Comenius University in Bratislava
Faculty of Mathematics, Physics and Informatics

Combined prediction of genomic structural variants with low coverage sequencing

Bachelor Thesis

2020

Zuzana Klinovská

Comenius University in Bratislava
Faculty of Mathematics, Physics and Informatics

Combined prediction of genomic structural variants with low coverage sequencing

Bachelor Thesis

Study Programme:	Computer Science
Field of Study:	Computer Science
Department:	Department of Computer Science
Supervisor:	Dr. techn. Marcel Kucharík

Bratislava, 2020
Zuzana Klinovská

Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta:	Zuzana Klinovská
Študijný program:	informatika (Jednoodborové štúdium, bakalársky I. st., denná forma)
Študijný odbor:	informatika
Typ záverečnej práce:	bakalárska
Jazyk záverečnej práce:	anglický
Sekundárny jazyk:	slovenský

Názov: Combined predicion of genomic structural variants with low coverage sequencing
Kombinovaná predikcia štrukturálnych variantov genómu pomocou sekvenovania s nízkym pokrytím

Anotácia: Práca sa zaoberá bioinformatickými metódami na predikciu štrukturálnych variantov v genóme pre dáta získané sekvenovaním druhej generácie s nízkym pokrytím. Cieľom je popísať rozdiely a porovnať existujúce nástroje, vybrať zopár najperspektívnejších nástrojov a evaluovať ich účinnosť pri rôznych parametroch štrukturálnych variácií. Skombinovať výstupy z týchto nástrojov a vytvoriť meta-prediktor, ktorý bude mať väčšiu presnosť predikcie, poprípade priamo skombinovať použité metódy normalizácie dát a filtrovania do jedného nového prediktora a tento otestovať.

Vedúci:	Dr. techn. Marcel Kucharík
Katedra:	FMFI.KI - Katedra informatiky
Vedúci katedry:	prof. RNDr. Martin Škoviera, PhD.
Dátum zadania:	28.10.2019

Dátum schválenia: 30.10.2019

doc. RNDr. Daniel Olejár, PhD.
garant študijného programu

....
študent

....
vedúci práce

44539933

Comenius University in Bratislava
Faculty of Mathematics, Physics and Informatics

THESIS ASSIGNMENT

Name and Surname:	Zuzana Klinovská
Study programme:	Computer Science (Single degree study, bachelor I. deg., full time form)
Field of Study:	Computer Science, Informatics
Type of Thesis:	Bachelor´s thesis
Language of Thesis:	English
Secondary language:	Slovak

Title: Combined predicion of genomic structural variants with low coverage sequencing

Annotation: The topic of the thesis are bioinformatics methods for prediction of structural variants in genome from low-coverage second-generation sequencing data. The goal is to compare existing tools and discuss their differences, strong and weak points. Choose several most promising tools for complete evaluation on different structural variants. Combine outputs from these tools to create a meta prediction tool with better accuracy. Alternatively, combine methods used in these tools for normalization and filtering in one novel tool and evaluate this tool.

Supervisor:	Dr. techn. Marcel Kucharík	
Department:	FMFI.KI - Department of Computer Science	
Head of department:	prof. RNDr. Martin Škoviera, PhD.	
Assigned:	28.10.2019	
Approved:	30.10.2019	doc. RNDr. Daniel Olejár, PhD. Guarantor of Study Programme

....

Student Supervisor

Acknowledgments: I would like to express my special thanks of gratitude to my supervisor Mr.Kucharík, who helped me complete this thesis. I thank company Geneton for providing data. Special thanks belongs to my dear friend who supported me and encouraged me during this work.

Abstrakt

Duplikácie a delécie rôznych častí DNA sú známym zdrojom chorôb a syndrómov. V našej práci sme uviedli porovnanie nástrojov, ktoré sa špecializujú na detekciu takýchto delécií a duplikácií. Porovnali sme dokopy štyri nástroje: CNV-caller, WisecondorX, iCopyDAV a CNVkit. Pre každý nástroj sme uviedli ich popis a niektoré výhody a nevýhody ich použitia. Pre porovnanie sme použili 54 vzoriek s potvrdeným výskytom duplikácií a delécií. Následne sme narvhli predikčný model na vylepšenie detekcie týchto javov.

Kľúčové slová: detekcia CNV, bioinformatika, nástroje na detekciu CNV

Abstract

Duplications and deletions of different sections of DNA are a cause of various genetic disorders and diseases. In our work we compared tool that specialize in detection of these kind of deletions and duplications. Together we compared four different tools: CNV-caller, WisecondorX, iCopyDAV and CNVkit. W described used methods of the individual tools and listed some of their advantages and disadvantages. For this comparison we used 54 samples with confirmed deletions and duplications. Subsequently, we proposed a combined predictive model that improves the detection of mentioned phenomenon.

Keywords: CNV detection, bioinformatics, CNV detection tools

Contents

List of Figures

List of Tables

Introduction

Bioinformatic technologies have developed dramatically over the past decades. It took over ten years to assemble first human genome. Nowadays, next generations sequencing technologies are capable of sequencing whole genome in one day and for much lower cost.

This resulted in numerous scientific researches, analyses and a development of new bioinformatic software tools. Major attention deserve mutations on DNA. Although, they are essential for evolution of all living organisms, they can cause numerous health problems and complications. Copy number variants (CNV) are types of mutations that are linked to several syndromes and diseases. Various software tools were created specifically for detection of this type of mutation.

In this thesis we present a comparison of four different CNV detection tools. We compare their limitations to different factors, which affect the detection process.

Furthermore, we propose a prediction model, which comprised of combination of selected tools and suggested alternations for a better prediction.

In the first chapter, we put fundamental terms into context, for instance genome, chromosome, nucleic acids and so on. We describe how are data obtained from DNA by various technologies and a brief history of sequencing is presented as well. To continue, we introduce copy number variations, their importance in nature as well as their unfavorable impact on humans. Subsequently, we describe a method, from which originate our test samples.

Second chapter comprises of a deeper description of selected CNV detection tool. Methods that individual tools use, are listed as well with a detailed elucidation of particular approaches. In addition, we present factors that have an impact on CNV prediction in this section.

Third chapter is dedicated to a software, through which we were able to run each tool. We present what are the benefits of this tool and how it simplified the analyses. We briefly explain how this software functions.

Fourth chapter presents final results and overall evaluation of the comparison. We added different tables and graphs, where we point out the overall performance of the individual tools. To directly demonstrate the efficiency, we included various statistics for each tool. Advantages and disadvantages are included in this section from both the user and the functional point of view. Finally, we propound a prediction model, where we selected tools that performed the best in the analyses and we describe what combinations and approaches would increase the overall fitness of the prediction itself.

Chapter 1

Biological and Bioinformatics background

This chapter will briefly cover some bioinformatic terms, that are important to understand and will be mentioned later in this thesis. We will introduce DNA structure and some basics about genetic information itself. Subsequently we will describe methods for sequencing DNA and finally, we will define Copy Number Variations.

1.1 DNA structure

Genetic information of all living organisms is carried by molecules of DNA (Deoxyribonucleic acid) [37]. These molecules are present in every cell of an organism. They encode how will the organism develop throughout its lifetime, how many limbs it will grow, what diseases it will be prone to and other information. DNA molecule consists of coding regions called **genes** and non coding regions. Synthesis of other gene products, such as proteins, is encoded in genes.

The term **genome** is used for genetic material of an organism. It includes genes and non coding regions of DNA as well as other genetic material.

DNA composes of two strands that form double helix structure , which consists of smaller structural units called **nucleotides** [37]. These two strands are of equal length and are aligned so that i-th nucleotide is connected by a hydrogen bond to the *(n-i+1)*-th nucleotide on the other strand. Each nucleotide stores one of the following bases: adenine, cytosine, guanine, thymine. Simple drawing of DNA is shown in Figure 1.1. Later in this thesis, we will not differentiate between the terms base and nucleotide, since nucleotides differ mainly in bases that they contain.

DNA molecule can be described as a string of letters A, C, G, T. **Segment** is a sub-sequence or sub-string of a sequence . This string is not symmetric meaning that it can be determined where the strand starts and where it ends. According to

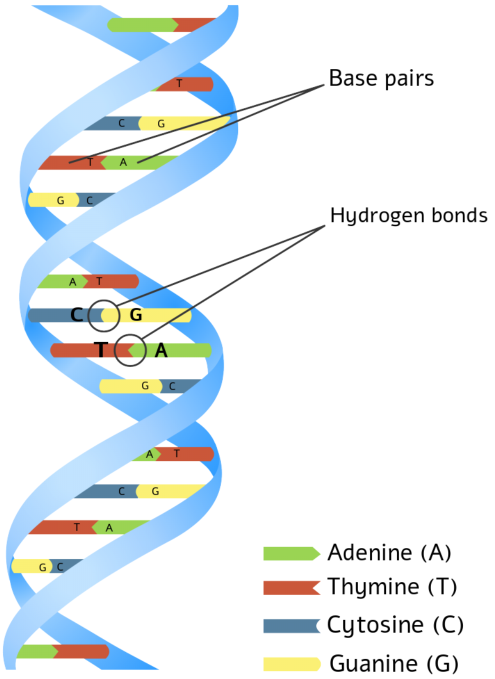

Figure 1.1: Structure of the DNA. Double helix is imitated by two blue strands [31]. Bases are bond together according to base pairing rules

base pairing rules Adenine (A) always pairs with thymine (T), whereas Guanine (G) pairs with Cytosine (C) as is showed in Figure 1.1. Therefore, one strand is enough to determine complete DNA sequence [44]. We can do this by replacing a base with its complementary base and subsequently reversing the order, for instance sequence AATGCC is complementary with GGCATT.

A nucleotide pair, for instance A connected with T, is a **base pair unit** of measurement. As expected, 1 *kb* (kilo base pairs) is equal to 1 000 *bp* (base pairs), 1 *Mb* (Mega base pairs) equals to 1 000 000 *bp*, etc.

Chromosome is a DNA molecule carrying genetic information. Every organism has its own set of chromosomes and each cell of the organism contains certain number of complete sets of chromosomes. Number of the sets is called **ploidy** and depending on this number the organism falls under the category of monoploids (one set of chromosomes), diploids (two sets), triploids (three sets) etc. Human is a diploid organism and a healthy individual has twenty two pairs of chromosomes and a pair of sex chromosomes.

Fixed location on chromosome is called **locus**.

1.2 Structural Variants

To define structural variants, first we need to clarify used terms such as evolution, mutation and so on.

Biological or organic evolution is an alternation in heritable characteristics of populations in the course of generations, usually caused by random mutations [22, Chapter 1].

Mutation is an alternation in DNA sequence. This change may have positive, negative or no effects depending on the location where mutation took place, how many bases have mutated and so on. Mutation itself is a very significant event, it can cause important evolutionary progress. Great example that introduces **positive mutation** is antibiotic resistance [22, Chapter 1]. Before Alexander Fleming's Discovery of Penicillin in 1940s, heart diseases or cancer were not as common. Instead people in hospitals were battling with tuberculosis, typhoid fever, cholera and other bacteria infections. However, as medical situation improved with the above mentioned discovery, most bacteria diseases have been defeated. Unfortunately, as a result of evolution, some of these diseases came back stronger and resistant to the used types of antibiotics. With each new antibiotic, bacterial diseases are evolving. Naturally, this is happening constantly and since every organism mutates, the evolution continues.

Opposite to positive mutation is negative mutation, where the organism is at a disadvantage or the phenomenon can even results in death.

The difference in DNA between populations of species is called **genetic variability**. It can be present in various forms from single nucleotide polymorphism (SNP, an alternation in single nucleotide) to DNA sequence modification or even change in chromosome structure [41].

Structural variation is a change in organism's chromosome, involving a DNA fragment that is approximately 1 *kb* or larger, therefore we can consider it a larger mutation event [19]. Under this category fall:

- **Translocation**, is a phenomenon when a fragment of a chromosome attaches itself to a different chromosome or when two chromosomes exchange their parts. If there is no loss or gain of genetic material, it is referred to as balanced translocation. Opposite to it is an unbalanced translocation, which are linked to several health problems such as leukemia and others [5].

- **Insertion**, occurs when an additional set of base pairs are inserted into a DNA sequence.

- **Inversion**, a phenomena when a segment of a chromosome is in a reversed order. Growth retardation, infertility, cancer and other diseases are linked to this type of mutation [1].

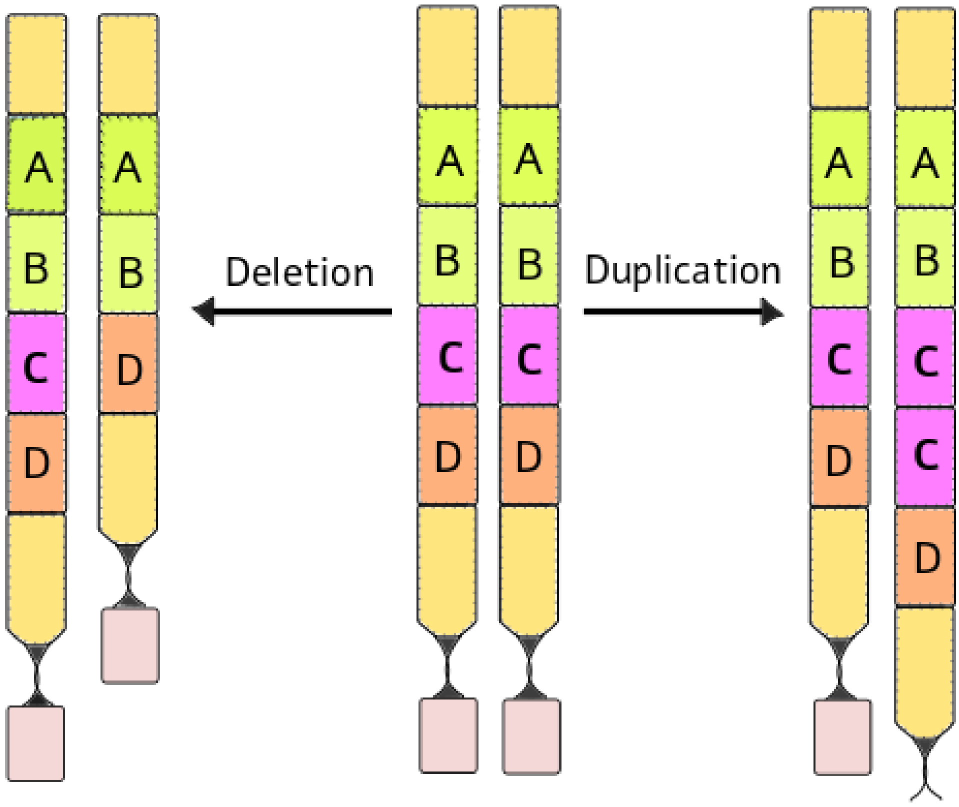

Figure 1.2: An example of duplication and deletion event on a chromosome. The chromosome in the middle is the original one, the right side displays deleted section **C** (neon pink color) and the left side shows chromosome with the same section duplicated.

- Deletions and duplications are in the same category, which is called **Copy Number Variations** or CNVs will be introduced in more details below, since it plays key role in this thesis.

CNV is a phenomenon in which sections of DNA repeat (duplicate) or they are deleted. We can see an artificial example of a CNV event in Figure 1.2.

This phenomenon represents a significant source of genetic diversity among different species including humans. For instance it was discovered that people with low-starch diet have less copies of salivary amylase gene (AMY1) than people with higher-starch diet [36]. As a result, higher number of AMY1 gene can have various benefits, especially for digestive system thus this mutation is positive.

However, CNVs are associated with various syndromes and diseases as well [47]. They are associated with schizophrenia, autism or susceptibility to HIV infection. What is more, CNVs can cover part of a gene, whole gene or even several genes and therefore they are likely to have a role in alternation of human physiological functions, which are essential processes such as metabolism, movements, reproduction etc [47].

Reference genome is often used as a guide to identify abnormal mutations. This reference represents idealized version of a DNA sequence of a species. Scientists assemble the reference from sub-sequences that originate from different donors. In our thesis we will be working with tools that detect copy number variations that differ from the reference genome. These variants are referred to as **aberrant CNVs**.

1.2.1 Non invasive prenatal testing

Non invasive prenatal testing or in short **NIPT**, is method used to detect the risk that a fetus will be born with a particular genome aberration. Small portions of DNA are tested. This kind of DNA is referred to as **cell-free DNA** (cfDNA) because fragments of genetic information are not within cells, but are free-floating in the bloodstream. Normally less than 200 DNA base pairs are contained in one fragment [7].

When a woman is pregnant, her bloodstream contains a mix of cfDNA that either originated from her cells or cells from the placenta [7]. Since fragments from placenta cells are almost identical to the DNA of the fetus, cfDNA can be analyzed for various genome abnormalities. NIPT is mostly used for aneuploidy detection, which is when an extra copy or a missing chromosome is found. Down syndrome (trisomy 21, three copies of chromosome 21), Edwards syndrome (trisomy 18) and other disorders are examples of aneuploidy, but other aspects can be examined such as a gender of the fetus [7].

NIPT falls under the category of screening tests, which means its purpose is to detect the potential risk of having a certain condition. It poses no danger to the fetus or to the pregnant woman. False positives and negatives may also occur. Test is **false positive** if the fetus was diagnosed with increased risk of genetic abnormality, but in reality the fetus was unaffected (for instance only the placenta was effected, not the fetus). Similarly, test is **false negative** when the results show decreased risk of the condition, but the fetus was affected. However, current commercial NIPTs have a very high accuracies (>99%) for whole chromosomal aneuploidies [48].

Data that we will test in this thesis all come from NIPT and we will analyze various structural variants that the fetus may have.

1.3 Introduction to DNA sequencing

In order to work with any DNA sequence it is essential to understand how to obtain it. Method that obtains and determines the order of the nucleotide acid sequence is called **DNA sequencing**. We will briefly introduce the history of sequencing and essential terms.

First human genome was sequenced in 2001 and the work took thirteen years.

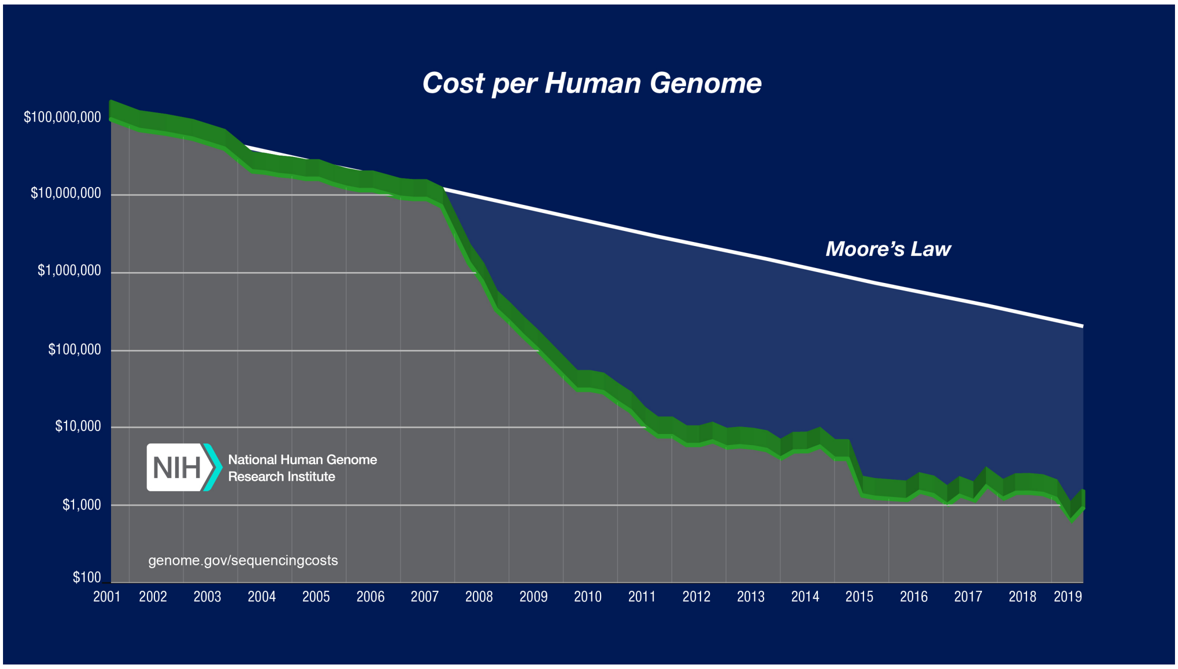

Figure 1.3: The evolution of sequencing cost of a human genome. The x-axis represents years and y-axis refers to the average sequencing cost of that year. The straight line represents Moore's Law, which describes the trend in computer industry, that the computing power doubles every two years [6].

However, with faster technologies and significantly lower sequencing cost, new genome analyses are produced [28]. Nowadays it takes a day in a well equipped laboratory [34]. The cost of sequencing decreased rapidly as is shown in Figure 1.3.

Methods of sequencing can be divided into three generations. The history of the **first generation sequencing** extends to 1970s. British scientist Frederick Sanger along with his colleagues introduced *Sanger's chain termination* [23]. It allowed scientist to finally distinguish the order of the nucleotide bases. Even though this method is very laborious and expensive, it is still used up to this day especially for validations of results of other sequencing methods, since it has a low error rate [21].

Following generation of sequencing technologies is referred to as -**Second-generation sequencing** or **Next-generation sequencing**. During this phase throughput was increased markedly, since biochemical reactions were performed concurrently. High number of short reads is characteristic for this generation. Examples of NGS technologies are Illumina MiSeq [2] from Illumina, SOLiD4 from Life Technologies company.

Third generation include single-molecule sequencing technologies. Most common characteristics among these technologies are long read lengths, lower cost and short computing time. An example of company that is focused in third generations sequencing technologies is Oxford NANOPORE Technologies [3], they are popular for their

	First gen.	Second gen.	Third gen.
Raw read accuracy	High	High	Low
Read length	Moderate (800 -1000 bp)	Short	>1000 bp
Throughput	Low	High	High
Cost per base	High	Low	Medium
Time to result	Hours	Days	Less than a day

Table 1.1: Comparison of First, Second and Third generation of sequencing [33].

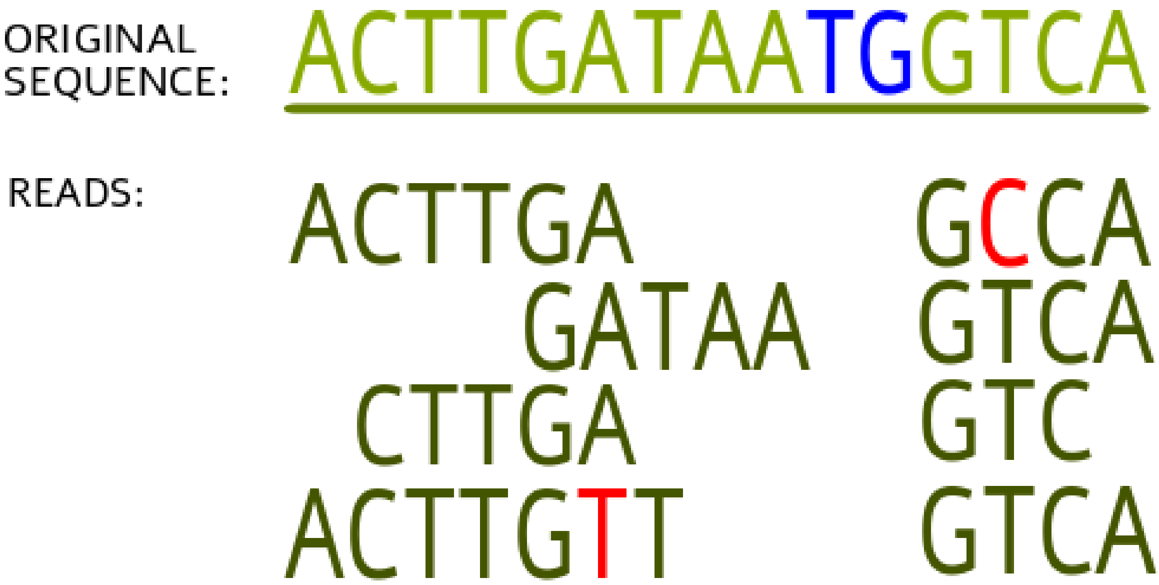

Figure 1.4: An artificial example of a genome(the original sequence) and its reads. The blue color resembles parts of genome that are not covered. The red color shows bases, where a sequencing error occurred.

sequencer MinION, which is very small and can easily fit into a packet.

A comparison of these three generations are described in Table 1.1.

Majority of sequencing methods is not able to read the whole genome at once. A genome is broken into smaller sections during fragmentation process and as a result, smaller subsections called **reads** are produced by sequencers. Reads are sub-strings of chromosome, which are finite and non-empty. Subsequently, heuristic algorithms connect smaller reads into a longer string that resembles the original DNA sequence. These smaller sections ideally cover the whole original sequence, but often there are gaps. Moreover, reads can overlap each other, they may contain sequencing errors due to incorrectly discarded reads and so on [9]. An example of genome and its reads is shown in Figure 1.4.

Genome assembling is a process of aligning and merging fragmented reads in order to remodel the original sequence. There are two approaches for genome assembling problem. The first one is called *de novo assembly*, where original genome sequence is recreated with no prior knowledge [10]. Latter approach is *read mapping*, where reads are aligned (mapped) to a reference genome, which could be created by either of these approaches.

Factors that affect genome assembling are:

- The length of the original sequence - shorted sequences are easier to process.
- The length of the individual reads - longer reads are easier to assemble.
- **Coverage** or **depth of coverage**. It refers to the average number of reads that covered a nucleotide. The term **genome coverage** is an average number of reads that align to a base in the original sequence. With higher coverage (>20x) even rare variants can be found and have other advantages, at the same time it is much costlier [12].

Summarization of the de-novo sequencing process can be seen in Figure 1.5.

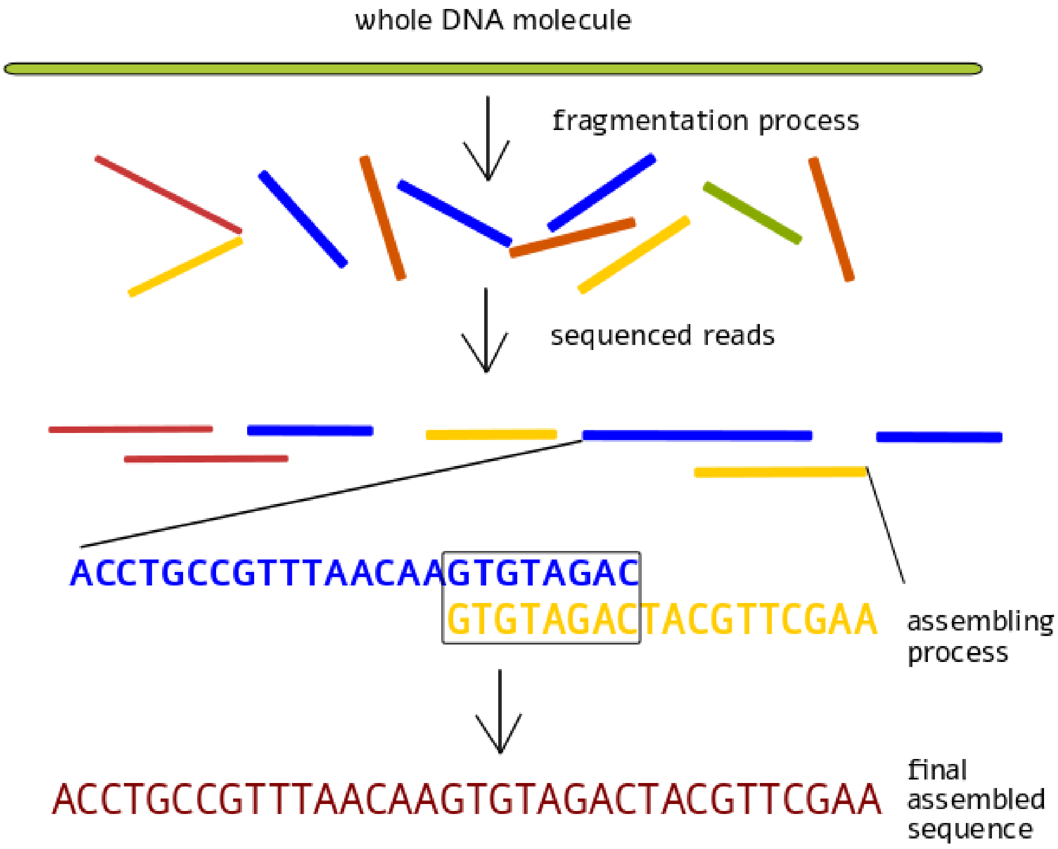

Figure 1.5: Summarization of the sequencing process illustrated on an artificial example. A molecule is broken into smaller fragments, then they are sequenced and afterwards they are assembled by assembling algorithm.

Chapter 2

CNV detection tools

In this chapter we point out what important factors have to be taken into account when comparing different software tools. We introduce various tools that detect Copy Number Variations(CNV) and desribe methods that were used for each tool.

2.1 Comparison of the CNV tools

Before we compare any CNV detection tools, we have to describe what elements affect the detection itself. Overall, detection of any microdeletion/microduplication syndrome is limited by four main factors [49]:

- fetal fraction,
- size of the particular CNV,
- coverage,
- biological and technical variability of the event region.

Fetal fraction is the proportion of cell-free fetal DNA (cffDNA), which refer to fetal DNA that circulates freely in the maternal blood. The time elapsed since the conception, maternal weight and other factors greatly affect the proportion of fetal fraction [25]. Naturally, when detecting any given MD, we want fetal fraction to be as high as possible. However, some factors might decrease this fraction [25].

Size of the CNV depends on a particular patient. Again, we can expect better results for bigger CNVs because they are easier to detect. Yet, we will not neglect analyses on data with small CNVs.

Coverage: Results from data with higher coverage can be more precise and sensitive, however, lower coverage is overall cheaper and faster. In contrast to fetal fraction and length of CNV, this factor can be directly changed to obtain more accurate results, but with higher production cost. In this study we analyze data with a very small coverages from 0.05x to 0.5x, that are usually in NIPT and similar tests.

Biological and technical variability of the event region refers to the fact that some sectors can be more variable than others. It can be caused by various factors such as repetitive elements, mapping ability and so on. As a result these regions are harder to detect and are usually filtered out from the analyses.

2.1.1 Detection of CNVs

Tools for CNV detection share some similarities in their approaches. Usually, there are four main steps. The reads of target sequence are separated into smaller sections called bins. Subsequently, normalization and noise correction techniques are applied and finally, normalized signal is segmented and scanned for CNVs. The scanning process is often referred to as CNV-calling.

Binning

Most CNV detection tools separate the sequence into smaller segments, bins. This process is called binning. Bin-size is the number of bases inside the bin. Although this parameter can be often adjusted by the user, some tools propose a method to determine the optimal bin-size. Final resolution strongly depends on the bin-size. Larger bin-size results in worse resolution and faster computational time, however, the sensitivity for smaller aberration decreases.

Term bin-count is the number of reads that are that are in the bin for a particular sequence. This number varies between different genomic regions due to different mappability and biological reasons. Thus it needs to be normalized to obtain bin-count purified from these biological biases.

Normalization

One of the most important steps is normalization. In this process program adjusts measured values to a hypothetically common scale. Process of normalization is very important since it reduces noise commonly seen in samples and therefore can considerably change final sensitivity and specificity. Normalization helps to highlight the aberrant segments as is shown in Figure 2.1.

Many systematic biases arise from whole genome NGS (next generation sequencing) data, which are more susceptible to create greater sequencing noise. In our work we included four CNV detection tools, each of them has to deal with one particularly important bias called GC bias. This bias is caused by different GC content (described

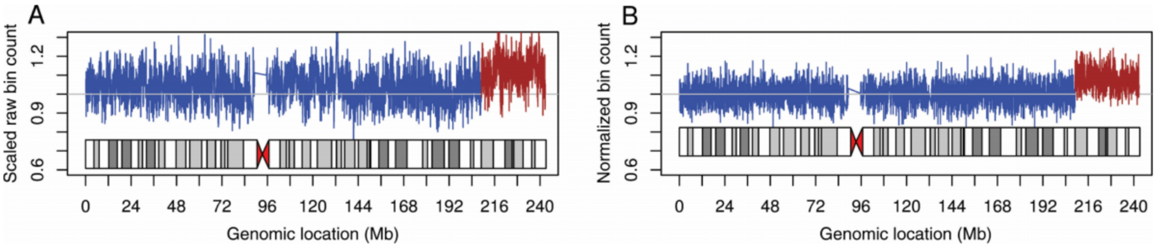

Figure 2.1: Bin counts of a particular chromosome (before and after normalization). Red color is the region region with microduplication, blue color refers to regions without abnormal microduplication or microdeletion, red triangles are centromere regions (part of the chromosome, which has a low mappability). The x-axis is the coordinate of each bin in Mb. The y-axis in picture A is scaled bin count and in picture B it stands for normalized bin count, where the expected bin count is 1 for the blue region. This picture shows how normalization process helped to detect the region with microduplication (red region), by reducing the sequencing noise [49]. In section A the region may seem like a duplication or deletion and thus is hard to identify, however after a bin-count normalization method is applied, as is shown in section B, the region becomes more evident.

below) across the genome.

GC content is the proportion of guanine and cytosine bases in DNA. The content of these bases can aggravate the sequencing process of the genome. Presence of regions with poor or rich GC content lead to uneven or no coverage of reads across genome. This inclination is called GC bias [14]. Local regression or local polynomial regression (LOESS) ic commonly used to deal with GC bias [8]. Loess regression merges together bins with similar GC content in a certain interval. This correction is applied to every bin and depends on the average read depth value of the bin and median read depth value for the whole genome. The GC bias is different for each sequenced sample and thus need to be dealt within the sample.

The mappability bias is another systematic bias that affects the results of any CNV detection tool. Mappability of a region is the chance that read will be sequenced and mapped successfully to the reference. However, particular regions are very hard to sequence and some regions (e.g. repeating regions) are challenging to map thus these regions will have a very low mappability. Mappability bias affects handling of ambiguous reads. Alignment of theses reads depends on what approach is used. Reads can be aligned to best-scoring positions, randomly assigned to any possible positions or even to all possible positions [16].

The DNA structure can differ between sub-populations and population [46]. This relation is called population stratification and it can results in spurios disease studies.

Principal component analysis (PCA) normalization is often used to remove this kind of bias. This is a reduction method for large-scale datasets. The aim of this method is to reduce the given data set while still preserving as much important information as possible. Firstly data has to be standardized, this way each variable contribus to the analyses equally. Next, covariance matrix is computed and finally principal components are obtained. Principal components are linear combinations of the original variables [46].

Segmentation

Another major step in CNV prediction is segmentation. The aim is to gather read depth signals with similar intensity, following this, the bin-count that deviate from the average read-depth signal are considered to be variant regions. However, read-depth signals are noisy due to different aspects such as different mapping ability of the tested sample and reference genome. As a result variant regions may be falsely identified. Crucial process in this step is distinguishing the spurious variants from true copy variants.

Circular binary segmentation (CBS) is a popular method used for segmentation. Vast majority of CNV detection tools that we compare in this work use CBS. This method is a recursive algorithm, that separates bins $x_1, x_2, ..., x_n$ into a smaller sets with similar read-depth values for each segment [16]. The algorithm recursively searches for an interval $i, ..., j$ where segments $x_i..., x_j$ have similar read-depth value and at the same time this value is different from $x_1..., x_i - 1$ and $x_j + 1..., x_n$ regions. If no such region is found, than the whole $x_1..., x_n$ region has no change point, otherwise three intervals are distinguished: $x_1..., x_1 - 1$, $x_i..., x_j$ and $x_j + 1..., x_n$. A region is consideret to contain a CNV aberration if a result from a certain statistic for that particular section exceeds a computed threshold.

2.2 Individual tools

In this section, we will introduce particular tools for CNV detection. Advantages and disadvantages will be pointed out and in addition we will briefly describe the process of CNV detection for every tool.

2.2.1 iCopyDav

Authors of this tool were focused in handling some of the systematic biases that rise from CNV detection in whole genome NGS (next generation sequencing) data. Framework comprises of five main steps: Data pre-treatment, Segmentation, CNV calling, Annotation and Visualization [16]. We will describe each step in better details bellow.

Data pre-treatment

At the beginning user can define desired bin-size. For low coverage data 500bp are recommended. However, this value is recommended for coverage around 3x, since our testing data can have coverage as low as 0.5x we decided to use same bin size for every tool (20 000bp).

To continue, this step copes with non-uniform coverage of reads that result from **GC content bias** and **mappability bias**. GC-bias correction is carried out by Local Polynomial Regression fitting (**Loess**) algorithm described in 2.1.1.

This tool deals with mappability bias, by scoring the individual bins according to their mappability. The score can reach values from zero to one and is computed along the genome. Subsequently, user can define the threshold for mappability score, meaning that no mappability bias is allowed if the value is set to zero and opposite to this no mappability correction is done if the value is set to one.

Segmentation

CBS is predominantly used for identifying larger CNVs. For smaller aberrations a different method is applied, which results in wider range of CNV prediction. Since we used only CBS during the execution of this tool, we will not describe the other mentioned method.

Variant Calling

In this step copy number regions are detected and start and end positions, type of variant and absolute copy number are reported. Absolute copy number estimates the credibility of a particular CNV. Variant calling is done by computing mean RD value of the chromosome that reported CNV. This value is not affected by deviant RD signals that originate from variant regions, since these segments are ignored. Subsequently, upper and lower threshold is computed using this value.

$$UpperThreshold(UT) = 1.45x(averageRDvalueofthechromosome)$$

$$LowerThreshold(LT) = 0.55x(averageRDvalueofthechromosome)$$

If RD-value exceeds UT that it is reported as duplication and opposite to this if the value is lower than LT than it is identified as deletion.

Results

This tool proved to be reliable in detection of CNVs larger than 1 Kb with coverage from 10x and higher (90 % occuracy was reached). However, for smaller CNVs the occuracy was around 50 % even though the sequencing depth was more than 30x. This indicates that iCopyDAV detects large CNVs with better precision than smaller ones [16].

To continue, iCopyDAV detected start and end positions better when sequencing depth was increased. This is naturally due to higher resolution. Large variation in detection of these positions is detected for cases with low coverage data (less than 30x). In addition authors state that in low coverage data duplications are easier to detect than deletions.

This tool is not originally meant for such low coverages as we are dealing with and it is likely to have problems detecting CNV on our datasets.

2.2.2 WISECONDOR

One of the most promising tool in our research seemed to be WISECONDOR [43]. Authors presented some novel approaches that reduced the testing costs.

WISECONDOR took inspiration from z-score method developed by Chiu *et al* [38]. It was used for detection of trisomy 21 (Down syndrome) and their goal was to develop high-resolution version of this approach by testing different alterations.

For normalization authors applied **within-sample comparison** method. This approach compares the read counts within the tested sample of each genomic region with regions on other chromosomes that behave alike in control samples. Regions with similar characteristics will behave in a similar way within a test sample, since they are exposed to the same experimental procedure.

In order to calculate sub-chromosomal scores, they developed and combined *individual bin method* and *sliding window method*, we describe both these methods bellow.

Individual bin method applies GC-correction for each target bin in test samples. After this, it calculates the z-score (standard score) from mean and standard deviation. They improved sensitivity by ignoring bins with aberration in the reference set for every target bin. Because excessive calls close to each other could slow-down the program they accepted small gaps around the detected part and work with this region as with one whole aberration.

The socond method for calculating sub=chromosomal scores is called **Sliding window analysis**. Sliding window approach takes into account the z-scores of the bins neighboring the target bin. For the detection they use Stouffer's z-score:

$$z_i^w = \frac{\sum_{k=i-v}^{i+v} z_k}{\sqrt{2 * v + 1}}$$

The symbol z_i^w is the z-score of each sliding window for particular i-th bin, this bin takes into account v bins on the left and right side of the processed bin i. After this following calculation is applied, which informes whether the region contains some kind of aberration or not. If the $\mid z_i^w \mid \geq 3$, than the bin is considered aberrated.

Results: Authors used samples from pregnant woman where fetal fraction was at least 5%. Their tool was tested for different bin sizes, specifically 1Mb, 500kb, 250kb and 100kb. The worst results were on smaller bins where their program suffered strong variations and noise. On the other hand mega base pair showed promising and stable results.

Sex chromosomes X and Y had to be omitted due to strong correlation with the percentage of fetal DNA, even though they stated that detection of subchromosomal disorders could be possible for chromosome X with more reference sample, chromosome Y remained challenging due to small size and repetitive sequence.

They achieved good results, when detecting deletion of 30Mb and unbalanced translocation. One sample could not be identified due to the combination of low fetal fraction and low sample coverage. WISECONDOR had hard time detecting aberrations on chromosome 19, although they did stated that this area has a high GC-content and is overal harder to analyze.

To sum everything up WISECONDOR did not show satisfying results on triploid and mosaic samples at low sequencing. Other than these mentioned samples this tool detected subchromosomal and chromosomal disorders.

Some disadvantages of WISECONDOR are its exclusiveness for NIPT and its extremely slow run-time for small, yet realistic bin sizes such as 15kb, where the program took 24 hours to finish [39]. We describeat their improved version WisecondorX bellow.

2.2.3 Wisecondor X

Wisecondor X is an improved version of WISECONDOR, which we described earlier. The need for WisecondorX comes from a fact that WISECONDOR could only work with NIPT and was also too slow for real life usage. With WisecondorX authors claim that they have altered some of the WISECONDOR limitations, yet they kept same normalization principles.

This tool uses 100kb bin-size for NIPT samples.

With Wisecondor X, gender is automatically predicted, which means the tool evaluates male and female samples separately. For gender prediction WisecondorX uses Gaussian mixture model, which is a probabilistic model. The tool generates reference for non-sex chromosomes and two additional references for both genders. When the tool scans new sample it chooses the correct reference and afterwards it is sent for normalization.

In the original tool authors used Stouffer's z-score method for segmentation. This method was replaced with Circular Binary Segmentation method described in 2.1.1. DNAcopy (v1.50.1) R package was used and this modification lowered computation time and made Wisecondor X usable for analyses beyond NIPT.

During reference creation, bin-wise values from healthy samples are saved into reference-matrix for later use in calculating z-scores.

When it comes to aberration calling, the authors follow this methodology: statistical significance is unrelated to the interest of type of analysis, meaning that even a small aberration from a healthy should be studied and observed. Therefore, user can define a cut-off for aberration calling.

Normalization process is very similar in both tools. Normalization method used in WISECONDOR prooved to be reliable enough.

Owning to the fact that this tool kept normalization approaches used in WISECONDOR and in addition is faster than its precursor, we expect this tool to be reliable and usable in real life.

2.2.4 CNVkit

Another tool we decided to include is CNVkit. Their approach is based on the fact that some regions are sequenced at higher coverage and thus they are creating biases related to CNV prediction. Massive parallel sequencing proved to be useful for CNV prediction by analyzing the read depth in sequencing data. For clinical use some genome partitions, for instance disease-relevant genes, are either re-sequenced or higher coverage is desired. However, other non-coding regions are neglected and as a result some CNVs (especially larger CNVs) are not detected.

Two main inputs that CNVkit takes are set of test samples and normal samples, from which a reference will be created. CNVkit calculates bin-size specifically for off-target (non-coding) regions. Normal samples go trough this process as well.

To continue read depth is calculated. Read depth of each base pair in every bin is summed and subsequently this value is divided by the size of the bin to calculate mean read depth of bin. Following this step, bins are log2 transformed.

Reference is created as well from normal samples in order to produce reference copy-number profile, which is used in correction of test samples. CNVkit calculates bins with systematically high or low coverage. In addition, bins where the coverage could not be computed are identified and de-emphasized in following steps.

This tool accept input with no control samples, thus it can work without reference from normal samples. However, reference genome such as hg19 should be provided. Created reference can be reused, which decreases detection time.

Next step is to normalize test samples to the reference. Bins that do not meet predefined criteria are removed, the rest undergoes bias correction (we will describe this step bellow). After further modification bins are weighted according to their size, if control samples are provided weight is further modified.

Since read depth is affected by various systematic biases a correction process is needed.

Repetitive sequences can cause various systematic biases, since regions where they occur indicate high variability in coverage. In reference genome repetitive regions are masked out and the proportion of masked bins is later used in bias correction. The bias correction procedure assigns an expected bias for each bin. We have to point out that if one or more normal samples are given for reference creation, systematic biases are largely removed by normalization. Still, some biases persist and bias correction has to be done.

This tool used the same segmentation algorithm as iCopyDAV: CBS (Circular Binary Segmentation). Next step is calling the absolute copy number, where for each segment absolute integer copy number is computed using a list of thresholds.

Finally, gene-level copy number information can be extracted with a special command.

CNVkit offers an easy usage pipeline tool. One command can create reference and simultaneously detect copy number from given test samples. In addition, this tool generates its native BED-like format, which has a strict form (additional parameters can be added) thus it is easy to read and analyze. However, it proposes conversion to other formats that are supported by other software tools, such as SEG format used by GenePattern [40], VCF format etc.

To sum everything up, CNVkit is easy to use since for a whole CNV detection the user has to run only one command.

2.2.5 CNVcaller

Finally, last tool we describe is CNVcaller.

Similarly as in CNVkit, GC normalization (correction) was applied 2.1.1. Subsequently bins were normalized using Principal component normalization described in 2.1.1.

Bins with low read coverage, bins from unmappable regions and bins with high variance or read coverage bins were filtered out, yet keeping at least 88% of bins. Finally, to obtain data normalized around zero, mean read count was subtracted from every bin.

Segmentation was done by CBS (Circular Binary Segmentation) to identify same-coverage segments. However, since this procedure partitions a chromosome excessively, following rule was applied to determine significance of a segment. Ideal deletion or duplication is a decrease or increase by a factor $(mb * ff/2)$, mb is a mean bin count and *ff* is fetal fraction. Segment was marked as significant (it carried fetal aberration) if it overstepped 75% of the ideal increase/decrease. Segments classified around 100% of the fetal fraction were classified as maternal in origin.

Authors carried a sensitivity test, to confirm the tools credibility. Sensitivity test was carried out with 200 different NIPT data sets and on 533 pathogenic regions. For samples where fetal fraction reached at least 10%, 99.6% sensitivity was measured, which implies that this tool might be successful in our analyses.

Chapter 3

Snakemake

In this chapter we characterize software that we used to compare CNV detection tools. We describe advantages of each software and reasons on why we choose to use each individual workflow.

In this thesis we defined four different CNV detection tools: CNV-caller, CNVkit, iCopyDAV, WisecondorX, each of them is a command line tool. However, working with command line can be protracted, impractical and often chaotic, especially if chained execution of multiple commands or applications is requested. Since this was our case, we decided to search for a workflow engine that would simplify the execution of different CNV tools. Snakemake [26] turned out to be the perfect tool for this. It is easy to use, it provides various effective features and can be run on a remote server as well as on any personal computer.

Snakemake is a Python-based environment that helps to automate pipelines. No graphical environment is needed, therefore workflows can be assembled directly on a remote server. This software can process commands from any installed tool, if input and output files are well-defined. In addition, Snakemake workflow supports file names with multiple wildcards.

3.1 Structure of the Snakefile

Snakemake workflow is defined in a Snakefile, which is essentially an extension of the Python language with defined structure for **rules** [26]. Each rule has a name, although an anonymous rule can be created. Rule consists of input, output and a shell or run blocks that are defined by user. **Input** and **output** blocks can contain one or more input files, where user can name each file individually. In the **shell** block user can invoke any tool or service that is installed or available, including basic shell commands. The three mentioned blocks are required in order to create a rule, however more additional pre-defined blocks are available.

Individual rules can be given a number of threads to use. This is done by creating the additional block called **threads**. It is not recommended to use more threads than there are cores available, to avoid excessive context switching.

Wildcards are a very useful feature of this software. They are used to generalize rules inputs and outputs so that they are applicable to a larger number of datasets. For instance, if a user wants to apply the same command on a file A.bam and B.bam, two separate commands has to be typed down in command line. However, in a Snakefile, user can define a rule with inputs samples{character}.bam, which is subsequently replaced by the regular expressions.

File names can be obtained from different data sets as well. For instance, user may create a list of file names through a Python script and then use it in rules.

3.2 Snakemake engine

The order in which Snakemake executes individual rules, is represented by directed acyclic graph (DAG) [26]. Nodes are executions of the particular rules. Directed edges between nodes determine the order of rule executions (jobs). For instance if a directed edge exists from node A to node B, it means that in order to execute job A an output from job B is needed.

Target rule represents the node that is the root of the DAG. The user can specify a target rule or created a rule named *all*. By default, the first rule in the Snakefile is executed. An example of a snakemake workflow is in Figure3.1.

Snakemake rule is executed only if the stated output files are not generated yet or if the input files have been modified. This quality is very useful for our work since it allows us to train any CNV tool and subsequently run it on various test samples without creating the reference again.

To analyze the workflow, a parameter *–dryrun* can be added in the command line. This command will print the jobs that are going to be executed, yet without the actual execution. Dryrun is crucial to perform before running the Snakefile on large-scale data.

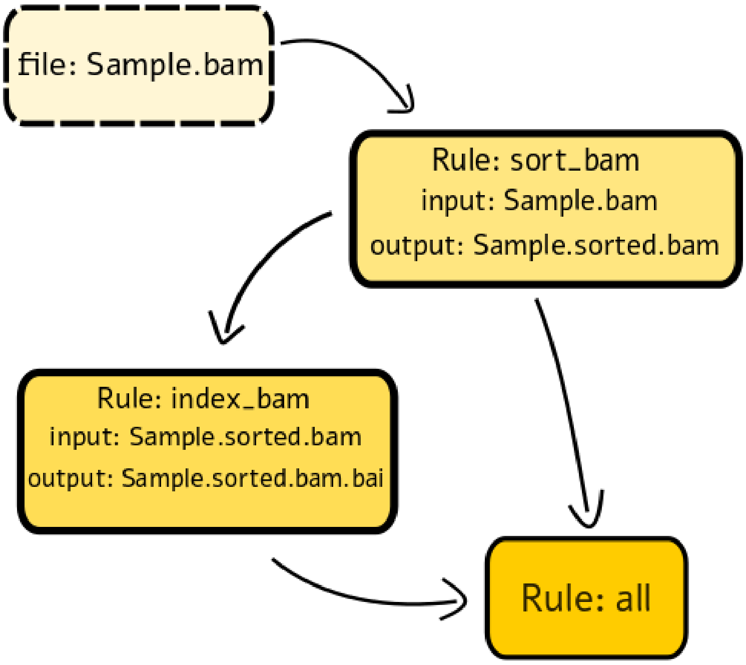

Figure 3.1: An example of a DAG for an artificial Snakefile. The program needs Sample.bam as main input and proceeds to create additional Sample.sorted.bam and Sample.sorted.bam.bai files.

Chapter 4

Results

In this section we present final results of our comparison. Advantages and disadvantages for each tool are pointed out and described. In addition, we briefly describe the comparison process, data preparation and other important details.

4.1 Data preparation

One of our goals was to discover the relation between detection accuracy and parameters such as fetal fraction and CNV size. Three sets of samples were used for this analyses: training samples, mixed data samples and normal NIPT samples. All obtained samples are from patients, who signed consent with the study.

4.1.1 Training samples

CNV-caller, WisecondorX and CNVkit use a constructed reference for CNV detection. For the reference creation process 134 samples were used.

Training data were collected from standard NIPT samples, originating from confirmed genetically healthy pregnancies. In NIPT, the prediction of any syndrome is affected by the number of fetuses thus we included only samples from singleton pregnancies [45].

4.1.2 Mixed samples

To analyze the affect of fetal fraction and CNV length, samples with different values of these factors were needed. NIPT data with confirmed microdeletion or microduplication are sparse. Therefore we used data mixed in the laboratory.

Since these data should resemble data obtained by NIPT testing, both maternal and fetal parts had to be assembled to create this data set. Plasma DNA from non-pregnant female volunteers was used for the maternal part. This material was subsequently mixed with male DNA with confirmed microdeletion syndromes. This way we were able to control the level of fetal fraction.

The selected microdeletion syndromes were: DGS-DiGeorge syndrome (chr 22), AS-Angelman syndrome (chr 15), PW-Prader Willy syndrome (chr15), CDC-Cri du chat syndrome (chr5), WHS-Wolf Hirchhorn syndrome (chr4), 1p36-1p36 (chr1).

Total number of obtained mixed samples is 19.

4.1.3 Normal NIPT samples

We gathered 35 NIPT samples with confirmed 41 CNVs. Maternal CNVs were processed as well (11 CNVs). An average percentage of fetal fraction for these samples is 14%. For each CNV fetal fraction and the approximate start/end position was stated.

All samples were sequenced by Illumina Nextseq [2]. Subsequently, reads were aligned by Bowtie2 [29] to a reference genome hg19 [17].

4.2 The Comparison of CNV tools

In our work we researched the affect of three main factors on CNV detection: Fetal fraction (ff), size of microdeletion/microduplication and coverage. We have selected samples with different values of the first two factors to demonstrate its affects.

We have constructed a table to demonstrate the comparison results. Each row represents one aberration, its CNV size, read count and percentage of fetal fraction. Each tool occupies one column, where its stated whether the tool detected the given sample variation or not. Results are shown in Table 4.1.

Additionally, we have calculated success rate for each tool along with its accuracy rate for positions for both fetal aberrations and maternal aberrations. To obtain this number we calculated average start/end position from CNV-caller, Wisecondor and predicted position of the particular aberration. If the tool did not detect a particular CNV we left it out. Subsequently we subtracted the detected start/end position from the mean position. We used absolute value of this number and divided it by two (since we have start and end positions).

4.2.1 CNV-caller

The best results yielded CNV-caller. The tool detected 16 CNVs from 19 testing Samples 4.1. This tool, as well as other tools, was tested on additional NIPT samples, where overall success rate was 92%. Additionally, all maternal CNVs were detected. Results are shown in Table 4.2.

Accuracy of positions is shown in Table 4.3.

Size	coverage	ff	CNV-caller	WisecondorX	CNVkit	iCopyDAV
0.9Mb	19.24M	5.90%	N	N	N	N
	20.36M	11.50%	**D**	**D**	N	N
	21.52M	17.30%	**D**	**D**	N	N
2.6Mb	19.6M	8.70%	N	N	N	N
	14.54M	16.69%	**D**	**D**	N	N
	19.45M	17.30%	**D**	**D**	N	N
3Mb	25.27M	4.50%	N	N	N	N
	20.59M	11.20%	**D**	N	**D**	N
	20.39M	20.10%	**D**	**D**	N	N
5.3Mb	15.3M	7.30%	**D**	N	N	N
	24.3M	13.40%	**D**	**D**	**D**	N
6Mb	19.31M	10.60%	**D**	**D**	N	N
	19.26M	14.60%	**D**	**D**	**D**	N
9.3Mb	8.3M	9,10%	**D**	**D**	N	N
	8.4M	14.10%	**D**	**D**	N	N
	16.2M	16.40%	**D**	**D**	N	N
17.7Mb	16.47M	5.11%	**D**	N	N	N
21 Mb	15.9M	4.86%	**D**	N	N	N
	21.5M	9.85%	**D**	N	N	N

Table 4.1: Comparison of CNV detection tools on mixed samples. Table is sorted by the size of CNV and subsequently by percentage of fetal fraction. For each tool letter D (Detected aberration) or N (Not detected aberration) is present.

Tool	Success rate (SR)	Fetal CNVs SR	Maternal CNVs SR
CNV-caller	92%	80%	100%
WisecondorX	60%	56%	73%
CNVkit	46%	16%	100%
iCopyDAV	25%	16%	70%

Table 4.2: Overall comparison of success rate for CNV detection tools. Overall success rate includes NIPT samples and mixed data samples.

Tool	**Overall**	**Mixed data**	**normal NIPT**	**Fetal CNVs from NIPT**	**maternal CNVs from NIPT**
CNV-caller	1930	1310	2118	1933	2591
WisecondorX	3493	625	3183	3293	3218
CNVkit	35696	16388	49829	36292	40522
iCopyDAV	13214	-	27428	3062	13214

Table 4.3: Accuracy rate for positions for CNV detection tools (in *kb*). Each number represents the average deviation from an expected position. For instance if WisecondorX has a position accuracy of 3493 *kb*, it means that on average the actual start/end position of the aberration would be located 3493 *kb* away from the detected position. Naturally, the smaller this number is the better precision the tool has. For each tool various rates from different data were calculated. Position accuracy for iCopyDAV from mixed data was not calculated since this tool failed to detect any CNV on this data set.

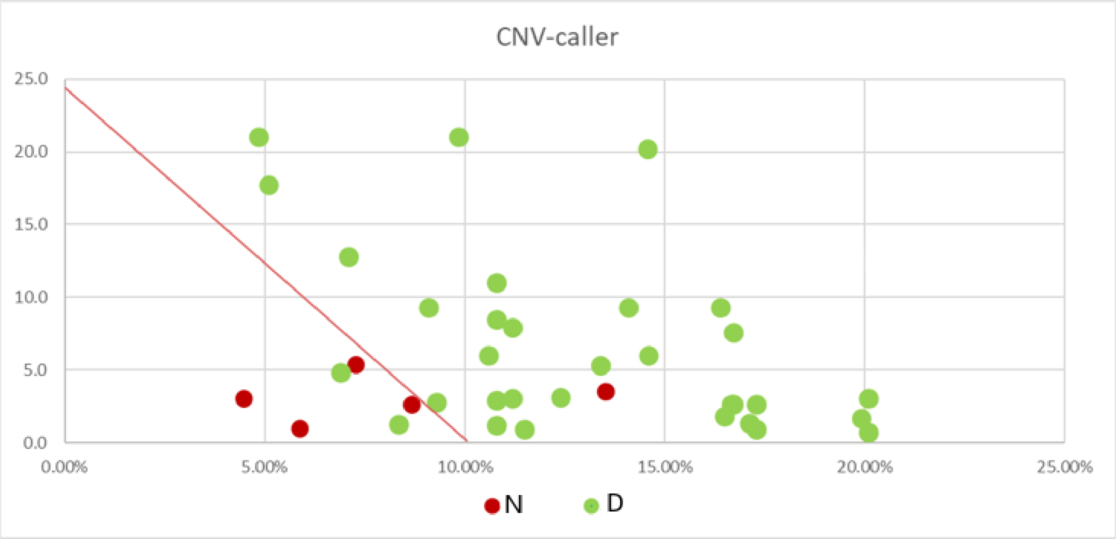

Figure 4.1: The effect of fetal fraction and CNV size on detection of CNVs for CNV-caller. The X-axis represents fetal fraction in percentage and Y-axis represents size of CNV in *Mb*. The green dots represent detected samples and red not-detected samples. Red line is separating the detected and not-detected samples. Detection for low fetal fraction (less than 10%) in combination with small CNV size (less than 8Mb) remained challenging for this tool.

The detection process for one sample took approximately one minute. However, computation time for each tool varied according to different data preparation. Tools often convert input files to a different format, which is not detection itself, but this process can be lengthy.

This tool could detect samples with lower fetal fraction (less than 5%), however samples with both low fetal fraction and small CNV size remained challenging. Generally, tool could deal with smaller CNVs better than with really low fetal fraction. Overall results for different levels of mentioned parameters are shown in Figure 4.1 .

Advantages: In the final output, CNVs where highlighted by color scale, which indicated the confidence of the detection. Naturally, the red/magenta colors indicated unambiguous CNV region with its start, end positions. Therefor the user knew exactly the level of confidence for the given CNV. The tool is also able to interpret the results in *.png* format for every chromosome, which is very helpful.

Disadvantages: This tool generates many different outputs, which may be confusing at first. Additional column that would state clearly what type of event was detected would be helpful.

4.2.2 WisecondorX

Following CNV-caller, WisecondorX performed second best. This tool successfully detected 11 out of 19 mixed samples. Overall success rate calculated from both mixed samples and NIPT samples is 60%. WisecondorX was one of the fastest tools, although training process took approximately two days, the detection itself was relatively fast. It took approximately an hour to detect 54 samples, still as mentioned earlier, data preparation took more time.

The detection of this tool was greatly affected by low fetal fraction as well. As is shown in Figure 4.2, samples with fetal fraction under 10% were hard to detect.

Accuracy of positions is shown in Table 4.3.

Advantages: This tool is relatively easy to install and run. The manual for it is very clear and exactly specifies which additional software are required for this tool. All outputs and their format are described in the manual, which is essential for making Snakefile with specific outputs. Detected CNVs are stored in *.bed* files, which are standardized files with strict structure, which makes the results easy to understand. Moreover, user can set a parameter to generate .png plot as well.

The information about the final result is usually divided into four files, this way the user is not overwhelmed by too many additional information and can choose what is needed for their analyses. For the purpose of this thesis file ending with *aberrations.bed* contained the main information: start, end positions of the CNVs, type (dup/del) and z-score, which was very simple and practical during comparison.

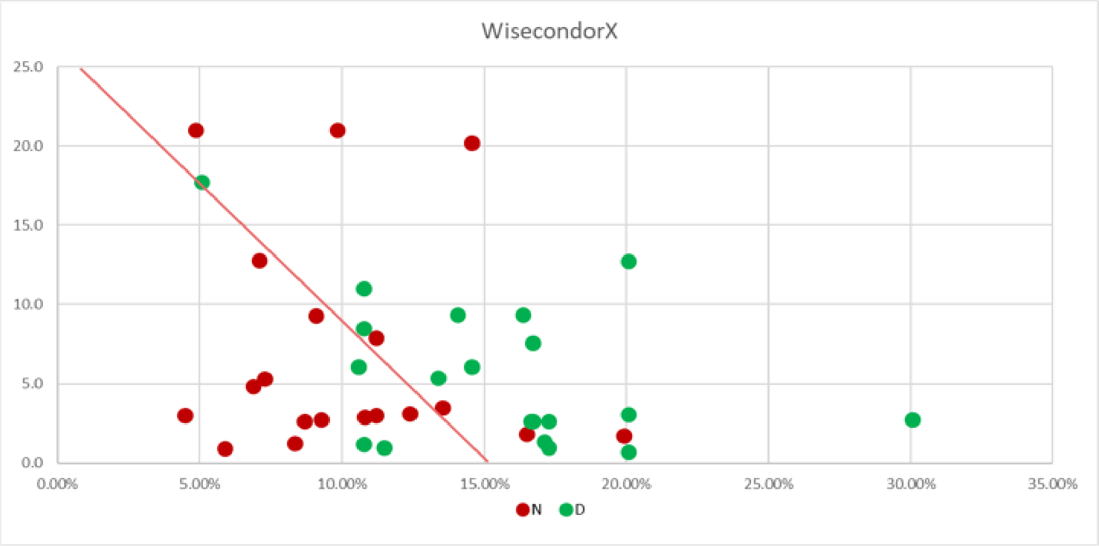

Figure 4.2: The effect of fetal fraction and CNV size on detection of CNVs for WisecondorX. The X-axis represents fetal fraction in percentage and Y-axis represents size of CNV in *Mb*. The green dots represent detected samples and red not-detected samples. The red line roughly shows the turning point from which the samples were either detected or not detected.

Disadvantages: The documentation for this particular document turned out to be slightly chaotic since WisecondorX carries many features from Wisecondor. The user has to assemble information from both tools, which can be confusing.

4.2.3 CNVkit

This tool along with iCopyDAV did not perform well in comparison to CNV-caller and WisecondorX. The success rate was only 16% for fetal aberrations and 46% for all samples as is shown in Table 4.2.

The detection was relatively fast, approximately 2 minutes per sample. Naturally, the training process took about two days, but this process is lengthy for every tool in our work.

However, it is important to note that the success rate on maternal aberrations reached 100% . This indicates that even though this tool performed poorly in detection of fetal aberrations, it is still efficient for normal CNV prediction.

The effect of fetal fraction and CNV size was hard to determine. Low coverage could led to spurious results. However, with increasing CNV size and percentage of fetal fraction a slight improvement can be detected 4.3.

Advantages: Documentation for this tool was well arranged and simple. Used

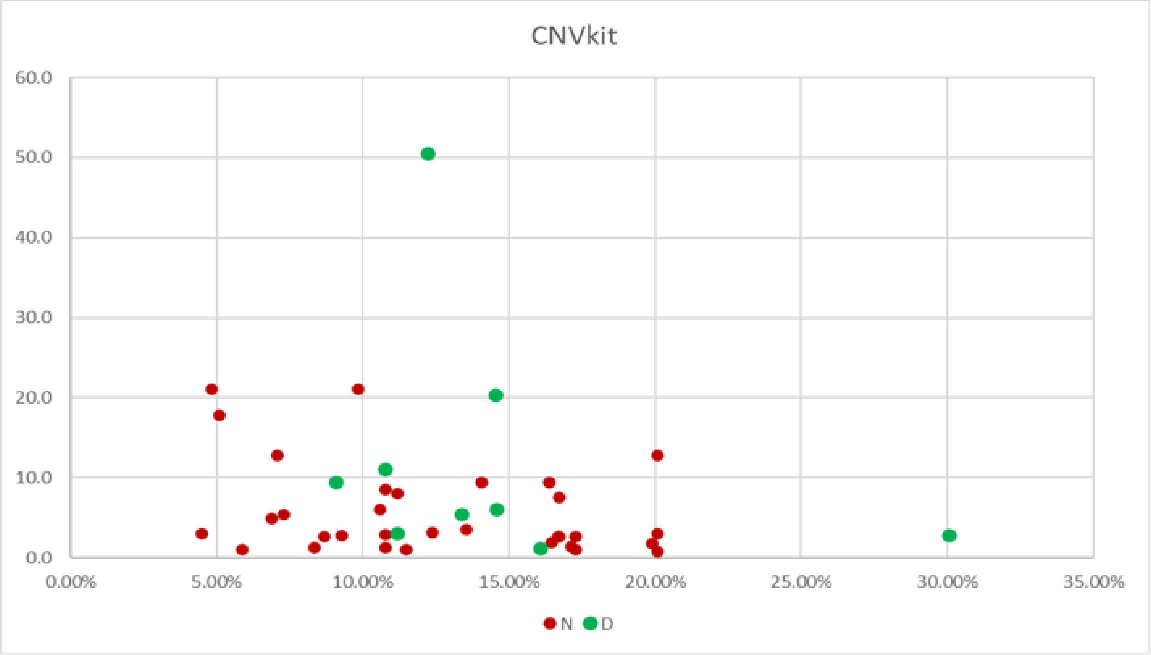

Figure 4.3: The effect of fetal fraction and CNV size on detection of CNVs for CNVkit. The X-axis represents fetal fraction in percentage and Y-axis represents size of CNV in *Mb*. The green dots represent detected samples and red not-detected samples. For average fetal fraction (15% - 20%) and average CNV length, the red and green dots are mixed together.

methods were not described in too much details and the user could easily understand the workflow. The activation of this tool was remarkable easy. The user has to run only one command for both training and CNV prediction.

Disadvantages: Understatement of the manual was much more complicated than in other tools. It was hard to find out what output files will be created during the prediction process. After running the main command for detection, user has to explicitly extract the desired format, but in the manual there are no clear explanations of what information each format stores.

4.2.4 iCopyDAV

This tool turned out to be useless for the comparison of mixed samples, since did not detect any CNVs. There may be several reasons for this (we address them below). However, overall success rate turned out to be 23%, which means that some aberrations were detected and thus we launched the detection correctly. The computation time was very long, more than four hours for a sample. Although the segmentation and cnv-calling itself ran under a minute, data-preparation for each chromosome took 15 minutes on average. iCopyDAV does not use any reference from healthy samples, but for each sample data-preparation still has to be done. In addition, this tool does not require any reference samples.

Accuracy of positions for fetal aberrations was not good either. This tool could not detect the whole aberration. This could be a results of low coverage. The results for this parameter are shown in Table 4.3.

Advantages: A great advantage of this tool is its documentation and manual. They were the most straightforward among other tools. Even though, the documentation was extensive it provided very clear and simple explanation for each basic normalization method and systematic bias. This was very advantageous for this thesis.

The manual contained exact information for each command. For each command a list of needed input files and expected output files was presented. The description of parameters was slightly confusing. However, a demonstrative example for each command was a great help during Snakefile creation. In addition, structure of every output file was shown, this way we could verify our results.

Accuracy of positions for mixed data could not be compute since no CNVs were detected. However, iCopyDAV managed to detected some of the fetal and maternal aberrations. Results are listed in Table 4.3.

Disadvantages: Although this tool has a great manual, the commands themselves were rather complicated. For instance, the user could not adjust the name of the output in-between steps. Almost every command needed a prefix of the requested input file in order to find the right file according to the suffix. This led to crammed folders and inconsistency.

To continue, this tools separates detected CNVs for each chromosome, which is inconvenient when whole genome analyses is needed. It does not create only 24 files because for each chromosome there are three files generated during the detection process. The fact that user can not modify the path of the output file arbitrarily complicates the whole process.

Output is stored in *.bed* files. Deletion is presented as 0 and duplication as 1. However, it would be more clear if deletions were stated as *loss* and duplications as *gain*. WisecondorX has the same output file (.bed), where it used this description.

iCopyDAV limitations: We suppose that low coverage was the main reason that this tool did not perform satisfyingly. Deeper research showed that this tool was unable to yield good results for the coverage under 1x [24]. An average coverage of our data is greatly under 1x (from 0.1x to 0.5).

Another complication that arose during the execution was that some samples failed to be processed by this tool. What is even more interesting is that for instance data-preparation and CNV detection for a chromosome 1 in sample A ran without an error, yet detection for the same sample, but different chromosome, it did not succeed. We left out these samples from analyses for this tool.

To sum everything up, we recommend to use this tool for analyses with higher coverage than 1x, this way the tool can function properly.

4.3 Combining predictors and future work

In this section we describe prediction model for CNV detection that would be based on combination of different CNV detection tools.

The results of this analyses showed that most reliable tools were CNV-caller together with WisecondorX. Combination of these tools could be used to improve the prediction itself.

4.3.1 Simple decision tree

If we simply wanted to increase the number of detected aberrations, we could made a very simple decision tree. There would be two options: An aberration is considered to be correctly detected if CNV-caller or WisecondorX detected a CNV, otherwise no aberration was reported. The probability that this prediction model will not detect an aberrant sample is 3,2%. This way we could seemingly catch more aberrations. However, the number of false positives would rise up considerably.

4.3.2 Decreasing the number of false positives

False positives are untrue aberrant detections, where in reality there are no CNVs. The number of these aberrations could be decreased by restricting the conditions for detection. Any aberration would have to be detected by both CNV-caller and WisecondorX, meaning we would double check any detection. However, CNV-caller performed excellently on problematic data while WisecondorX did not sense any aberrations. Therefore by choosing this approach, limitations for CNV detection would increase.

This approach could be further improved by adding an additional tool such as CNVkit. Naturally, proclaiming an aberration as detected only if it was found by all three tools would not yield satisfying results, since success rate of CNVkit was not high. Therefore instead of expecting three positive results, we would only need two from all mentioned tools.

Furthermore, an additional analyses could be done for a greater number of samples. This way we could find out limitations of each tool. According to this we could decide if the detected region is truly an aberration or if it was only a false positive caused for instance by small fetal fraction.

This way we could add an interesting parameter: confidence of detection. Although, tools already contain z-scores and confidence levels, this parameter would be more focused on percentage of fetal fraction. For instance if the tool is known to be incorrect for samples with fetal fraction under 10%, we would assign a lower significance for its detection.

Computation time would increase, since running all three tools would take more time. Still, WisecondorX and CNV-caller both require *.npz* files, which take long time to convert. We could hypothetically join this processes if the generated files are indeed the same. Segmentation process and CNV-calling is not as consuming as data preparations.

4.3.3 Recommended approach

Detection of microaberations from NIPT samples is a process that strongly depends on various factors (fetal fraction, coverage CNV size). These factors can disturb the correct estimation of CNVs. Therefore applying the first prediction model with the simple decision tree could result in many false CNV detections. In real life usage patients would be falsely informed, which may be unnecessarily unsettling.

On the other hand decreasing the number of false positives would solve this issue (the second proposed prediction model). This approach would result in lower success rate, therefore increasing the number of false negatives. Still the detected CNVs would be more reliable. Precision of the detected region would increase, which would help in identification of the particular syndrome or disease (for instance by comparing it to a valid CNV database). It could even uncover new or unexplored CNVs.

4.3.4 Future work

Although we managed to put together a comparison of several CNV detection tools, we address some important improvements that would enhance the quality of the results.

Comparing more than four tools would bring new insights into the analyses. With deeper study a more suitable tool could be found that would serve as a good oponent for other CNV detection tools. This way we could ended up with different ranking of the particular tools.

To continue, more training samples would definitely improve the detection accuracy. It is hard to determine the minimum number of training samples, but more reference samples will only improve the tools prediction capability.

More testing samples with different values of fetal fraction and CNV lengths would specify the limitations for each tool. This way the relation between these factors and the individual tool would be more relevant.

Therefore in the future we would like to try more extensive comparison, from which we would be able to compile more both sensitive and specific prediction model for CNV prediction.

Conclusion

The prediction of CNVs from NIPT samples can expose various structural variants that are linked to different types of diseases and syndromes. Even though, each CNV detection tool has certain limitations that are hard to overcome, combining these tools can result in more reliable and correct CNV detection that is applicable in real life.

We presented several different CNV detection tools. For each of them, we described their approaches in CNV prediction, explained the used methods and overall workflow. Practical and technical side was described for each tool as well. Although, some tools were not easy to execute and various errors had to be solved we managed to summarized different advantages and disadvantages and compile the overall comparison.

Different statistics from mixed samples and normal NIPT samples were presented. Tools were tested on samples with various values of fetal fraction and CNV size, to give a better perspective of the tool's limitations. In addition, we provided results for each tool on both fetal aberrations and maternal aberrations.

Finally, we proposed several approaches for combined CNV prediction that would result in improvement of correctness of CNV detection. We also addressed improvements that could be made to improve our comparison.

Our comparison points out the importance of fetal fraction and CNV size, reasonably describes benefits of each CNV tool and proposes a prediction model for better CNV detection, which may be found useful in future works, other CNV researches and analyses.

Bibliography

[1] Chromosome 9 inversion. https://rarediseases.info.nih.gov/diseases/10765/chromosome-9-inversion. [Online; accessed 03-May-2020].

[2] Introduction to ngs. https://www.illumina.com/science/technology/next-generation-sequencing.html. Accessed: 2020-05-30.

[3] Oxford nanopore technologies. https://nanoporetech.com/. Accessed: 2020-05-31.

[4] Solid® next-generation sequencing chemistry. https://www.thermofisher.com/sk/en/home/life-science/sequencing/next-generation-sequencing/solid-next-generation-sequencing/solid-next-generation-sequencing-systems-reagents-accessories/solid-next-generation-sequencing-chemistry.html. Accessed: 2020-05-31.

[5] Chromosome translocation. http://www.eurogentest.org/index.php?id=612, 2007. [Online; accessed 03-May-2020].

[6] Dna sequencing costs: Data from the nhgri genome sequencing program (gsp). www.genome.gov/sequencingcostsdata, 2019. [Online; accessed 25-May-2020].

[7] What is noninvasive prenatal testing (nipt) and what disorders can it screen for? https://ghr.nlm.nih.gov/primer/testing/nipt, 2020. [Online; accessed 30-April-2020].

[8] Can Alkan, Jeffrey M Kidd, Tomas Marques-Bonet, Gozde Aksay, Francesca Antonacci, Fereydoun Hormozdiari, Jacob O Kitzman, Carl Baker, Maika Malig, Onur Mutlu, S Cenk Sahinalp, Richard A Gibbs, and Evan E Eichler. Personalized copy number and segmental duplication maps using next-generation sequencing. *Nat Genet*, 41(10):1061-1067, 2009.

[9] M Baker. De novo genome assembly: what every biologist should know. *Nat Methods*, 9:333–337, 2012.

[10] Beat Wolf. De novo genome assembly versus mapping to a reference genome.

[11] Yuval Benjamini and Terence P. Speed. Summarizing and correcting the gc content bias in high-throughput sequencing. *Nucleic Acids Research*, 40:e72, 2012.

[12] Xu C, Wu K, Zhang JG, Shen H, and Deng HW. Low-, high-coverage, and two-stage dna sequencing in the design of the genetic association study. *Genet Epidemiol*, 41(3):187-197, 2017.

[13] C. P. Canales and K. Walz. *Cellular and Animal Models in Human Genomics Research*. Academic Press, 2019.

[14] Yen-Chun Chen, Tsunglin Liu, Chun-Hui Yu, Tzen-Yuh Chiang, and Chi-Chuan Hwang. Effects of gc bias in next-generation-sequencing data on de novo genome assembly. *PLoS ONE*, 8(4):e62856, 2013.

[15] Pareek CS, Smoczynski R, and Tretyn A. Sequencing technologies and genome sequencing. *J Appl Genet*, 52(4):413–435, 2011.

[16] Prashanthi Dharanipragada, Sriharsha Vogeti, and Nita Parekh. iCopyDAV: Integrated platform for copy number variations—detection, annotation and visualization. *Plos One*, 13(4):e0195334, 2018.

[17] Church DM, Schneider VA, and Graves T et al. Modernizing reference genome assemblies. *PLoS Biol*, 9(7):e1001091, 2011.

[18] ENCODE consortium. RNA-seq of thymus (Mus musculus, postnatal 0 day), 2014. [Cited 2020-04-29] Available at `https://www.encodeproject.org/files/ENCFF002EYX/@@download/ENCFF002EYX.fastq.gz`.

[19] L. Feuk, A. Carson, and S Scherer. Structural variation in the human genome. *Nat Rev Genet*, 7:85–97, 2006.

[20] National Center for Biotechnology Information (US). Genes and Disease. Bethesda (md): National center for biotechnology information (us). `https://www.ncbi.nlm.nih.gov/books/NBK22266/`, 1998. [Online; accessed 30-April-2020].

[21] Ottawa (ON): Canadian Agency for Drugs and Technologies in Health. Next generation dna sequencing: A review of the cost effectiveness and guidelines. `https://www.ncbi.nlm.nih.gov/books/NBK274079/`, 2014. [Online; accessed 30-April-2020].

[22] Douglas J Futuyma. *Evolution*. Sunderland, Mass. : Sinauer Associates, 2009.

[23] James M. Heather and Benjamin Chain. The sequence of sequencers: The history of sequencing dna. *Genomics*, 107(1):1 – 8, 2016.

[24] Tom Hill and Robert L. Unckless. A deep learning approach for detecting copy number variation in next-generation sequencing data. *G3: Genes, Genomes, Genetics*, 9(11):3575–3582, 2019.

[25] Yaping Hou, Jiexia Yang, Yiming Qi, Fangfang Guo, Haishan Peng, Dongmei Wang, Yixia Wang, Xiaohui Luo, Yi Li, and Aihua Yin. Factors affecting cell-free dna fetal fraction: statistical analysis of 13,661 maternal plasmas for non-invasive prenatal screening. *Human Genomics*, 13(62), 2019.

[26] Johannes Köster and Sven Rahmann. Snakemake—a scalable bioinformatics workflow engine. *Bioinformatics*, 28(19):2520–2522, 08 2012.

[27] T. Kuilman, A. Velds, and K. et al Kemper. Copywriter: Dna copy number detection from off-target sequence data. *Genome Biol*, 16(49), 2015.

[28] E. Lander, L. Linton, and B. et al Birren. Initial sequencing and analysis of the human genome. *Nature*, 409:860–921, 2001.

[29] B. Langmead and S Salzberg. Fast gapped-read alignment with bowtie 2. *Nat Methods*, 9:357–359, 2012.

[30] Yun Li, Carlo Sidore, Hyun Min Kang, Michael Boehnke, and Gonçalo R. Abecasis. Low-coverage sequencing:Implications for design of complex trait association studies. *Genome Res*, 21(6):940–951, 2011.

[31] Genome Research Limited. Unraveling the double helix, 2016. [Cited 2020-05-21] Available at `https://www.yourgenome.org/stories/unravelling-the-double-helix`.

[32] Gerstein MB, Bruce C, Rozowsky JS, and et al. What is a gene, post-encode? history and updated definition. *Genome Res*, 17(6):669-681, 2007.

[33] Karen Mestan, Leonard Ilkhanoff, Samdeep Mouli, and Simon Lin. Genomic sequencing in clinical trials. *Journal of translational medicine*, 9:222, 12 2011.

[34] N.A. Miller, E.G. Farrow, and M. et al Gibson. A 26-hour system of highly sensitive whole genome sequencing for emergency management of genetic diseases. *Genome Med*, 7(100), 2015.

[35] Adam B. Olshen, E. S. Venkatraman, Robert Lucito, and Michael Wigler. Circular binary segmentation for the analysis of array-based dna copy number data. *Biostat. Oxf. Engl.*, 5:557–572, 2004.

[36] George H. Perry, Nathaniel J. Dominy, Katrina G. Claw, Arthur S. Lee, Heike Fiegler, Richard Redon, John Werner, Fernando A. Villanea, Joanna L. Mountain, Rajeev Misra, Nigel P. Carter, Charles Lee, and Anne C. Stone. Diet and the evolution of human amylase gene copy number variation. *Nat Genet*, 10(10):1256–1260, 2007.

[37] A. Purcell. *Basic Biology: An Introduction.* Basic Biology Limited, 2018.

[38] Chiu R, Chan K, Gao Y, Lau V, Zheng W, Leung T, Foo C, Xie B, Tsui N, Lun F, and et al. Noninvasive prenatal diagnosis of fetal chromosomal aneuploidy by massively parallel genomic sequencing of dna in maternal plasma. *Proc. Natl Acad. Sci. USA.*, 105:20458–20463, 2008.

[39] Lennart Raman, Annelies Dheedene, Matthias De Smet, Jo Van Dorpe, and Björn Menten. WisecondorX: improved copy number detection for routine shallow whole-genome sequencing. *Nucleic Acids Research*, 47(4):1605–1614, 2018.

[40] M. Reich, T. Liefeld, and J. et al Gould. Copywriter: Dna copy number detection from off-target sequence data. *GenePattern 2.0. Nat Genet*, 38:500–501, 2006.

[41] R. Rieger, A. Michaelis, and M.M. Green. *A Glossary of Genetics and Cytogenetics: Classical and Molecular.* Springer Berlin Heidelberg, 2013.

[42] Pubudu Saneth Samarakoon, Hanne Sørmo Sorte, Asbjørg Stray-Pedersen, Olaug Kristin Rødningen, Torbjørn Rognes, and Robert Lyle. cnvScan: a CNV screening and annotation tool to improve the clinical utility of computational CNV prediction from exome sequencing data. *BMC Genomics*, 17(51), 2016.

[43] Roy Straver, Erik A. Sistermans, Henne Holstege, Allerdien Visser, Cees B. M. Oudejans, and Marcel J. T. Reinders. WISECONDOR: detection of fetal aberrations from shallow sequencing maternal plasma based on a within-sample comparison scheme. *Nucleic Acids Research*, 42(5):e31, 2014.

[44] Elliot Walsh. What is the complementary base pairing rule?, 2020. [Cited 2020-05-21] Available at `https://sciencing.com/complementary-base-pairing-rule-8728565.html`.

[45] Jiexia Yang, Yiming Qi, Yaping Hou, Fangfang Guo, Haishan Peng, Dongmei Wang, O. Y. Haoxin, Yixia Wang, Huajie Huang, and Aihua Yin. Performance of non-invasive prenatal testing for trisomies 21 and 18 in twin pregnancies. *Mol Cytogenet*, 11(447), 2008.

[46] Zakaria Jaadi. A step by step explanation of principal component analysis.

[47] Feng Zhang, Wenli Gu, Matthew E. Hurles, and James R. Lupski. Copy number variation in human health, disease, and evolution. *Annu Rev Genomics Hum Genet*, 39:451–481, 2009.

[48] H. Zhang, Y. Gao, F. Jiang, M. Fu, Y. Yuan, Y. Guo, Z. Zhu, M. Lin, Q. Liu, Z. Tian, H. Zhang, F. Chen, T. K. Lau, L. Zhao, X. Yi, Y. Yin, and W. Wang. Non-invasive prenatal testing for trisomies 21, 18 and 13: clinical experience from 146 958 pregnancies. *Ultrasound in Obstetrics & Gynecology*, 45(5):530–538, 2015.

[49] Chen Zhao, John Tynan, Mathias Ehrich, Gregory Hannum, Ron McCullough, Juan-Sebastian Saldivar, Paul Oeth, Dirk van den Boom, and Cosmin Deciu. Detection of fetal subchromosomal abnormalities by sequencing circulating cell-free DNA from maternal plasma. *Clinical Chemistri*, 61(4):608–616, 2015.

Appendix: Additional electronic files

comparison.xlsx: This table containes two sheets: Mixed samples, normal NIPT samples. All both set of samples are listed in these tables. Each tool is detected for these samples with their detected state, start of aberration and end of aberration.