NOVEL ALGORITHMS TO ESTIMATE GENOME COVERAGE USING HIGH THROUGHPUT SEQUENCING DATA

A Dissertation Presented to

the Faculty of the Department of Computer Science

University of Houston

In Partial Fulfillment

of the Requirements for the Degree

Doctor of Philosophy

By

Meenakshi Sharma

May 2014

NOVEL ALGORITHMS TO

ESTIMATE GENOME COVERAGE USING

HIGH THROUGHPUT SEQUENCING DATA

Meenakshi Sharma

APPROVED:

Dr. Ioannis Pavlidis, Chairman

Dr. Yuriy Fofanov, Committee Member

Dr. Barbara Chapman, Committee Member

Dr. Nikolaos Tsekos, Committee Member

Dr. William Widger, Committee Member

Dean
College of Natural Sciences and Mathematics

Acknowledgements

I am greatly indebted to many people without whom this dissertation would not have been possible.

I would like to express my deepest gratitude to Dr. Yuriy Fofanov for his guidance and critical feedback throughout the course of this research work. I am also thankful to him for sharing his insights on many research topics during several discussions. I am also extremely grateful to Dr. William Widger for his valuable advice and support which has been helpful in preparing this dissertation. I would like to extend my sincerest appreciation to Dr. Ioannis Pavlidis, Dr. Nikolaos Tsekos, and Dr. Barbara Chapman for having served on my committee. Thank you for being so accommodating and providing constant support and encouragement throughout the dissertation.

I am also very thankful to all my fellow research lab members Levent Albayrak, Dr. Georgiy Golovko, Dr. Mark Rojas, Kamil Khanipov, and Otto Dobretsberger for their help and suggestions. I would like to acknowledge the financial support provided by the Department of Computer Science, University of Houston-Main campus which assisted me in finishing my dissertation work and to attend technical conferences.

Last but not the least, I would like to dedicate this work to my parents and brother who have been my strongest proponents and source of inspiration. And thanks to all my dear friends, whose kind words of encouragement kept me going during the tough times.

NOVEL ALGORITHMS TO ESTIMATE GENOME COVERAGE USING HIGH THROUGHPUT SEQUENCING DATA

An Abstract of a Dissertation Presented to

the Faculty of the Department of Computer Science

University of Houston

In Partial Fulfillment

of the Requirements for the Degree

Doctor of Philosophy

By

Meenakshi Sharma

May 2014

Abstract

Genetic variation can occur in the form of single base changes called Single Nucleotide Polymorphisms (SNPs) or large-scale structural alterations called Copy Number Variations (CNVs). Identification and analysis of CNV(s) is critical in understanding its association with evolution, health, and disease. Over the past decade, new advancements in DNA sequencing technologies have fuelled the field of genomics and opened new doors for performing Copy Number Analysis (CNA).

To perform CNA, millions of short subsequences or *reads* produced by High Throughput Sequencing (HTS) platforms are aligned to reference genome sequence(s). The sequence alignment process produces total number of reads that aligned to each location in the genome and is collectively called as reads coverage.

The focus of this research is to develop novel algorithms to accurately estimate coverage in the presence of DNA repeats and single nucleotide mutations. The copy number distribution of the *reads* mapped to the reference sequence would ideally follow a Poisson distribution assuming that the nucleotide sequence of a genome is random and the sequencing reads came from the random locations in the genome. The coverage data, however, exhibits over-dispersion in the extreme ends of the distribution. Repeatable sequences and SNPs contribute to these unexpected high coverage frequencies.

This dissertation presents novel algorithms to estimate the average coverage using a model based on Poison distribution. The model was tested on both simulated and real data with different coverage depths and predicts the actual model parameters with reasonably good accuracy. The proposed approach improves estimation of average genome coverage which is central to gene-expression, DNA methylation, and metagenomic studies.

Contents

Chapter 1

INTRODUCTION

Genome is the complete genetic material of an organism carries necessary information for the growth and development. In any living cell, genome is present in the form of several densely packed chromosomes. The chromosomes, in turn, contain several coding regions called genes that code for specific proteins.

DNA (deoxyribonucleic acid) is an essential biomolecule which stores genetic information which is inherited from parent to daughter cells. DNA occurs mostly as a stable double-helical structure and is composed of 4 basic nucleotides: A (adenine), T (thymine), G (guanine), and C (cytosine).

The astounding amount of inter- and intra-species genetic diversity that we observe around us can be attributed to mutagenic events occurring in genomic DNA over a long period of time.

1.1 Copy Number Variation

Genetic mutations like SNPs and Copy Number Variations (CNVs) are introduced in the populations either through heredity or due to somatic mutations. Single Nucleotide Polymorphisms (SNPs) are single nucleotide change and comprise less than 1% of the human genome.

Large structural changes in chromosome, ranging from 1000 bp-5 Mb, include genomic events such as amplifications, insertions, and deletions. CNVs, together with epigenetic changes like DNA methylation, bring about expansion and diversification of genomes [49].

CNVs are important features of a genome, and modifications in CNVs have been associated with medical conditions like developmental disorders and cancer [40, 41]. Single point mutations and CNVs sometimes predispose an organism to a certain disease and often, when coupled with other environmental changes, alter the expression (phenotype) of a complex disease.

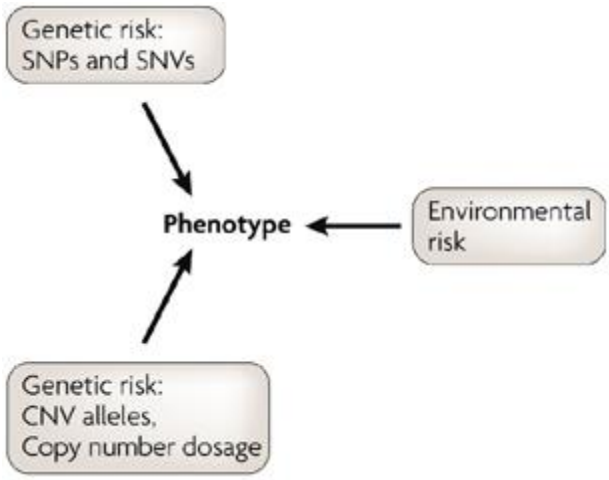


Figure 1.1: The genetic and environmental risks combined confer the total risk for a complex phenotype [4]

CNVs can be viewed at two levels: a) small-scale sequence level changes and b) large-scale chromosomal level variations. The first category includes small duplications, indels, while the second group of variations range from 1000 bp-5 Mb and include bigger amplifications, insertions, and deletions.

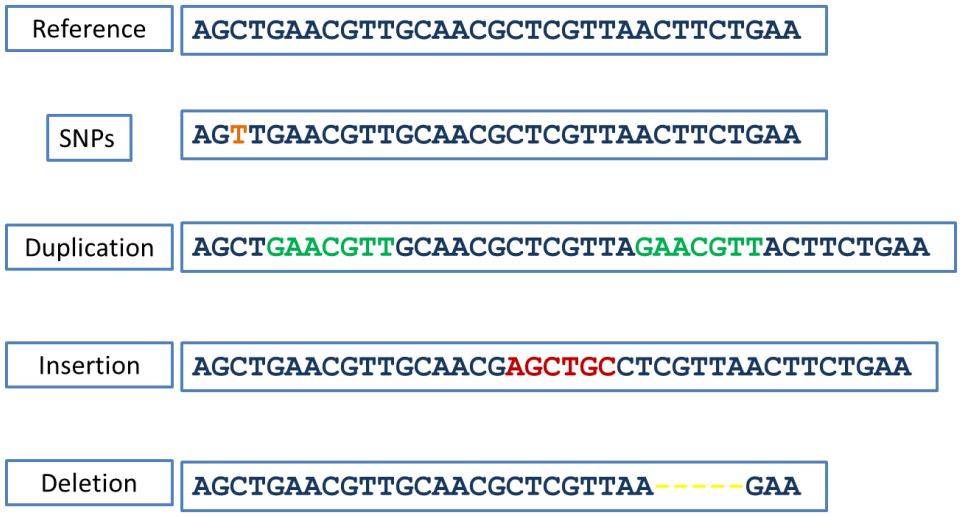


Figure 1.2: Sequence level copy number variations

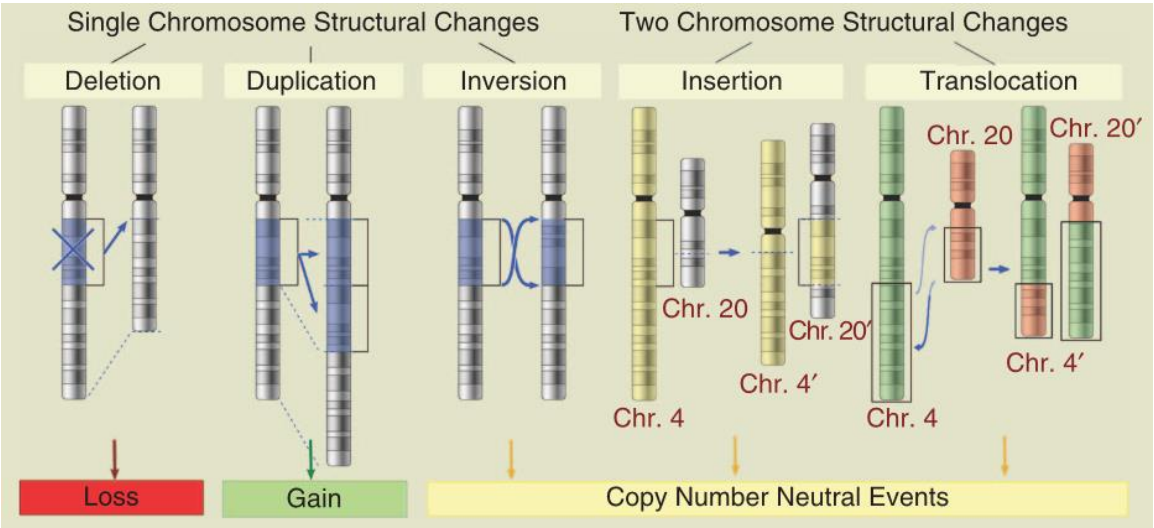


Figure 1.3: Chromosomal level copy number variations [1]

CNV in Evolution, Health, and Disease

Over the past several years, there has been a growing interest in the study of CNVs and the role they play in evolution, health, and disease. CNVs have been conserved for many years in the genome and have been passed through generations. CNVs can be either inherited or introduced by *de novo* mutation. These mutations can be the result of errors occurring during DNA synthesis, repair, cross-over, or environmental factors like exposure to radiation.

Depending upon the location of the mutation in the genome, varied results in gene activity can be observed. While CNVs occurring at non-coding regions generally do not amount to changes in gene expression, CNVs occurring at coding or regulatory regions can have wide-ranging effects on the gene expression. [48, 49] discuss how CNV could be helpful in creating new functional genes or affecting expression of genes.

One particular example where a copy number gain has been beneficial to human is that of salivary amylase gene (AMY1). Compared to two copies found in chimpanzees, humans on an average have more than six copies of AMY1 genes. The greater number of AMY1 gene copies suggest adaption in humans to high starch diet [50].

CNVs are also associated with several diseases. Duplication of part or all of chromosome 21 leads to *down syndrome*. Many developmental disorders such as *mental retardation, autism* and *schizophrenia* are few other examples of changes in CNVs [40]. Fewer segmental duplications in a particular gene cause susceptibility to HIV-1/AIDS [21].

Therefore, in this example CNVs are functioning to make an individual less prone to a disease.

CNV is widely observed among human genomic DNA, and around ~0.4% of CNV-based genomic differences are observed in two unrelated individuals [30]. Even identical twins differ from each other with respect to CNVs in their genomes. This is the basis of genomic fingerprinting. Figure below illustrates differences (>8Kb) between an individual and reference human genome [65].

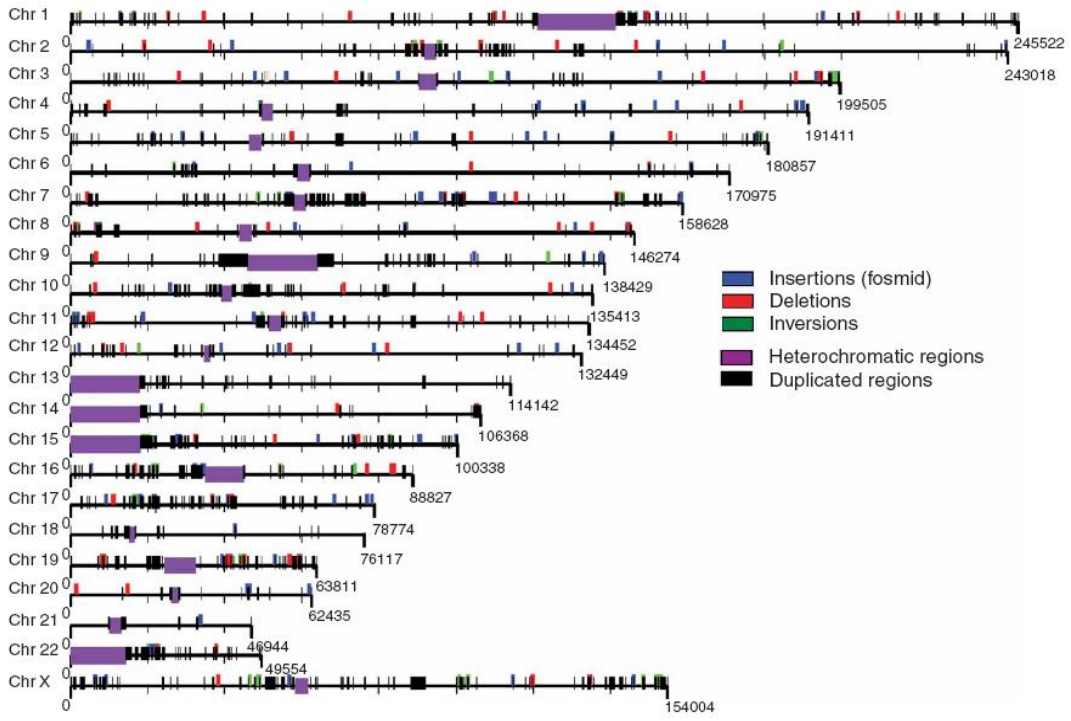


Figure 1.4: Structural variation map. The schematic summarizes the distribution of insertions, deletions and inversions (>8 kb) on each human chromosome [65].

1.2 DNA Methylation

Unlike CNVs which modify the genome sequence, *epigenetic* changes [22-23] do not involve alteration to the genome. DNA methylation is one of the well-studied epigenetic modifications and influences many biological processes. DNA methylation is the addition of methyl (-CH3) group primarily to the C (cytosine) base in the CpG di-nucleotide (p represents phosphodiester bond between C and G nucleotides adjacent to each other in the same strand) of a DNA sequence (Figure 1.5). The original cytosine base after reaction becomes 5-methyl cytosine, denoted as 5-mC (Figure 1.6).

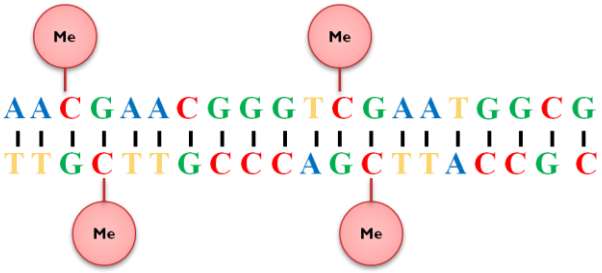


Figure 1.5 a: DNA Methylation (Me= Methyl group $–CH_3$)

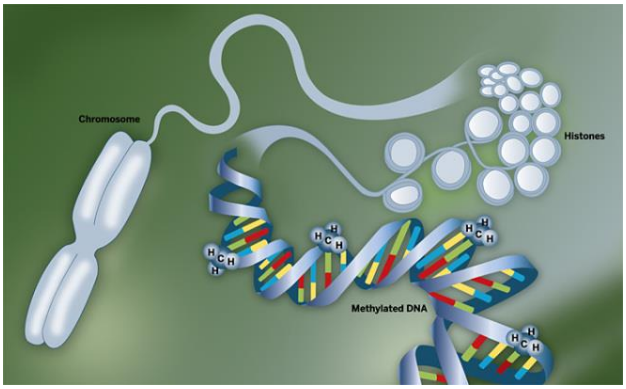


Figure 1.5 b: DNA Methylation (Methyl group $–CH_3$)

The conversion of a C to 5-mC is catalyzed by a group of enzymes named DNA methyltransferases (DNMT). These enzymes are capable of performing both *de novo* and maintenance DNA methylation. *De novo* methylation occurs when previously un-methylated DNA in stem or tumor cells is methylated. The maintenance of DNA methylation involves inheritance and conservation of the existing methylation patterns from parent to daughter cells.

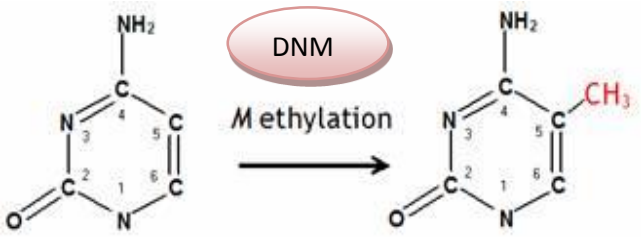


Figure 1.6: Addition of Methyl group $–CH_3$ to Cytosine (C) by DNA methyltransferase

---

DNA methylation is an epigenetic mechanism i.e. does not involve modifications to the genetic code. Nevertheless, the presence or absence of methyl groups near CpG-rich gene promoter regions change the way genetic information is decoded. In mammals, including humans, 70-80% of all CpG dinucleotides are methylated, present mostly in the repetitive sequences with lower CpG density [62]. Unlike DNA repeat sequences, the promoter regions of most expressed genes (56% of mammalian genes) are unmethylated, indicating that methylation might be inversely correlated to transcriptional activity. *De novo* methylation and de-methylation of certain genomic sequences can hence disrupt the normal gene transcription and functioning within the cell.

The alterations to the DNA methylation profile can be induced by several internal and external factors (Figure 1.7) which could be related to a person's lifestyle such as smoking [37], diet, or part of a systematic biological change like aging or due to infection caused by a bacteria/ virus.

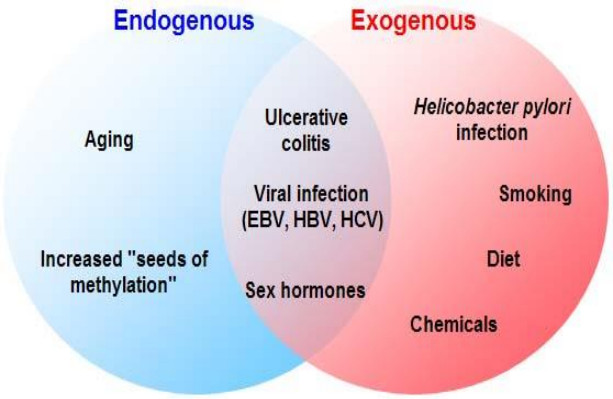


Figure 1.7: Factors that can cause aberrant DNA methylation [69]

Methylated cytosine (5-mC) was first detected in 1948 [24], and since then extensive research has been done to determine the significance of DNA methylation in molecular/ biological processes. Today, involvement of DNA methylation in regulating gene expression, embryonic development, genomic imprinting, X-Chromosome inactivation, aging, and maintaining chromosome stability is well-established [25] [26] [31] [55] [66].

DNA Methylation in Health and Disease

Gene Silencing

DNA methylation acts as a negative control over the gene regulation. The presence of methylated cytosines near the promoter region of a gene has been known to down-regulate the gene expression. On the other hand, *de novo* de-methylation of a gene activates gene expression. Currently, the complete mechanism of the process is not known. DNA methylation is not the sole mechanism leading to gene silencing. In fact, DNA methylation works in conjunction with other epigenetic and non-epigenetic processes like histone modifications to restrict the accessibility of a gene for transcription [51].

Development and Aging

The methylation profiles of different cells change dynamically during the lifetime of an organism, suggesting that epigenetic changes like DNA methylation play important role in development and disease [2-5] [59]. Figure 1.4 illustrates how environmental factors and aging influence methylation levels in different stages of cells from embryo development to adulthood.

Epigenomes of genetically identical twins are also distinct and rapidly change from birth to later period in their life [28] [41]. Methylation levels across the genome undergo a drift while aging, making the organism more susceptible to diseases [26].

Genomic Imprinting

Diploid organisms like mammals contain two set of chromosomes in most of their cells except for germline cells. During fertilization one set of chromosomes is inherited from each of the parent cells and expression of a gene in daughter cell depends on both the copies of parental genes. However, a very small percentage (<1%) of the genes are imprinted, which means that the expression of a few genes is completely decided by the genetic make-up of one parent only.

"*Imprinting is defined as the epigenetic marking of the parental genomes of a diploid organism with respect to their parental origin* [48]". The maintenance of imprinted genes and demethylation of other genes is an important feature of embryo development. During embryonic development through a process called *erasure* [8] most of the genes are demethylated, with an exception to the imprinted genes [52, 55].

Many imprinted genes found so far have also been associated with genetic diseases demonstrating parent-origin effects [47]. Beckwith–Wiedemann syndrome (BWS) on *chromosome 11p* and Prader–Willi/Angelman syndromes on *chromosome 15q* are both examples of such diseases. DNA methylation changes also act as a biomarker in cancer patients [9].

With advancements in the high throughput DNA sequencing technologies [18] and techniques to isolate non-methylated DNA from the methylated DNA, global measurement of DNA methylation levels is now easily possible.

1.3 DNA Sequencing

The history of DNA sequencing dates back to 1970s when the first RNA/ DNA sequencing methods: chemical methods [43] or chain termination methods [56, 57] were being developed. The chain-termination method developed by Frederick Sanger was widely accepted due to its high reliability and ease of execution. The methodologies came to be known as first generation (or Sanger) sequencing (spanning from 1980s to early 2000s) and were also employed in *Human Genome Project*.

The earliest second-generation sequencing platform was introduced in 2005 by *454 Life Sciences* which was based on "sequencing-by-synthesis" strategy. At present *Roche/454 FLX*, *Illumina/Solexa Genome Analyzer* and *Applied Biosystems (ABI) SOLiD Analyzer* are among the most popular next-generation, commercially available High Throughput Sequencing (HTS) platforms. In addition to providing lower "sequencing cost per genome", HTS technologies surpassed Sanger sequencing in terms of speed and throughput. Using NGS platforms researchers can generate high coverage sequence data (>1000x) and perform analysis of genome at single base resolution.

After the arrival of HTS technologies, sequencing costs have been falling rapidly [63] (from $100M to less than $10K per human genome), while the amount of data has been growing at an unprecedented rate (Figure 1.8). This has made storage, distribution, and analysis of genomic data a formidable task.

Advancements in computing speed and storage capacity have failed to keep pace with the exponential growth of data (Figure 1.9) and have only confounded the problem.

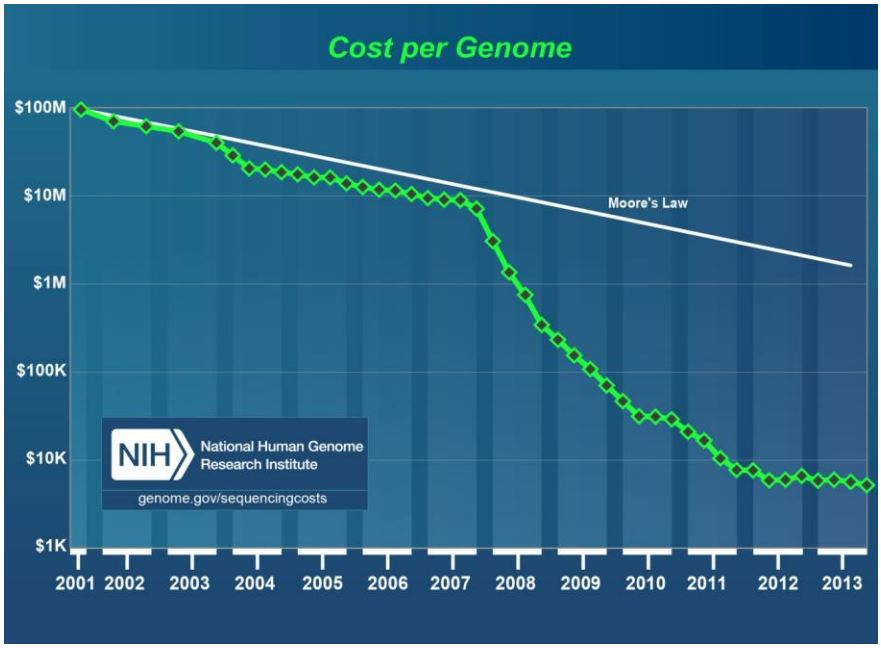


Figure 1.8: a) Cost per genome [71]

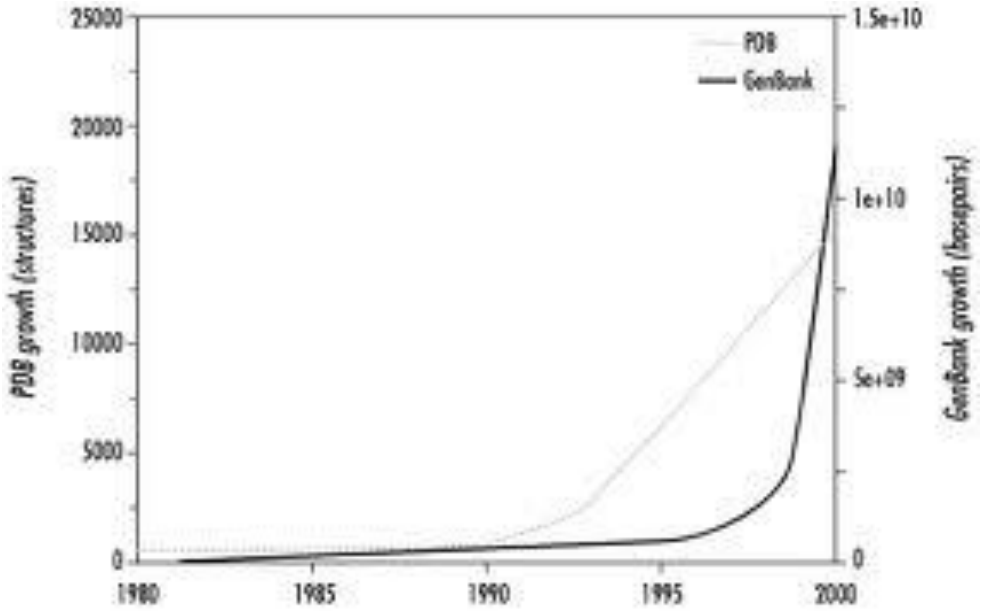


Figure 1.8: b) Growth of GenBank & PDB database [17]

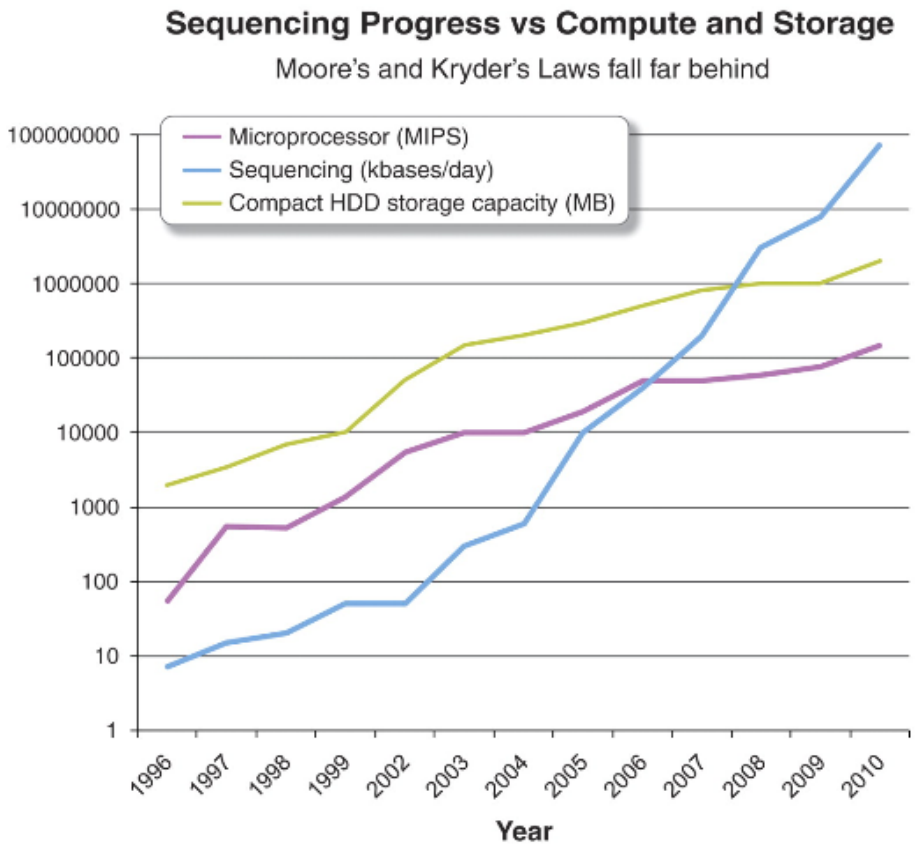


Figure 1.9: Sequencing rate versus computing speed and storage capacity [29]

The gap between the sequencing data and computational power brings new challenges and opportunities to the field of bioinformatics. Detailed discussions about the technologies and challenges associated with the resulting HTS data are presented in [19]. Sequencing methods have evolved in the past few decades and still continue to improve [33].

Table 1.1: Genome Sequencing Technologies [33]

	Sequencing Platform	Method	Advantages	Disadvantages	System (Cost per run)
First Generation					
	Sanger sequencing	Strands of fragmented DNA are resolved on gel and distributed in order of length, with end base labeled	High accuracy Validate findings of NGS	High cost Low throughput (time consuming)	US$2,000,000 (US$250,000)
Second Generation		Cyclic array-based sequencing; strands of fragmented DNA are amplified; bases are added sequentially using DNA polymerase; excess reagent is washed out; imaging identifies base incorporated; and process repeats	Higher throughput More economical	Short read length Complex sample preparation Need for amplification Long time to results Significant data shortage and interpretation requirements	
	Roche Applied Science 454® genome sequencer (Roche, Basel, Switzerland)	Pyrophosphate released at time of base incorporation	1–5 µg DNA needed	400 bp read lengths	US$500,000 (US$8439)
	HiSeq® (Illumina, CA, USA)	Fluorescent-labeled nucleotides added simultaneously	<1 µg DNA needed	75 (35–100) bp read lengths More false positives	~US$400,000 (US$8950)
	Miseq (Illumina)		Clinical applications	Unable for WES, WGS, ChIPSeq and RNA-seq 10 h per run	
	Applied Biosystems SOLiD® 4 (Life Technologies, CA, USA)	Driven by DNA ligase instead of DNA polymerase	2–20 µg DNA needed	35–50 bp read lengths	US$525,000 (US$17,447)

	Ion Torrent PGM (Life Technologies)	Nonoptical DNA sequencing; massively parallel semiconductor senses ions produced as nucleotides are incorporated by DNA polymerase-based synthesis	Less than 200 bases needed High accuracy Short run time (fast) Cheaper	Unable for WES, WGS, ChIP-Seq and RNA-seq	US$50,000 (<US$500)
	Helicos				
Third Generation		Novel technologies	No PCR amplification Less starting material Less error prone		
	PacBio RS (Pacific BioSciences, CA, USA)	Single-molecule real-time sequencing; imaging of dye-labeled nucleotides as they are incorporated during DNA synthesis by single DNA polymerase molecule	800–1000 bp read lengths		US$695,000 (~US$1000)
	Heliscope sequencer (Helicos BioSciences)	Single-molecule real-time sequencing; imaging of dye-labeled nucleotides as they are incorporated during DNA synthesis by single DNA polymerase molecule	<2 µg DNA Direct RNAsequencing application	35 bp read lengths	US$750,000 (~US$5,000)
Fourth Generation					
	Oxford Nanopore	Single molecule sequencing incorporating nanopore technology	Whole-genome scan 15 min Very low cost		Not commercially available

ChIPseq: Chromatin immunoprecipitation-sequencing; NGS: Next-generation sequencing; RNA-seq: RNA-sequencing; WES: Whole-exome sequencing; WGS: Whole-genome sequencing.

While the *Human Genome Project* (HGP) was nearing completion (1990-2003), efforts to sequence the human epigenome were already underway: *NIH Roadmap Epigenomics Program, Human Epigenome Project* (HEP) [6]. These huge collaborative initiatives gained from the advancements in current technologies; however researchers faced new challenges which were not encountered before.

HTS technologies produce millions of sub-sequences or ***reads*** per sample in a single day and required new software and tools to manage them. The length of the reads was also much shorter (30-700) compared to the reads produced by Sanger machines (600-1000). These millions of relatively short reads presented technical challenges during standard analyses such as reads alignment or genome reconstruction using *assembly*.

While human genome is thought to be identical for every tissue or cell-type in a person, the epigenome varies from cell to cell within an individual [16]. Methylation patterns for each tissue and cell type also differ over the lifetime of the organism. As a result, the epigenome of an individual could be thousand times larger (Figure 1.10) than the genome (3.3 Gb).

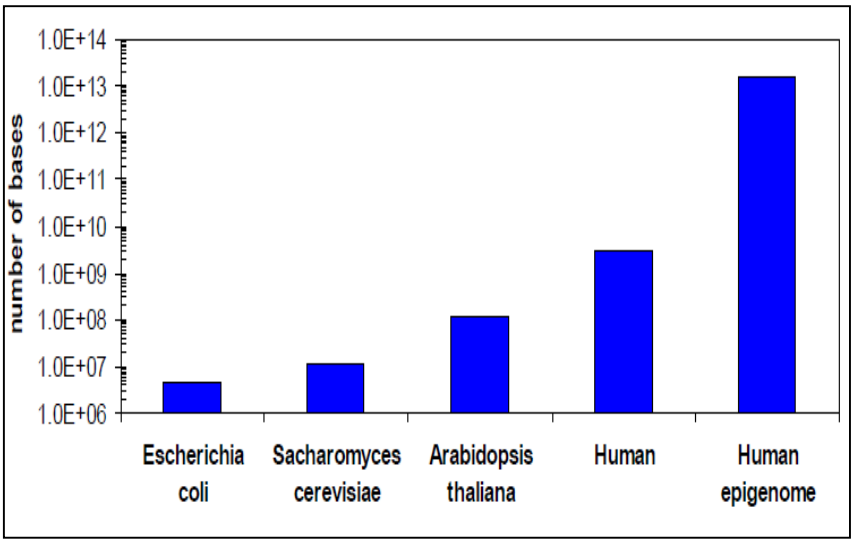


Figure 1.10: Comparison of the sizes of different genomes and the human epigenome. A minimum epigenome size was estimated by estimating 100 different cell types and 50 relevant developmental or disease specific states [72]

Study of epigenome offers new explanation(s) to disease mechanisms [13, 14] [27] [32] [42]. To summarize, sequencing and analysis of the epigenome implies dealing with data and issues of greater magnitude and complexity.

Chapter 2

USING HTS TO DETECT CNV ACROSS GENOME

2.1 High Throughput Sequencing (HTS) Technology

HTS platforms have been widely used for sequencing to solve scientific problems related to various research areas including d*e novo* sequencing, whole genome/ exome sequencing, transcriptome analysis, gene expression analysis, and copy number analysis, to name a few. *Illumina HiSeq* was selected as the most popular sequencing platform in one of the recent surveys conducted by *CLC Bio, a bioinformatics company. Illumina* instruments have been a consistent choice in the year 2013 followed by sequencers from other competitors *AB SOLiD, Roche 454* and *Ion Torrent* (Table 2.1).

Table 2.1: NGS Technologies in use: 2013 figures and % change compared to 2012 [39]

1. Illumina - HiSeq	**44.1%**	▲ 21.5%
2. Illumina - MiSeq	36.0%	▲ 40.8%
3. LifeTech - Ion Torrent PGM	24.3%	▲ 52.7%
4. None	23.0%	▼ 33.9%
5. Roche 454 - GS FLX	21.2%	▶
6. Illumina - GA 2	12.2%	▼ 19.2%
7. Complete Genomics	10.8%	▼ 1.8%
8. PacBio RS	10.4%	▲ 51%
9. LifeTech - SOLiD	9.5%	▲ 3.1%
10. LifeTech - Ion Proton	9.0%	▲ 72.2%
11. Roche 454 - GS Junior	7.2%	▲ 2.8%
12. Other	3.2%	▼ 17.9%

Depending on the specific sequencing technology used, the HTS platforms differ in terms of overall run time, sequencing throughput, and type of reads produced. *Illumina* and *AB SOLiD instruments* generate reads of same length and *Roche* and *Ion Torrent* generate reads of variable lengths. We used *Illumina Genome Analyzer (GA) IIx* to sequence the samples. The GA IIx system is a widely used HTS platform and consists of 4 basic steps: sample preparation, cluster generation, sequencing, and data analysis (Figure 2.1).

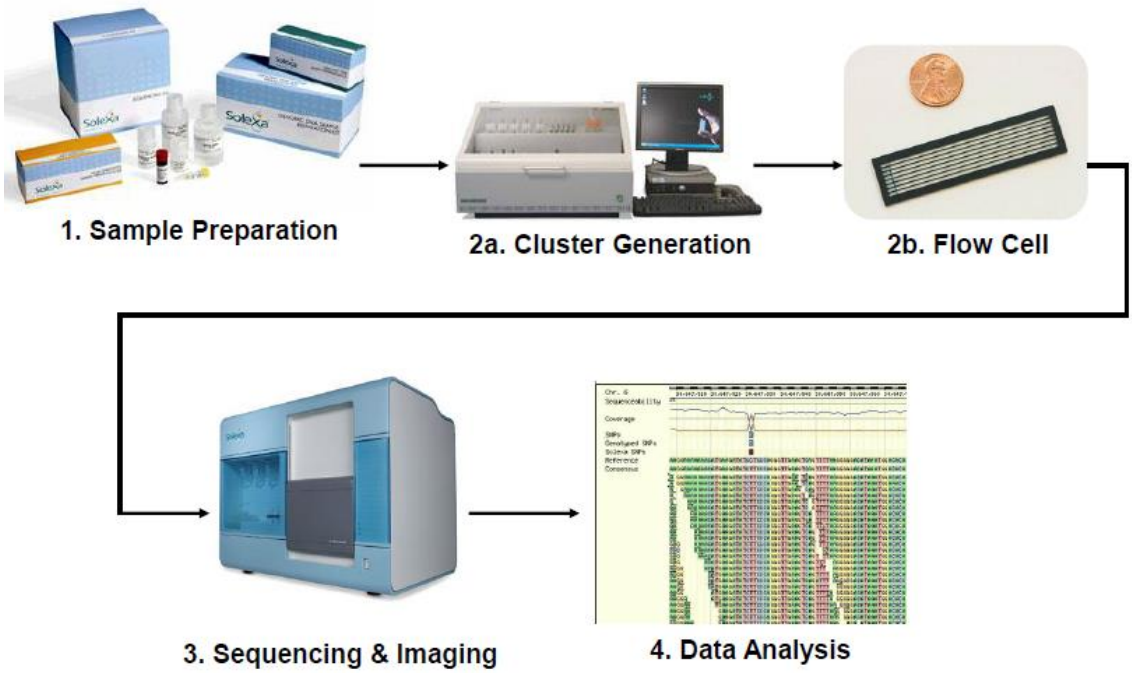


Figure 2.1: Genome Analyzer II workflow

Like its predecessors *Sanger* and *454/ Roche, Illumina GA* uses "sequencing-by-synthesis" technique to determine the sequence of nucleotides. A sequencing library is first prepared using reagents in the sample preparation kit provided by the manufacturer. The process includes fragmentation of the DNA, attaching flanking 'adaptors' supplied by *Illumina*, and performing DNA amplification (Figure 2.2). A sequencing library can be single-ended, paired-end, or multiplexed, depending on whether you want to perform uni-directional, bidirectional, or multiple libraries/ samples per 'lane' sequencing.

During the next step of cluster generation, prepared libraries are moved to a cluster station/ cBot where they are converted into clonal clusters immobilized on the surface of flowcells. A **flowcell** is a proprietary planar optically transparent surface containing fluidic channels called **lanes** specially designed to immobilize, amplify, and sequence millions of sequences in parallel. Each flowcell has 8 lanes and each lane can be set to run one or more (multiplexing) samples per experiment. The lanes contain *oligonucleotide sequences* (*oligos*) or *primers* complementary to the adaptor regions and help to tether the fragments to the flowcell. The fragments are amplified into several millions of copies and the original strand is washed away leaving only single strands in the flowcell ready to be sequenced.

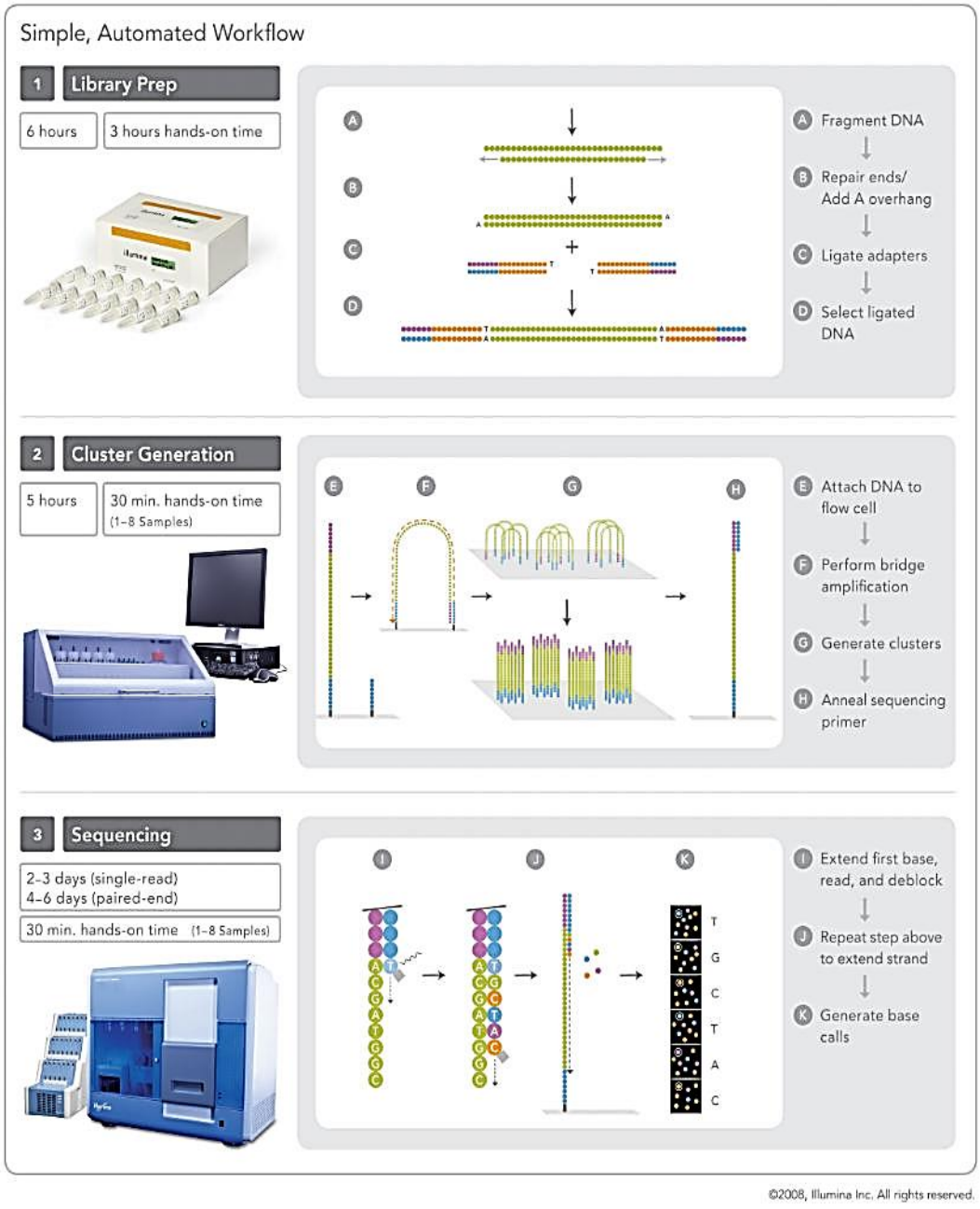


Figure 2.2: Genome Analyzer II: sequencing techniques (*Illumina Inc*)

Once cluster generation is complete, sequencing process begins at the first adaptor using “sequencing-by-synthesis”. Four fluorescently labeled nucleotides are employed to extend the other strand of the fragment one nucleotide at a time. After each addition, clusters are excited with a light source (laser beam) and a characteristic florescent signal is recorded. The emission wavelength and intensity of the signal determines the exact nucleotide that has been added to the strand. Figure 2.3 shows a flowcell image decoded into intensities and sequences. Illumina provides an array of software to perform real time analysis (RTA) of the raw image data and converts them to more meaningful and readable sub-sequences called **reads**.

Figure 2.3: Genome Analyzer II: Data generation

The amount of data and type of reads produced per sequencing run depend on the technology used. A single sequencing run produces gigabytes of data available for downstream analysis using software pipeline(s) and tools designed to meet specific needs of the study.

2.2 Sequencing Data Analysis

Any sequencing data analysis consists of one or all of the following 3 techniques: **Search, Map, and Assemble**. In this dissertation, some of the challenges associated with sequence alignment or mapping are addressed and discussed.

The procurement and analysis of sequencing data can be illustrated as a workflow (Figure 2.4) comprising of the following steps:

a. Sample preparation and DNA sequencing;

b. Quality assessment;

c. Mapping or sequence alignment;

d. Data analysis.

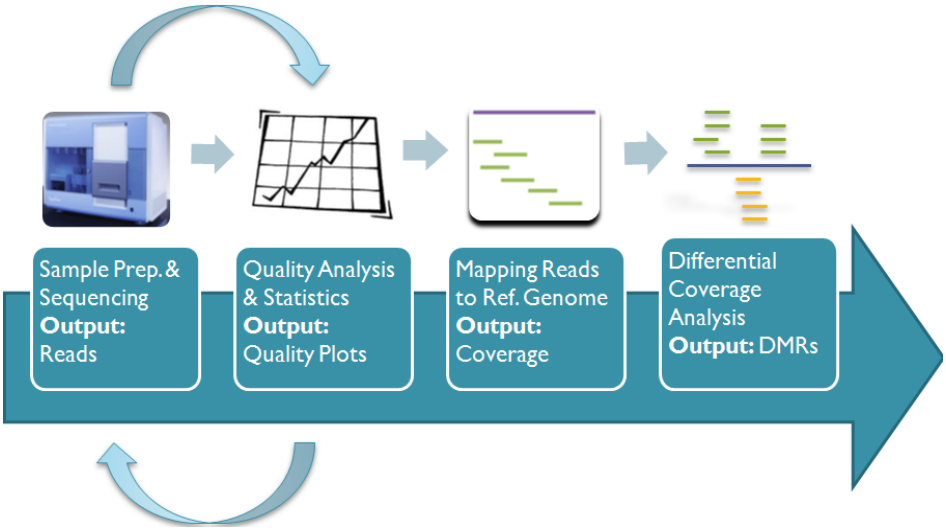


Figure 2.4: Sequencing and data analysis workflow

Sample Preparation and Sequencing

The library preparation is done using kits supplied by manufacturers followed by sequencing performed using HTS instruments like *Illumina Genome Sequence Analyzer IIx*. The *Illumina* sequencer produces short sequences of equal lengths stored in a FASTQ (Figure 2.5) file along with a quality score corresponding to every nucleotide in the read.

Quality Assessment

Quality control is the initial and critical step that must be performed in NGS data analysis. Quality control includes careful examination of the statistics to make certain that; 1) enough reads are available for *de-novo* assembly; 2) that quality is high enough for SNPs detection; and 3) that adapters/primers were not sequenced as part of the reads. More details about technical problems can be found in PIQA [20].

The quality of each nucleotide is measured as the probability (*e*) of the base to be incorrectly sequenced. The following formula calculates ***Phred*** **Score** which was first introduced in 1998 [10, 11].

$$Q = -10\ log_{10}\ (e)$$

FASTQ files contain information about each read and quality saved in 4 lines:

Sequence identifier: It is a description line marked by symbol "@" and contains information about the instrument, run, flowcell, lane, read etc. separated by a colon ":".

Sequence: The second line stores the read.

Quality score identifier: The line begins with a "+" character and usually has same description from the Sequence identifier line.

Quality score: This line contains ASCII encoding of Phred quality scores for each nucleotide in the sequence. Different Illumina software uses different encoding schemes. Illumina 1.3 implemented Phred+64 ASACII encoding scheme while 1.8 uses Sanger format (Phred+33) for encoding the quality scores.

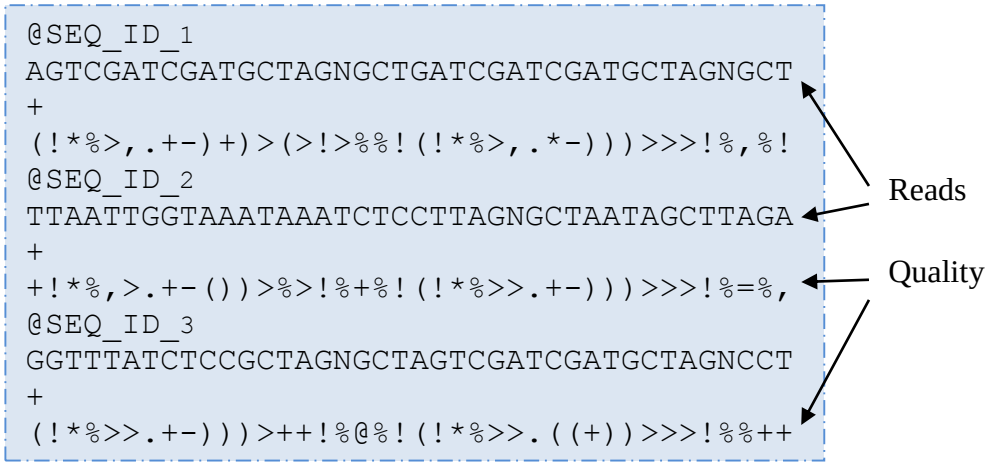


Figure 2.5: FASTQ file format

The *reads* are checked for low quality, low entropy and presence of adaptors or unknown nucleotide 'N' (Figure 2.6 and 2.7). Adaptors are known short sequences used in library preparation which help in immobilizing the unknown DNA fragments during sequencing.

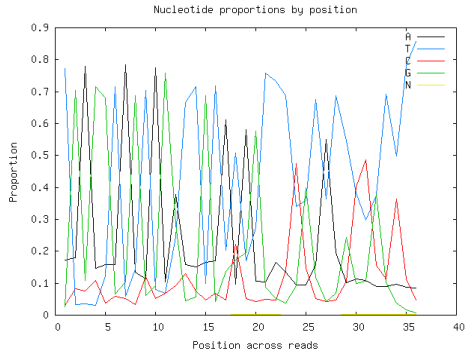


Figure 2.6: Nucleotide (A/ T/ G/ C/ N) proportions in all reads

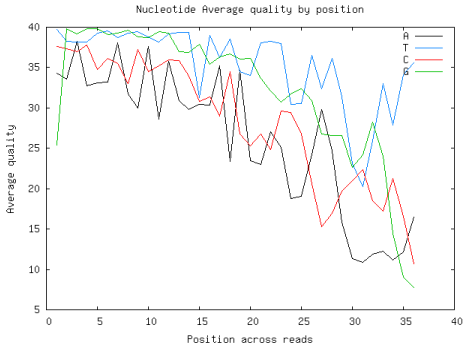


Figure 2.7: Nucleotide average quality by position

Illumina uses imaging technology to detect bases and often due to an unclear or blurred image correct basecall cannot be made. These ambiguous or unknown nucleotides are marked as 'N' in the FASTQ file. An in-house developed tool, *Slim-Filter* [20], is used to assess quality and discard low quality reads. Sequencing is repeated for a sample if majority of the reads fall below quality and get discarded in filtration process.

Mapping

Mapping is the process of aligning reads to reference sequences which can be a human or virus or bacterial genome. Mapping is one of the preliminary steps before the downstream analysis. *Query sequences* can be aligned against a *target sequence* allowing perfect matches, mismatches, or gaps. The output of mapping is coverage data which reflects the copy number variation within the individual's genome when compared to a reference.

Sequencing errors which are artifacts of the sequencing platforms or protocols prevent mapping reads with greater accuracy. Other issues like short read length, lower coverage depth, huge volume of reads generated, repetitive regions, and genetic variation present in the population all make mapping a complex process.

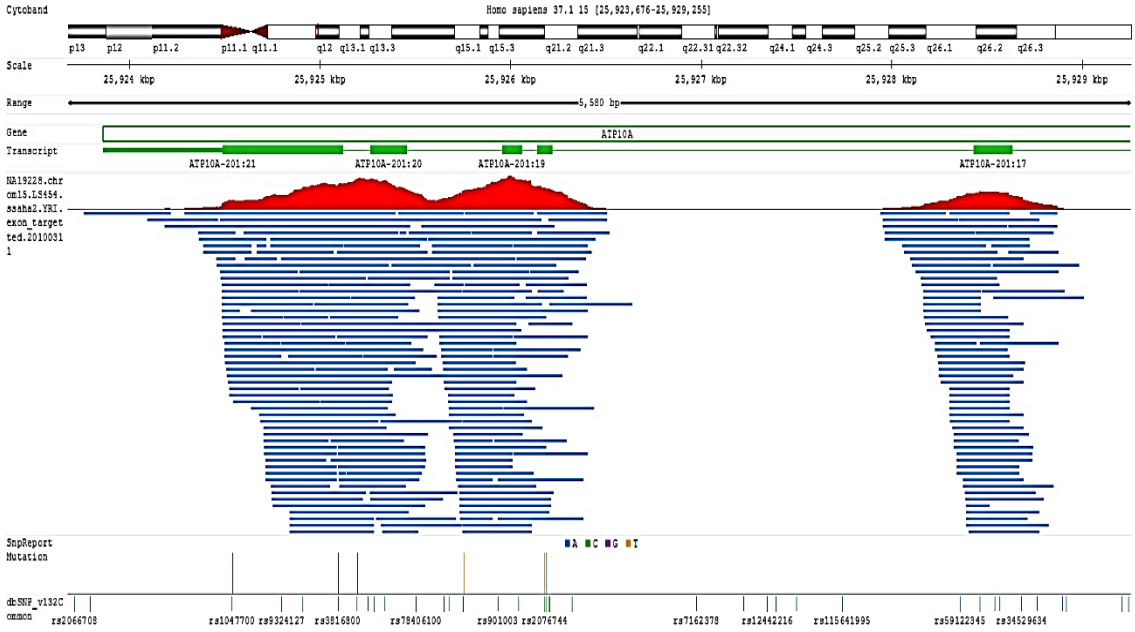


Figure 2.8: Genome coverage on a region of human chromosome 15 (Arrayserver.com)

After quality assessment is complete, there is no further need to save the quality codes in FASTQ files. In order to save processing time and space while mapping, FASTQ files are converted into FASTA files with extension *.fasta*. FASTA files serve as input to sequence search and alignment tools such as BLAST and Clustal.

Every single entry in FASTA file consists of 1 line description starting with symbol ">" followed by the sequence. The description line may contain sequence identifier, name and other additional information. FASTA files can be used to store nucleotide, protein, amino acid, or RNA sequences/ regions.

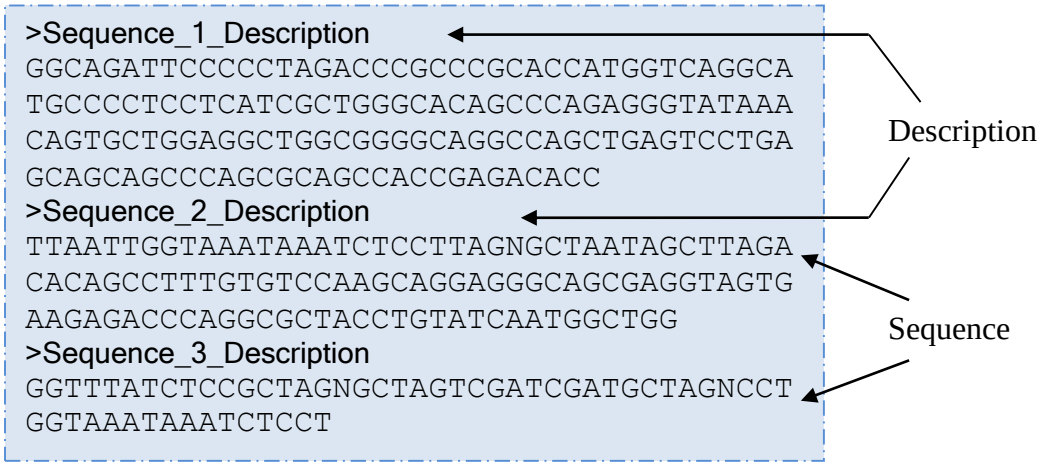


Figure 2.9: FASTA file format

Generally while mapping, sample reads are aligned against reference sequence(s) one by one (Figure 2.8), allowing perfect matches and number of matches are counted for every location in the genome. The position-by-position count of the number of reads mapped to the reference is called **reads coverage**. An in-house developed software,

mapping pipeline was used to perform sequence alignment which implements tree and hash-like data-structures [19].

The Mapping toolbox contains novel data-structures: a) to store reads and reference sequences; b) to search reads; and c) to store mapping output or reads coverage i.e. position-by-position count of number of reads mapped to the reference.

1. Store Reads in AS format (Figure 2.10):

   Base_Array_Subsequences, Array_Subsequences

2. Store Reference Sequences in ARS format:

   Reference_DNA_Sequence, Array_of_Reference_DNA_Sequences

ARS format has following format: It begins with total number of references and every entry starts with ">" character followed by Sequence_id, Sequence_length, Sequence_description_length, Sequence_description

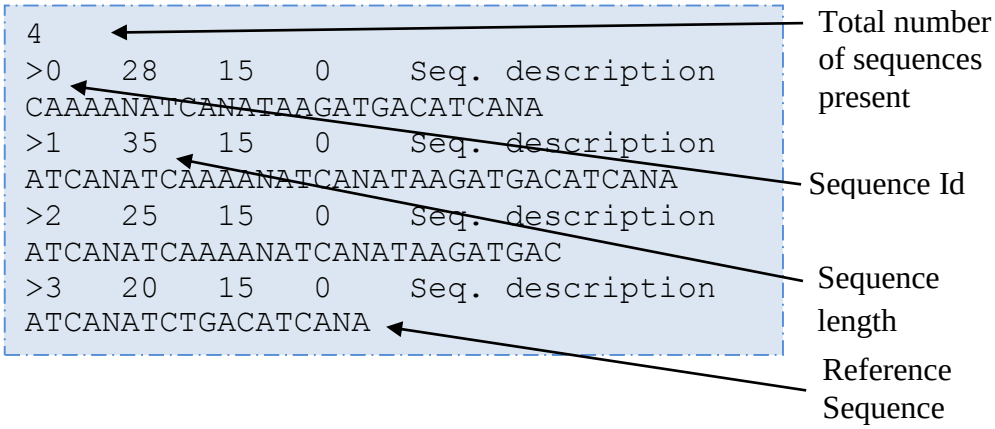


Figure 2.10: ARS file format

3. Perform efficient search using *BF_Tree*

4. Store Mapping output or DNA coverage in ARDS format:

DNA_Sequence_Coverage, Array_of_DNA_Sequence_Coverages

ARDS format has following format. It begins with reads length followed by total number of references and every entry starts with information about the coverage array: Sequence_id, Sequence_length, Average coverage. The next line contains location-by-location coverage values.

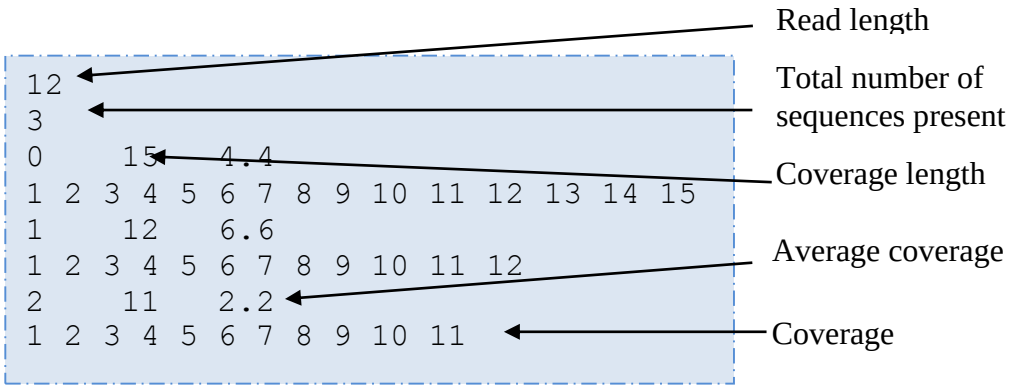


Figure 2.11: ARDS file format

Following is the detailed description of each of these datastructures. The datastructures were developed and implemented using C++ language.

Base_Array_Subsequences, Array_Subsequences:

The reads procured from the sequencing process are generally saved in the conventional file formats i.e. FASTQ or FASTA. These files store every read/ subsequence with additional quality or *run* information. Sequencing output contains a large number of

duplications (around 30-60% of reads are repeats) and storing them in traditional file format is not efficient which provided motivation to propose novel file formats. AS file format was designed to store every read and it's frequency in a single line in a tab-delimited file.

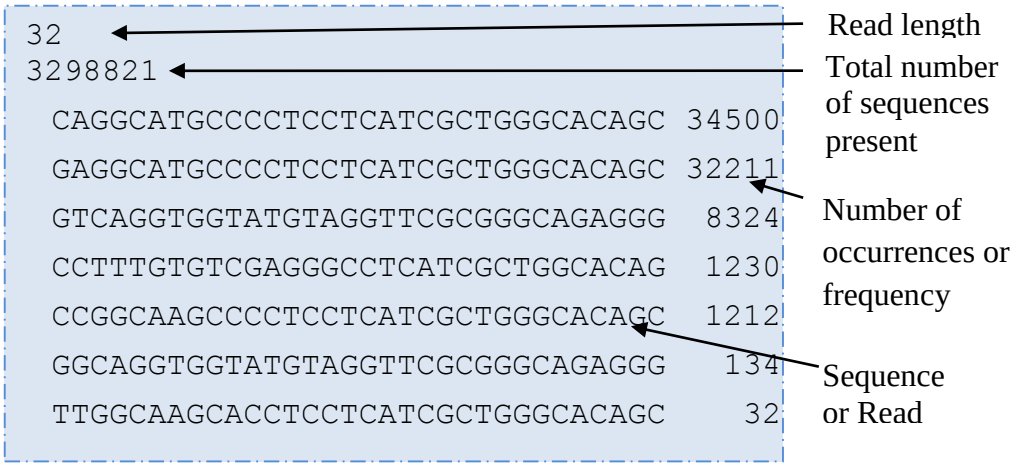


Figure 2.12: AS file format

Base_Array_Subsequences class basically saves equally sized reads and their copy numbers. It has two important member variables: a) a 2-dimensional character array to store the reads; and b) another synchronized 1-dimensional integer array to store their corresponding frequencies. If the array size becomes too small to add new subsequences, the *Base_Array_Subsequences* object automatically increases its size by a factor of 1.5. *Array_Subsequences* is an inherited class from *Base_Array_Subsequences* which stores DNA sequences allowing only A/ T/ G/ C/ N characters.

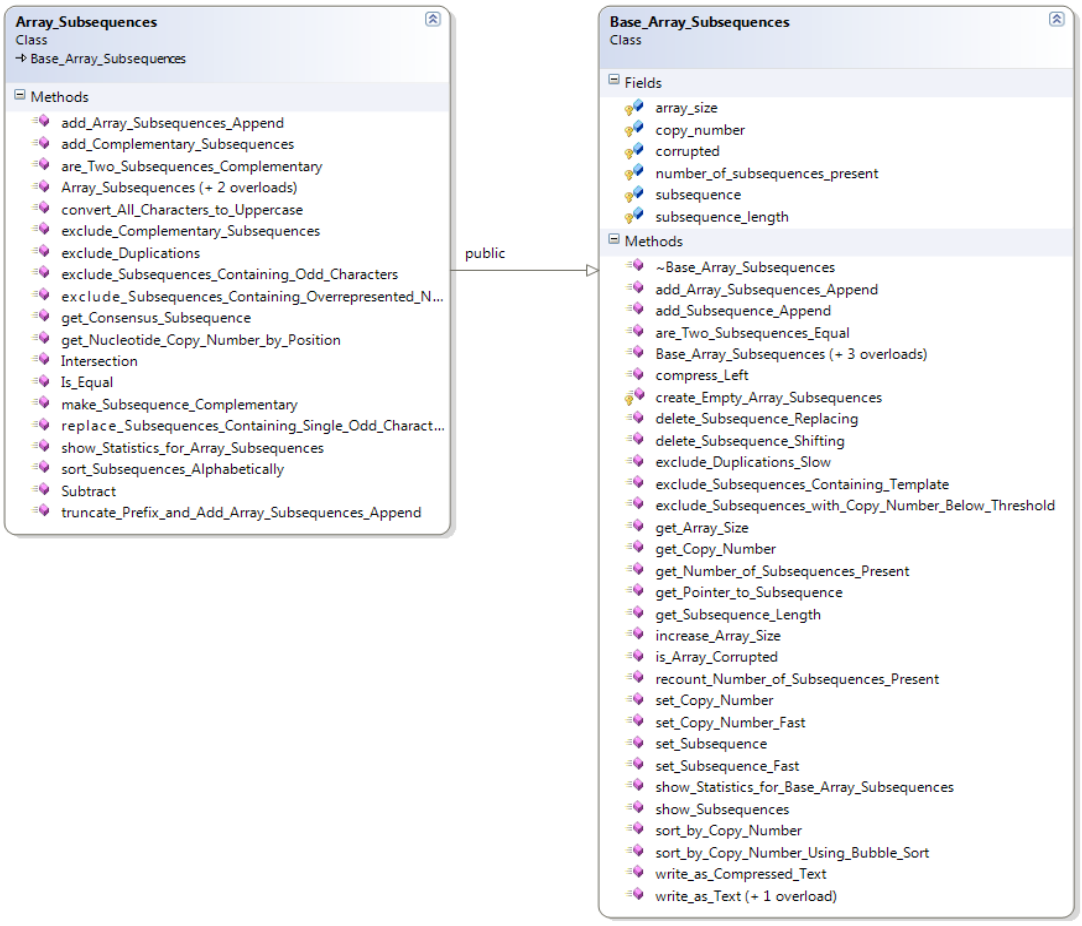


Figure 2.13: Class diagram of *Base_Array_Subsequences* and *Array_Subsequences*

The design of *Base_Array_Subsequences* and *Array_Subsequences* provides user with fast sorting mechanism. The reads can be sorted alphabetically or based on their copy numbers. The base class, *Base_Array_Subsequences* combines two sorting algorithms *counting sort* and b*ubble sort* to sort the array of subsequences' frequencies efficiently. A significant number of reads are present as single copies in human genome when disassembled into smaller *n-mer* of sizes 31, 32, and 36 [19]. This observation led to the

proposal and implementation of combining two sorting algorithms. The resulting algorithm sorts all the unique reads (existing in greater numbers) with computationally less expensive *Counting sort O(n)* and reads existing in multiple copies using *Bubble sort* $O(n^2)$. The inherited class *Array_Subsequences* provides functionality to alphabetically sort the sequences using Least Significant Digit (LSD) and Most Significant Digit (MSD) Radix sort with execution time of $O(k*n)$ where k is the number of entries and n is number of nucleotides in a read [19].

Together the storage containers, *Base_Array_Subsequences* and *Array_Subsequences* and the proposed sorting mechanism provide memory and time efficient design to store and map the reads.

In our implementation of *counting sort* algorithm, the class *Base_Array_Subsequences* has a 1D array of positive integers, copy_number[1 . . n] of size n where n is equal to number of subsequences present. The array is synchronized with a 2D character array containing sequences *subsequence [][]* and stores the count of each distinct subsequence with a minimum value of one. Two more auxiliary arrays are needed for storage, the array new_copy_number [1 . . n] holds the sorted output and the array counting_array[1 . . n] provides temporary working storage.

Algorithm SORT_BY_COPY_NUMBER ( )

Input:

array_size: unsigned integer value, size of array_subsequences

number_of_subsequences_present: total sequences present in array_subsequences

subsequence[number_of_subsequences_present][subsequence_length] 2d character array

copy_number[number_of_subsequences_present] initial 1d array holding frequencies of subsequences

Output:

Sorted *new_subsequence, new_copy_number* arrays

```
1.    MAX_COPY_NUMBER_FOR_COUNTING_SORT←10000
2.    for i ← 1 to MAX_COPY_NUMBER_FOR_COUNTING_SORT do
3.          counting_array [i] ← 0
4.    end for
5.    for i ← 1 to array_size do
6.          new_copy_number[i] ← 0
7.          new_subsequence[i] ← NULL
8.    end for
```

```
9.          num_of_seq_greater_than_max← 0
10.         for j ← 1 to number_of_subsequences_present do
11.                 copy_number← copy_number[j]
12.                 if copy_num >
                    MAX_COPY_NUMBER_FOR_COUNTING_SORT then
13.                 new_subsequence[num_of_seq_greater_than_max] ←
                    subsequence[j]
14.          new_copy_number[num_of_seq_greater_than_max] ← copy_num
15.          num_of_seq_greater_than_max← num_of_seq_greater_than_max +1
16.          continue
17.         else
18.           counting_array [copy_number] ← counting_array [copy_number] + 1
19.        end if
20.     end for
21.     for i ← 2 to MAX_COPY_NUMBER_FOR_COUNTING_SORT do
22.           counting_array [i] ← counting_array [i] + counting_array [i-1]
23.     end for
24.     for i ← 1 to number_of_subsequences_present do
25.           copy_num ← copy_number[i]
26.           if copy_num>MAX_COPY_NUMBER_FOR_COUNTING_SORT then
27.                 continue
28.           else
29.                 new_copy_number[number_of_subsequences_present-
                          counting_array[copy_num-1]] ← copy_num
30.                 new_subsequence[number_of_subsequences_present-
                    counting_array[copy_num-1]]← subsequence[i]
31.                 counting_array[copy_num-1] ← counting_array[copy_num-1] -1
32.           end if
```

```
33.     end for
        //bubble sort
34.     elements_switched←false
35.     if num_of_seq_greater_than_max>1 then
36.         for i ← 1 to num_of_seq_greater_than_max do
37.             elements_switched←false
38.             for j← 1 to num_of_seq_greater_than_max-1-i do
39.                 if new_copy_number[j]<new_copy_number[j+1] then
40.                   swap(new_copy_number[j],new_copy_number[j+1])
41.                   swap(new_subsequence[j],new_subsequence[j+1])
42.                   elements_switched ←true
43.                 end if
44.             end for
45.         end for
46.     end if
47.     if elements_switched=false then
48.         break
49.     end if
```

Reference_DNA_Sequence, Array_of_Reference_DNA_Sequences:

To align reads against assembled contigs or reference(s) like human / viral/ bacterial/fungal gene(s) or genome(s), the references should also be stored in memory. To store a reference or a set of references two classes namely *Reference_DNA_Sequence* and *Array_of_Reference_DNA_Sequences* were designed. *Reference_DNA_Sequence* primarily stores a reference as a 1D character array while *Array_of_Reference_DNA_Sequences* is an array of an object of *Reference_DNA_Sequence.*

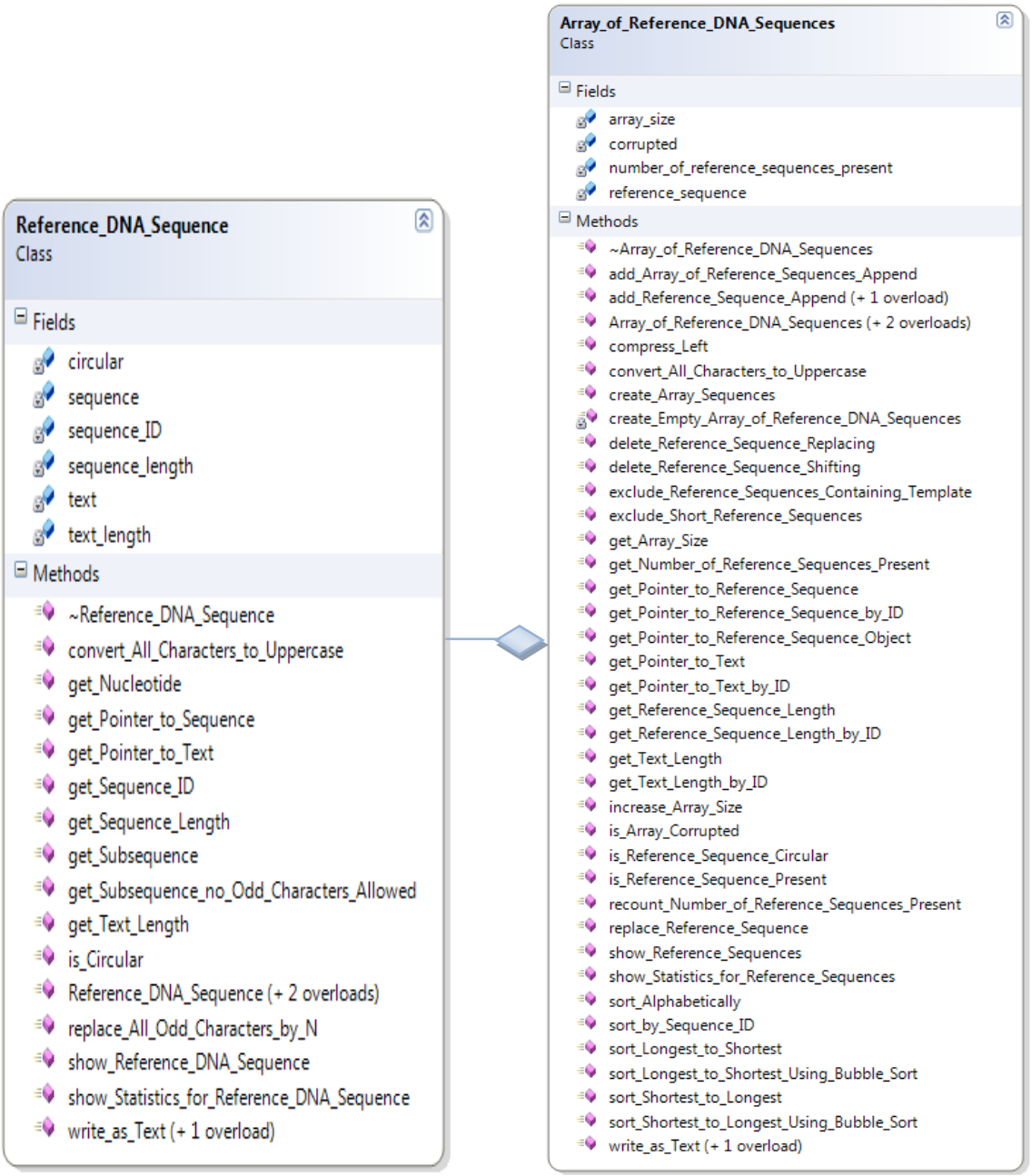


Figure 2.14: Class diagram of *Reference_DNA_Sequence and*

Array_of_Reference_DNA_Sequences

Bifurcation Tree or BF_Tree:

The complete design of BF Tree is divided into two classes, the basic *BF_Tree_Node* and *BF_Tree* class. They are specifically designed to efficiently store and search short sub-sequences while mapping.

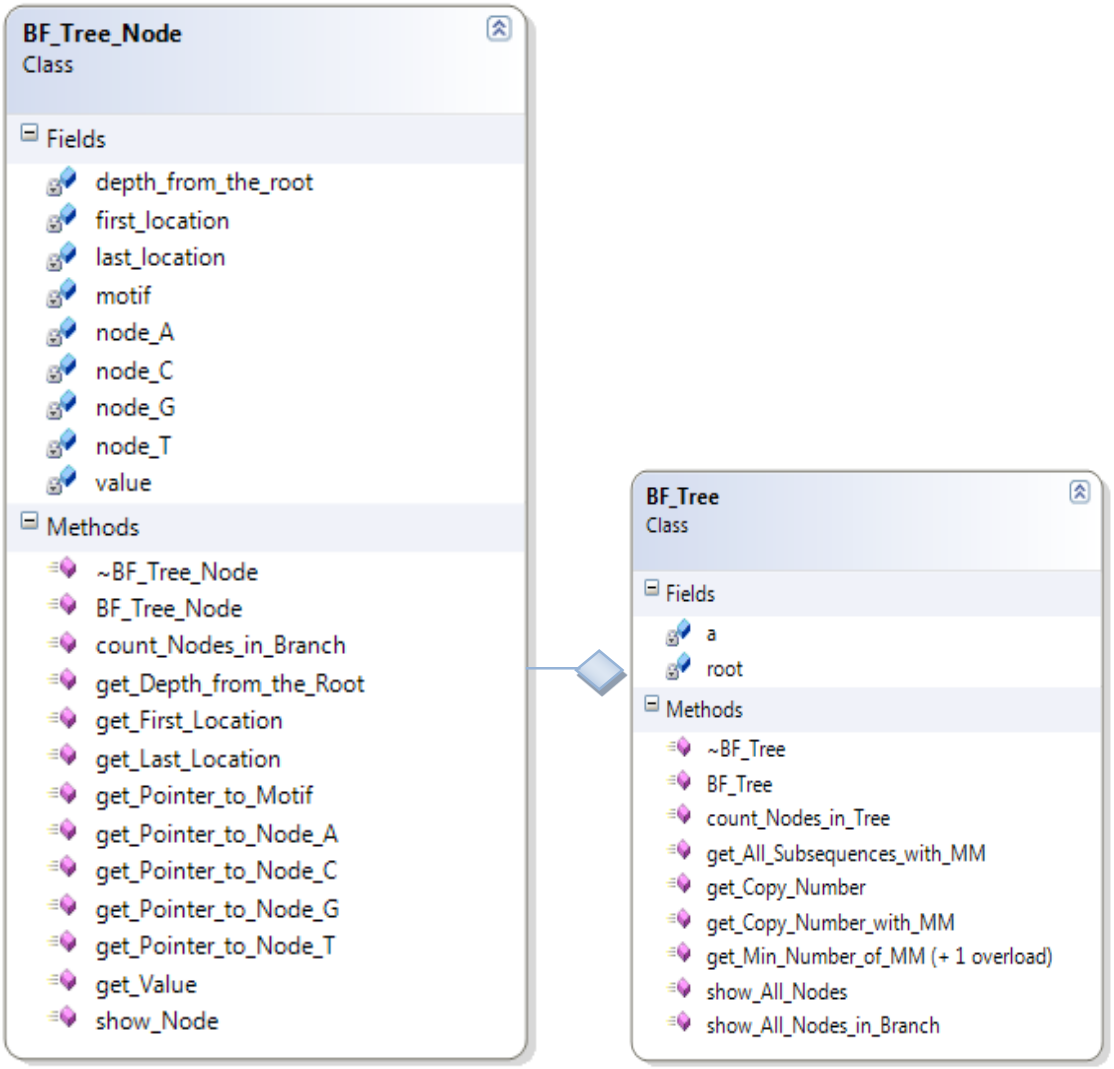


Figure 2.15: Class diagram of *BF_Tree_Node* and *BF_Tree*

In the Mapping pipeline, a *BF_Tree* is constructed from *Array_Subsequences* object containing the initial unmapped reads. All possible *n-mers* of size equal to the read are extracted from the reference and later these subsequences are searched in the *BF_Tree*. The approach hence maps reference against the *reads* which is contrary to the usual technique of mapping reads against the reference. The strategy of mapping reference to reads is very efficient considering the millions of reads to be aligned per sample.

Once mapping is complete the mapping scores or *coverage* values are obtained across the whole reference sequence for each sample. The array of integer numbers that represents the alignment scores is termed as *Reads Coverage* of sample on the given reference sequence. One of the initial steps is to visualize the coverage values as a line graph (Figure 2.18). In the line plots, the horizontal axis refers to position in the genome and vertical axis represents coverage value at the corresponding location.

Figure 2.16: Mapping *reads* to reference sequence to obtain *coverage* values

DNA_Sequence_Coverage, Array_of_DNA_Sequence_Coverages:

These two classes store the coverage values as positive integer values. *DNA_Sequence_Coverage* class has a 1D array of unsigned integer values corresponding to the reads coverage determined by mapping process. In order to store coverage values corresponding to multiple reference genomes stored in *Array_of_Reference_DNA_Sequences,* an equivalent class *Array_of_DNA_Sequence_Coverages* was designed which is an array of object of class *DNA_Sequence_Coverage.*

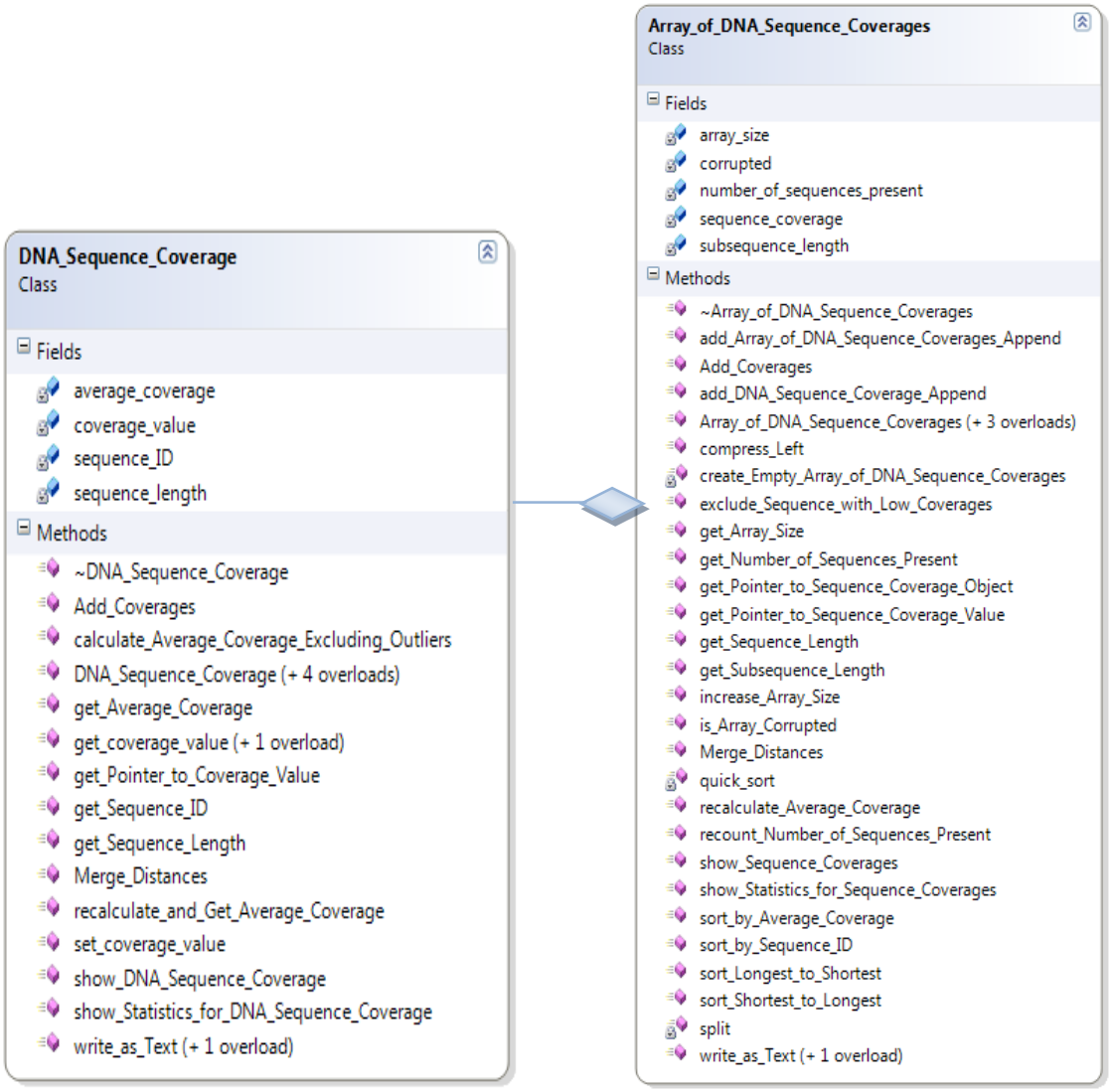


Figure 2.17: Class diagram of *DNA_Sequence_Coverage*, *Array_of_DNA_Sequence_Coverages*

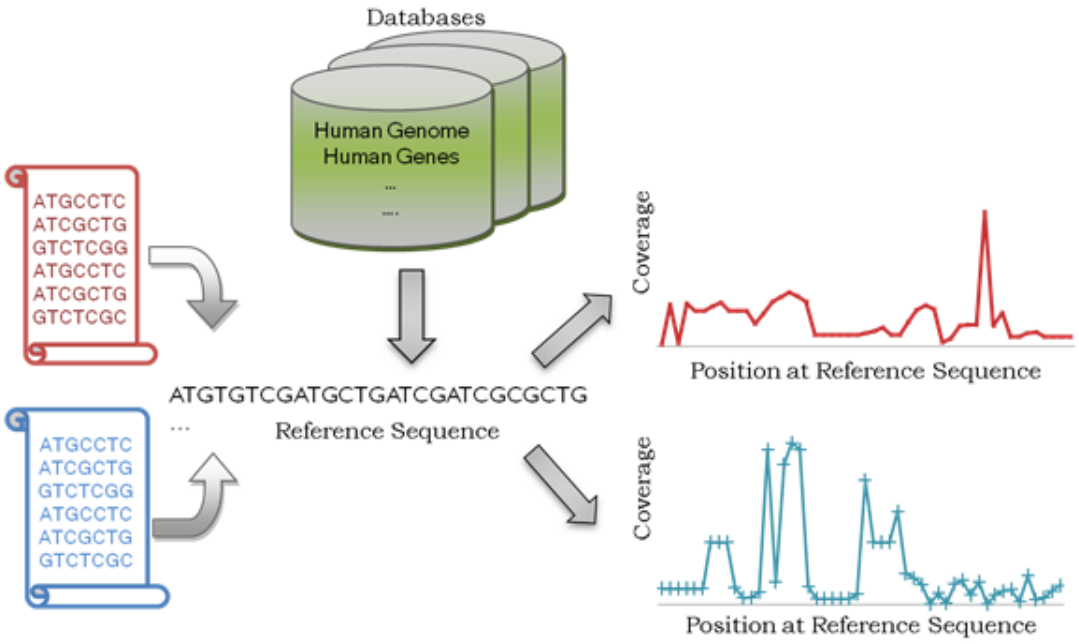


Figure 2.18: Mapping *reads* from 2 samples to reference sequence stored in database to obtain *coverage* value at every location

---

Coverage Data Analysis: The goal of our research is to detect CNV using HTS data by mapping "reads to reference" approach.

2.3 Sequence Alignment-based Analysis: Challenges

Apart from technical challenges briefly discussed in section 2.2, alignment-based analysis also face challenges due to biological properties of the genome in consideration. DNA repeatable sequences and single nucleotide mutations present in the genome modify alignment scores and often add bias to the mapping output.

DNA Repeat Sequences

The human genome contains large numbers of repetitive DNA elements (~48%) which range from two bases (mono- or di-nucleotides) to a complete gene [68]. Interestingly, only 1 % of the entire human genome encodes protein sequences or functional molecules. The remaining mostly repetitive DNA, was thought to be non-functional and even termed as 'junk DNA'. Soon it was discovered that the non-functional DNA plays an important role in evolution. For example, transposable elements, a family of repeat DNA sequences have been involved in the making of genes like RAG1 [68]. Some repeat sequences are associated with disease syndromes like myotonic dystrophy, oculopharyngeal muscular dystrophy, congenital hypoventilation syndrome, and mental retardation [35] [54].

Studying the role and functioning of DNA repeats is as important as of the non-repeatable regions; the repeatable sequences create technical problems while computing and analyzing the mapping results. The problem is illustrated using figure 2.17. In the figure, the coverage of methylated and non-methylated samples on human chromosome 5 is drawn as a line graph. It is clear that some regions in the human genome depict very high

coverage values. These *high peaks* correspond to human DNA repeats and affect our ability to visualize and analyze methylation levels at lower coverage areas. To be specific, the higher coverage values compromise detection of differentially covered regions. The sequences map at multiple locations and give a false impression of highly covered section.

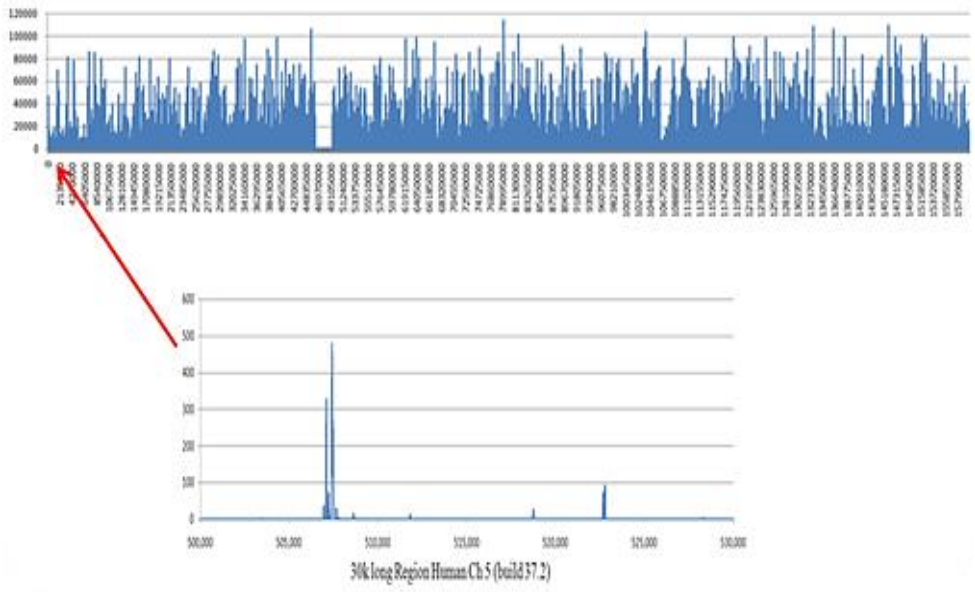


Figure 2.19: Effect of repeatable regions: mapping reads to human chr 5 (Build 37.2)

Single Nucleotide Polymorphisms (SNPs)

Single nucleotide changes or point mutations reduce effective reads coverage around the location of the SNP. The dbSNP databases provided and maintained by NCBI (*National Center for Biotechnology Information*) stores short variations in nucleotide sequences from a wide range of organisms. The database includes both common and rarely occurring SNPs. The SNPs represent polymorphism in the populations and during the process of sequence alignment subsequences containing the mutations do not map to the reference genome. The non-alignment of reads in these regions results in no coverage values (Figure 2.20). These mutations disrupt the sequence composition in a non-random manner.

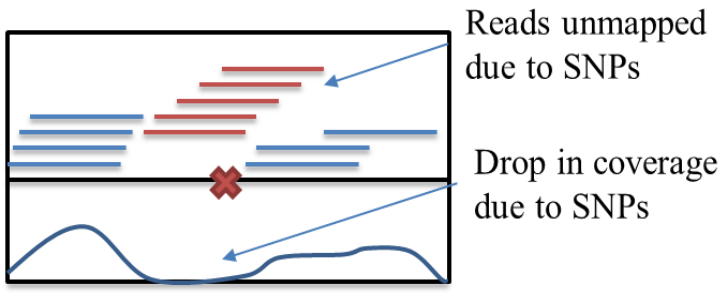


Figure 2.20: SNPs reduce effective coverage

Chapter 3

ALGORITHMS TO REDUCE EFFECTS OF REPEATABLE REGIONS

3.1 Current Methods to Calculate Coverage Values

Average Coverage

After aligning all of the filtered sequencing reads to a genome, the coverage at every location in the genome is obtained and the average coverage can be calculated as:

$$Avg.Cov. = \frac{\mathrm{n}}{\mathrm{L}} * \sum_{k=1}^{L} (x_{\mathrm{k}})$$

where, *n* length of the reads;

L length of the genome;

k location in the genome.

Considering the vast collection of single and multi-cellular organisms present on our planet, genome lengths can vary from few kilobases (RNA Viruses) to several gigabases in plants, mammals, etc. The study and visualization of the coverage data over the large genomes can sometimes be cumbersome. One of the possible ways to represent coverage is to average the coverage over a window of fixed length.

The position-by-position coverage values are hence studied over contiguous non-overlapping windows. For each window the *average coverage* is calculated using the following formula.

$$Avg.Cov. = \frac{\mathrm{n}}{\mathrm{w}} * \sum_{k=1}^{w} (x_{\mathrm{k}})$$

where, *n* length of the reads;

w length of the non-overlapping window;

k location in the window.

Finding average coverage across the genome is a simple way of estimating genome coverage but it is not free from defects. Average is not a good representation of the coverage in the presence of large number of DNA repeats and SNPs. The *average* statistic is affected by outlying values or outliers. Figure 3.1 illustrates this problem. The unusually high coverage spikes skew the coverage value while SNPs reduce effective coverage in a small region.

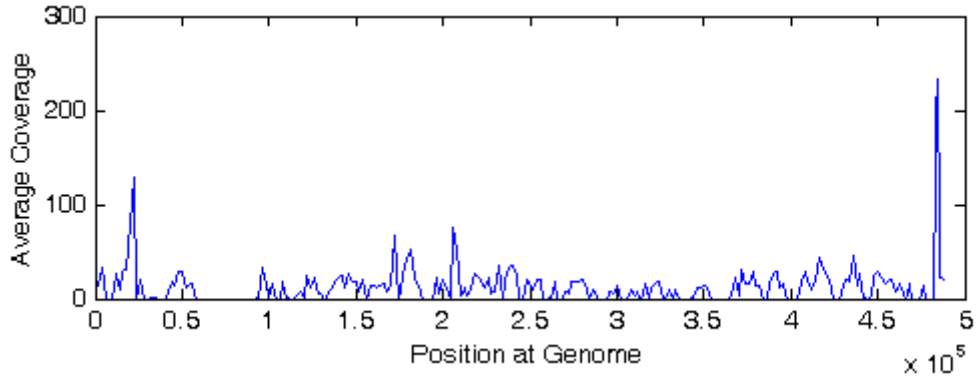


Figure 3.1: Coverage value across genome shows presence of repeat sequences and SNPs

Mask/ Exclude *Repeat Sequences* from Downstream Analysis

The presence of highly repeatable sequences in human genome interferes with the accurate estimation of the average genomic coverage. To eliminate the bias introduced by the repeatable sequences, one can exclude them from the downstream coverage analysis. However, many genomic elements like telomeres and centro-meres have not been integrated into the human reference sequence, making it difficult to identify all DNA repeats. This implies that the coverage spikes cannot be completely removed and the approach requires further improvement.

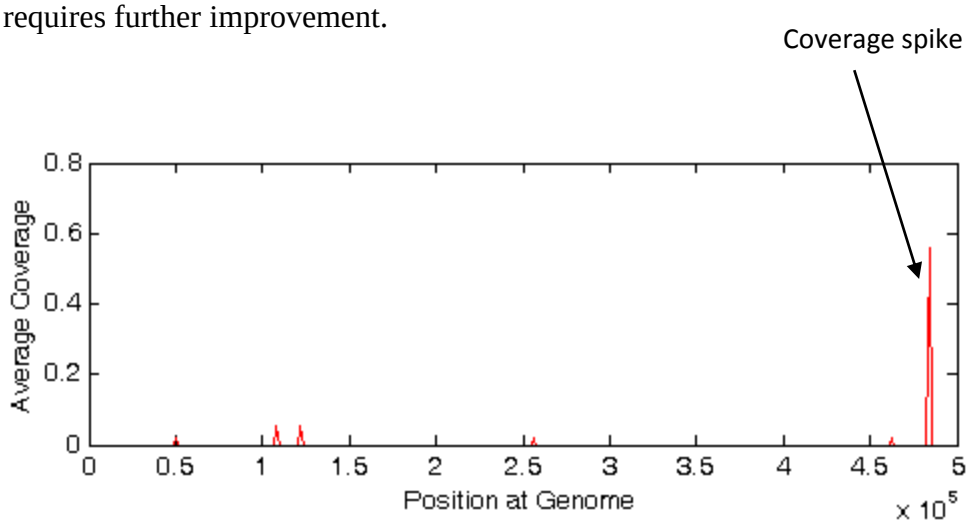


Figure 3.2: Coverage value across genome: Some coverage spikes not removed even after masking repeat regions

Excluding top and bottom X%

One of the possible ways to improve coverage estimation is removing the top and bottom x% of coverage values from the analysis. This method, however, does not guarantee removal of all the repeated sequences. The example illustrated in Figure 3.3 shows a real case where the repeated sequences cover more than 75% of the total number of locations in the selected window. In such a scenario, even after excluding top and bottom x% of coverage values, it is not possible to remove the coverage pile-ups completely.

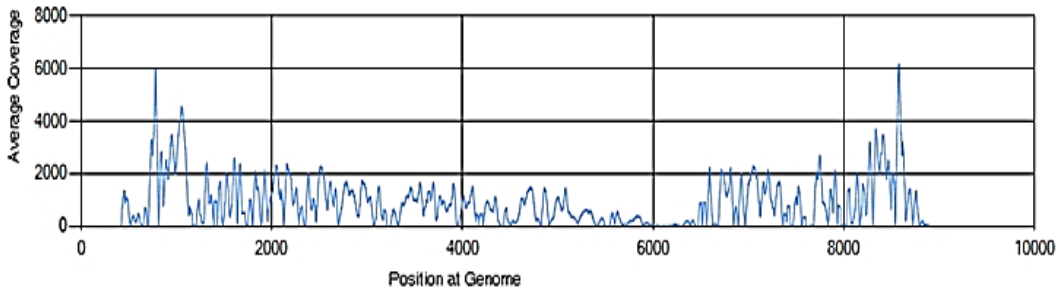


Figure 3.3: Coverage values across region containing mostly repeats

---

In summary, novel and more robust approaches are needed to estimate the average genome coverage.

3.2 Proposed Approach: A Poisson-based model

The copy number distribution of the reads mapped to a reference sequence should ideally follow a Poisson distribution, assuming that nucleotide distribution across genome can be modeled as a random process and the sequencing reads came from random locations in a genome. The coverage data, however, exhibit over-dispersion in the extreme ends of the distribution (as illustrated in Figure 3.4). Repeatable sequences and SNPs (single nucleotide differences between the reference and individual genome) and sequences with unknown characters 'N' cause these unexpectedly high coverage frequencies.

3.2.1 Basic idea

To give an overview of our proposed approach, consider a simple experiment. As mentioned before, if the nucleotide sequences present in the genome were randomly distributed, the frequency distribution of the coverage values would look like a Poisson process model [15].

To test this hypothesis, reads of length 20 corresponding to 5x coverage were randomly selected from (*E. coli)* genome. All of the reads were then aligned to the genome from which the reads originally came from. The line graph of location-by-location reads coverage showed unexpected high coverage values. Moreover, the average coverage value was now higher than the initial value of 5. The following experiment illustrates the variations in genomic sequences present in the form of repeats.

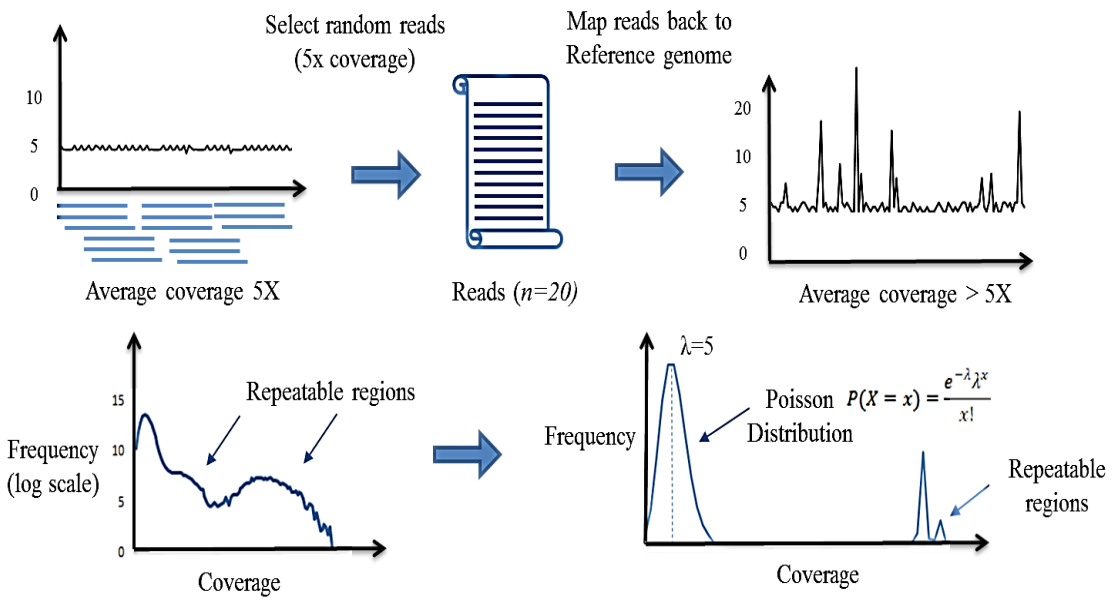


Figure 3.4: Basic idea of the Proposed Approach

The presence of repeatable regions was clearly visible in the distribution of coverage values in log scale. However, the coverage distribution still is close to Poisson distribution and varies slightly from the standard Poisson (mean=5). This simple experiment (Figure 3.4) demonstrated: a) the presence of repeatable regions; b) effect of repeatable regions on the estimate of average coverage value; and c) the empirical or observed coverage values are still close to standard Poisson model.

Assuming that coverage is a mix of Poisson (λ) and other distributions, it is possible to estimate parameter of Poisson distribution (λ) and proportion points coming from Poisson in observed coverage distribution.

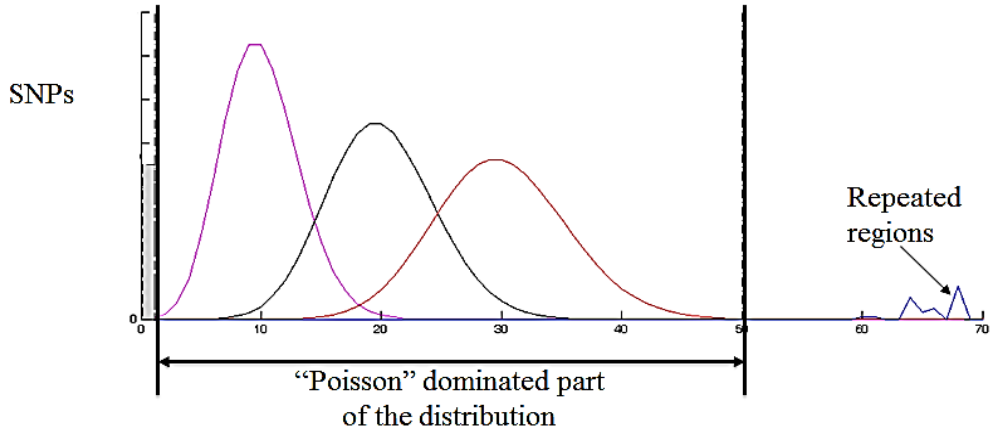


Figure 3.5: Main assumption: A portion of the observed coverage distribution contains Poisson-dominated distribution

---

Figure 3.5 illustrates the main assumption of the Poisson-based model. The main assumption is that a part of the coverage distribution contains Poisson-dominated distribution and the outlying values including coverage from SNPs and repeats are at far end of the distribution.

3.2.2 Estimation of model parameters

Assuming the genomic coverage demonstrate Poisson-like distribution, the next question is how the parameters of the model will be estimated.

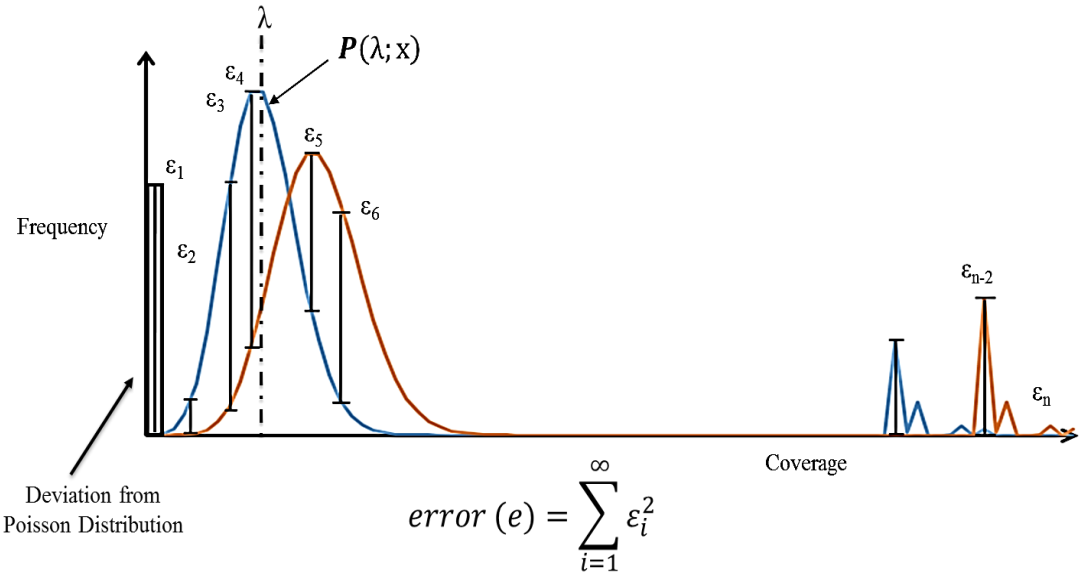


Figure 3.6: Estimating parameters of the model (λ, α)

The probability of a coverage value occurring at x time can be modelled as a combined probability from Poisson and non-Poisson distribution with α being percent of points coming from Poisson distribution. Let P (λ, x) and Z(x) be functions representing Poisson and non-Poisson distribution. Z(x) is contributed from outlying points and represent deviation from the Poisson distribution.

$$\mathrm{P}(x) = \alpha\, \boldsymbol{P}(\lambda; \mathrm{x}) + (1 - \alpha)\mathrm{Z}(\mathrm{x})$$

where, $(0 < \alpha < 1)$

To quantify the deviation of the observed coverage distribution from the expected Poisson probability distribution for different rate parameter values (*λ), the sum* of squares method was used. The error function calculates the error given by following formula:

$$S = \sum_{i=1}^{n} (O_i - E_i)^2$$

where, S sum of squares;

O_i observed coverage frequency or probability;

E_i expected (theoretical) Poisson frequency or probability;

n number of values.

To illustrate how the error values are affected from outliers at different values of lambda (λ) , the error values at different Poisson mean value (λ) were plot as a line graph. The error function is used to optimize parameter of the Poisson model (λ) and fit genomic coverage over the Poisson distribution. Figure 3.7 below shows how error values change at the extreme points in the either ends of the distribution. The results showed that error contribution from the extreme end points did not affected by the different empirical Poisson parameter (λ).

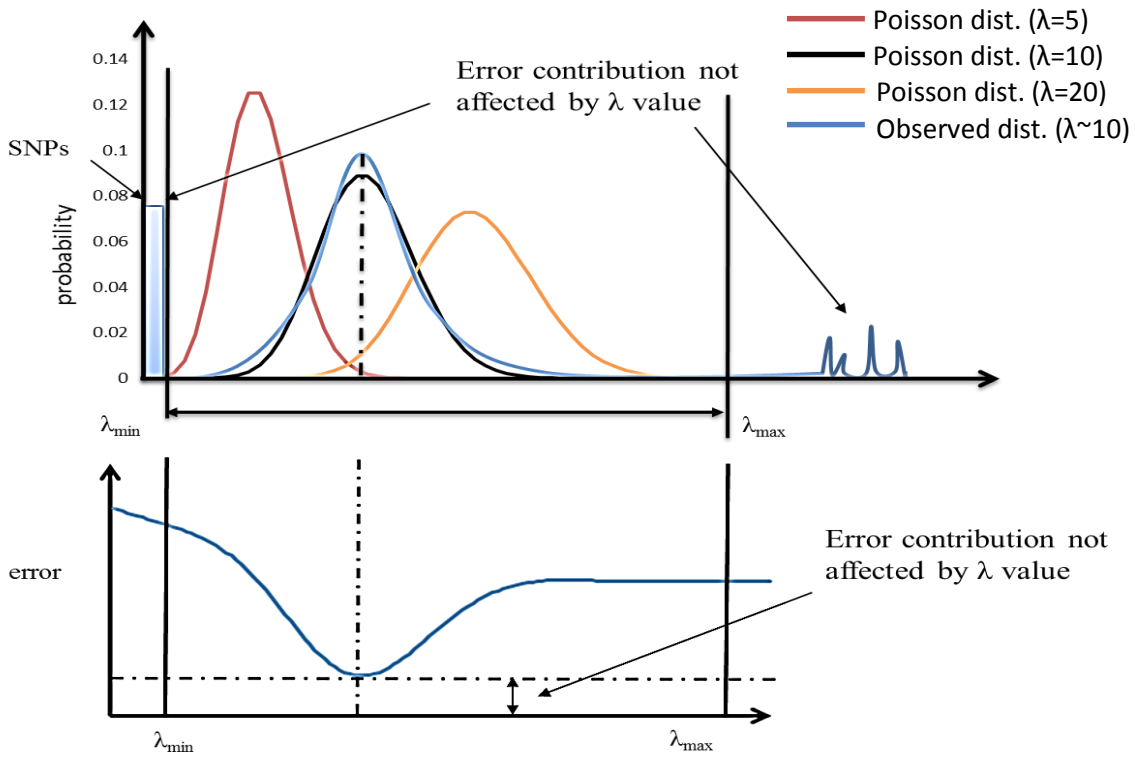


Figure 3.7: Error contribution not affected by lambda (λ) value.

Minimum error found at $(\lambda \sim 10)$

Hence, a model with single minimum error function 2-dimensional optimization (λ,α) is proposed. The objective function is given as:

$$\min_{0<\alpha<1}\left(\min_{\lambda_{min}<\lambda<\lambda_{max}}\sum_{i=0}^{n}\frac{\left(\mathrm{P}(x_i)-\alpha\,\boldsymbol{P}(\lambda;\mathrm{x}_i)\right)^2}{n}\right)$$

where,

P(x) the observed probability;

$\boldsymbol{P}(\lambda;\mathrm{x})$ the true or expected probability for given lambda and x values.

The model extends the standard Poisson model to adjust for the bias present in the genomic coverage values. The proposed model can be applied to the coverage data which takes positive integer values from $\{0, 1, 2\ldots\}$ in a genomic region. Since our data are not completely represented by the Poisson probability mass function, a model parameter α (alpha) representing the total percentage of values corresponding to a Poisson distribution is introduced.

Let alpha (α) be the percentage of values coming from Poisson distribution such that alpha ranges between 0 and 1 ($0 < \alpha < 1$). Then, the probability of a read being covered x times (or the coverage count) at location *i* can be calculated as:

$$\mathrm{P}(x_i) = \frac{f(x_i)}{N}$$

where,

$f(x_i)$ the total number of occurrences of the coverage value x;

N total number of points or genome length.

and all the probabilities sum up to 1.

$$\sum_{i=0}^{N} \mathrm{P}\,(x_i) = 1$$

Verifying main assumption on *E. coli* and human genome

To further verify the main assumption and test the composition and coverage of *E.coli*, reads were randomly selected from *E.coli* and then aligned back to the reference genome. For read length of 20 and different coverage depths (5, 10 and 20) coverage distribution was plot and compared with theoretical Poisson probabilities. The observed and expected coverage probabilities were in good agreement and matched with reasonable amount of accuracy. Following formula was used to calculate accuracy:

$$Accuracy = \sum_{i=1}^{n} \frac{|O_i - E_i|}{O_i} * 100$$

where,

O_i Observed coverage frequency or probability;

E_i Expected (theoretical) Poisson frequency or probability;

n Total number of values in frequency table.

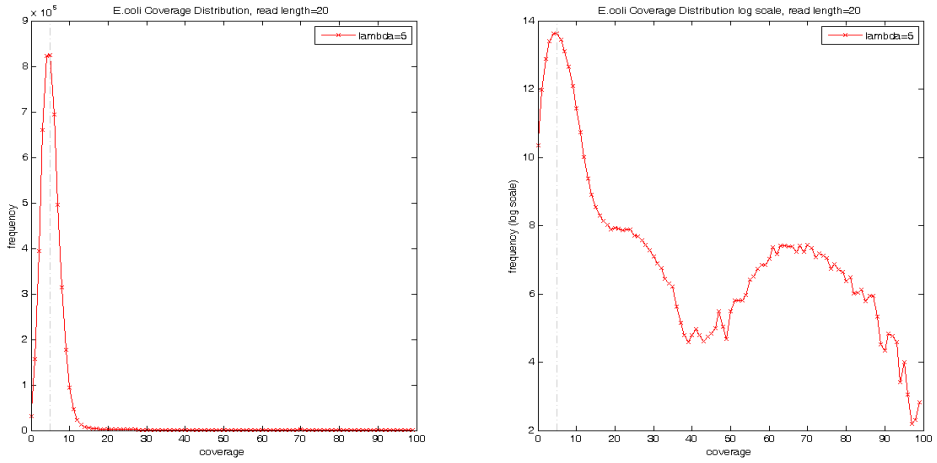


Figure 3.8: Coverage distribution of *E. coli* reads selected randomly for a given lambda (**5**) and read length (20) and mapped back to E.coli genome. The observed coverage distribution fits with a Poisson distribution (**λ=5**) with ~87% accuracy

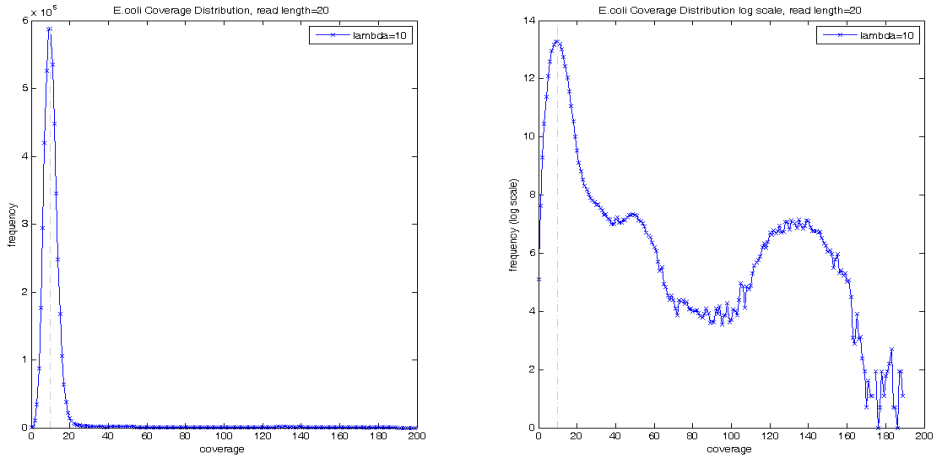


Figure 3.9: Coverage distribution of *E. coli* reads selected randomly for a given lambda (**10**) and read length (20) and mapped back to *E.coli* genome. The observed coverage distribution fits with a Poisson distribution (**λ=10**) with ~88% accuracy

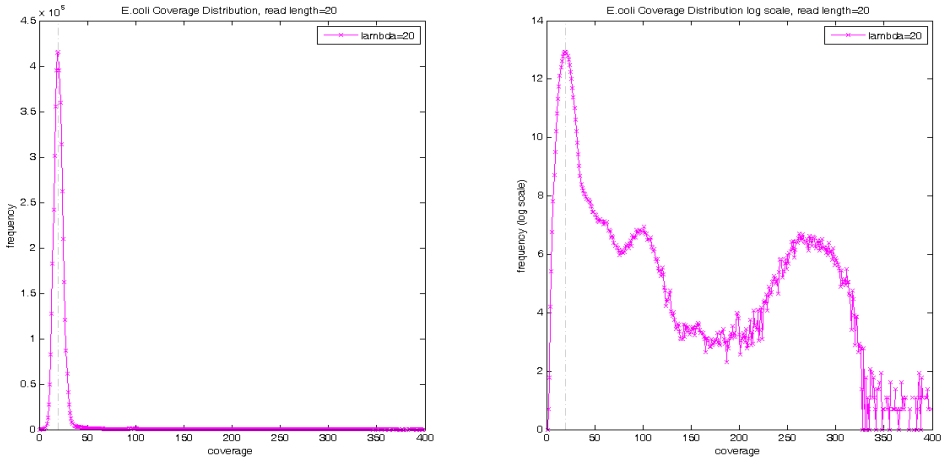


Figure 3.10: Coverage distribution of *E. coli* reads selected randomly for a given lambda (**20**) and read length (20) and mapped back to *E.coli* genome. The observed coverage distribution fits with a Poisson distribution (λ=**20**) with ~85.6% accuracy

---

Figures 3.8, 3.9, and 3.10 show that the coverage distribution of *E. coli* follows a Poisson distribution as the experimental and theoretical coverage distributions fit with reasonably good accuracy. To take the tests on *E. coli* a step further, SNPs were introduced artificially to the genome and check if the model is able to predict the average coverage value. A genomic sequence undergoes mutations over time. Figure 3.11, 3.12, and 3.13 shows the probability distribution of the *E. coli* coverage values with approximately 50,000 (1%) of SNPs. The coverage distributions again confirmed Poisson like characteristics.

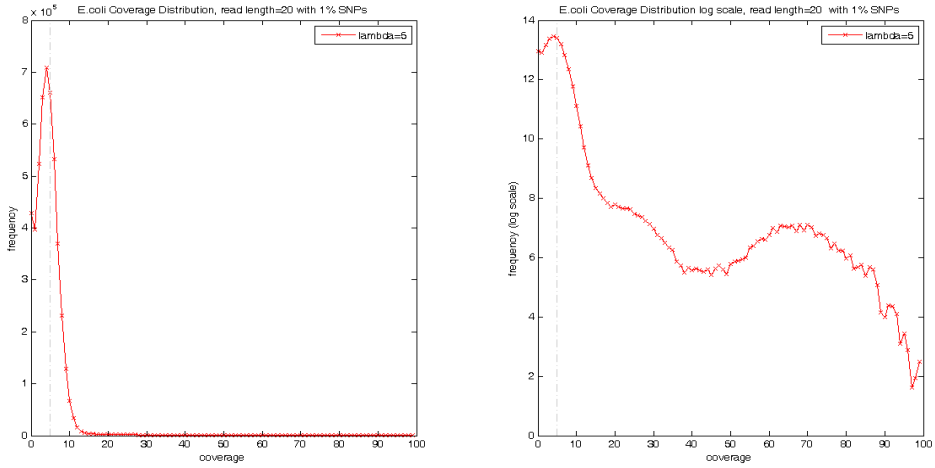


Figure 3.11: Coverage distribution of *E. coli* reads with **1% SNPs** selected randomly for a given lambda (**5**) and read length (20) and mapped back to *E.coli* genome. The observed coverage distribution fits with a Poisson distribution (λ=**5**) with ~90% accuracy

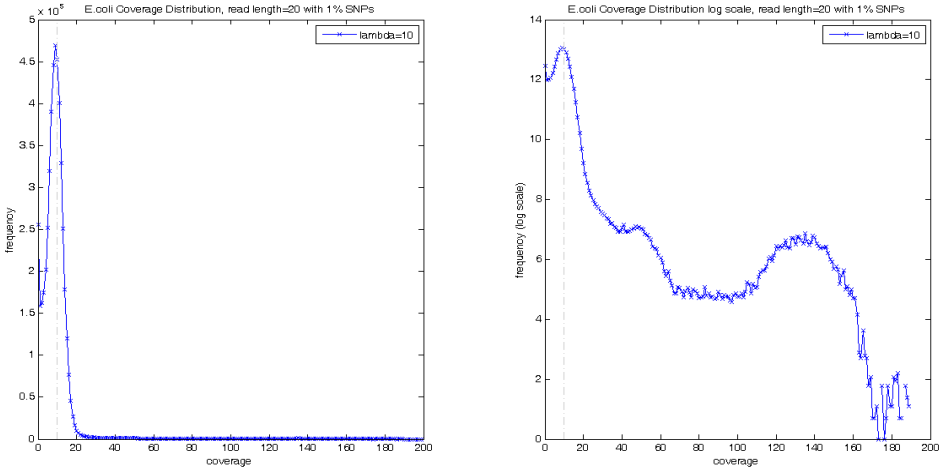


Figure 3.12: Coverage distribution of *E. coli* reads with **1% SNPs** selected randomly for a given lambda (**10**) and read length (20) and mapped back to *E.coli* genome. The observed coverage distribution fits with a Poisson distribution (λ=**10**) with ~91.7% accuracy

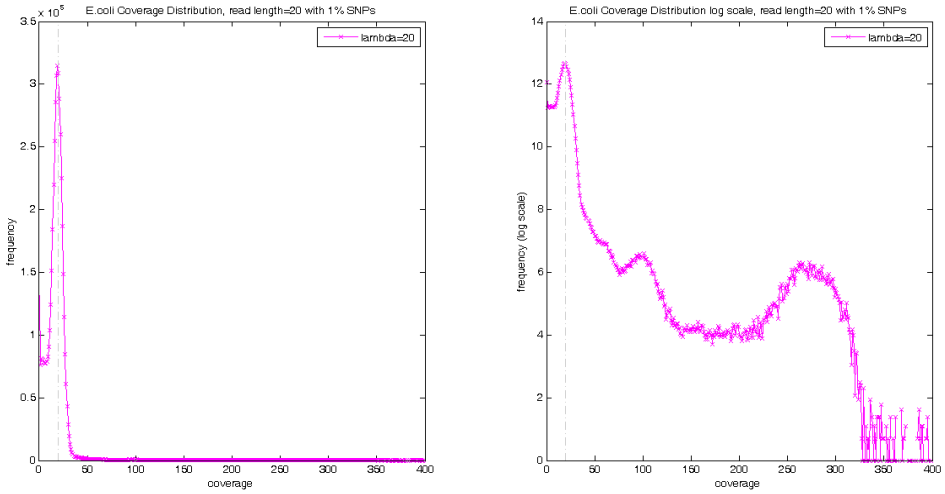


Figure 3.13: Coverage distribution of *E. coli* reads with **1% SNPs** selected randomly for a given lambda (**20**) and read length (20) and mapped back to *E.coli* genome. The observed coverage distribution fits with a Poisson distribution (λ=**20**) with ~88.5% accuracy

---

After observing that the assumption holds well on the *E. coli* bacterial genome, similar experiments were performed for human chromosome 1. For coverage depth of 5x, 10x, and 20x, reads were randomly selected from the chromosome and mapped back to the chromosome. The results were similar to that obtained with *E. coli* and the coverage distribution fit to Poisson distribution with high accuracy. The main assumption of the Poisson-dominated part being not too much affected with outliers was still holding good.

Characteristics of coverage obtained from human self-mapped reads are showed in the following plots (Figures 3.14, 3.15, and 3.16).

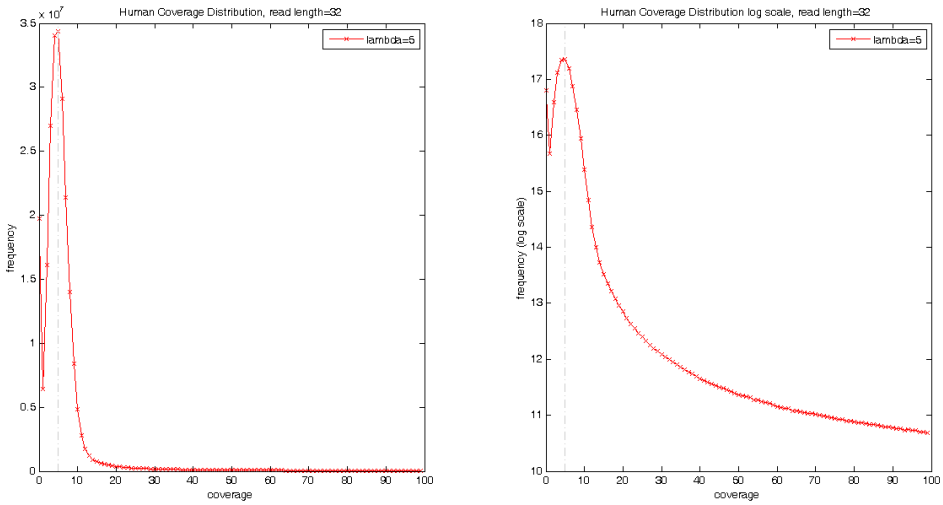


Figure 3.14: Coverage distribution of human chromosome 1 reads selected randomly for a given lambda (**5**) and read length (32). The observed coverage distribution fits with a Poisson distribution (λ=**5**) with ~99% accuracy

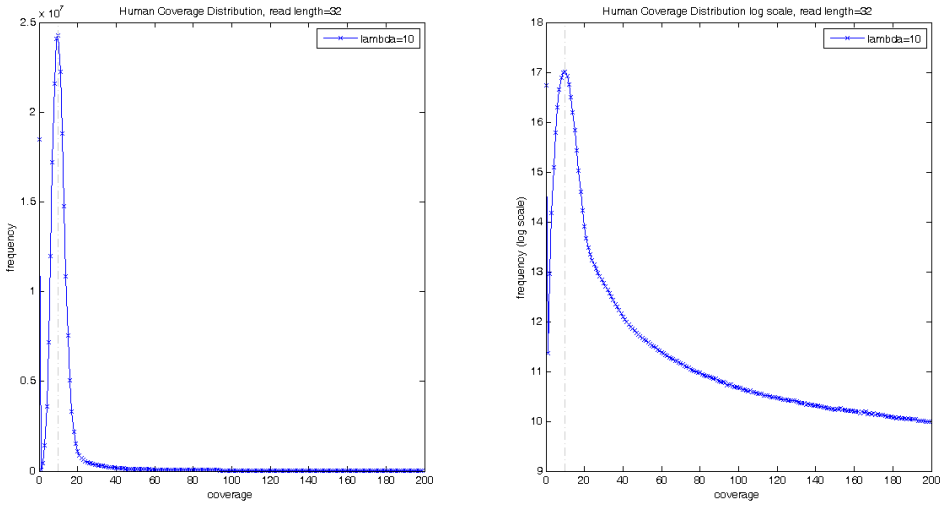


Figure 3.15: Coverage distribution of human chromosome 1 reads selected randomly for a given lambda (**10**) and read length (32). The observed coverage distribution fits with a Poisson distribution (λ=**10**) with ~98.5% accuracy

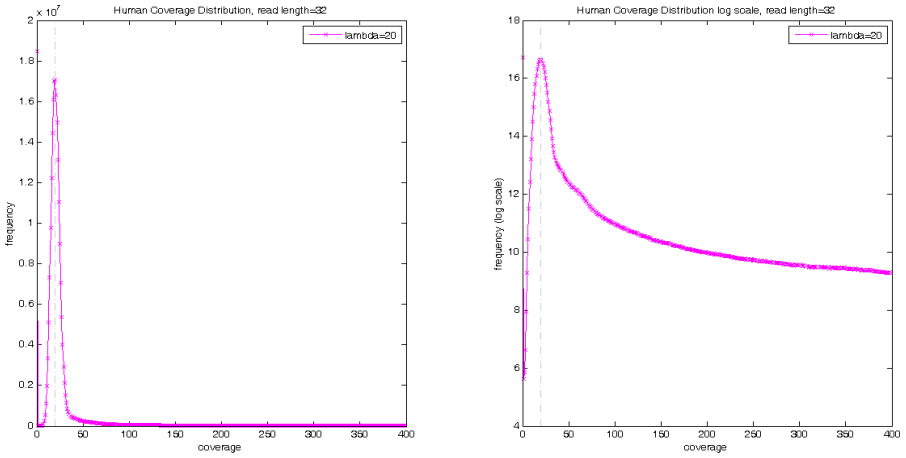


Figure 3.16: Coverage distribution of human chromosome 1 reads selected randomly for a given lambda (**20**) and read length (32). The observed coverage distribution fits with a Poisson distribution (λ=**5**) with ~97.6% accuracy

Next, in order to check effect of SNPs to coverage distribution in human chromosome, 1% SNPs were introduced to the human chromosome 1 and again the coverage distribution was plot both in normal and log scale.

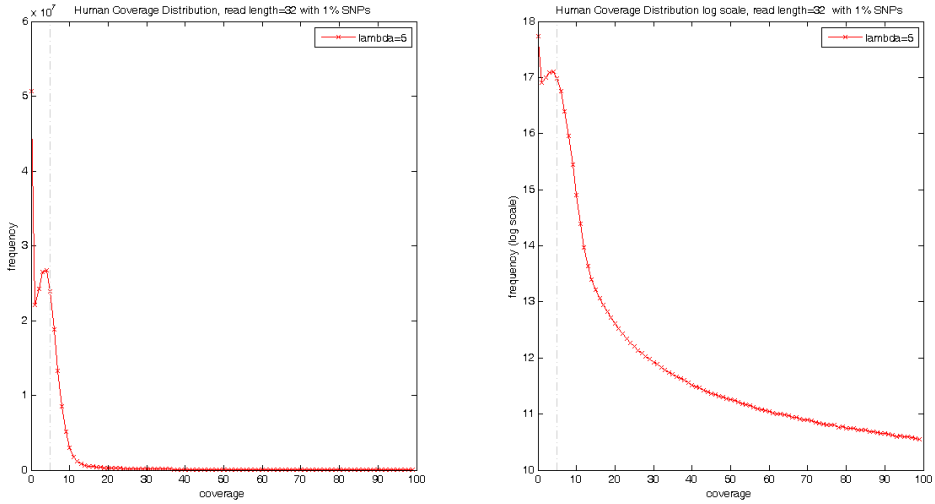


Figure 3.17: Coverage distribution of human chromosome 1 reads with **1% SNPs** selected randomly for a given lambda (**5**) and read length (32). The observed coverage distribution fits with a Poisson distribution (**λ=5**) with ~99.4% accuracy

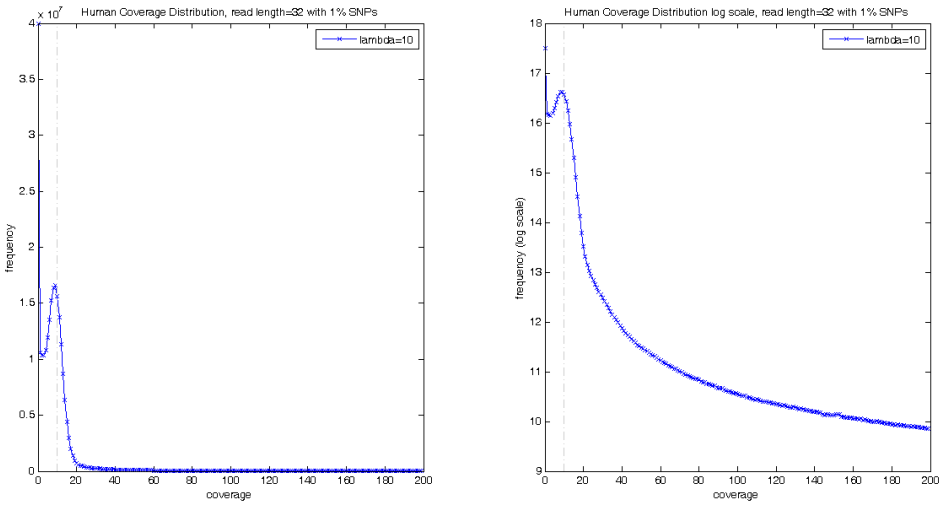


Figure 3.18: Coverage distribution of human chromosome 1 reads with **1% SNPs** selected randomly for a given lambda (**10**) and read length (32). The observed coverage distribution fits with a Poisson distribution (λ=**10**) with ~99.3% accuracy

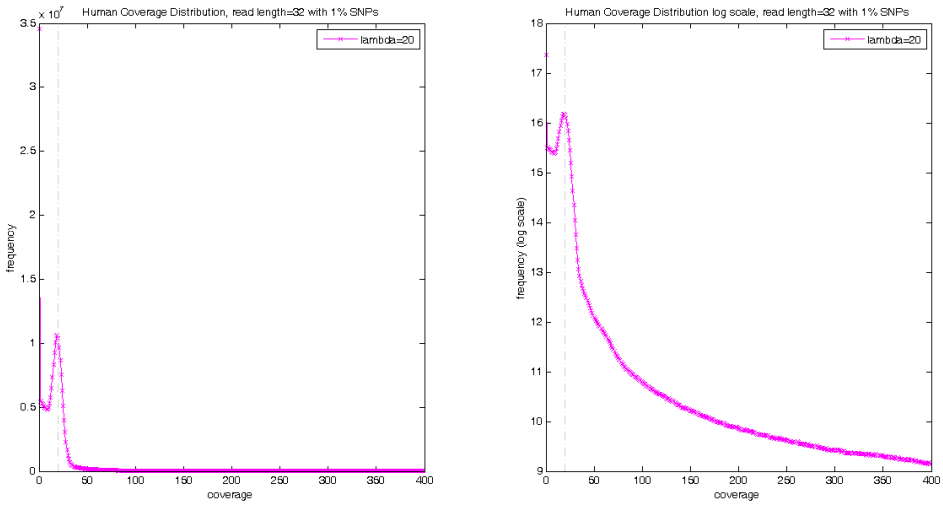


Figure 3.19: Coverage distribution of human chromosome 1 reads with **1% SNPs** selected randomly for a given lambda (**20**) and read length (32). The observed coverage distribution fits with a Poisson distribution (**λ=20**) with ~99.26% accuracy

---

The coverage data obtained after sequence alignment are a non-negative count data extending across the length of a genome. Note that percent of 0 coverage values is significantly higher than the expected probability in previous *E. coli* and human experiments. As one can see, the presence of repeated sequences leads to higher coverage values. To summarize, the coverage data is over-dispersed along the boundaries of the distribution. Within a bounded region the data still follow a Poisson distribution and are suitable for the application of a Poisson-based model.

Based on the observations in *E.coli* and human genome coverage distributions, a statistical model was proposed which estimates the parameter of the Poisson distribution

(λ) and parameter alpha (α) using 2D global optimization approach, where λ is the expected distribution mean and (1-α) represents the percentage of bias. The model parameters α and λ are optimized within a closed interval of the coverage data distribution using golden-ratio algorithm. The value of alpha is first optimized within range 0 to 1 while minimizing the error calculated using total sum of squares (TSS) statistic. The optimized value of alpha is then utilized to determine a value of lambda (or the distribution mean of the Poisson process) that fits better to the coverage data.

The model was tested on the simulated data (Appendix I) and genomic coverage data with different coverage depths and predicts the Poisson parameter with reasonable accuracy.

The proposed approach helps in eliminating bias introduced by the coverage from the repeatable regions and perform reasonable estimate of expected coverage value in a given genomic region.

Poisson distribution

The Poisson process model belongs to a family of generalized linear models (GLMs) and applies to a wide variety of problems in different fields of science. The model defines the probability of occurrence of an event in temporal and spatial settings. The standard Poisson distribution provides a discrete probability distribution of the frequency of events occurring randomly in a given interval of time or space. The following formula computes probability of x events taking place in a given time when λ represents average number of events per interval.

$$P(X = x) = \frac{e^{-\lambda}\lambda^{x}}{x!}$$

where, $x = 0, 1, 2, 3, 4, \ldots$

λ rate parameter and the mean of the distribution;

X the number of events in a given interval of time or space;

e a mathematical constant, base of the natural logarithm, e = 2.718282.

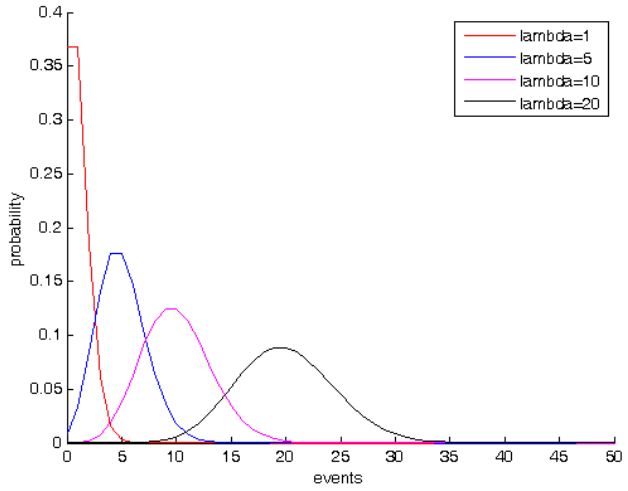


Figure 3.20: Standard Poisson probability distributions for lambda =1, 5, 10, 20

3.2.3 Optimization Method

Golden-Ratio Algorithm

The golden-ratio method [7] is used to find minimum or maximum of a function $f(x)$ given that the function is unimodal i.e. contains only one minimum or maximum in the interval [l,u]. The golden-ratio algorithm is our method of choice to optimize model

parameter (alpha) and Poisson parameter lambda in the one-dimensional unimodal function [7].

Binary Search Method

In the *binary search method*, the search interval is divided into 2 equal parts by calculating the midpoint $m = (l + u)/2$. Select two points x_1 and x_2, such that $f(x_1) \neq f(x_2)$ for a small value of epsilon (ε). If $f(x_1) < f(x_2)$, then $[l; x_1]$ is our new search interval, otherwise search is continued in the new interval $[x_2; u]$. The procedure of finding new upper and lower bounds is repeated until the desired region is found.

The golden-ratio algorithm finds a subinterval within the initial interval where a minimum/ maximum value can be found. The intermediate bounds are selected in a different way than the *binary search* method. The values of the two intermediate points x_1 and x_2 are selected such that:

$$\frac{a}{(a+b)} = \frac{b}{a} = 0.618\ (golden\ ratio)$$

$$\frac{b}{a} = \frac{(a-b)}{b} = 0.618\ (golden\ ratio)$$

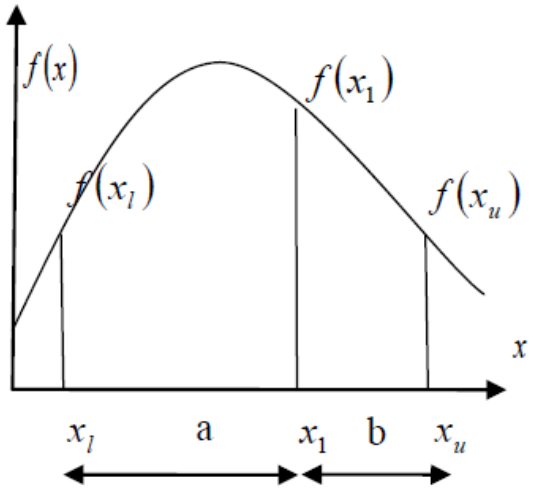


Figure 3.21 a: Finding first intermediate point x1

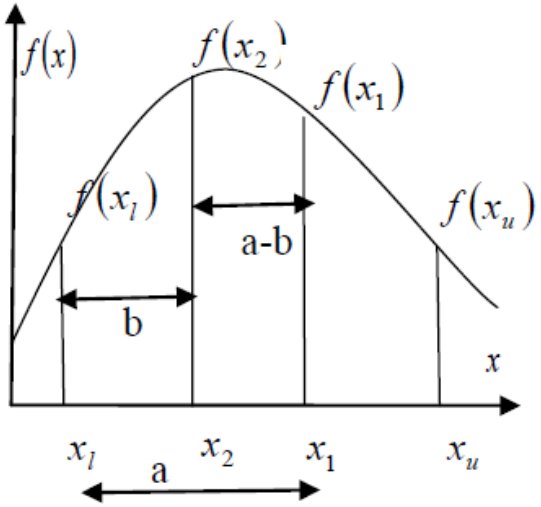


Figure 3.21 b: Finding second intermediate point x2

Figures 3.21 a. and b. illustrate the selection of the intermediate points using golden-ratio method on a unimodal function with single maximum value.

3.2.4 Algorithms to fit model into coverage data

The proposed model performs multi-level optimizations, and the golden-ratio method efficiently reduces the search area to select an optimal alpha and lambda value [61]. The pseudo code and algorithms of the proposed model are discussed in detail. There are mainly two functions

find_Best_Fit_Using_Golden_Ratio_Non_Cumulative_2D ( ) and

calculate_Error_Non_Cumulative_2D ( ).

The first function, calculate_Error_Non_Cumulative_2D ( ), is iteratively executed and alpha (or percentage of points in Poisson distribution) is optimized within the limits stated and later lambda is optimized in the function find_Best_Fit_Using_Golden_Ratio_Non_Cumulative_2D ( ) by minimizing sum of squares error.

Pseudocode find_Best_Fit_Using_Golden_Ratio_Non_Cumulative_2D ()

1. Initialize *coverage array, minimum_lambda, maximum_lambda*
2. Compute frequency array for the given coverage array
3. Find maximum coverage value in coverage array and assign it to *maximum_coverage_value*
4. *minimum_coverage_value* = 1;
5. Find region where coverage array follows a Poisson distribution
6. **for** *i = minimum_coverage_value to maximum_coverage_value* step 1 **do**
    a. **for** *maximum_number_of_iterations*

b. Optimize *alpha* and *lambda* within minimum and maximum lambda values using golden-section search for given interval (*xmin, xmax*) by minimizing error *E (λ)*.

To apply our model to the coverage values, the following functions calculate_Initial_Values() and get_Probability_Poisson_Distribution( ) were designed which assist in the calculation of Poisson probability for given lambda and *x* number of events. The Poisson formula can be re-written as

$$\begin{aligned} &= e(log\left(e^{-\lambda}\right) + x(log\lambda) - log(x!)) \\ &= e(-\lambda + x(log\lambda) - log(x!)) \\ &= e(x(log\lambda) - \lambda - log(x!)) \end{aligned}$$

log(*x!*) in above formula can be computed using the following formula:

$$\begin{aligned} log(x!) &= log(x * (x-1) * (x-2) * \ldots * 1) \\ &= log(x) + log(x-1) + log(x-2) + \cdots + log(1) \end{aligned}$$

Algorithm CALCULATE_INITIAL_FACTORIAL_VALUES ( )

Input: N/A

Output: log_factorial

1. log_factorial[0] ← 0.
2. **for** i←1to 5000 do
3. log_factorial[i] ←log_factorial[i-1]+log((double)i)
4. **end for**
5. **return**

Algorithm GET_PROBABILITY_POISSON_DISTRIBUTION ( )

Input: _lambda, k

Output: Poisson probability value for given lambda and k events

1. calculate_Initial_Factorial_Values ( )
2. **return** exp(_k * log(_lambda)-_lambda - log_factorial[_k]

Algorithm CALCULATE_ERROR_NON_CUMULATIVE_2D ( )
Input: _lambda

Output: _alpha

```
1.      max_number_of_iterations ← 32
2.      total_number_of_values ← 0
3.      for i←0 to frequency_array_size do
4.              total_number_of_values ← total_number_of_values + frequency[i]
5.      end for

6.      c ← 0.5*(sqrt(5.0)-1.0);  //0.61803398874989546837927170829958 3
7.      a ← 0.0
8.      b ← 1.0
9.      x1 ← b - c*(b-a)
10.     x2 ← a + c*(b-a)

11.     sum_error ← 0.0
12.     for i←min_frequency_considered_for_Poisson_distribution to
13.       max_frequency_considered_for_Poisson_distribution do
14.             tmp_error ← ((double)frequency[i])/ (total_number_of_values)-
                        (get_Probability_Poisson_Distribution(_lambda,i)*x1)
15.             sum_error ← sum_error +(tmp_error*tmp_error)
16.     end for

17.     f_1 ← sum_error
18.     sum_error ← 0.0
19.     for i←min_frequency_considered_for_Poisson_distribution to
20.      max_frequency_considered_for_Poisson_distribution do
21.             tmp_error ← ((double)frequency[i])/(total_number_of_values) –
                        (get_Probability_Poisson_Distribution(_lambda,i,out)*x2)
22.             sum_error ← sum_error +(tmp_error*tmp_error)
23.     end for

24.     f_2 ← sum_error

25.     for j←0 to max_number_of_iterations do
26.             if f_1 < f_2
27.                     b ← x2
28.                     x2 ← x1
29.                     x1 ←  b - c*(b-a)
30.                     f_2 ← f_1
```

```
31.                     sum_error ← 0.0

32.             for i←min_frequency_considered_for_Poisson_distribution  to
                        max_frequency_considered_for_Poisson_distribution do
33.                     tmp_error ←
                            ((double) frequency[i])/ (total_number_of_values) –
                            (get_Probability_Poisson_Distribution(_lambda,i)*x1)
34.                     sum_error ← sum_error + (tmp_error*tmp_error)
35.             end for

36.             f_1 ← sum_error
37.             else
38.             a ← x1
39.             x1 ← x2
40.             x2 ← a + c*(b-a)
41.             f_1 ← f_2
42.             sum_error ← 0.0

43.             for i←min_frequency_considered_for_Poisson_distribution to
                        max_frequency_considered_for_Poisson_distribution do
44.                     tmp_error ←
                        ((double) frequency[i])/ (total_number_of_values) –
                        (get_Probability_Poisson_Distribution (_lambda,i)*x2)
45.                     sum_error ← sum_error + (tmp_error*tmp_error)
46.                     end for
47.                     f_2 ← sum_error
48.             end if
49.       end for

50.     _alpha ← 0.5*(a+b)
51.     sum_error ← 0.0

52.     for i←min_frequency_considered_for_Poisson_distribution
        to max_frequency_considered_for_Poisson_distribution do

53.             tmp_error ← ((double)frequency[i]/(total_number_of_values)) –
                        (get_Probability_Poisson_Distribution(_lambda,i)*_alpha)
54.             sum_error +← (tmp_error*tmp_error)
55.     end for

56.     return (sum_error/(max_frequency_considered_for_Poisson_distribution-
        min_frequency_considered_for_Poisson_distribution + 1))
```

Algorithm FIND_BEST_FIT_USING_GOLDEN_RATIO_NON_CUMULATIVE_2D ()

Input: min_lambda_value
max_lambda_value

Output: lambda
alpha
error

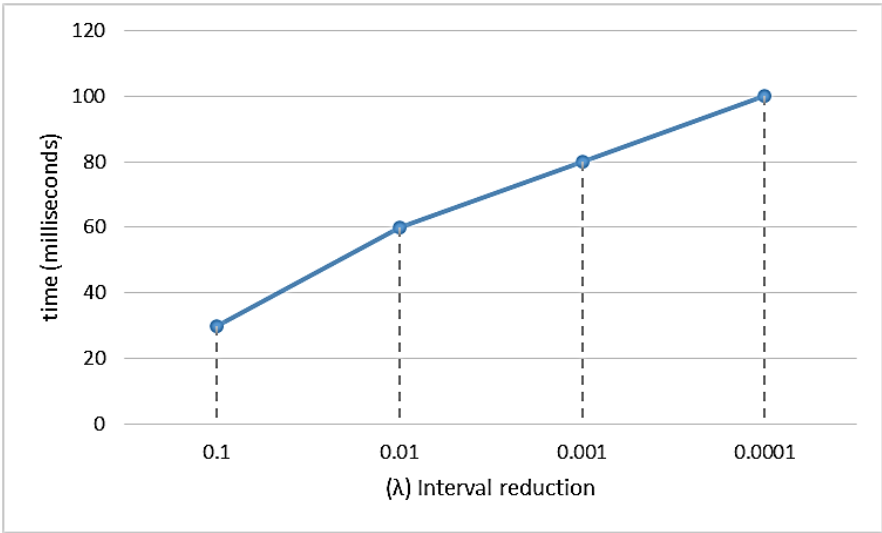
```
1.      c ← 0.5*(sqrt(5.0)-1.0) //Golden ratio
2.      max_number_of_iterations ← 32
3.       a ← min_lambda_value
4.      b ← max_lambda_value
5.      x1 ← b - c*(b-a)
6.      x2 ← a + c*(b-a)
7.      f_1 ← calculate_Error_Non_Cumulative_2D(x1,_alpha)
8.      f_2 ← calculate_Error_Non_Cumulative_2D(x2,_alpha)
9.      _lambda ← 0.0
10.     error ← 0.0
11.     for i←0 to max_number_of_iterations do
12.             if f_1 < f_2 then
13.                     b ← x2
14.                     x2 ← x1
15.                     x1 ←  b - c*(b-a)
16.                     f_2 ← f_1
17.                     f_1 ← calculate_Error_Non_Cumulative_2D(x1,_alpha)
18.             else
19.                     a ← x1
20.                     x1 ← x2
21.                     x2 ← a + c*(b-a)
22.                     f_1 ← f_2
23.                     f_2 ← calculate_Error_Non_Cumulative_2D(x2,_alpha)
24.             end if
25.     end for
26.     _lambda ← 0.5*(a+b)
        //alpha finalized here
27.     error ← calculate_Error_Non_Cumulative_2D(_lambda,_alpha)
28.     return
```

In summary, the proposed model fits the reads coverage more appropriately than the traditional Poisson model.

Algorithm Evaluation

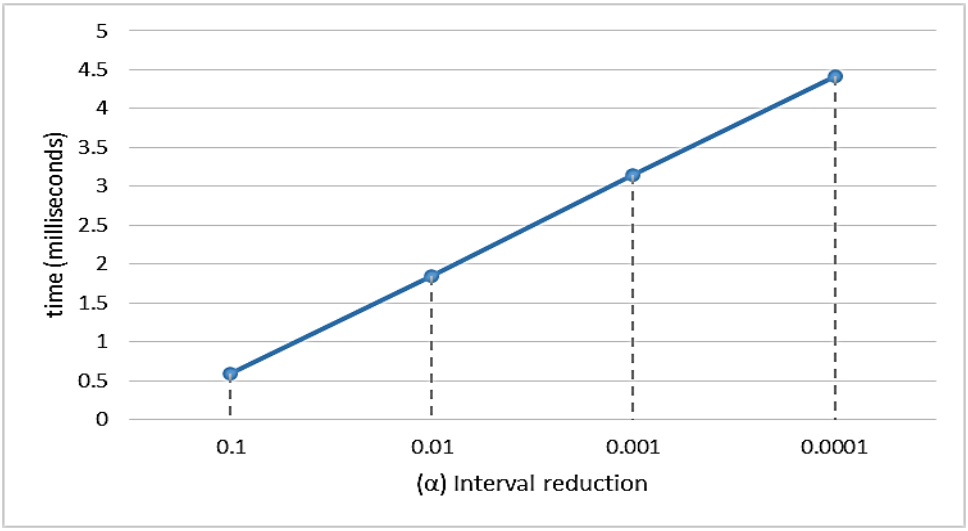
To evaluate the optimization algorithm, *E. coli* bacterial genome (~5Mb) was used and 5X coverage was generated (a total of 27,642,225 reads) with a read length of 20. The experiments were executed on Red Hat Enterprise Linux Server release 6.5 (Santiago) x86_64.

The optimization methods logarithmically reduce the search area and with two-level optimizations, one for lambda and one for alpha; the time complexity of the algorithm is *O (log (n) (log (m)));* where m and n are the length of the interval for parameters lambda (λ) and alpha (α) respectively.



$$\triangle \lambda = (\lambda_{max} - \lambda_{min}) / N$$

Figure 3.23: Interval reduction rate (λ) versus time



$$\triangle \alpha = (\alpha_{max} - \alpha_{min}) / N$$

Figure 3.23: Interval reduction rate (α) versus time

The time complexity of optimization algorithm is given as *O (log (n) (log (m)))*. Since no auxiliary memory was used in the algorithm, the Space complexity is *O (1)*.

3.3 Verification and Validation

3.3.1 Simulated data

The proposed Poisson-based model was first evaluated on simulated data using several experiments. The mapping process was simulated to generate coverage data. First a positive integer array was generated with values that belong to a Poisson distribution at a specific mean (λ). At this point, the coverage follows a standard Poisson distribution whose mean is equal to variance and all other properties of Poisson distribution also hold true. Next several perturbations (zeros and high values) were artificially added to the original array to distort the distribution resulting in more than expected zero and high coverage values.

Mersenne Twister, a random number generator implemented in C++, was used to randomly generate numbers to add outlying coverage values to the initial array. The number of zeros and high values were added in varying percentages and the model was tested for a good estimate of the initial mean of the distribution. The results with summary of the input parameters and output results are present in Appendix I. Following is the pseudo-code describing a step wise procedure followed to model the genomic coverage data.

Pseudo-code for simulation experiments:

1. Input genome_length, read_length, lambda, percent_of_0s, percent_of_high_values, minimum_lambda, maximum_lambda,

2. For the given lambda and genome_length generate an unsigned integer array of Poisson distribution
3. Randomly insert x number of zeros (representing SNPs and Ns) and y number of higher values using values given in percent_of_0s, percent_of_high_values
4. minimum_frequency=0, maximum_frequency=1000
5. Calculate frequency array of the new coverage array
6. Apply Poisson mixture model to estimate lambda value of the coverage array

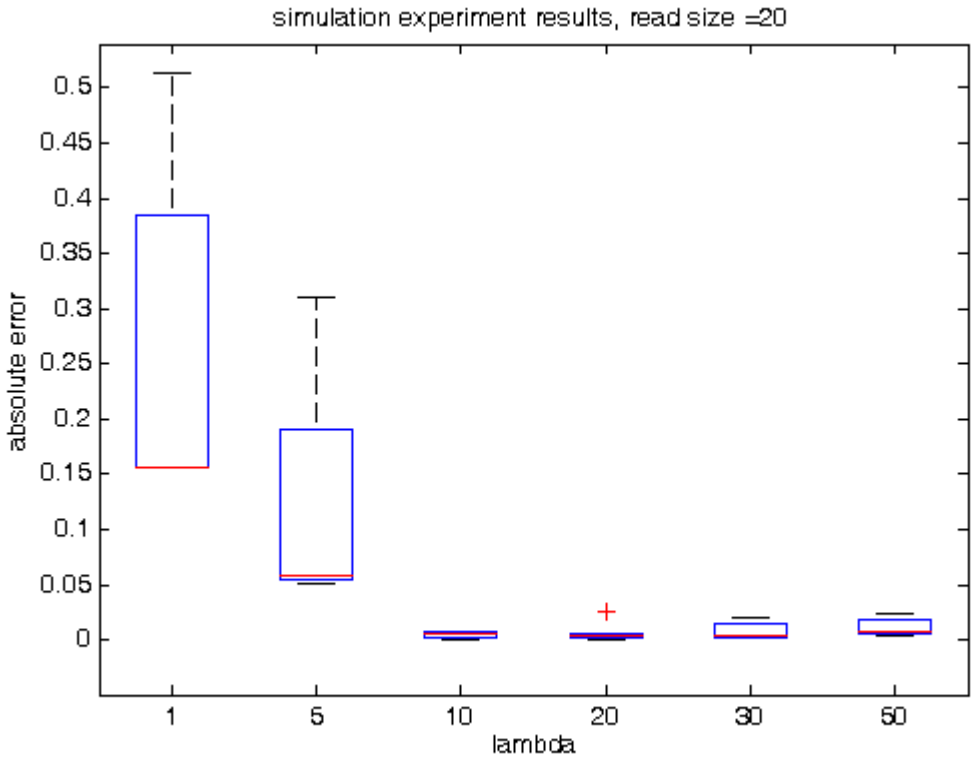


Figure 3.24: Boxplot of Absolute Error Values for each group of experiments

(Appendix I)

3.3.2 *Genome coverage data*

Results from simulation experiments presented good accuracy for synthetic data. Hence, the Poisson mixture model was applied to real coverage data from microbial genomes like *E. coli* bacteria and human genome. *E. coli* has of length ~5Mb and was used as a model organism in many biological experiments. Figure 3.25 below shows the CNV detection workflow.

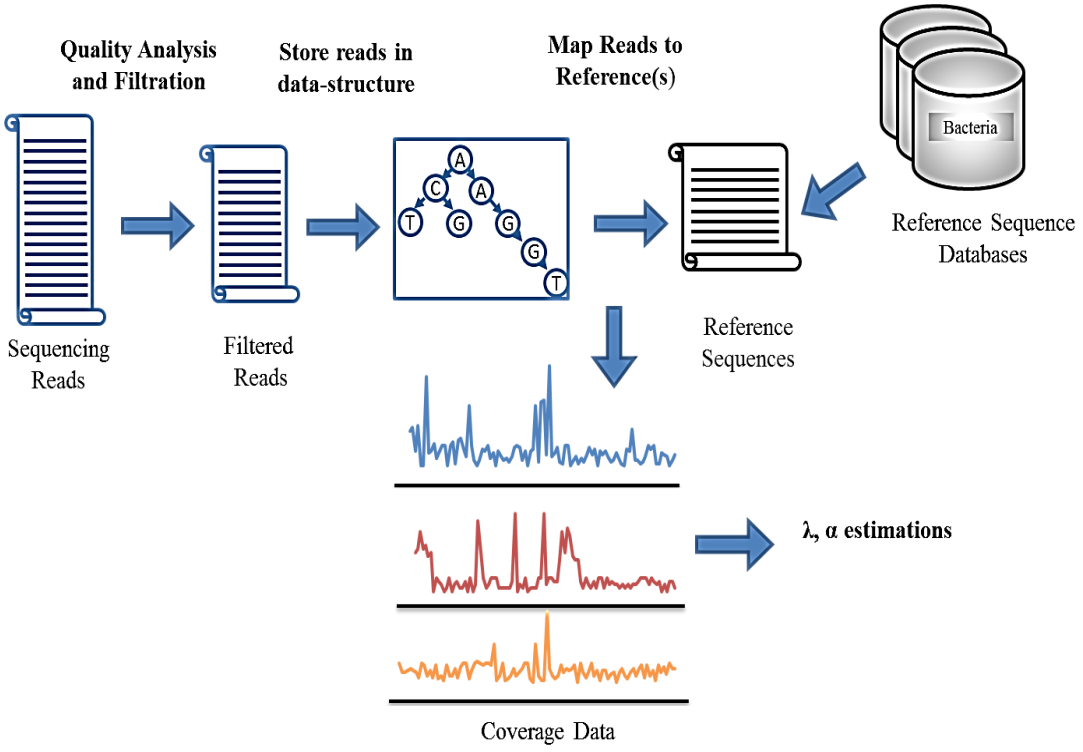


Figure 3.25: CNV detection workflow

E. coli bacterial genome has a genome length of 5,528,445 base pairs. Reads of length 20 were taken randomly for different coverage values and mapped back to genome and then Poisson-based model was applied on the coverage data to estimate the true coverage values.

Table 3.1: Average coverage estimations from the Poisson-based model.

Total reads	**Expected coverage (λ)**	**Estimated coverage (λ')**	**Estimated alpha (α)**	**Average**
27,642,225	5	5.085	0.935	5.72
55,284,450	10	10.073	0.921	11.40
110,568,900	20	20.026	0.913	22.85

Copy Number distributions were passed to the model. The table above (Table 3.1) lists the expected average coverage (**λ)** and predicted values of average coverage (λ') and proportion of Poisson distribution (α).

Chapter 4

APPLICATIONS

Meta-genomics is field of genomics that involves studying environmental samples. The metagenomic samples like water and soil samples, contain many co-existing microbial communities, and it can be a challenge to detect a microorganism or pathogen in the presence of many background species. Figure 4.1 below illustrates one problem in particular. Due to shared sequences like the 16S RNAs between organisms, the coverage spikes are even bigger than before.

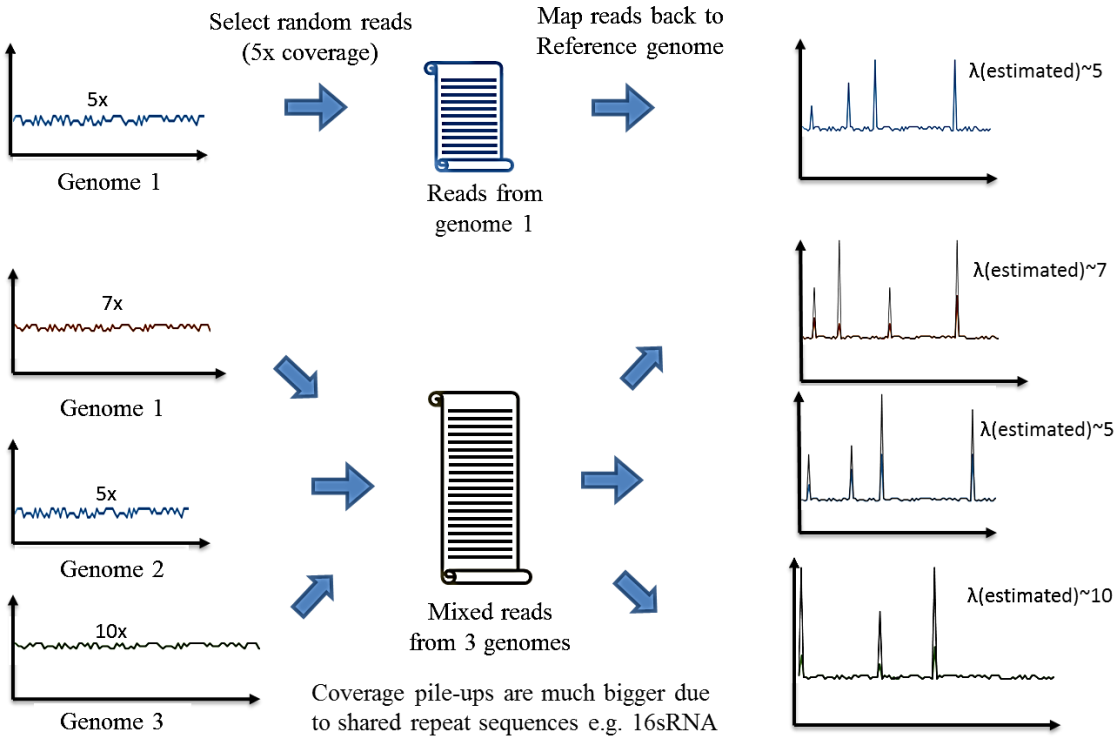


Figure 4.1: Meta-genomic samples: coverage pile-ups

4.1 Relative Abundance of Bacteria

To test the proposed approach on metagenomic sample, four genomes *E. coli*, *Haemophilus influenza*, *Legionella pneumophila*, and *Shigella flexneri* were chosen to generate a combined pool of reads by selecting reads corresponding to coverage of 8, 10, 15, and 20 respectively from these genomes. Next, all the reads were mapped to each of the genome. Coverage values were obtained across each genome and each of these arrays was passed to the optimization algorithm. Again, satisfactory results were obtained with a mix of microbial genomes.

Table 4.1: List of organisms used in the mixture with details

organism name	**sequence length**	**coverage expected**
Escherichia coli	5,528,445	8
Haemophilus influenzae	2,007,018	10
Legionella pneumophila	3,516,334	15
Shigella flexneri	4,650,856	20

Table 4.2: Coverage estimation after applying Poisson-based model on the bacterial mixture

Organism name	**Expected coverage (λ)**	**Estimated coverage (λ')**	**Estimated alpha (α)**	**average**
E. coli	8	8.130	0.584	17.01
H. influenzae	10	10.036	0.963	11.2
L. pneumophila	15	15.006	0.988	15.6
S. flexneri	20	22.962	0.782	50.17

The proposed model can be used for the estimation of relative abundance of microbial organisms in meta-genomic samples.

4.2 Pathogen Detection

The microbial samples are often found mixed with human genomic fragments. For example, the micro-biome in a natural water body would also contain human DNA elements. Therefore, in our next experiments a mixed sample of bacterial (*E. coli*, *Serratia liquefaciens)* and human reads was prepared (Table 4.3).

To generate a mixed sample, reads of size 32 were taken from *E. coli* and *S. liquefaciens* such that the total reads taken from each genome are 8% and 2% of the total sample respectively. Next the bacterial reads were mixed with reads obtained from human genome which comprised of 90% of the sample. The model was applied to this mixed sample and it was seen that the predicted coverage values are close to expected values. Table 4.4 summarizes the results.

Table 4.3: List of organisms used in the mixture with details

Organism name	**Sequence length**	**Percent of reads mixed**	**Number of reads**	**Coverage expected**
Human chr 1	248,956,422	90	83,315,216	10.7
Escherichia coli	4,835,601	8	8,000,000	52.94
Serratia liquefaciens	5,238,612	2	2,000,000	12.22

Table 4.4: Coverage estimation after applying Poisson-based model on the mixture

Organism name	Expected coverage (λ)	Estimated coverage (λ')	Estimated alpha (α)	average
Human chr 1	10.7	11.701	0.785	220.01
E. coli	52.94	52.983	0.97	59.79
S. liquefaciens	12.22	12	0.984	12.41

The proposed model can be used for pathogen detection.

NGS technologies are used to detect pathogen agent(s) in host organisms like human. The metagenomic sample taken from the host usually contains a lot of background noise for detecting the pathogen. The experiment demonstrates that the proposed approach can be used for pathogen detection by preventing/ minimizing the interference from the background/ host genomic material.

Discussion

In the future, the developed approach can be applied to prior projects which used average, exclusion of repeats, or exclusion of top or bottom x% of coverage values.

Example I: Pathogen Detection in Cancer Cell Lines

In this project a set of cancer samples were checked for *P*. acnes genes, which may play a role in inflammation of the prostate leading to cancer [12].

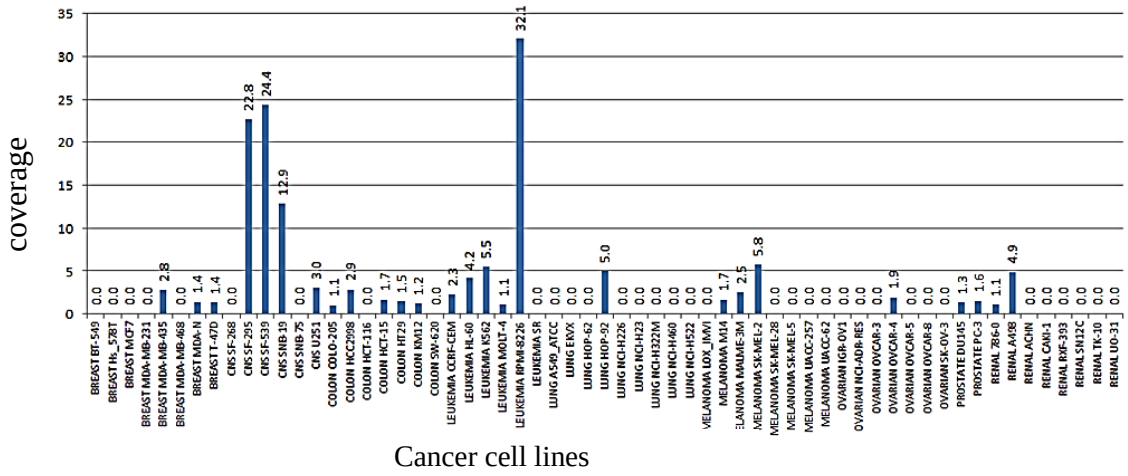


Figure 5.1: The presence of *P. acnes* HL096PA1 (NC_021085.1) genes identified across all 61 NCI-60 datasets.

The study was performed using average statistic to represent the coverage values over smaller fixed size windows.

Example II: Detecting Differentially Methylated Regions in Acute Lymphoid Leukemia (ALL) Cell Lines

Glucocorticoid-induced apoptosis (GCIA)

One of the characteristics of cancer cells is the loss of capability of programmed cell death or apoptosis. Other abnormalities include genetic (mutations and genomic re-arrangements) and/ or epigenetic changes, like abnormal DNA methylation patterns. In this research, the human childhood leukemia (CEM) cell lines were studied. Founding clones CEM-C1 and CEM-C7 which are resistive and sensitive to glucocorticoid-induced apoptosis (GCIA) respectively were cloned and studied. Glucocorticoids (GCs) are a kind of steroids which inhibit the proliferation of lymphoid cells by causing their apoptotic death. For this reason GCs find application in cancer treatments. Dexamethasone (Dex) is one of such synthetic steroid often used for therapeutic purposes [46].

The effect of these drugs has been established a long time ago. According to one of the theories, the direct interaction of GC with its receptor glucocorticoid receptor (GR) can lead to death of lymphoid cells. The GCs bind with the GR to up-regulate expression of anti-inflammatory proteins. However, the complete mechanism is still unknown and there exist different possibilities of GC induced cell death to leukemic cells.

Additionally, DNA of leukemic patients have shown aberrant methylation patterns in previous studies [34] [44] [53] [58] [64] [70]. Different DNA sequencing technologies have been increasingly employed to study methylation profiles [36].

The objective of the study was to check if DNA methylation accounts for the resistance of C1-15 cells to dexamethasone-driven apoptosis. The methylation states of DNA from 3 different cell lines were analyzed.

Human leukemic cell clone CEM C1-15 is **resistant** to GCIA. After treatment with 5-deoxy azacytidine (Dex), an agent that causes DNA demethylation, (also describe how dex works) these cells become sensitive to GCIA, similar to their **sensitive** sister clone, CEM C7-14. From the treated C1-15 cells, a stably **sensitive** clone, CEM C1-15 B 9 IV (B 9 IV) was isolated. Methylation states of 3 DNA sources: CEM C1-15, CEM C7-14 and B 9 IV cells were studied. Using Methylated-DNA IP Kit, DNA was separated into the methylated-depleted and methylated-enriched DNA fractions, based on immunoreactions to methylated DNA.

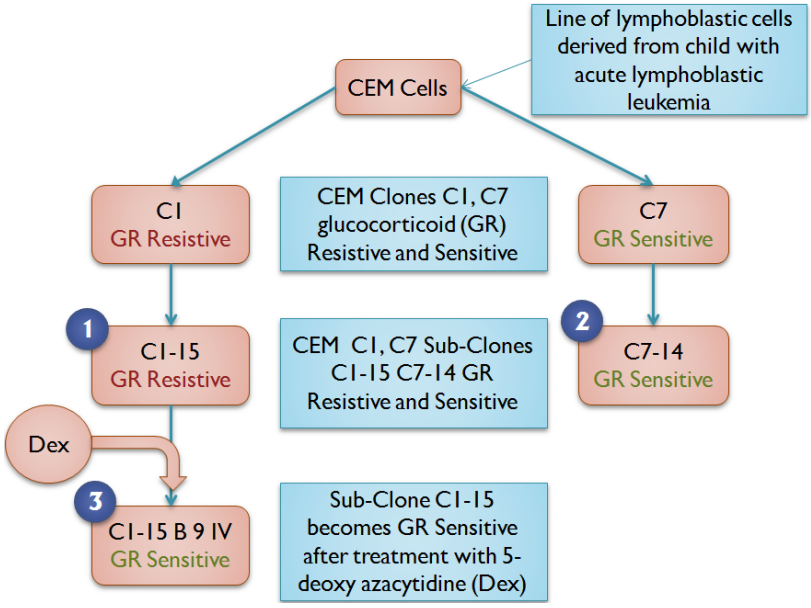


Figure 5.2: Leukemia cell lines used in study

Quality Analysis and Filtration

Methylated and non-methylated fractions from 3 samples; C1-15, C7-14, and BIV9 cells were sequenced. Filtration of sequenced reads was the primary step performed post sequencing. Quality assessment of the original reads procured from these samples helped us to decide filtration parameters. The quality charts can be found at Appendix. The number of reads from first run was not enough to complete the analysis. The sequencing was hence repeated for all the 6 samples. A brief description of filtration procedure and statistics for both the experiments is as follows: Reads obtained from first run were 36 bases long. First 5 nucleotides were trimmed from all the reads and reads containing low quality (<10) nucleotides were filtered out. Also, reads with 90 percent of same nucleotides (low complexity reads) were discarded.

Table 5.1: Read statistics after sequencing run 1

Sample Name	Total Reads Before Filtration	Total Reads After Filtration	Unique Reads After Filtration
METH_C1_15_Dex-R_Methylated_DNA	4,479,400	1,393,965	650,427
METH_C1_15_Dex-R_Non_Methylated_DNA	3,805,173	1,772,583	241,403
METH_C1_15_SubClone_B9_IV_Dex-S_Methylated_DNA	23,436,215	19,006,856	14,097,573
METH_C1_15_SubClone_B9_IV_Dex-S_Non_Methylated_DNA	16,252,409	12,979,598	11,984,083
METH_C7_14_Dex-S_Methylated_DNA	11,372,868	6,837,813	5,000,675
METH_C7_14_Dex-S_Non_Methylated_DNA	7,187,445	3,776,595	2,676,470

Reads obtained from second run were 40 bases long. The first 3 and last 6 nucleotides were trimmed resulting in reads of length 31 bases. Reads containing template "CGGAAGAGCACACGTCTGAACTCCAGTCACA" were excluded from next phase of analysis. The remaining parameters were kept the same. Finally, reads from run 1 and 2 were combined (Figure 4.3).

Table 5.2: Read statistics after sequencing run 2

Sample Name	Total Reads Before Filtration	Total Reads After Filtration	Unique Reads After Filtration
METH_C1_15_Dex-R_Methylated_DNA	30,793,945	13,197,866	8,982,651
METH_C1_15_Dex-R_Non_Methylated_DNA	32,379,143	13,968,453	13,043,892
METH_C1_15_SubClone_B9_IV_Dex-S_Methylated_DNA	28,853,504	12,299,227	10,042,092
METH_C1_15_SubClone_B9_IV_Dex-S_Non_Methylated_DNA	6,017,727	1,349,305	791,494
METH_C7_14_Dex-S_Methylated_DNA	24,250,777	7,929,301	7,325,859
METH_C7_14_Dex-S_Non_Methylated_DNA	14,724,414	4,520,012	3,286,762

Table 5.3: Total reads after combining reads from sequencing run 1 and 2

Input file/ Sample name	Combined Total Reads
METH2_C1_15_Dex-R_Methylated_DNA	14,591,831
METH2_C1_15_Dex-R_Non_Methylated_DNA	15,741,036
METH2_C1_15_SubClone_B9_IV_Dex-S_Methylated_DNA	31,306,083
METH2_C1_15_SubClone_B9_IV_Dex-S_Non_Methylated_DNA	14,328,903
METH2_C7_14_Dex-S_Methylated_DNA	14,767,114
METH2_C7_14_Dex-S_Non_Methylated_DNA	8,296,607

Mapping and Coverage Analysis

Reads combined from 6 samples were mapped to human genome build 37.3 which consists of chromosomes 1-22,, X and Y. For every sample, position-by-position coverages were obtained for each chromosome. To facilitate analysis and visualization of each chromosome, the values were binned and averaged into fixed size windows.

Reads containing ambiguous character(s) 'N' were ignored and not considered while mapping. All the 6 samples were mapped to 22 chromosomes and X, Y chromosomes from human build 37.3 (hg19). For every sample, position-by-position coverage values were obtained for each chromosome. To facilitate the analysis and visualization of the coverage for each chromosome studied, the values were binned and averaged into non-overlapping window of length 5K. For each window the average coverage was calculated using the following formula.

$$Avg.Cov. = \frac{\text{Read\_Size}}{\text{Window\_Size}} * \sum_{k=1}^{\text{window\_size}} (coverage\_value_k)$$

where,

k=location in a window,

Read_Size=31 and Window_Size=5000 for this particular study

The highly repetitive nature of the human genome is indicative of the high coverage values observed which could potentially compromise the data analysis and results. As discussed in section 3.3.1, the high coverage values were normalized by first creating a map of 31-mers uniquely present in human genome.

To proceed with exclusive analysis of repetitive and unique regions, repetitive *31-mers* were identified and as a mask. Coverage values were normalized for repeated and unique sites in a window and a separate analysis was performed. This prevented the higher coverage at repeats to over-shadow the methylation differences at locations covered uniquely. Normalized coverage values were studied to identify hypo- and hyper-methylated regions.

Finding DMRs:

Identification of DMRs from all 24 chromosomes was performed twice: once with masked unique regions and another time with masked repeats. For each chromosome normalized window-by-window coverage values were read and all zero coverage values were replaced with small non-zero number, say alpha=0.0001. Next the values were

converted to percent coverages by dividing each value by total coverage in the complete chromosome. For each set of methylated (Me) and non-methylated (NM) sample from the particular cell line MA plot was drawn. MA values for coverage corresponding to each window were calculated using following formula:

$$M = \log(Me/NM)$$

$$A = \frac{1}{2}\log(Me * NM)$$

where,

Me is the coverage point in methylated sample of a particular cell line

NM is the coverage point in non-methylated sample of particular cell line

M is difference/fold change in coverage values and

A is the geometric addition

Z-score of the normalized coverage values were computed to detect regions with considerable fold change (2x) using the formula:

$$Z = (xi - mean)/Std\_Dev$$

A cut-off Z-Score value of 2 was used to select hyper-methylated (Z-score>2) and hypo-methylated regions (Z-score<-2). Figure 5.3 and 5.4 show the MA plots drawn for repeatable and unique regions in chromosome 4 respectively.

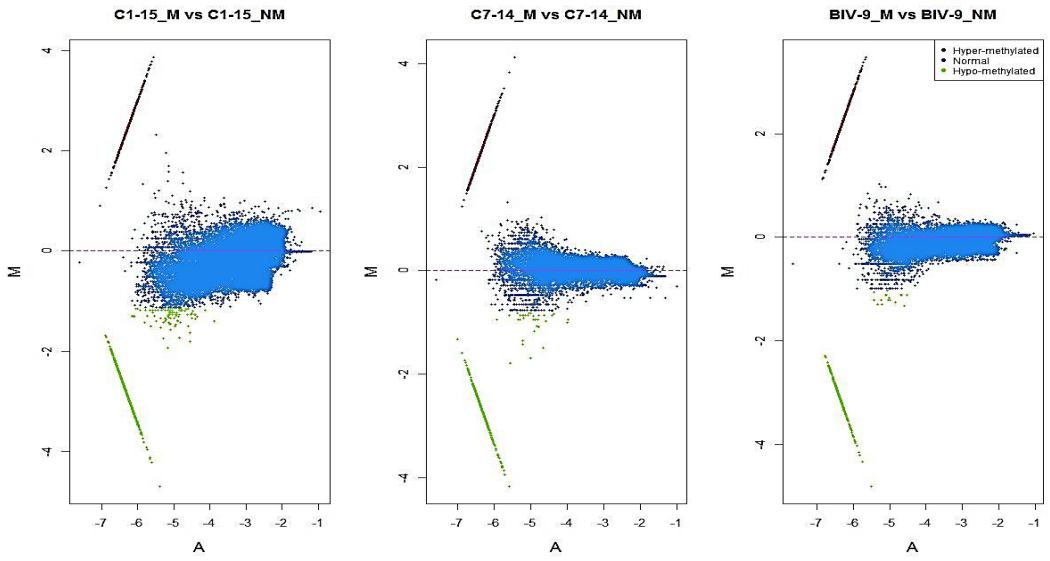


Figure 5.3: MA plots for methylated and non-methylated normalized coverages for repeatable sites in chromosome 4

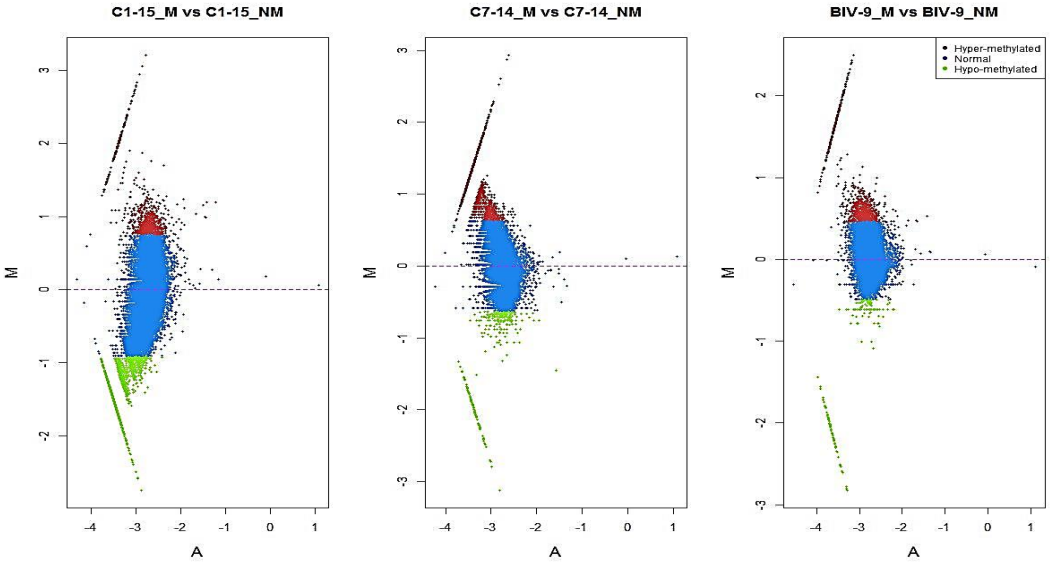


Figure 5.4: MA plots for methylated and non-methylated normalized coverages for unique sites in chromosome 4

Results and Discussion

Using proposed techniques 6 ALL samples were analyzed by computing their methylation status across the complete human genome (Build 37.3) and identified regions that were hyper- and hypo-methylated in each sample. Normalization of coverage values eliminated the ambiguity arising due to the presence of highly repeatable regions in the human genome.

At least 46 genes were found consisting of regions that are hyper-methylated in dex-resistive cells (C1-15) and hypo-methylated in dex-sensitive cells (C7-17 and IV-B9) and that could be involved in an epigenetic pathway(s) leading to apoptosis within the leukemic cells. These DMRs belonged to genes and non-coding regions. The shift in the DNA Methylation profile occurs when CEM cell lines change their sensitivity to GCIA.

A minimum of 46 regions of size 5K were found [60] corresponding to human genes which were hyper-methylated in the dex-resistive (C1-15) sample and hypo-methylated in dex-sensitive samples (C7-14 and IV-B9). The 46 genes are listed below: CAMTA1,HNRNPR,ADCK3,SIPA1L2,FMN2,CAPN13,TMEM178A,ALS2,MPP4,DIS 3L2, SYNPO2,IQGAP2,CYFIP2, CAP2,HLA-F-AS1, SDK1, CRHR2,LOC100132891, RIPK2,TRAPPC9,PTP4A3,FLJ43860,PTP4A3,ADARB2,B4GALNT4,SOX6,PDZRN4, DNAJC14,ORMDL2,SARNP,SLC25A21,ATL1,FOXL1,GALNS,CBX4,MAPK4,CYP4 F24P,LOC44051, ZNF831,KCNQ2,KCNJ6,COL6A1,TENM1,FUNDC2, FAM72B, FAM86B1, MRC1, FAM25B,FAM25C,FAM25G.

Some of the differentially methylated genes identified in the study like FMN2, PDZRN4, SLC25A21 and MAPK4 are already associated with leukemia or cancer. The results show that these genes were hypo-methylated in dex-sensitive samples which respond to GC-evoked apoptosis. This indicates that these genes are involved in an epigenetic pathway leading to apoptosis within the leukemic cells. In summary, the complete characterization of Methylation patterns of leukemia cancer cell lines opens doorway to further analysis of genes and molecular pathways affected in this cancer.

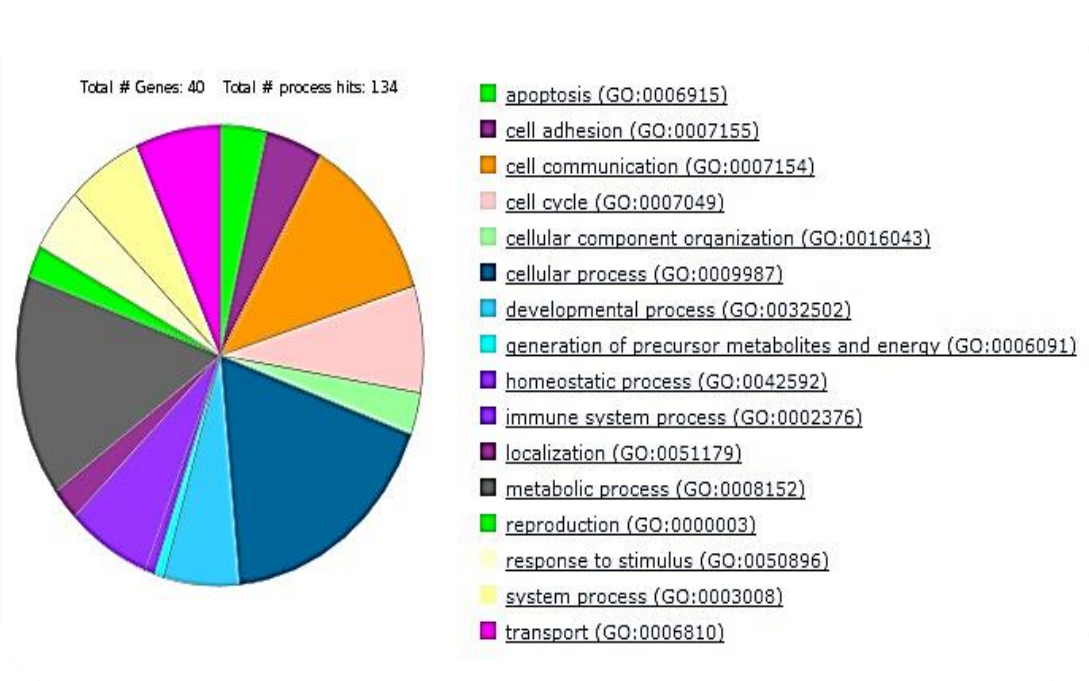


Figure 5.5: Distribution of differentially methylated genes based on biological processes [45]

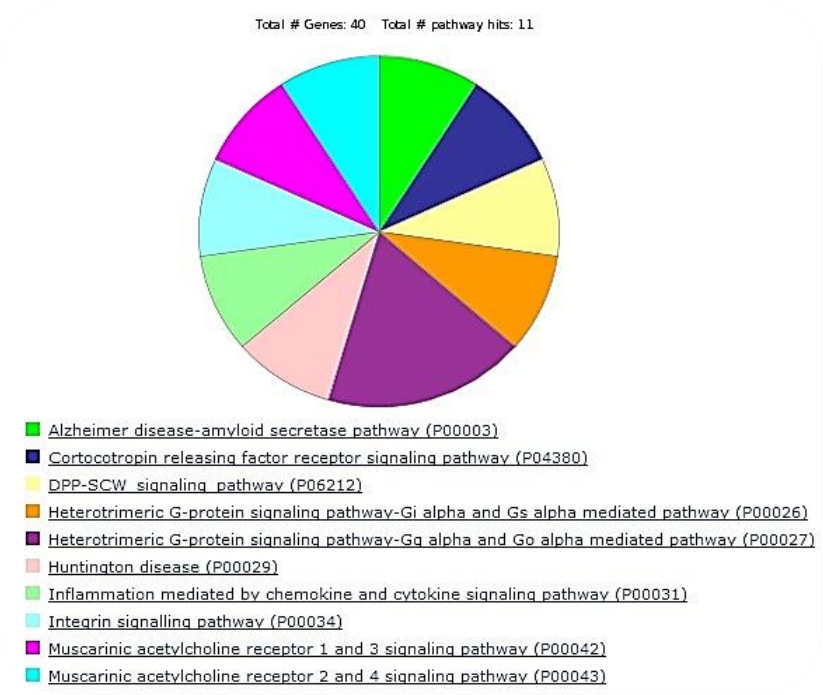


Figure 5.6: Distribution of differentially methylated genes based on pathways [45]

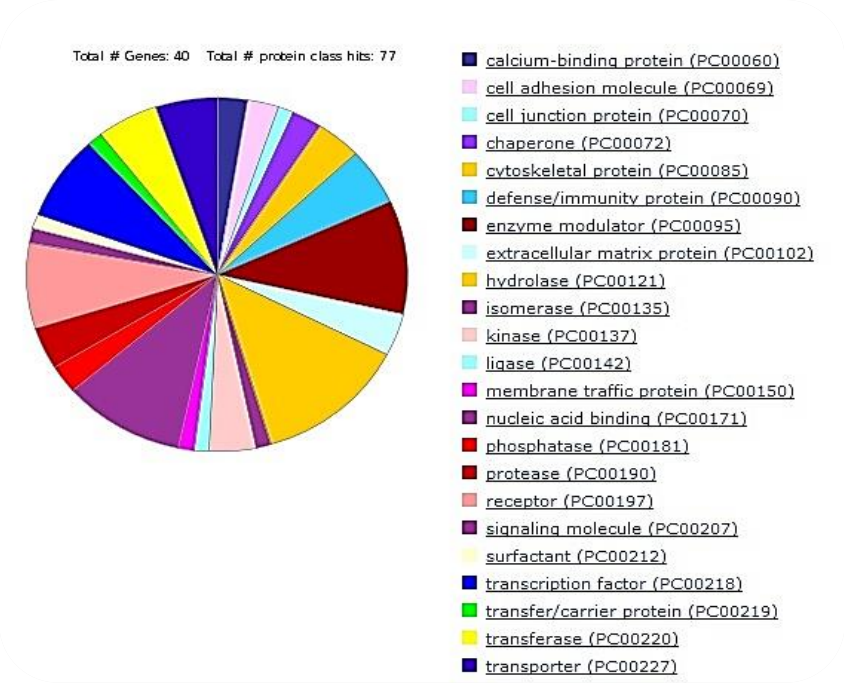


Figure 5.7: Distribution of differentially methylated genes based on protein classes [45]

Our novel approach enables researchers to look at global methylation signatures of genomic regions using immuno-precipitation technique and HTS. It is a fast, cost- effective technique to demarcate aberrant methylated regions within the genome and select candidate genes for locus-specific studies. Considering the uniqueness of methylome in every cell type and tissue, and the important role methylation plays in gene expression, it is imperative to have a complete map of methylation levels across the genome. With our computational methods and techniques it is possible to include whole genome in methylation analysis and perform unbiased estimate of the coverage values to detect novel differentially methylated regions.

LIMITATION AND FUTURE WORK

In all of our experiments discussed so far, we have excluded data with average coverage greater than 5 and data with lower values ($\lambda<5$) from the scope of the current study. At lower coverage depths, it is a challenge to distinguish between real coverage data and bias introduced by the SNPs. Mapping with mismatches is not feasible as it is a very computationally expensive task. The proposed approach has to be therefore extended to lower values of coverage ($\lambda<5$). Using a part of coverage distribution to estimate λ could be a practical solution as illustrated in the following figure.

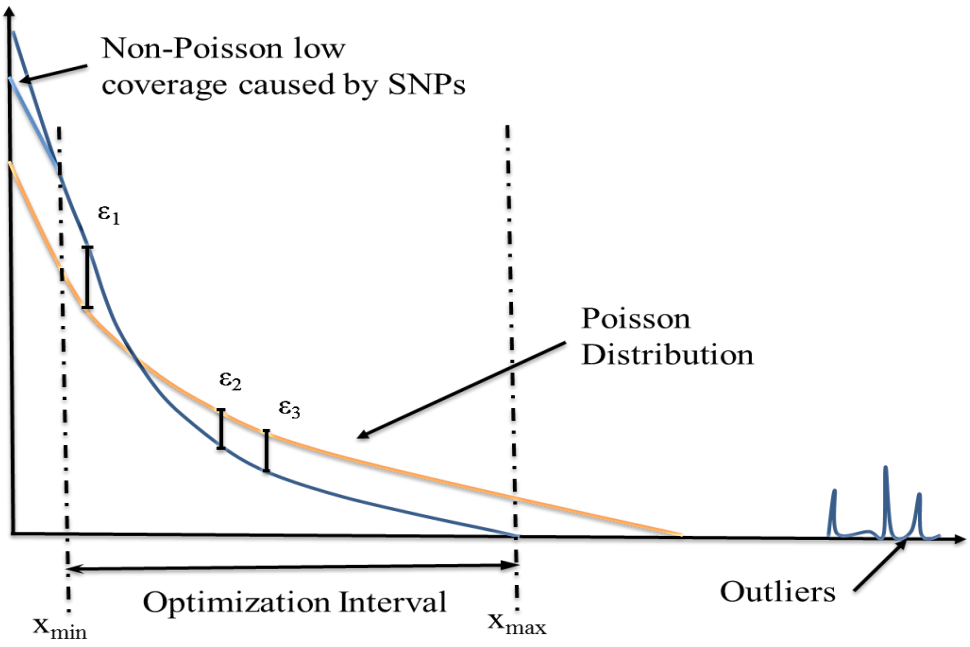


Figure 6.1: calculate (minimize/ optimize) error within a closed interval of the coverage distribution

In this case, our assumption of data following Poisson-like distribution holds good in a specific interval defined by [x_{min}, x_{max}], and the assumptions are significantly violated beyond these regions. Therefore, objective function is minimized using TSS as a goodness-of-fit statistic in the region of interest marked with [x_{min}, x_{max}].

CONCLUSIONS

Accurate quantification and analysis of CNVs is an important step in many genomics studies like gene-expression analysis, comparative genomics [67], differential methylation analysis, etc. Current available methods to compute coverage are sensitive to outlying values (SNPs and DNA repeats) and fail to perform unbiased estimate of average coverage in a genome. Novel algorithms proposed in this dissertation overcome limitations of the current methods and significantly improve CNV estimation by minimizing the effects of SNPs and/ or repeat regions.

In this dissertation, experiments on simulated and real data demonstrated that Poisson-based model can be applied to genomic data to estimate average coverage. The algorithms were developed and implemented in C++ language following object-oriented principles. The developed model was validated using metagenomic samples and performed fairly well.

In conclusion, the proposed novel algorithms were applied to a variety of mixed samples and could successfully employed to perform estimation of bacterial abundance and pathogen detection in presence of other organisms. To sum up, the model is suitable to be used in microbiome and pathogen detection studies and is likely to improve the analysis of CNVs in other areas like gene-expression and DNA methylation analysis.

REFERENCES

[1] Abdullah K. Alqallaf, Fuad M. Alkoot and Mash'el S. Aldabbous (2013). Discovering the Genetics of Autism, Recent Advances in Autism Spectrum Disorders - Volume I, Prof. Michael Fitzgerald (Ed.), ISBN: 978-953-51-1021-7, InTech, DOI: 10.5772/53797. Available from: http://www.intechopen.com/books/recent-advances-in-autism-spectrum-disorders-volume-i/discovering-the-genetics-of-autism

[2] Akalin, A., Garrett-Bakelman, F. E., Kormaksson, M., Busuttil, J., Zhang, L., Khrebtukova, I., Milne, T. A., Huang, Y., Biswas, D., Hess, J. L. & Others (2012). Base-pair resolution DNA methylation sequencing reveals profoundly divergent epigenetic landscapes in acute myeloid leukemia. *PLOS Genetics*, 8 (6), p. 1002781.

[3] Balmain, A. (1995). Cancer: exploring the bowels of DNA methylation. *Current Biology*, 5 (9), pp. 1013--1016.

[4] Beckmann, J. S., Estivill, X. & Antonarakis, S. E. (2007). Copy number variants and genetic traits: closer to the resolution of phenotypic to genotypic variability. *Nature Reviews Genetics*, 8 (8), pp. 639--646.

[5] Boon, K., Tomfohr, J. K., Bailey, N. W., Garantziotis, S., Li, Z., Brass, D. M., Maruoka, S., Hollingsworth, J. W. & Schwartz, D. A. (2008). Evaluating genome-wide DNA methylation changes in mice by methylation specific digital karyotyping. *BMC Genomics*, 9 (1), p. 598.

[6] Bradbury, J. (2003). Human epigenome project-up and running. *Plos Biology*, 1 (3), p. 82.

[7] Cheney, E. W. & Kincaid, D. (2004). *Numerical Mathematics and Computing.* 5th ed. Monterey, Calif.: Brooks/Cole Pub. Co.

[8] Constancia, M., Pickard, B., Kelsey, G. & Reik, W. (1998). Imprinting mechanisms. *Genome Research*, 8 (9), pp. 881--900.

[9] Dalay, N., Criekinge, W. V., Ling, S., Guerrero-Preston, R., Meijer, G., Demokan, S., Yalniz, Z., Kim, M., Louwagie, J., Pinto Morais De Carvalho, B. & Others (2009). Promoter DNA methylation of oncostatin m receptor-beta as a novel diagnostic and therapeutic marker in colon cancer, 4(8), p. 6555.

[10] Ewing, B., Hillier, L., Wendl, M. C., Green, P. (1998). Base-calling of automated sequencer traces using phred. i. accuracy assessment. *Genome Research*, 8(3), pp.175-85.

[11] Ewing, B. & Green, P. (1998). Base-calling of automated sequencer traces using phred. ii. error probabilities. *Genome Research*, 8 (3), pp. 186--194.

[12] Fassi Fehri, L., Mak, T. N., Laube, B., Brinkmann, V., Ogilvie, L. A., Mollenkopf, H., Lein, M., Schmidt, T., Meyer, T. F. and Bruggemann, H. 2011. Prevalence of Propionibacterium acnes in diseased prostates and its inflammatory and transforming activity on prostate epithelial cells. *International Journal of Medical Microbiology*, 301 (1), pp. 69--78.

[13] Feinberg, A. P. (2010). Epigenomics reveals a functional genome anatomy and a new approach to common disease. *Nature Biotechnology*, 28 (10), p. 1049.

[14] Feinberg, A. P., Vogelstein, B. & Others (1983). Hypomethylation distinguishes genes of some human cancers from their normal counterparts. *Nature*, 301 (5895), pp. 89--92.

[15] Fofanov, Y., Luo, Y., Katili, C., Wang, J., Belosludtsev, Y., Powdrill, T., Belapurkar, C., Fofanov, V., Li, T., Chumakov, S. & Others (2004). How independent are the appearances of n-mers in different genomes? *Bioinformatics*, 20 (15), pp. 2421--2428.

[16] Gama-Sosa, M. A., Midgett, R. M., Slagel, V. A., Githens, S., Kuo, K. C., Gehrke, C. W. & Ehrlich, M. (1983). Tissue-specific differences in DNA methylation in various mammals. *Biochimica Et Biophysica Acta (BBA)-Gene Structure And Expression*, 740 (2), pp. 212--219.

[17] Genome.gov. (2014). *DNA sequencing costs*. [online] Retrieved from: http://www.genome.gov/sequencingcosts [Accessed: 19 Feb 2014].

[18] Glenn, T. C. (2011). Field guide to next-generation DNA sequencers. *Molecular Ecology Resources*, 11 (5), pp. 759--769.

[19] Golovko, G. (2012). Effect of repeatable regions on ability to estimate copy number variation in human genome by high throughput sequencing. PhD. Department of Computer Science, University of Houston.

[20] Golovko, G., Khanipov, K., Rojas, M., Martinez-Alcantara, A., Howard, J. J., Ballesteros, E., Gupta, S., Widger, W. & Fofanov, Y. (2012). Slim-filter: an interactive windows-based application for illumina genome analyzer data assessment and manipulation. *BMC Bioinformatics*, 13 (1), p. 166.

[21] Gonzalez, E., Kulkarni, H., Bolivar, H., Mangano, A., Sanchez, R., Catano, G., Nibbs, R. J., Freedman, B. I., Quinones, M. P., Bamshad, M. J. & Others (2005). The influence of CCL3L1 gene-containing segmental duplications on HIV-1/AIDS susceptibility. *Science*, 307 (5714), pp. 1434--1440.

[22] Holliday, R. (1979). A new theory of carcinogenesis. *British Journal of Cancer*, 40 (4), p. 513.

[23] Holliday, R. (1991). Mutations and epimutations in mammalian cells. *Mutation Research/Fundamental and Molecular Mechanisms of Mutagenesis*, 250 (1), pp. 351--363.

[24] Hotchkiss, R. D. (1948). The quantitative separation of purines, pyrimidines, and nucleosides by paper chromatography. *Journal of Biological Chemistry*, 175 (1), pp. 315--332.

[25] Hysolli, E., Jung, Y. W., Tanaka, Y., Kim, K. Y. & Park, I. H. (2012). The lesser known story of x-chromosome reactivation: a closer look into the reprogramming of the inactive X chromosome. *Cell Cycle*, 11 (2), pp. 229--235.

[26] Johnson, A. A., Akman, K., Calimport, S. R., Wuttke, D., Stolzing, A. & De Magalhaes, J. P. (2012). The role of DNA methylation in aging, rejuvenation, and age-related disease. *Rejuvenation Research*, 15 (5), pp. 483--494.

[27] Jones, A., Teschendorff, A. E., Li, Q., Hayward, J. D., Kannan, A., Mould, T., West, J., Zikan, M., Cibula, D., Fiegl, H. & Others (2013). Role of DNA methylation and epigenetic silencing of hand2 in endometrial cancer development. *Plos Medicine*, 10 (11), p. 1001551.

[28] Kaminsky, Z. A., Tang, T., Wang, S., Ptak, C., Oh, G. H., Wong, A. H., Feldcamp, L. A., Virtanen, C., Halfvarson, J., Tysk, C. & Others (2009). DNA methylation profiles in monozygotic and dizygotic twins. *Nature Genetics*, 41 (2), pp. 240--245.

[29] Kahn, S. D. (2011). On the future of genomic data. *Science*, 331 (6018), pp. 728--729.

[30] Kidd, J. M., Cooper, G. M., Donahue, W. F., Hayden, H. S., Sampas, N., Graves, T., Hansen, N., Teague, B., Alkan, C., Antonacci, F. & Others (2008). Mapping and sequencing of structural variation from eight human genomes. *Nature*, 453 (7191), pp. 56--64.

[31] Koukoura, O., Sifakis, S., Sp & Idos, D. A. (2012). DNA methylation in the human placenta and fetal growth (review). *Molecular Medicine Reports*, 5 (4), p. 883.

[32] Krivtsov, A. V., Feng, Z., Lemieux, M. E., Faber, J., Vempati, S., Sinha, A. U., Xia, X., Jesneck, J., Bracken, A. P., Silverman, L. B. & Others (2008). H3k79 methylation profiles define murine and human mll-af4 leukemias. *Cancer Cell*, 14 (5), pp. 355--368.

[33] Ku, C. & Roukos, D. H. (2013). From next-generation sequencing to nanopore sequencing technology: paving the way to personalized genomic medicine. *Expert Review of Medical Devices*, 10 (1), pp. 1--6.

[34] Kuang, S., Tong, W., Yang, H., Lin, W., Lee, M., Fang, Z., Wei, Y., Jelinek, J., Issa, J. & Garcia-Manero, G. (2008). Genome-wide identification of aberrantly methylated promoter associated CpG islands in acute lymphocytic leukemia. *Leukemia*, 22 (8), pp. 1529--1538.

[35] La Spada, A. R. & Taylor, J. P. (2010). Repeat expansion disease: progress and puzzles in disease pathogenesis. *Nature Reviews Genetics*, 11 (4), pp. 247--258.

[36] Laird, P. W. (2010). Principles and challenges of genomewide DNA methylation analysis. *Nature Reviews Genetics*, 11 (3), pp.191--203.

[37] Lee, K. & Pausova, Z. (2012). Cigarette smoking and DNA methylation. *Frontiers in Genetics*, 4 pp. 132--132.

[38] Lobo, I. (2008). Copy number variation and genetic disease. *Nature Education* 1(1):65.

[39] Lykkebak, A. (2014). *NGS survey 2013 - clc bio*. [online] Retrieved from: http://www.clcbio.com/blog/ngs-survey-2013/ [Accessed: 21 Feb 2014].

[40] Mccarroll, S. A. & Altshuler, D. M. (2007). Copy-number variation and association studies of human disease. Nature Genetics, 39 pp. 37--42.

[41] Martino, D., Loke, Y. J., Gordon, L., Ollikainen, M., Cruickshank, M. N., Saffery, R. & Craig, J. M. (2013). Longitudinal, genome-scale analysis of DNA methylation in twins from birth to 18 months of age reveals rapid epigenetic change in early life and pair-specific effects of discordance. *Genome Biol*, 14 (5), p. 42.

[42] Maruyama, R., Toyooka, S., Toyooka, K. O., Harada, K., Virmani, A. K., Z"Ochbauer-M"Uller, S., Farinas, A. J., Vakar-Lopez, F., Minna, J. D., Sagalowsky, A. & Others (2001). Aberrant promoter methylation profile of bladder cancer and its relationship to clinicopathological features. *Cancer Research*, 61 (24), pp. 8659--8663.

[43] Maxam, A. M. & Gilbert, W. (1977). A new method for sequencing DNA. *Proceedings of the National Academy of Sciences*, 74 (2), pp. 560--564.

[44] Medh, R. D., Webb, M. S., Miller, A. L., Johnson, B. H., Fofanov, Y., Li, T., Wood, T. G., Luxon, B. A. & Thompson, E. B. (2003). Gene expression profile of human lymphoid cem cells sensitive and resistant to glucocorticoid-evoked apoptosis. *Genomics*, 81 (6), pp. 543--555.

[45] Mi, H., Lazareva-Ulitsky, B., Loo, R., Kejariwal, A., V, Ergriff, J., Rabkin, S., Guo, N., Muruganujan, A., Doremieux, O., Campbell, M. J. & Others (2005). The panther database of protein families, subfamilies, functions and pathways. *Nucleic Acids Research*, 33 (suppl 1), pp. 284--288.

[46] Miller, A. L., Geng, C., Golovko, G., Sharma, M., Yan, J., Schwartz, J. R., Sowers, L., Widger, W. R., Fofanov, Y., Vedeckis, W., and Thompson, B. E. (2014) Epigenetic alteration by DNA-demethylating treatment restores apoptotic response to glucocorticoids in dexamethasone-resistant human malignant lymphoid cells. *Cancer Cell International*, 14(1), p.35.

[47] Morison, I. M. & Reeve, A. E. (1998). A catalogue of imprinted genes and parent-of-origin effects in humans and animals. *Human Molecular Genetics,* 7 (10), pp. 1599--1609.

[48] Paulsen, M. & Ferguson-Smith, A. C. (2001). DNA methylation in genomic imprinting, development, and disease. *The Journal of Pathology*, 195 (1), pp. 97--110.

[49] Perry, G. (2009). The evolutionary significance of copy number variation in the human genome. *Cytogenetic and Genome Research*, 123 (1-4), pp. 283—287.

[50] Perry, G. H., Dominy, N. J., Claw, K. G., Lee, A. S., Fiegler, H., Redon, R., Werner, J., Villanea, F., Mountain, J. L., Misra, R. & Others (2007). Diet and the evolution of human amylase gene copy number variation. *Nature Genetics*, 39 (10), pp. 1256--1260.

[51] Raynal, N. J., Si, J., Taby, R. F., Gharibyan, V., Ahmed, S., Jelinek, J., Est'Ecio, M. R. & Issa, J. J. (2012). DNA methylation does not stably lock gene expression but instead serves as a molecular mark for gene silencing memory. *Cancer Research*, 72 (5), pp. 1170--1181.

[52] Reik, W. & Walter, J. (2001). Evolution of imprinting mechanisms: the battle of the sexes begins in the zygote. *Nature Genetics*, 27 (3), pp. 255--256.

[53] Rhounim, L., Rossignol, J. & Faugeron, G. (1992). Epimutation of repeated genes in ascobolus immersus. *The EMBO Journal*, 11 (12), p. 4451.

[54] Richards, R. I. & Sutherl (1996). Repeat offenders: simple repeat sequences and complex genetic problems. Human Mutation, 8 (1), pp. 1--7.

[55] Rodriguez-Rodero, S., Fernandez-Morera, J., Fern, Ez, A. F., Menendez-Torre, E. & Fraga, M. F. (2010). Epigenetic regulation of aging. *Discovery Medicine*, 10 (52), pp. 225--233.

[56] Sanger, F. & Coulson, A. R. (1975). A rapid method for determining sequences in DNA by primed synthesis with DNA polymerase. *Journal of Molecular Biology*, 94 (3), pp. 441--448.

[57] Sanger, F., Nicklen, S. & Coulson, A. R. (1977). DNA sequencing with chain-terminating inhibitors. *Proceedings of the National Academy of Sciences*, 74 (12), pp. 5463--5467.

[58] Selamat, S. A., Chung, B. S., Girard, L., Zhang, W., Zhang, Y., Campan, M., Siegmund, K. D., Koss, M. N., Hagen, J. A., Lam, W. L. & Others (2012). Genome-scale analysis of DNA methylation in lung adenocarcinoma and integration with mrna expression. *Genome Research*, 22 (7), pp. 1197--1211.

[59] Senner, C. E. (2011). The role of DNA methylation in mammalian development. *Reproductive Biomedicine Online*, 22 (6), pp. 529--535.

[60] Sharma, M., Fofanov, Y. and Widger, W. R. (2013). Detecting Altered Methylation States using High Throughput DNA Sequencing, Poster session presented at the meeting of the *11th Annual Rocky Mountain Bioinformatics Conference*. The International Society for Computational Biology (ISCB), Aspen, Colorado.

[61] Sharma, M., Albayrak, L., Fofanov, Y. and Widger, W. R. (2014) Estimating Genome Coverage Using a Poisson Mixture Model, Poster session presented at the meeting of the *7th Bayesian Biostatistics and Bioinformatics Conference.* MD Anderson, Houston, TX.

[62] Shen, L. & Waterl (2007). Methods of DNA methylation analysis. *Current Opinion in Clinical Nutrition & Metabolic Care*, 10 (5), pp. 576--581.

[63] Synthesis.cc. (2014). *Synthesis: search results*. [online] Retrieved from: http://www.synthesis.cc/cgi-bin/mt/mt-search.cgi?blog_id=1&tag=Carlson%20Curves&limit=20 [Accessed: 19 Feb 2014].

[64] Szyf, M. (2009). Epigenetics, DNA methylation, and chromatin modifying drugs. *Annual Review of Pharmacology and Toxicology*, 49 pp. 243--263.

[65] Tuzun, E., Sharp, A. J., Bailey, J. A., Kaul, R., Morrison, V. A., Pertz, L. M., Haugen, E., Hayden, H., Albertson, D., Pinkel, D. & Others (2005). Fine-scale structural variation of the human genome. *Nature Genetics*, 37 (7), pp. 727--732.

[66] Tomizawa, S., Nowacka-Woszuk, J., Kelsey, G. & Others (2012). DNA methylation establishment during oocyte growth: mechanisms and significance. *International Journal of Developmental Biology*, 56, pp. 867--875.

[67] Tomkins, J. P. (2011). How genomes are sequenced and why it matters: implications for studies in comparative genomics of humans and chimpanzees. *Answers Research Journal*, 4, pp. 81--88.

[68] Treangen, T. J. & Salzberg, S. L. (2012). Repetitive DNA and next-generation sequencing: computational challenges and solutions. *Nature Reviews Genetics*, 13 (1), pp. 36--46.

[69] Ushijima, T. & Okochi-Takada, E. (2005). Aberrant methylations in cancer cells: where do they come from? *Cancer Science*, 96 (4), pp. 206--211.

[70] Vaissi`Ere, T., Hung, R. J., Zaridze, D., Moukeria, A., Cuenin, C., Fasolo, V., Ferro, G., Paliwal, A., Hainaut, P., Brennan, P. & Others (2009). Quantitative analysis of DNA methylation profiles in lung cancer identifies aberrant DNA methylation of

specific genes and its association with gender and cancer risk factors. *Cancer Research*, 69 (1), pp. 243--252.

[71] Wetterstrand, K. A. (2014). *DNA Sequencing Costs*. [online] Retrieved from: http://www.genome.gov/sequencingcosts. [Accessed: 22 Mar 2014].

[72] Zhang, Y. & Jeltsch, A. (2010). The application of next generation sequencing in DNA methylation analysis. *Genes*, 1 (1), pp. 85--101.

Appendix I

Table I-1 summarizes the simulation results. For an integer array holding 1,000,000 coverage values generated for given Poisson distribution (λ=1, 5, 10, 20, 30, 40), additional values (0s and higher values) were inserted to simulate SNPs and DNA repeats in real genomes.

Table I-1: Simulation results for genome length=1,000,000, minimum frequency =0, maximum frequency = 1000, and read size=20

λ	% 0s	% high values	λ estimate	α estimate	error	Min λ	Max λ
1	0.1	0.2	0.843825	0.796768	4.46E-06	0.1	11
1	0.1	0.3	0.843641	0.720999	8.27E-06	0.1	11
1	0.1	0.4	0.843785	0.652036	1.30E-05	0.1	11
1	0.1	0.5	0.84438	0.590222	1.82E-05	0.1	12
1	0.2	0.2	0.703109	0.779029	5.69E-06	0.1	11
1	0.3	0.2	0.585248	0.766345	6.67E-06	0.1	11
1	0.4	0.2	0.487393	0.758197	7.17E-06	0.1	11
5	0.1	0.2	4.94975	0.742893	9.99E-06	0.1	15
5	0.1	0.3	4.94192	0.671991	1.30E-05	0.1	16
5	0.1	0.4	4.94345	0.60787	1.71E-05	0.1	16
5	0.1	0.5	4.94658	0.549822	2.20E-05	0.1	17
5	0.2	0.2	4.87423	0.674073	2.59E-05	0.1	15
5	0.3	0.2	4.78805	0.611818	4.88E-05	0.1	15
5	0.4	0.2	4.6902	0.556653	7.62E-05	0.1	14
10	0.1	0.2	9.99455	0.740442	1.01E-05	0.1	20
10	0.1	0.3	9.99296	0.669881	1.32E-05	0.1	21
10	0.1	0.4	10.0036	0.606632	1.76E-05	0.1	22
10	0.1	0.5	9.99971	0.547783	2.24E-05	0.1	23

Table I-1 (Continued)

λ	% 0s	% high values	λ estimate	α estimate	error	Min λ	Max λ
10	0.2	0.2	9.99276	0.670382	2.61E-05	0.1	20
10	0.3	0.2	9.99389	0.606661	4.89E-05	0.1	20
10	0.4	0.2	9.99923	0.548144	7.69E-05	0.1	19
20	0.1	0.2	19.9995	0.740729	1.04E-05	0.1	31
20	0.1	0.3	20.003	0.670669	1.38E-05	0.1	31
20	0.1	0.4	20.001	0.607138	1.83E-05	0.1	32
20	0.1	0.5	19.9747	0.54909	2.36E-05	0.1	34
20	0.2	0.2	19.9957	0.670786	2.64E-05	0.1	30
20	0.3	0.2	19.9949	0.606711	4.95E-05	0.1	30
20	0.4	0.2	19.9963	0.548766	7.72E-05	0.1	29
30	0.1	0.2	29.9984	0.740912	1.06E-05	0.1	41
30	0.1	0.3	29.9805	0.669991	1.45E-05	0.1	42
30	0.1	0.4	29.996	0.606635	1.91E-05	0.1	43
30	0.1	0.5	30.0115	0.549015	2.50E-05	0.1	45
30	0.2	0.2	30.0025	0.669873	2.67E-05	0.1	40
30	0.3	0.2	29.9955	0.606113	4.97E-05	0.1	39
30	0.4	0.2	29.9836	0.549076	7.72E-05	0.1	39
50	0.1	0.2	49.9959	0.740411	1.11E-05	0.1	61
50	0.1	0.3	50.0067	0.670306	1.52E-05	0.1	62
50	0.1	0.4	50.006	0.606184	2.06E-05	0.1	64
50	0.1	0.5	49.9947	0.548824	2.73E-05	0.1	66
50	0.2	0.2	49.9808	0.669957	2.70E-05	0.1	60
50	0.3	0.2	49.9761	0.606643	5.01E-05	0.1	59
50	0.4	0.2	49.9829	0.54779	7.81E-05	0.1	58

Appendix II

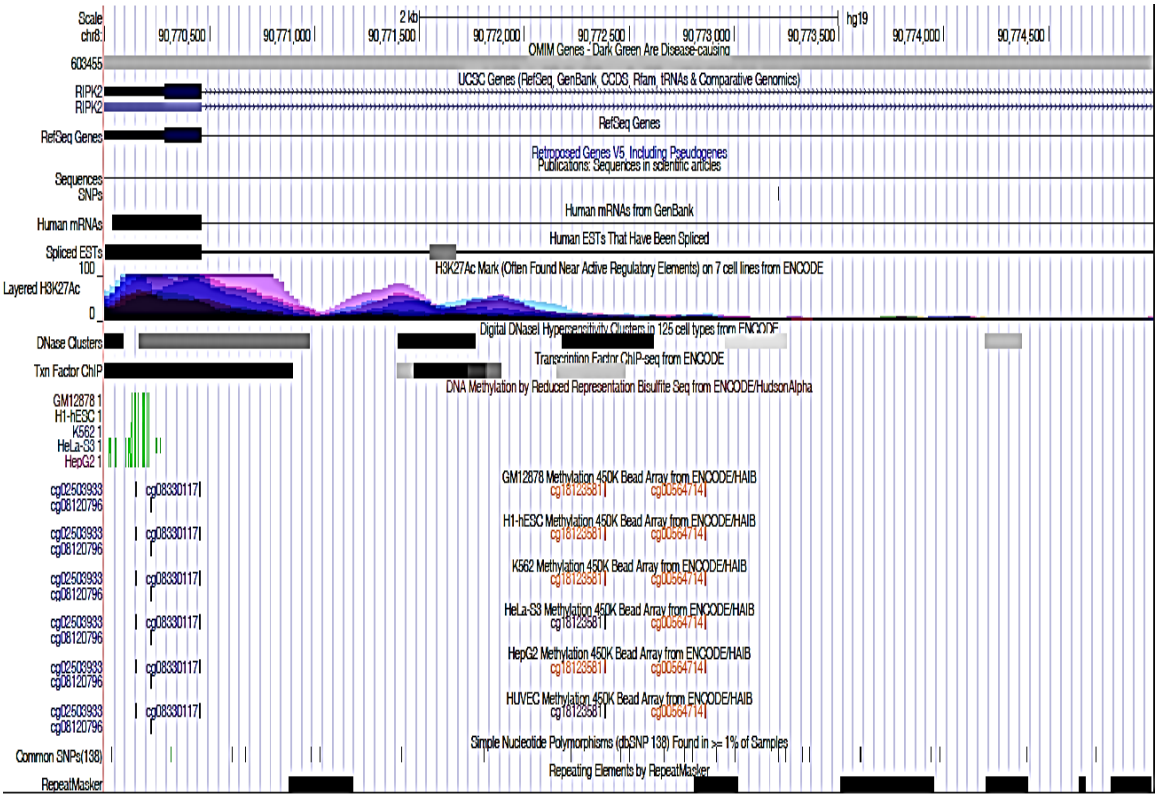


Figure II-1: Comparing methylation status of differentially expressed genes to existing studies, RIPK2 was found to be differentially methylated in our results. In order to validate our findings the results were compared with previously conducted experiments. Quite a few genes were reported and found to be differentially methylated in other cancer cell lines as well.

Appendix III

Table III-1: List of differentially methylated genes in ALL cell lines

Gene Name	Gene Description
ADARB2	adenosine deaminase, RNA-specific, B2
ADCK3	aarF domain containing kinase 3
ALS2,MPP4	amyotrophic lateral sclerosis 2 (juvenile)
ATL1	atlastin GTPase 1
B4GALNT4	beta-1,4-N-acetyl-galactosaminyl transferase 4
CAMTA1	calmodulin binding transcription activator 1
CAP2	CAP, adenylate cyclase-associated protein, 2 (yeast)
CAPN13	calpain 13
CBX4	chromobox homolog 4
COL6A1	collagen, type VI, alpha 1
CRHR2	corticotropin releasing hormone receptor 2
CYFIP2	cytoplasmic FMR1 interacting protein 2
CYP4F24P	cytochrome P450, family 4, subfamily F, polypeptide 2 4, Pseudogene
DIS3L2	DIS3 mitotic control homolog (S. cerevisiae)-like 2
DNAJC14,ORMDL2,SARNP	DnaJ (Hsp40) homolog, subfamily C, member 14
FLJ43860,PTP4A3	family with sequence similarity 25, member B
FMN2	formin 2
FOXL1	forkhead box L1
FUNDC2	FUN14 domain containing 2
GALNS	galactosamine (N-acetyl)-6-sulfate sulfatase
HLA-F-AS1	HLA-F Antisense RNA 1 (Non-Protein Coding)
HNRNPR	heterogeneous nuclear ribonucleoprotein R
IQGAP2	IQ motif containing GTPase activating protein 2
KCNJ6	potassium inwardly-rectifying channel, subfamily J, member 6
KCNQ2	potassium voltage-gated channel, KQT-like subfamily, member 2
LOC100132891	cDNA FLJ53548
LOC440518	golgin A2 pseudogene
MAPK4	mitogen-activated protein kinase 4
PDZRN4	PDZ domain containing ring finger 4

Table III-1 (Continued)

Gene Name	Gene Description
PTP4A3	protein tyrosine phosphatase type IVA, member 3
RIPK2	receptor-interacting serine-threonine kinase 2
SDK1	sidekick homolog 1, cell adhesion molecule (chicken)
SIPA1L2	signal-induced proliferation-associated 1 like 2
SLC25A21	solute carrier family 25 (mitochondrial oxodicarboxylate carrier), member 21
SOX6	SRY (sex determining region Y)-box 6
SYNPO2	synaptopodin 2
TENM1	teneurin transmembrane protein 1, ODZ1
TMEM178A	Transmembrane Protein 178A
TRAPPC9	trafficking protein particle complex 9
ZNF831	zinc finger protein 831
FAM25C	family with sequence similarity 25, member C
FAM25G	family with sequence similarity 25, member G
FAM72B	family with sequence similarity 72, member B
FAM86B1	family with sequence similarity 86, member B1
MPP4	membrane protein, palmitoylated 4 (MAGUK p55 subfamily member 4)
MRC1	mannose receptor, C type 1