aggregation method works analogously. For meta-information the user can choose between the maximum, minimum, mean or median aggregation method. Moreover, different methods can be selected for the nucleotide data and the meta-data. For example, the maximum aggregation method can be used for the nucleotide matrix while the corresponding meta-information columns are aggregated using the mean aggregation methods. This is especially useful if, for instance, gene expression information is available for the subjects. Here, a summarization using the mean or median aggregation method for the expression values of individual subjects provides more valuable information than a summarization based on the maximum or minimum expression value.

For the visual encoding of aggregated data, two different strategies have been followed. Each of these strategies focuses on a specific aspect of the data. If more attention should be drawn to the consensus base rather than its frequency, aggregated cells are represented using color with saturation of the cells adjusted according to the frequency of the consensus base. This is the default visual encoding, which has also been used in iHAT. If, however, more attention should be drawn towards the differences in consensus base frequency, then a saturation based approach is not very efficient. In fact, positioning along a common scale has proven to be a good alternative solution [95]. The second visual encoding therefore uses filled boxes for each cell, whose color represents the consensus base and whose height displays the consensus base frequency. With this strategy, the user can easily compare the frequencies of different SNVs or alleles with each other. If several individuals are aggregated, their representative rows are combined into a single row and the identifier of the new row has to be adjusted. The new identifier of the aggregated row is constituted by the prefix "AGN" followed by a number that uniquely identifies the respective row. Furthermore, the number of subjects that were chosen for the aggregation is shown in brackets as a suffix of the new identifier. Figure 4.5 shows an example of how aggregation can be used to identify differences in

rare variants between whole populations for the Metachromatic leukodystrophy (MLD) associated gene *ARSA*.

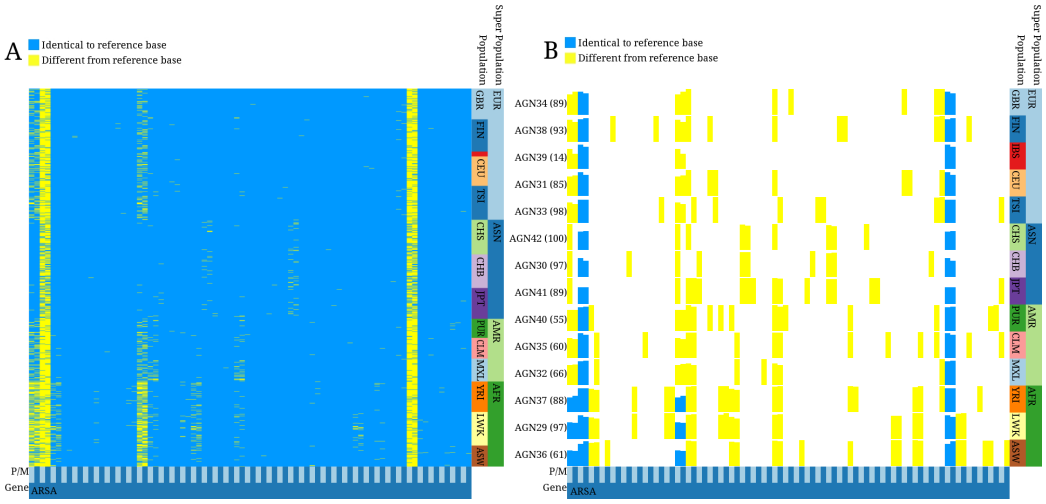

Figure 4.5: SNVs for the MLD associated gene *ARSA*. (A) Data is shown without aggregation. Individuals have been sorted based on their population affiliation. (B) Individuals have been aggregated based on their population affiliation using the "minimum" aggregation method for SNVs and the "maximum" aggregation method for subject meta-information. Box height based visual encoding has been used for the representation of aggregated data. For abbreviations of the population names see table A.1 and table A.2 in the Appendix.

Further Interaction Features Besides the major interaction features mentioned above, further minor changes to the introduced visualizations can be made. These include the change of the label font, style and size, the interactive switch between the different visual encodings, the fast and quick change of the colors used to represent the different data types as well as switching between the two visual representations of aggregated rows. All these features are accessible from the settings component with a click of a button. Furthermore, all visualizations can be exported to different file formats, including the pixel based `PNG` and `JPG` format, as well as vector based formats such as `SVG` and `PDF`.

4.1.3 Data Structures

Human genotype and phased haplotype data usually consists of two characters from the alphabet $\Sigma = \{A, T, G, C, -\}$, one for the maternal allele and one for the paternal allele. To enable the representation of missing allele information the character '$-$' is included. This is, for example, very common for SNVs

on the X chromosome in males, since many corresponding alleles are missing on the Y chromosome. Since the number of variants can become very large, and, in consequence, also the amount of data that has to be processed for a single individual, a memory efficient representation of the possible allele combinations was necessary. A binary encoding of the nucleotides was chosen using only two bits to store a character $c \in \Sigma$. Consequently, only 4 bits are necessary to store the maternal and paternal allele combination for a specific subject/SNV pair. With this strategy, the amount of necessary memory could be reduced by a factor of 8 in comparison to a naive implementation using character primitives, which would require two bytes per nucleotide in Java.

However, for the visualization of the underlying data, they have to be decompressed from their binary form. To keep interaction smooth at all time, only the data from the sub-matrix that is currently visible to the user are decompressed. Since only three to four differently colored cells are used, depending on the chosen visual representation of the data, a second memory and time saving strategy was implemented for the visualization process itself. For each character $c \in \Sigma$ a colored image is rendered in memory before the actual visualization takes place. When drawing the sub-matrix that is visible to the user, the pre-rendered images are used for painting. Since many cells encode similar values, the pre-rendered images can be used multiple times during a single repaint event, leading to a huge reduction in memory and time needed for redrawing the whole scene. Furthermore, repainting a pre-rendered image is much faster in Java than its recalculation, which also has a positive effect on the overall runtime. Changes that require recalculation, such as changing the colors used for the visual encoding of the nucleotides, can also be performed efficiently since for each change in color only a single image has to be recalculated followed by a single repaint event of the visible sub-matrix. For aggregated cells, where different saturation values are needed, an array of 100 differently saturated white images has been pre-rendered during startup of the INPHAP application. To represent an aggregated cell, one of the saturation images is chosen depending on the aggregation frequency and painted on top of the nucleotide image. With this strategy, frequencies can be displayed with a one percent accuracy. This is sufficient to cover the number of saturation values a human being is able to distinguish. Last but not least, the selection rectangle has also been implemented as a pre-rendered image, that can be drawn on top of selected cells.

Input File Formats Data can be imported for visualization with INPHAP in two different file formats. The variant call format (`VCF`) is the standard file format for genotype and phased haplotype data [32]. It is a tab-separated text file format containing one variant per line, while the variant information and the subjects allele combinations are structured into columns. In order

to apply INPHAP to data in the `VCF` file format, all the subject alleles have to be contained in a single `VCF` file. The import of multiple `VCF` files at the same time is currently not possible. The second option for importing data is the `IMPUTE2` file format. This is again a text based format used by the phasing program IMPUTE2 [64] to store genotype information. Here, the data is split into three different files, a `LEGEND` file containing information on the variants, a `SAMPLE` file, holding details on the subjects and their relations and a `HAP` file containing haplotype information for each subject. This file format is also used by other phasing programs such as SHAPEIT2 [34, 35] or BEAGLE [20, 21]. For the import of SNV as well as subject meta-information an INPHAP specific text-based file format has been developed. These meta-information files have tab-separated columns and contain two header lines, the first providing an identifier for each column and the second indicating the type of meta-information, which can be either numerical or categorical. The first column however is reserved for SNV or subject identifiers, depending on the type of contained meta-information.

4.1.4 Application to Phased Haplotype Data from the 1000 Genomes Project

In order to demonstrate the power and capabilities of INPHAP, it was applied to data from Phase I of the 1000 Genomes project [28]. The goal of this project is the identification of most of the genetic variants with frequencies larger than 1% in the populations studied. This is mainly achieved by sequencing of many individuals using next-generation sequencing platforms in order to provide a comprehensive resource on human genetic variation across several populations. In the most recent publication [28] from the 1000 Genomes Consortium a 100 kilo-base large region on chromosome 2 was highlighted, containing the genes *ALMS1* and *NAT8*, for which variations have been associated with kidney disease [22]. In contrast to the visualizations produced by Abecasis *et al.*, application of INPHAP to the same 100-kb region resulted in two figures, one for the common and one for the rare variants, instead of a single figure showing all variants at the same time. Figure 4.6 shows all frequent variants selected by Abecasis *et al.* (frequency $> 0.5\%$). Variants below this threshold were defined as rare by Abecasis *et al.* Figure 4.7 shows the same region, but only variants with a frequency $< 0.5\%$.
These figures have been produced by first loading all variants from chromosome 2 into INPHAP. For this, the variants were stored in the `VCF` file format and loaded using INPHAPs integrated `VCF` file import option. Afterwards, variants were filtered such that only those remained that are located in the 100-kb region described by Abecasis *et al.* In order to separate common and rare variants from each other frequency based filters were used with thresholds $> 0.5\%$ and $< 0.5\%$ respectively across all individuals in the data set. Further-

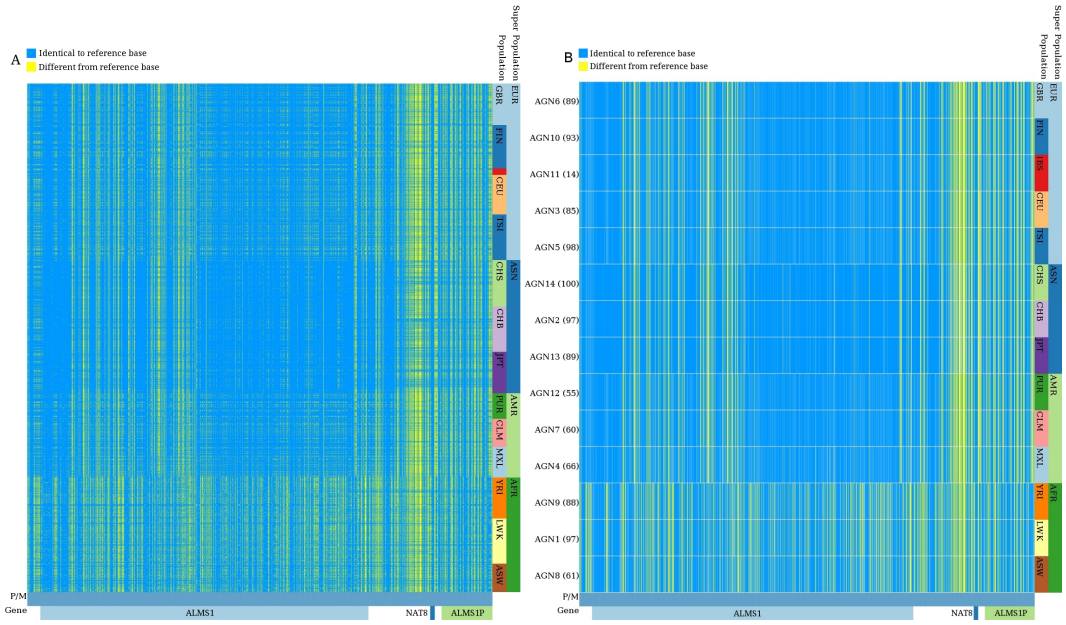

Figure 4.6: Phased haplotype visualization of a 100-kb region on chromosome 2 spanning the genes *ALMS1*, *NAT8*, and *ALMS1P*. SNVs have been filtered based on a frequency $> 0.5\%$ across the 1096 human individuals of Phase 1 of the 1000 Genomes project. (A) Individuals are sorted according to their population affiliation. (B) Individuals are aggregated according to their affiliation with a common population using the "maximum" aggregation method. For abbreviations of the population names see table A.1 and table A.2 in the Appendix.

more, aggregation techniques were applied to make the respective differences between populations stand out more prominently. Figure 4.6 shows that for the African (AFR) super population, there are more highly frequent SNVs in the *ALMS1* region than for the other super populations, whereas for the Asian (ASN) super population only very few variants are found in the central part of the *ALMS1* gene. These are more likely for Europeans (EUR) and Americans (AMR). While common variants are more uniformly distributed in this 100-kb region for the African population, for the other populations they are mainly located in two clusters, namely the first part of the *ALMS1* gene and an approximately 20-kb region spanning the genes *NAT8* and *ALMS1P*. All of the stated observations correlate with the findings of Abecasis *et al.* who showed that highly frequent variants are differentially distributed across populations. Taking a closer look at the rare variants one can see that the African population also has a higher number of these in comparison to the other populations. However, the numbers vary strongly between the different populations, even for those belonging to a common super population. An example is the Iberian population in Spain (IBS), where only very few rare variants are present. In

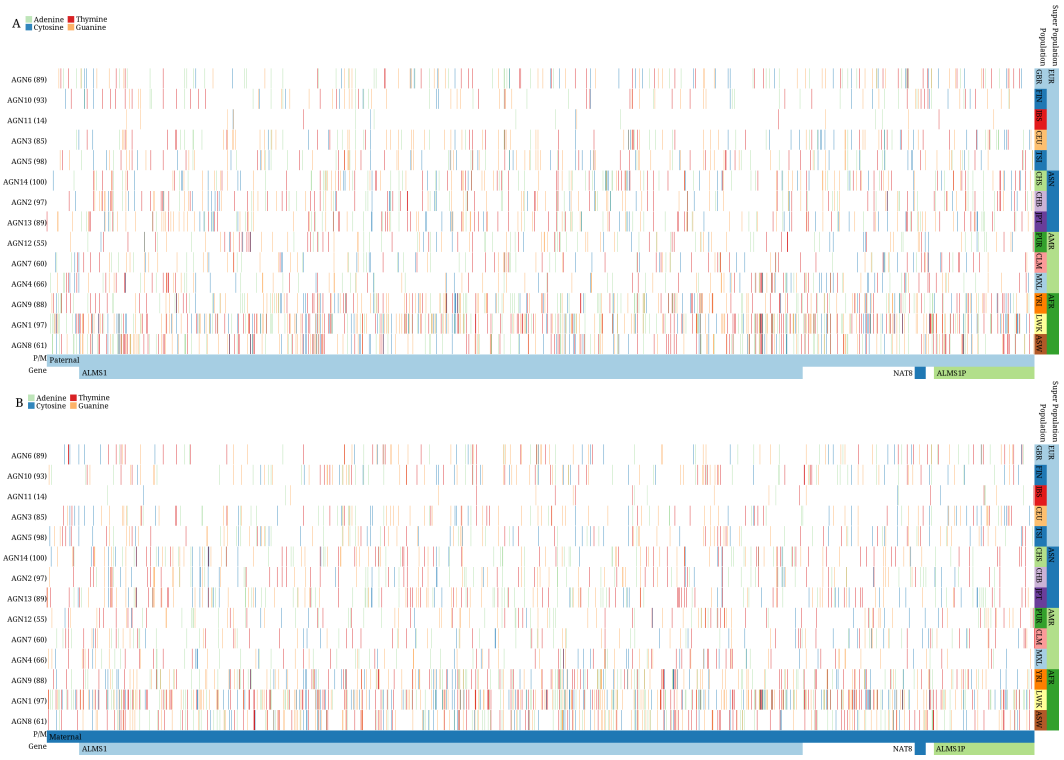

Figure 4.7: Phased haplotype visualization of a 100-kb region on chromosome 2 spanning the genes *ALMS1*, *NAT8*, and *ALMS1P*. SNVs have been filtered based on a frequency $< 0.5\%$ across the 1096 human individuals of Phase 1 of the 1000 Genomes project. Individuals are aggregated according to their affiliation with a common population using the "minimum" aggregation method. (A) Only SNVs for the paternal allele are shown. (B) Only SNVs for the maternal allele are shown. For abbreviations of the population names see table A.1 and table A.2 in the Appendix.

addition, variants are mostly heterozygous. This means that they are located either on the paternal or the maternal chromosome, but rarely on both. This again correlates well with the findings of Abecasis *et al.*, who argued that the main reason for rare variants in the Spanish (IBS) and the Finnish (FIN) population are events such as clan breeding or admixture of diverged populations [28].

Another important question is the influence of specific variations, especially rare ones on subgroups of a population or on only a few individuals. Studying such variants is usually challenging due to the difficulty to filter out common variants that are more alluring and secondly due to the overall number of subjects in the data set, which renders it difficult to concentrate on structures of interest. By the application of INPHAP to another subset of the data produced by the 1000 Genomes project, its capability in studying specific

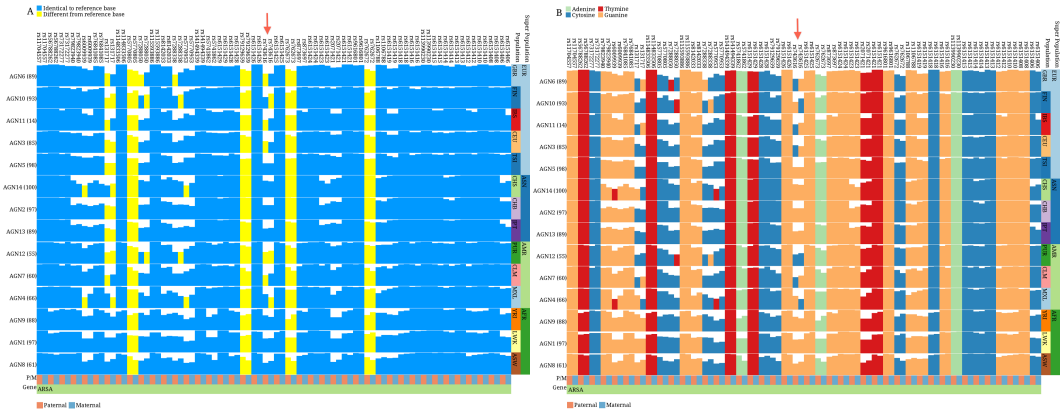

Figure 4.8: Two haplotype visualizations with INPHAP showing SNVs for the MLD associated gene *ARSA*. SNVs have been filtered based on their frequency across the 1096 individuals, such that only those with a frequency $> 0.5\%$ are shown. Individuals have been aggregated according to their population affiliation and populations have been sorted based on their super population affiliation. Bar heights for each SNV display the frequency of the aggregated consensus base. The arrow points to the maternal allele of the central SNV with dbSNP ID `rs743616`, which is assumed to be one of the causative mutations leading to MLD. (A) shows the selected SNVs using the reference-based visual encoding. (B) shows the selected SNVs using the nucleotide-based color encoding. In both visualizations differences between the maternal and paternal alleles stand out clearly. For abbreviations of the population names see table A.1 and table A.2 in the Appendix.

variants in more detail is demonstrated. For this, a region covering the gene *ARSA* on chromosome 22 has been selected, which is known to be associated with Metachromatic leikodystrophy (MLD), an inherited disorder, affecting the growth and development of myelin. Myelin is a crucial insulator around nerve fibers in the human central and peripheral nervous system. Several missense mutations on chromosome 22 lead to defects of the enzyme arylsulfatase A (ARSA), which as a result can no longer fulfill its original function [126]. One of these mutations is a SNP with dbSNP ID `rs743616`, which is a $C \rightarrow G$ substitution on the reverse strand leading to an amino acid change of Threonine $\rightarrow$ Serine in the ARSA protein. To visualize the important variants for the MLD associated gene *ARSA*, the data for the chromosome 22 has been imported in the `VCF` file format. SNVs contained in the *ARSA* gene have then be filtered using the chromosomal location filter. Afterwards, individuals have been aggregated according to their population affiliation followed by a sorting of the aggregated rows according to the super population affiliation. The result is shown in figure 4.8. Because the data from the 1000 Genomes project is provided relative to the forward strand, the highlighted position corresponds

to a $G \rightarrow C$ substitution in the figure. One can see that there are differences between super populations that can be spotted easily. For example the Asian (ASN) and African (AFR) super populations show low pathogenic `rs743616` allele counts for MLD, whereas the European (EUR) and American (AMR) super populations show significantly higher pathogenic allele counts. Especially the Puerto Rican (PUR) population stands out clearly. Furthermore, the allelic origin of the SNP `rs743616` can be distinguished. As can bee seen in figure 4.8 this SNP seems to be mostly maternal in the Mexican (MXL) population in Los Angeles. In addition, the aggregation technique, together with the bar height visual encoding for aggregated rows, gives a good estimate of the significance of the differences in allele counts.

4.2 Conclusion

inPHAP has been designed for the study of genotypes as well as phased haplotypes. Visualization of phase information allows for the investigation of the influence of certain alleles on specific phenotypes. Thereby, the design of the application was inspired by the computation information design approach presented by Ben Fry [46], who suggested seven main steps for an application that are needed to understand large and complex data, namely acquire, parse, filter, mine, represent, refine and interact. All these steps have been addressed in the inPHAP tool. Furthermore, inPHAP offers various visual representations and a large number of different interaction possibilities, including filtering, sorting and most notably aggregations. For the latter, it could be shown that it is a valuable tool for the identification of hidden patterns in the data and can help to make well informed interpretations. Enhancement of the visualization approach with additional meta-information, further improves the discovery of interesting patterns. By the time of this thesis, inPHAP was the only available interactive visualization tool capable of visualizing genotype as well as phased haplotype data.

5. An Integrative and Interactive Visual Analytics Tool for Single Nucleotide Variation Gene Expression Association Data

As shown in the previous chapter, genotype and haplotype patterns can be used to draw conclusions about similarities and differences between populations or between subgroups of populations. This enabled the identification of SNVs associated with a specific phenotype, such as a disease state. Furthermore, it was demonstrated how these SNVs are distributed within a single or between multiple populations. Although the application of INPHAP to GWAS data can provide valuable insight when dealing with binary phenotypes, more appropriate analysis methods are needed for complex phenotypes, such as gene expression levels. Especially when studying eQTL data, the application of statistical tests in combination with visualization approaches leads to more meaningful results. Genetic factors can either directly affect gene functionality or indirectly lead to changes in gene expression levels. However, to identify these factors information on SNVs and transcript abundances has to be combined to predict potential SNV related gene expression changes. Statistical approaches that allow for the prediction of eQTL associations are implemented in the software PLINK [129] (see chapter 2, section 2.6.2). However, the interpretation of the text-based results is often difficult and not suited to deal with large and complex data. Visualizations are of utmost interest, since they have the potential to reveal hidden patterns in the data.

There are various software solutions available to visualize eQTL associations. The eQTL Explorer [108], as well as the AssociationViewer [101] allow for the visualization of SNVs in their genomic context using genome browsers. The advantage of this approach is the convenient identification of cis- and trans-associations. However, due to the limitation of a solely linear representation, complex associations, such as epistasis, cannot be studied. Furthermore, the integration of meta-information, such as statistical test results, or the exhaustive study of gene expression levels, is not possible. A web-based alternative is the eQTL Viewer [174]. The key feature is a scatter plot, in which gene locations are plotted against SNV positions. Again, the identification of cis- and trans-associations is easily possible and, in addition, interactions with the plot are available. These include zooming as well as running interactive

queries for specific genes or SNVs, which are then highlighted within the plot. Furthermore, meta-information can be introduced and mapped to data points. However, eQTL Viewer also lacks the possibility to investigate gene–gene or complex SNV–gene interactions as well as to perform appropriate gene expression analyses. In contrast to these methods, Genevar [170] offers a combined visualization of a typical Manhattan plot with a tabular view of meta-information, such as statistical test results. These tests can either be applied directly within the software, or imported from external tools in text-based data formats. The two views are visually linked to each other, such that user selection of SNVs in either visualization is immediately reflected in the other. In addition, SNV–gene associations can be studied on the genotype level by using box plots, showing the expression value distribution of a gene within a sub-population. Thereby, groups of individuals are built according to their genotype and box plots for these genotypes are drawn next to each other for each gene. This enables to quickly spot changes in expression levels that are associated with specific genotype combinations. The drawbacks of this application are however, a lack of support for SNV–SNV interaction (epistasis), as well as for the analysis of gene expression differences between sub-populations. A software solution that allows for the generation of whole eQTL analysis workflows is provided with GenAMap [31]. It focuses on the analysis of SNV–gene association results and offers node-linked graphs and dot matrices to display associations of SNVs with their respective quantitative traits. For the visualization of statistical test results, Manhattan plots can be generated. GenAMap is a highly interactive tool, offering various interaction possibilities, including zooming, selections and graph manipulations. The latter are provided through the Jung graph library [114]. Furthermore, meta-information for specific data elements can be obtained through links to external databases. Although the included methods are useful to digest statistical association results and to explore SNV–gene associations in various ways, the analysis of the respective gene expression levels and their influences on disease states is limited. Thus, an integrative study of gene expression and eQTL associations can not be conducted.

To summarize, all the available tools for eQTL visualization and analysis share the same drawback of missing appropriate methods for the integrative study of SNV and gene expression data. Most approaches strongly focus on SNV-based analyses and are therefore well suited for GWAS, but limited for the analysis of the impact of SNVs on gene expression levels and consequently on disease. In this work, a new GWAS and eQTL analysis and visualization toolbox has been developed that offers both, powerful gene expression analyses through the integration into the MAYDAY software, as well as appropriate methods for SNV data analysis and visualization. Furthermore, integrative studies of these two data types are possible through a flexible data and view model (see sec-

tion 3.3) allowing for the development of visual analytical methods with which gene expression and SNV data are addressed equally well. The new software, called REVEAL, will be introduced in more detail in the following. Thereby, parts of this work, including text and figures, are based on a previous publication on eQTL data visualization, where REVEAL was first introduced [71].

5.1 REVEAL: A Foundation for GWAS and eQTL Data Analysis

As has been mentioned above, an appropriate solution for the integrative study of GWAS, eQTL and gene expression data in a common software framework has been missing. To address the need for such a software, REVEAL has been developed with a focus on data visualization and exploration. It has been implemented as an extension of the MAYDAY software suite (see chapter 3), which offered the possibility to make use of already existing methods and visualizations for gene expression data analysis. REVEAL can be operated through a graphical user interface (GUI), and contains various different statistical and visualization methods. In the following, the GUI and technical details about the integration into MAYDAY are described. REVEAL specific data structures will be introduced, followed by mechanisms needed for linking already existing visualization in MAYDAY to newly developed visualizations in REVEAL. Afterwards, focus is directed on visual analytical approaches developed in this work for the analysis of GWAS and eQTL association data. Finally, the developed methods will be demonstrated on two eQTL data sets that were part of the BioVis 2011 and 2012 data analysis challenges.

5.1.1 Graphical User Interface

REVEAL is a highly interactive visual analytics tool. Thus, a simple and comprehensive graphical user interface (GUI) is required that allows for quick and easy access to the incorporated methods and visualizations. In order to achieve this, REVEAL's GUI is structured into three parts, which are shown in figure 5.1. The upper part contains the menu bar, from which all available methods and plots can be accessed. Furthermore, a quick button bar, which can be configured individually by each user, offers immediate access to often needed visualizations. The second part of the GUI shows the project overview, where all the currently active data is displayed in a tree like structure. Here, the upper part displays the project specific subject, gene and SNV data. These are separated from associated meta-information in the lower part. The third component of the GUI is a tabbed pane, where each currently active visualization is shown in a different tab. The tab is further separated into two components. The left component corresponds to the graphic canvas and

is used for plot generation. The right component displays the available plot specific adjustment options. In order to make use of multi screen computers, tabs can be detached from their tab pane and shown in a separate window. This is also useful for single screen workstations to gain a complete picture of the data of interest by allowing for the investigation of multiple different visualizations next to each other.

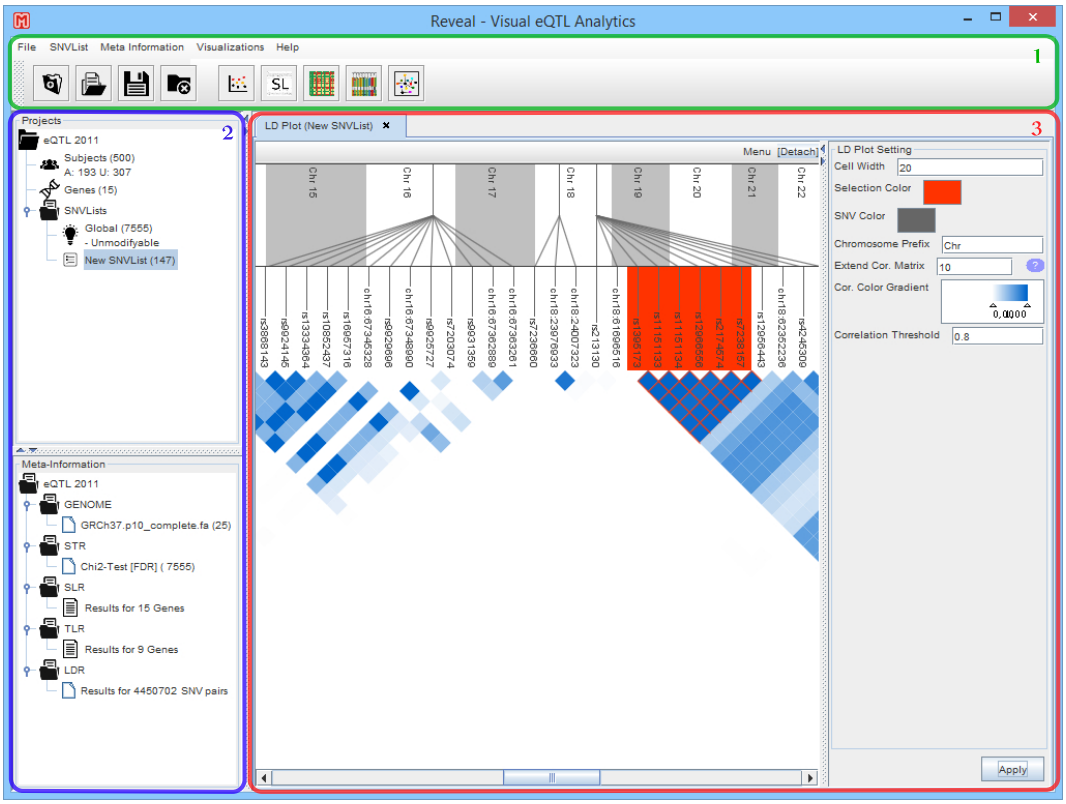

Figure 5.1: Overview of the different components of REVEAL's graphical user interface. The top component (1) consists of the menu and quick button bar, which grant access to the different methods and visualizations implemented in REVEAL. The second component shows a summary of the active projects and their associated data objects, separated into subject, gene and SNV specific data in the upper part and meta-information in the lower part. The third component is a tabbed view of the active visualizations. Each visualization consists of a canvas used to draw the plot and a setting panel providing plot specific manipulation options.

5.1.2 Data Structures and Data Handling

REVEAL has been implemented by following MAYDAY's plugin concept (see chapter 3 for more details). This means that REVEAL can be integrated into MAYDAY through a separate `JAR` file, which will be recognized by the

MAYDAY `PluginManager`. Furthermore, all features available in REVEAL, including statistics, plots or meta-information handling and processing, have also been implemented as REVEAL specific plugins following MAYDAY's plugin strategies. Thus, adding a new feature to REVEAL only requires to implement a specific abstract plugin class. Depending on the plugin's purpose, there are different types of abstract framework classes available. For example, there is a `RevealVisualization` class for the integration of new visualizations into REVEAL, or the `StatisticalTest` class, offering access to data structures needed to perform statistical testing in REVEAL. All these classes implement the interface `AbstractPlugin`, which is used by the `PluginManager` to identify and integrate plugins into MAYDAY.

However, in order to provide access to the underlying data, which is needed to apply the methods implemented in REVEAL, appropriate data structures for handling the different data types are needed. These have been implemented by following the extension concepts described in chapter 3, section 3.3. Consequently, the most general data structure in REVEAL is the `Project`, which holds references to the MAYDAY `DataSet` as well as to the REVEAL specific `DataStorage`. Furthermore, variation data typically consists of two types of data objects. The first is the SNV itself, for which a separate `SNV` class holds the necessary information, such as its location on the genome, as well as the respective reference nucleotide. In addition, information on an individual's genotype has to be available and linked to the respective `SNV` object. Therefore, a `Subject` class is available in REVEAL that represents individuals. `Haplotype` objects hold the information for specific `Subject`–`SNV` pairs and allow for an easy retrieval of such information, e.g. for visualization purposes. These data types are managed by the `DataStorage` object, which further provides access to subsets of SNVs of interest. Such subsets, which are represented as `SNVLists`, can, for example, be defined by the user by applying interactive filters (explained later in this chapter) to the *global* `SNVList` containing all SNVs in the data set.

Input File Formats REVEAL can be used to visualize the results of a GWAS analysis conducted e.g. by GATK for the genotyping, or an eQTL analysis conducted by PLINK. Therefore, REVEAL offers two different ways to import data into the application, either in the variant call format (`VCF` for GWAS analyses) or in the PLINK specific file format (for eQTL analyses), where the data is distributed over multiple different files (`DAT, MAP, PED, LOC, ASSOC`). The `VCF` file format is based on variation data only and its specification does therefore not offer the possibility to store gene expression data or any kind of association data. Thus, when importing SNV information in the `VCF` file format only GWAS based analyses can be performed in REVEAL. The second file format is based on eQTL analyses and provides SNV information as well as gene ex-

pression data. Optionally, single-locus or two-locus associations can be added. Thereby, the `DAT` file contains gene expression levels for each gene and individual together with meta-information regarding the clinical phenotype (e.g. healthy or diseased) of the respective individual. The `MAP` file contains additional SNV information, such as the SNV identifier, the exact chromosomal location and optionally the identifier of the gene in the `DAT` file to which the SNV is closest in proximity. The `PED` file contains the genotype information for each individual and SNV. Genotypes are thereby listed in the same order as the SNVs in the `MAP` file. In addition, a `LOC` file can be provided that contains genomic locations for the genes in the data set. SNV associations with gene expression values are typically stored in so-called `ASSOC` files. These files contain SNV–gene or SNV-pair–gene expression associations together with statistical values of the respective test statistic used to calculate the corresponding association. Furthermore, meta-information in tab-separated text format can be imported, including statistical test results, gene annotations, or linkage disequilibrium correlations. Lastly, genome information can be included. To this end, two file formats are supported, namely the `FASTA` format for sequence information and the `GFF3` format for genome sequence annotations.

REVEAL Snapshot An elementary feature of each analysis software is to save intermediate as well as final analysis results and to load these again when needed. MAYDAY already offered the so-called MAYDAY Snapshot, which is a compressed text based representation of the data objects in the current `DataSet`. In REVEAL this concept has been used to enhance the MAYDAY Snapshot with the data that is specific to REVEAL. Thus, each data structure representing a specific data type that needs to be stored has to implement two different methods, in particular a `serialize` as well as a `deserialize` method. Within the `serialize` method the content of the respective data structure object is transformed into a `String` representation, which can then be written to an external text file. Analogously, the `deserialize` method takes as input a `String` representation of the respective fields of the data structure object and restores it. Since the conventional MAYDAY Snapshot is simply a zip-archive containing multiple text files, this strategy of storing information could be extended by adding the REVEAL specific text files to this zip-archive. The advantage of this choice is that snapshots created within REVEAL can not just be loaded with REVEAL, but also solely in MAYDAY. Then, however, only expression data is read from the snapshot, while the REVEAL specific data files remain serialized in the archive. Only when opened with REVEAL all data, i.e. the MAYDAY as well as REVEAL specific files, become deserialized. This offers a comfortable way for saving and loading data for future use.

Filtering Filtering is an essential feature for each data exploration tool. It allows for the reduction of data complexity based on user-defined values for

specific data features. In REVEAL, SNVs are the main data type. However, the number of SNVs in a data set can become very large. As described above, SNVs are organized in `SNVLists`. Thereby, each project in REVEAL has a single global `SNVList`, which contains all available `SNVs` in the data set. Based on this list, filtering methods can be applied to generate new `SNVLists` containing only those `SNVs` satisfying the applied filter criteria. In MAYDAY, the concept of `DynamicProbeLists` has been defined (see chapter 3, section 3.1), which allows the user to build complex filters by combining simple ones. This concept has been transfered to `SNVLists` in REVEAL. In contrast to the `DataProcessor` instances implemented in MAYDAY, SNV-based filters have been realized by `DataProcessors` acting on `SNV` objects rather than `Probes`. These filters then, for instance, act on location data, such as the genomic position of a SNV or any other meta-information available in the respective REVEAL project. A detailed description of all available SNV filters is provided in the Appendix A.1. A typical scenario, where combinations of simple filters are required, is SNV quality control. For quality control of the raw SNV data several criteria have to be fulfilled. In particular, most GWAS require SNVs to have a specific minor allele frequency within a population, in order to avoid the problem of rare alleles being supported only by very few individuals (no power to detect an association). Furthermore, Hardy-Weinberg equilibrium (HWE, see chapter 2, section 2.5.2) is often required to exclude potential SNV calls resulting from sequencing or genotyping errors. For SNVs that are not in HWE it is difficult to distinguish between a real population effect or a genotyping error.

5.1.3 REVEAL's View Model

REVEAL makes use of the visualization concepts introduced in chapter 3, section 3.3 to extend MAYDAY's visualization framework for the analysis of SNV related data. By following this approach, visualizations within REVEAL are linked to each other on the gene level. Furthermore, visualizations can be linked to MAYDAY specific plots, since in REVEAL genes are internally treated as `Probe` objects. This allows for a smooth communication between the two applications. Moreover, the extension of MAYDAY's view model concept offered new opportunities for linking visualizations in REVEAL. In particular, the introduction of `SNVs` within the view model enabled to link visualizations in REVEAL on the SNV level, such that for example SNV selections are synchronized between plots. In particular, all visualizations that belong to the same REVEAL project share the same view model and SNV selections in one plot are immediately reflected in all the others. In addition, the selection of data objects is passed on from the view model to the data model. This offers the possibility to use the information about SNV selections to filter SNVs and apply subsequent statistics or to create new visualizations with respect to the selected SNVs.

5.2 Statistics and Visualizations for Case/Control based Genome-Wide Association Studies

Case-control studies are a popular approach for the identification of associations between SNVs and binary phenotypic traits, such as an individual's disease status. In order to identify SNVs of interest that show a significant association with the disease phenotype, statistical tests, such as the χ^2-test, or Fisher's exact test, are applied. In chapter 2, section 2.5.2 details are provided on how statistical significance for an association between a specific SNV and a disease phenotype can be calculated with various statistical tests. These tests are all based on the comparison of two cohorts, namely affected individuals and unaffected ones. In order to interpret statistical significance, appropriate visualizations are often needed. One very simply, but effective visualization is, for example, the Manhattan plot, where the negative logarithm of a p-value from a statistical test is plotted against a corresponding SNV location in the genome. In the following, the statistical tests available in Reveal are presented at first and subsequently, new advanced visualization approaches are described that allow for the visualization of the statistical test results.

5.2.1 Statistics in Reveal

Reveal provides different opportunities to either calculate statistical tests directly within Reveal, or for more computationally expensive calculations, to import the respective results. The most widely used toolkit for GWAS as well as eQTL statistical testing is PLINK [129]. Consequently, Reveal provides import functionality for various different PLINK result tables. A full list of the statistics that are available in Reveal, as well as those for which results can be imported from PLINK is given in table 5.1.

Furthermore, multiple testing correction is very important for GWAS as well as eQTL studies, since usually hundreds of thousands of single association tests have to be performed. Reveal offers various different multiple testing correction methods. A full list of all available methods is provided in table 5.2. Due to the integration of Reveal into Mayday, multiple testing correction methods available for statistics in Mayday can also be used within Reveal. However, as discussed in chapter 2, section 2.5.3, a permutation based correction method is usually the best choice for GWAS data. Thus, this method has been added in Reveal.

Table 5.1: Overview of the statistical methods available in REVEAL and those for which results can be imported from PLINK.

Statistical Test	REVEAL	PLINK
χ^2-Test	✓	✓
Fisher's Exact Test	✓	✓
RelativeRisk	✓	
OddsRatio	✓	
Difference of Proportions	✓	
Hardy-Weinberg Test	✓	✓
Armitage Trend Test	✓	✓
Linear Logistic Models		✓
Likelihood Ratio Test		✓
Wald Test		✓

5.2.2 Visualization of Genotypes and Statistics

To visualize the result of statistical tests, such as the χ^2-test for binary phenotypic traits or equivalent methods for quantitative traits, REVEAL offers the typical Manhattan plot as well as a tabular view of the statistical test results. Furthermore, the SNV Summary plot provides additional information on genotype distributions and individual genotypes. These visualization approaches will be described in the following.

Enhanced Manhattan Plot Manhattan plots are a commonly used technique for the visualization of statistical p-values for variation data. These

Table 5.2: Overview of the available multiple testing correction methods in REVEAL. Methods from MAYDAY are also available in REVEAL.

Multiple Testing Correction Method	REVEAL	MAYDAY
Bonferroni		✓
False Discovery Rate (FDR) - Benjamini Hochberg		✓
FDR (Benjamini-Yekutieli)		✓
Holm's		✓
FDR (Storey)		✓
Permutation Test	✓	

are scatter plots, where the genomic position of a SNV is plotted against the $-\log_{10}(p)$ value of a statistical test. Thus, significant associations of SNVs and a respective phenotype can easily be spotted, since these are the largest values in the Manhattan plot. In REVEAL's implementation of the Manhattan plot, additional user interactivity is offered. Firstly, SNVs satisfying a user-defined p-value threshold can be highlighted in the visualization using a different color. Secondly, gene locations can be visualized, which are represented as boxes with their width defined by the length of the respective gene. Cis- and trans-acting SNVs can also be highlighted. For this a custom range of genomic coordinates, for which SNVs are classified as cis-acting, can be defined. SNVs located within a gene, as well as 5' upstream or 3' downstream of a gene within the user-specified range are then drawn with a different color. Lastly, the selection of SNVs is possible, which is synchronized with other visualizations in REVEAL and can additionally be used for filtering (see section 5.1.2).

In order to compare statistical test results, for either binary or quantitative traits, multiple instances of Manhattan plots can be grouped within the same visualization tab (see figure 5.2). This allows for comprehensive comparisons between different statistical tests. An example would be the comparison of a set of SNVs based on their association with a number of different genes. For each gene, an individual Manhattan plot can be created. These are then arranged underneath each other, rather than in separate visualization tabs, to allow for a convenient comparison of significant influences of SNVs on these genes.

Statistical Results Table In addition to the Manhattan plot, the p-values can be inspected simultaneously in a tabular view. In this meta-information table, the first column corresponds to SNV identifiers and the p-values from the calculated statistical tests are shown in adjacent columns. Rows can be sorted according to the values in the columns in ascending or descending order. Furthermore, columns can be rearranged manually by the user. This allows one to compare results from different statistical tests or for different genotypic models. Lastly, rows can be selected in the table, which results in a selection of the corresponding SNVs, which are then highlighted in all connected visualizations, such as the Manhattan plot.

SNV Summary Plot The SNV Summary plot offers the possibility to explore SNV distributions in more detail. It combines visualization of statistical test results with different tracks, as for example for the comparison of SNV genotype distributions between two cohorts, i.e. affected and unaffected individuals. Hence, conclusions about a possible connection of a genotype and a disease state can be made more easily. In the SNV Summary plot

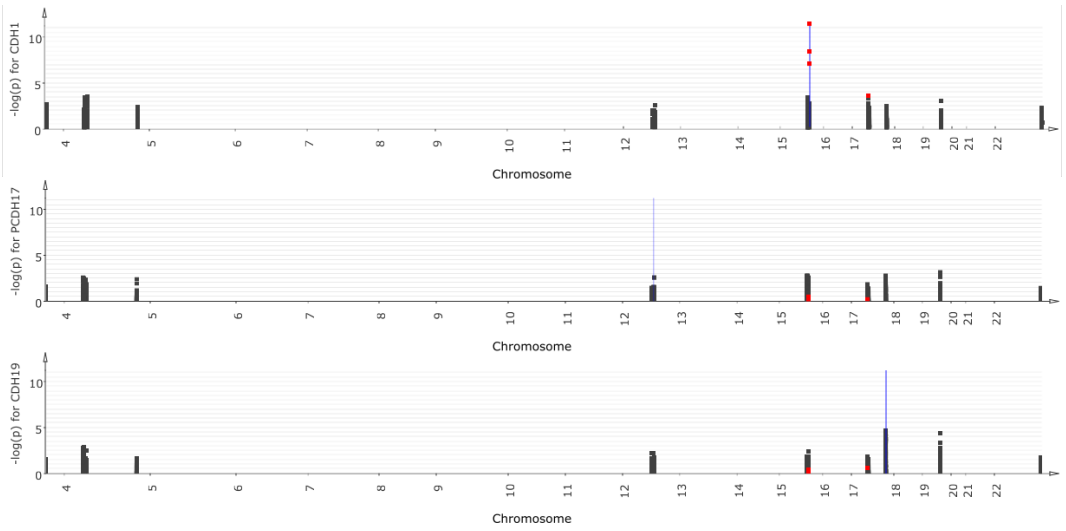

Figure 5.2: Example of multiple connected Manhattan plots in REVEAL. SNVs with a single-locus association p-value ≤ 0.001 for the CDH1 gene are highlighted in red in all Manhattan plots. Corresponding gene regions for each plot are shown with blue boxes.

SNVs are shown in columns. For the rows six different tracks can be added and interactively be switched on or off. An overview of the different tracks is shown in figure 5.3. These are described in more detail in the following starting with the track at the top.

First of all, a corresponding SNV identifier is shown for each column. This can either be an #rs number from dbSNP if available, or the chromosomal location of the SNV in the format `CHROMOSOME:POSITION`. In the second track, p-values from a statistical test are displayed as a bar chart. To enhance visual clarity, the corresponding $-\log_{10}(p)$ value is shown rather than the original p-value. Consequently, larger bars correspond to more significant SNVs.

In the third track, the cohort genotype distribution is shown, both for affected as well as unaffected individuals. For this, each column is separated into two sub-columns. The left sub-column represents affected individuals, and the right sub-column unaffected ones. Each sub-column shows a stacked bar chart, where each bar represents the genotype frequency of the respective genotype in the affected and unaffected cohort, respectively. To allow for a better comparison of the genotype distribution, corresponding genotype bars are aligned horizontally. Furthermore, genotypes are represented by different color encodings. Homozygous genotypes are encoded with solid colors, whereas hashed fills are used for heterozygous genotypes.

The fourth row track is an aggregated display of the cohort genotype distribu-

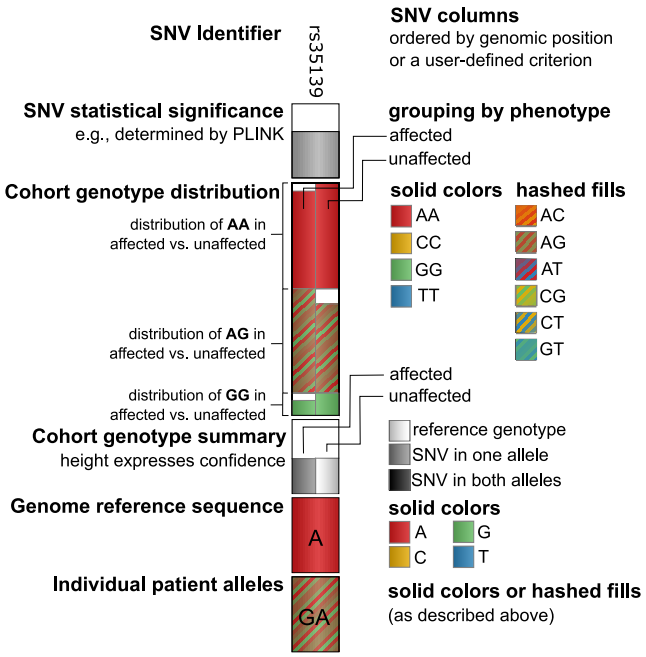

Figure 5.3: A single column of the SNV Summary plot showing the five possible tracks. From top to bottom these are: a SNV identifier (either an #rs number or the chromosomal location), a bar diagram showing statistical p-values, a cohort genotype distribution diagram, which is a stacked bar chart separated into two sub-columns (the right column for affected and the left for unaffected individuals), a cohort genotype summary track, which is an aggregated view of the cohort genotype distributions, a genome reference track, showing the reference base according to the underlying genome sequence, and an individual genotype view, where genotypes for a user-selected individual can be displayed.

tions shown in the third track. Here, the same aggregation strategy is used as for genotypes in the INPHAP software described in chapter 4, section 4.1.2. In summary, the majority genotype is displayed as a bar and a gray-scale encoding is used to represent homozygous SNVs (black), heterozygous SNVs (gray), and no variation in any of the two alleles (white). The frequency of the respective majority genotype is represented by the bar's height. Again sub-columns are used to display affected and unaffected individuals separately. Boxes are displayed next to each other by default, which allows for a comprehensive comparison. This representation can also be changed to a stacked view, which is more appropriate for zoomed-out overviews, since differences in the genotype composition between the affected and unaffected cohort can quickly be spotted, considering the fact that the comparison of the respective genotype frequencies becomes more difficult. However, for zoomed-out overviews details on genotype frequencies are usually of minor interest.

The fifth track shows the reference allele, where each of the four possible bases has been assigned a unique color. Thereby, these colors match with the colors chosen for the genotype distribution bar chart, which allows for a convenient correlation of these tracks. Finally, the bottom track can be used to display a specific individual's genotype. This offers the possibility to compare genotypes from individuals of interest with the genotype distributions of the whole population. This may, for example, help to determine disease susceptibility of individuals with an unclear phenotype.

Interaction possibilities with the SNV Summary plot include: zooming, scrolling, selection of SNVs, changing visual representations of specific tracks, as well as changing the order of the columns. For the latter, the default ordering is given by the chromosomal location of each SNV. However, this order can be adjusted, for example based on the majority genotype, or the statistical p-value. Furthermore, tool-tips for each column in each track provide additional details about the respective view.

5.3 Linkage Disequilibrium Block Visualization and Calculation

For SNVs that are located in close proximity to a gene, or even located inside a gene, it is very likely to observe low p-values in an association test. Especially when studying effects of epistasis (see chapter 2, section 2.5.4), Linkage Disequilibrium (LD) can lead to an over-representation of SNV-pairs that were built from SNVs in LD. Thus, disregarding LD structure during an association analysis can lead to false conclusions. Consequently, including LD information for the analysis of single-locus as well as two-locus association studies can provide valuable insights into the data. Software packages, such as PLINK [129] offer methods to calculate LD correlation values (r^2 values for SNV-pairs). However, the results do not include information about LD blocks. These are subsets of SNVs, where all pairwise combinations share a high r^2 correlation value, with respect to a user-defined threshold. Thus, the identification of LD blocks is important for subsequent analyses.

A typical approach is to visualize all pairwise SNV r^2 correlation values in a so-called LD plot. This visualization is similar to a correlation matrix, where only one half is filled and then rotated by 45°. Usually, color gradients are used to enhance the visual assessment of the respective correlation values in each cell. For a more detailed explanation of the LD plot see [7]. Figure 5.4 shows an example of the LD plot implementation in REVEAL. There, LD blocks can be defined manually by selecting ranges of SNVs for which all pairwise SNV r^2 values satisfy a user-defined r^2 threshold. One such LD block

selection is highlighted in figure 5.4 with dark cyan borders surrounding the cells of the respective sub-matrix. In addition, the SNV identifiers are colored in the same way.

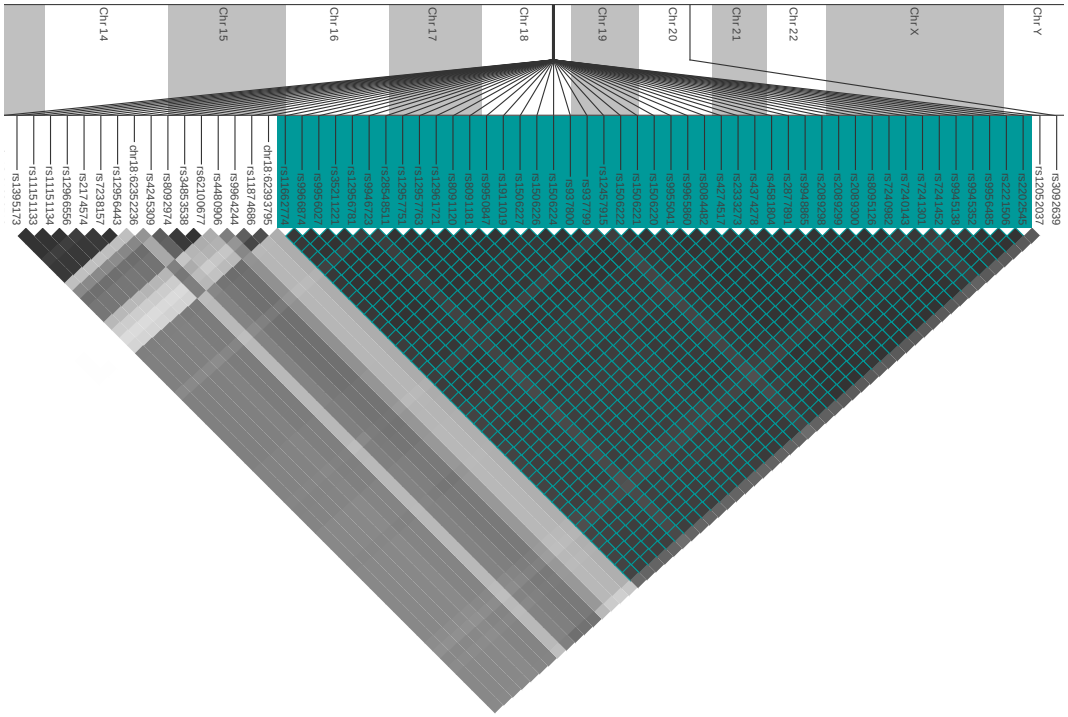

Figure 5.4: Example of the LD-Plot implemented in Reveal. A white-black color gradient is used to encode pairwise r^2 correlation values. A manually selected LD block is highlighted in dark cyan.

Nevertheless, the number of LD blocks can become very large (up to several thousands for a whole genome). This makes it infeasible to define all possible LD blocks manually using visual inspection. Hence, automated procedures for the identification of LD blocks are needed to assist the visual determination strategy. Clustering approaches based on the r^2 correlation value can be applied to group SNVs satisfying a user-defined quality criterion. Many clustering techniques exist, but most of them require the estimation of the number of cluster preliminary to the actual analysis. Although these methods are usually very fast, they are mostly not applicable to variation data, since the determination of the final number of clusters is challenging.

A possible solution is provided by the `QT_Clust` algorithm [59], which was developed for the clustering of gene expression data. The advantage of this method is that there is no need to define the number of clusters preliminary to the actual clustering. In fact, only a threshold for the quality of a cluster, de-

fined by the maximum distance of elements contained in it, is required and the final number of clusters is defined by the algorithm. An improved implementation with respect ot the overall runtime has been introduced in MAYDAY [70] for the analysis of gene expression data. Furthermore, Sebastian Nagel introduced parallelization of the procedure during his Bachelor thesis [110]. Hence, this algorithm is perfectly suited for the identification of LD blocks, requiring only little modifications, in order to make it applicable to SNVs. For cluster quality calculations, $1 - r^2$ is used as a distance measure.

5.4 SNV Annotation and Effect Prediction

Depending on the location of a SNV, it can have severe effects on a gene's functionality. For instance, SNVs located inside a gene can lead to amino-acid changes and consequently to a potential loss or modification of gene function. Furthermore, SNVs outside of a gene, but in close proximity to its 5' end can lead to reduced binding affinity of transcription factors, which influences the expression level of the affected gene. There are several different tools that allow for the prediction of SNV effects, as for example ANNOVAR [167], VAAST [65], or SNPeff [25]. Of these, SNPeff is by far the most widely applied SNV effect prediction solution. It provides information on SNVs on the basis of annotated genes, or other genomic elements. This includes the classification of a SNV into synonymous or non-synonymous, as well as start and stop codon gains or losses. Furthermore, annotation of SNVs regarding their genomic location can be made, such as intronic, 5' untranslated region (UTR), 3' UTR, upstream, downstream, or intergenic. SNPeff makes predictions starting from a `VCF` file and outputs results as an annotated `VCF` file.

In REVEAL, SNV effect information can be included in the analytical process. This can be achieved by importing SNVs together with their corresponding SNPeff results in the form of an annotated `VCF` file. The effect predictions will then be attached to the SNVs in the form of SNV meta-information data. Furthermore, if an annotated `VCF` file is provided during the creation of a new REVEAL project, the effect information will be imported automatically. However, SNPeff does only allow one subject per `VCF` file. For a typical GWAS data set, this requires to perform multiple SNPeff predictions and to import each of these into REVEAL individually. To overcome this hindrance, REVEAL offers the possibility to calculate SNV effects directly within itself. The contained implementation makes use of the original SNPeff prediction strategy, but in addition, it is able to perform multiple calculation in parallel for a user-defined subset of individuals. In order to use this feature, gene annotations in the `GFF3` [153] file format have to be imported first. REVEAL can then predict SNV effects either only for the genes included in the current project,

or based on all genes annotated in the `GFF3` file. In addition to the SNPeff based predictions, REVEAL summarizes the effect categories into four impact classes. These are *3=high* (very likely to have an impact), *2=middle* (probably has an impact), *1=low* (unlikely to have an impact, but still possible), and *0=none* (no impact). A full list of categories and their assigned impact classes is provided in table A.3 in the Appendix. The impact classes allow for a quick identification of interesting SNV effects, when combined with appropriate visualizations. Within REVEAL this is possible by using the SNV Effect Table described in the following.

5.4.1 SNV Effect Table

The SNV Effect Table offers a comfortable solution to explore SNV effect predictions within REVEAL. The table consists of multiple rows, which represent SNVs in the data set and several different columns, that provide annotations for each SNV. An example of the SNV Effect Table is given in figure 5.5 showing all the available columns. User interaction with the table comprises selection and sorting of SNVs based on values from a specific column, as well as highlighting fields of interest using color. For example, cells from the SNV impact column are colored red (for high impact), orange (for middle impact), and green (for low and no impact) by default.

Subject ID	Subject Na...	SNP	Reference N...	Genotype	Chromosome	Strand	Frame	SNP Position	Target Element	Target Name	Target Start	Target End	Class	Non-Synonymous	Impact	AA Change
1:1		rs35139	A	G|A	16	-	0	64991772	gene	CDH11	64980683	65155919	5' UTR Modifier	NO	2	
2:2		rs35139	A	G|A	16	-	0	64991772	gene	CDH11	64980683	65155919	5' UTR Modifier	NO	2	
3:3		rs35139	A	A|A	16	-	0	64991772	gene	CDH11	64980683	65155919	5' UTR	NO	2	
4:4		rs35139	A	A|G	16	-	0	64991772	gene	CDH11	64980683	65155919	5' UTR Modifier	NO	2	
5:5		rs35139	A	A|A	16	-	0	64991772	gene	CDH11	64980683	65155919	5' UTR	NO	2	
6:6		rs35139	A	A|A	16	-	0	64991772	gene	CDH11	64980683	65155919	5' UTR	NO	2	
7:7		rs35139	A	A|A	16	-	0	64991772	gene	CDH11	64980683	65155919	5' UTR	NO	2	
8:8		rs35139	A	A|A	16	-	0	64991772	gene	CDH11	64980683	65155919	5' UTR	NO	2	
9:9		rs35139	A	A|G	16	-	0	64991772	gene	CDH11	64980683	65155919	5' UTR Modifier	NO	2	
10:10		rs35139	A	G|G	16	-	0	64991772	gene	CDH11	64980683	65155919	5' UTR Modifier	NO	2	

Figure 5.5: Example of the SNV Effect Table for the visualization of SNV effect predictions with REVEAL. The effect of the SNV with dbSNP ID `rs35139` is shown for the genotypes of 10 different individuals.

5.5 Visual Genotype based Expression Analysis

Visual screenings of SNV associations with gene expression levels can be performed by comparing the distributions of the expression levels for the three possible genotypes in a population. If there is a significant association, then it is expected that this difference is reflected in the genotype distributions. A common strategy to visually assess differences between distributions are box plots, which have been widely applied in previous studies [41, 172].

5.5.1 Visualization for SNV Associated Gene Expression Differences

When using box plots to show genotype distributions, typically three different boxes are drawn, one for each possible allele combination. Then one can compare the median expression levels as well as the quantiles of the three allele combinations. Under the assumption of an additive model, one would for example expect the expression level of a gene associated with a homozygous SNV to be twice as high/low in comparison to the reference alleles, as it would be for the heterozygous genotype. Furthermore, if the population can be separated into two groups based on a phenotypic trait, such as susceptibility to disease, changes between the groups can also be assessed. An example of REVEAL's genotype box plot visualization is shown in figure 5.6. There, the genotype distributions for an affected and for an unaffected sub-population are compared to each other. One can clearly see that there is a significant increase in the expression level distributions of the homozygous SNV in comparison to the reference alleles for the affected group, whereas this association cannot be seen in the control group.

Although this visualization can be used to show differences between a specific SNV–gene pair, it becomes infeasible for larger numbers of genes or SNVs. If the number of SNVs in a data set is n and the number of genes is m, then $n \times m$ different box plots would have to be generated and visually inspected in order to assess the full information content. This number doubles if cases and controls have to be compared additionally. Thus, a more appropriate strategy that scales well, even for large data sets, is required. To address this issue, a data transformation approach, which can be used together with traditional gene expression visualizations, such as a heat map, has been developed.

5.5.2 SNV Derived Expression Level Transformation

To address the scalability issue of box plots for the visualization of SNV associated changes in gene expression, a new data transformation technique is introduced in this thesis, which can be used together with well-established gene expression visualizations. The basic idea is to transform the information gained by box plots into a single expression value, which can then be used for visualization. The following formula describes how genotype associated expression value distributions can be used to calculate such a single expression value $v_{\mathcal{C}}$ for a specific SNV s and gene g within a population $\mathcal{C}$.

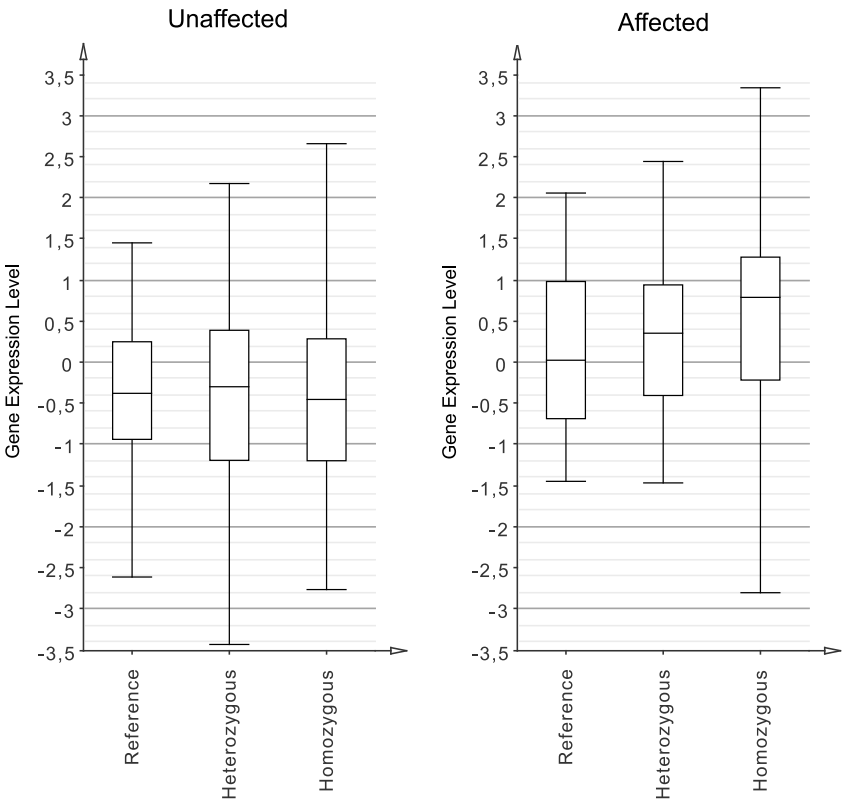

Figure 5.6: Example of two genotype box plots for an affected and unaffected sub-population, demonstrating the influence of a SNV to a genes expression level. On the right side, the gene expression level distributions associated with the SNV are shown for the affected group. Here, an increase in the expression level for the homozygous SNV can be observed. On the left, the gene expression level distributions for the unaffected group do not show such differences as for the affected group.

$$v_{\mathcal{C}}(s,g) = x\frac{\sum_{i\in\mathcal{C}^s_{ref}} exp_g(i)}{|\mathcal{C}^s_{ref}|} + y\frac{\sum_{i\in\mathcal{C}^s_{het}} exp_g(i)}{|\mathcal{C}^s_{het}|} + z\frac{\sum_{i\in\mathcal{C}^s_{hom}} exp_g(i)}{|\mathcal{C}^s_{hom}|} \tag{5.1}$$

$$= x \cdot mean_g(\mathcal{C}^s_{ref}) + y \cdot mean_g(\mathcal{C}^s_{het}) + z \cdot mean_g(\mathcal{C}^s_{hom}) \tag{5.2}$$

In this formula $\mathcal{C}^s_{ref}$ corresponds to the sub-population of $\mathcal{C}$, for which the corresponding individuals carry the reference base on both alleles for s. $\mathcal{C}^s_{het}$ and $\mathcal{C}^s_{hom}$ are defined similarly for individuals with a heterozygous and those with a homozygous SNV s. The coefficients x, y, and z can be used to weigh each

term with respect to a genotypic model. For example, under the assumption of an additive model, one would expect a linear relationship between the mean expression in the heterozygous group and the mean expression in the homozygous group. The mean expression value for the reference group can then be considered as the ground expression level against which comparisons should be made. Thus, the coefficients for an additive model would be defined as follows:

$$x = -1 \tag{5.3}$$

$$y = \frac{mean_g(\mathcal{C}^s_{hom})}{mean_g(\mathcal{C}^s_{het})} \tag{5.4}$$

$$z = 1 \tag{5.5}$$

Analogously, values for the coefficients can be defined for other genotypic models. Furthermore, in the formulas 5.2 and 5.4 the mean expression value of g can be replaced with the corresponding median expression if needed, which is more robust to outliers for small populations. The values $v_{\mathcal{C}}$ may be directly used for visualization in, e.g. a heat map. Furthermore, if the population $\mathcal{C}$ can be divided into two sub-populations $\mathcal{C}_{case}$ and $\mathcal{C}_{control}$, then fold-change values between these sub-populations can be calculated by applying formula 5.2 individually for $\mathcal{C}_{case}$ and $\mathcal{C}_{control}$. The respective fold-change (FC) for $\log_2$-transformed expression values is then given by:

$$FC_{\mathcal{C}_{case},\mathcal{C}_{control}}(s,g) = v_{\mathcal{C}_{case}}(s,g) - v_{\mathcal{C}_{control}}(s,g) \tag{5.6}$$

In REVEAL, calculations of $v_{\mathcal{C}}$ or FC values can be performed for all pairwise combinations of a chosen set of SNVs and a defined list of genes. The respective values are then subjected to MAYDAY for the visualization with MAYDAY's gene expression based plots. By default, SNVs are organized in rows and genes in columns. A SNV derived heat map example applied to data from the BioVis 2011 challenge data set is provided in section 5.9.2 figure 5.13.

5.6 Single-Locus Association Visualization

The discovery of SNVs associated with quantitative traits, especially gene expression values, is of great interest to clinicians as well as geneticists. Although, statistical approaches for the identification of significant associations within a population exist, the interpretation of the results from such statistical tests remains challenging. The most widely used toolkit allowing to make predictions for such associations is PLINK [129]. However, results from PLINK are

provided in text-based formats, rendering it difficult to gain meaningful insight into the data. Moreover, appropriate visualizations for quantitative trait associations are missing, forcing researches to deal with the data by using spreadsheet programs, such as Microsoft Excel. However, this can be very challenging and time consuming, especially when results from different tests need to be combined. Thus, in this work visualization approaches have been developed to address the need for more appropriate visual analytics techniques for the study of eQTL data.

5.6.1 Association Table

To represent single locus QTL associations in a tabular form, the meta-information table introduced for case/control based statistical test results (see section 5.2.2) can also be used to display the values from a PLINK based association test statistic. Hence, SNV identifiers are displayed in the first column of the table. The other columns are used to show the different values from the Wald test calculated in PLINK. A summary of the available columns is provided in table 5.3. All other features presented for the meta-information table, such as row selection, or row/column reordering, remain unchanged and can therefore also be used in the Single-Locus Association Table.

Table 5.3: Overview of the available columns in the Single-Locus Association Table. The presented column ordering corresponds to the default ordering in REVEAL. The statistical values are based on PLINK `QASSOC` result files [129].

Column	Value	Description
1	SNV ID	Either #rs number, or genomic location
2	P	Wald test asymptotic p-value
3	T	Wald test (based on t-distribution)
4	BETA	Regression coefficient
5	R^2	Regression r^2 value
6	SE	Standard error

5.6.2 Association Network for Single-Locus Association Results

The Single-Locus Association Network provides a graph-based visual representation of SNV associations with eQTLs by using the Jung library [114] for graph construction. In this graph, nodes represent genes from the data set. Furthermore, SNVs in the data set are assigned to the gene, which is closest in proximity, with respect to a user-defined maximum distance. Consequently, a SNV can be represented by a gene using this strategy. SNVs

that are not assigned to any gene are removed. With this approach, edges can be introduced in the graph as follows. If there is a significant association between a SNV s and the expression of a gene g_j, where s is located in close proximity to g_i, then a directed edge $e = (g_i, g_j)$ is drawn between the nodes representing the genes g_i and g_j. Clearly, the direction of the edge is dictated by the corresponding SNV association. Note that $i = j$ is possible, if a SNV is associated with the gene it has been assigned to. Moreover, there can be multiple SNVs located in close proximity to the same gene g_i, which are all associated with the expression level of the same gene g_j. To address this circumstance, edges are drawn with varying thickness, where the thickness of an edge is proportional to the number of different SNV–gene associations. Alternatively, the cumulative p-value of the underlying statistical tests for each SNV can be used to encode edge thickness. If the latter is chosen, the $-log_{10}(p)$ value is used, which results in thick edges, if there is a significant association and thin edges otherwise. Since the number of edges in such a graph can increase quickly when there is large number of significant SNV associations, the user can interactively filter edges based on their edge weight. This can be achieved by either defining a threshold on the number of different SNVs needed to draw an edge, or by defining a threshold for the p-value of the underlying test statistic. Consequently, only those edges are shown that satisfy the user-defined threshold, thus offering the possibility to concentrate only on very prominent associations.

In addition, gene expression values can be mapped to the nodes using a predefined color gradient. Nodes are then colored based on the $\log_2$ fold-change expression value between the affected and unaffected sub-population. This allows to quickly spot SNV associated expression changes in this node-linked graph. Further interaction possibilities include zooming, panning, rotating, as well as interactive selection of nodes and edges. If nodes are selected, they are highlighted with a colored border, where the color can be defined by the user. For selected edges, the edge color is changed from black (the default edge color) to a user-defined selection color. If an edge gets selected in the Single-Locus Association Network, all SNVs represented by the respective edge are selected simultaneously. Based on these selections, for example, a new `SNVList` can be created for a subsequent analysis or for visualization in other plots. An illustration of the Single-Locus Association Network is given in figure 5.7.

5.6.3 Association Matrix

In contrast to the Single-Locus Association Network, a matrix like visualization does not suffer from clutter and cannot become a hairball when the data set is large. However, it has the disadvantage of a less intuitive representation of gene connections. In order to offer the advantages of both visualizations,

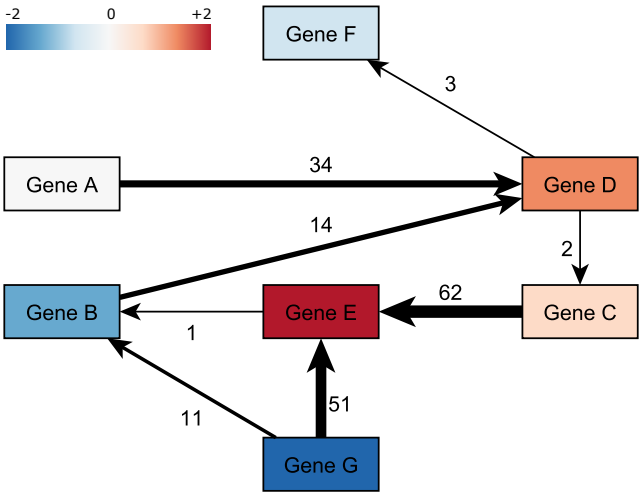

Figure 5.7: Illustration of a Single-Locus Association Network in Reveal. Gene expression log2 fold-changes have been mapped to nodes using a blue-white-red color gradient.

a Single-Locus Association Matrix has been developed to complement the network representation. The matrix is divided into two parts, an upper part showing gene expression data and a lower part, for the visualization of SNV counts or statistical values of an eQTL association test. For the upper, as well as the lower part, columns represent genes. The upper part shows gene expression values summarized with respect to the disease phenotype. Furthermore, there are four different summarization methods available, namely MIN, MAX, MEDIAN, and MEAN. Each cell in the upper part encodes for a specific cohort based gene expression value using a user-defined color gradient.

In the lower part of the matrix, rows correspond to genes for which SNVs are in close proximity. In each cell, circles of variable diameter and color are drawn. With this strategy two different values from the corresponding eQTL association test can be encoded. Circle size is used to either display the number of SNVs associated with the gene in the column, or to show the $-\log_{10}(p)$ value of the statistical test. Thus, the larger a circle, the more significant is its association with the respective gene expression value. In addition, a color-gradient for the circles is used to encode for the direction and size of the genetic effect. These values correspond to the regression coefficient of the underlying likelihood ratio or Wald test used to calculate statistical significance of the associations. For more details on statistical testing for quantitative traits, see chapter 2, section 2.6.1. By default, a blue–white–red gradient is used. Red color corresponds to a positive effect (an increase in gene expression), whereas blue color indicates a negative effect (a decrease in gene

expression) [38]. The magnitude of the regression coefficient is proportional to the effect size. Consequently, the saturation of the circle indicates the strength of the respective effect. Thus, white circles have no effect, since their regression coefficient equals 0. When drawn on a white background, the circle becomes invisible in such cases, which is mostly the preferred representation, since low p-values have no meaning, if there is no measurable effect. Alternatively, the default color gradient can be changed by the user, in order to indicate such cases. An example of the Single-Locus Association Matrix, derived from the BioVis 2011 eQTL data analysis challenge, is shown in figure 5.8.

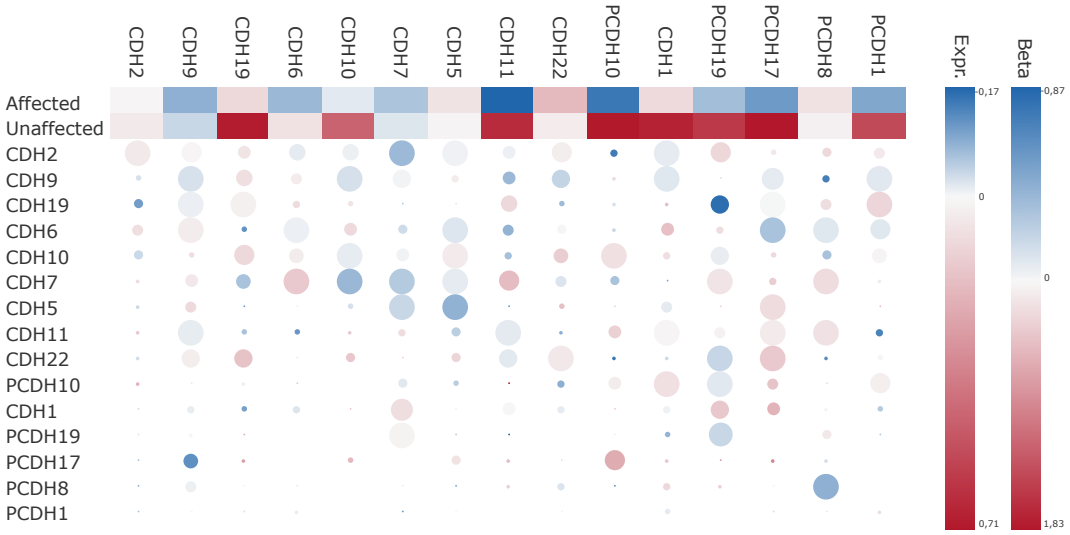

Figure 5.8: Example of the Single-Locus Association Matrix implemented in Reveal. This example has been created with eQTL data from the BioVis 2011 challenge data set. The plot is shown with default settings for the color gradients and the gene ordering.

Clearly, if each SNV is assigned to the gene in closest proximity, as in the Single-Locus Association Network, this can lead to an assignment of multiple SNVs to the same gene. Hence, if the user chooses to display p-values instead of SNV counts, the circle size is defined by the mean p-value of all SNVs assigned to the same gene with an association to the gene represented by the respective column.

Interaction possibilities with the Single-Locus Association Matrix include: interactive filtering of SNVs based on a user-defined threshold for the p-value, selection of cells, which results in a selection of the corresponding SNVs, and lastly rows of the lower part of the Single-Locus Association Matrix can be sorted according to different criteria. By default, rows are arranged such that

SNVs that show large numbers of significant associations with the genes in the columns are presented on top. However, sorting by chromosomal location is also possible.

5.7 Two-Locus Association Visualization

Although, single-locus association analyses already provide good indications of genome sequence variations affecting phenotypic outcomes and especially gene expression levels, complex diseases, such as specific cancers, may involve networks of interacting variations. In such cases, gene expression levels are not affected obviously by single SNVs, but by the interplay of various different variations. The presence and severity of a disease phenotype is thus conditional on the presence or absence of specific SNV combinations. This requires the identification of multi-locus associations. However, the computational effort increases exponentially with the number of SNVs that are investigated at once, rendering it difficult to calculate associations based on more than two or three SNVs. Consequently, the typical approach is to concentrate on two-locus associations, which already means to analyze all possible pairwise combinations of SNVs within a data set. Again software solutions, such as PLINK, provide statistical methods to assess the significance of an association of a SNV pair with a quantitative trait (see chapter 2, section 2.6.2 for more details). However, the interpretation of the results of such epistatic effects is even more complex than with single-locus associations. Thus, appropriate visual analytical approaches become even more important when studying epistasis. In Reveal, three different visualizations are available to address epistatis, which will be explained in more detail in the following.

5.7.1 Association Table

Similarly to the Single-Locus Association Table, the meta-information table can be used to represent statistical values from an epistasis analysis conducted with PLINK [129]. This table, however, contains two columns with SNV identifiers, one for each SNV in a SNV pair. Further columns are used to represent the statistical values from the respective pairwise SNV association tests. Table 5.4 shows all available columns. Furthermore, selection of rows can no longer be uniquely mapped to a specific SNV, since rows represent SNV pairs in this table. Thus, in the Two-Locus Association Table both SNVs of a SNV pair get selected. All other interaction possibilities introduced for the meta-information table remain unchanged.

Table 5.4: Overview of the available columns in the Two-Locus Association Table.

Column	Value	Description
1	SNV ID 1	Identifier for the first SNV in the SNV pair
2	SNV ID 2	Identifier for the second SNV in the SNV pair
3	BETA	Coefficient for SNV interaction
4	STAT	χ^2-statistic with 1 degree of freedom
5	P	Asymptotic p-value

5.7.2 Association Network

The Two-Locus Association Network provides a graph-based visualization of SNV-pair associated gene expression changes. To construct the graph, the Jung library [114] has been used. In this graph, each gene is represented by a node. Furthermore, SNVs are assigned to genes based on their chromosomal location. Thus, for each SNV the gene, which is closest in proximity, is identified based on a user-defined maximum distance. This approach is similar to the Single-Locus Association Network described above. It enables to determine SNV pairs, where the two SNVs are assigned to different genes. Nodes representing genes, for which such SNV-pairs exist are assigned a unique color, while the other nodes are colored gray. Colors are chosen based on ColorBrewer color maps [52], if less then 12 color values are needed, and based on a rainbow color gradient otherwise. Edges between the nodes are constructed following a simple strategy. If there is at least one SNV pair showing a statistically significant association (e.g. based on PLINK results) with one of the gene expression values from the data set, then an edge between the corresponding genes is drawn. However, the SNV pair can influence any of the genes in the data set. Thus, edges are colored based on the gene whose expression is influenced. This means that for each edge in the graph a gene triple (g_i, g_j, g_k) is created, where one SNV of the pair is assigned to gene g_i and the other to gene g_j. The edge's color is defined by the color of the gene g_k, whose difference in expression between affected and unaffected individuals is associated with the respective SNV pair. Clearly, there can be more than one such SNV pair for a specific gene combinations. Some of which may influence the same gene g_k. Thus, edge weights are introduced that correspond to the number of different SNV pairs between the genes g_i and g_j influencing the gene g_k. Nevertheless, different target genes are also expected. To address this issue, multiple edges between the same two genes are allowed, which differ in color and possibly also in edge weight. An example graph demonstrating the possible node relationships is shown in figure 5.9. Alternatively, mean p-values of the underlying statistical test can be used instead of

SNV pair counts to encode edge weight. In this case, the same strategy as for the Single-Locus Association Network described in section 5.6.2 is followed.

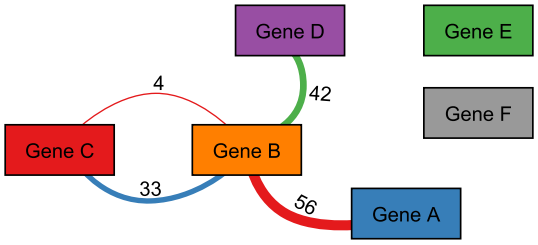

Figure 5.9: Illustration of a Two-Locus Association Network. Nodes represent genes and edges correspond to SNV pairs that are significantly associated with one of the gene's expression level. Each gene to which a SNV from a SNV pair has been assigned to is colored using a unique color value. Genes without any associated SNV-pair are colored grey. Edge colors indicate specific associations between SNV pairs and genes, and edge weights correspond to the number of SNV pairs associated with the same gene. Multiple edges are allowed for cases where SNV pairs from two genes influence more than one additional gene.

Since the number of edges can become very large, additional interaction possibilities are provided in order to increase visual clarity and to explore the visualized data in more detail. First of all, edges can be interactively filtered based on their edge weights. The user can define a threshold τ, such that only edges with weight $w > \tau$ are displayed. Furthermore, nodes can be rearranged manually, or by using layout algorithms provided by the Jung library [114]. Besides general graph interaction features, such as panning, scaling, rotating, zooming, and selection, users can map gene expression values to nodes, resulting in node sizes relative to the mean fold-change between affected and unaffected individuals. In order to make interesting associations more visually prominent, one can activate node or edge highlighting for selections. This means that nodes connected to a selected node are drawn with a black border, and edges drawn with the same color as the selected node are highlighted with increased saturation to make them stand out more clearly. In addition, edges can be selected, which leads to a selection of the corresponding SNV pairs. Based on this selection new `SNVLists` can be created for further processing or for more detailed visual inspections within other plots.

5.7.3 Association Matrix

Although, the Two-Locus Association Network nicely shows the relationships between different genes on the level of their correlated SNVs, it suffers from the same disadvantage as the Single-Locus Association Network. With an increasing number of SNV pairs it can become difficult to spot interesting

patterns due to visual clutter. Thus, interactive filtering has been introduced to reduce the amount of edges that have to be drawn. However, for large data sets, this approach may not always lead to satisfying results. Furthermore, with an increasing number of genes in the network, it becomes more difficult to differentiate between gene and edge colors. A matrix like approach does not suffer from these hindrances and can therefore lead to a more comprehensive understanding of the data, when complemented with a network visualization. Thus, the visualization strategy introduced for the Single-Locus Association Matrix above (see section 5.6.3) can also be used to visualize epistatic effects. This requires only few modifications. Instead of showing single genes in the rows of the matrix, as it is the case for the Single-Locus Association Matrix, gene pairs are displayed for two-locus gene expression associations using row labels in the form of `Gene1:Gene2`. Furthermore, association tests for epistasis estimate three different gene effect coefficients, one for the first SNV in the pair, one for the second one, and an interaction coefficient. The latter describes how the interaction of two SNVs affects the expression of the respective gene. In the Two-Locus Association Matrix, this interaction coefficient is used for SNV effect visualization and is encoded using a blue–white–red color gradient by default. As for the Single-Locus Association Matrix, the size of the circles in each cell either displays the number of different SNV pairs, or the mean p-values of the corresponding association tests.

5.8 Interaction between INPHAP and REVEAL

In chapter 4, INPHAP has been introduced as a tool for the analysis of genotype and phased haplotype data. With REVEAL a step further has been taken to address the need for the integration of genotype data, association data and expression data, in order to identify SNV related factors causal for disease. Although REVEAL does not allow for the direct integration of phased haplotype data or for the correlation of such data with population specific meta-data, a mechanism to link REVEAL to INPHAP has been implemented. As a result, INPHAP can be invoked from within a running REVEAL instance, which allows for the exchange of genotype data and the visualization of such within INPHAP. The data exchange is thereby realized with a so-called `DataLinker`, which holds references to REVEAL's as well as INPHAP's data model. Furthermore, the `DataLinker` contains methods for an online data transformation, such that genotype data from REVEAL is provided in the right format to be visualized within INPHAP. With this strategy the advantages of REVEAL and INPHAP are combined to enable more powerful analyses.

5.9 Application Examples based on the BioVis 2011 and 2012 Challenge Data Sets

5.9.1 The eQTL Biological Data Visualization Challenge

In 2011, the BioVis (Conference on Biological Data Visualization) initiated a new data visualization contest. The idea of this contest was to elevate research on visualizations for specific biological problem domains, with the main goal to increase the development of enhanced applications that address the needs of the biological community. For the BioVis 2011 and the following year 2012, the contest involved the analysis of eQTL data. These data were generated from actual eQTL analysis data, by applying a so-called observation shuffling technique [8]. With this technique it was possible to introduce a hypothetical disease *hoompalitis*. The contest data set was designed around this disease by introducing spiked-in interaction networks in order to establish a ground truth. The advantage of this technique was that the biological complexity of the data was preserved, such that realistic biological conditions for the development of new software solutions were provided. The first challenge data set contained 7555 genomic loci (SNVs) and gene expression levels for 15 genes for a total of 500 different individuals. Furthermore, for each individual a disease state for hoompalitis was provided. These data were accompanied with results from statistical association testing conducted with PLINK for both, single-locus as well as two-locus associations. Based on this information the challenge included the visual identification of the patterns of variations, gene expression levels and their interaction in order to elucidate the impact of these factor for the incidence of hoompalitis. Due to the success of this challenge in 2011, it was repeated in 2012 with a more complex data set comprising 230,912 SNVs, 44 genes and the same 500 different individuals. In addition, data from two sources were available, namely blood as well as tissue samples. Again the question was to identify the biological factors leading to disease. Furthermore, participants were asked to give a prediction whether these factors can only be found in the tissue samples or if a detection with blood samples would also be possible.

Reveal was developed as a solution for the analysis of the respective problems introduced with these data sets and was selected as the visualization experts' favorite application in 2011. In the following, the approaches taken with Reveal to solve the questions from the BioVis 2011 and 2012 data analysis challenges will be explained in detail.

5.9.2 Analysis of the BioVis 2011 Challenge Data Set

To identify the SNVs most relevant for the disease hoompalitis, attention was drawn towards epistatic effects described by the data. Thus, in a first step two-locus association testing results were used to construct a network of associations using the Two-Locus Association Network visualization. In total 62,136 different SNV pairs were contained in this network with an association p-value ≤ 0.05. Furthermore, to concentrate only on the most prominent features edges were filtered by edge weight (number of different SNV pairs between two genes) with a threshold of $\tau \geq 50$, leaving 3843 SNVs forming pairs within the graph. The resulting network is shown in figure 5.10 (A). One can clearly see that there are four very prominent edges in the graph, which all correspond to trans effects, because the colors of the edges differ from the colors of the connected nodes. In particular, SNV pairs for the gene combinations CDH22–CDH7, CDH22–CDH10, and CDH22–CDH11 show significant effects on the expression of CDH1. In addition, SNV pairs for the gene combination CDH11–CDH7 have significant effects on the expression of PCDH8.

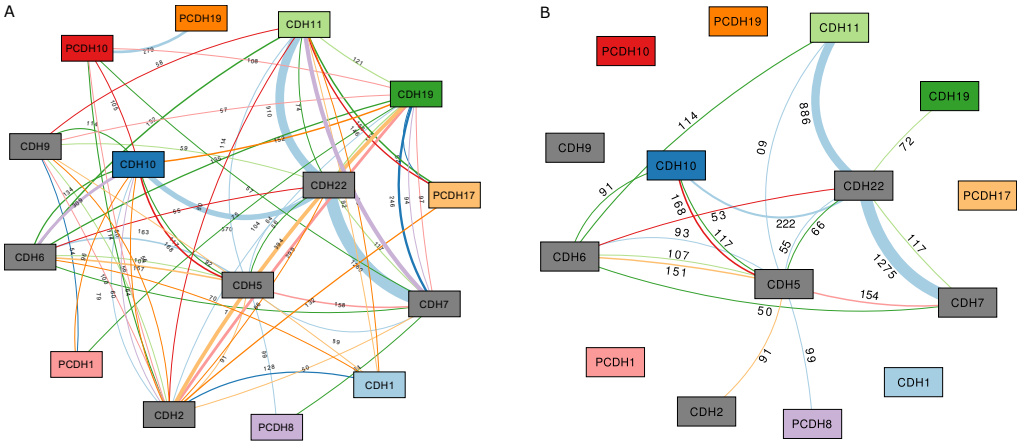

Figure 5.10: Two-Locus Association Networks of the BioVis 2011 data set based on single-locus and two-locus associations. (A) Network based on 3843 different SNVs from the BioVis 2011 contest data set forming significant two-locus associations ($p \leq 0.05$) with gene expression values of 15 genes. Only edges with an edge weight ≥ 50 (representing the number of different SNV pairs between the respective genes) are shown; (B) Network from (A) showing 696 remaining SNVs after included significant single-locus associations ($R^2 > 0.1$, $p \leq 0.05$).

In a second step, data from single-locus associations was added to the Two-Locus Association Network. This allowed for the filtering of only those SNVs that show both, a significant association solely by themselves, as well as an

epistatic effect together with another SNV. Thereby, SNVs with single-locus associations were filtered that showed a regression value of $R^2 \geq 0.1$ and had a statistical significance of $p \leq 0.05$. In total 845 SNVs could be identified. Overlapping these with the SNV pairs from the Two-Locus Association Network in figure 5.10 (A) such that at least one SNV in a SNV pair has to be contained in the list of 845 single-locus associated SNVs, reduced the number of total SNVs in the network from 3843 to 696. The resulting network based on these 696 SNVs is shown in figure 5.10 (B).

Further analysis of the remaining SNVs was performed using the SNV Summary plot, which visualizes genotype cohort distributions. There, effects between two cohorts based on the individuals disease states can be investigated for each SNV. In particular, those SNVs that showed either a difference in the cohort distributions (different color values in the aggregation row) or a difference in the consensus strength of at least 10% points (different bar heights in the aggregation row) between affected and unaffected individuals were filtered. The resulting SNV Summary plot showing the remaining 33 SNVs that are likely to be correlated with the disease is given in figure 5.11. In addition, a χ^2-test followed by an FDR correction for multiple testing was performed for each of the 696 SNVs. The statistical test revealed that 375 of the 696 SNVs had a significant association ($p \leq 0.05$) with the disease state, of which only 13 satisfied the criteria of the 33 SNVs chosen during visual inspection. This shows that pure statistical testing would have missed 20 putative candidate SNVs with clear differences between affected and unaffected individuals.

To study the effect of the 33 identified SNVs on gene expression levels, a gene expression analysis was performed in Mayday. For the 15 genes in the data set t-tests between the affected and unaffected individuals have been computed followed by an FDR correction for multiple testing. Genes with a p-value ≤ 0.05 have been considered significantly differentially expressed. In particular, eight genes satisfied this criterion, namely CDH1, CDH10, CDH11, CDH19, PCDH1, PCDH10, PCDH17, and PCDH19. An aggregated heatmap showing the mean expression levels of the 15 genes for the affected and unaffected individuals is given in figure 5.12. Comparing these results with the association results for the 33 selected SNVs reveals that all 33 SNVs are contained in SNV pairs associated with at least one of the differentially expressed genes. To show the influence of these SNVs on the expression levels of the differentially expressed genes, a SNV derived expression heatmap, based on the transformation described in section 5.5.2 has been produced under the assumption of an additive genotypic model (see figure 5.13). In this visualization, one can clearly see the effect of the 33 identified SNVs. Altogether, this allows for the conclusion that these 33 SNVs are very likely correlated with (or even causal for) hoompalitis.

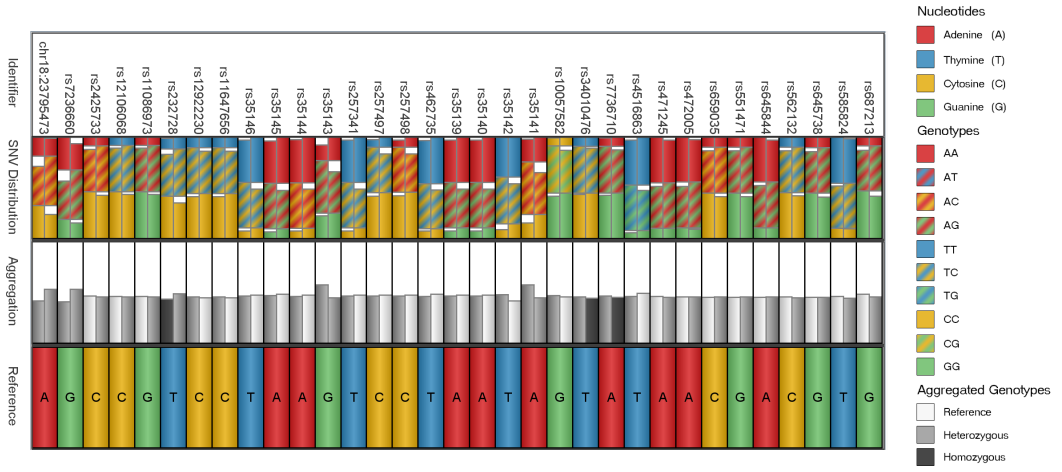

Figure 5.11: SNV Summary plot showing the remaining 33 SNVs after visual selection based on SNV distribution differences (either in the simplified cohort genotype or at least 10% points in consensus strength) between affected and unaffected individuals. The statistic and individual track are hidden due to irrelevant information for the purpose of this figure.

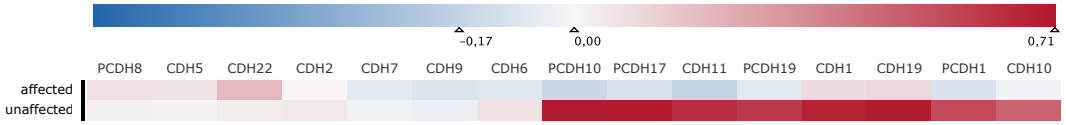

Figure 5.12: Aggregated heatmap showing the mean expression levels of the 15 genes from the BioVis 2011 contest data set for affected and unaffected individuals. Expression levels were mapped to a blue–white–red color gradient centered on zero. Genes were ordered according to an expression-based clustering using the Neighbor Joining algorithm. Distances were calculated with the Euclidean distance measure.

5.9.3 Analysis of the BioVis 2012 Challenge Data Set

For the analysis of the BioVis 2012 data set, a slightly different approach has been chosen, because of SNV quality issues (e.g. missing genotypes) that were introduced in the second round. However, since this data set basically comprises the same analysis steps as the BioVis 2011 data set, only the most important steps and results are described in the following.

Firstly, focus was directed on the tissue samples. Due to the large number of SNVs in the tissue data set and the introduction of noise, such as missing SNV calls or SNV calls located in unannotated reference sequence regions (missing sequence information indicated with the letter N), filtering was necessary before continuing with a visual analytical approach. Thus, SNVs with an N as a reference nucleotide have been removed as well as those with a

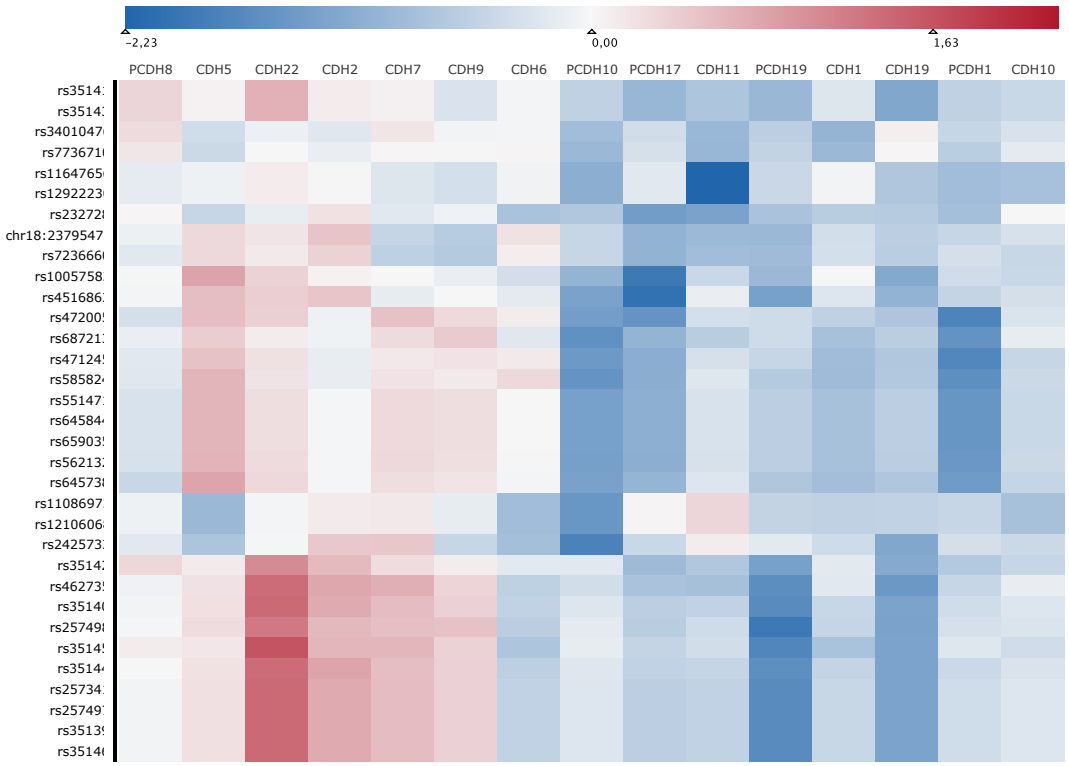

Figure 5.13: SNV derived log2 expression fold-change visualization for the BioVis 2011 contest data set using Mayday's heat map. Only the 33 disease related SNVs are shown. Log2 expression fold-changes were mapped to a blue–white–red color gradient centered on zero. Genes were ordered according to an expression-based clustering using the Neighbor Joining algorithm. Distances were calculated with the Euclidean distance measure.

minor allele frequency of $\leq 5\%$. With this the 230912 initial SNVs have been reduced to 35937. Furthermore, to reduce the number of false positives during further analyses, a statistical test for Hardy-Weinberg equilibrium has been conducted and SNVs have been filtered with a p-value ≤ 0.05 after Bonferroni correction. As a result, 4861 SNVs passed these filters and could be used for further visual inspection. These steps were not necessary with the BioVis 2011 data set, since equivalent issues have not been observed there.

In a second step, differentially expressed genes have been identified with a t-test followed by an FDR correction for multiple testing. The significance threshold was set to $p \leq 0.001$. This procedure revealed eight highly significant differentially expressed genes, namely DRD4, DRD3, SLC6A4, DRD2, SLC6A3, CNTN4, CNTNAP4, and NRG3. To identify those SNVs that affect the expression levels of the differentially expressed genes, a Two-Locus Association Network was constructed based on the 4861 SNVs. Again, edge

weight filtering was used to concentrate only on very prominent features. Thereby, all edges with a weight $\tau < 200$ were removed. In addition, the graph was filtered based on single-locus associations ($R^2 \geq 0.1$, $p \leq 0.05$) as for the BioVis 2011 data set. After these filtering steps, the Two-Locus Association Network was reduced to only display those edges build from SNV pairs, which have a significant association with the expression level of at least one of the differentially expressed genes. The resulting network is shown in figure 5.14.

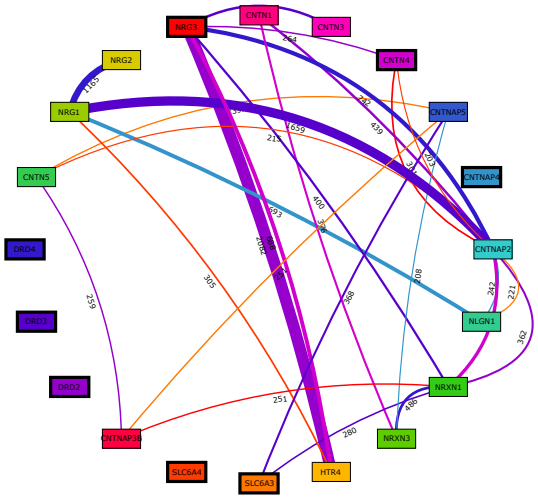

Figure 5.14: Two-Locus Association Network based on 4861 filtered SNVs associated with gene expression levels from the BioVis 2012 data set. Only those edges are shown, where either the SNV pairs have an epistatic effect on the expression level of a differentially expressed gene, or at least one SNV of the SNV pair has a significant single-locus association with a differentially expressed gene. In total eight genes could be identified (highlighted with a thick black border in the network) showing expression differences between affected and unaffected individuals.

As with the BioVis 2011 data set, the SNVs represented by the remaining edges were selected. In total 2575 SNVs remained, which were filtered based on their aggregated cohort distribution. This resulted in 91 SNVs, that are very likely to be correlated with the disease.

To answer the question if blood samples are sufficient to extract the genetic factors leading to disease, the procedure described for the tissue samples was repeated. However, differential expression analysis using a t-test with an FDR correction did not result in any significantly expressed genes. Corrected p-values were in the range of $0.15 - 0.98$. Consequently, no genes associated with the disease could be identified from blood samples. Thus, a SNV association

analysis using gene expression information from blood samples would not be able to identify the genetic factors leading to the disease.

5.10 Conclusion

In this chapter, Reveal has been presented as a toolkit that allows for the analysis of disease related SNVs. For this purpose several different linked visualizations are available and gene expression data can be included through the tight interaction with Mayday. The various features for data filtering, statistical testing as well as the rich visualization capabilities make Reveal a powerful tool for visual exploration of genetic factors leading to disease. With the application of Reveal to the BioVis 2011 and 2012 data sets it was shown how the identification of SNV associated with gene expression changes can be performed. Note that no concrete solutions for the problems in these contests were provided, since the main challenge was the development of new and interactive visual analytics tools. Accordingly, the purpose of the application examples above is to demonstrate how the methods implemented in Reveal would be used to analyze typical eQTL data sets.

Reveal won the visualization experts' favorite award in the BioVis 2011 data analysis challenge, which addressed the visualization of epistatic effects in Reveal as well as the data integration with iHAT, the predecessor of inPHAP. This emphasizes the necessity of Reveal as a visual analytical software solution for the integrative study of SNV and gene expression data.

6. An Innovative and Interactive Visualization Approach for Comparative Multiple Whole Genome Analyses

In the previous chapters, it was shown that with the dramatic increase of the number of sequenced genomes by large-scale studies, such as the 1000 Genomes project [28], an in depth study of variations between different individuals and even whole populations has been made possible. Most research projects thereby focused on investigations concerning small variations such as SNVs or small insertions, duplications, or copy number variations, since non-lethal, larger variations are very unlikely in mammals [168].

Studies similar to the 1000 Genomes project also exist for prokaryotic organisms. In contrast to mammals and many other eukaryotes, research in prokaryotes focuses on the identification of the genetic factors that lead to diversity, involving, for example, pathogenicity or drug resistance. Consequently, researchers are interested in the gene content rather than small variations within a specific gene. With the increasing number of available prokaryotic genome sequences a comparative analysis of whole genomes is possible, including small as well as larger genomic variations. Comparisons are usually made on the basis of whole genome alignments, which can be performed using tools such as MAUVE [33]. However, to be able to efficiently interpret such alignments, comprehensive visualizations are helpful. Basically two approaches have been followed in the past to achieve this. The first approach is to visualize differences between single genomes in comparison to a common reference genome (one to many relation). Genome browsers are typically used in this context. The second approach is to compare all genomes against each other in parallel (many to many relation). There, mainly two strategies have been followed. The first strategy is a linear visualization of the resulting alignment, which was done, for example, in the MAUVE alignment viewer (see figure 6.1). The second one focuses on circular representations, like in the Circos [83] tool shown in figure 6.2. For both visualization types, usually arcs or arrows are used to indicate relationships between different genomic regions. Furthermore, several layers of meta-information can be included, such as experimental data or genome annotations. While Circos is especially useful for the generation of aesthetically attractive figures, it has the disadvantage that only static views can be produced and that no interaction

with the figure is possible. MAUVE, on the other hand, has the disadvantage that with a rising number of aligned genomes and rising complexity of the alignments (including large insertions, deletions and rearrangements) visual clutter is very likely. In general, however, all visualization strategies available so far share one specific drawback. When dealing with large variations, especially insertions, deletions and rearrangements, genomic coordinates of the different genomes have to be mapped to one another. This problem becomes even more obvious when additional information, such as individual genome annotations, have to be added. In this case, information is usually displayed in the coordinate system of the respective genome. This implies manual interaction to adjust coordinates when different genomic areas are visualized. One example is the MAUVE alignment viewer, where the user can interactively align specific sub-regions of the genomes to the coordinates of one chosen reference for comparison. In addition, such coordinate mappings are often implemented such that one of the genomes is used as a reference with the coordinates of the other genomes expressed in relation. However, this approach does not cover cases where specific genomes contain individual sequence information that is not shared by any other genome in the alignment, especially not by the chosen reference. Hence, no coordinate mapping between this respective genome and the reference can be made.

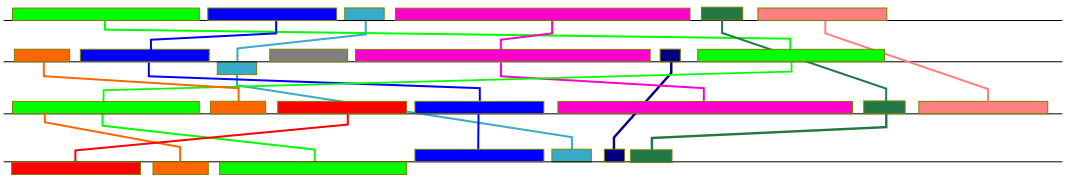

Figure 6.1: Example of Mauve's multiple whole genome alignment visualization strategy. Genomes are shown on different tracks and lines are used to connect similar blocks between the genomes. In addition, identical color values are used for similar blocks. This figure is adapted from [58].

A possible solution would be the calculation of a joint coordinate system that is then shared by all genomes in the alignment. Furthermore, such a joint coordinate system would provide the possibility for consistent annotation of individual genomes with additional information that can easily be compared at any time without further manual intervention. In his PhD thesis, Alexander Herbig introduced the SuperGenome concept, which allows for the creation of such a joint coordinate system on the basis of a multiple whole genome alignment [57]. The SuperGenome is thereby constructed by interpreting genomic rearrangements as a collection of local alignments that are called blocks. The SuperGenome offers bidirectional mappings between each individual genome and the calculated common coordinate system, which in contrast to previously discussed approaches, also allows for the assignment

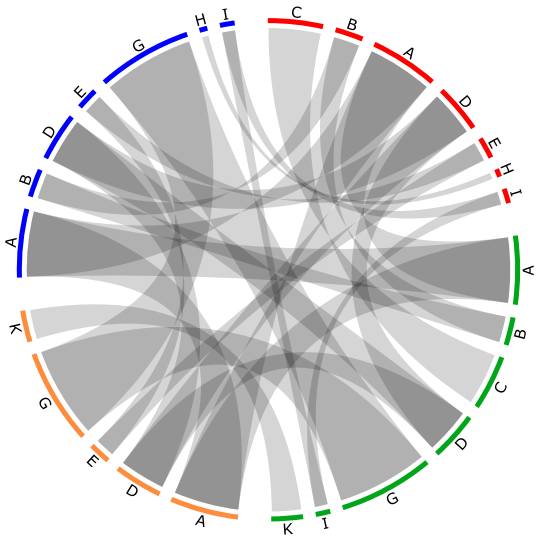

Figure 6.2: Example of Circos' multiple whole genome alignment visualization strategy. Colored arcs are used to represent the genomes on a single circle and ribbons connect similar blocks between different genomes. This figure is adapted from [58].

of coordinates to unaligned regions. Furthermore, starting from an existing multiple whole genome alignment, the SuperGenome allows the detection of large insertions or deletion within blocks, by scanning for gaps that are longer than a user-defined threshold. Such gaps are then used to split blocks into sub-blocks that represent insertions or deletions in one or more aligned genomes.

Based on the SuperGenome concept a multiple whole genome alignment visualization tool, called GenomeRing, has been generated in cooperation with Alexander Herbig and Florian Battke. The SuperGenome algorithm has been included in GenomeRing to allow users to construct sub-blocks from an existing multiple whole genome alignment prior to visualization. This permits the user to decide which types of events are relevant and thus to draw attention towards smaller or larger genomic variations. Through the integration of GenomeRing in the visual analytics software Mayday (see chapter 3), connections to further visualizations implemented in Mayday, or software packages based on Mayday, such as Reveal (see chapter 5), can easily be made. With this, the visualization of gene expression values or single nucleotide variation data within the GenomeRing visualization is possible.

The main contribution of this work to the GenomeRing visualization is the development of a block optimization strategy for an enhanced visual experience, i.e. reducing visual clutter. To achieve this, a block sorting algorithm

has been developed, which is introduced in section 6.2 after a description of GenomeRing's general visualization concept in section 6.1. Furthermore, this algorithm can be used to optimize for three different criteria, for which a comprehensive assessment of the ability to increase visual clarity is given. Text and figures in this chapter are based on collaborative work previously published in [58].

6.1 GenomeRing Design

GenomeRing has been designed as a visualization containing two circles with several different segments. Each segment on the outer ring, together with the corresponding segment on the inner ring, represents one sub-block in the constructed SuperGenome. Furthermore, the outer ring represents genomic sequences in forward direction, while the inner ring shows the same genomic locus in reverse direction. Thus, a specific block of the genome alignment is either represented by the outer ring, if the respective sequence is in forward direction, or by the inner ring, if the block is inverted in this particular genome. The different segments can be labeled, either numerically or by a user-defined identifier, to highlight segments of interest. Furthermore, each segment in the visualization is separated into different lanes, where each lane represents one genome in the multiple whole genome alignment. For each lane a color is chosen using ColorBrewer color maps, if the number of aligned genomes is less than 12. For larger numbers a rainbow gradient is used to assign a unique color to each genome. Furthermore, lanes representing the same genome, are always located at the same positions within different blocks, which in addition to the color values, helps to keep track of genomes. In this, each genome is still uniquely defined, even if the distinction between different colors becomes difficult, for example, if the number of different genomes is very large. The lanes within one segment are connected by paths of the same color, indicating the natural order of represented blocks in the respective genome. Such paths, drawn outside of circle segments, however, do not encode for sequence information, but are solely used to illustrate which blocks of the SuperGenome are contained in an aligned genome and in which order they appear. If, for example, a block in the SuperGenome is inverted in one of the aligned genomes, then a path leads through the respective segment on the inner ring, rather than the outer ring. With this strategy, complete genome sequences can be reconstructed by following paths of the same color and concatenating the segments in the order dictated by the path. In addition to all these view elements, start and end of a genome alignment have to be indicated. In GenomeRing, two small flags are drawn inside of the inner ring for each genome, one for the start position and one for the end position of the respective genome sequence. The flags are colored by the respective color value representing the corresponding genome.

Furthermore, start and end flag are distinguished by the direction of the flag. If a flag is drawn away from the flagpole, then this flag represents a genome start position. In contrast to that, a flag pointing towards the flagpole encodes for the end position. The visualization is complemented by a scale that shows the number of bases displayed per radial range of the circle. An additional legend at the bottom of the visualization mapping color values to genome identifiers, completes the list of visual components. An example visualization of an artificial whole genome alignment containing three different genomes is shown in figure 6.3 to illustrate the use of the different view elements.

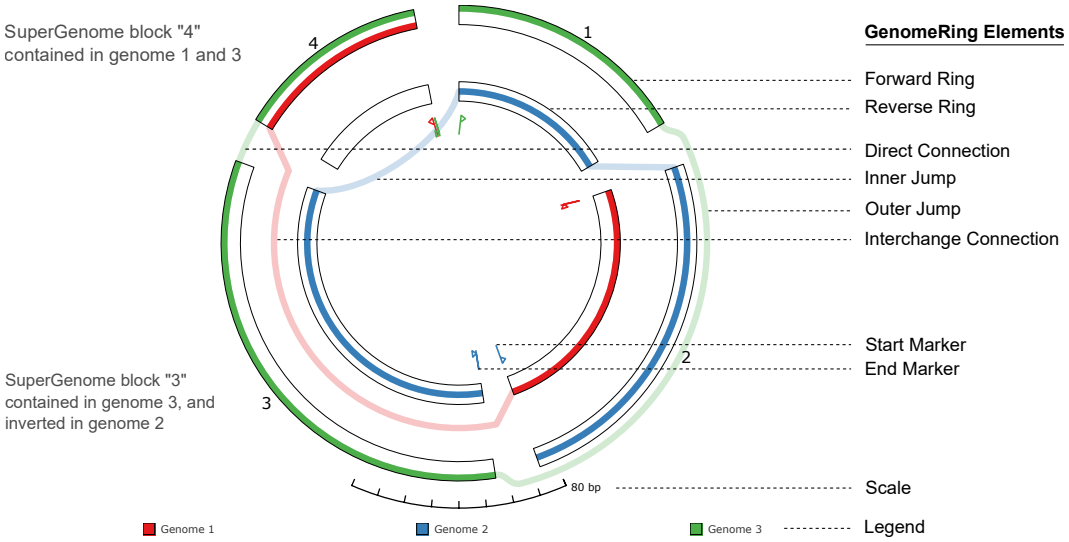

Figure 6.3: Example of a GenomeRing visualization showing an artificial multiple whole genome alignment containing three different genomes. The view elements of GenomeRing are highlighted using dotted lines and labeled accordingly. This figure is based on an illustration previously published in [58].

Clearly, the order of the segments in the GenomeRing visualization, although initially based on the order of blocks in the SuperGenome, does not encode any information. Thus, changing the order would only lead to the introduction of different paths connecting the segments, but the order of blocks in the genomes would be preserved. This can be used for block order optimization, i.e. to reduce visual clutter (see section 6.2).

6.1.1 Visual Representation of Circle Segment Connections

By definition of the SuperGenome concept, it is clear that a single genome does not necessarily have to contain all of the blocks contained in the SuperGenome. For the GenomeRing visualization this means that there can be segments that are not connected by a genome path. Due to this

fact, several different types of segment connections had to be implemented. All of these connections are shown in figure 6.3. The simplest connection is between two segments, either on the outer ring, or on the inner ring, in consecutive order, where no third block comes in between. Here, a direct connection between the segments can be made, using a path with the same radius as the lanes that have to be connected. The second type of connections are indirect connections, which can further be separated into two different classes. The first class of indirect connections contains paths that have to be drawn between segments that are both located either on the outer ring, or on the inner ring, with at least one additional segment in between. In this case, a jump connection has to be made, skipping the additional segments. A jump connection for the outer ring is called outer jump and a jump connection on the inner ring is called inner jump, respectively. Outer jumps are represented by paths, drawn outside of the outer ring with a defined radius that is determined by the maximal number of other paths skipping the same segments. Inner jumps are defined in a similar way, with the only difference that they are drawn inside of the inner ring.

The second class of indirect connections represents connections where one segment lies on the outer and the other one on the inner ring. Paths connecting these segments are called interchange connections. Such interchange connections are visualized using paths in-between the inner and the outer ring. Thus, the distance between the inner and the outer ring is defined by the maximal number of interchange connections that have to be drawn.

6.1.2 Directions of Segment Paths

The segments in GenomeRing have been designed in a way that sequence information represented by lanes inside of these segments can always be read from left to right. However, the connecting paths do not necessarily need to follow the same direction. Paths are just used to connect segments, whereas their direction has no biological meaning. This concept offers additional freedom for the placement of indirect connections, in order to minimize visual clutter. To be precise, a layout algorithm for the connecting paths can choose path directions such that the maximal angle of a path is at most 180°. Consequently, this minimizes the total number of blocks that have to be skipped by any indirect connection.

6.2 Block Order Optimization

The choice to use paths to indicate the order of segments in GenomeRing requires further possibilities for avoiding visual clutter. The main reason for vi-

sual clutter are indirect connections, where blocks have to be skipped. Changing the order of segments in the GenomeRing visualization does not affect the biological meaning, yet it can largely increase visual clarity and consequently interpretability. An example of how changing the block order can increase visual clarity is shown in figure 6.4. There, the reduced number of block skipping arcs (see figure 6.4 (B)) clearly simplifies the assessment of the overall topology of the underlying multiple whole genome alignment.

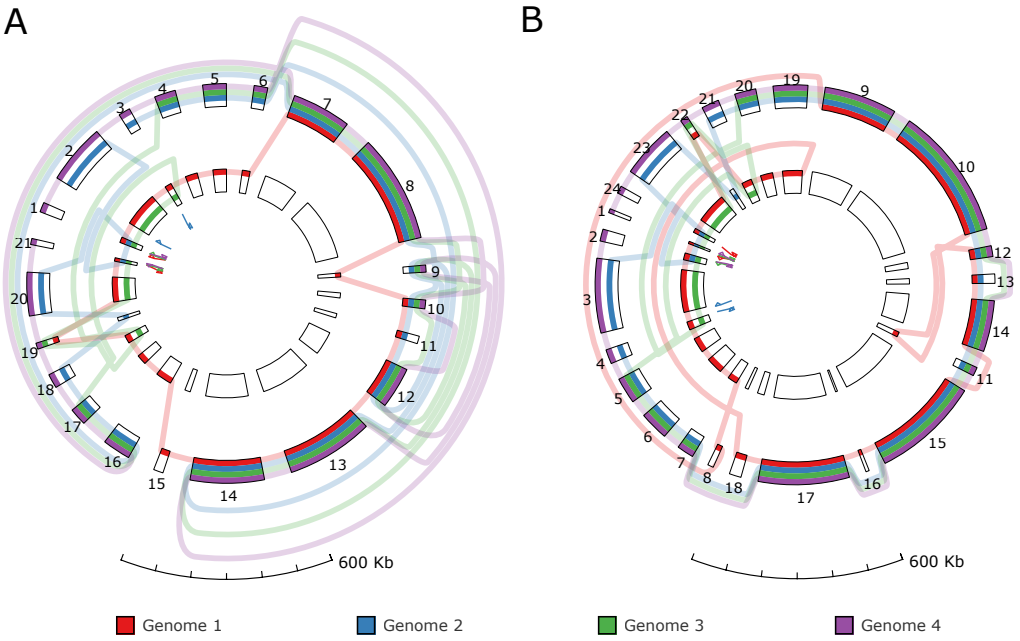

Figure 6.4: GenomeRing example of a multiple whole genome alignment of four different genomes showing how changing the block order can increase visual clarity. **A** shows the initial ordering of blocks as calculated by the SuperGenome algorithm; **B** shows an optimized ordering with reduced visual clutter allowing to better assess the overall topology.

An optimal constellation would only contain direct connections, because these do not introduce visual clutter. Since this is usually impossible, an arrangement of segments has to be found that minimizes, for example, the number of indirect connections. This minimization problem can be reduced to the well known Traveling Salesman Problem (TSP), which has been proven to be NP-complete [118]. Thus, in general an exact solution for minimizing visual clutter in GenomeRing cannot be found efficiently. The default ordering of segments is defined by the ordering of blocks in the SuperGenome. To improve this default ordering, several optimization strategies have been implemented. Firstly, segments can be arranged based on the natural ordering of the blocks of one of the aligned genomes, which is used as a reference for the others. With this strategy one can easily concentrate on a specific genome of interest and

investigate the structures of the other genomes in relation to that. However, if the objective is visual clarity, more sophisticated approaches have to be taken. A heuristic can be used to approximate an optimal solution based on three different criteria defined in the following:

1. **Minimization of the number of indirect connections**

   An indirect connection links two blocks, if these have a consecutive order in at least one of the genomes, but not in the SuperGenome. This suggests that a possible optimization strategy is to rearrange blocks in the SuperGenome, such that the total number of indirect connections is minimized for all aligned genomes. Hence, a sub-optimal arrangement for a single genome is permitted, if it leads to a more optimal conformation regarding all genomes at once.

2. **Minimization of the number of skipped blocks**

   An indirect connection leads to at least one skipped block. However, visual clutter increases proportionally with the number of blocks that are skipped by such an indirect connection. Therefore, the focus of this second approach lies in the minimization of the number of blocks that have to be skipped for all aligned genomes, regardless of the number of indirect connections that have to be made. This may increase the total number of indirect connections, but can lead to a much more appealing visualization, since the length of indirect connections becomes fairly small.

3. **Minimization of the total jump length**

   The third approach, in contrast to the second, also takes the length of a block into account. If blocks have to be skipped, visual clutter can increase with the length of the skipped bock, since large blocks lead to longer arcs. Keeping arc lengths short is therefore preferable. Here, the total jump length is defined as the sum of the absolute angles for all indirect connections in the visualization. Thus, a minimization of the total jump length leads to shorter arcs and can consequently result in a clearer visual representation.

Algorithm 1: Block order optimization

```
   Data: Cost function f, Initial block ordering O_I, Set of genomes G
   Result: Ordering O_M minimizing f
 1 O_M ← O_I // initialize final block ordering
 2 repeat
 3 |   foreach g ∈ G do
   |   |   // order blocks based on the natural ordering in g
   |   |   // the order of blocks that are not in g is preserved
 4 |   |   O' ← orderBlocksUsingTemplate(g,O_M);
 5 |   |   if f(O') < f(O_M) then
 6 |   |   |   O_M ← O';
 7 |   |   end
   |   |   // swap pairs of blocks
 8 |   |   foreach b_i,b_j ∈ O_M do
 9 |   |   |   O' ← swapBlocks(b_i,b_j,O_M);
10 |   |   |   if f(O') < f(O_M) then
11 |   |   |   |   O_M ← O';
12 |   |   |   end
13 |   |   end
14 |   end
15 until f(O_M) converges;
16 return O_M
```

Since there is no polynomial time algorithm that minimizes these criteria, a heuristic approach has been taken in GENOMERING, using an iterative procedure that evaluates a cost function f. Thereby, f measures the costs given the current conformation of segments. The costs themselves can vary depending on the user-defined minimization strategy. However, the heuristic is the same irrespective of the cost function. Given a specific cost function and an initial segment ordering, the heuristic procedure iteratively tries to decrease the current costs. A concrete pseudocode description of the whole procedure is given in Algorithm 1. With this strategy an overall worst case runtime of $O(n^2 \cdot b^2)$ was achieved, where n corresponds to the number of aligned genomes and b to the total number of blocks in the SuperGenome. In contrast, a naive approach enumerating all possible segment conformations and choosing the optimal one afterwards, would have resulted in a runtime of $O(b!)$.

6.3 Integration into MAYDAY

The integration of GENOMERING into the MAYDAY visual analytics platform offers several different possibilities for extending the visualization with addi-

tional information, such as gene expression data. This allows, for example, to visualize genes of interest in the context of a whole genome alignment. In MAYDAY, such genes can be identified by applying a statistical test or based on further meta-information. In order to map genes to blocks in GENOMERING, chromosomal locations have to be available. In MAYDAY, such information is stored in so-called meta-information objects (`MIOs`). Specific `Visualizer` objects, that are able to read and process `MIOs` provide functions that allow visualization without prior knowledge about the information content. Hence, to visualize genes in GENOMERING, shared instances of `Visualizer` objects can be created using a common `ViewModel`. All visualizations sharing the same `Visualizer` are then linked to each other. This means that, for example, selections made on encapsulated objects in the `Visualizer` in one of the connected plots, result in a parallel selection of the same objects in all the other linked visualizations. In addition to that, GENOMERING makes use of MAYDAY's integrated export functionalities. This allows the user to generate high quality images in various bitmap formats, such as `PNG`, `TIFF`, or `JPG` in arbitrary resolution, as well as in vector graphics formats (`SVG`, `PDF`) without any loss in quality.

6.3.1 Gene Visualization

In GENOMERING, each path representing a genome in the alignment can be linked to a visualizer object from a MAYDAY visualization showing a specific set of `ProbeLists`. All genes contained in these `ProbeLists` are then drawn on top of the respective path. To be able to distinguish between the path itself and the genes drawn on top, the color of the path is brightened up, while genes are drawn with the color value defined by the respective `ProbeList`. This leads to a dashed look of the genome path. With this strategy, genomic co-locations of genes in different genomes can easily be identified. Furthermore, genes mapping to genomic islands can easily be spotted, allowing for a quick identification of pathogenic markers. This is shown in figure 6.5 for a multiple whole genome alignment of four different *Campylobacter jejuni* strains, namely *C. jejuni* RM1221, *C. jejuni* NCTC 11168, *C. jejuni* 81-176, and *C. jejuni* 81116. For these bacteria, so-called CJIEs (*C. jejuni* integrated elements) have been identified previously [44, 119]. In the GENOMERING visualization one can easily locate gene containing genomic islands that may be linked to pathogenicity. For example, block 4 in figure 6.5 corresponds to CJIE4, which contains phage-related proteins.

6.3.2 Single Nucleotide Variation Visualization

Since all `Visualizer` objects are managed through MAYDAY, it is possible to include single nucleotide information in GENOMERING provided by REVEAL.

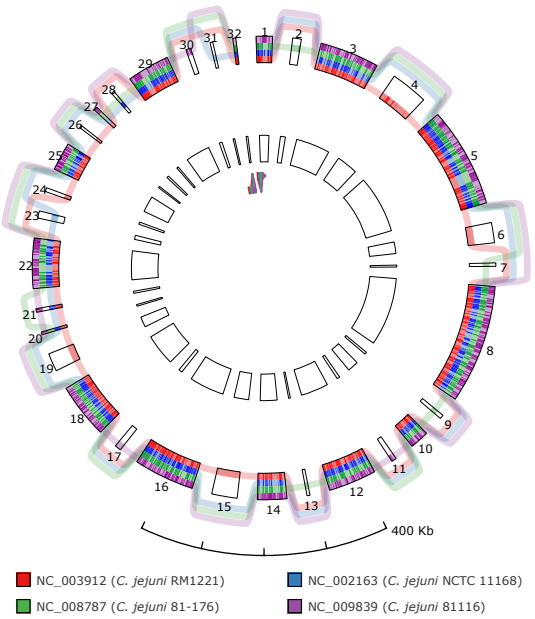

Figure 6.5: GENOMERING visualization of a multiple whole genome alignment of four different *Campylobacter jejuni* strains. Gene information has been mapped to and drawn on top of the circle segments for each genome to allow for the comparison of the overall gene content between the bacteria.

To do so, a SNV based visualization of a specific `SNVList` has to be created in REVEAL. The respective `Visualizer` will then be available in MAYDAY. As for genes, the `Visualizer` can be linked to genomes in GENOMERING to enable the visualization of SNVs. The visualization itself is performed in the same way as for genes, however, the color for a SNV can be modified to distinguish it from a gene. By default black color is used for SNVs. To enable SNV visualization also on a zoomed out overview, an additional scale factor has been introduced. This allows the user to scale the region occupied by a single SNV in GENOMERING to a size that can easily be spotted. The scaling factor can be adjusted for various zoom levels, providing optimal visualization of SNVs for all kinds of GENOMERING representations.

6.3.3 Linkage to MAYDAY's Genome Browser

MAYDAY has an integrated genome browser [155] (see figure 6.6) that allows for the visualization of genes, expression data, genome annotations, meta-information, such as statistical p-values, as well as mapped RNA-seq reads using different tracks. The genome browser can thus be used to investigate detailed information about specific genomic regions. Using a shared instance of a `Visualizer` object between GENOMERING and the genome browser provides the opportunity to concentrate on specific genomic regions from a multiple

whole genome alignment. To provide this functionality, mouse interactions between GenomeRing and Mayday's genome browser are linked, if they share the same `Visualizer` object. To be precise, a double click on a region within the GenomeRing visualization results in centering the view of Mayday's genome browser to the exactly same location. This allows for a linearized visualization of regions of interest and additional annotation of such regions using various supporting meta-information, either imported from external sources or generated using functions included in Mayday.

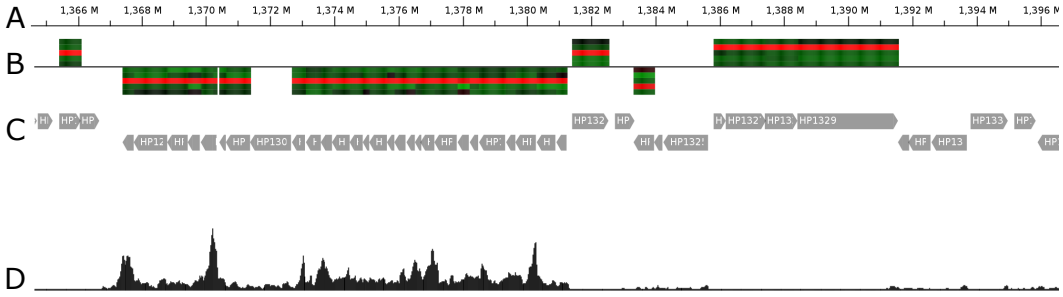

Figure 6.6: Example of Mayday's track-based genome browser. Four different tracks are shown, which are as follows: (A) genomic coordinates of the visualized genome; (B) expression value heatmap of differentially regulated genes; (C) protein coding gene annotation; (D) wiggle track of corresponding RNA-seq data for the reverse strand. This figure has been adapted from [58].

6.4 Interaction Possibilities

Interaction with a visualization offers the possibility to explore various aspects of the visualized data in more detail. In GenomeRing several different interaction features have been integrated, including free rotation, zooming and panning by combinations of keyboard keys and the mouse wheel. Furthermore, modifications to the visualization itself can be made, in order to enhance the visual experience. For example, the visualization of connecting paths between segments can be switched off. This is especially useful, if keeping track of the order of segments is of minor importance in comparison to the differences between individual blocks in the SuperGenome. Furthermore, the spacing between segments can be adjusted, which also helps in the identification of similarities and differences between segments. If the paths are of interest, visual clutter can still be an issue, even when a block order optimization has been performed. To address this issue, two different strategies have been implemented. Firstly, the user can adjust the spacing between indirect connections, as well as modify the width of each path with the combination of keyboard keys and the mouse wheel. This helps in keeping

track of each respective path, even when color values are very similar.

The second strategy focuses on the visualization of the ordering of the segments. Providing additional freedom in the placement of paths, as discussed above, can in some cases lead to confusion regarding the directions of the paths, especially when there are many blocks in the SuperGenome. This has been addressed in GenomeRing by path animations, that can interactively be switched on. The animation of paths is realized using a dash pattern that moves along the direction of the respective path, traveling from the genomes start position to its end position. This concept allows the user to better understand the complex structure of the multiple whole genome alignment. Furthermore, mouse interactions include the display of tooltip information for specific genomes. This can be achieved by pointing the mouse cursor over a position of interest in one of the segments for more than 2 seconds. The displayed tooltip then shows the exact coordinates of the chosen position in the respective genome as well as in the SuperGenome. In addition, general information about the chosen block is given, such as the size of the block, its index in the SuperGenome, or the relative offset of the mouse position from the blocks start position in base pairs. If additional meta-information is visualized at the mouse location, as for example genes, or single nucleotide variations, information on the respective element, such as the gene or SNV identifiers or the exact location in base pairs, are also shown in the tooltip. The visualization of such meta-information can be interactively turned off for specific genomes to improve visual clarity if needed.

As mentioned above, visual clarity can additionally be improved by changing the order of the segments. Besides the introduced optimization strategies, the arrangement of individual segments can be interactively modified by the user. Each segment is assigned a label during the SuperGenome construction. By default, numbers starting with 1 are used to assign unique identifiers to each block. However, in the GenomeRing visualization, the labels of the blocks can be changed to a user-defined value, or hidden completely. In addition, genome colors as well as genome identifiers can be modified. This circumvents additional cumbersome post-export modifications and guarantees publication ready high quality images. To assist the user with all the provided interaction features, an overlaying help page showing all the available keyboard and mouse commands can be displayed using the `F1` key on the keyboard within the GenomeRing visualization.

6.5 Application Examples of the Block Order Optimization Strategies

In this work a block order optimization heuristic for the GenomeRing visualization has been developed that makes use of different optimization criteria. Although the general algorithm stays the same for each optimization criterion, the outcome varies significantly and each strategy emphasizes different data aspects and helps to clarify specific questions to the data. In the following, example applications for each of the four strategies are shown that demonstrate their capability to increase visual clarity for the individual GenomeRing visualizations.

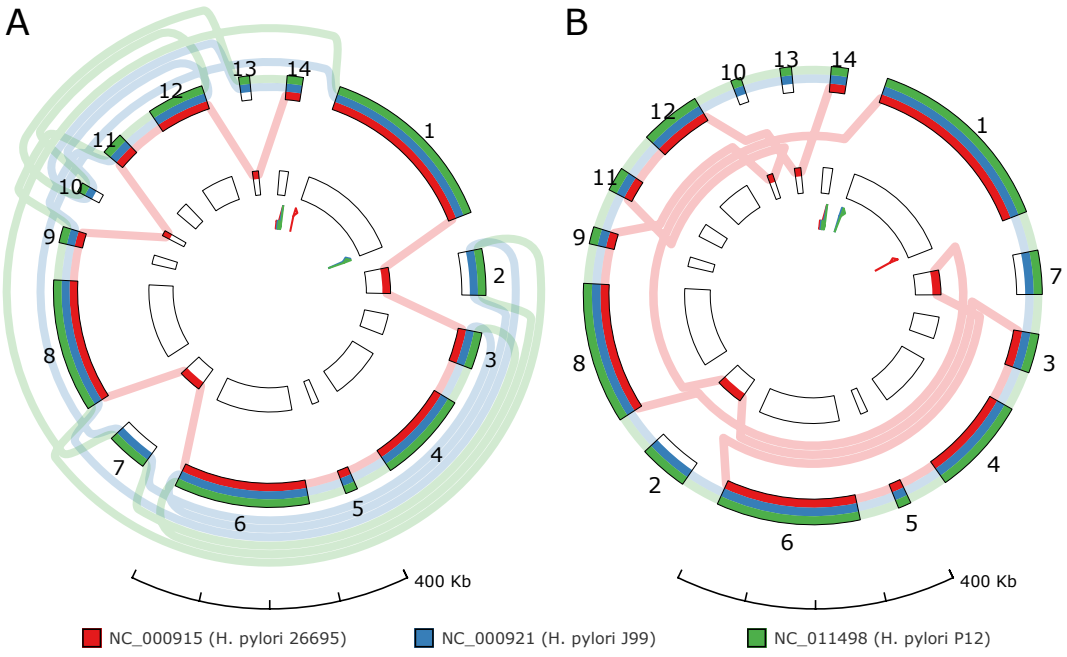

Figure 6.7: GenomeRing visualization based on a multiple whole genome alignment of three different *Helicobacter pylori* strains. **A** shows the default ordering of the blocks as provided by the SuperGenome; In **B** the blocks have been rearranged with respect ot the natural ordering of blocks in the genome of the *H. pylori* J99 strain.

The simplest sorting strategy is to use one of the genomes as a template for the block ordering. With this strategy visual clarity is guaranteed for the selected genome and comparisons to the other genomes can easily be made. An example, showing how changing the block order based on a template affects the overall topology is shown in figure 6.7 for a multiple whole genome alignment of three different *Helicobacter pylori* strains (*H. pylori* 26695, *H.*

pylori J99, and *H. pylori* P12), with a minimal block length of 20kb. Despite the risk of increasing visual clutter, this strategy is helpful in making decisions on how genomes differ in comparison to a specific genome of interest. This enables, for example, to distinguish between genomes with only small and those that show large differences, either in the number of structural changes or in the sizes of these events. In figure 6.7 *H. pylori* J99 has been chosen as a reference. One can easily see that *H. pylori* P12 shows no differences to *H. pylori* J99 with respect to the chosen minimal block size, but *H. pylori* 26695 has some larger structural modifications and inversions.

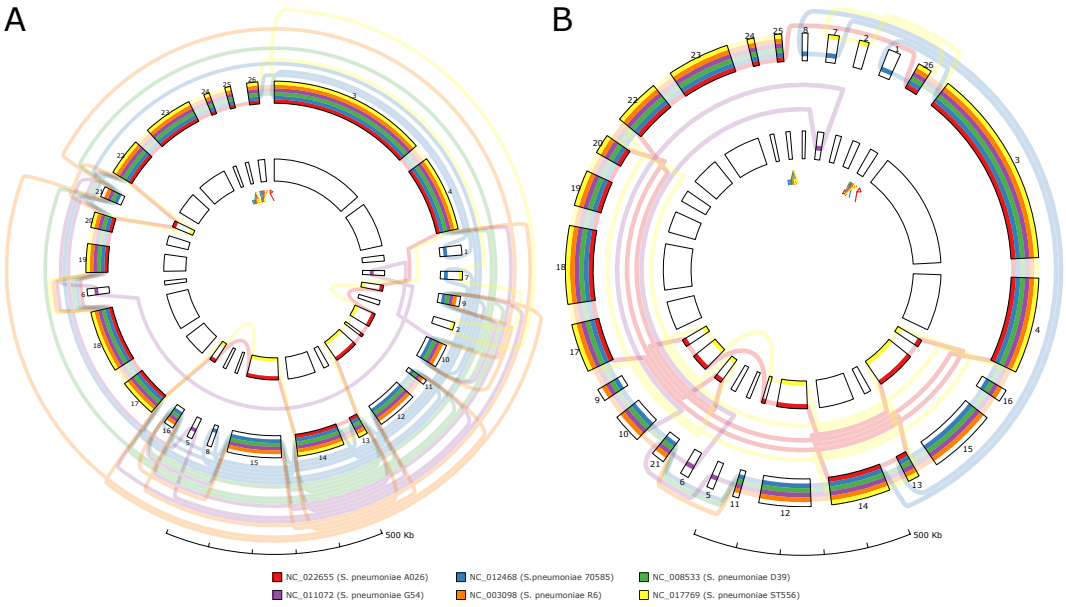

Figure 6.8: GenomeRing visualization based on a multiple whole genome alignment of six *Streptococcus pneumoniae* strains. **A** shows the default block ordering as defined by the SuperGenome; **B** shows an optimized version, where the number of jumps and interchange connections has been minimized to improve visual clarity.

The second ordering strategy minimizes the number of jumps and interchange connections. This results in a maximization of the number of direct connections between blocks in the SuperGenome and is thus well suited to visualize continuous parts shared by the underlying genomes. In figure 6.8 a multiple whole genome alignment of six different *Streptococcus pneumoniae* strains (*S. pneumoniae* A026, *S. pneumoniae* G54, *S. pneumoniae* 70585, *S. pneumoniae* R6, *S. pneumoniae* D39, and *S. pneumoniae* ST556) with a minimal block size of 15kb is shown. As can bee seen, after optimization continuous blocks between the majority of strains in the alignment stand out clearly. Thus, this optimization strategy assists with the identification of consecutive, constant

parts within the genomes in comparison to variable regions.

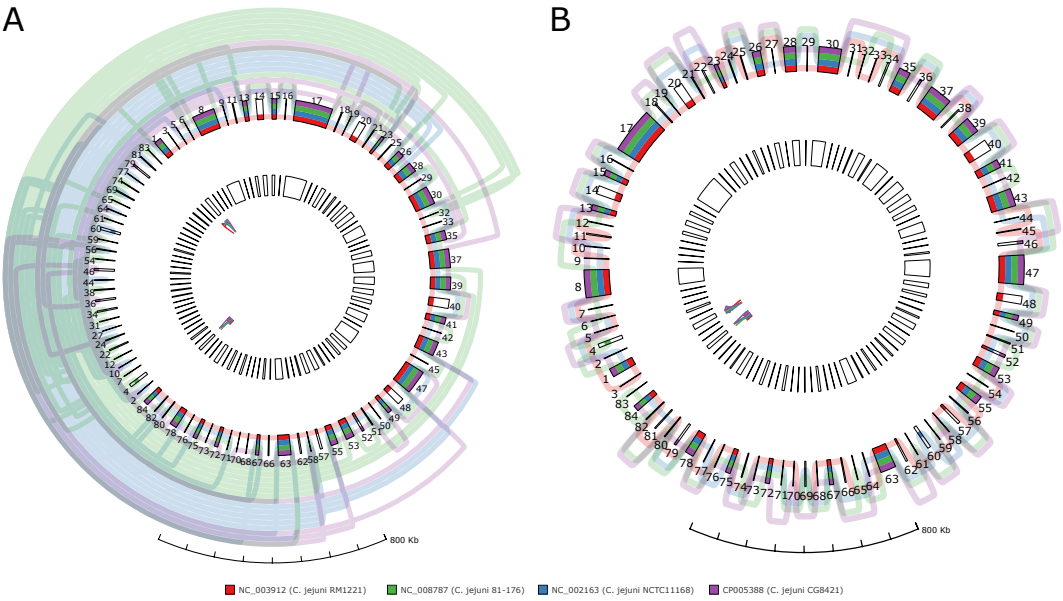

Figure 6.9: GenomeRing visualization based on a multiple whole genome alignment of four *Campylobacter jejuni* strains. **A** shows the block ordering based on the RM1221 strain; **B** shows an optimized version where the number of skipped blocks has been minimized to improve visual clarity for the comparison of *C. jejuni* RM1221 to the other strains. Genomic islands can easily be identified after block order optimization.

Figure 6.9 shows a multiple whole genome alignment of four *Campylobacter jejuni* strains (*C. jejuni* RM1221, *C. jejuni* 81-176, *C. jejuni* NCTC11168, and *C. jejuni* CG8421) with a minimal block size o 2kb. For *C. jejuni* it is known that pathogenicity or drug resistance is mostly introduced by horizontal gene transfer, which can be seen as so-called genomic islands within a multiple whole genome alignment. Thus, it is expected that the genome alignment does not contain larger structural changes, but mutliple different insertion elements that are contained in only one of the genomes. In figure 6.9 (A) blocks have been arranged according to the RM1221 strain, with the intention to improve comparability with respect to that strain. However, the template ordering strategy does not work well in this case leading to visual clutter, which makes it difficult to follow the respective arcs. Since in this case, mostly insertions between the visualized genomes are expected the skipped block ordering strategy suits best to improve visual clarity, as can be seen in figure 6.9 (B). After optimization individual genomic regions stand out clearly and genome paths can easily be followed.

The last strategy focuses on the improvement of the visualization with respect

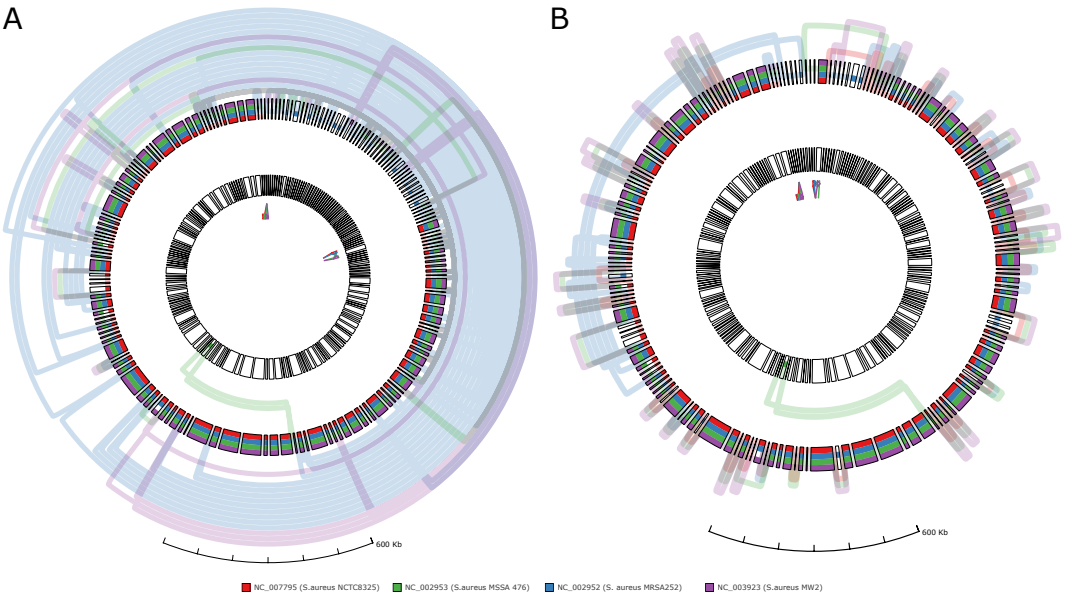

Figure 6.10: GenomeRing visualization based on a multiple whole genome alignment of four *Staphylococcus aureus* strains. **A** shows the default block ordering as specified by the SuperGenome; **B** shows an optimized version, where the total arc length criterion has been used to improve visual clarity. Block labels have been removed to reduce visual clutter.

to the overall complexity. This is achieved by minimizing the length of the arcs that have to be drawn. Small arcs can easily be followed by the user, while for larger ones it is more difficult to keep track of, especially if these are drawn in close proximity to each other. Thus, ordering the blocks of the SuperGenome in a way that small arcs are preferred over large ones, leads to more comprehensive topologies. Consequently, this strategy is well suited to gain appropriate overviews of the visualized data. Figure 6.10 shows two GenomeRing visualizations for a multiple whole genome alignment of four *Staphylococcus aureus* strains (*S. aureus* NCTC8325, *S. aureus* MSSA476, *S. aureus* MRSA252, and *S. aureus* MW2) with a minimal block size of 2kb. While the default ordering of blocks does not reveal the general structure of the multiple whole genome alignment very well, the optimized version clearly shows the similarities and dissimilarities between these strains with only minimal visual clutter. By applying additional interaction features after block reordering, such as zooming or highlighting path directions, a comprehensive picture of the structures in the multiple whole genome alignment can be obtained.

To summarize, each of the presented block ordering strategies provides a more comprehensive visualization of the respective multiple whole genome alignments. However, the choice of which ordering strategy to use largely depends

on the questions one wants to answer and on the individual structure of the underlying multiple whole genome alignment. Consequently, a single optimization strategy would not suffice to provide appropriate insights into the manifold patterns that are often hidden in the data.

6.6 Conclusion

The GenomeRing visualization, which is based on the SuperGenome concept, offers new ways of assessing structural similarities and differences between genomes. However, visual clarity of this visualization largely depends on the arrangement of the blocks within the visualization itself. Due to GenomeRing's flexible design, the order of the blocks does not encode any information, but can be used to increase visual clarity. In this work, an algorithm comprising three different possible optimization criteria has been developed that helps in finding optimal arrangements for user specific needs. As was shown in section 6.5 the right choice for an optimization strategy largely depends on the questions one wants to answer with the visualization. Thus, each of the three optimization criteria can be useful in an appropriate analysis scenario. Clearly, the block sorting heuristic does not remove visual clutter completely. Depending on the underlying data, i.e. the complexity of the multiple whole genome alignment, complete removal of visual clutter is impossible. Thus, the intension of the heuristic described in this work is to improve the visual experience with GenomeRing, in order to clarify structural differences as good as possible. In cases, where visual clutter still remains an issue after block order optimization, the user is provided with functionalities to interactively change specific characteristics of GenomeRing. Especially, the possibility to animate the genome paths in order to highlight the direction by which blocks are traversed, is a very powerful tool to better understand genome structure. Thus, the block order optimization strategy together with the interaction possibilities introduced in this dissertation provide a comprehensive solution for an improved visual experience, when analyzing multiple whole genome alignments with GenomeRing.

7. A Pipeline for the Reconstruction and Comparative Analysis of Ancient and Modern Bacterial Genomes

The development of the NGS technology has created new opportunities for genetic research. Due to the decreasing sequencing costs, the study of whole genomes has become very attractive. Also meta-genomic approaches have been followed, where a whole collection of different species is sequenced at once. This is especially useful for the study of bacteria, since a clear separation of bacterial DNA from environmental samples is often difficult or even impossible. However, NGS is not restricted to samples from modern DNA. It is applicable to all sources of DNA including conserved DNA samples that are hundreds or thousands of years old. Such ancient DNA (aDNA) samples can provide insight into the evolutionary history of organisms. Furthermore, comparing the information gained from such samples with information from modern DNA can help in the clarification of still unexplained historical events, such as undocumented epidemics or plagues. Moreover, the evolution of bacterial pathogenicity is of great interest, since diseases such as leprosy have been devastating many years ago leading to the death of hundreds of thousands of people. Investigations of ancient graves now provide the possibility to study the concrete factors of mass mortality, which mostly go back to bacterial infections. Due to the lack of knowledge and the fast outspread of pathogenic microorganisms, such events are often poorly documented. Hence, using NGS can help in the description of untraced historical events and identify the reasons of mass mortality by assessing the remaining genomic information content. The goal is to identify single nucleotide variations (SNVs), that enable researches to answer questions on the genetic background of specific organisms.

In this context, Bos *et al.* showed that sequencing coupled with a sophisticated bioinformatic analysis allows for the reconstruction of the genomes of ancient human pathogens [18]. They also described that when dealing with aDNA, various problems have to be faced. First of all, DNA does not stay in its original condition over time. In fact, it gets degraded, which leads to misincorporations as a consequence of nucleotide deamination. In addition to that, the aDNA content of an environmental sample, i.e. a sample taken from water, soil, a corpse, or any other biological material, is usually small and likely to be contaminated with DNA from modern organisms. Thus,

appropriate procedures are needed to enrich the DNA fragments of interest and subsequently to authenticate their ancient origin. Another issue is the fragmentation of DNA over time, leading to small DNA molecules with mean lengths between 60 and 150 base pairs [117], but this number can vary greatly from sample to sample. Furthermore, sequenced reads from such fragments usually show base calls with low quality Phred scores (low probability that the corresponding base call is correct), thus sequencing errors are very likely. A possible approach to address the quality issue has been applied in previous studies of aDNA [18]. The idea is to generate redundant information during sequencing in order to increase the overall quality of the resulting reads. This is achieved by using paired-end sequencing. Due to the short DNA fragment sizes of aDNA, the forward and reverse reads often overlap, i.e. showing a negative insert size. This redundant information can be used to improve the overall quality of the sequencing by applying merging procedures that generate single reads out of read pairs that cover the whole sequenced fragment (see figure 7.1 for an example). Naturally, only those bases with higher quality in the overlapping region are used for consensus read generation. A further strategy to improve the quality of reads for a subsequent mapping analysis is quality trimming of reads that could not be merged, since sequencing quality usually decreases towards the 3' end of the read. Last but not least, care has to be taken during the mapping process itself, in order to find a good trade-off between mapping specificity and sensitivity. Hence, parameters have to be chosen, such that a balance between false-positively mapping reads and false negative reads is achieved.

Although all these practices are highly useful to improve the overall quality and thus enable researches to study not just modern, but also aDNA, there is currently no satisfying solution available for high throughput analysis of multiple different samples. This renders it difficult to study samples in parallel. Especially connecting information from sequenced samples with already existing genome assemblies is challenging. In such cases, a system is required that connects the different steps needed for read pre-processing, mapping and subsequent analysis. To address this need, Kircher *et al.* introduced a pipeline that covers read pre-processing and mapping, which is suited for aDNA samples [80]. However, this pipeline is very time and disk intensive and therefore not suitable for projects covering more than a handful of genomes. Especially the read merging step in the Kircher pipeline is extremely slow. One reason for that is the lack of parallelization. Furthermore, post-analysis methods are not covered. These comprise the reconstruction of whole genomes by e.g. mapping assembly, as well as the subsequent phylogenetic analysis of the reconstructed genomes together with already existing genomes. Here, the term mapping assembly refers to the reconstruction of a genome by incorporating nucleotide variations into the genome of a reference organism.

Figure 7.1: Example of the read merging procedure. First sequencing adapters are clipped from the 3' end of the forward and reverse read. Afterwards the reverse complement of the clipped reverse read is calculated and merged with the clipped forward read, such that the base with the higher quality phred score is chosen at each position for the final merged read. *F-Read* represents the forward read and *F-Qual* the corresponding quality string. *R-Read* and *R-Qual* are defined equivalently for the reverse read and *M-Read* and *M-Qual* equivalently for the merged read.

With this approach, variation effect prediction, based on reference gene annotations, is possible. Thus, valuable information on a genes functionality can be gained, or conclusions about a genes expression state can be inferred.

In order to provide an automated and efficient processing of the described analysis steps, a pipeline has been implemented together with Alexander Herbig [57] that covers all of the mentioned methods. Furthermore, it addresses limitations of the Kircher pipeline, such as the time and disk consumption problem. In the following, the different steps of the pipeline are explained and special attention is drawn towards the merging step, for which a new software solution, called ClipAndMerge, has been developed in this thesis. For the merging procedure itself comparisons have been made to already existing approaches. A prove of concept application of this pipeline has been described in our corresponding paper for the comparative analysis of medieval and modern *Mycobacterium leprae* strains [144]. There, detailed information is given on how the pipeline can be used to address the specific needs of ancient and modern DNA processing and analysis. In this dissertation, two additional scenarios are described, demonstrating the manifold applicability of the pipeline to modern as well as ancient DNA. The first application, which will be explained in section 7.2, focuses on the study of the pathogenicity of different *Treponema pallidum* strains, which cause syphilis, one of the most devastating

diseases in human history. Syphilis has been and is still a major threat to human health with millions of new cases worldwide [116]. Studying the genetic differences of syphilis associated bacterial strains on a population based level may provide major insights into the evolution of this bacterium. Hence, the methods described in this chapter have been used on DNA from samples collected from different countries all over the world to elucidate the population genetic patterns of *Treponema pallidum*. In addition to that, a microarray based DNA capture technique is described in chapter 8. There, it is shown how this pipeline in combination with the methods developed for DNA capturing helps to process DNA sequencing data from ancient as well as modern pathogenic organisms.

7.1 Individual Pipeline Steps

The main focus of the following pipeline lies in the automated, efficient and parallel processing of ancient as well as modern DNA samples. It does not just provides insights into the evolutionary events that occurred over time, but also gives rise to a comparative analysis including already existing genome sequences from organisms with large phylogenetic distances. During the execution of the pipeline, several different analysis steps are carried out, including quality assessment of the raw reads, adapter clipping and subsequent read merging and mapping. Based on the mapping results, draft genomes can be created, which allow for the application of advanced analyses not included in the pipeline directly. An example would be a phylogenetic analysis based on a multiple whole genome alignment. Such an alignment is necessary to allow the user to include already existing genomes, besides the created drafts, for a following comparative analysis of the differences in each of the sample genomes. Furthermore, the effects of variations on coding and non-coding genes can be assessed through the pipeline. The connection of the individual steps has been realized using `bash` scripts, which only require the raw input `FASTQ` files in order to carry out a whole analysis. Figure 7.2 shows an overview of all the contained components of the pipeline and illustrates how the different components are connected to each other.

7.1.1 Read Preprocessing and Mapping

Quality Assessment Assuring a good read quality is crucial for the success of the read mapping procedure. Only if the read quality is high, it can be guaranteed that reads get mapped correctly if at all. There are several existing tools for quality control of raw reads in the `FASTQ` file format. An example is fastx_quality_stats contained in the fastx toolkit [84], which is a console-based tool without graphical output for interpretation. Another

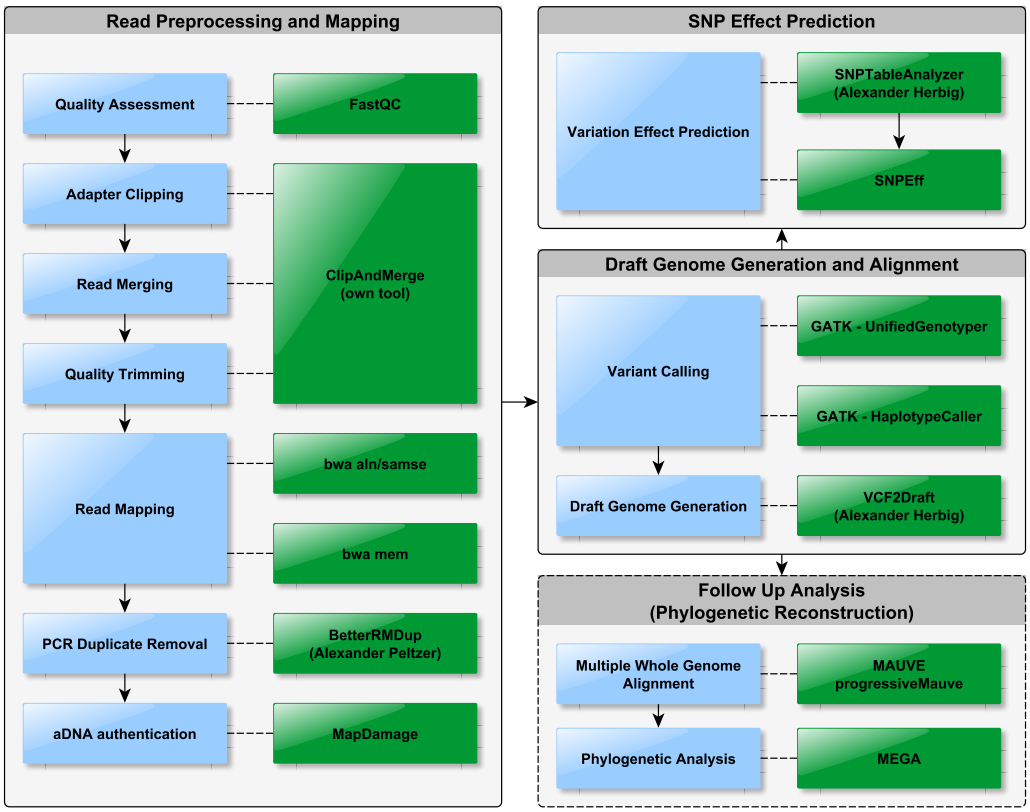

Figure 7.2: Overview of the different steps of the ancient and modern bacterial genomes processing pipeline. Blue boxes indicate the individual step in the pipeline and corresponding green boxes highlight the tool that is used at the respective step. Arrows show the direction of the pipeline. The dashed box shows the application of a phylogenetic analysis as a follow up step to the pipeline itself.

possibility is the NGS QC Toolkit [120] written in PERL, which performs quality control analyses for the Illumina and 454 platform.

For this pipeline, the FastQC tool [4] is used, which provides several different quality measures, such as assessment of the per base sequence quality, the k-mer content, as well as an adapter contamination estimation. These measures are combined with informative graphs and charts to improve the interpretability of the quality measuring results. In general, a per base quality Phred score of ≥ 20 is wanted to decrease the probability of mismatches during read mapping. Furthermore, the k-mer content should be homogeneously distributed and no adapters should be left after adapter clipping. The quality of the reads is assessed at two different stages during the processing of the pipeline, namely directly on the raw reads and once again before mapping, after all

pre-processing steps have been performed. This second quality assessment allows for an additional control of the success of the read merging and quality trimming procedure described in the following.

Adapter Clipping, Read Merging and Quality Trimming Read merging is a necessary step to improve the overall quality of the reads in order to prevent mismatches in the subsequent read alignment, which may lead to false positive variant calls in later analyses. Thus, this step is very important, since it modifies the reads and therefore has large influence on the subsequent mapping of the reads to a specified reference genome. In consequence, it also influences all analysis steps relying on the mapped reads. To perform the merging step, the ClipAndMerge tool has been implemented in the context of this thesis that is capable of clipping adapter sequences, merging clipped paired-end reads if possible and trimming non-merged reads based on a user-defined quality threshold. The first step includes the removal of sequencing adapters. To do so, a clipping strategy was developed that is motivated by the fastx toolkit [84]. Clipping is performed for forward and reverse reads in parallel, making use of multi-core systems. In order to identify adapter sequences at the 3' ends of the reads, a semi-local alignment, based on the Smith-Waterman algorithm [147], is calculated between the adapter sequence and the read. If an alignment satisfies the user-defined threshold for the minimal alignment length, all bases between the start position of the alignment and the 3' end of the read are removed. However, in some cases it may occur that the start position of the alignment and the start position of the adapter sequence are different. In such cases, the alignment start position is shifted towards the 5' end of the read by the number of unaligned bases at the 5' end of the adapter sequence. Although this strategy is very conservative and may trim off bases that do not belong to the adapter, it also ensures that no adapter bases are left in the read sequence, which is important for the subsequent merging step. Only then it is possible to prevent reads from being merged due to overlapping adapter sequences.

As mentioned above, the adapter clipping procedure is performed in parallel for the forward and reverse reads, using separate threads and combining the results of the clipping afterwards in a third additional thread. For the merging part, first the reverse complement of the reverse read has to be calculated. Then a maximal overlap between the 3' ends of the forward read and the reverse complement of the reverse read are calculated by starting with the largest possible overlap and a pairwise comparison of the nucleotides in the overlap region. The overlap gets accepted, if the edit distance in the overlap region is below a user-defined threshold and if the size of the overlap region is larger than a user-defined minimal overlap size. Default parameters require a minimal overlap of 10 bases with at most 5% mismatches in the overlap

region. Bases with very low sequencing quality are treated as undefined nucleotides and do not contribute to the edit distance in the temporary overlap region. By default, all bases with a Phred score ≤ 5 are treated as undefined. The Phred score is a measure for the sequencing quality of a base and varies between different sequencing technologies. For state of the art Illumina platforms this score typically ranges from 1 to 41. If the criteria for an overlap cannot be fulfilled, the temporary overlap is reduced by one base and calculations are repeated. This is done by shifting the reverse read along the forward read, until a valid overlap is found. If no overlap exists given the quality criteria, both reads are further processed as single-end reads. In case an overlap can be found, the two read sequences are merged into a single read sequence, where the bases in the overlap region are chosen such that for each base the one with the higher sequencing quality, either in the forward read or in the reverse read, is taken. If the merged read is longer than the user defined minimal read length, it is reported as a single-end read in the final output `FASTQ` file. The default value for the minimal read length is 25 bases. If no overlap fulfilling the quality criteria can be found, both, the forward as well as the reverse read are reported in the output file.

Special care has to be taken of reads, where the pairing read, either the forward or the reverse read, has been removed during adapter clipping, because it become shorter than the minimal overlap required for merging. In such cases, only one read remains and no merging can be performed. Such reads are treated as single-end reads for further processing. After the merging step, quality trimming of the reads can be performed before the reads get written to the output `FASTQ` file. This is, however, only necessary for the non-merged reads, since read quality is usually good at the 5' ends and drops when moving to the 3' ends of the reads. Bases are therefore trimmed from the 3' end of single-end reads until a user-defined threshold is satisfied, which by default is a Phred score of at least 20. If quality trimming is selected, all non-merged reads undergo this procedure before they are written to the output `FASTQ` file. Here, again, a filter for minimal read length can be defined to ensure that all output reads fulfill the quality criteria for mapping.

To demonstrate the power of the ClipAndMerge tool developed in this dissertation, a comparison between ClipAndMerge and five other state of the art adapter clipping and merging tools has been made. These tools have been selected based on their capability to merge paired-end reads with a user-defined minimal quality. One of the tools, that was used in various ancient DNA analysis projects [81, 127, 146] is MergeReadsFastQ by Martin Kircher [80] that is implemented in the Python programming language. It offers adapter clipping for paired-end reads, as well as the identification of overlaps between corresponding read pairs. FLASH [96] follows a similar approach as

MergeReadsFastQ, but does not perform adapter clipping. For the adapter clipping step, tools such as CutAdapt [100] can be used. FLASH focuses on high performance merging of overlapping paired-end reads with respect to the overall runtime. It has therefore been parallelized to take advantage of multi-core systems. The third tool in this comparison is SeqPrep [150], which was designed for Illumina reads and is therefore limited to these. It also offers both adapter clipping as well as merging of overlapping paired-end reads. In contrast to most other adapter clipping algorithms, SeqPrep offers a lot of different parameters to control the actual local alignment step for adapter identification and can, in addition, deal with potentially high mismatch rates during adapter clipping and read merging [150]. leeHom [134] uses a Bayesian maximum *a posteriori* probability approach to tackle the problem of adapter clipping and merging of overlapping paired-end reads. In contrast to the other tools, leeHom, does not separate the process of adapter clipping and merging, but considers both steps within one probabilistic model [134]. Finally, AdapterRemoval [92] can process both single- and paired-end reads and performs adapter removal with subsequent read merging. In addition to the other tools, AdapterRemoval can also trim low quality bases from the 3' end of a read if necessary. Furthermore, adapter clipping is possible from the 5' and the 3' end, which makes AdapterRemoval a very flexible tool for different experimental settings and sequencing platforms [92].

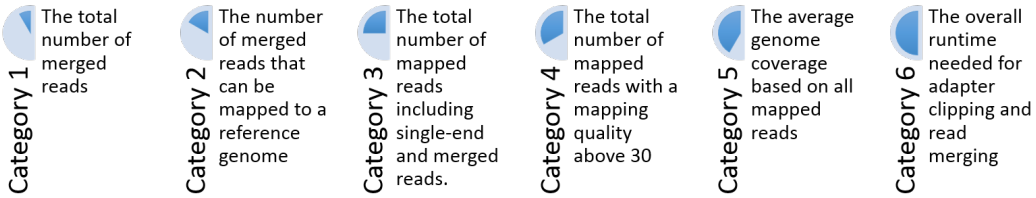

Figure 7.3: Overview of the six different categories that were used for the comparison of adapter clipping and overlapping paired-end read merging tools.

For a fair comparison between all mentioned tools, the respective default parameters for clipping and merging have been adjusted such that the following three criteria are fulfilled. Firstly, the overlap region has to contain at least 10 bases in order to merge two reads. In addition to that, the number of mismatches in the overlap region is not allowed to be higher than 5% with respect to the overlap size. And the third criterion requires a minimal overall read length of 25 bases for merged reads as well as non-merged reads. For tools, where one of these requirements could not be

defined, filtering of the reads was performed after merging, if possible. Comparisons have been made for six different categories, as illustrated in figure 7.3.

Of these categories, the number of merged reads is a direct measure of the merging efficiency. However, a high merging rate does not suffice to evaluate if the merging was correct, i.e. the two merged reads really originated from the same DNA fragment. For this purpose, the mapping efficiency, in terms of the mapping rate for merged reads, has been calculated. For merged reads that can be mapped to the reference genome it is very likely that the merging was correct, while unmapped reads do either not originate from the reference organism or merging let to an artificial DNA fragment. Furthermore, also the total number of mapped reads is calculated in order to provide a quality measure for the adapter clipping and quality trimming step. If the adapters are clipped with high accuracy, followed by quality trimming of low quality bases from the 3' end, the probability to map the resulting read to the reference genome is increased. In contrast, remaining adapter sequences or bad quality bases lead to mismatches in the alignment and consequently to a reduced mapping rate. Additionally, the mapping quality is a measure for the number of mismatches, insertions, deletions, and soft clipped bases introduced in the alignment. This number is minimal if technical biases due to the read preprocessing can be excluded. Thus, counting the reads with a high mapping quality and comparing this number between the different tested tools allows for the evaluation of the capability of producing high quality reads. Furthermore, for downstream analyses the genome coverage is a very important measure, since many applications, as for instance variant calling, can only be performed if the genome coverage is large enough. Therefore, this measure has been taken into account for the merging tool comparison. Last but not least, one of the most important measures is the overall runtime. Although, high quality reads are preferable in comparison to lower quality reads in general, the amount of time that has to be spent in order to achieve a high quality plays an important role. One might, for example, be willing to accept a slightly lower quality, if in consequence the overall runtime for the read preprocessing is largely reduced. This makes clear, that a good merging tool has to find a trade-off between these categories, rather than concentrating only at a single one.

Of all tested tools, leeHom is the only one that does not provide the possibility to distinguish between merged and non-merged reads after processing. Furthermore, there is no possibility to define criteria for the actual overlap region or to filter out reads shorter than 20 base pairs. Therefore, not all of the measures described above could be applied.

For the mapping step, the BWA `aln/samse` algorithm has been used with

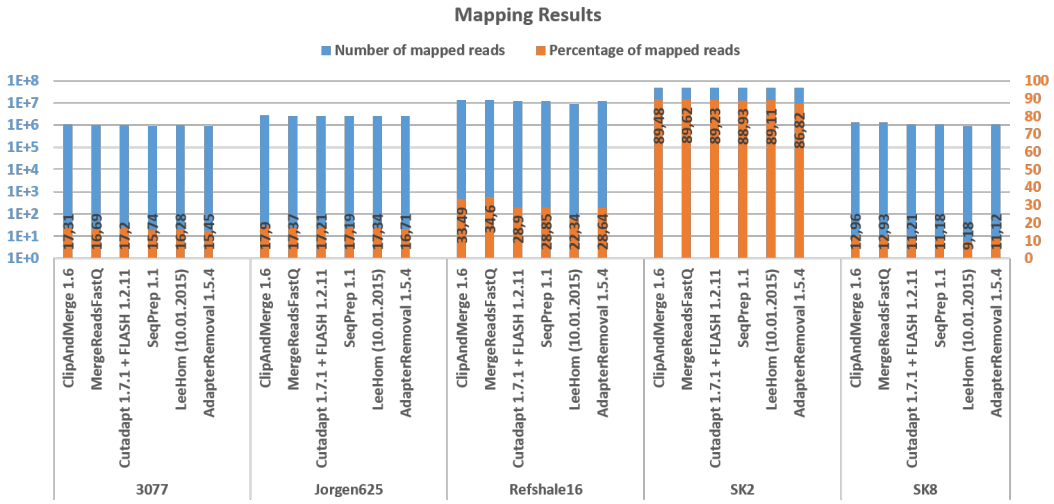

Figure 7.4: Mapping evaluation of the ClipAndMerge tool in comparison to five other tools capable of clipping sequencing adapters and merging overlapping paired-end reads. Blue bars show the total number of mapped reads and orange bars the percentage of mapped reads with respect to the total number of input reads.

default parameters. The tools have been tested on five different data sets from *Mycobacterium leprae* samples, for which it could be shown that they contain DNA of ancient origin [144]. The runtime has been measured on a machine with 64 cores and 512 gigabytes of memory. Detailed information about the results of each measurement, for each of the tested tools, is summarized in table A.4 in the Appendix. The results show that ClipAndMerge outperforms the other tested tools in most of the categories for the five data sets. Figure 7.4 shows the total number of mapped reads for each of the tested tools and data sets used for the comparison. One can see that ClipAndMerge outperforms the other tools or shows at least comparable numbers. The only tool that had a slightly higher number of merged reads in three of the five datasets is MergeReadsFastQ. However, in MergeReadsFastQ remaining single-end reads that cannot be merged, due to the lack of a respective pair after adapter clipping, are also classified as merged. Due to this fact, such reads could not be distinguished from truly merged reads in the output file. Therefore, the number of merged reads for the MergeReadsFastQ tool is usually overestimated. Looking at the number of mappable merged reads (see table A.4 in the Appendix), one can see that again ClipAndMerge shows comparable results to the other tools. Here, especially SeqPrep and MergeReadsFastQ have to be mentioned, where SeqPrep stands out as the overall winner in this category. However, differences to ClipAndMerge are only marginal. For the third category in this comparison, all remaining reads, single-end as well as merged reads, are used for the subsequent mapping.

The results show no clear differences between the compared tools. All tools perform equaly well, with a marginal advance for ClipAndMerge and MergedReadsFastQ. Regarding the number of mapped reads with a quality above 30, ClipAndMerge shows the highest number in three out of five data sets. For the other data sets MergeReadsFastQ is slightly better. These results can also be seen in figure 7.5. However, at this point it has to be mentioned that MergeReadsFastQ performs a quality adjustment for merged reads, which does not only include the region of the actual overlap. This procedure is guided by assigning identity values to each of the bases and quality is then adjusted based on these estimated identities, which usually leads to higher qualities at the 5' ends and consequently to a larger number of reads with a mapping quality above 30. Although ClipAndMerge does not adjust Phred scores outside the overlap region, results show a comparable number of high quality mapping reads with respect to MergeReadsFastQ.

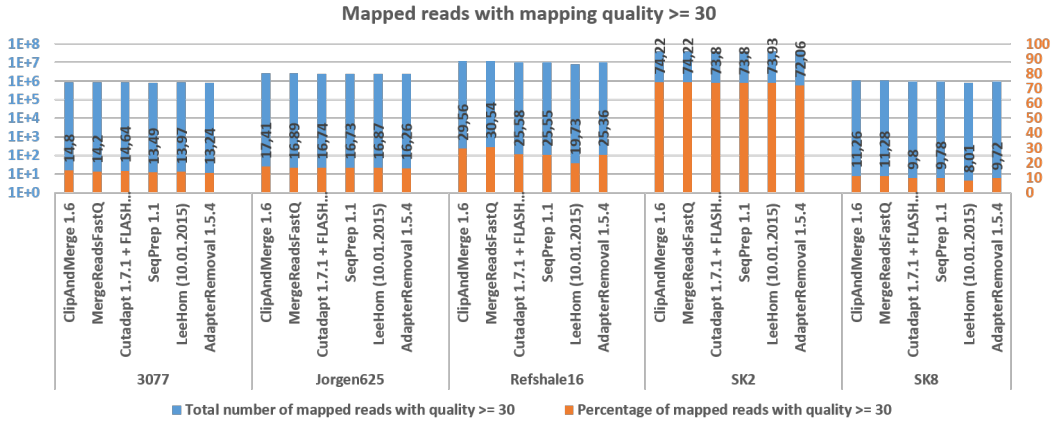

Figure 7.5: Mapping quality evaluation of the ClipAndMerge tool in comparison to five other tools capable of clipping sequencing adapters and merging overlapping paired-end reads. Blue bars show the total number of mapped reads with a mapping quality above 30 and orange bars show the percentage of mapped high quality reads with respect to the overall number of input reads.

Regarding the genome coverage, results show that it is a bit lower for the ClipAndMerge tool in comparison to the other tools, but the differences are only small. Furthermore, a higher number of high quality mapping reads is usually preferable to a slightly higher genome coverage. Last but not least, the runtime for adapter clipping and merging has been calculated. For CutAdapt and FLASH the runtime is composed of the total runtime needed to clip the forward as well as the reverse adapter, plus the runtime needed for the merging. When one of the tools did not allow to filter reads smaller than 20 base pairs in length, a simple `awk` script was used and the runtime of that script was added to the overall runtime, in order to get comparable

results. Figure 7.6 shows the results of the runtime analysis. One can see that ClipAndMerge performs second best for all data sets. Only the combination of CutAdapt and FLASH is a little bit faster. However, regarding the much lower number of merged reads, this seems not surprising, because fewer computational steps had to be taken by FLASH during merging. Furthermore, FLASH makes use of all available 64 cores of the evaluation system, while ClipAndMerge uses only 4 cores. Nevertheless, ClipAndMerge is nearly on par with CutAdapt and FLASH and a future version that allows to use more than 4 cores is expected to outperform all existing tools. In this context, MergeReadsFastQ has to be mentioned, because it was the only one that achieved comparable results in the previous categories. Although showing a good performance with respect to merging, mapping, as well as genome coverage, the runtime is extremely high, which makes this tool unsuitable for larger data sets with deeper sequenced samples.

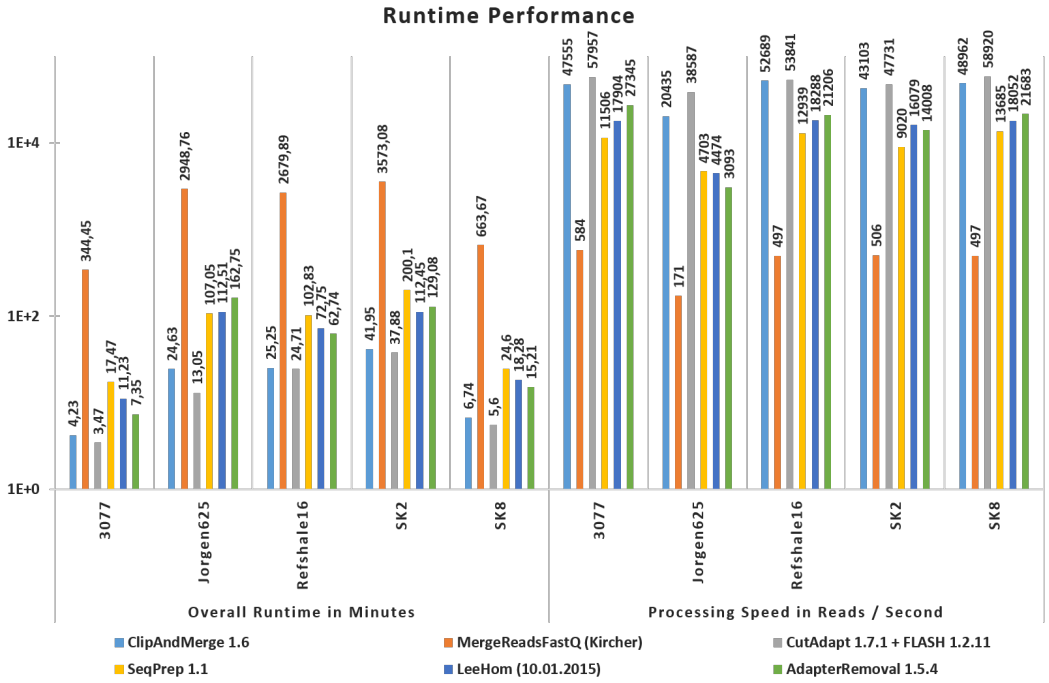

Figure 7.6: Runtime evaluation of the ClipAndMerge tool in comparison to five other tools capable of clipping sequencing adapters and merging overlapping paired-end reads. Runtime is shown as two measures, the overall runtime needed for the adapter clipping plus merging measured in minutes and the processing speed measured in reads per second.

To conclude, the results of the measurements that have been performed show that ClipAndMerge has a very good overall performance that is, in most cases, better than for already existing tools with respect to sensitivity during the

merging step, quality of the resulting reads and most notably overall runtime with respect to the number of merged reads.

Read Mapping and Duplicate Removal Alignment of preprocessed reads to a user-defined reference genome is conducted using the Burrows-Wheeler Aligner (BWA) by applying either the BWA `aln/samse` algorithm [90] or the BWA `mem` algorithm. The BWA `aln/samse` algorithm has proven to provide a good trade-off between sensitivity and specificity in earlier studies [80] when applied to DNA samples from ancient or modern origin. While the BWA `aln/samse` is preferable for short reads, the developers suggest to use the BWA `mem` algorithm for reads with lengths of 70bp or longer. In this pipeline, for both algorithms, the default parameters are used for the mapping procedure. Details about these parameters can be found in table A.5 in the Appendix. The resulting `SAM` file containing all reads and information about their mapping position in the reference genome is then filtered for mapped reads only and converted into the equivalent binary format (`BAM`) for subsequent PCR duplicate removal.

The general strategy for duplicate removal differs between single-end and paired-end reads. For single-end reads only the start position can be used for potential PCR duplicate identification, due to e.g. quality trimming from the 3' end. For paired-end reads both the start position of the forward as well as the start position of the reverse read can be used. This means that the prediction of PCR duplicates is more accurate for paired-end than for single-end reads. If we would apply this strategy to the merged reads, they would be treated as single-end reads and only the start position of a merged read would be considered for duplicate identification. However, merged reads correspond to a single DNA fragment, which was shorter than twice the length of a single-end read. In addition, merged reads do not suffer from shortening due to quality trimming, because the 5' end of a read is usually of high quality. Taking these two characteristics into account, PCR duplicates for merged reads are those that agree on both start and end position of the respective read. The modified tool for removing PCR duplicates on data containing merged reads, called BetterRmDup, operates in two different ways, depending on the underlying data. For merged reads, the start and end position are considered during duplicate identification. For single-end reads only the start position is considered. The concept of this strategy was developed together with Alexander Peltzer [123], who implemented the tool during his master thesis. Since remaining paired-end reads in the raw data set are also treated as single-end reads in the mapping stage, no duplicate identification method for paired-end reads is needed, because every mapped read is now treated as a single-end read.

Authentication of Ancient DNA Samples If samples containing ancient DNA are analyzed using this pipeline, the DNA that mapped to the reference genome has to be verified to be of ancient origin. Samples containing aDNA usually also contain modern DNA. During mapping no discrimination between ancient and modern DNA is made. Ancient DNA suffers from degradation, which are postmortem mutations that increase over time. Substitutions resulting from deamination of cytosine molecules are thereby overrepresented in aDNA samples. Consequently, conversions from $C \rightarrow T$ and $G \rightarrow A$, respectively, make up the majority of errors. Especially the 5' and 3' ends of aDNA are affected by these mis-incorporations, hence software solutions try to validate aDNA fragments and to distinguish between modern and ancient DNA based on the nucleotide substitution patterns of aDNA fragments at their 5' and 3' ends. For detailed information on nucleotide misincorporations see [140]. In this pipeline the tool MapDamage is used. MapDamage can calculate misincorporation patterns from mapped reads given the reference sequence. Thereby, the 5' and 3' ends of the reads are tested for all possible nucleotide conversions and graphs are produced showing the frequency of the first and last 25 bases of the mapped reads in the data set. To guarantee that the misincorporation results are not influenced by the pre-processing of the reads, such as adapter clipping or quality trimming, only mapped merged reads are used here, since merged reads basically consist of two 5' ends, one from the forward read and one from the reverse read, that have not been touched during the quality control and subsequent improvement methods. If the mapped reads are of ancient origin, an increased frequency of $C \rightarrow T$ conversions should be observed at the 5' end of the merged reads as well as an increased $G \rightarrow A$ conversion frequency at the 3' end [140]. Figure 7.7 shows a typical example of a MapDamage misincorporation plot from reads of ancient origin.

7.1.2 Draft Genome Generation and Multiple Whole Genome Alignment

Based on the mapping results for each sample in the pipeline, a so-called mapping assembly is conducted and the genomes of the sequenced samples are reconstructed. As a starting point of the mapping assembly the reference genome is taken and single nucleotide variations (SNVs) are introduced based on the respective mapping result. In order to identify SNVs in comparison to the reference genome, the `Unified Genotyper` module of the Genome Analysis Toolkit (GATK) is used, which produces `VCF` [32] (Variant Call Format) files that typically contain information about all SNVs that were identified. However, to be able to use the resulting `VCF` file for constructing a draft genome, the option `EMIT_ALL_SITES` has to be set. This results in a `VCF` file containing all bases that are also found in the reference genome. Applying this procedure

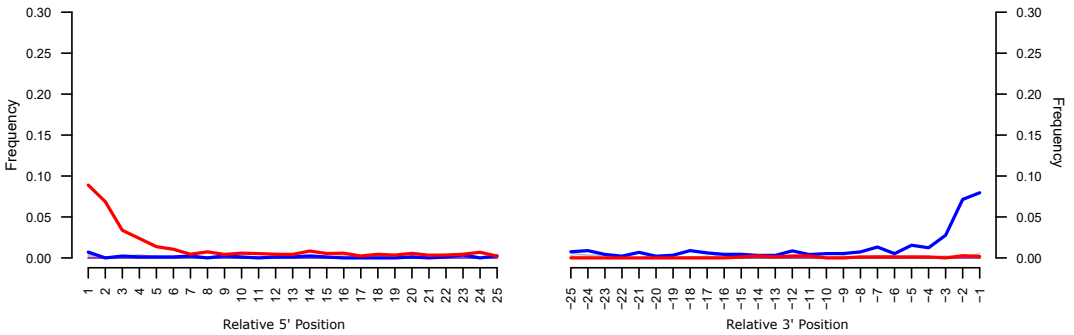

Figure 7.7: Typical misincorporation plot showing increased $C \to T$ conversion frequencies at the 5' end of mapped reads that are of ancient origin (red line), as well as increased $G \to A$ conversion frequencies at the 3' end (blue line). The y-axis denotes the percentage of sites containing a nucleotide change from the reference, and the x-axis denotes the relative position along the reads. In this example, reads originated from an ancient *Mycobacterium leprae* bone sample, called SK8 [144].

to each sample in the data set, results in a `VCF` file for each sample, which can be used for genome reconstruction. For this purpose, the tool VCF2Draft [57] that was implemented by Alexander Herbig, is used to transform the `VCF` files into `FASTA` files, allowing the user to additionally apply filter criteria for SNV incorporation. By default, a reference base is used, if there are at least 5 reads covering the respective base and the quality of the call is at least 30. If, in addition to these parameters, the fraction of the reads containing a variant call is at least 90%, then a SNV is incorporated instead of the reference base. If these requirements are violated and neither a reference call nor a variant call can be made, then the character 'N' is inserted.
However, to distinguish between clear calls and borderline decisions, VCF2Draft produces three types of `FASTA` files:

Type 1: In this `FASTA` file the character 'N' indicates bases, where neither a reference call, nor a variant call could be made.

Type 2: In this file, the character 'N' is replaced with symbols 1, 2, 3, and 4 encoding A, T, C, and G respectively, to illustrate cases in which a specific call was rejected due to low coverage, although the mapped reads suggest a specific call.

Type 3: Here, all 'N' characters from the type 1 file are replaced with their respective reference base, disregarding the fact, that a reference call could not be made.

In order to improve the subsequent multiple whole genome alignment with MAUVE [33], a `FASTA` file that does not contain any other symbol than A, T, G, or C is required. Thus, the `FASTA` output files of type 3 are used to proceed. To conduct the multiple whole genome alignment, the `progressiveMauve` algorithm included in the MAUVE software is applied. The resulting MWGA is then imported into the MAUVE alignment viewer and a SNV table is created using the internal SNP table export feature. Thereby, each position, where at least one of the genomes in the alignment differs from the others, is reported in the resulting table. Further processing of the table involves the removal of unwanted positions for downstream analyses. In general, SNVs in repeat regions or at positions covered by negative controls are removed during this step. The last processing step involves the transformation of the resulting SNV table into formats that can be used by downstream analysis software for further processing. For a phylogenetic analysis of the genomes in the data set, usually a `FASTA` file containing only the relevant variant positions is needed. Programs such as MEGA [156] or BEAST [37] can then be applied to calculate phlyogenetic relationships between the different organisms. Another application would be the prediction of variant effects. For example, the software SNPeff allows the user to predict the effect of a specific variant on protein coding genes. However, a specific SNPeff [25] compatible input file is required. To generate the mentioned file formats the tool SNPtableAnalyzer [57] that was implemented by Alexander Herbig, has been included into this pipeline.

7.1.3 Phylogenetic Reconstruction

Inferring phylogenetic relationships can provide great insight into bacterial history and the evolution of phenotypes, such as pathogenicity. Based on the results of the multiple whole genome alignment (MWGA), a phylogenetic tree can be constructed. In this work, the software package MEGA [156] was used for this task. To perform an analysis with MEGA, the informative positions of the MWGA have to be in `FASTA` format, where each `FASTA` entry corresponds to an individual sample. The phylogenetic tree is calculated using the `Maximum Parsimony` method with the following parameters: the substitution type parameter is set to *Nucleotide* and the gaps/missing data treatment parameter to *Partial deletion* with a site coverage cutoff of 90%. This allows for an improved calculation regarding gaps introduced by MWGA. Furthermore, to test the constructed phylogeny, the *Bootstrap* method is selected with 500 replications and *Subtree-Pruning-Regrafting (SPR)* is set for the maximum parsimony search method. The constructed trees can then be exported either in a vector based graphics format, such as `PDF` or in the pixel based formats `JPG` or `PNG`.

7.1.4 Variant Effect Prediction

Pathogenicity of a microorganisms is often defined by the effects of its respective SNVs. Furthermore, the effect of a SNV largely depends on its location. It can, for example, lead to severe changes or complete loss of gene function, if located in a coding gene. There, it can lead to amino-acid substitutions, that in consequence influence the folding and function of the respective protein. However, not every nucleotide substitution consequently leads to an amino-acid change, since most amino-acids are encoded redundantly by more than one base-triplet. Therefore, one refers to a non-synonymous SNV, if it leads to an amino-acid change and the term synonymous SNV is used otherwise. SNPeff [25] is a tool that allows the user to determine, if a SNV is synonymous or non-synonymous. In addition, predictions on the impact of SNVs lying outside of a coding region are made. An example would be a substitution in a gene promoter region, leading to changes in the expression profile of the respective gene. To apply SNPeff to a set of SNVs, a database with gene annotations is needed, which can either be downloaded from the respective website [1], if available, or which can be constructed manually. In order to include additional features, such as pseudogenes or non-coding genes, which are by default not included in the pre-calculated databases provided by the SNPeff developers, a custom database needs to be built. To do so, feature annotations for the respective genome are required in the `GTF` or `GFF` file format. In the pipeline presented in this thesis, the default parameters for a SNPeff analysis were chosen as suggested by the developers, except for the *number of up-/downstream bases* that should be considered for the SNV impact prediction. This number was changed to 100 bases, in order to obtain a better coverage of the up- and downstream regions of coding genes. As input for an analysis within this pipeline, the SNPeff specific input file created with the tool SNPtableAnalyzer is used. The resulting predictions can then once again be parsed with the SNPtableAnalyzer together with the draft genomes of type 2, in order to distinguish between clear and ambiguous SNV or reference calls. The result is a SNV table for all samples in the data set, together with effect predictions for each SNV from the SNPeff tool.

7.2 Comparative Analysis of Modern *Treponema pallidum* Strains

The pathogen *Treponema pallidum subsp. pallidum* is a sexually transmitted spirochete microorganism that causes syphilis. Other human diseases caused by related *Treponema pallidum* include yaws (subspecies *pertenue*), pinta (subspecies *carateum*), and bejel (subspecies *endemicum*). However, syphilis was

[1] `http://snpeff.sourceforge.net/download.html#databases(29/10/2015)`

the most devastating disease in human history, prior to HIV. It has severe effects on the cardiovascular and neurological system and can also facilitate the transmission of HIV [61, 171]. According to the world health organization (WHO), in the year 2008, there were more than 36 million people suffering from syphilis with around 10.6 million new cases [116]. Nowadays, this disease is still a threat to human health, since the number of antibiotic resistant strains rises for second line antibiotics. As of today, there is no effective vaccine available and the only way to reduce the risk of infection is by abstinence of intimate physical contact, since even latex condoms do not provide complete protection [151]. Studies on syphilis so far mostly focused on strains that were propagated in rabbits, since the clinical symptoms are very similar to humans. In 1998 the first whole genome sequence of syphilis was published from a rabbit isolate [45]. Since then further studies have been conducted and with the rise of next-generation sequencing technologies another six different strains, namely Nichols, SS14, DAL-1, MexicoA, Chicago, and Sea81-4 were published. In addition, three whole genome sequences of yaws became available, which are Gauthier, CDC2, and Samoa D. However, all these strains are also isolated from rabbits and not clinical human samples. Although, studies on clinical samples were conducted [111], these mainly focused on specific genetic markers and not on whole genomes. Thus, the information content was insufficient for phylogenetic analyses on a population based level. In this thesis, the phylogenetic and population genetic patterns of *Treponema pallidum* were investigated. In total 64 different *T. pallidum pallidum* strains, 5 *T. pallidum pertenue*, and 1 *T. pallidum endemicum* strain were made available from various countries all over the world. In particular, 13 samples originated from Switzerland, 10 from the Netherlands, another 12 from the Czech Republic, and 4 from Argentina. Spain, Samoa, and Iraq contributed each with 1 sample, 11 samples came from the USA, 6 from Austria and UK respectively, as well as 2 from Ghana and another 2 from Indonesia. Preprocessing of the collected samples involved a genome-wide enrichment using a capture hybridization approach based on a capture microarray with 60 base pair long oligos with a 4 base pair tiling density. Paired-end sequencing of the enriched DNA fragments was performed on an Illumina HiSeq 2500 platform resulting in read lengths of 150 base pairs per read. The introduced processing pipeline was then applied to these raw data. However, some intermediate steps had to be adjusted to fulfill the specific needs of the *T. pallidum* samples. In the following, the application of the pipeline and the modifications that had to be made, are explained in detail.

Analysis of Hyper-Variable Regions and Draft Genome Generation Mapping against the Nichols reference strain (NC_021490.2) using the BWA `aln/samse` algorithm resulted in low coverage regions, where read coverage dropped below 5 reads, although the rest of the genome showed very high

coverage values. Visual inspection of these regions in the Integrative Genomics Viewer (IGV) [135] revealed that the coverage drop is most likely the result of various SNVs located in these regions. Pairwise whole genome alignments of the Nichols strain with the 10 other available genome sequences further confirmed that these regions are in fact hyper-variable. Altogether, seven different gene loci were identified that could not be covered using the default parameters for the BWA `aln/samse` algorithm, because the number of allowed mismatches was too low. These genes were TP0136, TP0326, TP0548, TP0623, TP0897, TP1029, and TP1031. Simply increasing the maximal number of mismatches, however, has not been a satisfying solution, because an introduction of false positive variations in other regions was observed as a consequence. Therefore, the BWA `mem` algorithm was used for the mapping step. This algorithm is more sensitive to regions with a large number of variations, without decreasing alignment quality too much for the rest of the genome. Using BWA `mem`, in total 29 of the samples fulfilled the criteria for draft genome generation. However, these criteria were relaxed in comparison to the defaults of the pipeline, since modern DNA is usually of high sequence quality. Thus, only a coverage of 80% of the genome with at least 3 different reads was required. A list of the remaining strains for draft genome generation is provided in table 7.1. Furthermore, switching to BWA `mem` in the pipeline resulted in a sufficient coverage for six of the seven hyper-variable regions for subsequent variant calling. For TP0136, however, some reads did not map uniquely, thus SNV calling was not possible. Further investigations of the TP0136 gene locus showed that a small sub-region of 96 nucleotides was repeated within this gene. Furthermore, reads mapping into one of the duplicated regions did not map elsewhere in the genome. However, since it is unclear whether variations in the duplicated region result from the first or the second copy, inclusion of such variations was not trustworthy. Variant calling was performed using GATK's `Unified Genotyper`, following the pipeline guidelines. However, the parameter for the minimal coverage of a SNV position was reduced from at least 10 different reads to at least 3 reads. The default value of 10 offers a good trade-off when ancient DNA, in addition to modern DNA, is being processed. However, here only modern strains were analyzed and DNA fragments from the incorporated *T. pallidum* strains were expected to result in high read qualities during sequencing. Thus a reduction of the coverage per variant position was reasonable.

Analysis of 23S rRNA Antibiotic Resistance Variations For *T. pallidum* the 23S rRNA gene has two operons, where one resulted from a duplication event from the other. Two mutations in these regions, namely A2058G and A2059G, have been associated with antibiotic resistance in clinical samples, i.e. resistance to azithromycine [24]. Therefore, these mutations have a high clinical significance and should be included in the draft genomes for downstream analyses. Based on the `GFF3` annotation file available for the

Table 7.1: Details on the 29 *T. pallidum* samples that fulfilled the requirements for draft genome generation (a coverage of ≥ 3 reads for at least $\geq 80\%$ of the genome). TPA stands for the subspecies *Treponema pallidum pallidum* and TPE for the subspecies *Treponema pallidum pertenue*.

Strain	Country	Source	Subspecies	Cov. ≥ 3
S1	Switzerland	Clinical	TPA	99.98%
S2	Switzerland	Clinical	TPA	84.07%
S4	Switzerland	Clinical	TPA	99.53%
S6	Switzerland	Clinical	TPA	99.82%
S8	Switzerland	Clinical	TPA	99.99%
S13	Austria	Clinical	TPA	90.62%
S15	Austria	Clinical	TPA	100%
S16	Austria	Clinical	TPA	99.97%
S17	Austria	Clinical	TPA	99.78%
N12	Netherlands	Clinical	TPA	97.46%
N13	Netherlands	Clinical	TPA	95.11%
N14	Netherlands	Clinical	TPA	95.73%
N15	Netherlands	Clinical	TPA	99.76%
N17	Netherlands	Clinical	TPA	99.99%
N19	Netherlands	Clinical	TPA	97.85%
N20	Netherlands	Clinical	TPA	99.99%
C27	Czech Republic	Clinical	TPA	99.99%
C33	Czech Republic	Clinical	TPA	91.72%
ARG2	Argentina	Clinical	TPA	99.98%
UW1	USA	Rabbit	TPA	99.31%
GRA2	Atlanta, USA	Rabbit	TPA	98.13%
SEA86	Seattle, USA	Rabbit	TPA	97.22%
BAL3	Baltimore, USA	Rabbit	TPA	99.95%
BAL73	Baltimore, USA	Rabbit	TPA	99.74%
NIC 2	Washington DC, USA	Rabbit	TPA	99.93%
NIC 1	Washington DC, USA	Rabbit	TPA	100%
GHA1	Ghana	Rabbit	TPE	98.95%
IND1	Indonesia	Rabbit	TPE	99.83%
SAM1 (reseq.)	Western Samoa	Rabbit	TPE	99.93%

Nichols (NC_021490.2) strain, the mutations are located in the respective 23S rRNA gene at positions A2110G and A2111G. The Nichols wild-type (TAGACGGAAAGACCCC), however, does not carry any of these two mutations. A straight forward mapping approach followed by variant calling,

was not capable of identifying mutations in these regions. This is due to the fact that reads mapping to two different locations in the reference genome get assigned a mapping quality value of 0 by the BWA `mem` algorithm and are therefore disregarded during the variant calling process. Furthermore, SNVs from these regions can only be included in the draft genomes, if they appear in both copies of the 23S rRNA operon, since identification of the correct operon from NGS data, in the case of only one operon carrying the mutation, is impossible. Therefore, all variations that are shared by only one operon had to be identified and excluded, while variations shared by both operons should be included in the respective draft genomes. To identify mutations in the 23S rRNA operons, one operon sequence was extracted from the reference genome including 200 bases on each side of the operon. Mapping was then conducted individually for the extracted operon sequence. For variant calling the settings were modified such that also heterozygous SNVs, with a major allele frequency $< 90\%$, were included in the resulting `VCF` file, since those variations indicate a difference in the two operon sequences. Investigations of the resulting `VCF` files for each strain, however, revealed that there are no heterozygous mutations in any of the strains. Furthermore, with this approach the corresponding variation A2058G for azythromicine resistance could be identified for 15 of the 29 clinical *T. pallidum* strains. For three additional clinical samples the coverage was not high enough to make a clear statement, although the mapped reads suggest that the corresponding mutation is also present. Furthermore, none of the 29 strains carried the A2059G mutation. As an additional result, this analysis showed that resistance to the second-line antibiotic azithromycin is widespread in Europe, involving all European countries in this study. Sequencing of further samples from other countries would provide even deeper insights into the current degree of resistance against azithromycin in Europe.

Multiple Whole Genome Alignment For the multiple whole genome alignment the 29 *T. pallidum* strains were supplemented with 11 additional *T. pallidum* sequences that were already available from the NCBI genome database. A list of these 11 strains, together with their respective database identifier, the country they were collected from, and their subspecies type is provided in table 7.2. Furthermore, the genome sequence of *Treponema paraluiscuniculi* was also included, which allows for an out-group analysis during phylogenetic tree construction. In a first attempt, the draft genomes of the 29 sequenced strains were aligned together with the original sequences of the 11 additional reference genomes. Investigations of the resulting alignment showed major rearrangements, especially for the reference strain MexicoA. In depth analysis of the resulting alignment revealed that the observed rearrangements occur basically due to flanking identical sub-regions. These regions correspond to the 16S rRNA, which is found in two different operons, thus the rearrangements are artifacts of the `progressiveMauve` alignment algorithm.

Table 7.2: *T. pallidum* reference genomes that were obtained from the NCBI genome database and used to complement the 29 newly sequenced *T. pallidum* strains for phylogenetic analysis. In the Treponema subspecies column TPA stands for *Treponema pallidum supsp. pallidum* (agent of syphilis), TPE for *Treponema pallidum subsp. pertenue* (agent of yaws), and TEN for *Treponema pallidum subsp. endemicum* (agent of bejel, endemic syphilis).

Strain	NCBI RefSeq ID	Country	Subspecies	Source
Nichols	NC_021490.2	Washington DC, USA	TPA	Rabbit
SS14	NC_021508.1	Atlanta, USA	TPA	Clinical
Chicago	NC_017268.1	Chicago, USA	TPA	Rabbit
MexicoA	NC_018722.1	Mexico	TPA	Rabbit
DAL-1	NC_016844.1	Dallas, USA	TPA	Rabbit
SEA81	NZ_CP003679.1	Seattle, USA	TPA	Rabbit
Fribourg	NC_021179.1	Guinea, West Africa	TPE	Baboon
SamoaD	NC_016842.1	Western Samoa	TPE	Rabbit
CDC2	NC_016848.1	Akorabo, Ghana	TPE	unknown
Gauthier	NC_016843.1	Ghana	TPE	Rabbit
BosniaA	NZ_CP007548.1	Bosnia	TEN	Rabbit

To prevent such artificial rearrangements that would lead to falsely discovered SNVs, a different alignment approach had to be taken. Since investigations of the conducted multiple whole genome alignment also showed that the main differences of the included strains result from SNVs, it seemed reasonable to calculate draft genomes for the already available *T. pallidum* strains, too. With this strategy the positions of the SNVs in the genomes correlate with the SNV positions of the newly sequenced strains, which results in an improved multiple whole genome alignment without artificial rearrangements. To calculate draft genomes from the available *T. pallidum* strains, artificial reads were produced by cutting the genome sequences with a 6 base pair tiling approach using a sliding window of length 150. Base pair qualities in the resulting `FASTQ` files were set to the maximum value in the Illumina 1.8 phred score encoding scheme, which corresponded to the letter I. These artificial reads were then used to be processed by the pipeline in the same way as the 29 newly sequenced strains. Afterwards, draft genomes for all 40 strains as well as the out-goup *T. paraluis-cuniculi* were available for multiple whole genome alignment. Since now all of the incorporated genome sequences were based on the Nichols reference, no

large rearrangements were observed and SNV calling using the integrated SNP table extraction feature in the MAUVE alignment software could be applied as suggested by the pipeline.

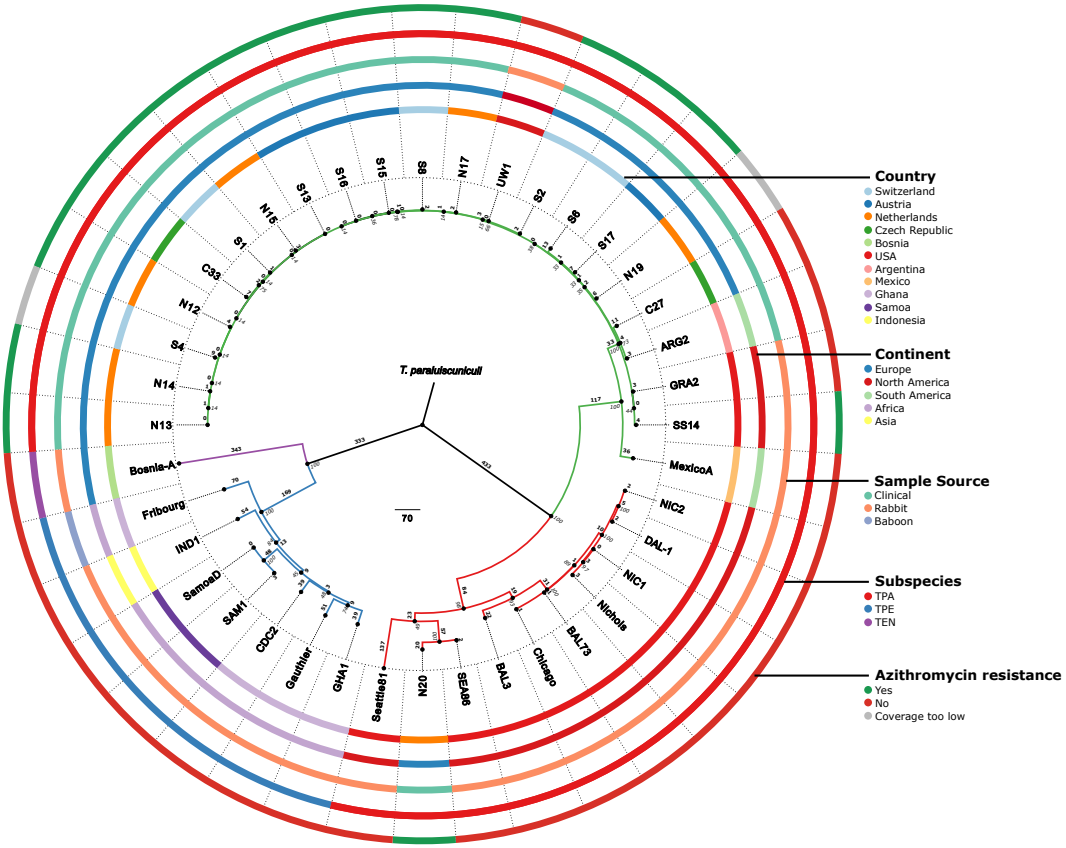

Figure 7.8: Phylogenetic tree of 40 different *T. pallidum* strains, including 32 *T. pallidum subsp. pallidum* strains, 7 *T. pallidum subsp. pertenue* strains, and 1 *T. pallidum subsp. endemicum* strain. *T. paraluiscuniculi* was used as an out-group. The strains cluster into three different branches, highlighted with different colored edges (green: SS14-like syphilis, red: Nichols-like syphilis, blue: yaws, purple: bejel). Numbers on the edges represent the phylogenetic distance between respective nodes as the absolute number of single nucleotide substitutions. The significance of phylogenetic distances was assessed with bootstrap values written in italic style next to the internal nodes of the phylogenetic tree. Colored arcs around the phylogenetic tree display additional meta-information for each of the strains.

Phylogenetic Analysis The phylogenetic relationships between the 40 samples, constituted by 29 clinical strains and 11 references, together with 1 out-group, were computed in MEGA 5 using the Maximum Parsimony (Subtree-Pruning-Regrafting) method and partial deletion of missing data. Thereby,

sites with more than 10% missing data had been removed, reducing the initial number of 2329 different variants to 1487 variants that could be used for phylogenetic analysis. For significance testing the bootstrap method with 500 replications had been chosen. The resulting phylogenetic tree is shown in figure 7.8. The strains cluster mainly into two different branches, one representing SS14-like samples and the other containing Nichols-like samples. All clinical samples from this study, except for one Dutch sample, fall into the SS14-like cluster. In order to evaluate why one dutch individual falls into a cluster consisting only of rabbit samples, additional information about ecological circumstances of this individual would have to be collected. Furthermore, no clear separation of clinical samples from different European countries can be made. The second branch consists only of *T. pallidum subsp. pertenue* samples and one *T. pallidum endemicum* sample. Clearly, a separation between different *T. pallidum* subspecies can be made. The fact that some rabbit samples cluster in between clinical samples further indicates that the genetic diversity between *T. pallidum* subspecies may be larger than between human and rabbit strains. Regarding the A2058G antibiotic resistance mutation, one can see that most SS14-like strains carry this mutation, as it is the case for SS14 itself. Moreover, Nichols-like strains do not carry the mutation.

7.3 Conclusion

During this dissertation, a new pipeline for the automated processing and comparison of ancient and modern bacterial DNA samples has been implemented in cooperation with Alexander Herbig. Besides the integration of various available software solutions, new tools have been introduced that address specific needs of ancient samples, such as the merging of overlapping paired-end sequencing reads. With the application to various *M. leprae* samples in our previous publication [144], the pipelines capabilities for analyzing ancient and modern bacterial strains in parallel has been demonstrated. Furthermore, with the application to modern *T. pallidum* strains it could be shown that the pipeline is flexible and allows for modifications of individual steps, in order to address the specific characteristics of different data sets. Although, at its current point, this pipeline mainly focuses on the analysis of bacterial data, steps towards the inclusion of eukaryotic samples, especially from ancient human remains, have already been taken by Alexander Peltzer, who is going to continue the aDNA pipeline project under the acronym EAGER (Efficient Ancient GEnome Reconstruction).

8. Parallel Detection of Human Pathogens via Array-Based DNA Capture

In the previous chapters it was shown that next-generation sequencing (NGS) offers new possibilities for data acquisition and analysis. Another field of application is the identification of DNA contained in environmental samples. Together with an appropriate bioinformatic analysis, NGS can be an extremely helpful technique to achieve a better understanding of the genetic information content. Furthermore, environmental samples taken from soil, water or extracted from corpses of ancient humans or animals can provide deep insights into our history. Especially the study of bacterial DNA, which is contained in almost every sample, has huge potential in solving questions of ancient pathogenicity, bacterial development in general or the evolution of pathogenic agents. In this way, the extraction of biological material containing DNA from pathogenic organisms can help to explain historical events such as pandemics. However, to study the organism that led to such historically relevant events it is important to isolate the DNA of those organisms of interest from a conglomerate of different organisms that are also present in every environmental sample. For example, when dealing with soil samples, a whole collection of different microorganisms, especially bacteria, archaebacteria and viruses, but also other organisms, such as flies, worms or larger animals, are necessarily present. Furthermore, not all strains of a specific microorganism are pathogenic, meaning that it is important to also isolate non-pathogenic strains from the species in question. To do so, appropriate methods are needed, since a rather straight forward approach, such as performing a shotgun sequencing of the whole DNA that is present in the sample and doing a meta-genome analysis afterwards might not lead to satisfying results. A lack of sufficient enrichment can lead to an underrepresentation of DNA fragments from the organisms of interest, making them undetectable in the sequencing and subsequent mapping process. Moreover, the isolation of ancient DNA, which in comparison to modern DNA suffers from the availability of only short DNA fragments as well as from DNA degradation [117], is extremely difficult. To overcome this hindrance, DNA capture is highly suitable to filter those DNA fragments belonging to the organisms of interest. This is usually done using DNA capture microarrays with specifically designed probes that bind to the organism's DNA fragments. Unfortunately, already available techniques are so far restricted to single organisms only [18, 143]. However, when dealing, for example, with ancient tissue it is often not clear what type of infection had been present. Mor-

phological changes lack specificity and co-infections could have occurred [93]. This renders it difficult to decide which pathogen's DNA fragments might be present and consequently which DNA capture array should be used. Furthermore, progress in the field was hindered by the lack of an economical screening technique that is capable of detecting a larger number of different pathogens in parallel. A solution would be to use DNA capture microarrays that are able to detect multiple organisms in parallel rather than concentrating on one pathogen at a time. Ideally, such an approach should include bacteria, DNA viruses, protozoa and multi-cellular organisms with both, a high specificity and sensitivity. In this thesis, such an approach that is capable of detecting up to 92 different pathogens was implemented and named APSA, Ancient Pathogen Screening Array. For manufacturing, the Agilent SureSelect 1-million feature DNA capture array with oligo lengths of 60 base pairs was used. Although, the array and the subsequent bioinformatic analysis are designed for the detection of pathogens from ancient samples, the overall analysis process is not restricted to those and can also be applied to samples from modern sources.

In this chapter, detailed information is given on the bioinformatic challenges in the design steps involved in the identification of pathogen specific regions, followed by information on oligo preparation and selection. This is completed by details about the analysis of the APSA captured DNA fragments, including the processing of the sequenced reads as well as the subsequent data cleaning, mapping and visualization techniques. Text and figures in this chapter were adapted with minor modifications from work previously published in [17].

8.1 Design of the APSA

The precise detection of specific pathogens bears several bioinformatic challenges that have to be addressed in the design of a DNA capture microarray. First of all, pathogens differ in the size of their genomes. This means that pathogens with large genomes are more likely to be captured than those with relatively smaller genomes when assuming equal distribution of organisms in an environmental sample. Secondly, it has to be guaranteed that the captured sequences are unique to the corresponding pathogen, especially then, when not all strains of a specific species are pathogenic. Furthermore, special attention has to be given to the analysis of the captured data, since technical issues such as cross-hybridization cannot be excluded completely.

In the following, the design pipeline for the pathogen specific DNA capture microarray will be explained and detailed information will be given regarding the bioinformatic challenges and how they have been addressed.

8.1.1 Identification of Pathogen Specific Genomic Regions

When dealing with microorganisms, pathogenicity can arise on different hierarchical levels. While some organisms are pathogenic on the species level, for others only a subset or even a single strain is pathogenic. This fact has to be taken into account during the design of the pathogen specific oligos to make a clear separation of pathogenic and non-pathogenic microorganisms. Clearly, each individual organism has to be treated differently depending on its specific level of pathogenicity. A summary of the strategy chosen in this work for the identification of pathogenic sequences is shown in figure 8.1(A). In total 92 different microorganisms have been selected covering most of the pathogens that could be present in archival samples including bone, dental pulp, mummified tissue or anatomical collections of soft tissue. However, RNA viruses have been omitted, because of their predicted poor conservation in ancient tissue [112]. Since for most of the selected pathogens pathogenicity was shared between several different strains, the "least common ancestor (LCA)" [1] of all these strains had to be chosen for oligo design. In order to identify the respective LCAs for the 92 selected pathogens, taxonomy identifiers were assigned based on NCBI's taxonomy database [1]. This database contains unique taxonomical identifiers for each known organism as well as their ancestral relatives. It can therefore be seen as a phylogenetic tree, where each leaf node as well as each internal node has been given a unique taxonomy ID. Thereby, IDs have been distributed such that internal nodes always have lower IDs than nodes in the subtree rooted at the respective internal node. For the identification of unique pathogenic regions for oligo design, each individual is not given its own taxonomy ID, but the taxonomy ID of the respective LCA. In a subsequent sequence comparison analysis all similar subsequences between different individuals sharing the same taxonomy ID are treated as self-hits, which allows one to address pathogenicity at different hierarchical levels defined by the assigned taxonomy IDs. Genome sequences, however, are only available on the strain level. Therefore, for organisms where an LCA could be found, one of the strains in the taxonomical subtree was chosen randomly as the reference. This was repeated for all of the 92 pathogens until one strain for each had been chosen. Sequences for the selected strains were then obtained from the NCBI genome database [1] for further processing.

The next step in the design pipeline is the identification of unique regions in the genome of each pathogen. Therefore, all collected genome sequences were used for a subsequent BLAST search against the NCBI BLAST nucleotide sequence database (*nt*) using the `blastn` algorithm (version 2.2.26) [3]. This nu-

[1] LCA is here not meant in an evolutionary, but just hierarchical sense

cleotide collection database consists of GenBank [13], EMBL [75], DDBJ [105], PDB [14] and RefSeq [128] sequences and is therefore perfectly suited for the identification of subsequences in the genomes of the 92 pathogens that are common among other organisms. An E-value threshold of $\leq 10^{-3}$ was used in order to be specific enough during the BLAST search. Since the *nt* database also contains the genomes of our selected pathogens, only those hits are relevant that are truly similar to a different organism, as defined by the assigned taxonomy IDs. To exclude self-hits from the results, each BLAST hit was assigned a taxonomy ID, too, allowing to filter based on these taxonomy IDs. Each hit with an ID equal to or greater as the query's taxonomy ID was removed from the output table, since those hits correspond to the same branch in the underlying phylogenetic tree and are therefore declared as self-hits. With this strategy, all hits corresponding to an organism below the defined LCA of the respective pathogen were considered as unique regions, while the remaining hits in the BLAST output correspond to non-unique regions. The sequences from the remaining hits were then collected and removed from the genomes of the 92 pathogens used for the array, resulting in subsequences for each pathogen that can be used for oligo selection. These regions ranged from 84 up to 3,291,871 base pairs.

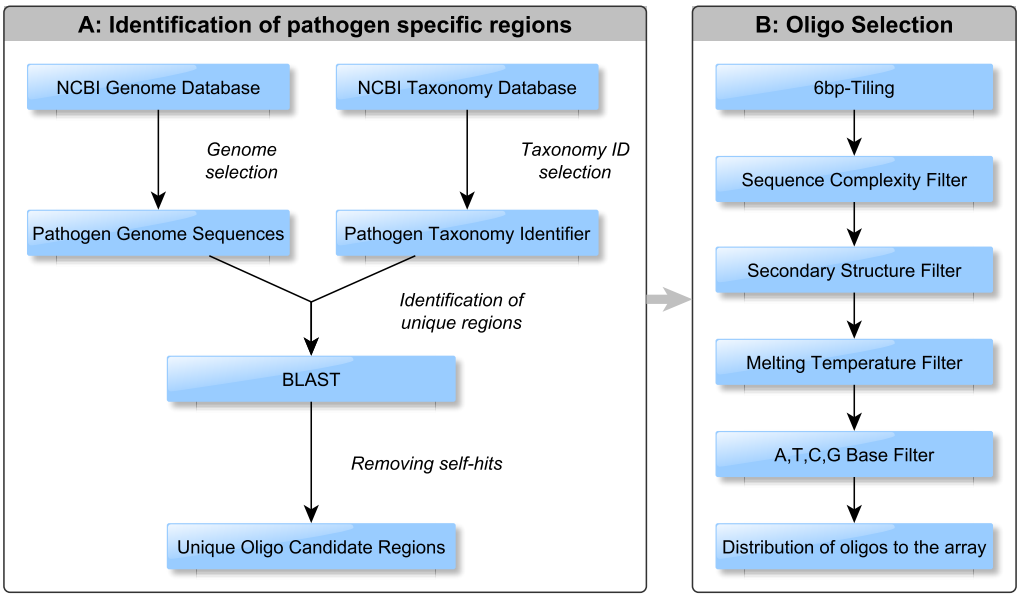

Figure 8.1: Overview of the work-flow followed by the oligo design pipeline of the APSA. It can be divided into two parts, where **A** shows the steps needed for the identification of unique genomic regions for each pathogen and **B** shows the selection process from these resulting regions.

8.1.2 Oligo Selection from Pathogen Specific Genomic Regions

In order to generate oligos from the unique regions identified by the taxonomy and sequence search based strategy explained above, a 6-base-pair-tiling approach was used with subsequent filtering for oligo quality. In figure 8.1(B) a summarized overview of the oligo selection procedure is given. Unique regions were dissected into oligo candidates of 60 base pairs in length, where adjacent oligos share an overlap of 54 base pairs. This strategy provides a good trade-off between sequences being unique and not loosing too much sequence information for the actual fragment capturing. However, uniqueness with respect to a specific pathogen is not sufficient for good oligos, because of the technical circumstances that have to be addressed in the microarray hybridization step. Here, additional quality measures have to be fulfilled such as a good affinity to the target sequences (sensitivity criterion) as well as sharing approximately the same melting temperature in comparison to the other oligos on the array (isothermal criterion). Only oligos that are specific, sensitive and isothermal can be used for the array [87]. Moreover, additional filtering had to be performed for all candidates. Therefore, basically three different issues have been addressed, namely sequence complexity, the ability to fold into a stable secondary structure and the melting temperature. Sequence complexity has been addressed by implementing a measure of complexity, that is based on a length comparison of the condensed sequence of an oligo candidate with the candidate's original sequence. The condensed sequence is calculated by using the Lempel-Ziv-Welch lossless data compression algorithm [169]. Oligos were filtered such that only those with a length ratio ≥ 0.5 between the condensed form and the original sequence remained in the candidate set. The secondary structure criterion was addressed by applying the `mfold` (version 3.5) algorithm [175] to each oligo candidate and calculating the Gibbs free energy (ΔG) of a possible folding. Filtering was then performed such that only those oligos with a positive ΔG value (endothermal reaction) remained in the candidate set. The third filtering step addressed the isothermal criterion by calculating the melting temperature of each remaining oligo candidate and keeping only those with a melting temperature in the range of 60 to 85° C.

After that, one last filter had to be applied in order to exclude those regions in the genomes where nucleotide resolution was not sufficient enough and ambiguous bases, other than A,T,C, or G were present. After removing all oligos that contained ambiguous nucleotides, the remaining set of candidates fulfilled all the criteria for being good oligos for a DNA capture microarray. The resulting number of available candidates, however, exceeded the number of oligos that could be used for the array. In total 974,016 oligos are permitted on the Agilent 1-million feature array. This means that on average 10,587 oligos can be used

for each of our 92 pathogens. However, due to the large differences in genome size and the strict filtering criteria, for only a few pathogens this number of oligos was still available. In fact, for most of the pathogens much less than the average number of oligos remained. Therefore, a different strategy had to be applied. For those pathogens, where less than the average amount of oligos was available, all the remaining oligos were selected. For those pathogens, were more than the average number of oligos was available, the remaining spots on the array were distributed equally. With this strategy, the number of oligos for each pathogen differs from the other pathogens, which has to be accounted for during the analysis of the array capture results. This is important, because DNA from those organisms with large numbers of oligos on the array is more likely to be captured than DNA from organisms with only a few oligos, such as viruses.

8.2 Analysis of APSA captured DNA

When analyzing captured DNA from potentially ancient origin, special care has to be taken during the pre-processing of the sequenced DNA fragments. Higher damage rates are expected and DNA fragments are usually much shorter than modern DNA fragments. Therefore, a paired-end sequencing was performed with read lengths of 100 base pairs for the forward as well as for the reverse read. Due to the short fragment size of the captured DNA, a positive insert size for the paired-end reads can not always be guaranteed. In the following, detailed information is given on the pre-processing of APSA captured paired-end reads and the subsequent read mapping and counting procedure. The latter is used to determine the number of reads mapping to each organism, which is correlated to the organism's amount of DNA present in the sample. An overview summarizing all steps of the analysis protocol is shown in figure 8.2.

8.2.1 Captured Read Preprocessing and Mapping

For the processing of the APSA captured paired-end reads the newly developed ancient and modern DNA processing pipeline described in chapter 7 was applied including all steps up to the mapping procedure. However, some minor modifications have been made with respect to the huge diversity of the organisms contained on the APSA. Quality assessment, adapter clipping, read merging as well as quality trimming were performed as described in chapter 7. Mapping itself was performed using the BWA `aln/same` algorithm keeping all parameters as suggested by the BWA developers, except for the parameter that controls the number of allowed mismatches ($-n$). This parameter was increased from 0.04 to 0.1 in order to decrease the average number of mismatches for a read of 100 nucleotides in length from at most 6 to no more

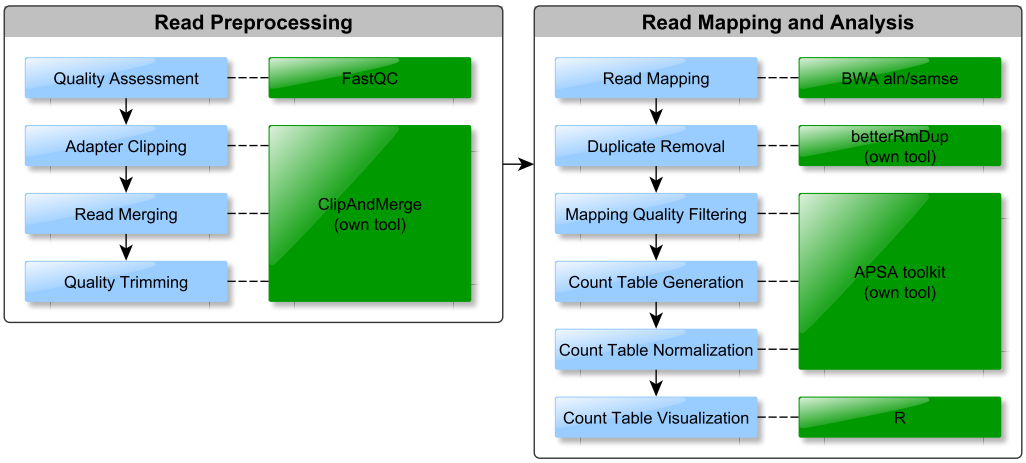

Figure 8.2: Overview of the APSA read processing and analysis strategy. Blue boxes indicate a specific step of the processing/analysis pipeline and green boxes represent the tool used at the respective step.

than 2. With that, a highly specific mapping of the captured reads can be guaranteed. All genomes used for the oligo design of the APSA were combined in a single `FASTA` file that was used as the reference for the mapping. Post-processing of the mapped reads included removal of non-uniquely mapped reads as well as non-mapped reads. For that the samtools toolbox [91] was applied. Afterwards, potential PCR duplicates were removed using the BetterRmDup method, thereby following again the steps of the pipeline introduced in chapter 7. Since the purpose of the APSA is the identification of pathogens in environmental samples, rather than a detailed analysis of individual organisms and/or their genomes, no further steps of the pipeline have to be applied after duplicate removal. In fact, the resulting mapped reads are directly used for a subsequent pathogen specific read count analysis.

8.2.2 APSA Read Count Analysis

After mapping all reads to the reference genomes of the pathogens selected for the array, the amount of captured DNA for each of the APSA specific pathogens has to be calculated. For this purpose, the reads were, after duplicate removal, filtered based on mapping quality, to make sure that only highly reliable reads are taken into account. The BWA `aln/samse` algorithm reports mapping qualities in the range of 0 to 37. By default all reads with a mapping quality ≥ 25 undergo the subsequent counting process. During this process, the reads mapping to a region in the genomes of the APSA specific pathogens are counted and reported. In order to distinguish between reads that map to

an oligo region and those that do not map to such a region, a minimal overlap of at least 10 bases is required for a read to be counted. After following these strategies a count table is generated that holds the number of reads mapped to an oligo region for each of the selected pathogens on the array. As mentioned in the APSA design description, pathogens that have a large number of oligos on the array are more likely to capture DNA fragments than those with lesser numbers. Therefore, an additional normalization of the raw count table is required. Here, for each pathogen the number of read counts is divided by the respective number of oligos on the APSA. This gives the relative amount of DNA mapped to the array for each organism. Both tables, the raw count table as well as the normalized count table, are reported for post-processing and interpretation.

8.2.3 Visualization of Read Count Results

For a better interpretation of the resulting count tables, a scatterplot visualization has been implemented that shows the APSA results and allows one to spot overrepresented pathogens immediately. An example visualization showing results for a *Mycobacterium leprae* positive control is given in figure 8.3. On the x-axis the different pathogens are shown sorted lexicographically. On the y-axis the normalized number of mapped reads is shown. For species located next to each other on the x-axis, the respective read count values are connected with lines and by default the largest 10 count values are labeled with the respective organism's name. This number can, however, be adjusted by the user. In this, overrepresented pathogens show up as spikes in this type of scatterplot, as can be seen in figure 8.3 for *M. leprae*. This visualization is implemented in the statistical programming language R. The only required input file is the count table in `TSV`-format, where each row represents a pathogen and each column a specific sample. If more than one sample is contained in the count table, separate plots are produced for each of the samples.

8.2.4 The APSA Analysis Toolkit

To simplify the process of analyzing APSA captured read data, the APSA analysis toolkit has been implemented. This is a fully automated program that allows users to produce raw as well as normalized count-tables from paired-end reads resulting from APSA captured DNA fragments. It offers a user-friendly front-end that is shown in figure 8.4. In order to perform an analysis, the specifications of the array have to be known. Therefore, an APSA specific design file has been created. This design file contains information about the number of oligos and their locations in the respective genomes for each of the pathogens on the array. Together with this information, the APSA array toolkit uses mapped reads in the `SAM/BAM` file format to generate counts

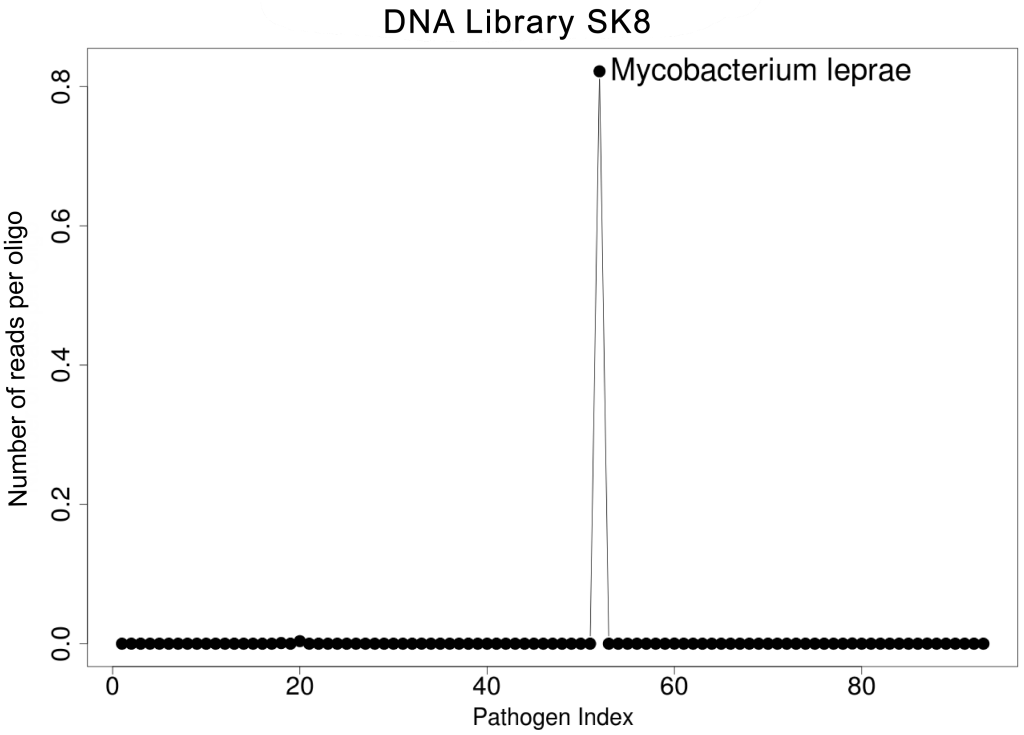

Figure 8.3: Scatterplot showing the normalized read count results for the *Mycobacterium leprae* SK8 positive control [144]. On the x-axis the pathogens on the APSA are ordered lexicographically and on the y-axis the normalized number of reads per oligo is shown. Here, only the top hit is labeled due to the large difference to and the very low number of mapped reads for the second best hit.

for each pathogen. In addition, it is possible to analyze multiple samples at the same time. To do so, multiple `SAM/BAM` files can be provided and are processed altogether in parallel. This results in count tables, where rows represent pathogens and each sample is represented by a separate column. Additional filter opportunities allow the user to include only those reads in the counting procedure that fulfill a specific mapping quality threshold.

8.3 Application of the APSA Capture Technique

The APSA capture approach was applied to several different negative control samples for the estimation of the number of falsely discovered organisms as well as to a positive control sample that was known to contain ancient leprosy DNA. With this strategy, the false discovery rate as well as the specificity and sensitivity of the APSA were measured.

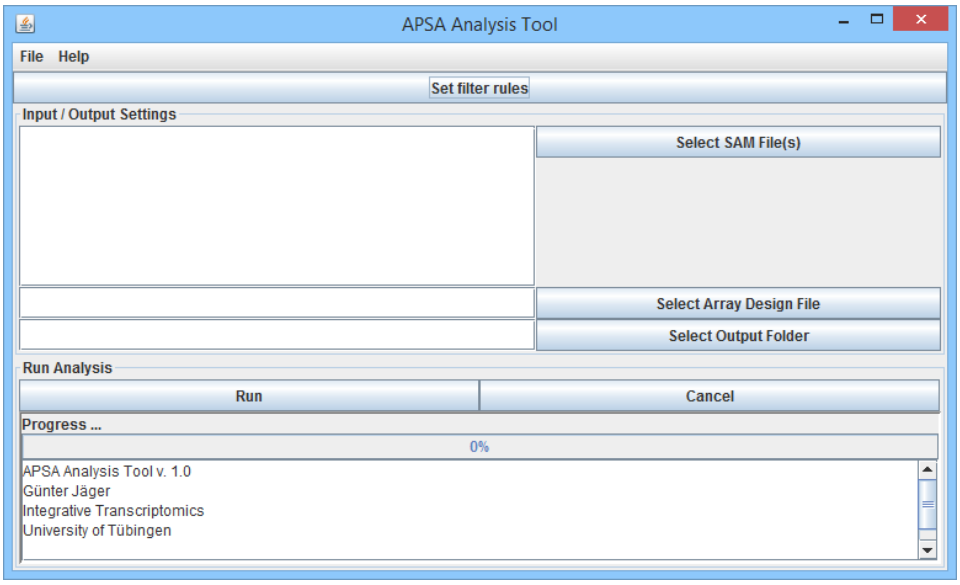

Figure 8.4: Graphical user interface of the APSA analysis toolkit for the counting and normalization of mapped paired-end reads resulting from the APSA capturing approach.

Application to Negative Controls For an appropriate estimation of the false discovery rate of the APSA capture approach, shotgun data of several different negative control samples were produced constituting between 71,766 and 279,577 reads per sample. In addition, APSA capture reads (between 44,761 and 991,460) were compared to the shotgun data. The majority of the organisms that showed an enrichment in the APSA approach had no detectable reads in the shotgun data. The detection of a few soil dwelling organisms, such as *V. parvula, E. coli or B. cepacia* can be explained by the ubiquity of these bacteria in the environment. Although, some of these organisms show up during the analysis with rather low read counts (between 1 and 34), the introduced capture approach coupled with the high mapping strictness yields only very few false positives.

Application to a *Mycobacterium leprae* Positive Control To show the detective power of the APSA combined with the strict analysis strategy, sequencing of a *Mycobacterium leprae* positive control sample was conducted. Here, the library SK8 from a medieval bone sample from the United Kingdom was used. The ancient origin as well as the containment of *Mycobacterium leprae* in this sample was shown in Schuenemann *et al.* in 2013 [144] via damage plot investigations using MapDamage [74]. This sample contains enough preserved DNA to conduct a whole genome assembly. Again shotgun data as well as APSA captured sequencing data were produced resulting in 3,058,969

shotgun reads and 213,611 APSA captured reads. From the shotgun reads 149 mapped to unique capture regions and 148 mapped to *M. leprae* specific regions. For the APSA 4,774 reads mapped to capture regions. Out of these 4,756 reads solely mapped to *M. leprae* resulting in a 460-fold enrichment and 1.16 hits per probe. Duplicate removal after mapping reduced the number of uniquely mapping reads to 3,656. To verify the ancient origin of the mapped reads damage plots were produced showing about 10% C to T damage at the 5' ends. The remaining 18 reads that did not map to *M. leprae* all belong to *B. pseudomallei*, a pathogenic organism, that leads to meloidosis in humans. Since this organisms survives in soil and water, its environmental origin cannot be excluded.

8.4 Conclusion

In this chapter, an economical screening technique has been presented that is capable of detecting up to 92 different pathogens in parallel. The microarray based approach thereby requires a number of different quality criteria that had to be addressed. For this purpose a custom oligo design pipeline has been introduced, followed by specialized read processing pipeline with a focus on the analysis of ancient DNA samples. For the design of the array, a new bioinformatic approach has been introduced with the application of taxonomy identifiers for the definition of oligo uniqueness. Furthermore, applications of the array and the subsequent processing pipeline have shown that the presented approach yields only very few false positives. Moreover, for true positives very high enrichment rates can be achieved, which shows the high sensitivity of the array. Furthermore, specificity of the array and the subsequent analysis is guaranteed by unique and high quality oligos produced following the presented design work-flow, as well as a highly stringent mapping procedure used for the identification of captured DNA fragments. To simplify the analysis of mapping results, the APSA toolkit has been developed that offers read count calculation and normalization with a comprehensive graphical user interface. Altogether, the APSA capturing together with the introduced processing strategy and the subsequent application of the APSA toolkit for read count analysis perform well for the detection of pathogens present in biological samples.

9. Discussion

For years researchers have been struggling with the question what makes an organism unique. Nowadays, we know that an organism is first of all defined by its genome. It is the sequence of nucleotides that dictates the nature of life. Modifications of that sequence may lead to genetic diversity but also to malfunction. Life largely depends on the genes and the interplay of their products. Alterations in these complex networks often lead to disease. Thus, understanding which genetic variations influence gene function, the way genes interact with each other and how this correlates with disease, has become one of the most important questions in current biomolecular research. With the development of next-generation sequencing technologies, a new era has begun for the detection of genetic variations and the study of their phenotypic implications. With these technologies, it has become possible to economically decipher an organism's complete DNA sequence, paving the way for more comprehensive analyses of genome sequence variations by studying thousands of individual genomes from one species.

The goal of this dissertation was to develop new visual and analytical approaches for the identification, characterization and interpretation of variations with a large focus on single nucleotide variations (SNVs), the most common source of genetic diversity.

MAYDAY, an expression analysis workbench, has been used as a framework and extended on the data handling as well as visualization level to allow for more comprehensive analyses with respect to variation and expression data.

For the visual analysis of genotype and phased haplotype data, INPHAP has been developed. At the time of writing, INPHAP was the only interactive visual analytics tool capable of displaying and exploring phased haplotype data. In previous studies in out group, the data manipulation strategies, and here most prominently the aggregation technique, as implemented in the iHAT toolkit [54] have proven powerful for the identification of meaningful patterns with respect to SNV data. In INPHAP these have been integrated and extended. With the application of INPHAP to data from the 1000 Genomes project, it could be shown that its visualization concepts together with the interaction possibilities are a valuable tool for the identification of population specific SNV patterns.

INPHAP was complemented by and combined with REVEAL. While INPHAP is restricted to genotype data, REVEAL offers integrative analyses of SNV and gene expression data. Due to its various statistical methods, the comprehen-

sive visualizations and the high level of user interactivity, and because of the integration into MAYDAY, REVEAL became a powerful software solution for the analysis of GWAS and eQTL data. Its value as a visual analytical tool has been proven in the BioVis 2011 challenge, where REVEAL was selected as the visualization experts' favorite.

GENOMERING is an important tool that takes the visualization of SNVs together with structural variations to the genome-wide level. In this, differences between organisms are intuitively shown in a circular layout and SNV information can be mapped to the respective genomes in the resulting plot. In this work, special attention was given to the optimization of the visual experience with GENOMERING through the development of different layout optimization strategies with the purpose to reduce visual clutter. Furthermore, GENOMERING has been integrated into MAYDAY, which allows for the visualization of gene information as well as SNV data from REVEAL in the context of a multiple whole genome alignment.

Another important and immensely growing field where the analysis of genomic variations plays an important role is paleogenomics. There, DNA of ancient origin is studied and compared to modern samples to elucidate evolutionary history. For this purpose, a computational pipeline for the comparative analysis of NGS data from modern and ancient DNA samples has been developed in cooperation with Alexander Herbig. One essential part of this pipeline is the read merging procedure, which was developed in this work and compared to other read merging approaches. This pipeline was successfully applied to distinct use cases, as for example for the comparative analysis of medieval and modern *Mycobacterium leprae* strains. In this thesis, the general applicability and the advantages of the read preprocessing steps in the pipeline also for the analysis of purely modern DNA samples were demonstrated with a comparative analysis of different *Treponema pallidum* strains from all over the world in order to investigate the evolution of the syphilis disease (see chapter 7, section 7.2). One of the major problems with aDNA is that it is usually only retrievable in very low amounts and that samples are contaminated with modern DNA, especially from microorganisms. In this work, a microarray based DNA capture techniques has been developed for the parallel enrichment and analysis of aDNA from human pathogens. As a proof of concept, this technique was applied to a *Mycobacterium leprae* positive control. The resulting data could subsequently be successfully analyzed with the aforementioned pipeline showing enrichment rates $> 460\times$.

Altogether, the methods developed in the course of this dissertation provide a valuable contribution to biologists, clinicians, researchers and bioinformaticians struggling with the analysis and interpretation of single nucleotide varia-

tions and complement existing solutions for variant detection and visualization. In the following, each of the developed tools will be discussed in more detail.

9.1 MAYDAY, a Framework for the Integrative Study of Gene Expression and Variation Data

Due to the large amount of data that can be produced nowadays for various different research fields, there is continuously growing importance of integrative software solutions offering the possibility to elucidate all facets of different data types. This dissertation focused on the integration of variation data, especially single nucleotide variations, and gene expression data in order to identify genetic factors leading to gene expression changes and consequently to disease. For this purpose, MAYDAY, an expression analysis software with a large focus on visualization of gene expression data, has been used as a general framework for the integration of variation data and the development of new visualizations approaches. The choice to use MAYDAY, rather than any other software or starting from scratch, was motivated by several different factors. First of all, MAYDAY has been designed as a general framework that allows for an easy integration of new algorithms and visualizations through a flexible plugin system. In addition, it already offers a powerful visualization framework, which makes it easy to develop and integrate new visualizations, since basic view elements and functions are already available. With this, a software developer can concentrate on the main aspects of a new visualization and does not have to take care of functionality that is needed regularly, as for example zooming. Furthermore, MAYDAY is an open source project written in the Java programming language with the advantages of addressing a large user community and being platform independent. However, MAYDAY has been designed with a strong focus on gene expression data. Thus, all methods and algorithms, as well as the underlying data structures are not readily applicable to other data types, such as variation data.

To address this hindrance and to provide a general design concept for the integrative study of other data types within MAYDAY, a detailed description of how MAYDAY can be adapted to support other data sources by using already existing data structures has been given in chapter 3. This general approach allowed for the integration of variation data in the form of new plugins. The major advantage of this approach is that it does not require to change original data structures. Consequently, features for the analysis and visualization of variation data in MAYDAY can be integrated only if needed, without forcing users to make themselves familiar with the structures required

for the additional data types, if only gene expression analyses are of interest.

Clearly, a direct modification of the `DataSet`, the main class providing access to all available data in MAYDAY's data model, would also have been possible in order to introduce new data sources. However, this idea was rejected for two different reasons. Firstly, a direct modification would have required additional modifications in the future, if other kinds of data were to be integrated. Secondly, a modification of the `DataSet` would not have been possible without larger structural changes in the MAYDAY core application and thus risking to break any core functionality needed by other plugins. However, the strategy taken in this thesis (see chapter 3, section 3.3), leads to better compatibility across versions and to a better stability of the core Mayday application, since new source code does not have any influence on the functionality of already existing core functions or plugins. Although, in this thesis the extension strategy was only applied for the integration of SNV data, it is not limited to this data type. Moreover, it can be applied for the integration of other valuable data types, such as methylation data, proteomics data or metabolomics data, which would allow for more systems biology based analyses in the future.

To conclude, with these design strategies, MAYDAY can be used as a completely integrative analysis software for the study of gene expression and variation data as well as other data types that may be integrated in the future.

9.2 Interactive Genotype and Phased Haplotype Visualization

When studying SNV related phenotypic changes, genotype information alone is often not sufficient to digest the mechanisms responsible for the observed phenotype. Especially complex traits are usually the result of an interplay of SNVs at different genomic locations. However, if such traits only manifest themselves when genomic modifications accumulate on the same chromosome, genotype information can not provide the insights needed to explain which SNVs are the result of the phenotypic outcome. In such cases, phase information is needed to link variations to their respective chromosome and to build haplotypes that can be associated with a specific trait.

In this dissertation, INPHAP has been developed, which is currently the first and only tool capable of analyzing and visualizing genotype and phased haplotype data interactively. INPHAP uses a tabular approach to visualize genotype data with or without phase information for diploid organisms. If phase information is missing, a clear representation of the data can be

obtained by using e.g. rows to represent individuals and columns for SNVs, respectively. Then, each cell represents a specific individual's genotype, which can either be equal to the reference, a heterozygous or a homozygous variation. However, if phase information is available, a single cell would have to be used to represent two different values, in particular one value for the maternal and one for the paternal allele. Other tools, such as Flapjack [104], separate the cells into triangles to display both values. In contrast, the two alleles are represented by two individual columns in INPHAP, which allows for additional meta-information integration and user interactivity. This design choice was highly motivated by the 1000 Genomes project data published by Abecasis *et al.*. There, also rows were used to represent individuals and columns for SNVs. To represent phased data, the authors used two rows instead of two columns. However, using multiple columns offers the possibility to include meta-information for individuals as well as SNVs with a clear visual separation. Furthermore, this allows the user to group SNVs based on the respective meta-information. An example would be the grouping based on the chromosomal origin of SNVs to compare haplotypes on the paternal and maternal chromosomes and between sub-populations in parallel.

Although, the initial design of INPHAP is focused on diploid organisms, the general concept can easily be extended to more complex genomes, as for example omniploid or polyploid ones. In such cases, the number of columns in the visualization would have to be equal to the number of alleles of the respective organism's genome. Furthermore, such an extension would also be interesting for the study of cancer genomes, where ploidity is often not even known.

A disadvantage of the table based approach and the representation of SNVs with two columns is that reordering of SNVs based on chromosomal origin can lead to comparability issues. This is for example the case, if haplotype regions are very large. Then comparability is restricted by the screen resolution of the user's workstation, because it can happen that haplotypes for the paternal and maternal chromosome cannot be displayed at the same time. A possible solution for this hindrance would be to split the main visualization panel based on paternal and maternal SNV annotation into two linked panels, such that moving within the paternal panel would lead to a synchronized movement in the maternal panel and the other way round. Such functionality could be introduced in the future.

Besides the various possibilities for user interaction, as for example zooming, selection, or switching between different visual representations, INPHAP makes use of the concept of aggregation. As has been shown in chapter 4, aggregation can be a valuable tool to assess features that would remain hidden by investigating solely the raw data. However, aggregation always

leads to loss of information, which might still be valuable. An example was shown in chapter 4, section 4.1.4, where it was no longer possible to assess rare variants after applying the maximum aggregation method. To address this issue, several different summarization strategies are offered in inPHAP, together with the possibility to revert aggregated rows if needed in order to apply other aggregation methods.

In order to provide immediate response of the main visualization in inPHAP to the discussed interaction features, the decision was made to store all data in memory before presenting them to the user. This required an appropriate strategy for data storage and visual representation. The binary encoding for genotype data introduced in this thesis, allows for the analysis of whole chromosomes (e.g. for data from the 1000 Genomes project) with only a little more than 20 gigabytes of RAM. However this number could be reduced in a future release of inPHAP by using a strategy that is similar to the one used by the IGV (Integrative Genomics Viewer) [135], where data is kept on the hard drive and only loaded into memory as needed for smooth interaction with the visualization.

To conclude, inPHAP offers visual and analytical methods to digest genotype and phased haplotype data in order to assist researchers in making well informed interpretations. However, some additional features would further improve the application in the future. In particular, an additional view showing the exact chromosomal location of a SNV would support the user in the interpretation of haplotypes. Furthermore, an interaction feature that allows for keeping specific regions in the tabular view fixed while continuing to inspect others, could improve comparability between different SNVs. In addition, an advanced strategy for loading data into memory would increase scalability of the application. Finally, inPHAP is solely applicable to SNV data. However, other variations might also be of interest to clinicians or researches, such as copy number changes. Thus, the design choices introduced with inPHAP could be reused to build a visual analytical application for the assessment of copy number variations.

9.3 Visual Analytics for SNV Associated Gene Expression Changes

The analysis of variations, especially SNVs as the most common representative, provides valuable insight into the genetic factors leading to disease. For complex diseases, where not a single variation, but the interaction of many different ones are possibly involved in a disease, the identification and interpretation of the effect of these SNVs on the phenotype is usually very

difficult. Moreover, SNVs can influence genes on different levels. Depending on their location in the genome, they can either alter a gene's expression or lead to a complete loss of gene function. Furthermore, the analysis of gene expression changes allows for the interpretation of phenotypic outcomes and is thus important to combine with SNV data in disease related studies. Although, these so-called eQTL studies provide a much better representation of the factors leading to disease, their application is limited by either computational issues or the lack of appropriate eQTL analysis applications. While solving the computational problem, especially for the analysis of epistasis, will probably not be possible without further technical advances, the development of software solutions for the analysis of eQTL data, including single-locus and two-locus associations, is readily possible. Unfortunately, currently available software solutions do not offer a completely integrative study of SNV and gene expression data, but are usually focused strongly on either of these two data types. Furthermore, most available software solutions suffer from a lack of appropriate visualizations for the interpretation of statistical results.

In this dissertation, a new visual analytical approach has been taken with the development of REVEAL, a highly interactive and integrative software solution for the study of GWA and eQTL data. Due to the integration into the gene expression analysis software MAYDAY, REVEAL is able to perform both SNV based analyses as well as corresponding gene expression analyses equally well, and results from both can be integrated and used to provide comprehensive visualizations that assist with the interpretation of the underlying phenotypes. In this, REVEAL is the first tool that follows an integrative approach for the study of eQTL data, which is not centered on either SNV or gene expression analysis exclusively. Besides the integration of well established methods for SNV based analyses, such as commonly used statistics or filtering techniques, REVEAL offers new and innovative visualization approaches for the integrative study of SNVs and gene expression changes. In the following, the most important developments and features of REVEAL will be discussed independently.

In traditional GWAS, SNV distributions between two cohorts, typically with different clinical phenotype, are compared to identify those SNVs that show a significant difference. For this purpose, often pie charts are used. However, pie charts have the huge disadvantage, that a direct comparison of distributions is difficult. A better solution would be to compare values on a common scale [95]. This has been realized in the SNV Summary plot. There, genotype distributions can easily be compared between the two cohorts for each individual SNV, because bars of different height are used to represent genotype distributions. Furthermore, additional tracks can be added to compare these distributions against a reference allele or to compare a single individual against a cohort. The latter allows for the classification

of individuals based on selected SNVs, if the disease state is unclear or not manifested during the time of the analysis. The disadvantage of this visualization is that it does not scale well with respect to the number of SNVs that are visualized. Although, zooming techniques have been introduced, in a zoomed out view individual distribution values are hard to compare. This can be compensated to some extent with the aggregation track, which shows a summarized representation of the genotype distribution track. However, the SNV Summary plot is not meant for the investigation of a large collection of SNVs, but as a tool that provides detailed information about individual ones.

For the visualization of single- and two-locus association results, REVEAL offers three different representations, which are linked to each other to provide a comprehensive picture of the underlying data. The available data tables display the statistical data imported from, for example, a PLINK based analysis and provide a detailed representation for each individual association. However, tables are not well suited for the representation of complex interactions. Thus, two additional representations have been developed, a graph-based visualization and a matrix view. Each of these representations has specific advantages and disadvantages, which will be briefly discussed. The node-linked graph has the advantage of showing the overall complexity of associations and is well suited to display interactions between different genes. Although the node-linked graph in general is a standard tool for the representation of complex data, REVEAL's specific implementation, the Association Network, has been selected, together with the genotype based visualization available in INPHAP, as the visualization experts' favorite during the eQTL biological data visualization challenge at the BioVis conference in 2011. This especially addressed the innovative design choices made for the network-based visualization of two-locus associations, in particular the utilization of edge color and thickness to represent associations of SNV pairs with gene expression levels (see chapter 5, section 5.7.2 for more details). The disadvantage of any node-linked graph is that it can quickly become a hairball, which is a known problem in every graph-based visualization. In REVEAL, this issue has been addressed with the introduction of user interactivity, as for example interactive edge filtering to reduce visual clutter. Moreover, visual clarity largely depends on the number of genes, SNVs and the corresponding number of associations. In the Two-Locus Association Network, color values are used to link edges (SNV pairs) to nodes (genes). Clearly, a unique identification is only possible if the number of genes is rather small, optimally less than 12, which corresponds to the maximal number of different colors that can easily be distinguished [52]. For a much larger number of genes a clear separation cannot be made. Again, this issue has been addressed by the introduction of user interactivity. Nodes can be selected by the user and all edges with the same color are highlighted, which

allows for a unique identification of corresponding SNV pair associations. Although, these mechanisms compensate the disadvantages of this visual representation to some extent, displaying the underlying associations in a matrix-like visualization does not suffer from the discussed hindrances, since a clear assignment of e.g. SNVs (or SNV pairs) to rows and genes to columns can always be made. A further advantage of the association matrix is that distinct colors can be used to represent meta-information, because in the matrix the color property is not needed to link SNV pair associations to their respective gene. Nevertheless, in matrix-like visualizations it is usually harder to identify complex interactions. Thus, the network-based and the matrix-like views are recommended to be used in parallel to capture a comprehensive picture of the underlying data. Moreover, all visualizations in REVEAL are linked to each other on the SNV, gene and subject level (see chapter 5, section 5.1.3 for more details). In this, the user has the opportunity to overcome hindrances of one visualization by the application of several visualizations showing the same information, but from different perspectives. With this strategy, a comprehensive insight into the data can be obtained.

Although REVEAL offers various ways for data visualization and processing, its capabilities are limited when it comes to raw data processing and statistical evaluation. In particular, REVEAL does not offer algorithms for the calculation of single- or two-locus associations or other computationally intense statistics, such as the calculation of linkage disequilibrium (LD) correlation values. The initial purpose of REVEAL was to provide a software solution that supports users with the interpretation of their data. This is why REVEAL has been designed as an interactive desktop solution providing a graphical user interface, such that users that are not familiar with console based applications can obtain valuable insights into their data. Consequently, the software was not intended to take computationally intense calculation to the desktop pc, but to offer a platform, which is able to integrate results from more specialized applications for eQTL data analysis, in particular PLINK [129], in a common software environment.

Linkage of gene expression values and SNV data can, however, also be performed without the need for statistical association testing. In typical GWAS, box plots are used to show the correlation of allele combinations to a gene's expression level in a given population. However, for diploid organisms this requires three boxes for the possible allele combinations (homozygous reference base, heterozygous SNV, homozygous SNV). If there is an additive effect of the SNV on the gene's expression level, then this effect should be visible as an increase or decrease in the mean value of the expression level distributions between the three possible allele combinations. Although this approach nicely shows possible associations of a SNV with a gene's expression

level, it does not scale well with the number of SNVs and genes. In particular, for each possible SNV/gene combination an individual box plot visualization has to be constructed. With the data transformation approach described in chapter 5, the increase or decrease in expression is encoded as a single value. With this strategy, the most important information provided by the box plot visualization has been used to create scalable visualizations, such as a profile plot or a heat map. This offers a comprehensive representation of SNV derived expression changes within well established and easy to interpret visualizations. By additionally applying sorting or clustering techniques known from traditional gene expression analyses, further improvement of the visualizations can be achieved. Clearly, each summarization technique results in a loss of information. In this case, only information about the mean value of the expression level distribution for each allele combination is used and other parameters of the distribution are disregarded. However, if the population sizes are small, then the mean value may not be very representative. Furthermore, information about the first and third quantile of the respective distribution may also be of interest. To compensate for small population sizes and a biased mean value, also the median value can be used in the transformation. However, in cases, where a visualization of the distribution itself is needed one has to use the box plot approach, which is of course also possible in MAYDAY itself.

Since the publication of REVEAL at the beginning of 2012 [71], other software solutions have been introduced that are based on the design choices made in REVEAL. For example, single-locus gene expression association and epistasis are also addressed with the Aracari tool [136]. Aracari makes use of the visualization concepts introduced with the single- and two-locus association visualizations in REVEAL and combines these with a distribution based visualization of gene expression levels. It offers a modified version of the association matrix introduced in this work and combines it with histograms and Q-Q plots for the comparison of gene expression distributions between different populations. This demonstrates that the visualization concepts introduced in this work have already and will probably continue to influence how eQTL data are visualized nowadays and in the future.

9.4 Optimization of Structural Variation Visualization with GENOMERING

GENOMERING is a highly interactive tool developed for the visualization of structural similarities and differences between genomes based on a multiple whole genome alignment. Although the visualization of genomes in GENOMERING is largely improved by the application of the SuperGenome in comparison to existing multiple whole genome alignment visualizations, such

as Mauve, visual clutter can still not be prevented completely. Whenever there are missing blocks in one of the genomes, arcs have to be drawn to indicate a skipping event. With a rising number of structural variation events, more and more arcs have to be drawn, which eventually leads to visual clutter. The disadvantage is shared by all existing genome visualization approaches, including the two most commonly used ones, Mauve and Circos. Furthermore, existing visualizations usually keep the order of blocks for each genome fixed. Although this simplifies the identification of the individual genome compositions, it can also result in a large number of arcs needed to represent block identity. In GenomeRing, blocks can be arranged independently of the visualized genomes, since block composition is represented by genome specific paths drawn with unique color and location. In addition, the direction of a path within a block does not encode for the sequence direction in the genome. In GenomeRing a second ring is used to represent the reverse sequences with respect to the SuperGenome. This feature offers additional freedom in the placement of blocks to improve visual clarity.

In this dissertation, a quadratic time heuristic with respect to the number of blocks and genomes has been developed that finds an optimal block ordering based on a user-defined optimization criterion. Currently, three different criteria are available, namely optimizing the total number of arcs, the total number of skipped blocks, and the sum of all arc lengths. Each of these criteria can be used to improve visual clarity within a GenomeRing visualization and their specific properties are discussed in the following.

The first criterion minimizes the number of arcs that have to be drawn. This strategy is based on the assumption that visual clutter correlates with the number of arcs in the GenomeRing visualization. Thus, reducing the number of arcs would consequently lead to a better visual experience. Investigations on real data sets have shown (see chapter 6) that this strategy can provide good results for genomes, where only a few structural differences are expected, but a large number of arcs is observed due to a non-optimal ordering. This gives the impression of large structural differences and can lead to false interpretation without proper optimization. However, for such scenarios the optimization of the total number of arcs is beneficial.

The second optimization approach focuses on the number of skipped blocks rather than the number of arcs in the visualization. This method assumes that arcs that span only very few blocks do not lead to visual clutter, but those spanning multiple different blocks do. Thus, minimizing the number of blocks that have to be skipped with respect to all genomes may lead to a better visual representation. In chapter 6, it could be shown that this strategy is especially useful to improve the visualization of genomic islands between

genomes.

If the intention of the user is to get a first overview of the structural variations, and if there is no general interest in specific genomic structures, then a global optimization of the whole GenomeRing visualization is beneficial. In such cases, an optimal representation would only allow for short arcs if any. This is achieved with the third optimization criterion, which minimizes the total arc length. The disadvantage is that it does not improve visualization of specific structures, but tries to improve visualization of the whole multiple genome alignment. It is thus best suited to gain a first overview of the similarities and differences between the aligned genomes.

Besides these automatic optimizations, the user can also switch blocks manually or reorder blocks based on the natural ordering of blocks in one of the genomes. This provides additional freedom for the comparison of structural differences. Clearly, not every optimization method is well suited for each data set. Consequently, if an appropriate optimization can be obtained largely depends on the complexity of the underlying multiple whole genome alignment, the question that should be solved and the parameters used to calculate the respective SuperGenome.

Nevertheless, the described layout optimization strategies offer the possibility to assess different aspects of the data, which are then visually more appealing. Thus, the ordering methods described in this work are of great importance to draw well informed conclusions from multiple whole genome alignments with GenomeRing.

Although all of the block order optimization methods can largely improve the user's experience with the software, GenomeRing has not been designed for the visualization of hundreds of genomes, blocks or events. The complexity of a multiple whole genome alignment rises with each additional genome that is included. Thus, when reaching a specific number of structural events, it would not be possible to further improve visual clarity by changing the block order. Moreover, for a large number of genomes, the question arises whether the display of detailed information for a whole genome alignment is still feasible in order to draw meaningful conclusions, or if an appropriate summarization focusing only on the important events would be satisfactory in such situations. Future work on GenomeRing could therefore include the exploration of summarization methods for structural variation detection and visualization.

To conclude, GenomeRing focuses on the visualization of similarities and dissimilarities between aligned genomes. It is thus a valuable tool that com-

plements other applications, such as conventional genome browsers, in order to provide a more comprehensive representation of structural as well as single nucleotide variations. With its innovative design together with the application of the SuperGenome concept, GenomeRing was able to win the Illumina Challenge Award 2011 in the category Most Creative Algorithm.

9.5 Automated Analysis of NGS Data from Ancient and Modern DNA Samples

In contrast to the analysis of modern DNA of good quality, the analysis of very old DNA is more complicated. DNA gets degraded over time, which poses challenges to the DNA processing in the lab and also to the data analysis afterwards. A major problem is the fragmentation of DNA and its consequences for the application of next-generation sequencing for DNA identification.

In this dissertation, a pipeline has been developed together with Alexander Herbig that deals with sequencing data from DNA of ancient as well as modern origin. The crucial step in this pipeline is the preprocessing of the sequenced reads, in order to provide accurate and sensitive mapping results. Due to the fragmentation of DNA over time, paired-end sequencing results in overlapping read pairs, which can be used to improve the overall quality of the reads. The per base quality of a read tends to decrease from the 5' towards the 3' end. This is even worse with DNA of ancient origin, where drastic quality drops at the 3' end are observed frequently. A common approach to address this issue is merging of overlapping read pairs. For this purpose various different methods have been introduced over the last years. However, none of the existing methods was satisfying with respect to quality (measured in mapping rates after merging) and runtime (measured in the overall time needed to cut remaining adapter sequences and merge overlapping reads afterwards). Thus, a new tool, called ClipAndMerge, has been developed and compared to those tools that were most commonly used at the time of writing. In chapter 7, a detailed comparison of the performance with the different applications was made. This evaluation revealed that MergeReadsFastQ outperforms all other applications on most of the tested data sets with respect to merging and mapping rates. However, due to the extremely large runtime requirement, it is unfeasible to use this tool in any larger sequencing project. In particular, whole genome sequencing projects or projects including multiple different species cannot be performed within acceptable time. With ClipAndMerge, a very fast adapter clipping and read merging tool has been introduced that outperforms existing methods either in mapping quality or in time and that can compete with the merging and mapping rates of the MergeReadsFastQ

application. In particular, none of the other methods was able to produce significantly better results when taking merging and mapping rates into account. In cases where merging and mapping rates were comparable, ClipAndMerge usually took much less time for the processing. Furthermore, ClipAndMerge provides additional features in comparison to the other applications, such as quality trimming, adapter clipping for single-end reads, or the automatic concatenation of `FASTQ` files resulting from multiple sequencing lanes. This makes ClipAndMerge widely applicable to various different sequencing projects and thus the best choice for the described data processing pipeline.

The power of the read processing steps within the described pipeline was already successfully demonstrated in a high impact paper in Science [144], for which I served as a co-author. In this thesis, I concentrated on showing how the pipeline performs for special types of modern data, where similar quality issues arise as with ancient data. The application to modern *Treponema pallidum* strains showed that high mapping rates could be achieved with the application of the ClipAndMerge tool within the processing pipeline. Furthermore, application of the whole pipeline enabled the characterization and phylogenetic classification of *Treponema pallidum* strains from all over the world. One disadvantage that was observed during the analysis of different *Treponema pallidum* strains is that genomic regions with a large number of variations, so-called hypervariable regions, lead to coverage drops during mapping when strictly following the steps of the pipeline. In this thesis, this hindrance was overcome by switching to a more appropriate mapping algorithm with relaxed mapping parameters regarding read alignment specificity for these regions. Although this approach worked well for the *Treponema pallidum* strains studied in this work, it remains to be investigated how such an approach would perform with more complex organisms, where in addition repetitive sequences hinder the analysis process. Furthermore, automation of this strategy could be included in the pipeline in the future to increase applicability for complex, highly variable regions.

Based on this pipeline a step further has already been taken by Alexander Peltzer with the introduction of EAGER (Efficient Algorithms for Genome Reconstruction, manuscript under revision in Genome Biology). The development of EAGER was highly motivated by the successful preliminary work described in this thesis. Thus, the methods included in this pipeline are reused in EAGER. Moreover, EAGER has been designed to additional provide methods for the analysis of ancient population-based data in the future. To conclude, the steps taken during this dissertation already have greatly improved and will continue to improve how genomic sequencing data from ancient and modern DNA samples are processed.

9.6 A Microarray Based Ancient DNA Screening Technique for Human Pathogens

By the development of the ancient sequencing data processing pipeline discussed in the last section, the comparative analysis of ancient and modern DNA samples has been simplified and automated. However, a large issue still remains, namely the low amount of ancient DNA that is usually available in a corresponding sample. Especially when studying ancient microorganisms the amount of DNA that can be obtained is much lower than for mammals or any other larger organisms. The main reason for that is that samples containing ancient DNA, as for example soil, water or other biological material, are usually contaminated with DNA from other modern microorganisms living under such conditions. In comparison to the amount of DNA from these modern microorganisms, the amount of ancient DNA is usually much less, rendering it difficult to gather enough DNA for an appropriate identification and the subsequent analysis of the organism of interest. To overcome this hindrance, approaches have been taken to enrich for the DNA of interest. For this purpose, DNA capture microarrays have been commonly applied. However, by the time of this thesis only single organism specific arrays have been available. These have the huge disadvantage of only being able to screen for a single organism at once and are therefore not suited for applications, where the specific organism is unknown.

In this dissertation, a microarray based DNA capture and enrichment approach, named APSA (Ancient Pathogens Screening Array), has been developed that is able to identify almost 100 different human pathogens with a single microarray. Using a multi organism array is very economical, but clearly the captured DNA content differs significantly in comparison to a single organism capture array. For the development of APSA, it was necessary to identify unique regions in each of the organisms, which were then used for probe selection. For a typical whole genome single organism array a genome tiling approach with e.g. a 6 base pair tiling for 60 base pairs long oligos would suffice to evaluate the presence or absence of an organism in a biological sample. However, for the development of APSA cross-hybridization with other organisms had to be avoided to assure that the captured DNA provides a specific signal. To achieve this, a phylogenetically based approach was chosen, where for each pathogen added to the array regions were selected to be unique with respect to a specific taxonomic level. Although, identification of the respective taxonomic level for each pathogen was possible by using the NCBI Taxonomy Database, the problem of choosing the right genome sequence for oligo design remained. In this work, one of the possible genomes for a pathogen was chosen randomly, which strongly relies on the assumption

that sub-sequences that are unique on a specific taxonomic level are shared by all strains of that species underneath this taxonomic level. Sequence identity was thereby defined by a BLAST search. The disadvantage of this approach is that with a local alignment complete sequence identity cannot be guaranteed, which leaves the risk of varying sub-sequences between summarized strains. Furthermore, some of the strains may also completely lack selected sequence regions. Although, such differences could not be avoided completely with the approach described in chapter 8, variations are not expected to be very large and the amount of resulting oligos was large enough to capture a broad range of all the organisms' genomes. Thus, selecting one possible genome sequence randomly seemed reasonable.

Another decision concerns the distribution of available oligos on the APSA. Because of the different genome sizes of the organisms that were chosen for the APSA, different numbers of usable oligo sequences for each pathogen were obtained. In particular, viruses revealed the least number of oligos, whereas for worms up to hundreds of thousands of oligos could be obtained. However, the total number of oligos on the APSA was limited, which means that oligos had to be selected carefully to provide a good coverage over each individual genome. Clearly, if the number of oligos for each pathogen differs on the APSA, then for those with a larger number it is more likely that also a larger amount of DNA gets captured, whereas pathogens with only very few oligos can only capture very few DNA fragments. To solve this issue, basically two different strategies were possible. With the first strategy an equal representation of the pathogens on the array needs to be achieved by, for example, placing copies of the same oligo multiple times for those pathogens were only a few oligos were available. With the second strategy each oligo is placed only once, which results in a different number of oligos for each pathogen. For the latter, a normalization has to be performed afterwards to account for the varying capturing potential of the APSA computationally. In this work, the decision was made to use a normalization approach rather than placing oligos multiple times. This strategy had the advantage of covering large genomes equally well in relation to smaller ones, with respect to the overall genome representation.

To evaluate the detection and enrichment capability of the APSA, it was applied to a positive control containing ancient *Mycobacterium leprae* DNA, which resulted in a 460× enrichment in comparison to a shotgun sequencing without enrichment. This demonstrates the huge enrichment and detection potential of the APSA. However, further validation in the future with varying data sets would be beneficial to evaluate the detection and enrichment potential with respect to the other pathogens represented on the APSA.

Although the design pipeline in this work was used to build oligos for microarray based DNA capturing, the general concept can readily be applied to other capturing techniques, such as in-solution capturing [97]. In contrast to a microarray-based procedure, additional magnetic beads would have to be added to the oligos for later DNA retrieval. However, this does not affect the general oligo design process, which is why the design procedure described in this thesis can be used without larger modifications to build oligos for in-solution capturing. Furthermore, the analysis toolkit could also be used to analyze in-solution captured DNA fragments without the necessity of modifications.

To conclude, the APSA is the first and only array based enrichment technique that allows for the detection and enrichment of up to 100 different human pathogens in parallel. Although, its main intention was the identification of ancient DNA material, it is not limited to ancient DNA and can also be applied for the identification of modern pathogenic DNA. Together with an appropriate analysis toolkit, that has been developed specifically for this array and the analysis pipeline described in chapter 7, an easy to use, automated analysis of APSA captured sequencing reads has been made possible.

9.7 Conclusion

With the development of the next-generation sequencing technologies new possibilities to understand the complexity of biological data have been provided. However, with the rising amount of data that could be generated, appropriate methods were needed to process the data and comprehensive visualization were required to help with their interpretation. In this dissertation several new data processing and visualization approaches have been presented, including the analysis of genotype and haplotype data in the context of an eQTL study, the visual assessment of structural variations as well as methods for processing very old DNA that satisfy specific requirements in order to obtain a comprehensive set of SNVs for further analysis.

In the future the methods and visualizations presented in this dissertation will continue to support researchers and clinicians with the analysis and interpretation of SNVs. As has been shown above, some of the design concepts already had great impact on how SNV data is processed and visualized and will probably continue to inspire researches in the process of creating new visualization and data processing concepts. Furthermore, this work highly focused on the integration of various data types, in particular genotype/haplotype data, gene expression data, as well as structural variation data, into a single study. As technologies advance, more and more integrative approaches will be needed

that unite available data of different data types. Only then a complete picture of e.g. a specific disease or condition of interest can be obtained. The design choices made in this dissertation pave the way to a fully integrative study of variation and expression data, offering the opportunity to come to a better understanding of the complexity of life and disease.

Bibliography

[1] A. Acland, R. Agarwala, T. Barrett, J. Beck, et al. "Database resources of the National Center for Biotechnology Information." In: *Nucleic Acids Res* 42 (2014), pages D7–17. DOI: 10.1093/nar/gkt1146.

[2] A. Agresti. *An Introduction to Categorical Data Analysis*. 2nd edition. John Wiley & Sons, Inc., 2007. DOI: 10.1002/0471249688.

[3] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman. "Basic local alignment search tool." In: *J Mol Biol* 215.3 (1990), pages 403–410. DOI: 10.1016/S0022-2836(05)80360-2.

[4] S. Andrews. *FastQC - A quality control tool for high throughput sequence data*. 2014. URL: http://www.bioinformatics.babraham.ac.uk/projects/fastqc/.

[5] F. J. Anscombe. "Graphs in Statistical Analysis". In: *The American Statistician* 27.1 (1973), pages 17–21.

[6] P. Armitage. "Tests for linear trends in proportions and frequencies." In: *Biometrics* 11.3 (1955), pages 375–386. DOI: 10.2307/3001775.

[7] J. C. Barrett, B. Fry, J. Maller, and M. J. Daly. "Haploview: analysis and visualization of LD and haplotype maps." In: *Bioinformatics* 21.2 (2005), pages 263–265. DOI: 10.1093/bioinformatics/bth457.

[8] C. W. Bartlett, S. Y. Cheong, L. Hou, J. Paquette, et al. "An eQTL biological data visualization challenge and approaches from the visualization community". In: *BMC Bioinformatics* 13 Suppl 8.Suppl 8 (2012), S8. DOI: 10.1186/1471-2105-13-S8-S8.

[9] F. Battke. "Computational Methods for High-Throughput Transcriptomics Data". PhD Thesis. University of Tübingen, 2012.

[10] F. Battke and K. Nieselt. "Mayday SeaSight: Combined analysis of deep sequencing and microarray data". In: *PLoS ONE* 6.1 (2011), e16345. DOI: 10.1371/journal.pone.0016345.

[11] F. Battke, S. Symons, and K. Nieselt. "Mayday–integrative analytics for expression data." In: *BMC Bioinformatics* 11 (2010), page 121. DOI: 10.1186/1471-2105-11-121.

[12] Y. Benjamini and Y. Hochberg. "Controlling the false discovery rate: a practical and powerful approach to multiple testing." In: *Journal of the Royal Statistical Society B* 57.1 (1995), pages 289–300.

[13] D. A. Benson, M. Cavanaugh, K. Clark, I. Karsch-Mizrachi, D. J. Lipman, J. Ostell, and E. W. Sayers. "GenBank". In: *Nucleic Acids Research* 41 (2013). DOI: 10.1093/nar/gks1195.

[14] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov, and P. E. Bourne. "The Protein Data Bank." In: *Nucleic Acids Research* 28.1 (2000), pages 235–242. DOI: 10.1093/nar/28.1.235.

[15] S. E. Bojesen, K. A. Pooley, S. E. Johnatty, J. Beesley, et al. "Multiple independent variants at the TERT locus are associated with telomere length and risks of breast and ovarian cancer." In: *Nature Genetics* 45.4 (2013), 371–84, 384e1–2. DOI: 10.1038/ng.2566.

[16] B. M. Bolstad, R. A. Irizarry, M. Åstrand, and T. P. Speed. "A comparison of normalization methods for high density oligonucleotide array data based on variance and bias." In: *Bioinformatics* 19.2 (2003), pages 185–193. DOI: 10.1093/bioinformatics/19.2.185.

[17] K. I. Bos, G. Jäger, V. J. Schuenemann, Â. Vågene, M. A. Spyrou, A. Herbig, K. Nieselt, and J. Krause. "Parallel detection of ancient pathogens via array-based DNA capture." In: *Royal Society* 370.1660 (2014). DOI: 10.1098/rstb.2013.0375.

[18] K. I. Bos, V. J. Schuenemann, G. B. Golding, H. A. Burbano, et al. "A draft genome of Yersinia pestis from victims of the Black Death." In: *Nature* 478.7370 (2011), pages 506–510. DOI: 10.1038/nature10549.

[19] K. W. Broman and S. Sen. *A Guide to ATL Mapping with R/qtl.* Springer, 2009. ISBN: 978-0-387-92125-9.

[20] B. L. Browning and S. R. Browning. "A fast, powerful method for detecting identity by descent." In: *American Journal of Human Genetics* 88.2 (2011), pages 173–182. DOI: 10.1016/j.ajhg.2011.01.010.

[21] S. R. Browning and B. L. Browning. "High-Resolution Detection of Identity by Descent in Unrelated Individuals." In: *American Journal of Human Genetics* 86.4 (2010), pages 526–539. DOI: 10.1016/j.ajhg.2010.02.021.

[22] J. C. Chambers, W. Zhang, G. M. Lord, P. van der Harst, et al. "Genetic loci influencing kidney function and chronic kidney disease." In: *Nature Genetics* 42 (2010), pages 373–375. DOI: 10.1038/ng.566.

[23] P. Chandra, M. Messier, and J. Viega. *Network security with OpenSSL.* O'Reilly Media, 2002. ISBN: 978-0-596-00270-1.

[24] C. Y. Chen, K. H. Chi, A. Pillay, E. Nachamkin, J. R. Su, and R. C. Ballard. "Detection of the A2058G and A2059G 23S rRNA gene point mutations associated with azithromycin resistance in Treponema pallidum by use of a TaqMan real-time multiplex PCR assay." In: *Journal of Clinical Microbiology* 51.3 (2013), pages 908–913. DOI: 10.1128/JCM.02770-12.

[25] P. Cingolani, A. Platts, L. L. Wang, M. Coon, et al. "A program for annotating and predicting the effects of single nucleotide polymorphisms, SnpEff: SNPs in the genome of Drosophila melanogaster strain w 1118; iso-2; iso-3." In: *Fly* 6.2 (2012), pages 80–92. DOI: 10.4161/fly.19695.

[26] W. S. Cleveland and R. McGill. "Graphical Perception: Theory, Experimentation, and Application to the Development of Graphical Methods." In: *Journal of the American Statistical Association* 79.387 (1984), pages 531–554. DOI: 10.2307/2288400.

[27] The 1000 Genomes Project Consortium. "A map of human genome variation from population-scale sequencing." In: *Nature* 467 (2010), pages 1061–1073. DOI: 10.1038/nature09534.

[28] The 1000 Genomes Project Consortium, G. R. Abecasis, A. Auton, L. D. Brooks, et al. "An integrated map of genetic variation from 1,092 human genomes." In: *Nature* 491.7422 (2012), pages 56–65. DOI: 10.1038/nature11632.

[29] The International HapMap Consortium, R. A. Gibbs, J. W. Belmont, P. Hardenbol, et al. "The International HapMap Project." In: *Nature* 426.6968 (2003), pages 789–796. DOI: 10.1038/nature02168.

[30] F. Crick. "Central dogma of molecular biology." In: *Nature* 227.5258 (1970), pages 561–563. DOI: 10.1038/227561a0.

[31] R. E. Curtis, P. Kinnaird, and E. P. Xing. "GenAMap: Visualization strategies for structured association mapping." In: *IEEE Symposium on Biological Data Visualization 2011, BioVis 2011 - Proceedings*. 2011, pages 87–94. DOI: 10.1109/BioVis.2011.6094052.

[32] P. Danecek, A. Auton, G. Abecasis, C. A. Albers, et al. "The variant call format and VCFtools." In: *Bioinformatics* 27.15 (2011), pages 2156–2158. DOI: 10.1093/bioinformatics/btr330.

[33] A. C. E. Darling, B. Mau, F. R. Blattner, and N. T. Perna. "Mauve: Multiple alignment of conserved genomic sequence with rearrangements." In: *Genome Research* 14.7 (2004), pages 1394–1403. DOI: 10.1101/gr.2289704.

[34] O. Delaneau, J. Marchini, and J. Zagury. "A linear complexity phasing method for thousands of genomes." In: *Nature Methods* 9 (2011), pages 179–181. DOI: 10.1038/nmeth.1785.

[35] O. Delaneau, J. Zagury, and J. Marchini. "Improved whole-chromosome phasing for disease and population genetic studies." In: *Nature Methods* 10.1 (2013), pages 5–6. DOI: 10.1038/nmeth.2307.

[36] B. Devlin and N. Risch. "A comparison of linkage disequilibrium measures for fine-scale mapping." In: *Genomics* 29.2 (1995), pages 311–322. DOI: 10.1159/000154430.

[37] A. J. Drummond, M. A. Suchard, D. Xie, and A. Rambaut. "Bayesian phylogenetics with BEAUti and the BEAST 1.7." In: *Molecular Biology and Evolution* 29.8 (2012), pages 1969–1973. DOI: 10.1093/molbev/mss075.

[38] J. T. Dudley and K. J. Karczewski. *Exploring Personal Genomics.* 1st edition. Oxford University Press, 2013.

[39] O. J. Dunn. "Estimation of the Medians for Dependent Variables." In: *The Annals of Mathematical Statistics* 30.1 (1959), pages 192–197.

[40] R. A. Eeles, A. A. A. Olama, S. Benlloch, E. J. Saunders, et al. "Identification of 23 new prostate cancer susceptibility loci using the iCOGS custom genotyping array." In: *Nature Genetics* 45.4 (2013), 385–91, 391e1–2. DOI: 10.1038/ng.2560.

[41] L. Fejerman, N. Ahmadiyeh, D. Hu, S. Huntsman, et al. "Genome-wide association study of breast cancer in Latinas identifies novel protective variants on 6q25." In: *Nature Communications* 5.5260 (2014). DOI: doi:10.1038/ncomms6260.

[42] R. A. Fisher. "On the interpretation of $\chi 2$ from contingency tables, and the calculation of P." In: *Journal of the Royal Statistical Society* 85.1 (1922), pages 87–94. DOI: 10.2307/2340521.

[43] M. Fiume, E. J. M. Smith, A. Brook, D. Strbenac, et al. "Savant Genome Browser 2: Visualization and analysis for population-scale genomics." In: *Nucleic Acids Research* 40.W1 (2012), W615–W621. DOI: 10.1093/nar/gks427.

[44] D. E. Fouts, E. F. Mongodin, R. E. Mandrell, W. G. Miller, et al. "Major structural differences and novel potential virulence mechanisms from the genomes of multiple campylobacter species." In: *PLoS Biology* 3.1 (2005), e15. DOI: 10.1371/journal.pbio.0030015.

[45] C. M. Fraser, S. J. Norris, G. M. Weinstock, O. White, et al. "Complete genome sequence of Treponema pallidum, the syphilis spirochete." In: *Science* 281.5375 (1998), pages 375–388. DOI: 10.1126/science.281.5375.375.

[46] B. J. Fry. "Computational Information Design". PhD Thesis. Massachusetts Institute of Technology, 2004. DOI: 10.1111/j.1468-3083.2010.03837.x.

[47] B. J. Fry. *Visualizing Data: Exploring and Explaining Data with the Processing Environment.* 1st edition. O'Reilly Media, Inc., 2007. ISBN: 978-0-596-51455-6.

[48] M. Garcia-Closas, F. J. Couch, S. Lindstrom, K. Michailidou, et al. "Genome-wide association studies identify four ER negative-specific breast cancer risk loci." In: *Nature Genetics* 45.4 (2013), 392–8, 398e1–2. DOI: 10.1038/ng.2561.

[49] G. Gibson. "Hints of hidden heritability in GWAS." In: *Nature Genetics* 42.7 (2010), pages 558–560. DOI: 10.1038/ng0710-558.

[50] A. H. Goodman. "Toward Genetics in an Era of Anthropology." In: *American Ethnologist* 34.2 (2007), pages 227–229. DOI: 10.1525/ae.2007.34.2.227.

[51] A. Grada and K. Weinbrecht. "Next Generation Sequencing: Methodology and Application." In: *Journal of Investigative Dermatology* 133.8 (2013), e11. DOI: 10.1038/jid.2013.248.

[52] M. A. Harrower and C. A. Brewer. "ColorBrewer.org: An Online Tool for Selecting Color Schemes for Maps." In: *The Cartographic Journal* 40.1 (2003), pages 27–37.

[53] C. A. Heid, J. Stevens, K. J. Livak, and P. M. Williams. "Real time quantitative PCR." In: *Genome Research* 6.10 (1996), pages 986–994. DOI: 10.1101/gr.6.10.986.

[54] J. Heinrich, C. Vehlow, F. Battke, G. Jäger, D. Weiskopf, and K. Nieselt. "iHAT: interactive Hierarchical Aggregation Table for Genetic Association Data." In: *BMC Bioinformatics* 13.Suppl 8 (2012), S2. DOI: 10.1186/1471-2105-13-S8-S2.

[55] Golden Helix. *SNP and Variation Suite (SVS 7).* URL: http://www.goldenhelix.com (visited on 05/30/2015).

[56] M. J. Heller. "DNA microarray technology: devices, systems, and applications." In: *Annual Review of Biomedical Engineering* 4 (2002), pages 129–153. DOI: 10.1146/annurev.bioeng.4.020702.153438.

[57] A. Herbig. "Computational Methods for the Identification and Characterization of Non-Coding RNAs in Bacteria." PhD Thesis. University of Tübingen, 2014, page 153.

[58] A. Herbig, G. Jäger, F. Battke, and K. Nieselt. "GenomeRing: alignment visualization based on SuperGenome coordinates." In: *Bioinformatics* 28.12 (2012), pages i7–15. DOI: 10.1093/bioinformatics/bts217.

[59] L. J. Heyer, S. Kruglyak, and S. Yooseph. "Exploring expression data identification and analysis of coexpressed genes." In: *Genome Research* 9.11 (1999), pages 1106–1115. DOI: 10.1101/gr.9.11.1106.

[60] J. N. Hirschhorn, K. Lohmueller, E. Byrne, and K. Hirschhorn. "A comprehensive review of genetic association studies." In: *Genetics in Medicine* 4 (2002), pages 45–61. DOI: 10.1097/00125817-200203000-00002.

[61] E. L. Ho and S. A. Lukehart. "Syphilis: Using modern approaches to understand an old disease." In: *Journal of Clinical Investigation* 121.12 (2011), pages 4584–4592. DOI: 10.1172/JCI57173.

[62] D. Holder, R. Raubertas, V. Pikounis, and V. Svetnik. *Statistical analysis of high density oligonucleotide arrays: a SAFER approach.* GeneLogic Workshop on Low Level Analysis of Affymetrix GeneChip Data. 2001.

[63] M. Höss. "Ancient DNA." In: *Hormone Research* 43.4 (1995), pages 118–120.

[64] B. N. Howie, P. Donnelly, and J. Marchini. "A flexible and accurate genotype imputation method for the next generation of genome-wide association studies." In: *PLoS Genetics* 5.6 (2009), e1000529. DOI: 10.1371/journal.pgen.1000529.

[65] H. Hu, C. D. Huff, B. Moore, S. Flygare, M. G. Reese, and M. Yandell. "VAAST 2.0: improved variant classification and disease-gene identification using a conservation-controlled amino acid substitution matrix." In: *Genetic Epidemiology* 37.6 (2013), pages 622–634. DOI: 10.1002/gepi.21743.

[66] Illumina. *Mate-Pair Sequencing.* URL: http://www.illumina.com/technology/next-generation-sequencing/mate-pair-sequencing_assay.html (visited on 05/31/2015).

[67] National Human Genome Research Institute. *The Human Genome Project Completion.* 2010. URL: https://www.genome.gov/11006943 (visited on 07/21/2015).

[68] R. A. Irizarry, B. Hobbs, F. Collin, Y. D. Beazer-Barclay, K. J. Antonellis, U. Scherf, and T. P. Speed. "Exploration, normalization, and summaries of high density oligonucleotide array probe level data." In: *Biostatistics* 4.2 (2003), pages 249–264. DOI: 10.1093/biostatistics/4.2.249.

[69] F. Jacob and J. Monod. "Genetic regulatory mechanisms in the synthesis of proteins." In: *Journal of Molecular Biology* 3 (1961), pages 318–356. DOI: 10.1016/S0022-2836(61)80072-7.

[70] G. Jäger. "QT Clustering für Mayday." Bachelor Thesis. University of Tübingen, 2008.

[71] G. Jäger, F. Battke, and K. Nieselt. "REVEAL–visual eQTL analytics." In: *Bioinformatics* 28.18 (2012), pages i542–i548. DOI: 10.1093/bioinformatics/bts382.

[72] G. Jäger, A. Peltzer, and K. Nieselt. "inPHAP: interactive visualization of genotype and phased haplotype data." In: *BMC Bioinformatics* 15 (2014), page 200. DOI: 10.1186/1471-2105-15-200.

[73] M. Jallow, Y. Y. Teo, K. S. Small, K. A. Rockett, et al. "Genome-wide and fine-resolution association analysis of malaria in West Africa." In: *Nature Genetics* 41.6 (2009), pages 657–665. DOI: 10.1038/ng.388.

[74] H. Jónsson, A. Ginolhac, M. Schubert, P. L. F. Johnson, and L. Orlando. "MapDamage2.0: Fast approximate Bayesian estimates of ancient DNA damage parameters." In: *Bioinformatics* 29.13 (2013), pages 1682–1684. DOI: 10.1093/bioinformatics/btt193.

[75] C. Kanz, P. Aldebert, N. Althorpe, W. Baker, et al. "The EMBL nucleotide sequence database." In: *Nucleic Acids Research* 33 (2005), pages D27–D30. DOI: 10.1093/nar/gki098.

[76] S. Kathiresan, B. F. Voight, S. Purcell, K. Musunuru, et al. "Genome-wide association of early-onset myocardial infarction with single nucleotide polymorphisms and copy number variants." In: *Nature Genetics* 41.3 (2009), pages 334–341. DOI: 10.1038/ng.327.

[77] K. Kawaguchi, A. Bayer, and R. T. Croy. *Jenkins CI - An extendable open source continuous integration server.* 2014. URL: http://jenkins-ci.org/ (visited on 05/14/2015).

[78] D. Keim, J. Kohlhammer, G. Ellis, and F. Mansmann. *Mastering the Information Age Solving Problems with Visual Analytics.* Eurographics Association, 2010. ISBN: 978-3-905673-77-7.

[79] B. Kerem, J. M. Rommens, J. A. Buchanan, D. Markiewicz, T. K. Cox, A. Chakravarti, M. Buchwald, and L. C. Tsui. "Identification of the cystic fibrosis gene: genetic analysis." In: *Science* 245.4922 (1989), pages 1073–1080. DOI: 10.1126/science.2570460.

[80] M. Kircher. "Analysis of high-throughput ancient DNA sequencing data." In: *Methods in Molecular Biology* 840 (2012), pages 197–228. DOI: 10.1007/978-1-61779-516-9_23.

[81] L. Kistler, A. Montenegro, B. D. Smith, J. A. Gifford, R. E. Green, L. A. Newsom, and B. Shapiro. "Transoceanic drift and the domestication of African bottle gourds in the Americas." In: *Proceedings of the National Academy of Sciences of the United States of America (PNAS)* 111.8 (2014), pages 2937–2941. DOI: 10.1073/pnas.1318678111.

[82] S. Köhne and I. Pigeot. *Resampling-Based Multiple Testing. Examples and Methods for p-Value Adjustment.* Volume 20. 2. John Wiley & Sons, Inc., 1995, pages 235–236. DOI: 10.1016/0167-9473(95)90129-9.

[83] M. Krzywinski, J. Schein, I. Birol, J. Connors, R. Gascoyne, D. Horsman, S. J. Jones, and M. A. Marra. "Circos: An information aesthetic for comparative genomics." In: *Genome Research* 19.9 (2009), pages 1639–1645. DOI: 10.1101/gr.092759.109.

[84] Hannon Lab. *FASTX-Toolkit - FASTQ/A short-reads pre-processing tools.* 2014. URL: http://cancan.cshl.edu/labmembers/gordon/fastx_toolkit/index.html (visited on 06/19/2015).

[85] E. S. Lander and S. Botstein. "Mapping mendelian factors underlying quantitative traits using RFLP linkage maps." In: *Genetics* 121.1 (1989), pages 185–199.

[86] E. S. Lander, L. M. Linton, B. Birren, C. Nusbaum, et al. "Initial sequencing and analysis of the human genome." In: *Nature* 409.6822 (2001), pages 860–921. DOI: 10.1038/35057062.

[87] S. Lemoine, F. Combes, and S. Le Crom. "An evaluation of custom microarray applications: The oligonucleotide design challenge." In: *Nucleic Acids Research* 37.6 (2009), pages 1726–1739. DOI: 10.1093/nar/gkp053.

[88] C. M. Lewis. "Genetic association studies: design, analysis and interpretation." In: *Briefings in Bioinformatics* 3.2 (2002), pages 146–153. DOI: 10.1093/bib/3.2.146.

[89] C. M. Lewis and J. Knight. "Introduction to genetic association studies." In: *Cold Spring Harbor Protocols* 7.3 (2012), pages 297–306. DOI: 10.1101/pdb.top068163.

[90] H. Li and R. Durbin. "Fast and accurate short read alignment with Burrows-Wheeler transform." In: *Bioinformatics* 25.14 (2009), pages 1754–1760. DOI: 10.1093/bioinformatics/btp324.

[91] H. Li, B. Handsaker, A. Wysoker, T. Fennell, et al. "The Sequence Alignment/Map format and SAMtools." In: *Bioinformatics* 25.16 (2009), pages 2078–2079. DOI: 10.1093/bioinformatics/btp352.

[92] S. Lindgreen. "AdapterRemoval: easy cleaning of next-generation sequencing reads." In: *BMC Research Notes* 5 (2012), page 337. DOI: 10.1186/1756-0500-5-337.

[93] N.C. Lovell. *Paleopathological description and diagnosis.* Edited by M. A. Katzenberg and S. R. Saunders. Wiley-Liss, Inc.: Biological Anthropology of the Human Skeleton, 2000.

[94] R. Luo, B. Liu, Y. Xie, Z. Li, et al. "SOAPdenovo2: an empirically improved memory-efficient short-read de novo assembler." In: *GigaScience* 1.1 (2012), page 18. DOI: 10.1186/2047-217X-1-18.

[95] J. Mackinlay. "Automating the design of graphical presentations of relational information." In: *ACM Transactions on Graphics (TOG)* 5.2 (1986), pages 110–141. DOI: 10.1145/22949.22950.

[96] T. Magoč and S. L. Salzberg. "FLASH: Fast length adjustment of short reads to improve genome assemblies". In: *Bioinformatics* 27.21 (2011), pages 2957–2963. DOI: 10.1093/bioinformatics/btr507.

[97] L. Mamanova, A. J. Coffey, C. E. Scott, I. Kozarewa, et al. "Target-enrichment strategies for next-generation sequencing." In: *Nature Methods* 7.2 (2010), pages 111–118. DOI: 10.1038/nmeth.1419.

[98] T. A. Manolio. "Genomewide association studies and assessment of the risk of disease." In: *The New England Journal of Medicine* 363.2 (2010), pages 166–176. DOI: 10.1056/NEJMra0905980.

[99] M. Margulies, M. Egholm, W. E. Altman, S. Attiya, et al. "Genome sequencing in microfabricated high-density picolitre reactors." In: *Nature* 437.7057 (2005), pages 376–380. DOI: 10.1038/nature03959.

[100] M. Martin. "Cutadapt removes adapter sequences from high-throughput sequencing reads." In: *EMBnet.journal* 17.1 (2011), page 10. DOI: 10.14806/ej.17.1.200.

[101] O. Martin, A. Valsesia, A. Telenti, I. Xenarios, and B. J. Stevenson. "Association Viewer: a scalable and integrated software tool for visualization of large-scale variation data in genomic context." In: *Bioinformatics* 25.5 (2009), pages 662–663. DOI: 10.1093/bioinformatics/btp017.

[102] M. L. Metzker. "Sequencing technologies - the next generation." In: *Nature Reviews Genetics* 11.1 (2010), pages 31–46. DOI: 10.1038/nrg2626.

[103] K. Michailidou, P. Hall, A. Gonzalez-Neira, M. Ghoussaini, et al. "Large-scale genotyping identifies 41 new loci associated with breast cancer risk." In: *Nature Genetics* 45.4 (2013), 353–361, 361e1–2. DOI: 10.1038/ng.2563.

[104] I. Milne, P. Shaw, G. Stephen, M. Bayer, L. Cardle, W. T. B. Thomas, A. J. Flavell, and D. Marshall. "Flapjack - Graphical Genotype Visualization." In: *Bioinformatics* 26.4 (2010), pages 3133–3134. DOI: 10.1093/bioinformatics/btq580.

[105] S. Miyazaki, H. Sugawara, K. Ikeo, T. Gojobori, and Y. Tateno. "DDBJ in the stream of various biological data." In: *Nucleic Acids Research* 32 (2004), pages D31–D34. DOI: 10.1093/nar/gkh127.

[106] S. B. Montgomery, O. L. Griffith, M. C. Sleumer, C. M. Bergman, et al. "ORegAnno: An open access database and curation system for literature-derived promoters, transcription factor binding sites and regulatory variation." In: *Bioinformatics* 22.5 (2006), pages 637–640. DOI: 10.1093/bioinformatics/btk027.

[107] J. H. Moore and M. D. Ritchie. "The challenge of whole-genome approaches to common diseases." In: *JAMA : the journal of the American Medical Association* 291.13 (2004), pages 1642–1643. DOI: 10.1001/jama.291.13.1642.

[108] M. Mueller, A. Goel, M. Thimma, N. J. Dickens, T. J. Aitman, and J. Mangion. "eQTL Explorer: Integrated mining of combined genetic linkage and expression experiments." In: *Bioinformatics* 22.4 (2006), pages 509–511. DOI: 10.1093/bioinformatics/btk007.

[109] T. Munzner. *Visualization Analysis and Design.* A K Peters/CRC Press, 2014. ISBN: 978-1466508910.

[110] S. Nagel. "Parallelisierung des QT-Clusterings in Mayday." Bachelor Thesis. University of Tübingen, 2012.

[111] L. Nechvátala, H. Pětrošováa, L. Grillováa, P. Pospíšilováa, et al. "Syphilis-causing strains belong to separate SS14-like or Nichols-like groups as defined by multilocus analysis of 19 Treponema pallidum strains." In: *Internation Journal of Medical Microbiology* 304.5-6 (2014), pages 645–653. DOI: 10.1016/j.ijmm.2014.04.007.

[112] J. B. A. Okello, L. Rodriguez, D. Poinar, K. Bos, et al. "Quantitative assessment of the sensitivity of various commercial reverse transcriptases based on armored HIV RNA." In: *PLoS ONE* 5.11 (2010), e13931. DOI: 10.1371/journal.pone.0013931.

[113] G. Olson. *"Newick's 8:45" Tree Format Standard.* 1990. URL: http://evolution.genetics.washington.edu/phylip/newick_doc.html (visited on 01/30/2015).

[114] J. O'Madadhain, D. Fisher, P. Smyth, S. White, and Y. Boey. "Analysis and Visualization of Network Data using JUNG". In: *Journal of Statistical Software* 10.2 (2005), pages 1–35.

[115] Oracle. *Hudson - Extensible continuous integration server.* 2010. URL: http://hudson-ci.org/ (visited on 05/15/2015).

[116] World Health Organization. *Global incidence and prevalence of selected curable sexually transmitted infections - 2008.* Edited by Deptartment of Reproductive Health and Research. 2012. ISBN: 9789241503839.

[117] S. Pääbo, H. Poinar, D. Serre, V. Jaenicke-Despres, et al. "Genetic analyses from ancient DNA." In: *Annual Review of Genetics* 38 (2004), pages 645–679. DOI: 10.1146/annurev.genet.37.110801.143214.

[118] C. H. Papadimitriou. "The Euclidean travelling salesman problem is NP-complete." In: *Theoretical Computer Science* 4.1977 (1977), pages 237–244. DOI: 10.1016/0304-3975(77)90012-3.

[119] C. T. Parker, B. Quiñones, W. G. Miller, S. T. Horn, and R. E. Mandrell. "Comparative genomic analysis of Campylobacter jejuni strains reveals diversity due to genomic elements similar to those present in C. jejuni strain RM1221." In: *Journal of Clinical Microbiology* 44.11 (2006), pages 4125–4135. DOI: 10.1128/JCM.01231-06.

[120] R. K. Patel and M. Jain. "NGS QC toolkit: A toolkit for quality control of next generation sequencing data." In: *PLoS ONE* 7.2 (2012), e30619. DOI: 10.1371/journal.pone.0030619.

[121] K. Pearson. "On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling." In: *Philosophical Magazine* 50.302 (1900), pages 157–175.

[122] T. A. Pearson and T. A. Manolio. "How to interpret a genome-wide association study." In: *JAMA : the journal of the American Medical Association* 299.11 (2008), pages 1335–1344. DOI: 10.1001/jama.299.11.1335.

[123] Alexander Peltzer. "Efficient algorithms for Ancient Human Genome Reconstruction". Master Thesis. University of Tübingen, 2013.

[124] J. Peterson, S. Garges, M. Giovanni, P. McInnes, et al. "The NIH Human Microbiome Project." In: *Genome Research* 19.12 (2009), pages 2317–2323. DOI: 10.1101/gr.096651.109.

[125] P. D. P. Pharoah, Y.-Y. Tsai, S. J. Ramus, C. M. Phelan, et al. "GWAS meta-analysis and replication identifies three new susceptibility loci for ovarian cancer." In: *Nature Genetics* 45.4 (2013), 362–370, 370e1–2. DOI: 10.1038/ng.2564.

[126] P. Poeppel, M. Habetha, A. Marcão, H. Büssow, L. Berna, and V. Gieselmann. "Missense mutations as a cause of metachromatic leukodystrophy: Degradation of arylsulfatase A in the endoplasmic reticulum". In: *FEBS Journal* 272.5 (2005), pages 1179–1188. DOI: 10.1111/j.1742-4658.2005.04553.x.

[127] K. Prüfer, F. Racimo, N. Patterson, F. Jay, et al. "The complete genome sequence of a Neanderthal from the Altai Mountains." In: *Nature* 505.7481 (2014), pages 43–49. DOI: 10.1038/nature12886.

[128] K. D. Pruitt, G. R. Brown, S. M. Hiatt, F. Thibaud-Nissen, et al. "RefSeq: an update on mammalian reference sequences." In: *Nucleic Acids Research* 42 (2014), pages D756–D763.

[129] S. Purcell, B. Neale, K. Todd-Brown, L. Thomas, et al. "PLINK: a tool set for whole-genome association and population-based linkage analyses." In: *American Journal of Human Genetics* 81.3 (2007), pages 559–575. DOI: 10.1086/519795.

[130] J. Quackenbush. "Microarray data normalization and transformation." In: *Nature Genetics* 32 (2002), pages 496–501. DOI: 10.1038/ng1032.

[131] M. A. Quail, M. Smith, P. Coupland, T. D. Otto, et al. "A tale of three next generation sequencing platforms: comparison of Ion Torrent, Pacific Biosciences and Illumina MiSeq sequencers". In: *BMC Genomics* 13 (2012), page 341. DOI: doi:10.1186/1471-2164-13-341.

[132] F. L. Raymond and P. Tarpey. "The genetics of mental retardation." In: *Human Molecular Genetics* 15.suppl 2 (2006), R110–R116. DOI: 10.1093/hmg/ddl189.

[133] T. B. K. Reddy, A. Thomas, D. Stamatis, J. Bertsch, et al. "The Genomes OnLine Database (GOLD) v.5: a metadata management system based on a four level (meta)genome project classification." In: *Nucleic Acids Research* 43 (2014), pages D1099–D1106. DOI: 10.1093/nar/gku950.

[134] G. Renaud, U. Stenzel, and J. Kelso. "leeHom: adaptor trimming and merging for Illumina sequencing reads." In: *Nucleic Acids Research* 42.18 (2014), e141. DOI: 10.1093/nar/gku699.

[135] J. T. Robinson, H. Thorvaldsdóttir, W. Winckler, M. Guttman, E. S. Lander, G. Getz, and J. P. Mesirov. "Integrative genomics viewer." In: *Nature Biotechnology* 29.1 (2011), pages 24–26. DOI: 10.1038/nbt.1754.

[136] R. Sakai, C. W. Bartlett, D. Popovic, W. C. Ray, and J. Aerts. *Aracari: exploration of eQTL data through visualization.* BioVis 2012 Conference Poster. 2012.

[137] F. A. San Lucas, N. A. Rosenberg, and P. Scheet. "Haploscope: A tool for the graphical display of haplotype: Structure in populations." In: *Genetic Epidemiology* 36.1 (2012), pages 17–21. DOI: 10.1002/gepi.20640.

[138] F. Sanger, S. Nicklen, and A. R. Coulson. "DNA sequencing with chain-terminating inhibitors." In: *Proceedings of the National Academy of Sciences of the United States of America* 74.12 (1977), pages 5463–5467. DOI: 10.1073/pnas.74.12.5463.

[139] P. D. Sasieni. "From genotypes to genes: doubling the sample size." In: *Biometrics* 53.4 (1997), pages 1253–1261. DOI: 10.2307/2533494.

[140] S. Sawyer, J. Krause, K. Guschanski, V. Savolainen, and S. Pääbo. "Temporal patterns of nucleotide misincorporations and DNA fragmentation in ancient DNA". In: *PLoS ONE* 7.3 (2012), e34131. DOI: 10.1371/journal.pone.0034131.

[141] D. Schaid and S. Jacobson. "Biased Tests of Association: Comparisons of Allele Frequencies When Departing from Hardy-Weinberg Proportions." In: *American Journal of Epidemiology* 149.8 (1999), pages 706–711.

[142] M. Schena, D. Shalon, R. W. Davis, and P. O. Brown. "Quantitative Monitoring of Gene Expression Patterns with a Complementary DNA Microarray." In: *Science* 270.5235 (1995), pages 467–470. DOI: 10.1126/science.270.5235.467.

[143] V. J. Schuenemann, K. Bos, S. DeWitte, S. Schmedes, et al. "Targeted enrichment of ancient pathogens yielding the pPCP1 plasmid of Yersinia pestis from victims of the Black Death." In: *Proceedings of the National Academy of Sciences (PNAS)* 108.38 (2011), E746–E752. DOI: 10.1073/pnas.1105107108.

[144] V. J. Schuenemann, P. Singh, T. A. Mendum, B. Krause-Kyora, et al. "Genome-wide comparison of medieval and modern Mycobacterium leprae." In: *Science* 341.6142 (2013), pages 179–183. DOI: 10.1126/science.1238286.

[145] Z. Sidak. "On Multivariate Normal Probabilities of Rectangles: Their Dependence on Correlations." In: *The Annals of Mathematical Statistics* 39.5 (1968), pages 1425–1434.

[146] P. Skoglund, B. H. Northoff, M. V. Shunkov, A. P. Derevianko, S. Pääbo, J. Krause, and M. Jakobsson. "Separating endogenous ancient DNA from modern day contamination in a Siberian Neandertal." In: *Proceedings of the National Academy of Sciences of the United States of America (PNAS)* 111.6 (2014), pages 2229–2234. DOI: 10.1073/pnas.1318934111.

[147] T. F. Smith and M. S. Waterman. "Identification of common molecular subsequences." In: *Journal of Molecular Biology* 147.1 (1981), pages 195–197. DOI: 10.1016/0022-2836(81)90087-5.

[148] G. K. Smyth and T. Speed. "Normalization of cDNA microarray data." In: *Methods* 31.4 (2003), pages 265–273. DOI: 10.1016/S1046-2023(03)00155-5.

[149] E. M. Southern. "Detection of specific sequences among DNA fragments separated by gel electrophoresis." In: *Journal of Molecular Biology* 98.3 (1975), pages 503–517.

[150] J. St. John. *SeqPrep*. 2011. URL: https://github.com/jstjohn/SeqPrep (visited on 06/15/2015).

[151] L. V. Stamm. "Global challenge of antibiotic-resistant Treponema pallidum." In: *Antimicrobial Agents and Chemotherapy* 54.2 (2010), pages 583–589. DOI: 10.1128/AAC.01095-09.

[152] A. E. Stark. "The Hardy-Weinberg principle." In: *Genetics and Molecular Biology* 28.3 (2005), page 485. DOI: 10.1590/S1415-47572005000300027.

[153] L. Stein. *Generic Feature Format Version 3 (GFF3).* 2013. URL: http://www.sequenceontology.org/resources/gff3.html.

[154] Student. "The Probable Error of a Mean." In: *Biometrika* 6.1 (1908), pages 1–25. DOI: 10.1093/biomet/6.1.1.

[155] S. Symons, C. Zipplies, F. Battke, and K. Nieselt. "Integrative systems biology visualization with MAYDAY." In: *Journal of Integrative Bioinformatics* 7.3 (2010). DOI: 10.2390/biecoll-jib-2010-115.

[156] K. Tamura, G. Stecher, D. Peterson, A. Filipski, and S. Kumar. "MEGA6: Molecular evolutionary genetics analysis version 6.0." In: *Molecular Biology and Evolution* 30.12 (2013), pages 2725–2729. DOI: 10.1093/molbev/mst197.

[157] R Core Team. "R: A language and environment for statistical computing." In: *R Foundation for Statistical Computing, Vienna, Austria* (2014). URL: URLhttp://www.R--project.org/.

[158] D. C. Thomas and J. S. Witte. "Point: Population Stratification: A Problem for Case-Control Studies of Candidate-Gene Associations?" In: *Cancer Epidemiology, Biomarkers and Prevention* 11.6 (2002), pages 505–512.

[159] J. W. Tukey. *Exploratory Data Analysis.* 1st edition. Addison-Wesley, 1977. DOI: 10.1007/978-1-4419-7976-6.

[160] S. D. Turner. "qqman: an R package for visualizing GWAS results using Q-Q and manhattan plots." In: *bioRxiv* (2014). DOI: 10.1101/005165.

[161] F. Utro, N. Haiminen, D. Livingstone, O. E. Cornejo, et al. "iXora: exact haplotype inferencing and trait association." In: *BMC Genetics* 14 (2013), page 48. DOI: 10.1186/1471-2156-14-48.

[162] V. E. Velculescu, L. Zhang, B. Vogelstein, and K. W. Kinzler. "Serial analysis of gene expression." In: *Science* 270.5235 (1995), pages 484–487. DOI: 10.1126/science.270.5235.484.

[163] J. C. Venter, M. D. Adams, E. W. Myers, P. W. Li, et al. "The sequence of the human genome." In: *Science* 291.5507 (2001), pages 1304–1351. DOI: 10.1126/science.1058040.

[164] A. Wald. "Tests of Statistical Hypotheses Concerning Several Parameters When the Number of Observations is Large." In: *Transactions of the American Mathematical Society* 54.3 (1943), pages 426–482. DOI: 10.2307/1990256.

[165] A. Walker, G. Nicholas, D. Pullman, and A. Goodman. *Ancient DNA (aDNA) What is it? Why is it important?* Fact Sheet - Presented by the Intellectual Property Issues in Cultural Heritage Project. 2014.

[166] F. O. Walker. "Huntington's disease." In: *Lancet* 369.9557 (2007), pages 218–228. DOI: 10.1016/S0140-6736(07)60111-1.

[167] K. Wang, M. Li, and H. Hakonarson. "ANNOVAR: functional annotation of genetic variants from high-throughput sequencing data." In: *Nucleic Acids Research* 38.16 (2010), e164. DOI: 10.1093/nar/gkq603.

[168] J. Weischenfeldt, O. Symmons, F. Spitz, and J. O. Korbel. "Phenotypic impact of genomic structural variation: insights from and for human disease." In: *Nature Reviews Genetics* 14.2 (2013), pages 125–38. DOI: 10.1038/nrg3373.

[169] T. A. Welch. "A Technique for High Performance Data Compression." In: *IEEE Computer* 17.6 (1984), pages 8–19. DOI: 10.1109/MC.1984.1659158.

[170] T. P. Yang, C. Beazley, S. B. Montgomery, A. S. Dimas, M. Gutierrez-Arcelus, B. E. Stranger, P. Deloukas, and E. T. Dermitzakis. "Genevar: A database and Java application for the analysis and visualization of SNP-gene associations in eQTL studies." In: *Bioinformatics* 26.19 (2010), pages 2474–2476. DOI: 10.1093/bioinformatics/btq452.

[171] N. M. Zetola and J. D. Klausner. "Syphilis and HIV Infection: An Update." In: *Clinical Infectious Diseases* 44.9 (2007), pages 1222–1228.

[172] D. Zhang, L. Cheng, J. A. Badner, C. Chen, et al. "Genetic Control of Individual Differences in Gene-Specific Methylation in Human Brain." In: *American Journal of Human Genetics* 86.3 (2010), pages 411–419. DOI: 10.1016/j.ajhg.2010.02.005.

[173] K. T. Zondervan and L. R. Cardon. "Designing candidate gene and genome-wide case-control association studies." In: *Nature Protocols* 2.10 (2007), pages 2492–2501. DOI: 10.1038/nprot.2007.366.

[174] W. Zou, D. L. Aylor, and Z.-B. Zeng. "eQTL Viewer: visualizing how sequence variation affects genome-wide transcription." In: *BMC Bioinformatics* 8 (2007), page 7. DOI: 10.1186/1471-2105-8-7.

[175] M. Zuker, D. H. Mathews, and D. H. Turner. "Algorithms and Thermodynamics for RNA Secondary Structure Prediction: A Practical Guide." In: *RNA Biochemistry and Biotechnology*. Edited by J. Barciszewski and B. F. C. Clark. NATO ASI Series. Kluwer Academic Publishers, 1999, pages 11–43.

A. Supplementary Material

Table A.1: Population abbreviations used during data analysis of Phase 1 of the 1000 Genomes project.

Population abbreviation	Full population name
ASW	People with African ancestry in Southwest United States
CEU	Utah residents with ancestry from Northern and Western Europe
CHB	Han Chinese in Beijing, China
CHS	Han Chinese South, China
CLM	Colombians in Medellin, Colombia
FIN	Finnish in Finland
GBR	British from England and Scotland
IBS	Iberian populations in Spain
LWK	Luhya in Webuye, Kenya
JPT	Japanese in Tokyo, Japan
MXL	People with Mexican ancestry in Los Angeles, California
PUR	Puerto Ricans in Puerto Rico
TSI	Tuscani in Italy
YRI	Yoruba in Ibadan, Nigeria

Table A.2: Super population abbreviations used during data analysis of Phase 1 of the 1000 Genomes project.

Super population abbreviation	Full super population name
AFR	Africans
AMR	Americans
ASN	East Asians
EUR	Europeans

Table A.3: Overview of the available SNPeff effect prediction categories and the REVEAL impact classes assigned to each category. The categories have been separated based on their regional (top), or functional (bottom) classification.

SnpEff Effect Category	Description	REVEAL impact class
5' UTR	SNV in untranslated region upstream of a gene	middle
CDS	SNV in protein coding sequence (introns excluded)	middle
Gene	SNV in gene region (introns included)	middle
Transcript	SNV in specific gene transcript	middle
Exon	SNV in exon region	middle
Intron	SNV in intron region	none
3' UTR	SNV in untranslated region downstream of a gene	low
Intragenic	SNV in non-coding intragenic region	none
Start codon gained	gain of an additional start codon upstream of a gene	middle
Start codon lost	SNV in the start codon leading to a shorter gene product	high
Splice site acceptor	SNV in one of the two bases before an exon (not the first)	high
Splice codon lost	SNV in one of the two bases after an exon (not the last)	high
Non-Synonymous	amino-acid change	high
Synonymous	no amino-acid change	none
Stop codon gained	SNV causes an additional stop codon before the original one	high
Stop codon lost	SNV in the original stop codon leading to a longer gene product	high

Table A.4: Performance evaluation of the ClipAndMerge tool in comparison to five other tools capable of performing adapter clipping and read merging of overlapping paired-end reads. As test data five different aDNA leprosy data sets (3077, Jorgen625, Refshale16, SK2, and SK8) were used, which were introduced in the publication on the comparison of medieval and modern *Mycobacterium leprae* [144].

		Category 1 Read Merging		Category 2 Mapping merged reads		Category 3 Mapping all reads		Category 4 Mapping Quality		Category 5 Genome Coverage	Category 6 Runtime	
	Read pairs	Merged Reads	% Merged Reads	Merged Mapped Reads	% Merged Mapped Reads	Total Mapped Reads	% Total Mapped Reads	Mapped Quality >= 30	% Mapped Quality >= 30		Overall Runtime	Speed (Reads /Second)
3077												
ClipAndMerge 1.6	6029646	5733289	95.09	978911	16.23	1043672	17.31	892546	14.8	19.05	4m14s	47555
MergeReadsFastQ		5614634	93.12	959070	15.91	1006194	16.69	855926	14.2	19.5	344m27s	584
Cutadapt 1.7.1 + FLASH 1.2.11		1206663	20.01	360360	5.98	1036940	17.2	882756	14.64	20.68	3m28s	57957
SeqPrep 1.1		5170731	85.76	894553	14.84	949073	15.74	813371	13.49	18.83	17m28s	11506
LeeHom (10.01.2015)		5550461	92.05	NA	NA	981558	16.28	842205	13.97	18.85	11m14s	17904
AdapterRemoval 1.5.4		4953032	82.14	867846	14.39	931529	15.45	798394	13.24	18.59	10m11s	19738
Jorgen625												
ClipAndMerge 1.6	15101591	12966130	85.86	2469221	16.35	2703428	17.9	2628489	17.41	130.5	24m38s	20435
MergeReadsFastQ		14885564	98.57	2613970	17.31	2623243	17.37	2551352	16.89	136.89	2948m46s	171
Cutadapt 1.7.1 + FLASH 1.2.11		6336789	41.96	738251	4.89	2599158	17.21	2528504	16.74	138.83	13m3s	38587
SeqPrep 1.1		14495019	95.98	2569043	17.01	2595366	17.19	2526173	16.73	135.51	107m2s	4703
LeeHom (10.01.2015)		14762403	97.75	NA	NA	2617909	17.34	2547350	16.87	136.62	112m31s	4474
AdapterRemoval 1.5.4		13925528	92.21	2472718	16.37	2524087	16.71	2455254	16.26	132.34	172m14s	2923
Refshale16												
ClipAndMerge 1.6	39915365	32134465	80.51	8300013	20.79	13368593	33.49	11800301	29.56	294.7	25m15s	52689
MergeReadsFastQ		35094974	87.92	8965762	22.46	13812705	34.6	12190382	30.54	351.62	2679m53s	497
Cutadapt 1.7.1 + FLASH 1.2.11		21240199	53.21	6201518	15.54	11533714	28.9	10210110	25.58	358.26	24m43s	53841
SeqPrep 1.1		33999864	85.18	8784400	22.01	11516472	28.85	10197992	25.55	351.45	102m50s	12939
LeeHom (10.01.2015)		29999086	75.16	NA	NA	8916759	22.34	7874692	19.73	236.91	72m45s	18288
AdapterRemoval 1.5.4		33091793	82.9	8501406	21.3	11431717	28.64	10123045	25.36	351.6	74m9s	17944
SK2												
ClipAndMerge 1.6	54243849	51967552	95.8	48326422	89.09	48536318	89.48	40257497	74.22	811.8	41m57s	43103
MergeReadsFastQ		51925800	95.73	48606916	89.61	48610983	89.62	40262488	74.22	813.904	3573m5s	506
Cutadapt 1.7.1 + FLASH 1.2.11		1826703	3.37	1688348	3.11	48402551	89.23	40032482	73.8	814.01	37m53s	47731
SeqPrep 1.1		51467042	94.88	48224482	88.9	48240750	88.93	40031770	73.8	809.4	200m6s	9020
LeeHom (10.01.2015)		51231481	94.45	NA	NA	48337919	89.11	40101607	73.93	811.19	112m27s	16079
AdapterRemoval 1.5.4		50206047	92.56	47073729	86.78	47095207	86.82	39086741	72.06	804.142	162m46s	11108
SK8												
ClipAndMerge 1.6	9898159	7738156	78.18	831640	8.4	1283126	12.96	1114590	11.26	28.55	6m44s	48962
MergeReadsFastQ		8554467	86.42	913647	9.23	1280119	12.93	1116550	11.28	33.33	663m40s	497
Cutadapt 1.7.1 + FLASH 1.2.11		4099160	41.41	656450	6.63	1109626	11.21	969662	9.8	33.9861	5m36s	58920
SeqPrep 1.1		8066460	81.49	893847	9.03	1107013	11.18	968141	9.78	33.32	24m7s	13685
LeeHom (10.01.2015)		7027405	71	NA	NA	908549	9.18	792546	8.01	24.64	18m17s	18052
AdapterRemoval 1.5.4		7580444	76.58	868848	8.78	1100326	11.12	962319	9.72	33.35	17m16s	19100

Table A.5: Overview of the default parameters used by the BWA `mem` and BWA `aln/samse` algorithms. For more information on the individual parameters please see the official manual page at `http://bio-bwa.sourceforge.net/bwa.shtml` (last accessed: September 20, 2015). NA indicates that the respective parameter is not available.

Parameter	BWA `mem`	BWA `aln/samse`
Minimum seed length	-k 19	NA
Band width	-w 100	NA
Off-diagonal X-dropoff	-d 100	NA
Trigger re-seeding	-r 1.5	NA
Max occurrences	-c 10000	NA
Matching score	-A 1	NA
Mismatch penalty	-B 4	-M 3
Gap open penalty	-O 6	-O 11
Gap extension penalty	-E 1	-E 4
Clipping penalty	-L 5	NA
Minimum alignment score	-T 30	NA
Maximum number of gap opens	NA	-o 1
Maximum number of gap extensions	NA	-e −1
Long extension threshold	NA	-d 16
Indel length threshold	NA	-i 5
Number of subsequences as seed	NA	-l inf
Maximum edit distance in the seed	NA	-k 2

A.1 Available SNV Filter Methods in Reveal

- ▷ *Aggregation:* This filter is based on the aggregated genotype distribution of a SNV in the case group and in the control group. Only those SNVs for which the aggregated distributions differ, with respect to the maximum aggregation method, are selected.
- ▷ *Aggregation Difference:* In addition to the aggregation filter, the aggregation difference filter takes user-defined differences in the aggregated frequencies into account.
- ▷ *Chromosomal Location:* SNVs can be selected based on their chromosomal location. Thereby, a position range on a specific chromosome can be defined. All SNVs contained in that region are filtered.
- ▷ *Closest Gene:* Each SNV is assigned to one of the genes in the Reveal project according to its genetic distance. The user can then select any of

the genes resulting in a selection of all SNVs for which the selected gene is closest in proximity with respect to all genes in the current project.

- ▷ *Contained in SNVList:* This filter allows one to intersect or merge sets of SNVs, by specifying already created SNVLists and combining them using set methods.

- ▷ *SNV No-Call Rate:* This filter allows for the selection of only those SNVs, for which the number of individuals with missing genotype information is within a user-defined threshold.

- ▷ *SNV Selection:* SNVs can be selected within any of the visualizations provided in REVEAL. With this filter, user made selection can be transformed into SNVLists for further processing or selection storage.

- ▷ *SNV Identifier:* Regular expressions can be defined for SNV identifiers. Furthermore, comma separated lists of different SNV identifiers can be provided based on which SNVs from the project are selected.

- ▷ *Minor allele frequency (MAF):* not all SNVs are polymorphic, some show only one allele across all individuals (monomorphic) or one of the alleles will be at a very low frequency. The association between a phenotype and a rare allele might be supported by only a few individuals (no power to detect the association). The result should be interpreted with caution. SNV filtering based on MAF is often used to exclude low MAF SNVs (usual thresholds are between 1% and 5%).

- ▷ *Hardy-Weinberg Equilibrium:* The statistical test used for the calculation of significance with respect to the HW equilibrium is a typical Fisher's Exact test followed by a Bonferroni correction for multiple testing. Common p-value thresholds for HW are e.g. 10^{-4} or less.

- ▷ *Case-/Control Statistics:* In REVEAL various different statistical tests for case-/control-based studies are available. With this filter p-value (either corrected for multiple testing or uncorrected) thresholds can be defined for SNV selection.

- ▷ *Single-Locus eQTL Association Tests:* Allows for the definition of p-value thresholds from single-locus eQTL association tests conducted with PLINK.

- ▷ *Two-Locus eQTL Association Tests:* Allows for the definition of p-value thresholds for pairs of SNVs with respect to a pairwise SNV based eQTL association test conducted with PLINK. Both SNVs from a significantly associated SNV pair are selected.

B. Publications

B.1 Articles

2011

- **Günter Jäger**, Florian Battke, and Kay Nieselt. *TIALA - Time Series Alignment Analysis.* Biological Data Visualization (BioVis), 2011 IEEE Symposium on, Oct 2011; 55-61

2012

- **Alexander Herbig**, **Florian Battke**, **Günter Jäger**, and Kay Nieselt. *GenomeRing: alignment visualization in SuperGenome coordinates.* Bioinformatics, Jun 2012, 28(12):i7-15

- **Günter Jäger**, **Florian Battke**, and Kay Nieselt. *Reveal - Visual eQTL analytics.* Bioinformatics, Sep 2012, 28(18):i542-i548

- **Julian Heinrich**, Corinna Vehlow, Florian Battke, Günter Jäger, Daniel Weiskopf, and Kay Nieselt. *iHAT: interactive Hierarchical Aggregation Table for Genetic Association Data.* BMC Bioinformatics, May 2012, 13(Suppl 8):S2

- **Christopher W. Bartlett**, Soo Yeon Cheong, Liping Hou, Jesse Paquette, Pek Yee Lum, Günter Jäger, Florian Battke, Corinna Vehlow, Julian Heinrich, Kay Nieselt, Ryo Sakai, Jan Aerts, and William C. Ray. *An eQTL biological data visualization challenge and approaches from the visualization community.* BMC Bioinformatics, May 2012, 13(Suppl 8):S8

2013

- **Verena J. Schuenemann**, **Pushpendra Singh**, **Thomas A. Mendum**, **Ben Krause-Kyora**, **Günter Jäger**, **Kirstin I. Bos**, Alexander Herbig, Christos Economou, Andrej Benjak, Philippe Busso, Almut Nebel, Jesper L. Boldsen, Anna Kjellström, Huihai Wu, Graham R. Stewert, G. Michael Tayler, Peter Bauer, Oona Y.-C. Lee, Houdini H.T. Wu, David E. Minnikin, Gurdyal S. Besra, Katie Tucker, Simon Roffey, Samba O. Sow, Stewert T. Cole, Kay Nieselt, and Johannes Krause. *Genome-wide comparison of medieval and modern Mycobacterium leprea.* Science, Jul 2013, 341(6142): 179-83

2014

- **Günter Jäger**, Alexander Peltzer, and Kay Nieselt. *inPHAP: Interactive visualization of genotype and phased haplotype data.* BMC Bioinformatics, Jul 2014, 15:200

- **Kirsten I. Bos**, **Günter Jäger**, Verena J. Schuenemann, Ashild Vagene, Maria A. Spyrou, Alexander Herbig, Kay Nieselt, and Johannes Krause. *Parallel Detection of Ancient Pathogens via Array-based DNA Capture.* Philosophical Transactions of the Royal Society B Biological Science, Jan 2015, 370(1660):20130375

B.2 Posters & Presentations

2011

- Günter Jäger, Florian Battke, and Kay Nieselt. *Tiala - Visual Time Series Alignment Analysis.* German Conference on Bioinformatics (GCB) 2011

2012

- Alexander Herbig, Florian Battke, Günter Jäger, and Kay Nieselt. *GenomeRing: alignment visualization in SuperGenome coordinates.*

- Günter Jäger, Florian Battke, Corinna Vehlow, Julian Heinrich, and Kay Nieselt. *Reveal - Visual eQTL Analytics.* **Presentation** at the VizBi 2012

2013

- Günter Jäger, Florian Battke, Karsten Borgwardt, and Kay Nieselt. *Using Reveal to visually detect significant SNPs in eQTL data.* VizBi 2013

- Günter Jäger, and Kay Nieselt. *Interactive Visualization and Decision Support for eQTL data.* **Invited Talk** at the Visualization in Medicine and Life Sciences (VMLS) 2013

2014

- Günter Jäger, Alexander Peltzer, and Kay Nieselt. *inPHAP: Interactive visualization of genotype and phased haplotype data.* **Presentation** at the BioVis 2014

B.3 Awards

2011

- Florian Battke, Stephan Symons, Günter Jäger, Aydin Can Polatkan, Alexander Herbig, and Kay Nieselt. *GenomeRing: Visual Comparison of Multiple Genomes.* Winner of the **Most Creative Algorithm** award (academic entries) at the iDEA challenge 2011
- Günter Jäger, Florian Battke, Corinna Vehlow, Julian Heinrich, and Kay Nieselt. *Reveal - Visual eQTL analytics.* **Vis Experts Favorite** at BioVis 2011

C. Academic Teaching Experience

C.1 Supervised Lectures and Courses

WS 2011/12

- Tutorial: *Microarray Bioinformatics* for Bachelor students

SS 2012

- Tutorial: *Grundlagen der Bioinformatik* for Bachelor students
- Tutorial: *Einführung in die Bioinformatik* for Bachelor students
- Practical Course: *Practical Transcriptomics* for Master students

WS 2012/13

- Tutorial: *Microarray Bioinformatik* for Bachelor students

SS 2013

- Tutorial: *Grundlagen der Bioinformatik* for Bachelor students

WS 2013/14

- Tutorial: *Bioinformatics I* for Master students
- Tutorial: *Microarray Bioinformatik* for Bachelor students

SS 2014

- Practical Course: *Software Engineering* for Bachelor students

C.2 Supervised Bachelor/Master and Diploma Theses

2012

- Sebastian Nagel. *Parallelisierung des QT-Clusterings in Mayday.* August 2012
- Jennifer Lange. *Schnelle und interaktive Clusterverfahren für Transkriptomdaten von Cyanobakterien.* September 2012

2013

- Eugen Netz. *Interaktives visuelles Clustering.* August 2013
- Ina Spirer. *Mikrogravitationsabhängige Genexpression in Arabidopsis thaliana.* August 2013
- Alexander Peltzer. *Efficient algorithms for Ancient Human Genome Reconstruction.* November 2013

2014

- Simon Heumos. *TOPAS - Toolkit for Processing and Annotating Sequence Data.* April 2014
- André Hennig. *From the SuperGenome to the Pangenome of Bacteria.* April 2014
- Alicia Owen. *The expression landscape of monozygotic twins.* August 2014
- Natalya Sabirova. *StreptoExpress - genome-wide expression analyses of Streptomyces coelicolor.* September 2014