Comenius university in Bratislava

Faculty of Mathematics, Physics and Informatics

Prediction of Properties of Polymorphic Genomes from Sequencing Data

diploma thesis

2018

Bc. Werner Krampl

Comenius university in Bratislava

Faculty of Mathematics, Physics and Informatics

Prediction of Properties of Polymorphic Genomes from Sequencing Data

diploma thesis

Study programme:	Computer science
Study field:	Informatics
Study department:	Department of Computer Science
Supervisor:	doc. Mgr. Bronislava Brejová, PhD.

Bratislava, 2018

Bc. Werner Krampl

Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta:	Bc. Werner Krampl
Študijný program:	informatika (Jednoodborové štúdium, magisterský II. st., denná forma)
Študijný odbor:	informatika
Typ záverečnej práce:	diplomová
Jazyk záverečnej práce:	anglický
Sekundárny jazyk:	slovenský

Názov: Prediction of Properties of Polymorphic Genomes from Sequencing Data
Predikcia vlastností polymorfných genómov zo sekvenačných dát

Anotácia: Moderné prístupy k sekvenovaniu DNA produkujú veľké množstvo krátkych reťazcov pochádzajúcich z cieľového genómu. Cieľom práce je rozšíriť existujúce metódy na odhad veľkosti cieľového genómu z takýchto sekvenačných dát. Tieto metódy sú založené na počítaní výskytov podreťazcov dĺžky K v sekvenačných dátach a na pravdepodobnostných modeloch odhadujúcich očakávané počty týchto výskytov. Cieľom práce je rozširovať použité modely, aby brali do úvahy ďalšie črty reálnych dát, najmä polymorfizmus, kde sa dve rôzne formy tej istej oblasti genómu nachádzajú v jednom jedincovi (jedna forma zdedená od matky, druhá od otca).

Vedúci:	doc. Mgr. Bronislava Brejová, PhD.
Katedra:	FMFI.KI - Katedra informatiky
Vedúci katedry:	prof. RNDr. Martin Škoviera, PhD.
Dátum zadania:	18.11.2015

Dátum schválenia: 16.12.2015

prof. RNDr. Rastislav Kráľovič, PhD.
garant študijného programu

.... študent

.... vedúci práce

Comenius University in Bratislava
Faculty of Mathematics, Physics and Informatics

THESIS ASSIGNMENT

Name and Surname: Bc. Werner Krampl
Study programme: Computer Science (Single degree study, master II. deg., full time form)
Field of Study: Computer Science, Informatics
Type of Thesis: Diploma Thesis
Language of Thesis: English
Secondary language: Slovak

Title: Prediction of Properties of Polymorphic Genomes from Sequencing Data

Annotation: Current DNA sequencing technologies can produce large numbers of short strings originating from the genome of interest. The goal of the thesis is to extend existing methods for estimating the size of the target genome from such sequencing data. These approaches are based on counting occurrences of substrings of length K in sequencing data and on probabilistic models estimating expected occurrence counts. The goal of the thesis is to extend the current models to consider additional features of real data, particularly polymorphisms, where two different forms of a certain genomic region occur in a single individual (one form inherited from the father, one from the mother).

Supervisor: doc. Mgr. Bronislava Brejová, PhD.
Department: FMFI.KI - Department of Computer Science
Head of department: prof. RNDr. Martin Škoviera, PhD.

Assigned: 18.11.2015

Approved: 16.12.2015 prof. RNDr. Rastislav Kráľovič, PhD.
Guarantor of Study Programme

....

Student Supervisor

I hereby declare that I wrote this thesis by myself, only with the help of the referenced literature, under the careful supervision of my thesis supervisor.

Werner Krampl

Acknowledgment:

I would like to thank my supervisor, Broňa Brejová, for her infinite patience, her professional guidance, her detailed explanations, her friendly support and all the things I learned from her.

I would like to thank my brother, my mother, my father, my grandma and all my friends for their love and great support during my studies.

I would like to thank my colleagues at room M25 for the creation of unrepeatable atmosphere during the last days of writing this thesis.

Last, but not least, I would like to thank Katka for her love and support, when it was most needed.

Abstrakt

Krampl, Werner: Predikcia vlastností polymorfných genómov zo sekvenačných dát. Diplomová práca. Univerzita Komenského v Bratislave; Fakulta matematiky, fyziky a informatiky; Katedra informatiky. Bratislava (2018). 71 strán. Školiteľ: doc. Mgr. Bronislava Brejová, PhD.

Táto diplomová práca sa zaoberá odhadovaním vlastností sekvenačných dát genómu ako veľkosť genómu, hĺbka pokrytia sekvenačnými dátami či chybovosť týchto dát. Kým štandardné postupy spracovania sekvenačných dát majú pre väčšie genómy vysoké požiadavky na výpočtové zdroje, techniky optimalizácie parametrov pravdepodobnostných modelov sa dajú použiť na získanie odhadu vlastností genómu v oveľa kratšom čase. Analyzujeme sekvenačné dáta z polymorfných organizmov a ich vplyv na už existujúce modely. Následne vytvárame nové predikčné modely, ktoré explicitne uvažujú dáta z diploidných organizmov, testujeme tieto modely na rôznych dátach a vyhodnocujeme ich výsledky. Ďalej sme identifikovali problém v už zverejnenom modeli, a navrhli sme riešenie.

Kľúčové slová: bioinformatika, odhad vlastností genómu, optimalizácia parametrov, sekvenačné dáta

Abstract (English)

Krampl, Werner: Prediction of Properties of Polymorphic Genomes from Sequencing Data. Diploma thesis. Comenius University in Bratislava; Faculty of Mathematics, Physics and Informatics; Department of Computer Science. Bratislava (2018). 71 pages. Supervisor: doc. Mgr. Bronislava Brejová, PhD.

This diploma thesis focuses on estimations of attributes from genome sequencing data such as genome size, depth of coverage by sequencing data or error rate of this data. As standard procedures of sequencing data processing have high demands on computational resources for larger genomes, parameters optimization of prediction models techniques can be used to obtain estimations of genome attributes in a much shorter time. We analyse sequencing data from polymorphic organisms and their influence on existing prediction models. We then create new prediction models that explicitly consider the existence of diploid organisms, we test them on various datasets and evaluate their results. Furthermore, we have identified one problem in an already published model and suggest a solution.

Keywords: bioinformatics, genome attributes estimation, parameter optimization, sequencing data

Contents

List of Figures

List of Tables

Introduction

Development of genome sequencing techniques allowed researchers to study genomes of a wide spectrum of organisms - from microscopic bacterias to largest mammals. However, genome sequencing process creates a huge amount of data that needs to be efficiently processed. Beside standard techniques of sequecing data processing such as assembly and alignment, that have high resource complexity, emerges a different approach to estimate attributes of genomes - optimization of parameters of probabilistic models.

Such parameter optimization returns estimates of several attributes such as genome length or number of errors in sequencing data at a much higher speed than the standard methods. One of the main uses is for planning sequencing experiments, when parameter optimization is used on preliminary data that are not sufficient for genome assembly. Furthermore, it can be used in specific cases, when standard techniques are infeasible due to high computational demands such as obtaining information on genome attributes from a larger genome or larger population of organisms.

There are already several tools for genome parameter optimization, however they have their conditions on the data they can process. One of the least studied areas of genome parameter optimization are models that consider data originating in diploid organisms - organisms, that have two highly similar copies of each chromosome. In this work, we are focusing on estimation from diploid data first by studying its influence on the existing prediction models, and then we create new models that estimate attributes of diploid organisms.

We introduce necessary biological background in the first chapter of this thesis, then we explain problems connected with genome assembly. We mention several existing tools and models, their usage and their limitations and at the end of the chapter, we propose our areas of research.

In the second chapter, we begin with a more detailed explanation of an existing model upon which we are building our first model. We subsequently explain the optimization process in a tool named CovEst, that we are expanding with our work. We then analyze how the existence of diploid genomes affect estimations of models not incorporating diploid data. Later, we create a new model that considers existence of heterozygous positions under the condition that there are no repeating sequences in the genome. At the end of this chapter, we evaluate the results of this model together with other already existing models on simulated data.

In the third chapter, we have expanded an existing model from CovEst to conduct an experiment to see whether more parameters will affect the estimation of attributes by this model. We further notice an inconsistency in claims about results by original authors of this model and our observations. We state a hypothesis explaining the difference in our results and their claims and experimentally confirm it, making their statements invalid. We further suggest a system of equations to estimate values that original authors desired and test it.

In the last chapter, we expand the considered genomes to diploid genomes containing repeating sequences. We start by experimenting with existing model from CovEst on simulated sequencing data corresponding to a real genome. Subsequently, we create a model that, under a strict assumption, estimates values for a diploid genome and evaluate it on simulated genomes. Next, we suggest a new model with strong biological basis. In the end, we evaluate the results of several models on real sequencing data.

Chapter 1

Biological background and problem statement

In this chapter we will introduce our research problem from both biological and computational points of view. We will define terminology used throughout our work, explain steps of genome sequencing and, at the end of the chapter, we will describe aim of our work.

1.1 Biological background

Genomics as a science studies genetic information of organisms. Genome sequencing techniques and bioinformatic algorithms are needed to study genetic information.

Genetic information is stored in every cell of an organism, where it is formed by two strands of **deoxyribonucleic acid (DNA)**.[4] These strands are composed of nitrogen-containing **nucleobases** (adenine - A, cytosine - C, guanine - G, thymine - T) encoding genetic information and from deoxyribose sugar and phosphate group.

Two strands further connect via hydrogen bonds (weak chemical bonds) into a double helix.

Different orders of nucleobases (A, C, G, T) are sources of variability of species. [4] Figure 1.1 depicts structure and composition of DNA.

DNA is organised in structures called **chromosomes**. Chromosomes are part of cell nucleus and in eukaryotic organisms (which include all animals and plants), shorter chro-

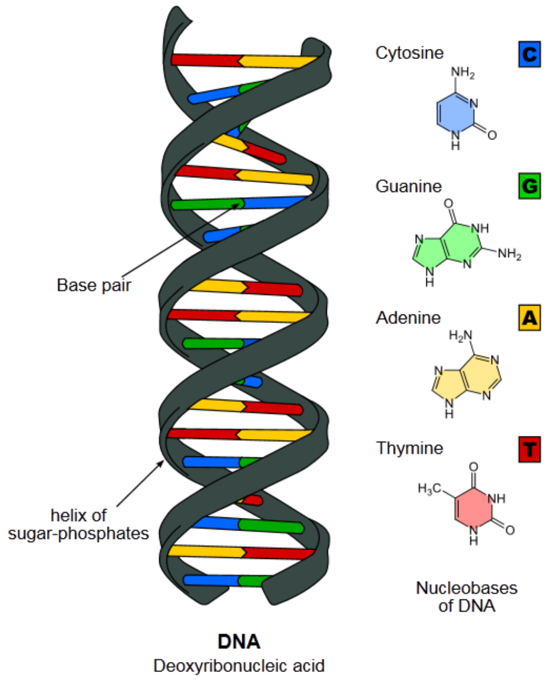


Figure 1.1: Double helix DNA and nucleotides. Source: wikipedia.org

mosomes are also stored in cell organelles. [4]

Every biological specie has its characteristic chromosomal constitution, called **karyotype**.[4] For example, human karyotype consists of 46 chromosomes, from which 22 has its copy (22 chromosomes is present two times) called autosomes and the remaining two ale called allosomes (sex chromosomes, either two X chromosomes representing woman or X and Y chromosomes representing male).[4]

Figure 1.2 depicts human male karyotype.

Term **genome** represents *Complete genetic information of a given organism, including genes and non-coding sequences.*[5]

Now we will define aforementioned terms computationally.

Definition 1.1.1 ***Base alphabet*** *is alphabet $\Sigma_D = \{A, C, T, G\}$. Symbols of Σ_D alphabet are called bases.*

Definition 1.1.2 ***Chromosome*** *is a finite, non-empty string over Σ_D.*

Definition 1.1.3 ***Genome*** *is a finite, non-empty language of chromosomes.*

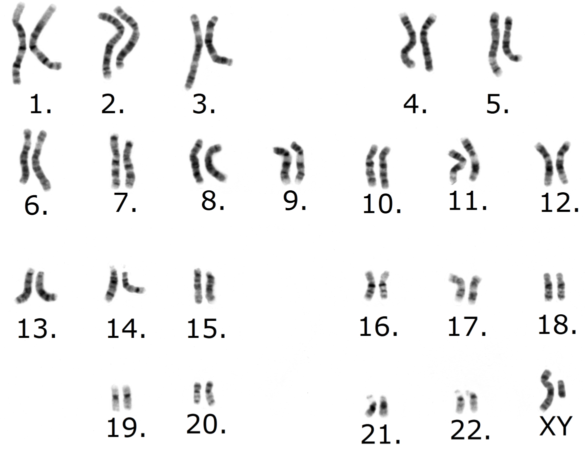


Figure 1.2: A human male karyotype. Source: wikipedia.org

Definition 1.1.4 ***Genome size*** *is a sum of lengths of all genome chromosomes.*

A million bases long string from Σ_D is called a **megabase**. For example, the human genome has length approximately three thousand megabases.[6]

As we have indicated, the karyotype of an organism may contain one or more copies of the same chromosome. Number of copies is called **ploidy**, an organism is **haploid** if it has one set of chromosomes, **diploid** for two sets of chromosomes etc. For example, green algae *Chlamydomonas elegans* is a haploid organism[7] and aforementioned human is a diploid organism. Term *haploid* also denotes one complete set of chromosomes in a diploid organism. Example given, human has 46 chromosomes, so haploid set is 23 chromosomes. Closely related to the term diploid is term **heterozygosity**. In most diploid organisms, a chromosome and its diploid copy are not identical, they contain positions with different bases in the two copies.[6] These positions are called **heterozygous positions** and they account (although not exclusively) for inherited variations between individuals of the same species. They can cause a disease or a trait (for example, people with a certain heterozygosity have a higher incidence of hypertension [8]). Figure 1.3 demonstrates heterozygosity in one position. Note that **genome size** is the size of only haploid set of chromosomes. Genome also may contain **repeated sequences** or *repeats* - strings of bases that occur multiple times in the genome. Human genome consist of ca. 50% of repeats. For example,

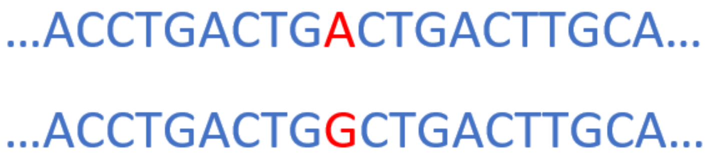


Figure 1.3: Example of a heterozygosity in one position in a diploid genome.

a 300 bases long repeat called *Alu* makes up for 11% of human genome size with over million copies. [5]

1.2 Introduction to sequencing

DNA sequencing is a process that analyses a given DNA molecule and yields its order of bases. This procedure alone is a combination of biochemical processes and bioinformatic algorithms [9].

History of DNA sequencing extends to 1970s, during which first experiments with genetic information extraction were conducted. An important milestone was creation of the so-called *Sanger method* by British biochemist Frederick Sanger. This method is in an updated form used up to this day [10] and belongs to so-called First-Generation Sequencing (abb. FGS). A common feature of FGS methods is that their usage to obtain higher quantity of data is difficult[10].

Rapid development of DNA sequencing techniques began at the beginning of the nineties. A variety of new methods and sequencing technologies has been since developed with diverse properties, such as used resources, sequencing approaches and final cost of obtained data. Comparison of several technologies can be seen in table 1.1.

method	**read length**	**accuracy**	**reads per run**	**time per run**	**cost per 1 million bases (in US dollars)**
Chain termination (Sanger sequencing) - FGS	400-900 bases	99.9%	9 000	20 minutes to 3 hours	$2400
Pyrosequencing (454) - SGS	700 bases	99.9%	1 million	24 hours	$10
Sequencing by synthesis (Illumina) - SGS	70-500 bases	99.9%	1.2-1.4 billion	1 to 2 weeks	$0.13
Nanopore Sequencing - TGS	up to 500 kilobases	95%	dependent on read length	1 minute to 48 hours	$500-$999
Single-molecule real-time sequencing (Pacific Biosciences) - TGS	10 kilobases up to >40 kilobases	87%	50 000	30 minutes to 4 hours	$0.13-$0.60

Table 1.1: Comparison of several sequencing methods. Source: [1, 2]

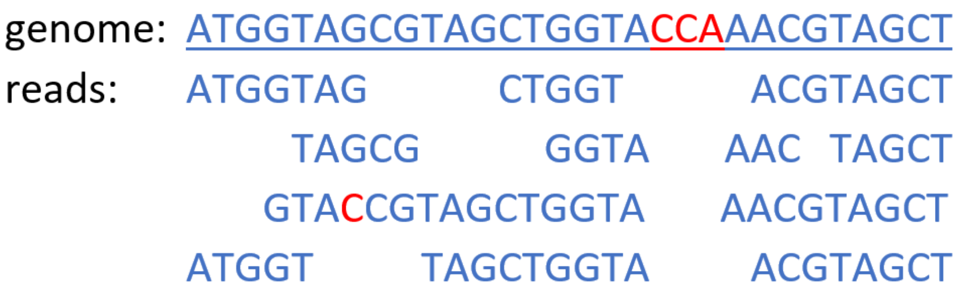

genome: ATGGTAGCGTAGCTGGTACCAAACGTAGCT
reads: ATGGTAG CTGGT ACGTAGCT
TAGCG GGTA AAC TAGCT
GTACCGTAGCTGGTA AACGTAGCT
ATGGT TAGCTGGTA ACGTAGCT

Figure 1.4: A genome and its reads. Genome sequence with red mark is not covered. One read contains a sequencing error marked red.

Up to this point in time, there are two more generations of sequencing technologies - Second-Generation Sequencing (abb. SGS) and Third-Generation Sequencing (abb. TGS). SGS technologies are defined as technologies that massively increased throughput by parallelising many biochemical reactions. TGS allows direct sequencing of single DNA molecules.[9]

Prevalent feature shared by the majority of sequencing techniques is that they do not read entire chromosome as a continuous string. Instead, they yield high amount of shorter DNA sequences called **reads** (the length between 100-1000 in length for SGS).[10] Reads are finite, non-empty substrings of chromosomes, however as we later explain, they might not be identical to their source. Reads can overlap and ideally they cover the entire input genome. Example of genome and its reads can be seen in figure 1.4.

New genome reads assembly is called *de novo assembly.*[10] We will describe de novo assembly in the following subsection.
Second option of standard reads processing is called *reads mapping*. During mapping, reads are not assembled together and, instead, they are aligned to a reference genome that has already been created (either by assembly or mapping to yet another reference).[10] Overview of sequencing process can be seen in figure 1.5.

Term *coverage* or *depth of coverage* denotes the number of reads that contain given

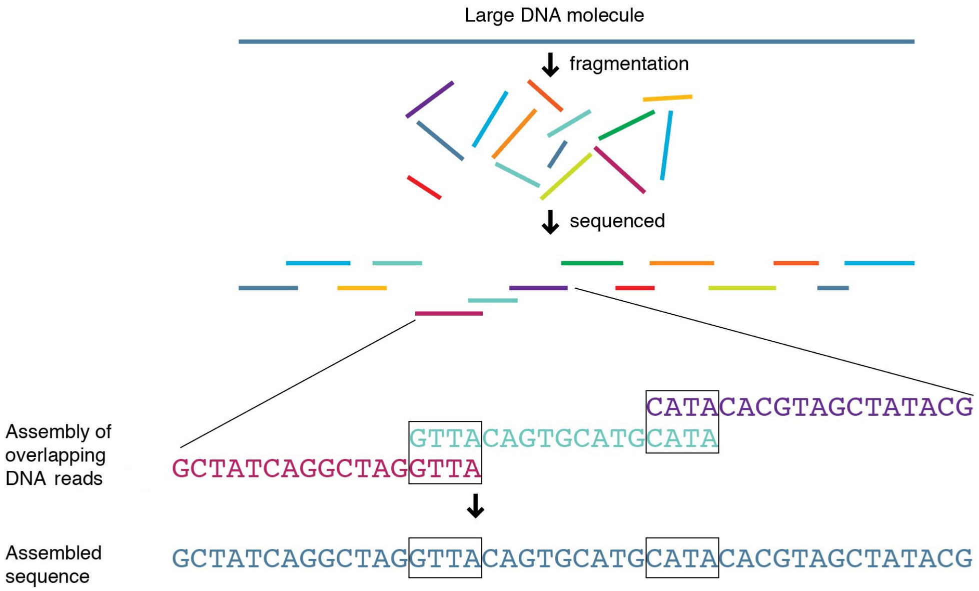


Figure 1.5: Overview of sequencing process. Source: rayhuitech.com

genome position. *Genome coverage* is a mean of coverage of every base position the genome contains. Note that the depth of coverage of individial positions in a genome may vary with overrepresented regions and regions with no coverage at all. Example of uncovered region in genome can be seen in figure 1.4.
Unfortunately, reads created by DNA sequencing are not error-free as sequencing methods are yet to be flawless. This results in false bases in reads. Figure 1.4 includes a read with a sequencing error, where base G was read as C. Sequencing errors in reads create several complications as we describe in next subsections. Error rates of several sequencing techniques can be seen in Table.1.1

Lastly, we will define term *k-mer*:

Definition 1.2.1 ***K-mer*** *is a substring of length k of string.*

Figure 1.6 shows read with 10 bases in size and all its k-mers for k = 5.

1.3 Genome assembly

As we have mentioned in the previous chapter, *de novo assembly* is a process during which given set of overlapping reads are joined (assembled) together forming longer string called

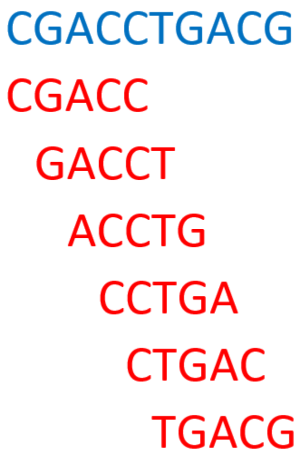


Figure 1.6: 10 bases long read and all its k-mers for k = 5.

contig.[6]

There are several important complicating factors that adds difficulty to de novo assembly process [6]:

- Genome division to chromosomes,
- Genome of higher species might have heterozygous positions.
- Genomes of higher species contain *repetitive sequences*. During assembly it is difficult to obtain actual number of repeats.
- Genome sequencers create erroneous reads that do not exist in genome further influencing coverage.

We can divide most assembly algorithms into two groups: **Overlap-Layout-Consensus algorithms (abb. OLC)** and **algorithms based on De Bruijn graph (abb. DBG)**. [11]

Overlap-Layout-Consensus algorithms

OLC algorithms for genome assembly searches for locally best assembly in input reads. Algorithms are typically divided into three parts: [6]

Overlap:

Overlap part searches for overlapping reads.

Layout:

Layout part tries to find relative positions of reads within contigs. Problem is occurrence of repeats, when one suffix of contig is another contigs prefix and it is unknown how many times does repeating sequence between these two contigs occur.

Consensus:

Consensus part picks most probable base in contig from every read covering given position, e.g. given position might be heterozygous or there might be occurrence of erroneous read.

Specific algorithm in each part is subject to individual implementation. However, common feature of every implementation is high demand on computational resources due to high amount of data being processed. Given several examples, in Overlap part, naively comparing every pair of reads will result in $O(n^2)$ time complexity with n representing amount of reads.[6] This could be improved by building k-mers out of every read, sorting them and comparing their positions.[6] Unfortunately, given billions of reads (for example 10-fold coverage of relatively small Drosophila melanogaster [12] yields 6 million reads with length 300 bases) that needs to be processed, demand on computational resources is still high.

De Bruijn Graph algorithms

DBG algorithms for genome assembly aim to find globally best assembly in input reads. Algorithms usually follow these steps:[6]

1. Separation of every read into its k-mers.
2. Creating graph G, where vertices represents all $k-1$ substrings of k-mers - two vertices form one k-mer.
3. Adding edges between vertices that overlap on k-1 bases (k-mer $B_1...B_k$ connects vertices $B_1...Bk-1$ and $B_2...B_k$.
4. Final order of bases in contig is given by Hamiltonian path in G.

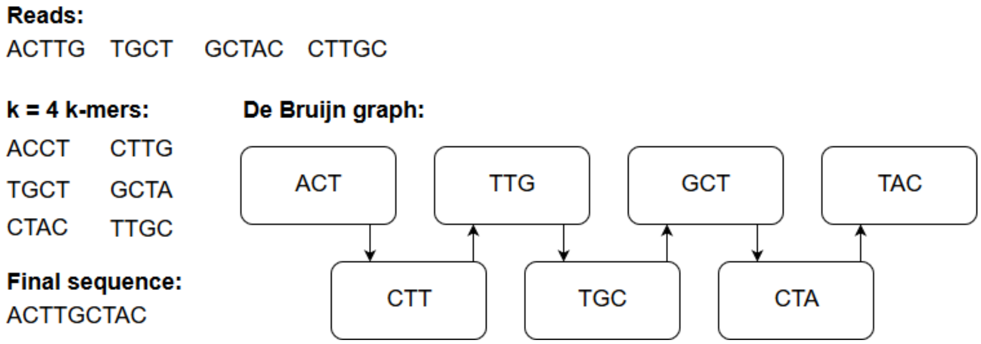


Figure 1.7: Example of DBG with 4 input reads and k = 4.

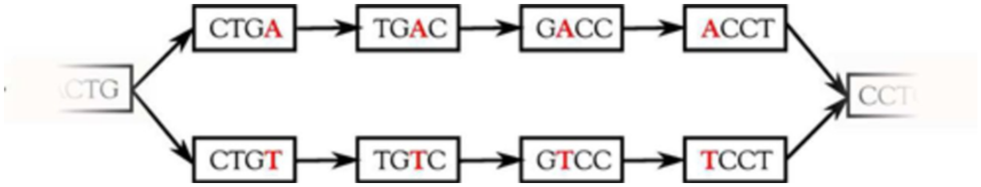


Figure 1.8: Example of DBG bubble. Source: researchgate.net

Example of DBG can be seen in picture 1.7. DBG algorithms in practice have to solve several complications such as processing reads shorter than k, existence of several Hamiltonian paths or existence of none etc. Determining proper k is another issue as short k-mers will add many uncertainties and long k-mers will have high memory demand.[6] Lastly, finding Hamiltonian path in graph is NP-complete in general, demanding high computational resources.

Sequencing errors will cause existence of so-called *bubbles* in graph. Bubble is a part of graph, where exists two ways from one vertex to another and vertices in these ways are identical with the exception of erroneous base. Way with low coverage is a candidate to be marked as containing sequencing error. If coverage of both ways is approximately same (and whole genome coverage is sufficient), this bubble contains polymorphic position.[13] (picture 1.8)

Table 1.2 shows number of assembled contigs, assembly size, running time and memory peak of several genome assembly tools on sequencing data from human chromosome 14.

	Sparse Assembler	**Gossamer**	**Minia**	**Diginorm Velvet**	**DiMA**	**ZeMA**	**Original Velvet**
Number of contigs	52785	67160	52926	55002	61039	68253	52085
Assembly size	101600523	73046277	74079569	79129375	80448331	81139464	81190207
Time (hours min sec)	1:1:37	3:6:50	1:33:13	1:18:16	1:21:8	1:15:09	2:27:46
Memory Peak (GBs)	1.72	3	0.76	3.34	8.7	1.2	49.3

Table 1.2: Comparison of several assemblers on sequencing data from human chromosome 14. Source: [3]

1.4 Properties prediction

Approaches introduced in previous section provide us with approximate order of bases in genome. From this assembled (or aligned) genome we can easily infer several basic properties of genome and its sequencing run: coverage, genome size, error rate, polymorphism rate, repeat structure. However, computational resources required to obtain order of bases in genome are high and not applicable for larger genomes.

If our target is to obtain only genome properties, we can bypass the unnecessary assembly. Instead, we can estimate these properties straight from sequencing data using probabilistic models and parameter optimization. This approach carries several advantages over assembly:

- is faster
- is less memory demanding
- avoids additional biases introduced by assemblers
- can be used on smaller set of reads not sufficient for assembly

Disadvantage to this approach is that it only estimates properties instead of explicitly finding them. We believe that this estimation is sufficient for several important tasks. One of main examples of usage is using estimation of genome size on preliminary data. As we will later explain, genome size is closely related to genome coverage. To create sequencing data, one needs to know genome size to obtain desired coverage. However, without the information on genome size, correct amount of data can be only guessed. Using estimation of genome size on small preliminary sequencing data allows us to plan overall sequencing experiment to desired amount. There are also many other examples of genome properties estimation uses as phylogeny of large genomes, comparison of polymorphism rates of a high amounts of individuals of same species etc.

There are several works that focused on properties prediction. Works from Waterman et al. [14] and Williams et al. [15] estimate only coverage, genome size and repeats structure. Hozza et al. [16] adds error rate to prediction. However, all three works do not

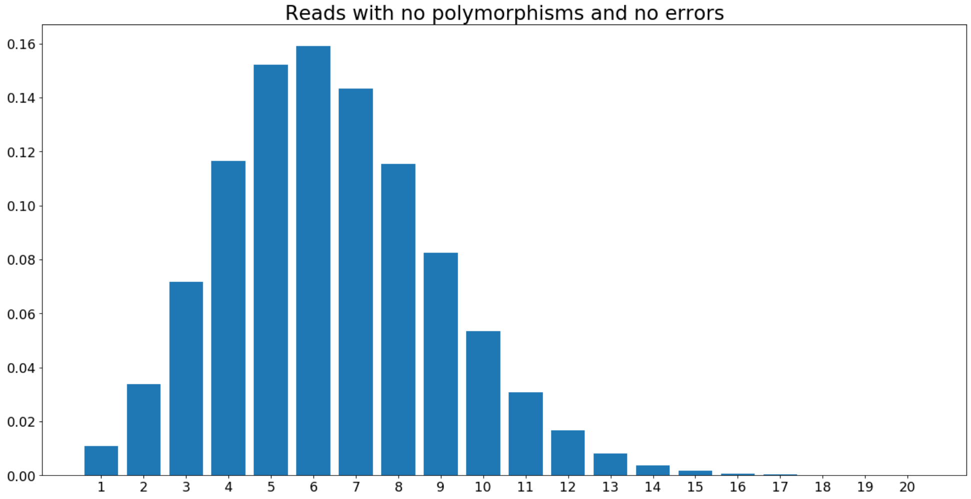


Figure 1.9: Example of k-mer abundance spectrum for reads containing no polymorphisms and no errors, depth of coverage = 8.

consider existence of heterozygous positions in genome and thus are not suitable for use on diploid organisms. Relatively new work by Vurture et al. [17] (published during our research) estimates coverage, genome size, simplified repeats structure and polymorphism rate and ignores error rate that again restricts its usage.

Prediction models in works listed above do not work with reads directly, but with their k-mers. More specifically, they work with summary statistics called *k-mer abundance spectrum* - histogram, that contains for every class j the number of distinct k-mers with j occurrences in sequencing data. Example of simple k-mer abundance spectrum containing no erroneous k-mers and no polymorphic positions can be seen in picture 1.9

Existence of errors in reads gives rise to histogram classes with low-occurring k-mers as errorneous k-mers tends to be unique with low occurences in data. Subsequently, other histogram classes decreases. We will explain this phenomenon in next chapter. Picture 1.10 depicts a histogram with erroneous k-mers.

There are several approaches to manage sequencing errors in reads. Waterman et al. [14] errors simply ignores stating that if the error rate is too low, errors have no significant influence on estimation. We argue that even relatively low error rate of 0.1% might have significant impact on estimations. Consider standard NGS read length 300. If we set

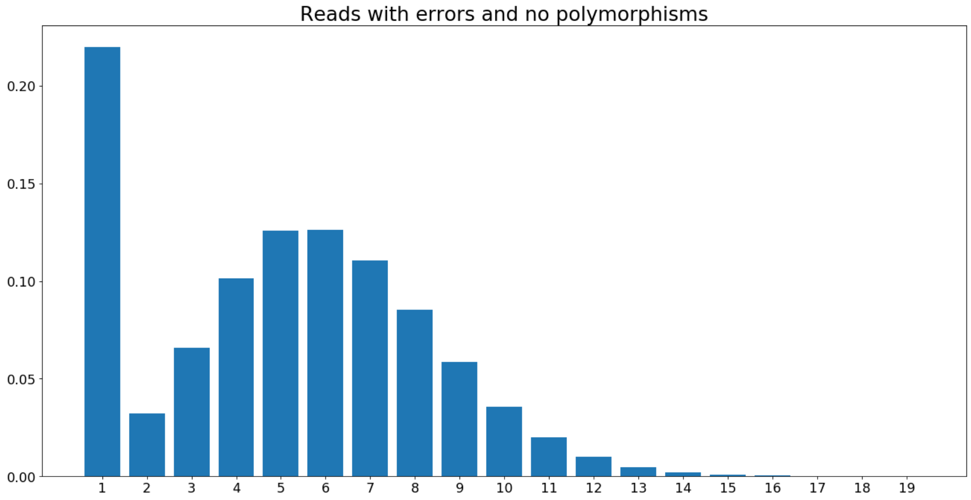


Figure 1.10: Example of k-mer abundance spectrum for reads containing no polymorphisms and 0.2% error rate, depth of coverage = 8.

k to 21, out of 280 possible k-mers up to 63 will be erroneous. This will further affect estimations as model will consider erroneous k-mers as unique parts of the genome.

Another approach is to exclude erroneous k-mers through removing all low-coverage k-mers from histogram. This approach was adopted by authors in works by Williams et al. [15] and Vurture et al. [17]. We again disagree with this technique as sequencing does not produce reads and subsequently k-mers from every position with same coverage and as we stated, there are regions with low-to-no coverage. Trimming low coverage classes from histogram will, together with erroneous k-mers, dismiss also k-mers that occur in genome. This further restricts their models to usage only on data with high coverage and low error rate.

Due to expectation of low-to-none error rate all models accomodating previous approaches are unable to cope with reads generated by TGS sequencers, as they have higher error-rates than NGS sequencers (see 1.1). Third option is to incorporate possibility of occurrences of erroneous k-mers into prediction model. This approach is used in work by Hozza et al. [16], which results in correct estimations of coverage, genome size and error rate on datasets with higher error rates and lower coverage.

Out of mentioned works, only Vurture et al. [17] can be used on diploid data and estimate polymorphism rate (as authors of said work state, it can be used *only* on diploid data).

They further limits their model to only consider k-mer duplication in genome and ommiting k-mers with three or more repeats. Reason for this limit, as they state, is that *only a small proportion of a genome sequence typically occurs in higher level repeats, since copy number typically falls off quickly in real genomes following a Zeta distribution.* We argue that this gives no guarantee that genome with higher number of more repeating k-mers can occur.

As we have stated, none of already existing models incorporate all properties allowing universal usage. Therefore, aim of our thesis is to **create a model, that estimates coverage, genome size, error rate, heterozygosity rate and repeats structure** with possibility to be used independently on **haploid** and **diploid** organisms. As a basis of our work, we will use work by Hozza et al. [16].

We have to further **analyse properties** of selected genomes to create a concept of how diploid organisms change behavior of already existing models and to choose the best approach to model polymorphism rate in sequencing data.

Additionaly, we will **test existing *RE* model** from Hozza et al. [16] with more repeats parameters and **analyse results**.

During our research, we have also found out that *repeats parameters* in Hozza et al. [16] - and probably in other works too - do not describe what the authors state. Our objective is to **clarify** what repeats parameters actually describe.

Chapter 2

Errors and polymorphisms model

In this chapter we will analyse heterozygosity of genome and discuss the effect of heterozygous genome positions on the existing models. Later we will suggest a new model for diploid genomes without repeating regions, we will describe the optimization process and test new model. Note that from further on, we will use term *polymorphism* instead of *heterozygosity*, as k-mers that have two variants are *polymorphic*.

2.1 E model

No errors and no polymorphisms in reads

As we have stated in previous chapter, the input data to prediction models is a *k-mer abundance spectrum* - a histogram, that contains for every class j the number of distinct k-mers with j occurrences in the input sequencing data. There already exist several tools designed for efficiently counting occurences of k-mers using text indexing techniques such as hashing [18], Bloom filters [19] or streaming techniques [20].

Note that throughout this chapter we assume that there are no repeating k-mers in genome (every k-mer in genome occurs only once) and therefore no polymorphic k-mer has a non-polymorphic copy in genome.

If the reads contain no errors and sequenced genome is without polymorphisms, the shape of k-mer abundance spectrum follows a Poisson distribution with its peak at genome coverage. [21] To be more precise, the peak is not at genome coverage C, but at k-mer coverage C_k (mean number of reads covering a k-mer in genome), which is:

$$C_k = \frac{C \cdot (r - k + 1)}{r} \tag{2.1}$$

where r is the average length of reads.

Idea behind k-mer coverage is that not every read covering part of k-mer will cover its entire length. Therefore a k-mer coverage C_k is generally less to genome coverage C. Example of error-free and polymorphism-free k-mers following Poisson distribution can be seen in Figure 1.9.

Errors in reads

Sequencing errors in reads add new k-mers to the set of distinct k-mers in the input reads. These new k-mers, however, have low occurence in input data as sequencing error creates new k-mers randomly and, subsequently, give rise to low-occurence classes at the left side of the histogram. This further decreases the size of the other classes. Figure 1.10 demonstrates an example of histogram with errorneous k-mers.

We will first demostrate building the Full error (E) model with steps by Hozza et al. [16].

As frequences of error-free and polymorphism-free k-mers follow a Poisson distribution centered at C_k, it is a foundation for E model. However, due to the fact that abundance of k-mers is for some k-mers equal to zero and histogram will never contain them, model E applies *truncated Poisson distribution* defined as:

$$p_j = \frac{C_k^j e^{-C_k}}{j!(1 - e^{-C_k}} = f(j; c_k) \tag{2.2}$$

where p_j is the probability of observing abundance j.

Next step lies within prediction of erroneous k-mers and results in creating a mixture of probabilities. As we have already noted, there is a probability that a sequencing error creates the same k-mer several times and thus, it will contribute to higher classes in histogram.

Considering two k-mers A and B with Hamming distance (defined as a number of positions with different characters in two strings of equal length) s, probability of acquiring k-mer B as a result of sequencing k-mer A is $\epsilon^s(1-\epsilon)^{k-s}3^{-s}$. Sequencing k-mer A C_k times will in expectation results in $\epsilon^s(1-\epsilon)^{k-s}3^{-s}C_k$ copies of k-mer B. If we denote λ_s as the previous quantity and create a histogram with Poisson distribution with parameter λ_s, we will get a histogram that describes probabilities of occurence of erroneous k-mers with exactly s errors.

The overall probability distribution p_j for abundance class j is modeled as a mixture of $k+1$ probability distributions with λ_s parameters, where $s \in 0 \ldots k$. These partial distributions need to be weighted.

Let α_s be the fraction of k-mers with s errors and n_s be the expected number of observed unique k-mers with Hamming distance s from their respective source k-mers:

$$n_s = n\binom{k}{s}3^s(1-e^{\lambda_s}) \tag{2.3}$$

where:

- n: the number of source k-mers,
- $\binom{k}{s}3^s$: the number of different k-mers with s errors for a single source k-mer,
- $(1-e^{\lambda_s})$: probability of occurence of at least one erroneous k-mer with s errors to all k-mers.

Mixture weights α_s are computed by normalizing n_s with their sum, removing unknown parameter n.

The overall probability p_j in model E is then [16]:

$$p_j = \sum_{s=0}^{k} \alpha_s f(j;\lambda_s) \tag{2.4}$$

where f is truncated Poisson distribution with parameter λ_s.

Coverage C (computed from C_k) is then used to determine genome size by the equation:

$$G \cdot C = \sum_{read \in Reads} size(read) \tag{2.5}$$

$$G = \frac{\sum_{read \in Reads} size(read)}{C} \tag{2.6}$$

where G is the genome size and *Reads* is a set of all reads.

2.1.1 Optimization process

Prediction models E and RE that we are using as a basis of our models are implemented in a tool named CovEst.[16] We will use the same optimization process as is in their tool.

First step of parameter optimization is creating k-mer abundance spectrum out of sequencing reads. We have decided to use tool named Jellyfish [18] for its fast, parallel computing of k-mer occurrences. As Williams et al. [15] suggest, we use k = 21. By using an odd number we avoid a situation where one k-mer is a reverse-complement of itself.
Optimization process subsequently uses k-mer abundance spectrum to find the highest log-likelihood estimate of model parameters. Log-likelihood of observing k-mer abundance spectrum W is defined for every prediction model in this work as follows:

$$L(W|\theta) = \sum_{j=1}^{m} log(p_j) \cdot w_j \tag{2.7}$$

where:

- θ: Model parameters,
- m: highest observed class of W,
- p_j: probability of observing abundance j in a particular model,
- w_j: abundance of j in W.

To speed up computations, histogram is trimmed of higher abundance classes so that the trimmed histogram will contain 99.999% of all sequenced k-mers.

Next step is choosing the initial values of parameters for optimization. CovEst [16] uses a heuristic that assumes that all k-mers with abundance at least 2 are correct. If estimation of initial values fail due to insufficient amount of available data, $c = 1$ and $\epsilon = 0.5$ is used.

Authors of CovEst noticed that sampling the histogram to target depth of coverage between 12-15 increases accuracy of parameters estimation.

Optimization in CovEst further continues with L-BFGS-B optimization algorithm [22], more specifically, with its implementation in SciPy library for Python. [23]

2.2 Polymorphism analysis

Recall that in diploid genomes autosomal chromosomes occur in pairs. The two chromosomes in a pair have highly similar sequences, yet they differ in some positions which we call polymorphic. Also recall that we aim to estimate the genome haploid chromosome set, i.e. we include only one chromosome from each pair.

Sequencing a polymorphic position results in reads, and subsequently k-mers, that differ at this position. Origin of this difference lies within the source haploid chromosome of the pair of chromosomes. Since each read is created from one chromosome, it contains only one base from a pair of bases in a polymorphic position. Since both chromosomes in a pair enter the sequencing process, sequencer randomly creates reads from both of them. As the probability of choosing the source chromosome for a read is uniform, i.e. 0.5 for both chromosomes, the coverage of polymorphic k-mers will be one half of the overall coverage. An example of a diploid chromosome and its reads can be seen in Figure 2.1.

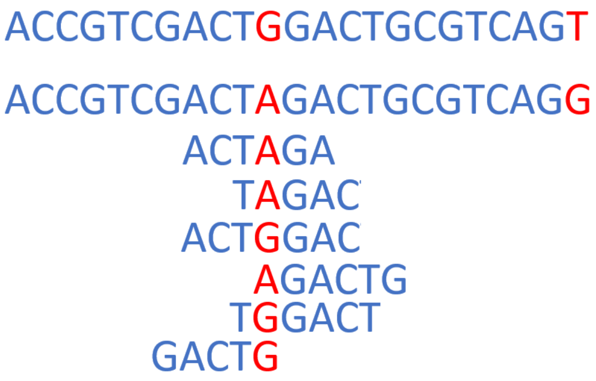


Figure 2.1: Example of a diploid chromosome pair and some of its reads. Roughly half of the reads cover one haploid chromosome and the rest the other haploid chromosome.

Polymorphism in reads

If the sequenced organism was diploid, the resulting k-mers will contain polymorphic positions and thus, polymorphic k-mers covering these positions. The abundance spectrum created from these k-mers will show a characteristic two-peak profile researched by Kajitani et al. [24]. Peaks will be centered at coverage C_k and half of coverage $C_k/2$ - given the reasonable assumption that abundance of sequenced polymorphic k-mers in one position is divided in half.

Size of peaks is regulated by the polymorphism rate of the genome. The higher the rate, the higher the polymorphism peak and consequently, lower the non-polymorphism peak. The relatively large polymorphism peaks, even at low polymorphism rates are caused by rapid increase of fraction of k-mers containing a polymorphic position with relatively small increase in polymorphism rate, since every polymorphic position creates $2k$ different k-mers with coverage $C_k/2$. Table 2.1 shows different polymorphism rates and their corresponding k-mer polymorphism rates for $k = 21$.

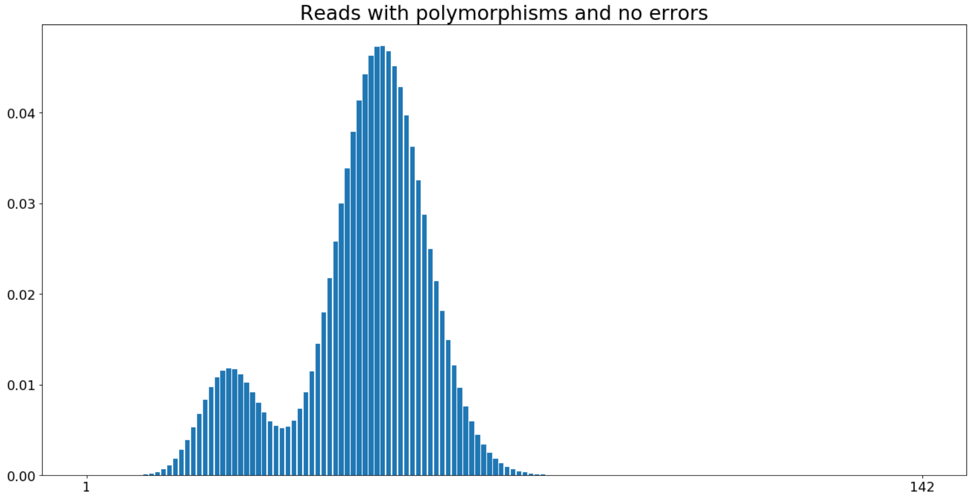


Figure 2.2: An example of a k-mer abundance spectrum for simulated reads containing no errors and polymorphism rate 0.4%, 64-fold depth of coverage from section 2.4

π	0.1%	0.2%	0.4%	0.8%	1.6%	3.2%	6.4%
p'	2.07%	4.11%	8.07%	15.52%	28.73%	49.49%	75.07%

Table 2.1: Different polymorphism rates in genome and their respected polymorphism rates in k-mers, $k = 21$, π is polymorphism rate in genome and p' is k-mer polymorphism rate.

An example of a histogram from polymorphic k-mers is in Figure 2.2. By [17], *"...the height of the heterozygous peak grows very quickly and matches the height of the homozygous peak at around only 1.2% heterozygosity for k=21."* Note that for data with low-coverage data (less than 8) and low polymorphic rate the polymorphic peak is difficult to distinguish as it is absorbed by the higher non-polymorphic peak on the right side and the error peak on the left side.

Polymorphism and existing models

Our model presented in this chapter extends the existing E model from CovEst.[16] E model does not consider existence of polymorphisms, and therefore does not yield sufficiently accurate estimates for highly polymorphic genomes, as we will demonstrate in next section.

In general, E model is insufficient to estimate parameters of diploid data. Recall that a single polymorphic position overlaps $2k - mers$. If we consider a haploid chromosome of length G, then a diploid pairs of chromosomes will have up to $\pi \cdot G \cdot k$ new distinct k-mers, π being the polymorphism rate. However, E model will count these new k-mers as part of a haploid chromosome and as a result it underestimates the coverage and by 2.6 overestimates genome size.

Nevertheless, there are still cases when E model estimates parameters correctly on diploid data. From our experiments in section 2.4, we see that there are cases when coverage is relatively low - on our tests on a million bases long genome, E model yields steadily correct estimates on 4-fold coverage with polymorphism rate up to 0.8%, since in such inputs, polymorphic k-mers are not sufficiently covered and therefore only one polymorphic k-mer out of a pair occurs. Average relative accuracies for a 4-fold coverage and various error rates and polymorphism rates are in table 2.2. Note that E model, even if its prediction is accurate, still gives no information on the polymorphism rate in genome, which we would like to estimate.

We define accuracy of parameter Θ estimation as:

$$acc = 100(1 - abs(1 - \theta/\theta*)) \tag{2.8}$$

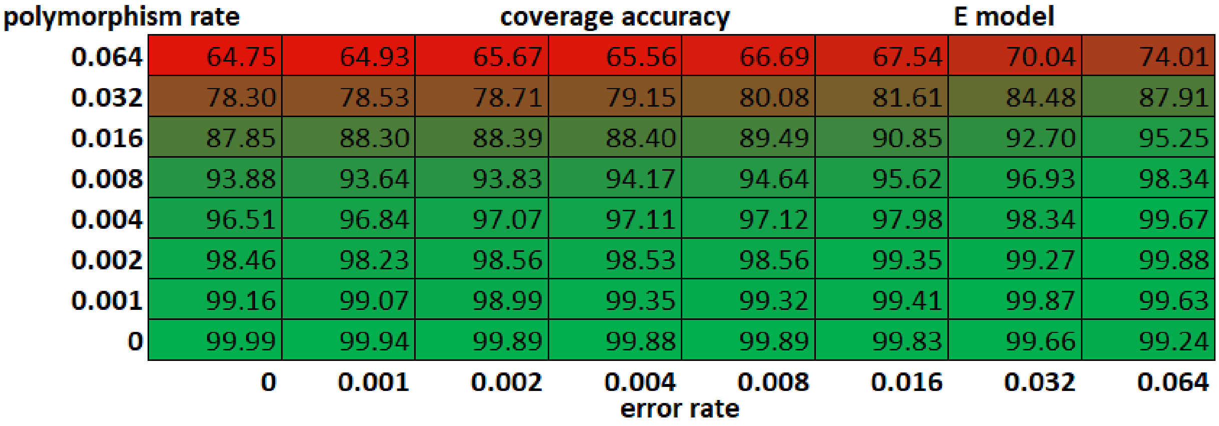
where θ is estimated value of parameter Θ and $\theta*$ is the real value of parameter Θ. Acc is then the percentage of how close estimation of Θ is to real Θ, 100 being the same value.

2.3 EP model

EP model

Keeping in mind our previous observations, we will create a prediction model Errors and polymorphisms (EP) with three parameters:

- Coverage C



polymorphism rate coverage accuracy E model

0.064	64.75	64.93	65.67	65.56	66.69	67.54	70.04	74.01
0.032	78.30	78.53	78.71	79.15	80.08	81.61	84.48	87.91
0.016	87.85	88.30	88.39	88.40	89.49	90.85	92.70	95.25
0.008	93.88	93.64	93.83	94.17	94.64	95.62	96.93	98.34
0.004	96.51	96.84	97.07	97.11	97.12	97.98	98.34	99.67
0.002	98.46	98.23	98.56	98.53	98.56	99.35	99.27	99.88
0.001	99.16	99.07	98.99	99.35	99.32	99.41	99.87	99.63
0	99.99	99.94	99.89	99.88	99.89	99.83	99.66	99.24
	0	0.001	0.002	0.004	0.008	0.016	0.032	0.064

error rate

Table 2.2: Mean accuracy of coverage estimation in percent from 10 samples estimated by E model for C = 4 and various sets of coverage and polymorphism rate parameters in samples. Standard deviation is 4.86%. Green color denotes higher accuracy.

- Error rate ϵ
- Proportion of distinct polymorphic k-mers in reads to all distinct k-mers in reads γ

Let n_n be a number of distinct non-polymorphic k-mers in reads. Let n_p be a number of distinct non-polymorphic k-mers in reads. Then:

$$\gamma = \frac{n_p}{n_n + n_p} \tag{2.9}$$

We will note probability distribution of E model with parameters C_k and ϵ as E(C_k,ϵ).

Our EP model determines size p_j of histogram class j as:

$$p_j = (1-\gamma)p_{n,j} + \gamma p_{p,j} \tag{2.10}$$

where

- $p_{n,j}$: probability of abundance j from E(C_k,ϵ),
- $p_{p,j}$: probability of abundance j from E($C_k/2$,ϵ).

Estimation of p_j is divided into two separate non-polymorphic and polymorphic parts by parameter γ (describing how many k-mers are non-polymorphic and polymorphic).

We have programmed EP model to CovEst as a new model. Starting value of γ is 5%.

Polymorphism rate prediction

By results in subsection 2.4, our model can be used to estimate the coverage and error rate from sequencing data, and consequently, the genome size. However, parameter γ that predicts ratio of polymorphic k-mers in reads including erroneous polymorphic k-mers is not a useful parameter on its own regarding information it provides to the user. Nevertheless, we can use it to further estimate the frequency of polymorphic positions in genome.

Probability of a k-mer containing one or more polymorphic positions p' is:

$$p' = \sum_{i=1}^{k} \binom{k}{i} \pi^i (1-\pi)^{k-i} = 1 - (1-\pi)^k \tag{2.11}$$

where:

- p' = probability that k-mer contains at least one polymorphic position,
- π = polymorphism rate in the genome (heterozygosity)

To predict polymorphism rate p' and π, we use estimated parameters C_k and ϵ to reconstruct E($C_k/2$,ϵ) distribution that contains only the abundance of polymorphic k-mers. Let $X_{p,a}$ be the number of all polymorphic k-mers in input reads. We can estimate $X_{p,a}$ as:

$$X_{p,a} = \gamma W_d \sum_{i=1}^{\infty} Z_i \cdot i \tag{2.12}$$

where:

- W_d: the number of distinct k-mers in reads (this quantity is known from input data),
- Z_i = probability of k-mer abundance i from E($C_k/2$,ϵ)

The second equation computes $X_{p,a}$ from the genome size as follows:

$$X_{p,a} = 2p'G\frac{C_k}{2} \tag{2.13}$$

where:

- p' = probability of a haploid k-mer being polymorphic
- G = genome size

The idea behind the second equation is that out of all haploid k-mers (approximated by the genome size), polymorphic k-mers are covered on average $C_k/2$ times. Due to the fact that one haploid k-mer overlapping a polymorphic position creates two distinct polymorphic k-mers, we need to multiply the number of polymorphic k-mers by 2.

By combining estimates 2.12 and 2.13, we obtain the following equation:

$$\gamma W_d \cdot \sum_{i=1}^{\infty} Z_i \cdot i = 2p'G\frac{C_k}{2} \tag{2.14}$$

From equation 2.14, we define p' as follows:

$$p' = \frac{\gamma W_d}{\sum_{i=1}^{t} P_i i G C_k} \tag{2.15}$$

We then use p' to find polymorphism rate π using 2.11:

$$\pi = 1 - (1 - p')^{1/k} \tag{2.16}$$

We estimate the infinite sum by a finite sum up to a threshold t. In CovEst, we have set t to half the size of the input histogram because the mean polymorphic coverage is half the mean of non-polymorphic coverage. Nevertheless, changes in polymorphism rate prediction accuracy for any t between half the size of input histogram and size of input histogram are insignificant by our observation.

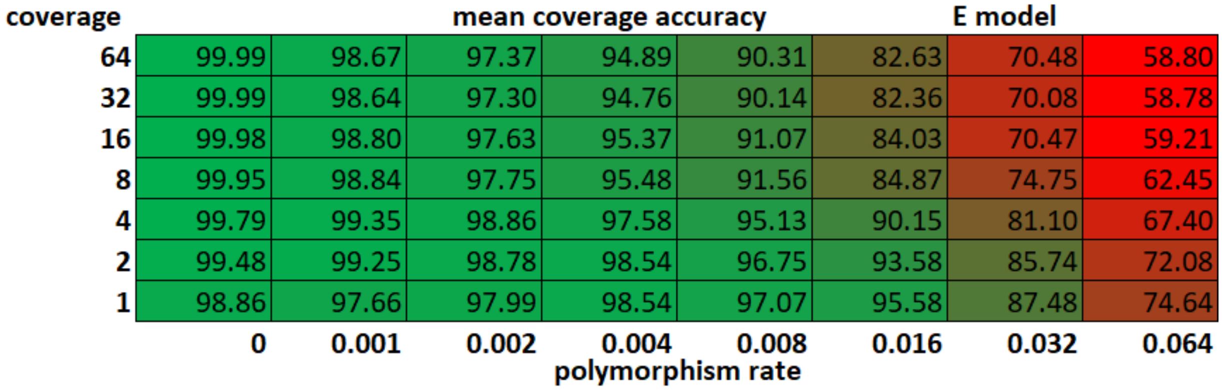
2.4 Results

We have evaluated our EP model on simulated data with various parameters settings together with E model and GenomeScope [17], the newest diploid genome attributes prediction tool up to date. For every set of parameters we created ten simulated genomes and reads. Genome is simulated as a random string of bases. Polymorphism simulation then

copies the genome and places random bases on random positions. Reads are simulated by randomly choosing one chromosome of diploid pair, randomly choosing a position and creating a read with changed bases with respect to set error rate.
Let $B = \{2^b | b \in \{0..6\}\}$. Tested coverage values were elements of B and tested error rate and polymorphism rate values were $\{0\} \cup \{0.001 \cdot b | b \in B\}$. Simulated genomes sizes were set to one million bases.

For the E model, the overall accuracy of parameter ϵ is 99.91% with standard deviation 0.12% and for the EP model, overall accuracy of ϵ is 99.95% with standard deviation 0.11%. As the accuracy of this parameter is very high for both models, we summarize results for all values of this parameter and report the mean over every error rate value for a given coverage and polymorphism rate. Tables for every set of parameters can be seen in attached CD, file *simulations_for_ep.xlsx*.

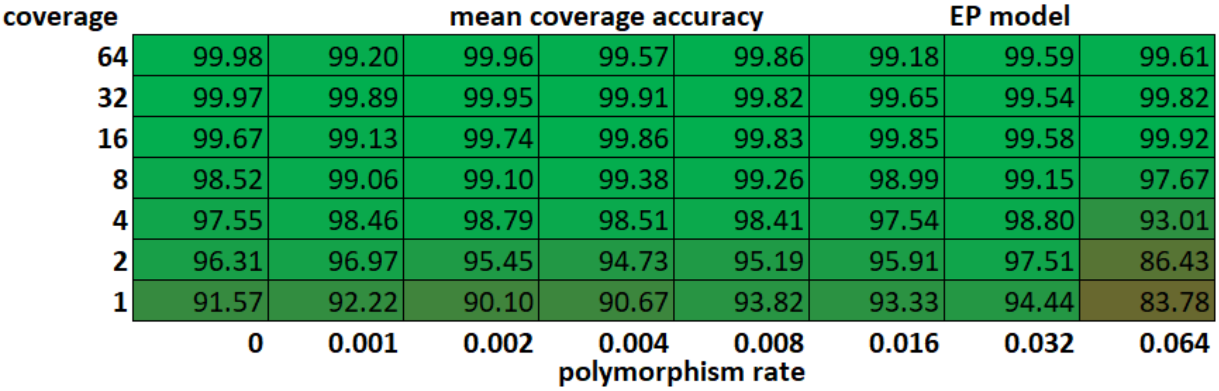
As we can see in Table 2.3, E model can estimate coverage with high accuracy for data with low-coverage and low polymorphism rate. However, with rising real coverage, E model coverage estimation quickly falls down between 4-fold and 8-fold coverage at still relatively low polymorphism rate 1.6%. For higher coverage and polymorphism rate, E model proves to be insufficient.
EP model (seen in tables 2.4,2.5), designed to estimate parameters from data containing polymorphic positions, has coverage estimation accuracies equal or higher to E model at 8-fold real coverage and higher. For lower depth of coverage, there is not enough polymorphic k-mers to accurately predict parameter γ, e.g. for $C = 4$, C_k is 3.2, subsequently polymorphic k-mer coverage is 1.6 meaning that approximately every second polymorphic k-mer is sequenced both times. This further creates a situation where EP model consider these polymorphic k-mers as non-polymorphic and as a part of haploid genome.
For higher depths of coverage, EP model shows no problem in identifying polymorphic k-mers, resulting in coverage estimations close to actual value.
Estimation of polymorphism rate by EP model has similar challenges for low-coverage data as it has for coverage estimation. Nevertheless, for depth of coverage 8 and higher, our EP model estimation accuracy is near to hundred percent.



coverage			mean coverage accuracy				E model	
64	99.99	98.67	97.37	94.89	90.31	82.63	70.48	58.80
32	99.99	98.64	97.30	94.76	90.14	82.36	70.08	58.78
16	99.98	98.80	97.63	95.37	91.07	84.03	70.47	59.21
8	99.95	98.84	97.75	95.48	91.56	84.87	74.75	62.45
4	99.79	99.35	98.86	97.58	95.13	90.15	81.10	67.40
2	99.48	99.25	98.78	98.54	96.75	93.58	85.74	72.08
1	98.86	97.66	97.99	98.54	97.07	95.58	87.48	74.64
	0	0.001	0.002	0.004	0.008	0.016	0.032	0.064
				polymorphism rate				

Table 2.3: Mean accuracy of coverage estimation in percent estimated by E model for various sets of coverage and polymorphism rate parameters in samples.

Comparison to web-based GenomeScope tool shows to be difficult (tables 2.6,2.7). GenomeScope tool, by examples provided by its authors, expects depth of coverage at least 15. Our tests show that their tool will not work at all with our data having coverage up to 16 - GenomeScope expects the input histogram to be wider (meaning there are at least 50 abundance classes). For depth of coverage 32, only a small fraction of samples had been actually processed - most of samples either have fewer than 50 abundance classes or their model simply did not converge. Even for coverage equal to 64 the impossibility of model to converge occurred. From small a amount of results we have managed to obtain we can clearly see that our EP model outperformed GenomeScope in both coverage prediction and polymorphism rate prediction. This situation further proves that their model is only useful for genomes with relatively high coverage and even for those our model yields better estimations. However, note that due to laborous providing of samples to the web-based tool without the possibility to create automatic script, only four samples from every set of samples with same parameters were tested in GenomeScope.

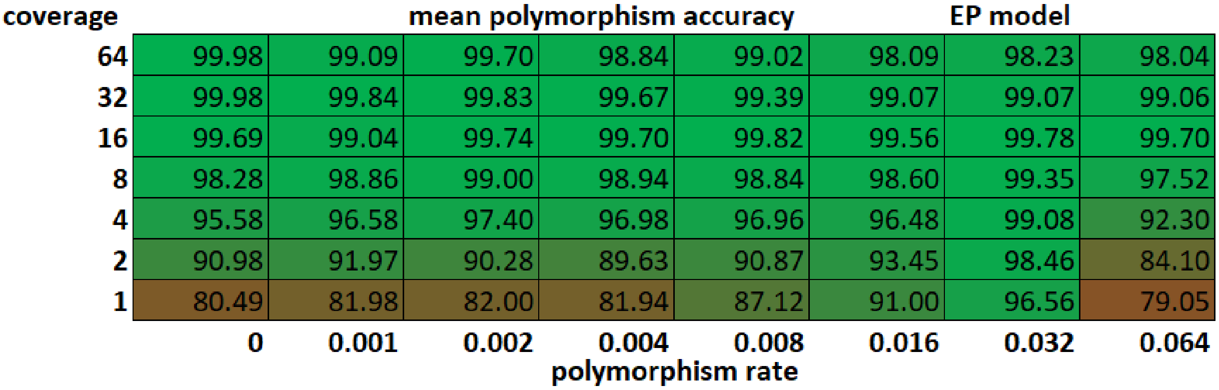


coverage mean coverage accuracy EP model

64	99.98	99.20	99.96	99.57	99.86	99.18	99.59	99.61
32	99.97	99.89	99.95	99.91	99.82	99.65	99.54	99.82
16	99.67	99.13	99.74	99.86	99.83	99.85	99.58	99.92
8	98.52	99.06	99.10	99.38	99.26	98.99	99.15	97.67
4	97.55	98.46	98.79	98.51	98.41	97.54	98.80	93.01
2	96.31	96.97	95.45	94.73	95.19	95.91	97.51	86.43
1	91.57	92.22	90.10	90.67	93.82	93.33	94.44	83.78
	0	0.001	0.002	0.004	0.008	0.016	0.032	0.064

polymorphism rate

Table 2.4: Mean accuracy of coverage estimation in percent estimated by EP model for various sets of coverage and polymorphism rate parameters in samples.

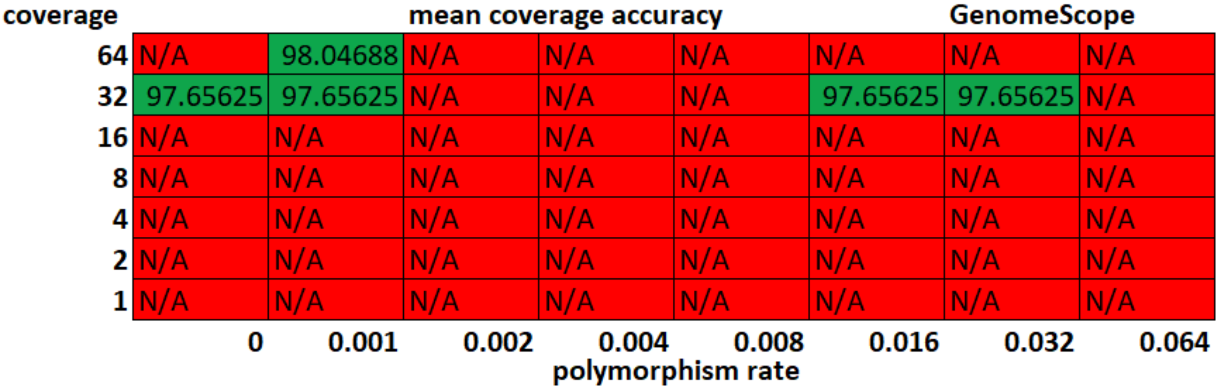


coverage mean polymorphism accuracy EP model

64	99.98	99.09	99.70	98.84	99.02	98.09	98.23	98.04
32	99.98	99.84	99.83	99.67	99.39	99.07	99.07	99.06
16	99.69	99.04	99.74	99.70	99.82	99.56	99.78	99.70
8	98.28	98.86	99.00	98.94	98.84	98.60	99.35	97.52
4	95.58	96.58	97.40	96.98	96.96	96.48	99.08	92.30
2	90.98	91.97	90.28	89.63	90.87	93.45	98.46	84.10
1	80.49	81.98	82.00	81.94	87.12	91.00	96.56	79.05
	0	0.001	0.002	0.004	0.008	0.016	0.032	0.064

polymorphism rate

Table 2.5: Mean polymorphism accuracy of polymorphism rate estimation in percent estimated by EP model for various sets of coverage and polymorphism rate parameters in samples.

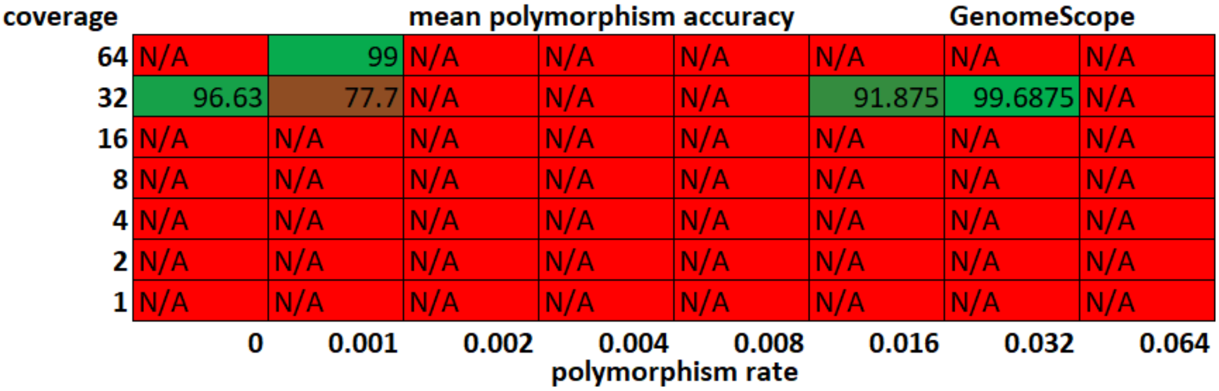


coverage mean coverage accuracy GenomeScope

64	N/A	98.04688	N/A	N/A	N/A	N/A	N/A	N/A
32	97.65625	97.65625	N/A	N/A	N/A	97.65625	97.65625	N/A
16	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
8	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
4	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
2	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
1	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
	0	0.001	0.002	0.004	0.008	0.016	0.032	0.064

polymorphism rate

Table 2.6: Mean accuracy of coverage estimation in percent estimated by GenomeScope for various sets of coverage and polymorphism rate parameters in samples.



mean polymorphism accuracy GenomeScope

coverage	0	0.001	0.002	0.004	0.008	0.016	0.032	0.064
64	N/A	99	N/A	N/A	N/A	N/A	N/A	N/A
32	96.63	77.7	N/A	N/A	N/A	91.875	99.6875	N/A
16	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
8	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
4	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
2	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
1	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A

polymorphism rate

Table 2.7: Mean accuracy of coverage estimation in percent estimated by GenomeScope for various sets of coverage and polymorphism rate parameters in samples.

Chapter 3

Extended repeats and errors model

The Repeats and errors (RE) model in CovEst captures k-mers with more occurences in genome. In this chapter we will discuss extensions of the RE model. In particular, we will evaluate RE with additional parameters, we will discuss actual meaning of β_o fractions in this model and we will introduce new algorithm to estimate the true value of desired fractions β_o described by Hozza et al. [16].

3.1 RE model

The Repeats and errors model (RE) is an extension of the E model by Hozza et al. [16] that incorporate existence of several identical k-mers in genome. Parameters that this model estimates are:

- Coverage C
- Error rate ϵ
- Parameters for repeat rate estimation q_1, q_2, q

We will define term *copy number* as number of occurrences of k-mer in genome.

Let β_o be the fraction of k-mers with copy number o following geometrical distribution.

The probability p_j of each abundance class j is:

$$p_j = \sum_{o=1}^{\infty} \beta_o p_{o,j} \tag{3.1}$$

where $p_{o,j}$ is the probability computed according to E model with $o \cdot C_k$ coverage as the k-mer with copy number o is o times more likely to be sequenced.

The infinite number of β_o weights is parametrized by three parameters q_1, q_2 and q as follows:

- $\beta_1 = q_1$
- $\beta_2 = (1 - q_1)q_2$
- $\beta_o = (1 - q_1)(1 - q_2)q(1 - q)^{o-3}; o \geq 3$

This means that this model estimates β_1 and β_2 separately and repeats classes with at least three occurences together, geometrically decreasing β_o estimations for higher copy numbers.

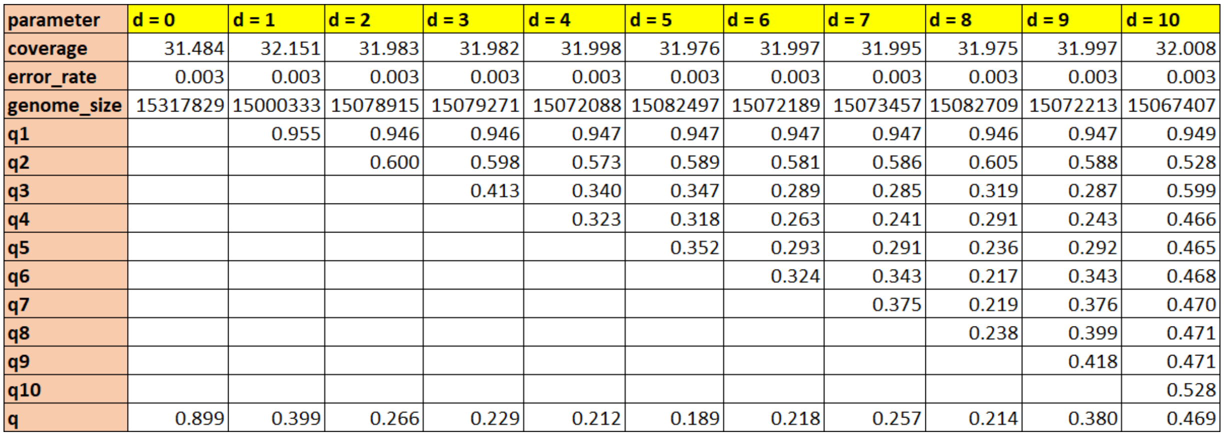
3.2 Extended RE model

In this section we extend RE (ERE_d) model by introducing d as the number of repeats parameters. Fractions β_o will be parametrized by $q, q_1 \dots q_d$, if $d = 0$ we use only q. Purpose of new parameters is to further increase accuracy by separately optimizing more copy number classes. β_o are in ERE_d model computed as follows:

$$\beta_o = \begin{cases} q_1 & \text{if } o = 1 \text{ and } d > 0 \\ q_o \prod_{i=1}^{o-1}(1 - q_i) & \text{if } o > 1 \text{ and } o \leq d \\ q(1 - q)^{o-d-1} \prod_{i=1}^{o}(1 - q_i) & \text{if } o > d \end{cases}$$

ERE_d model computes probability p_j for abundance class j with the same equation as RE model.

As Hozza et al. [16] claim, the number of repeat parameters $d = 2$ in the RE model was set ad-hoc with only a brief analysis of available genomes.



parameter	d = 0	d = 1	d = 2	d = 3	d = 4	d = 5	d = 6	d = 7	d = 8	d = 9	d = 10
coverage	31.484	32.151	31.983	31.982	31.998	31.976	31.997	31.995	31.975	31.997	32.008
error_rate	0.003	0.003	0.003	0.003	0.003	0.003	0.003	0.003	0.003	0.003	0.003
genome_size	15317829	15000333	15078915	15079271	15072088	15082497	15072189	15073457	15082709	15072213	15067407
q1		0.955	0.946	0.946	0.947	0.947	0.947	0.947	0.946	0.947	0.949
q2			0.600	0.598	0.573	0.589	0.581	0.586	0.605	0.588	0.528
q3				0.413	0.340	0.347	0.289	0.285	0.319	0.287	0.599
q4					0.323	0.318	0.263	0.241	0.291	0.243	0.466
q5						0.352	0.293	0.291	0.236	0.292	0.465
q6							0.324	0.343	0.217	0.343	0.468
q7								0.375	0.219	0.376	0.470
q8									0.238	0.399	0.471
q9										0.418	0.471
q10											0.528
q	0.899	0.399	0.266	0.229	0.212	0.189	0.218	0.257	0.214	0.380	0.469

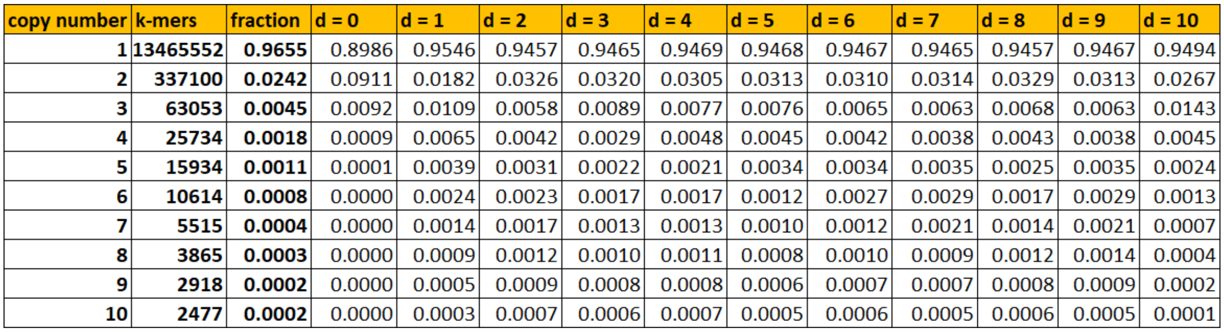
Table 3.1: Results of $ERE_d | d \in \{0 \dots 10\}$ on simulated reads of Caenorhabditis elegans chromosome I, the actual depth of coverage is 32, error rate is 0.3%, genome size is 15072434 bases.

We have analyzed several real-life genomes from both haploid and diploid organisms (more specifically in diploid organisms, one of their haploid sets). Next we have simulated reads with various coverage and error rates from analyzed genomes. As a following step, we have optimized $ERE_d; d \in \{1 \dots 10\}$ models on genomes reads.
As an example, we show results of two experiments on diploid *Caenorhabditis elegans* chromosome I. Other experiments, that can be seen in attached CD - file *ered_estimations.xlsx*, yield similar results.

As can be seen in Table 3.1, optimizing ERE_d model with $d \geq 2$ gives no significant increase in accuracy of coverage or error rate.

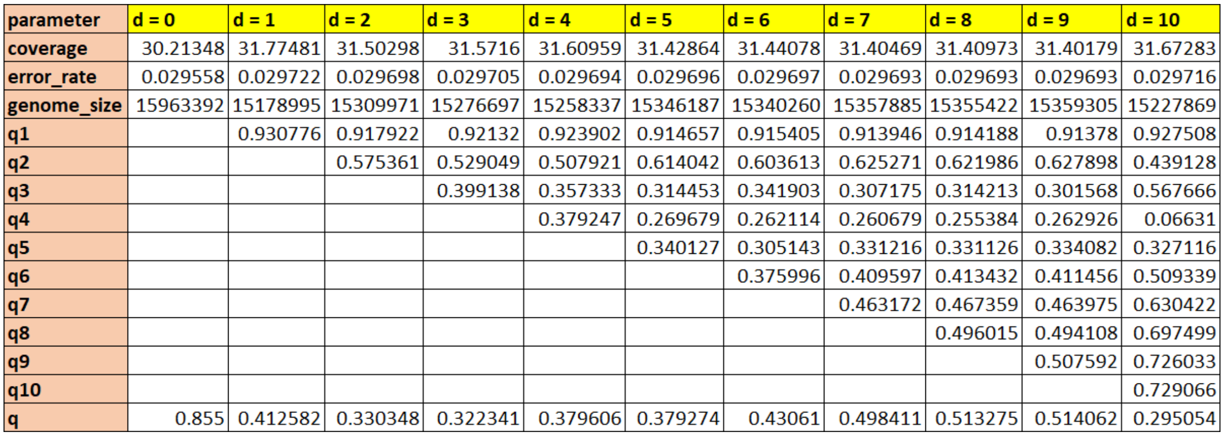
However, we have noticed in Table 3.2 that values of β_o computed from k-mer repeat parameters do not correspond to actual fractions of repeating k-mers in the genome.

As Hozza et al. [16] states: *Let β_o be the fraction of k-mers with o occurences in the genome.* However, as RE model works with k-mers in reads and not with k-mers in the genome, we claim that β_o will be ratio of distinct k-mers in reads originating from genomic k-mers with o copies in genome to all distinct k-mers. This includes erroneous k-mers that increase number of distinct k-mers in reads. If Hozza et al.[16] claim was true, we should get the same β_o for a higher error rate as we got in our previous experiment. We have



copy number	k-mers	fraction	d = 0	d = 1	d = 2	d = 3	d = 4	d = 5	d = 6	d = 7	d = 8	d = 9	d = 10
1	**13465552**	**0.9655**	0.8986	0.9546	0.9457	0.9465	0.9469	0.9468	0.9467	0.9465	0.9457	0.9467	0.9494
2	**337100**	**0.0242**	0.0911	0.0182	0.0326	0.0320	0.0305	0.0313	0.0310	0.0314	0.0329	0.0313	0.0267
3	**63053**	**0.0045**	0.0092	0.0109	0.0058	0.0089	0.0077	0.0076	0.0065	0.0063	0.0068	0.0063	0.0143
4	**25734**	**0.0018**	0.0009	0.0065	0.0042	0.0029	0.0048	0.0045	0.0042	0.0038	0.0043	0.0038	0.0045
5	**15934**	**0.0011**	0.0001	0.0039	0.0031	0.0022	0.0021	0.0034	0.0034	0.0035	0.0025	0.0035	0.0024
6	**10614**	**0.0008**	0.0000	0.0024	0.0023	0.0017	0.0017	0.0012	0.0027	0.0029	0.0017	0.0029	0.0013
7	**5515**	**0.0004**	0.0000	0.0014	0.0017	0.0013	0.0013	0.0010	0.0012	0.0021	0.0014	0.0021	0.0007
8	**3865**	**0.0003**	0.0000	0.0009	0.0012	0.0010	0.0011	0.0008	0.0010	0.0009	0.0012	0.0014	0.0004
9	**2918**	**0.0002**	0.0000	0.0005	0.0009	0.0008	0.0008	0.0006	0.0007	0.0007	0.0008	0.0009	0.0002
10	**2477**	**0.0002**	0.0000	0.0003	0.0007	0.0006	0.0007	0.0005	0.0006	0.0005	0.0006	0.0005	0.0001

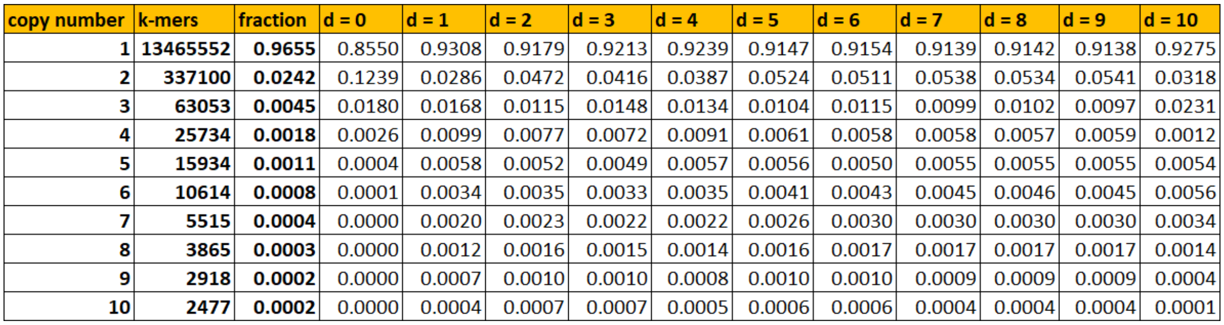
Table 3.2: The values of β_o for $o = 1 \ldots 10$ for ERE_d ($d \in \{1 \ldots 10\}$) computed from q and q_i shown in Table 3.1. The first columns contain copy number class, number of distinct k-mers in class and fraction of number of distinct k-mers in class to number of all distinct k-mers.



parameter	d = 0	d = 1	d = 2	d = 3	d = 4	d = 5	d = 6	d = 7	d = 8	d = 9	d = 10
coverage	30.21348	31.77481	31.50298	31.5716	31.60959	31.42864	31.44078	31.40469	31.40973	31.40179	31.67283
error_rate	0.029558	0.029722	0.029698	0.029705	0.029694	0.029696	0.029697	0.029693	0.029693	0.029693	0.029716
genome_size	15963392	15178995	15309971	15276697	15258337	15346187	15340260	15357885	15355422	15359305	15227869
q1		0.930776	0.917922	0.92132	0.923902	0.914657	0.915405	0.913946	0.914188	0.91378	0.927508
q2			0.575361	0.529049	0.507921	0.614042	0.603613	0.625271	0.621986	0.627898	0.439128
q3				0.399138	0.357333	0.314453	0.341903	0.307175	0.314213	0.301568	0.567666
q4					0.379247	0.269679	0.262114	0.260679	0.255384	0.262926	0.06631
q5						0.340127	0.305143	0.331216	0.331126	0.334082	0.327116
q6							0.375996	0.409597	0.413432	0.411456	0.509339
q7								0.463172	0.467359	0.463975	0.630422
q8									0.496015	0.494108	0.697499
q9										0.507592	0.726033
q10											0.729066
q	0.855	0.412582	0.330348	0.322341	0.379606	0.379274	0.43061	0.498411	0.513275	0.514062	0.295054

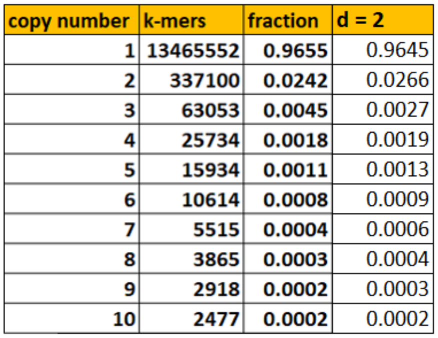
Table 3.3: Results of $ERE_d|d \in \{0 \ldots 10\}$ on simulated reads of Caenorhabditis elegans chromosome I, depth of coverage is 32, error rate is 3%, genome size is 15072434 bases.

run ERE_d models on reads with the same coverage, but this time, error rate was set to 3%. Optimized parameters can be seen in Table 3.3 and values of β_o for this experiment are shown in Table 3.4. Comparing β_o from these two experiments, we can see that they differ and thus, the claim by Hozza et al. [16] is invalid. We will conduct one additional experiment supporting our definition of β_o. If reads contain no errors, the resulting β_o corresponds to k-mer repeats in the genome, because in the absence of errors and with sufficient coverage, the set of k-mers in reads and in the genome is practically the same. Table 3.5 confirms our claim.



copy number	k-mers	fraction	d = 0	d = 1	d = 2	d = 3	d = 4	d = 5	d = 6	d = 7	d = 8	d = 9	d = 10
1	**13465552**	**0.9655**	0.8550	0.9308	0.9179	0.9213	0.9239	0.9147	0.9154	0.9139	0.9142	0.9138	0.9275
2	**337100**	**0.0242**	0.1239	0.0286	0.0472	0.0416	0.0387	0.0524	0.0511	0.0538	0.0534	0.0541	0.0318
3	**63053**	**0.0045**	0.0180	0.0168	0.0115	0.0148	0.0134	0.0104	0.0115	0.0099	0.0102	0.0097	0.0231
4	**25734**	**0.0018**	0.0026	0.0099	0.0077	0.0072	0.0091	0.0061	0.0058	0.0058	0.0057	0.0059	0.0012
5	**15934**	**0.0011**	0.0004	0.0058	0.0052	0.0049	0.0057	0.0056	0.0050	0.0055	0.0055	0.0055	0.0054
6	**10614**	**0.0008**	0.0001	0.0034	0.0035	0.0033	0.0035	0.0041	0.0043	0.0045	0.0046	0.0045	0.0056
7	**5515**	**0.0004**	0.0000	0.0020	0.0023	0.0022	0.0022	0.0026	0.0030	0.0030	0.0030	0.0030	0.0034
8	**3865**	**0.0003**	0.0000	0.0012	0.0016	0.0015	0.0014	0.0016	0.0017	0.0017	0.0017	0.0017	0.0014
9	**2918**	**0.0002**	0.0000	0.0007	0.0010	0.0010	0.0008	0.0010	0.0010	0.0009	0.0009	0.0009	0.0004
10	**2477**	**0.0002**	0.0000	0.0004	0.0007	0.0007	0.0005	0.0006	0.0006	0.0004	0.0004	0.0004	0.0001

Table 3.4: First 10 β_o of $ERE_d|d \in \{0 \dots 10\}$ on simulated reads of Caenorhabditis elegans chromosome I, depth of coverage is 32, error rate is 3%, genome size is 15072434 bases.



copy number	k-mers	fraction	d = 2
1	**13465552**	**0.9655**	0.9645
2	**337100**	**0.0242**	0.0266
3	**63053**	**0.0045**	0.0027
4	**25734**	**0.0018**	0.0019
5	**15934**	**0.0011**	0.0013
6	**10614**	**0.0008**	0.0009
7	**5515**	**0.0004**	0.0006
8	**3865**	**0.0003**	0.0004
9	**2918**	**0.0002**	0.0003
10	**2477**	**0.0002**	0.0002

Table 3.5: β_o of ERE_3 on simulated reads of Caenorhabditis elegans chromosome I, depth of coverage is 32, error rate is 0%, genome size is 15072434 bases. Estimated coverage is 31.9820, estimated error rate is 0.0%.

ERE_d	0	1	2	3	4	5	6	7	8	9	10
time	15.22	16.49	30.54	37.11	45.39	59.22	70.23	71.46	77.09	85.45	91.34

Table 3.6: Mean running time of ERE_d models in seconds for various parameters settings of Caenorhabditis elegans chromosome I data.

Back on our original question if adding more repeats parameters will improve accuracy of estimations, our answer, based on our experiments, is that improvement is insignificant and as can be seen on Table 3.6, optimization time is rising with more repeat parameters.

3.3 Genome repeats estimation

Algorithm

As we have stated before, our experiments show that β_o are in fact a ratio of distinct k-mers with copy number o in the genome to all distinct k-mers in reads. We would like to extend the RE model to estimate the fraction of k-mers with o copies in genome among all k-mers in the genome, which we denote $\beta_{G,o}$. To make the distinction between $\beta_{G,o}$ and β_o more explicit, we will further denote β_o as $\beta_{R,o}$. We will also define several variables:

- W_d: the number of all distinct k-mers in reads,
- G'_a: the number of all k-mers in genome,
- G'_d: the number of all distinct k-mers in genome.

Variable W_d is known from the input abundance spectrum and we set $G'_a = G$, where G is the genome size we know from previous estimation (in fact, G is higher than G'_a due to chromosomal ends, however the difference is negligible for $\beta_{G,o}$ estimation).

The idea of our algorithm resembles the estimation of polymorphism rate from γ in the EP model. The first step is to create histograms Z_o with $\mathrm{E}(o \cdot C_k, \epsilon)$ $o \in \{1 \ldots m\}$ for some value m so that remaining $\beta_{G,o}$ are negligibly small. If the highest copy number m in the genome is known, we can use it as m. Value $Z_{o,i}$ denotes the k-mer abundance probability

p_j for abundance class j with copy number o. Multiplying this probability by $\beta_{R,o}W_di$ gives us the expected number of all k-mers in reads with abundance i and copy number o, which we denote $X_{a,o}$. Sum through all classes of abundance gives us number of all k-mers with copy number o in reads. Value $\beta_{R,o}W_d$ is the expected number of all distinct k-mers with copy number o in reads. This computation gives us $X_{a,o}$ as follows:

$$X_{a,o} = \sum_{i=1}^{\infty} Z_{o,i} \cdot i \cdot \beta_{R,o} \cdot W_d \tag{3.2}$$

On the other hand, we can compute $X_{a,o}$ from the genome size as well. $\beta_{G,o}G'_d$ is the expected number of distinct k-mers with copy number o in the genome. Multiplying $\beta_{G,o}G'_d$ by o gives us all k-mers with copy number o in genome. We then multiply this number by k-mer coverage C_k to obtain $X_{a,o}$, as every k-mer is sequenced C_k times on average. $X_{a,o}$ is therefore computed as:

$$X_{a,o} = \beta_{G,o} \cdot G'_d \cdot o \cdot C_k \tag{3.3}$$

Value of G'_d in 3.3 is not known, we only have an estimate of G'_a. However, $\beta_{G,o}G'_d$ is the number of distinct k-mers with copy number o in the genome. Multiplying this by o yields the number of all k-mers with copy number o in the genome and sum through all copy number classes o is the number of all k-mers in genome G'_a:

$$G'_a = \sum_{o=1}^{\infty} \beta_{G,o} \cdot o \cdot G'_d \tag{3.4}$$

As we already know G'_a, we can express G'_d as:

$$G'_d = \frac{G'_a}{\sum_{o=1}^{\infty} \beta_{G,o} \cdot o} \tag{3.5}$$

Combining equations 3.2 and 3.3 through $X_{a,o}$, we get the central equation of our algorithm:

$$\beta_{G,o} \cdot o \cdot G'_d \cdot C_k = \sum_{i=1}^{\infty} Z_{o,i} \cdot i \cdot \beta_{R,o} \cdot W_d \tag{3.6}$$

copy number	**k-mers**	**real** $\beta_{G,o}$	**estimated** $\beta_{G,o}$	**real** $\beta_{R,o}$	**estimated** $\beta_{R,o}$
1	4999970	0.8333	0.8327	0.7872	0.8174
2	999985	0.1667	0.1673	0.2128	0.1826

Table 3.7: Predicted $\beta_{G,o}$ from a simulated genome, genome size is 7 million bases, depth of coverage is 32, error rate is 0.1%, m = 2.

For simplicity, we will note the right side of equation as $X_{a,o}$ (this side is completely known). By expressing $\beta_{G,o}$, we get:

$$\beta_{G,o} \cdot o \cdot G'_a \cdot C_k = X_{a,o} \sum_{o'=1}^{m} \beta_G, o' \cdot o' \tag{3.7}$$

This is, in fact, $m \times m$ system of linear equations. We will append one more equation to our system:

$$\sum_{o=1}^{m} \beta_{G,o} = 1 \tag{3.8}$$

Equation 3.8 is a natural condition that fractions $\beta_{G,o}$ sum to 1. However, appending this equation creates an overdetermined system of linear equations with m unknowns and $m + 1$ equations. We have approximated the solution to this system using least-square method [25] as implemented in SciPy library. [23]

Results

We have conducted experiments with our genome repeat estimation algorithm on several simulated and real genomes.

Our first experiment is with a simulated genome containing only unique and duplicated k-mers ($m = 2$) and 7 million bases in size. Generated reads have depth of coverage 32 and 0.1% error rate. Table 3.7 shows the results of the experiment. Estimation of $\beta_{G,1}$ and $\beta_{G,2}$ is close to real values.

repeats	**k-mers**	**real** $\beta_{G,o}$	**estimated** $\beta_{G,o}$	**estimated** $\beta_{G,o}$ **from real** $\beta_{R,o}$	real $\beta_{R,o}$	**estimated** $\beta_{R,o}$
1	7999933	0.7273	0.7621	0.7275	0.6471	0.6970
2	1999986	0.1818	0.1599	0.1816	0.2169	0.1950
3	999986	0.0909	0.0720	0.0909	0.1361	0.1080

Table 3.8: Predicted $\beta_{G,o}$ from simulated genome, genome size is 15 million bases, depth of coverage is 32, error rate is 0.1%, m = 3.

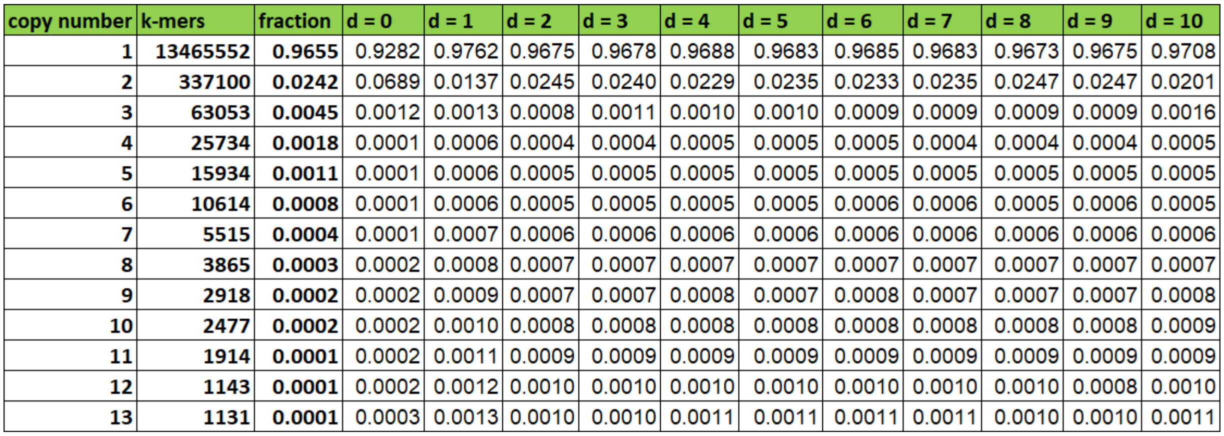
Our second experiment is with simulated genome with k-mers occuring up to three times in genome and 15 million bases in size. Generated reads have depth of coverage 32 and 0.1% error rate. Table 3.8 shows us results of the experiment. Estimations of $\beta_{G,o}$ from predicted $\beta_{R,o}$ by RE model differ significantly (up to three percent). However, using our algorithm with real $\beta_{R,o}$ parameters, we get estimation of $\beta_{G,o}$ close to real values.

Third experiment we conducted is with Caenorhabditis elegans chromosome I, with 32-fold coverage and 0.3% error rate and with $\beta_{R,o}$ from 3.2. Results of this experiment are in Table 3.9. Results show that estimations of $\beta_{G,o}$ are correct for copy number 1 and 2 classes. However, starting from copy number 3, our estimates start to differ from the actual values. As our previous experiments show, our algorithm yields good results on known m and correct $\beta_{R,o}$. In this experiment, we have set $m = 13$ on the basis that it is higher than highest used $d = 10$, so an effect of estimation from geometric distribution of $\beta_{R,o}$ would manifest.

Different values of m significantly affect fractions of each $\beta_{G,o}$ as can be seen in Table 3.10 created from ERE_5.

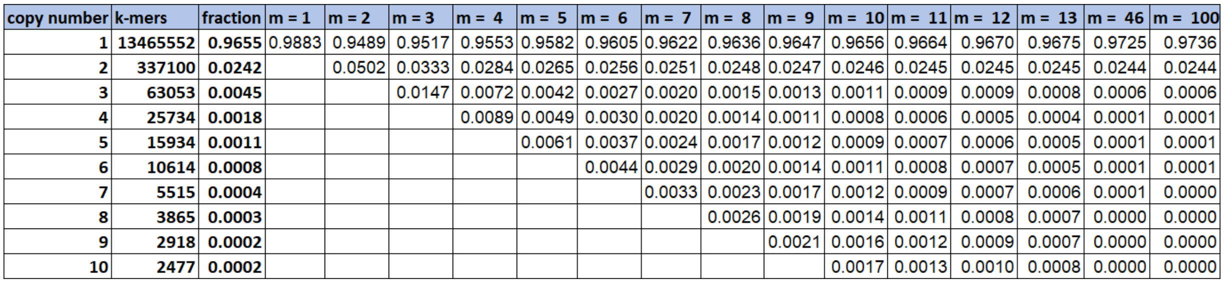
Other results can be seen in attached CD, file *ered_estimations.xlsx*.

Even if our estimation of $\beta_{G,o}$ is not optimal, it still outperforms $\beta_{R,o}$ at denoting k-mer repeats in genome as can be seen by comparing Tables 3.1 and 3.9.



copy number	k-mers	fraction	d = 0	d = 1	d = 2	d = 3	d = 4	d = 5	d = 6	d = 7	d = 8	d = 9	d = 10
1	**13465552**	**0.9655**	0.9282	0.9762	0.9675	0.9678	0.9688	0.9683	0.9685	0.9683	0.9673	0.9675	0.9708
2	**337100**	**0.0242**	0.0689	0.0137	0.0245	0.0240	0.0229	0.0235	0.0233	0.0235	0.0247	0.0247	0.0201
3	**63053**	**0.0045**	0.0012	0.0013	0.0008	0.0011	0.0010	0.0010	0.0009	0.0009	0.0009	0.0009	0.0016
4	**25734**	**0.0018**	0.0001	0.0006	0.0004	0.0004	0.0005	0.0005	0.0005	0.0004	0.0004	0.0004	0.0005
5	**15934**	**0.0011**	0.0001	0.0006	0.0005	0.0005	0.0005	0.0005	0.0005	0.0005	0.0005	0.0005	0.0005
6	**10614**	**0.0008**	0.0001	0.0006	0.0005	0.0005	0.0005	0.0005	0.0006	0.0006	0.0005	0.0006	0.0005
7	**5515**	**0.0004**	0.0001	0.0007	0.0006	0.0006	0.0006	0.0006	0.0006	0.0006	0.0006	0.0006	0.0006
8	**3865**	**0.0003**	0.0002	0.0008	0.0007	0.0007	0.0007	0.0007	0.0007	0.0007	0.0007	0.0007	0.0007
9	**2918**	**0.0002**	0.0002	0.0009	0.0007	0.0007	0.0008	0.0007	0.0008	0.0007	0.0007	0.0007	0.0008
10	**2477**	**0.0002**	0.0002	0.0010	0.0008	0.0008	0.0008	0.0008	0.0008	0.0008	0.0008	0.0008	0.0009
11	**1914**	**0.0001**	0.0002	0.0011	0.0009	0.0009	0.0009	0.0009	0.0009	0.0009	0.0009	0.0009	0.0009
12	**1143**	**0.0001**	0.0002	0.0012	0.0010	0.0010	0.0010	0.0010	0.0010	0.0010	0.0010	0.0008	0.0010
13	**1131**	**0.0001**	0.0003	0.0013	0.0010	0.0010	0.0011	0.0011	0.0011	0.0011	0.0010	0.0010	0.0011

Table 3.9: Results of estimated $\beta_{G,o}$ from $ERE_d | d \in \{0 \ldots 10\}$ on simulated reads of Caenorhabditis elegans chromosome I, depth of coverage is 32, error rate is 0.3%, genome size is 15072434 bases. M was set to 13.



copy number	k-mers	fraction	m = 1	m = 2	m = 3	m = 4	m = 5	m = 6	m = 7	m = 8	m = 9	m = 10	m = 11	m = 12	m = 13	m = 46	m = 100
1	**13465552**	**0.9655**	0.9883	0.9489	0.9517	0.9553	0.9582	0.9605	0.9622	0.9636	0.9647	0.9656	0.9664	0.9670	0.9675	0.9725	0.9736
2	**337100**	**0.0242**		0.0502	0.0333	0.0284	0.0265	0.0256	0.0251	0.0248	0.0247	0.0246	0.0245	0.0245	0.0245	0.0244	0.0244
3	**63053**	**0.0045**			0.0147	0.0072	0.0042	0.0027	0.0020	0.0015	0.0013	0.0011	0.0009	0.0009	0.0008	0.0006	0.0006
4	**25734**	**0.0018**				0.0089	0.0049	0.0030	0.0020	0.0014	0.0011	0.0008	0.0006	0.0005	0.0004	0.0001	0.0001
5	**15934**	**0.0011**					0.0061	0.0037	0.0024	0.0017	0.0012	0.0009	0.0007	0.0006	0.0005	0.0001	0.0001
6	**10614**	**0.0008**						0.0044	0.0029	0.0020	0.0014	0.0011	0.0008	0.0007	0.0005	0.0001	0.0001
7	**5515**	**0.0004**							0.0033	0.0023	0.0017	0.0012	0.0009	0.0007	0.0006	0.0001	0.0000
8	**3865**	**0.0003**								0.0026	0.0019	0.0014	0.0011	0.0008	0.0007	0.0000	0.0000
9	**2918**	**0.0002**									0.0021	0.0016	0.0012	0.0009	0.0007	0.0000	0.0000
10	**2477**	**0.0002**										0.0017	0.0013	0.0010	0.0008	0.0000	0.0000

Table 3.10: $\beta_{G,o}$ from various m and their effect on estimation accuracy of individual repeats classes.

Chapter 4

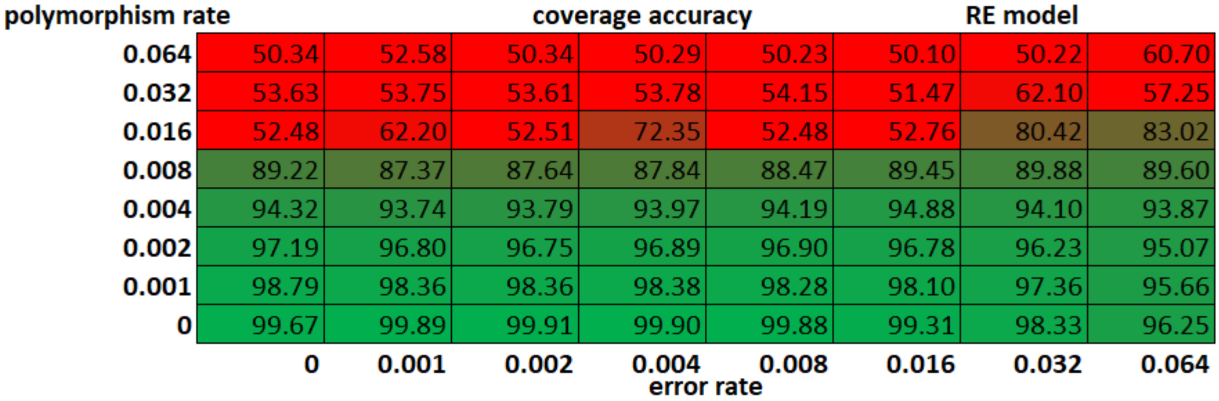
Repeats, errors and polymorphisms model

In this chapter, we will combine partial results from the previous two chapters and study genomes exhibing both repeats and polymorphisms. We will first test RE model on diploid sequencing data and we will create two more models containing repeats, errors and polymorphisms.

4.1 RE model on diploid data

We have tried the RE model on haploid chromosome V from Caenorhabditis elegans. We have created another copy of this chromosome and randomly modified bases in the copy to simulate polymorphism. Coverage, error rate and polymorphism rate were in this simulated cases the same as in EP model testing 2.4.

There are two reasons to try RE model on diploid data. First, we want to see and confirm that, similarly to EP model, RE model will underestimate the coverage as it has no means of processing polymorphic reads. Second, by our previous observation, RE model might still be able to model histogram with repeating, erroneous and polymorphic k-mers as a haploid genome of double the size and half the coverage. That is, non-repeat polymorphic k-mers correspond to $o = 1$ in the RE model and non-repeating, non-polymorphic k-mers to $o = 2$. Then we can simply multiply the coverage by two to obtain the actual size of



coverage accuracy RE model

polymorphism rate	0	0.001	0.002	0.004	0.008	0.016	0.032	0.064
0.064	50.34	52.58	50.34	50.29	50.23	50.10	50.22	60.70
0.032	53.63	53.75	53.61	53.78	54.15	51.47	62.10	57.25
0.016	52.48	62.20	52.51	72.35	52.48	52.76	80.42	83.02
0.008	89.22	87.37	87.64	87.84	88.47	89.45	89.88	89.60
0.004	94.32	93.74	93.79	93.97	94.19	94.88	94.10	93.87
0.002	97.19	96.80	96.75	96.89	96.90	96.78	96.23	95.07
0.001	98.79	98.36	98.36	98.38	98.28	98.10	97.36	95.66
0	99.67	99.89	99.91	99.90	99.88	99.31	98.33	96.25

error rate

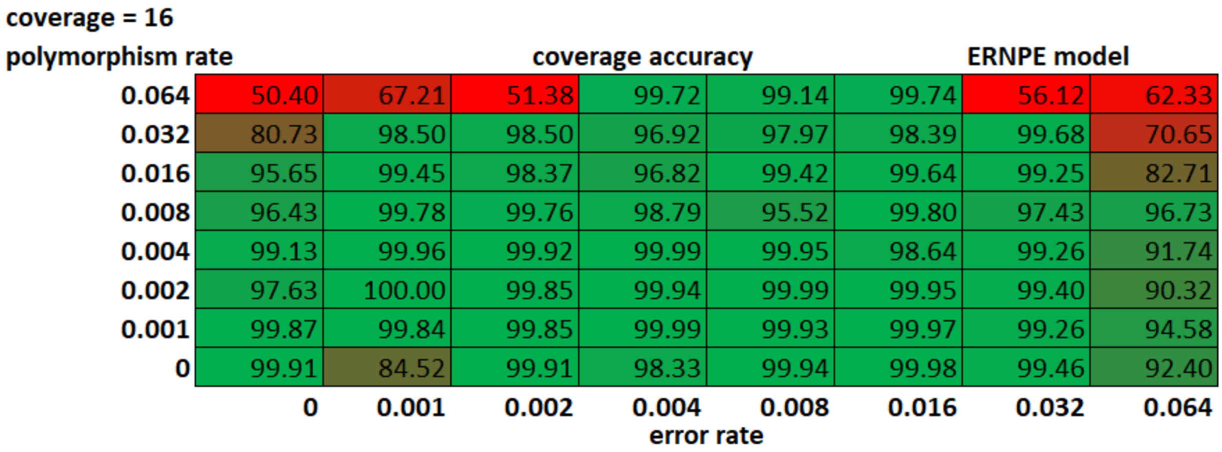
Table 4.1: Mean accuracy of coverage estimation in percent estimated by RE model for C = 16 and various sets of error rate and polymorphism rate parameters in samples.

genome. This approach will, similarly to E model on diploid data, yield no information on polymorphism rate. Again, to simplify our discussion, we will present only one table with results. Other similar results are in attached CD - file *simulations_for_ernpe.xlsx*.

As we can see in Table 4.1, RE model has relatively high accuracy of coverage estimation for small polymorphism rates ($o = 1$ corresponds to non-polymorphic k-mers). However, with increasing polymorphism rate accuracy of coverage estimations gradually falls and for high polymorphism reaches a sate where $o = 2$ corresponds to non-polymorphic k-mers. As the decrease is gradual, there is no distinctive line after which the coverage estimation should be doubled. Furthermore, even if there was this line, we still do not have any information whether coverage estimation in a particular dataset needs to be doubled or not.

4.2 ERNPE model

Our next prediction model *Equal Repeats for Non-polymorphism and Polymorphism and Error* (ERNPE) assumes, that if polymorphism occurs in repeated k-mer, all repeats of that k-mer will be polymorphic. Although this assumption is not realistic, it greatly simplifies modeling. We will again introduce a parameter γ with same meaning as in EP



coverage = 16

polymorphism rate — coverage accuracy — ERNPE model

polymorphism rate	0	0.001	0.002	0.004	0.008	0.016	0.032	0.064
0.064	50.40	67.21	51.38	99.72	99.14	99.74	56.12	62.33
0.032	80.73	98.50	98.50	96.92	97.97	98.39	99.68	70.65
0.016	95.65	99.45	98.37	96.82	99.42	99.64	99.25	82.71
0.008	96.43	99.78	99.76	98.79	95.52	99.80	97.43	96.73
0.004	99.13	99.96	99.92	99.99	99.95	98.64	99.26	91.74
0.002	97.63	100.00	99.85	99.94	99.99	99.95	99.40	90.32
0.001	99.87	99.84	99.85	99.99	99.93	99.97	99.26	94.58
0	99.91	84.52	99.91	98.33	99.94	99.98	99.46	92.40

error rate

Table 4.2: Mean accuracy of coverage estimation in percent estimated by ERNPE model for C = 16 and various sets of coverage and polymorphism rate parameters in samples.

model - fraction of distinct polymorphic k-mers in reads to all distinct k-mers in reads. ERNPE predicts abundance p_j for class j as:

$$p_j = (1-\gamma)\sum_{o=1}^{\infty}\beta_o p_{n,o,j} + \gamma\sum_{o=1}^{\infty}\beta_o p_{p,o,j} \tag{4.1}$$

where $p_{n,o,j}$ is probability for j class computed by E($o \cdot C_k$,ϵ) and $p_{p,o,j}$ is probability for j class computed by E($o \cdot C_k/2$,ϵ).

Results on the same simulated reads as in previous model are in attached CD, file *simulations_for_ernpe.xlsx*. We will, again, show only results for C = 16 in Table 4.2. Despite the strong polymorphism assumption, this model yields results with high accuracy of coverage estimation on simulated data only with the exception of high polymorphism rate even for relatively low coverage C = 4.

Tests on real sequencing data are in chapter 4.4.

4.3 REP model

Even if our ERNPE model has relatively good results on simulated data, its assumption that if polymorphism occurs in repeated k-mer, all repeats of that k-mer will be polymorphic has no biological basis. Furthermore, this assumption makes estimation of polymorphism rate more difficult. Therefore, we would like to create more general prediction model for diploid organisms.

Keeping in mind our findings from chapter 3.3, specifically difference between $\beta_{R,o}$ and $\beta_{G,o}$, we have chosen a different approach to model repeats in genome than previous models. We have one assumption, that there are no polymorphic repeats in genome.

Parameters that will our last model Repeats, errors and polymorphisms optimize are:

- Coverage C
- Error rate ϵ
- Parameters for repeat rate estimation q_1, q_2, q
- K-mer polymorphism rate p'

Contrary to usage of repeats parameters q_1, q_2, q in RE and ERNPE model, we use this parameters to model k-mers repeats in genome rather than in reads.

Again, G'_d is the number of distinct k-mers in genome and W'_d is the number of distinct k-mers in reads.

Probability p_j of abundance class j is computed as:

$$p_j = \sum_{o=1}^{\infty} \beta_{R,P,o} \cdot p_{o,j} \tag{4.2}$$

where:

- $\beta_{R,P,o}$: a fraction of distinct k-mers with o occurences in diploid genome to all distinct k-mers in reads, that considers occurences of polymorphic k-mers,
- $p_{o,j}$: probability for j abundance class computed by E($o \cdot C_k$,ϵ).

Computation of $\beta_{R,P,o}$ starts with estimation of $\beta_{G,o}$ in haploid genome. We use the same estimation process as RE model uses for $\beta_{R,o}$. As our experiment with no erroneous k-mers in section 3.2 shows, geometric distribution yields relatively high estimates of k-mer occurence in genome, thus it is suitable for our estimation of $\beta_{G,o}$ as genome contains

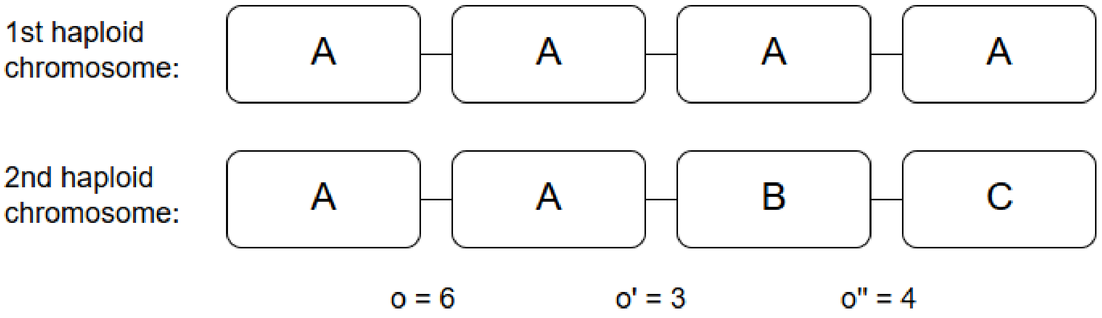


Figure 4.1: Example of k-mer with o copies in diploid genome.

no erroneous k-mers.

$\beta_{G,o}$ is then used to compute $N_{G,P,o}$ - number of repeats of non-polymorphic k-mers in diploid genome. Computation of $N_{G,P,o}$ is divided into several cases given the number of repeats in diploid genome:

$$N_{G,P,o} = \begin{cases} \sum_{o''=1}^{\infty} \sum_{m=1}^{o''} \beta_{G,o''} G'_d \binom{o''}{m} m (1-p')^{o''-m} p'^m & \text{if } o = 1 \\ \sum_{o''=o'}^{o} \beta_{G,o''} G'_d \binom{o''}{2o''-o} (1-p')^{o-o''} p'^{2o''-o} & \text{if } o = 2o' \\ \sum_{o''=o'+1}^{o} \beta_{G,o''} G'_d \binom{o''}{2o''-o} (1-p')^{o-o''} p'^{2o''-o} & \text{if } o = 2o'+1 \end{cases}$$

These equations denote how many k-mers with o copies is present in diploid genome as a number of k-mers with preserved repeats (repeats, that have not changed because of polymorphism).

We will now explain these equations in detail:

- o': case identification variable,
- o: number of preserved repeats - repeats, that are not polymorphic,
- o'': number of repeats in diploid genome.

Idea behind equations for cases $o = 2o'$ and $o = 2' + 1$ is following: For every copy number class o we compute how many k-mers from repeats classes o' to o (or $o' + 1$ to o) will transfer to repeats class o as a result of polymorphism process. We compute the probability of one k-mer with copy class o'' preserving o copies as: $\binom{o''}{2o''-o}(1-p')^{o-o''}p'^{2o''-o}$. Example of k-mer with o preserved copies is in Figure 4.1.

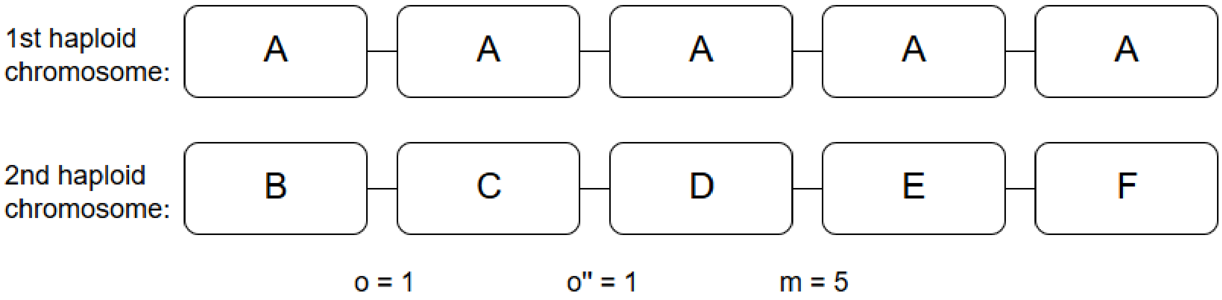


Figure 4.2: Example of m polymorphic k-mers from k-mer with m copies in diploid genome.

Case when $o = 1$ is specific as it describes the number of polymorphic k-mers. One k-mer with o'' repeats in haploid genome may have all repeats in second haploid genome (in the diploid sense) changed to different k-mers. This number of different k-mers is m. Example of m polymorphic k-mers from one source k-mer is in Figure 4.2.

We then create fractions $\beta_{G,P,o}$ by renormalizing $N_{G,P,o}$ by their sum.

We further compute $\beta_{R,P,o}$. Let $X_{P,a,o}$ be the number of all k-mers in reads with copy number o and $Z = E(o \cdot c_k, \epsilon)$. Similarly to the previous models, we can compute the number of distinct k-mers with copy number o by weighting W_d with $\beta_{R,P,o}$. Sum of all abundance probabilities p_j from Z_o multiplied by abundance classes j gives us the number of k-mers with copy number o in reads. Subsequently, we can compute $X_{P,o,a}$ as:

$$X_{P,a,o} = \beta_{R,P,o} W_d \sum_{i=1}^{\infty} Z_{o,i} \cdot i \tag{4.3}$$

We can again compute $X_{P,a,o}$ from diploid genome as number of all occurences of k-mers with o occurences in genome multiplied by k-mer coverage C_k:

$$X_{P,a,o} = \beta_{G,P,o} \cdot G'_d \cdot o \cdot C_k \tag{4.4}$$

By connecting equations above through $X_{P,a,o}$ we get:

$$\beta_{R,P,o} W_d \sum_{i=1}^{\infty} Z_{o,i} \cdot i = \beta_{G,P,o} \cdot G'_d \cdot o \cdot C_k \tag{4.5}$$

Subsequently, we can express $\beta_{R,P,o}$ as:

$$\beta_{R,P,o} = \frac{\beta_{G,P,o} \cdot o \cdot C_k \cdot G}{W_d \cdot \sum_{i=1}^{\infty} Z_{o,i} \cdot i} \tag{4.6}$$

Our implementation of REP model in CovEst does not yield accurate results of parameters estimation. We were unable to identify the source of innacuracy in estimations, whether the problem lies within the optimization process or the prediction model itself. However, we still believe that biological base of REP model relatively closely models the real situation.

4.4 Repeat models on real data

We have tested all prediction models incorporating repeats on real sequencing data from diploid organism Caenorhabditis elegans (worm), strain JU258 (sample taken from Madeira) from Caenorhabditis elegans Natural Diversity Resource project [26]. Additionaly, we have tested GenomeScope tool on this data as well.
Size of reference genome WS245 of Caenorhabditis elegans has 100 286 401 bases. Strain JU258 has 2.428% polymorphism rate. Reads come from an Illumina sequencer 1.1. We have tried two read sets, one with 33.3159 depth of coverage, the other with 16.6158 (half of the first set). Both reads sets has 0.42% error rate and mean read size 98.6 bases. Coverage and error rate were computed from provided aligned reads using Qualimap tool. [27]

Table 4.3 compares results of parameters estimations for set of reads with higher coverage. GenomeScope tool estimation of genome size is closest to original size. Its coverage estimation is highly accurate only for haploid chromosome, to obtain coverage of both, we have to multiply coverage by two (although authors does not state if their coverage is meant as a haploid or diploid). ERNPE model also surprisingly yields relatively correct estimate of coverage.

Model	**Coverage**	**Genome size**	**Error rate**	**Polymorphism rate**
Real	33.3159	100 286 401	0.420%	2.428%
RE	19.7410	179 025 988	0.124%	N/A
ERNPE	37.03	95440588	0.120%	N/A
GenomeScope	16.75	98 897 045	0.122%	0.178%

Table 4.3: Comparison of results of several models on Caenorhabditis elegans sequencing data, coverage 33.3159.

Table 4.4 shows results for parameters estimation for second set of reads. Best estimation of genome size has again GenomeScope and if we consider coverage C only as a haploid, then its estimation of this parameter is also the highest. Error rate was not correctly estimated by any of models.

Model	**Coverage**	**Genome size**	**Error rate**	**Polymorphism rate**
Real	16.6158	100 286 401	0.420%	2.428%
RE	14.7715	120 102 218	0.111%	N/A
ERNPE	18.7159	94 790 300	0.0010%	N/A
GenomeScope	8.4125	99 730 353	0.102%	0.218%

Table 4.4: Comparison of results of several models on Caenorhabditis elegans sequencing data, coverage 16.6158.

Conclusion

Aim of our work was to create prediction models, that could estimate properties from diploid sequencing data. Models created prior to our work either does not consider an existence of reads from diploid genome (specifically existence of polymorphic positions) or have only limited usage due to conditions placed upon input data.

During our research on ERE_d model, we have found out the true meaning of β_o weight in RE model as a fraction of distinct k-mers with copy number o in reads. We have supported our claim with results of exhaustive experiments. We consider this discovery as an extraordinary result, as it both clarifies results of RE model and encourages a future expansion of genome attributes estimations. Furthermore, we have suggested a system of equations to compute real number of k-mers with o copy number. Our experiments with both simulated and real genomes showed a promising path for future research.

We have expanded RE model to ERE_d model - RE model with d repeats parameters. Motivation behind ERE_d creation was to experiment with higher repeats parameters to find out whether the accuracy of estimations would increase or not, as number of repeats parameters in RE model was set after only a brief genome analysis. Results of our experiments shows that there is no significant increase in estimations accuracy.

Next, we have explained how diploid genome affects haploid models (specifically E model). Then, we have expanded the E model to consider an existence of polymorphic k-mers in reads by adding a new parameter γ, thus creating the EP model suited for diploid genomes without repeats. We have also incorporated an estimation of polymorphism rate into EP model and added EP model into CovEst. We have tested EP model on simulated data together with E model and GenomeScope, where EP model had the highest accuracy of all parameters.

Following the EP model creation, we were researching the possibility of usage of RE model on polymorphic genomes with repeats. As we have shown, RE model can be only used on diploid data with low polymorphism rate. We have further created an ERNPE model with assumption that repeating k-mer has polymorphism in all its haploid k-mers. This model yields surprisingly good results on simulated data from real Caenorhabditis elegans genome. As the last model, we have suggested an REP model and although we were unable to obtain meaningful results, this model has relatively strong biological basis.

Genome properties estimation yields several challenges for future research. Continuing our work on genome repeats estimations from $\beta_{R,o}$ for RE model may bring fruitful results. Other approach to address this problem is to avoid computation from $\beta_{R,o}$ at all and instead model repeats in genome directly in RE model.
Another challenge is the investigation of REP model, its optimization process and results it yields. Following research is the creation of prediction model without any assumptions on input data for diploid genomes.
Prediction models can be also expanded to include other biological phenomenons such as GC bias, when sequencer prefer reads from areas in genome rich in G and C bases, thus creating spaces with low-to-no coverage, or eventually expand prediction models to estimate attributes from several genomes (e.g. bacterias) in one set of reads.

Bibliography

[1] M. Quail et al., “A tale of three next generation sequencing platforms: comparison of ion torrent, pacific biosciences and illumina miseq sequencers,” *BMC Genomics*, vol. 13(1), p. 341, 2012.

[2] L. Liu et al., “Comparison of next-generation sequencing systems,” *Journal of Biomedicine and Biotechnology*, vol. 2012, p. 1–11, 2012.

[3] D. Kleftogiannis et al., “Comparing memory-efficient genome assemblers on stand-alone and cloud infrastructures,” *PLoS One*, vol. 8(9), 2013.

[4] M. Šašinka et al., *Vademecum Medici.* Vydavateľstvo Osveta s.r.o., 2003.

[5] J. Nosek et al., *Genomika.* CreateSpace Independent Publishing Platform, 2013.

[6] B. Brejová and T. Vinař, *Metódy v bioinformatike.* Knižničné a edičné centrum, Fakulta matematiky, fyziky a informatiky, Univerzita Komenského v Bratislave, 2015.

[7] P. A. Lefebvre and C. D. Silflow, “Chlamydomonas: The cell and its genomes,” *Genetics*, vol. 151(1), pp. 9–14, 1999.

[8] C.-C. Wu et al., “20-hete and blood pressure regulation,” *Cardiology in Review*, vol. 22(1), pp. 1–12, 2014.

[9] J. M. Heather and B. Chain, “The sequence of sequencers: The history of sequencing dna,” *Genomics*, vol. 107, pp. 1–8, 2016.

[10] O. T. Bishop et al., *Bioinformatics and Data Analysis in Microbiology.* Caister Academic Press, 2014.

[11] J. R. Miller et al., “Assembly algorithms for next-generation sequencing data,” *Genomics*, vol. 95(6), pp. 315–327, 2010.

[12] M. D. Adams et al., “The genome sequence of drosophila melanogaster,” *Science*, vol. 287, pp. 2185–2195, 2000.

[13] R. M. Leggett et al., “The genome sequence of drosophila melanogaster,” *PLoS ONE*, vol. 8(3), e60058, 2013.

[14] L. Xiaoman and M. S. Waterman, “Estimating the repeat structure and length of dna sequences using l-tuples,” *Genome Research*, vol. 13, pp. 1916–1922, 2003.

[15] D. Williams et al., “Rapid quantification of sequence repeats to resolve the size, structure and contents of bacterial genomes,” *BMC Genomics*, vol. 14.537, 2013.

[16] M. Hozza, T. Vinař, and B. Brejová, “How big is that genome? estimating genome size and coverage from k-mer abundance spectra,” *Springer International Publishing Switzerland*, vol. 10.1007, pp. 199–209, 2015.

[17] G. W. Vurture et al., “Genomescope: Fast reference-free genome profiling from short reads,” *Bioinformatics*, vol. 33(14), pp. 2202–2204, 2017.

[18] G. Marcais and C. Kingsford, “A fast, lock-free approach for efficient parallel counting of occurrences of k-mers,” *Bioinformatics*, vol. 27(6), pp. 764–770, 2011.

[19] P. Melsted and J. K. Pritchard, “Efficient counting of k-mers in dna sequences using a bloom filter,” *BMC Bioinformatics*, vol. 12:333, 2011.

[20] R. Chikhi and P. Medvedev, “Informed and automated k-mer size selection for genome assembly,” *Bioinformatics*, vol. 30(1), pp. 31–37, 2014.

[21] D. R. Kelley et al., “Quake: quality-aware detection and correction of sequencing errors,” *Genome Biology*, vol. 11:R116, 2010.

[22] C. Zhu et al., “Algorithm 778: L-bfgs-b: Fortran subroutines for large-scale bound-constrained optimization,” *ACM Transactions on Mathematical Software*, vol. 23(4), pp. 550–560, 1997.

[23] E. Jones, T. Oliphant, P. Peterson *et al.*, “SciPy: Open source scientific tools for Python,” 2001–, [Online; accessed <today>]. [Online]. Available: http://www.scipy.org/

[24] R. Kajitani et al., “Efficient de novo assembly of highly heterozygous genomes from whole-genome shotgun short reads,” *Genome Res*, vol. 24(8), pp. 1384–1395, 2014.

[25] “Matriisi-algebra 03019p - 5.5. overdetermined system, least squares method ; university of oulu,” lecture notes. [Online]. Available: https://s-mat-pcs.oulu.fi/~mpa/matreng/ematr5_5.htm

[26] D. E. Cook et al., “Cendr, the caenorhabditis elegans natural diversity resource,” *Nucleic Acids Research*, vol. 45(D1), 2016.

[27] F. Garcıa-Alcalde et al., “Qualimap: evaluating next-generation sequencing alignment data,” *Bioinformatics*, vol. 28.20, pp. 2678 – 2679, 2012.

[28] “Math32031: Coding theory - part 2: Hamming distance, school of mathematics, the university of manchester,” lecture notes. [Online]. Available: http://www.maths.manchester.ac.uk/~pas/code/notes/part2.pdf

Appendix A

Content of attached CD is:

- *ered_estimations.xlsx*
- *simulations_for_ep.xlsx*
- *simulations_for_ernpe.xlsx*