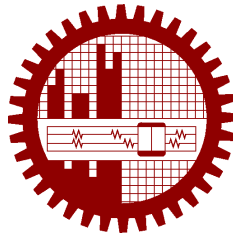


M.Sc. Engg. (CSE) Thesis

A Reference Free Method for Designing Guide RNAs for
CRISPR-Cas9

Submitted by
Mahmudur Rahman Hera
0417052024

Supervised by
Dr. Atif Hasan Rahman



Submitted to
Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology
Dhaka, Bangladesh

in partial fulfillment of the requirements for the degree of
Master of Science in Computer Science and Engineering

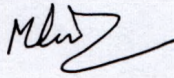
August 2020

Dedicated to Mahmuda and Habib

Candidate's Declaration

I, do, hereby, certify that the work presented in this thesis, titled, "A Reference Free Method for Designing Guide RNAs for CRISPR-Cas9", is the outcome of the investigation and research carried out by me under the supervision of Dr. Atif Hasan Rahman, Assistant Professor, Department of CSE, BUET.

I also declare that neither this thesis nor any part thereof has been submitted anywhere else for the award of any degree, diploma or other qualifications.

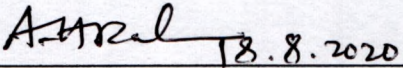
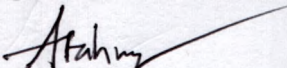
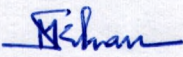
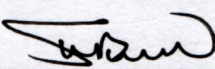
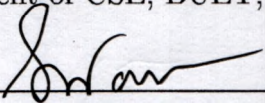


Mahmudur Rahman Hera

0417052024

The thesis titled "A Reference Free Method for Designing Guide RNAs for CRISPR-Cas9", submitted by Mahmudur Rahman Hera, Student ID 0417052024, Session April 2017, to the Department of Computer Science and Engineering, Bangladesh University of Engineering and Technology, has been accepted as satisfactory in partial fulfilment of the requirements for the degree of Master of Science in Computer Science and Engineering and approved as to its style and contents on August 18, 2020.

Board of Examiners

1.  18.8.2020
Dr. Atif Hasan Rahman
Assistant Professor
Department of CSE, BUET, Dhaka
Chairman
(Supervisor)
2. 
Dr. A.K.M. Ashikur Rahman
Professor and Head
Department of CSE, BUET, Dhaka
Member
(Ex-Officio)
3. 
Dr. M. Sohel Rahman
Professor
Department of CSE, BUET, Dhaka
Member
4. 
Dr. Md. Shamsuzzoha Bayzid
Assistant Professor
Department of CSE, BUET, Dhaka
Member
5. 
Dr. Swakkhar Shatabda
External Examiner
Associate Professor
Department of CSE
United International University, Dhaka
Member
(External)

Acknowledgement

Bismillahir rahmanir rahim.

I am thankful to Allah for guiding me through the hard times I faced while doing this work. Research comes with a lot of uncertainty, lack of confidence and mental pressure. Thankfully, I was able to overcome those and finish this work with His help, which came in the form of supportive friends, an intelligent, persevering and understanding wife, and a very smart and helpful supervisor.

I am grateful to all of these persons, without whom, my thesis would be unfinished, and my life incomplete.

Dhaka
August 18, 2020

Mahmudur Rahman Hera
0417052024

Contents

Candidate's Declaration	i
Board of Examiners	ii
Acknowledgement	iii
List of Figures	vii
List of Tables	ix
Abstract	x
1 Introduction	1
1.1 DNA and life	1
1.2 Genome editing technologies	1
1.3 Importance of genome editing	2
1.4 CRISPR/Cas9 to edit the genome	3
1.5 Importance of reference-free work	4
1.6 Contributions of the research	5
1.7 Thesis organization	6
2 Literature review	7
2.1 Introduction of CRISPR/Cas9 as a genome editing technology	7
2.1.1 CRISPR/Cas system	8
2.1.2 CRISPR/Cas9 system and the guide RNA	9
2.2 gRNA and success of CRISPR/Cas9 experiments	10
2.3 The gRNA designing problem	11
2.4 Current state of gRNA dsigning problem	12
2.4.1 Models to predict on-target cutting efficiency	12
2.4.2 Models to identify off-target cuts	13
2.4.3 Computational tools using the models	14
2.5 Motivation of the work	15

3	Preliminaries	17
3.1	Sequenced reads in DNA sequencing	17
3.2	k -mer	17
3.3	Prefix tree: trie	18
3.4	k -mer counting in sequenced reads: k -mer spectrum	19
3.5	Maximum Likelihood Estimation	20
3.6	Expectation Maximization algorithm	21
3.7	The Poisson distribution to model sequenced reads	21
3.8	EM in mixture of Poisson distributions	22
3.9	Bayes decomposition: priors and posteriors	23
4	Methods	24
4.1	Overview	25
4.2	Computing personalized target sequence	25
4.3	Identifying candidate gRNAs	26
4.4	Counting k -mers in sequenced reads	27
4.4.1	Storing the k -mer counts	27
4.4.2	Determining the k -spectrum of sequenced reads	27
4.5	Scoring candidate gRNAs	28
4.5.1	Determining expected number of cuts in the genome	28
4.5.2	Determining expected number of cuts in the target	30
4.5.3	Determining $C_{G,t}$: number of appearances of the target in reference	31
4.5.4	Calculating the score	33
4.5.5	Learning probability distributions and estimating priors	33
4.5.6	Calculating the posterior probabilities	35
4.6	Determining on-target scores	36
4.7	Importance of the error component	37
4.8	Interpretation of the score	37
4.8.1	Direct interpretation	37
4.8.2	Relation with specificity score	38
5	Experiments and results	39
5.1	Datasets	39
5.1.1	The genome assemblies	39
5.1.2	The sequenced reads	40
5.1.3	Using paired-end reads	40
5.2	Comparing inverse of specificity scores: with and without a reference	41
5.3	Determining the default cut-off score and pruning guide RNAs	43

5.4	Comparison with GuideScan to validate the performance	45
5.5	Comparison with GuideScan in terms of variant-aware gRNA design . .	47
5.6	Comparison with other gRNA designing tools	49
5.6.1	Number of guide RNAs recognized	51
5.6.2	Overlap of the guide RNAs recognized by kRISP-meR and other tools	51
5.6.3	Off-target cuts made by the guide RNAs	52
5.7	Benchmarking the running time of kRISP-meR	54
5.8	Experimenting the accuracy of the personalized target site	56
6	Conclusions	58
6.1	Discussions	59
6.2	Future works	61
6.3	Availability	62
	References	63

List of Figures

1.1	CRISPR defense mechanism at a high level (This image was taken from [1])	3
2.1	CRISPR/CAS defence mechanism (This image was taken from [2])	9
2.2	Editing the genome after double-strand break has been introduced by the Cas9 enzyme (This image was taken from [3]).	10
3.1	k -mer example. The sequence ATGG has two 3-mers: ATG and TGG.	18
3.2	An example trie.	18
3.3	An example 8-mer spectrum for <i>Escherichia coli</i> comparing 8-mers' frequency with their number of occurrences.	19
3.4	Flow of EM algorithm in determining the parameters of a very simple Gaussian Mixture Model (This image was taken from [4]).	23
4.1	Overview of kRISP-meR.	24
4.2	23-spectrum of <i>E. coli</i> sequenced reads (read coverage: 74).	27
4.3	Posterior probabilities determined by kRISP-meR working with sequenced reads of <i>Staphylococcus aureus</i> (taken from GAGE [5]).	36
5.1	Comparison of inverse-specificity scores calculated by kRISP-meR and separately calculated using a reference (each point in the figure denotes a gRNA).	42
5.2	CDFs of specificity scores calculated by kRISP-meR and separately calculated using a reference.	43
5.3	Number of off-target cuts (predicted and actual) made by guide RNAs with a specificity score lower/higher than a certain value.	44
5.4	Venn diagrams showing numbers of candidate guide RNAs found by only kRISP-meR, only GuideScan and both tools for randomly chosen regions in <i>S. aureus</i> genome, <i>S. cerevisiae</i> genome and <i>H. sapiens</i> chromosome 14.	45
5.5	Numbers of guide RNAs predicted by kRISP-meR and GuideScan with off-target effects with no mismatches.	46
5.6	Partial phylogenetic tree constructed using the strains of <i>E. coli</i> . [6]	47

5.7	Number of guide RNAs resulting in perfect off-targets identified by kRISP-meR and GuideScan. GuideScan uses K-12 assembly and kRISP-meR uses ATTC8739 reads. The off-targets are counted in ATTC8739 assembly.	48
5.8	Overlap of the guide RNAs recognized by CRISPR and other tools (aggregated over a number of experiments).	51
5.9	Percentage of high-specificity guide RNAs captured by kRISP-meR. . .	52
5.10	Number of guide RNAs that resulted in perfect off-target cleavages. . .	53
5.11	Number of off-target cleavages.	53
5.12	Running time of the various components of kRISP-meR. The running times presented here are the average of multiple experiments for a particular number of mismatches considered.	55
5.13	Capturing individual genetic variations and determining personalized target sequence by kRISP-meR	56

List of Tables

- 5.1 Datasets used to perform various experiments and their sources 41
- 5.2 Genomic co-ordinates of the target sites 41
- 5.3 Genomic co-ordinates of the target sites of *Saccharomyces cerevisiae*, and number of guide RNAs recognized by different tools for these targets . 49
- 5.4 Genomic co-ordinates of the target sites of *Caenorhabditis elegans*, and number of guide RNAs recognized by different tools for these targets . 50
- 5.5 Genomic co-ordinates of the target sites of *Drosophila melanogaster*, and number of guide RNAs recognized by different tools for these targets . 50

Abstract

DNA or deoxyribonucleic acid is a double helix polymer molecule that carries the genetic information and controls every biochemical process in the body of all organisms. By altering the DNA artificially, it is possible to correct genetic problems, treat diseases, and even eradicate diseases. Genome editing technologies have been developed in this regard to correct the DNA. Such technologies often work by cutting the DNA double helix at an intended location. These double-helix breaks can later be repaired by modifying the sequence at those places. CRISPR/Cas9 is one such genome editing technology, which has been recognized to be highly specific, cost-effective, and less time-consuming compared to other technologies such as ZFN and TALEN. CRISPR/Cas9 system introduces a DNA cleavage by splicing the double-helix with the Cas9 enzyme. This enzyme is guided to a particular genomic location by a single guide RNA (sgRNA), or simply, guide RNA (gRNA). Genome editing using the CRISPR/Cas9 system, therefore, requires designing sgRNAs that are efficient and specific. These RNAs are usually designed using reference genomes, by scanning the genome for probable locations where a double-helix break could result. The requirement to have to scan the reference genome limits their use in organisms with incomplete reference genomes. We show that it is possible to design sgRNAs without a reference genome. We do this by directly utilizing genome sequencing reads and estimating the number of cuts introduced by a particular sgRNA by counting k -mers in the reads. Using this estimation, we give an alternative definition of the specificity score of a sgRNA. We also show that sgRNAs filtered and sorted using this score are highly specific by separately scanning the reference. We further show that our list of sgRNAs is similar to those of other sgRNA designer tools that work with a reference genome.

Chapter 1

Introduction

1.1 DNA and life

DNA, or deoxyribonucleic acid, is the hereditary material in humans and almost all other organisms. Nearly every cell in a person's body has the same DNA. Most DNA is located in the cell nucleus (where it is called nuclear DNA), but a small amount of DNA can also be found in the mitochondria (where it is called mitochondrial DNA or mtDNA) [7].

The information in DNA is stored as a code made up of four chemical bases: adenine (A), guanine (G), cytosine (C), and thymine (T). Human DNA consists of about 3 billion bases, and more than 99 percent of those bases are the same in all people [8,9]. The order, or sequence, of these bases determines the information available for building and maintaining an organism, similar to the way in which letters of the alphabet appear in a certain order to form words and sentences [7].

The information stored in DNA essentially determines everything about how an organism lives, reproduces, survives, even dies. All biochemical process with in the body of all organisms are controlled by the DNA.

1.2 Genome editing technologies

Genome editing technologies enable scientists to make changes to the DNA, leading to changes in physical traits, like eye color, and disease risk. Scientists use different technologies to do this. These technologies act like scissors, cutting the DNA at a specific spot. Then scientists can remove, add, or replace the DNA where it was cut.

The key to genome editing is creating a double-strand break (DSB or ds-break) at a specific location within the genome. Commonly used restriction enzymes are effective

at cutting DNA, but generally recognize and cut at multiple sites. To overcome this challenge and create site-specific DSB, three distinct classes of nucleases have been discovered and bioengineered to date.

- Zinc finger nucleases (ZFNs) [10]
- transcription-activator like effector nucleases (TALEN) [11]
- clustered regularly interspaced short palindromic repeats (CRISPR/Cas9) system [12, 13]

Based on the use of these nucleases, different genome-editing technologies are named. These vary in their ease of designing a genome-editing wet-lab experiment, required time, efficacy of the cut, specificity of targeting a particular location, and the ability to target multiple sites within the genome simultaneously.

1.3 Importance of genome editing

Genome editing is a method that lets scientists change the DNA of many organisms, including plants, bacteria, and animals. Editing DNA can lead to changes in physical traits, like eye color, hair color etc. By editing the genome, scientists are also able to reduce the risks of diseases [14]. This is often referred to as “gene therapy” [15]. Gene therapy, or treatments involving genome editing, have the potential to help treat diseases at a genomic basis, like cystic fibrosis and diabetes, and many genetic disorders [14]. This is a way to fix a genetic problem at its source [16]. The first clinical use of TALEN-based genome editing was in the treatment of CD19+ acute lymphoblastic leukemia in an 11-month old child in 2015. Modified donor T-cells were engineered to attack the leukemia cells, to be resistant to Alemtuzumab, and to evade detection by the host immune system after introduction [17, 18].

By editing the genome, scientists are also making way to eradicate diseases. Researchers have used CRISPR-Cas9 gene drives to modify genes associated with sterility in *A. gambiae*, the vector for malaria [19]. This technique has further implications in eradicating other vector borne diseases such as yellow fever, dengue, and Zika [20]. Antiviral applications for therapies targeting human viruses such as HIV, herpes, and hepatitis B virus are under research. CRISPR editing can be used to target the virus or the host to disrupt genes encoding the virus cell-surface receptor proteins [21].

A particular area where genome editing has much potential and has been used extensively is agriculture. Although regulatory frameworks have been imposed on

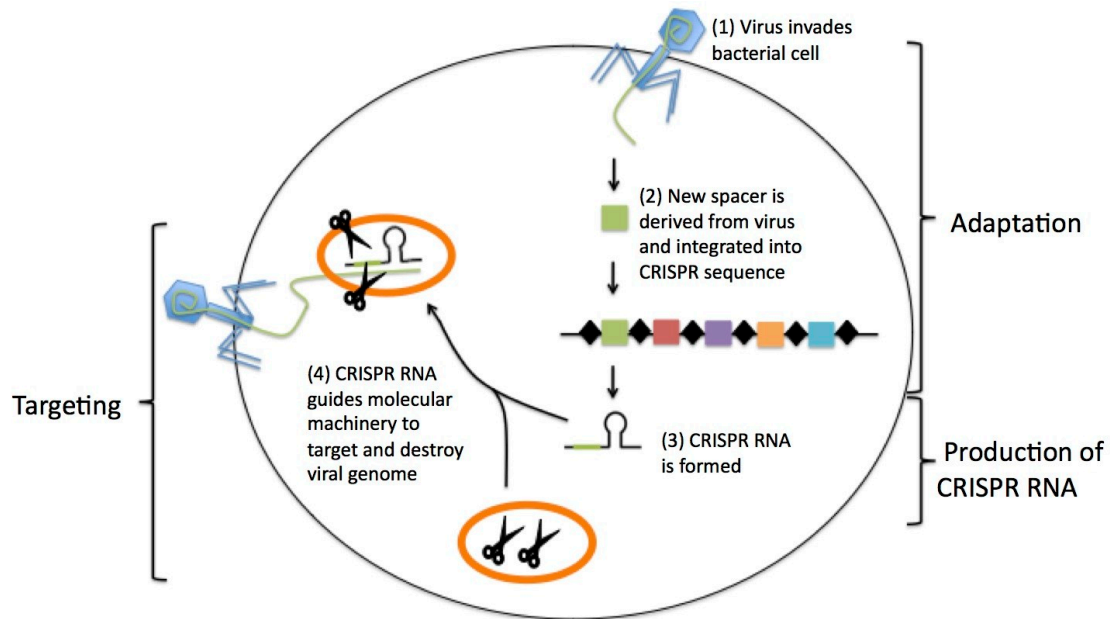


Figure 1.1: CRISPR defense mechanism at a high level (This image was taken from [1])

genetically modified (GM) crops with a view to ensure environmental biosafety, and thereby incurring more cost [22], genome-editing systems have been utilized in a wide variety of plant species to characterize gene functions and improve agricultural traits [23]. Trait stacking for maize and rice have been achieved using ZFN (a genome editing technology) [24, 25]. TALEN has been used to build bacterial blight resistance in rice [26], to introduce fragrance in rice grains [27], to increase oleic acid contents in soybean [28], and to improve saccharification efficiency in sugarcane [29].

CRISPR/Cas9 system has also been employed in agriculture, resulting in a number of breakthroughs. Use of CRISPR/Cas9 has enabled larger grain size with higher production in rice [30], increased protein content in wheat [31], higher amylopectin content in potato [32], bacterial speck resistance in tomato [33], and virus resistance in cucumber [34]. The list is not exhaustive, extensive review has been done to list the genome-editing contributions in agriculture [23, 35, 36].

1.4 CRISPR/Cas9 to edit the genome

In recent times, CRISPR/Cas9 has emerged as a popular genome editing technology. Originally, CRISPR (clustered regulatory interspaced palindromic repeats) is a bacterial defense mechanism against viruses. As viruses attack bacteria, the DNA snippets of the virus called spacers are recorded by the bacteria and stored in its own CRISPR sequence as a memory. When the same virus attacks later, CRISPR-RNA is formed using those

memory, guiding the splicer enzymes to the DNA of the virus. Then, the DNA of the virus is torn apart by cutting them at places. A high level pictorial representation is shown in Figure 1.1.

Using the same idea, CRISPR/Cas9 has been modified to edit the genome of other organisms. The DNA snippet(s) of the genomic region (where genome editing is intended) are recorded by the CRISPR/Cas9 system in form of single guide RNAs (sgRNAs, or simply, guide RNAs or gRNAs). The Cas9 is an enzyme (CRISPR-associated protein 9) that plays the role of a scissor which cuts the DNA at a certain location. The guide RNAs “guide” the scissor to that location. Naturally, if the guide RNAs are not designed with care, they can carry the Cas9 enzyme to an unintended genomic location and introduce a ds-break at an unintended location within the genome. Therefore, it is crucial to design the guide RNAs so that the gRNAs that may bind off-target are ruled out and the highly specific gRNAs are identified.

1.5 Importance of reference-free work

Compared to other genome editing technologies, CRISPR/Cas9 is favorable for the ease of use, cost-effectiveness, and reduced experiment time. As a result, huge number of CRISPR related research is being undertaken by the research community. Consequently, numerous computational tools have been developed to design these guide RNAs. Most of these tools scan the target site (where CRISPR cut is intended), identify guide RNAs, and then scans the whole reference genome to identify potential sites where those guide RNAs are likely to make a cut. Finally, particular scores indicating the number of off-target cuts are calculated for a certain target site.

However, having the reference genome at disposal is not always practical since genome assembly itself is a complicated problem. Assembling the whole genome of an organism and annotating all the loci accurately may take years to complete. Reference genome for a large number of crops is not available to date. This impediment makes CRISPR experiments for organisms with incomplete genome difficult and challenging. Moreover, the reference genome of an organism often does not represent the individuals accurately. That is, designing CRISPR/Cas9 experiments for an individual using the reference genome for that particular organism may not be a clever idea, since the genome of that individual can contain genetic variations with the reference.

To address these challenges, it is important to develop a guide RNA designing tool that can operate without the necessity of a reference genome.

1.6 Contributions of the research

In this thesis, we present kRISP-meR, a guide RNA designing tool that does not rely on a reference genome. The reference genome for a particular organism is assembled over years using the sequenced reads collected from individuals of that organism. The sequencing reads, being collected from the DNA of an individual, also contains all information of the DNA of that individual. Therefore, we utilize the sequencing reads directly both to decouple the use of a reference genome, and to identify guide RNAs personalized for that individual (whose sequencing reads are being used).

Goal of any gRNA designing tool is to identify guide RNAs with good on-target activity as well as minimal off-target activity. Good on-target activity refers to the characteristic of a guide RNA introducing a ds-break with high efficacy. On the other hand, minimal off-target activity refers to the ability of a guide RNA to target a location within the genome with high specificity. We identified the off-target activity of a guide RNA without any reference genome. Typically, the off-target activity of a particular guide RNA is summarized as a single score, ranging from 0.0 to 1.0, indicating the specificity of the gRNA. The existing gRNA designing tools use different definitions of this specificity score, almost all of which use features of the reference genome. In this thesis, we define a novel score to indicate the off-target activity of a particular guide RNA which does not require the use of a reference genome. To calculate this score, we model the sequencing reads as a probabilistic model, capturing the attributes of the sequencing technology (including the sequencing errors while collecting sequencing reads) with a mixture of probabilistic distributions. We calculate various components of this score from this model using Expectation Maximization algorithm and Maximul Likelihood Estimation. With no reference genome to work with, our calculations engage k -mer counting heavily. We use Jellyfish [37] to count k -mers in sequencing reads. To identify the personalized target sequence, we use Bowtie2 [38], Samtools [39] and Pilon [40] in a pipeline. We also calculate the on-target efficacy score of a target-site for a guide RNA using the Doench model [41]. We implemented all of these in Python 2.0 to develop kRISP-meR as a bioinformatics tool, which is available in <https://github.com/mahmudhera/kRISP-meR>.

We verify the definition of our score by calculating the same score with the reference genome and making a contrast of the two. To demonstrate that kRISP-meR, without a reference, generates highly specific gRNAs, we compare ourselves with three other gRNA designing tools that use the reference genome. Our experiments show that the off-target activity exhibited by the gRNAs identified by the other tools is similar to that of ours.

1.7 Thesis organization

This thesis is organized as follows: it introduces the literature review on CRISPR/Cas9 gRNA designing in Chapter 2. Next, the mathematical concepts used to develop kRISP-meR are elaborated in Chapter 3. After that, Chapter 4 describes the methods used to develop kRISP-meR. In Chapter 5, we compare kRISP-meR with other gRNA designing tools that use a reference genome. Finally, Chapter 6 concludes the thesis by mentioning future directions of the work.

Chapter 2

Literature review

In this chapter, we review the research undertaken to design guide RNAs for CRISPR/Cas9. In order to fully introduce the context, we first study the research works on CRISPR/Cas9 technology itself (Section 2.1), even though these works are more inclined towards biology and biochemistry rather than computation. Later, we introduce the complications in designing gRNAs for CRISPR/Cas9 (Sections 2.2 and 2.3) and recognize the research conducted to design gRNAs for CRISPR/Cas9 (Section 2.4). Finally, we identify the gaps in the state of the art methods to design gRNAs for CRISPR/Cas9 which lead towards our motivation to develop kRISP-meR (Section 2.5).

2.1 Introduction of CRISPR/Cas9 as a genome editing technology

CRISPR (clustered regularly interspaced short palindromic repeats) is a family of DNA sequences found in the genomes of prokaryotic organisms such as bacteria and archaea [42]. These sequences are derived from DNA fragments of bacteriophages that had previously infected the prokaryote. They are used to detect and destroy DNA from similar bacteriophages during subsequent infections. Hence these sequences play a key role in the antiviral (i.e. anti-phage) defense system of prokaryotes [42].

Cas9 is an enzyme that uses CRISPR sequences as a guide to recognize and cleave specific strands of DNA that are complementary to the CRISPR sequence. Cas9 enzymes together with CRISPR sequences form the basis of a technology known as CRISPR-Cas9 that can be used to edit genes within organisms [43]. This editing process has a wide variety of applications including basic biological research, development of biotechnology products, and treatment of diseases [44, 45].

The CRISPR-Cas system is a prokaryotic immune system that confers resistance to foreign genetic elements such as those present within plasmids and phages [46, 47] that provides a form of acquired immunity. RNA harboring the spacer sequence helps Cas (CRISPR-associated) proteins recognize and cut foreign pathogenic DNA. Other RNA-guided Cas proteins cut foreign RNA [48]. CRISPR are found in approximately 50% of sequenced bacterial genomes and nearly 90% of sequenced archaea [49].

In 2012, Jennifer Doudna and Emmanuelle Charpentier were the first to propose that CRISPR-Cas9 (enzymes from bacteria that control microbial immunity) could be used for programmable editing of genomes [12, 13]. Ever since, CRISPR/Cas9 has been one of the most notable technologies in field of genetic engineering looming in the limelight for these past few years. Genome editing technologies such as ZFNs are seldom completely specific, and some may cause a toxic reaction. However, the toxicity has been reported to be reduced by modifications done on the cleavage domain of the ZFN [50]. TALEN, on the other hand, has low on-target cutting efficiency. Both of these require time and labor to design a genome editing experiment. On the other hand, CRISPR is highly efficient and highly specific at the same time, and is able to target multiple target-sites simultaneously [51].

Because of the ease of use and cost-efficiency of CRISPR, extensive research is currently being done on it. There are now more publications on CRISPR than ZFN and TALEN despite how recent the discovery of CRISPR is [42]. Science selected this to be the 2015 Breakthrough of the Year [52].

2.1.1 CRISPR/Cas system

CRISPR (clustered regularly interspaced short palindromic repeats) is a family of DNA sequences in bacteria and archaea that contain snippets of DNA from viruses that have attacked the bacterium [42]. When a virus attacks a bacterium, snippets from the genome of the attacking virus is copied into the genome of the bacterium. This is done with the help of CAS enzyme. Later, these snippets act like a memory. Using this memory, the bacterium is able to construct CRISPR-RNA (crRNA) containing those snippets.

Cas9 (or “CRISPR-associated protein 9”) is an enzyme that uses CRISPR sequences as a guide to recognize and cleave specific strands of DNA that are complementary to the CRISPR sequence. Cas9 enzymes together with CRISPR sequences form the basis of a technology known as CRISPR-Cas9 that can be used to edit genes within organisms [43]. The crRNA containing memory snippets and CAS enzyme together form CAS-crRNA complex. This complex attacks the viruses that later attack the

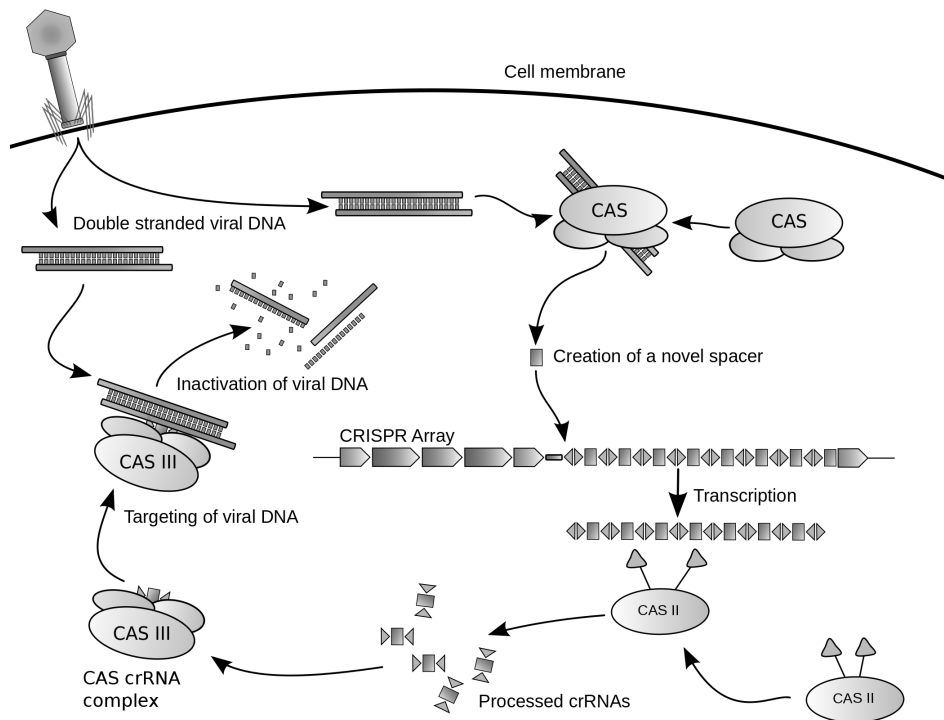


Figure 2.1: CRISPR/CAS defence mechanism (This image was taken from [2])

host bacterium. Since this complex has the crRNA formed earlier (which contains the snippets from the attacking virus), using the CAS enzyme, this complex can bind itself at certain positions of the genome of the attacking virus. After that, the genome of the virus is cut at the position where the complex is bound. Therefore, the genome of the attacking virus is shredded into disintegrated pieces which are no longer a threat to the host bacterium. This entire process is known as CRISPR prokaryotic antiviral defense mechanism [2], which is shown in Figure 2.1.

2.1.2 CRISPR/Cas9 system and the guide RNA

A simple version of the CRISPR/Cas system, CRISPR/Cas9, has been modified to edit genomes. CRISPR/Cas genome editing techniques have many potential applications, including medicine and crop seed enhancement. The use of CRISPR/Cas9-gRNA complex for genome editing [53] was the AAAS's choice for breakthrough of the year in 2015 [54]. Bioethical concerns have been raised about the prospect of using CRISPR for germline editing [55].

By delivering the Cas9 nuclease complexed with a synthetic guide RNA (gRNA) into a cell, the cell's genome can be cut at a desired location, allowing existing genes to be removed and/or new ones added. The Cas9-gRNA complex corresponds with the

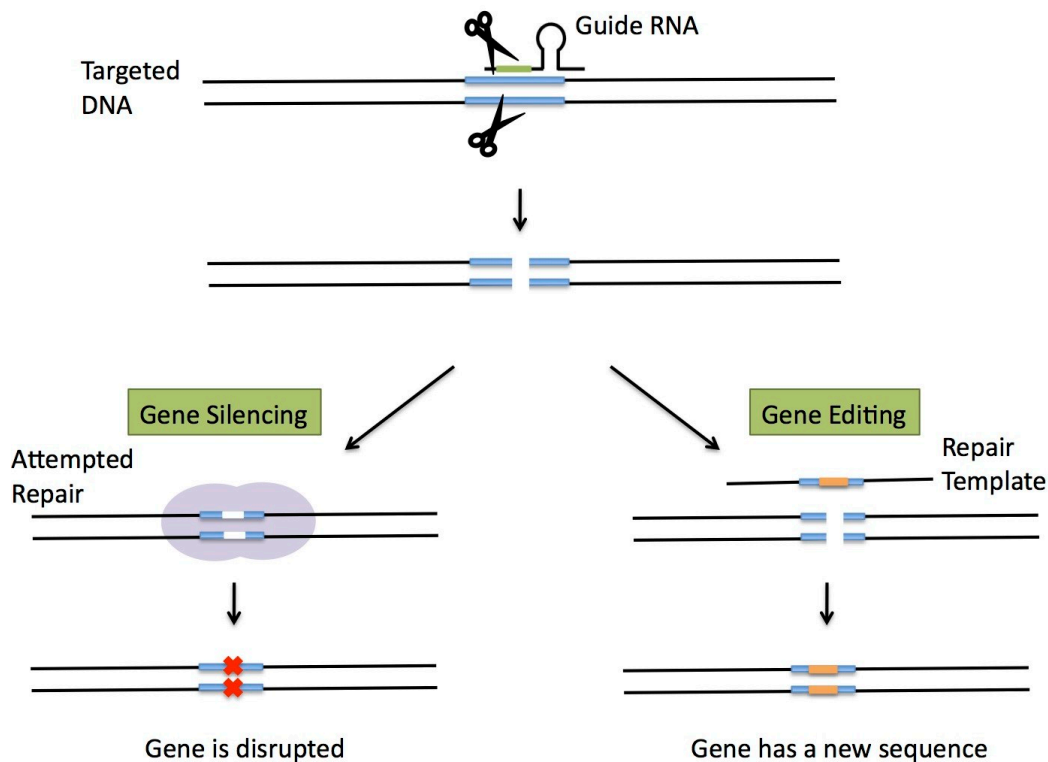


Figure 2.2: Editing the genome after double-strand break has been introduced by the Cas9 enzyme (This image was taken from [3]).

CAS-III crRNA complex in Figure 2.1. The guide-RNA is the determinant of where the CAS enzyme will make its cuts in the cell's genome. Thereby, the Cas9 nuclease is guided by the gRNA, and hence, engineering the gRNA properly so that the cut is made in the intended target region and not off-target, is crucial.

Once the Cas9 enzyme cuts both strands of the targeted DNA, this DSB can be repaired in two ways, as shown in Figure 2.2. During gene silencing, the cell attempts to repair the broken DNA in a process called Non-Homologous End Joining (NHEJ), but often does so with errors that disrupt the gene and effectively silencing it. For gene editing, a repair template with a specified change in sequence is added to the cell and incorporated into the DNA during the repair process. The targeted DNA is now altered to carry this new sequence [3].

2.2 gRNA and success of CRISPR/Cas9 experiments

CRISPR enables rapid, targeted, genome editing in a wide variety of systems [13]. The CRISPR-associated Cas9 protein, an RNA-directed DNA endonuclease, can be used to manipulate the genome effectively. A number of works have been undertaken to demonstrate this effectiveness of Cas9. Jinek et. al. [12] first showed that Cas9, guided

by CRISPR-RNAs (crRNAs) can introduce ds-breaks in the DNA. Later, Cong et al. [56] also showed that Cas9 nucleases can be directed by short RNAs to induce precise cleavage at endogenous genomic loci in human and mouse cells. Cas9 was also used in CRISPR to target and edit 293T cells, K562 cells and pluripotent stem cells [57]. These works, establishing the successful use of Cas9 in CRISPR experiments – all share the same aspect: use of guide-RNAs to guide Cas9 nuclease complex to intended genomic regions.

The ability to target a specific genomic location by designing guide-RNAs has made CRISPR a robust, low-cost method – opening up the scope of many applications [45]. CRISPR based genetic screens have been developed using the CRISPR/Cas9 system [58], which in turn led to identification of multiple gene-function for various phenotypes [59–61]. Scientists have successfully removed HIV virus out of the cells of infected rats [62]. CRISPR/Cas9 genome editing system was used to precisely remove the HIV-1 genome spanning from human T-lymphoid cells [63]. Treatment of primary open-angle glaucoma was successfully achieved in mouse eyes using CRISPR [64]. CRISPR/Cas9 was successfully used to distort sex-ratio in *Anopheles gambiae* [65], reducing number of females in the population and thus controlling the growth. Human T cells were modified using CRISPR for the treatment of X-Linked Hyper-IgM Syndrome, a condition that affects the immune system [66]. Besides treatments, CRISPR/Cas9 has also been used successfully in agriculture to enhance crops including rice, wheat, maize, potato, tomato, orange, cucumber and mushroom [23, 35, 36].

2.3 The gRNA designing problem

A protospacer adjacent motif (PAM) is a 2–6-base pair DNA sequence immediately following the DNA sequence targeted by the Cas9 nuclease in the CRISPR bacterial adaptive immune system [67]. In the CRISPR/Cas9 system, the Cas9 protein is guided by the sgRNA but will not be able to make an effective cut if the DNA-sequence (where the Cas9 protein is to make a cut) is not followed by a PAM sequence [12, 67–69]. Usually, for CRISPR/Cas9, the PAM with most effective outcome is 5'-'NGG' [70].

When a guide RNA binds to a genomic site unintentionally and makes a cleavage (or a cut), resulting in a double-strand break (or a ds-break), the effect is known as off-target effect. Due to error in design (not considering indels and SNPs) and/or experimental conditions, the ds-breaks can result at an undesired location [71, 72]. Off-target effects are widely studied whenever using a genome editing technology (including CRISPR) [73–75].

Key to the success of CRISPR/Cas9 in modifying the genome is the design of a

“good” guide-RNA; which is, therefore, an important research problem that has received notable attention – requiring careful use of high-quality genome sequence and gene annotations [76]. The primary goal of the design of an sgRNA is to make sure that the Cas9 makes a cut only at the intended target region and nowhere else, minimizing the off-target effect, which involves identifying sequences in the genome where a guide RNA can bind to and Cas9 can make a cut.

Besides this, it is also important to design a guide RNA that completes making a cut and does not stop midway. This is referred to in the literature as “on-target activity” or “on-target cutting efficiency”. Thus, designing guide RNAs involve looking for gRNAs with high specificity (minimizing off-target effects) and high cutting-efficiency (maximizing on-target activity).

2.4 Current state of gRNA dsigning problem

To design sgRNAs with high efficacy and high specificity, several models and computational tools have been developed. We are going to look into them in more detail in the following sections.

2.4.1 Models to predict on-target cutting efficiency

Ideally, if the 5'-end of the 20-nt sequence is complementary to the DNA sequence, then the cutting enzyme should make a cut. However, practically, this is not always the case [12,56]. Therefore, models predicting target sites with high efficiency are necessary. The first attempt to develop such a predictive model was undertaken by Doench et. al. [77]. This model involves analyzing all possible target sites for six mouse genes and three human genes, including 1841 sgRNAs. In their model, they employed a support vector machine (SVM) model to choose subsets of features from 586 features [77]. The features that have been filtered by the SVM classifier were used to predict the on-target efficiency of an sgRNA. Their efficiency score is in the range of [0,1]; 1 being the most efficient. This model was used to develop sgRNA Designer [77]. Being developed based on experimental data, this tool is highly reliable to identify highly efficient sgRNAs. Later, CRISPRpred outperformed this model, which also resorts to learning algorithms to recognize highly efficient guide RNAs [78].

Later, in 2016, Doench et. al. improved the model and proposed Rule Set 2.0 scoring algorithm [41]. Several new features were incorporated in this new scoring scheme, such as counts of position-independent nucleotide, location of target site in the corresponding gene, and melting temperature.

Xu et. al. analyzed CRISPR screen data and analyzed for features that affect sgRNA cutting efficiency [79]. Their use of Elastic Net [80] reveals new features, including preference for cytosine. Moreno-Mateos et. al. found that guanine enrichment and adenine depletion contribute to the efficiency of sgRNA by an analysis of target sites of 128 genes [81]. Their model was integrated in their tool CRISPRScan [81]. Chari et. al. developed an in vivo method to evaluate sgRNA activity across thousands of genes [82]. Wong et. al. later identified some new features from the experimental data published by Doench et. al. [77], and used SVM to develop an sgRNA potency prediction model [83]. Kuan et. al. used Elastic-Net algorithm in the Huang Laboratory to predict sgRNA efficiency [84]. Labuhn et. al. analyzed knock-out assays to reveal nine additional features affecting sgRNA efficacy [85].

2.4.2 Models to identify off-target cuts

Hsu et. al. evaluated more than 700 sgRNA variants and investigated the features that affect sgRNA target specificity [86]. They found that the mismatch tolerance between the sgRNA and the DNA target site is influenced by the number, position and distribution of mismatches [87]. They used their analyses to summarize the effect of mismatch position into a single penalty matrix. Based on this penalty score, each sgRNA can be assigned a specificity score by iterating over all potential off-target sites. The model developed by Hsu et. al. [86] was very slow [87]. Therefore, Stemmer et. al. later developed CCTop to address those issues [88], which uses alignment to filter sgRNAs with high specificity. The off-target score defined in CCTop is as follows.

$$\text{Score} = \sum_{\text{off target}} \left(\frac{\log \text{dist} + \text{score}_{\text{off-target}}}{\text{total-off-targets}} \right) - \text{total-off-targets} \times \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Here, dist means the distance between this off-target site and the closest exon, $\text{score}_{\text{off-target}}$ is calculated by the relative position of mismatches in PAM, and total-off-targets means the total number of candidate off-target sites.

Later, Doench et. al. analyzed pooled screening data, performed extensive experiments to profile mismatch conditions between sgRNA and DNA, including alternative PAMs, mismatches, deletions and insertions [41]. Based on their findings, the authors proposed a scoring algorithm known as CFD scoring, and showed that this outperformed the models developed by Hsu et. al. [86] and Stemmer et. al. [88].

Mendoza et. al. identified that these scoring methods are not suitable for using across organisms and proposed their own novel algorithm to determine off-target effects known as CASPER [89]. Many of the tools also developed their own scoring scheme, including

WU-CRISPR [83], CROPT-IT [90] and E-CRISP [91].

2.4.3 Computational tools using the models

A number of publicly available methods and software have been developed to design guide RNAs with a view to employ the use of these software directly in wet labs. Popular among these tools include CCTop [88], CHOPCHOP [92], COSMID [93], CRISPRdirect [94], Crispr Finder [95], Crispr-P [96], CRISPRseek [97], E-crisp [91], sgRNAscas9 [98], Cas-OFFinder [99]. These tools vary in their fundamental use of the off-target or on-target model, and in their functionalities (batch processing, available PAM sequences, availability of web servers, ability to support alternate scoring schemes etc.) [87,100]. Recent efforts are focusing on developing machine-learning based models for on and/or off-target prediction [101] and tools that directly predict guides using machine-learning [102].

CRISPRScan [81] is one of the popular guide RNA designing tools that was developed after extensive wet-lab experiments. From the experiments carried out in various conditions, guide RNA activity was recorded. Then, various features were extracted from the guide RNA sequences and a regression tree was developed from the recorded gRNA activity data, which is used to predict the specificity of a guide RNA. CRISPRScan experiments revealed that guanine enrichment and adenine depletion increases sgRNA activity. Therefore, CRISPRScan has the tendency to drop sgRNAs with high adenine content.

One of the more recently developed tools with promising results are GuideScan [103] and MultiGuideScan [104] (which is a parallel and faster version of GuideScan). GuideScan, like many other tools, scans the target region for PAMs and identifies the potential 20-nt gRNA sequences. Next, together with the NGG PAM, these 23-nt sequences are annotated with a specificity score and then ranked using that score. More specifically, for a given gRNA, GuideScan enumerates all its neighbors up to Q mismatches, calculates the CFD score for each neighbor using the schemes developed by Doench et. al. [41], and then multiplies that score by the number of times the neighbor occurs in the genome. It then aggregates the CFD values into a single composite score utilizing the formula used by Hsu et. al. [86].

$$\text{Specificity score}_{\text{GuideScan}} = \frac{1}{\sum \text{CFD}_i \times q_i}$$

Here, n represents the number of unique targetable sites within up to a certain number of mismatches. The GuideScan specificity score for a gRNA that generates no off-target cut (considering 0 mismatches) will be 1.0, since the CFD score as well as the number

of occurrences will both be 1.0.

This CFD score proposed by Doench et. al. [41] largely profiles the in vivo experiments involving CRISPR experiments. From these experiments, data of gRNA activity was recorded which was expressed in a matrix form that gives the probability of making a cut for a certain neocleotide of the gRNA aligned with a neocleotide of the target sequence. Scanning over all neocleotide positions of the target sequence and the guide RNA, a CFD score can be calculated. Hsu et. al. [86], on the other hand, extensively profiled the *Streptococcus pyogenes* Cas9 (spCas9) activity in wet-lab experiments and recorded the specificity in various conditions. They expressed specificity in many forms and finally proposed a specificity score to be used in their computational tool to facilitate the selection and validation of sgRNAs (available at <http://www.genome-engineering.org/>). The specificity score developed and proposed by Hsu et. al. [86] has been used to develop CRISPOR [105]. Along with the Hsu specificity score, CRISPOR also generates some other scores and annotates the guide RNAs with those scores.

2.5 Motivation of the work

All these mentioned tools are common in one aspect: that they all require the reference genome to design a guide. The design of sgRNAs is, in fact, a complex optimization problem. The multitude of sequences comprising a candidate gene for targeting can be used as targets for guides. However, ideally guide targets are unique in the genome to reduce off-target binding. Thus, effective sgRNA design requires knowing all the sequences of the genome to be targeted. This is why the long list of tools mentioned before all use the reference genome to start with. Unfortunately, most genomes are not completely sequenced and assembled yet. As a result, designing guide RNAs for many organisms, especially non-model organisms, is difficult if we are restricted within the reference based tools. Therefore, it is important to develop a method that is capable of designing guide RNAs without relying on the reference. Recent works have started to focus on reference-free methods to design guide RNAs. Sun et al. presented a tool to design sgRNAs for non-reference plant genomes [106].

Another key factor is that even if we have a reference genome assembled for an organism, the reference seldom represents all individuals accurately. Recent study shows that human genome does not represent all sequences present in individuals [107, 108]. Thus, it is also important to address the issue of individual specific genome variation and design “personalized guides”. This has been partly addressed by a few tools, namely AlleleAnalyzer [109], CrisFlash [110] and CRISPRitz [111]. However, these tools still use reference genomes and sequencing reads to detect variation and may lack specificity

if the individual contains duplications or regions missing from the reference. Synthetic sex-ratio distortion using CRISPR was achieved by Papathanos et al. [19] using a tool that does not require a reference genome, yet the method is not generally applicable for general sgRNA design.

All in all, in order to design guide RNAs for organisms with incomplete reference genome, and for individuals with genomic variations with the reference, it is important to develop a gRNA designing tool that decouples the use of a preassembled reference genome.

Chapter 3

Preliminaries

In this chapter, we introduce the concepts that are later used in Chapter 4 to develop the methods used in kRISP-meR. As the primary purpose of developing kRISP-meR is to not use the reference, kRISP-meR relies on the information available in the sequenced reads. Therefore, a lot of the following concepts relate to a set of sequenced reads.

3.1 Sequenced reads in DNA sequencing

In a typical DNA sequencing experiment, a genome is output as fragmentation of many molecules [112]. After the end of sequencing process, a sequence of base pairs corresponding to all or part of a single DNA fragment is obtained which can be represented as a string over the alphabet $\Sigma = \{‘A’, ‘C’, ‘G’, ‘T’\}$. These “reads” are stored in text-based FASTQ/FASTA file format.

A set of sequenced reads can contain characters other than the alphabet mentioned above. Often, the reads contain the character ‘N’, indicating any of the valid characters may take a place there, and the particular read does not know which for sure.

Length of the sequenced reads can affect biological experiments [113]. Therefore, it is important for bioinformatics pipelines to consider the length of sequenced reads [114]. kRISP-meR is designed and developed considering short read libraries.

3.2 k -mer

k -mers are substrings of length k contained within a biological sequence [115]. Primarily used within the context of computational genomics and sequence analysis, in which k -mers are composed of nucleotides (i.e. A, T, G, and C). An example k -mer is shown in Figure 3.1.



Figure 3.1: k -mer example. The sequence ATGG has two 3-mers: ATG and TGG.

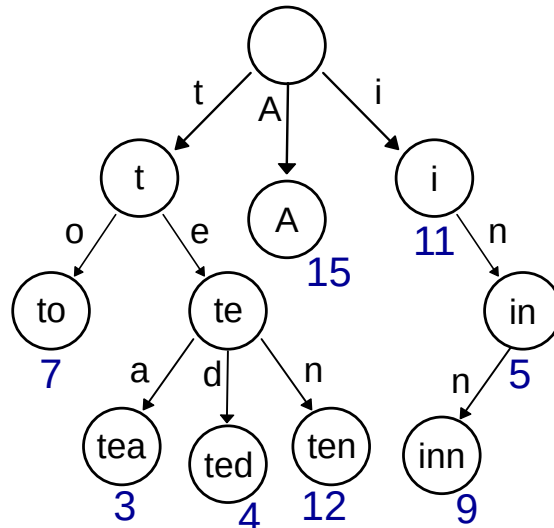


Figure 3.2: An example trie.

In general, a sequence of length L will have $L - k + 1$ k -mers and n^k total possible k -mers, where n is number of possible monomers (four in the case of DNA) [115].

3.3 Prefix tree: trie

A trie, also called digital tree or prefix tree, is a kind of search tree — an ordered tree data structure used to store a dynamic set or associative array where the keys are usually strings [116]. Unlike a binary search tree, no node in the tree stores the key associated with that node; instead, its position in the tree defines the key with which it is associated. All the descendants of a node have a common prefix of the string associated with that node, and the root is associated with the empty string [117].

An example of a trie is shown in Figure Figure 3.2 for keys “A”, “to”, “tea”, “ted”, “ten”, “i”, “in”, and “inn”. Note that this example does not have all the children alphabetically sorted from left to right as it should be (the root and node ‘t’). The image was taken from [117].

Using a trie has the advantage of minimizing search time at the expense of higher storage requirements. If we store keys in binary search tree, a well balanced BST will

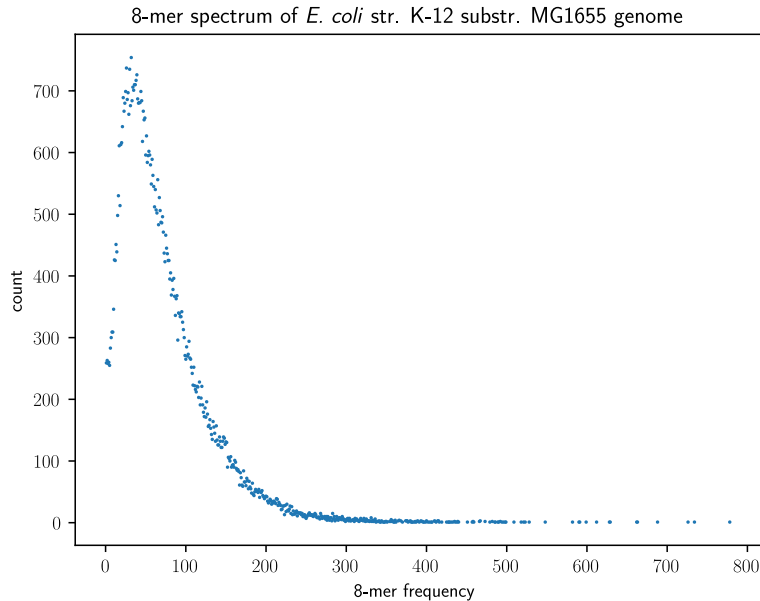


Figure 3.3: An example 8-mer spectrum for *Escherichia coli* comparing 8-mers' frequency with their number of occurrences.

need time proportional to $M * \log N$, where M is maximum string length and N is number of keys in tree. Using Trie, we can search the key in $O(M)$ time [118].

Using a trie for storing genomic sequences has been particularly popular in literature. In a trie, any node can have upto $|\Sigma|$ children, where Σ is the alphabet. For the genomic sequences, $\Sigma = \{'A', 'C', 'G', 'T'\}$. Therefore, even though the storage requirement is higher, the maximum number of children one node can have is small. Besides that, in kRISP-meR, we store sequences of the same length as a guide RNA in a trie. For CRISPR/Cas9 system, this length is always 23. Therefore, at the expense of higher space requirement, kRISP-meR is able to retrieve these sequences in constant time.

3.4 k -mer counting in sequenced reads: k -mer spectrum

k -mer abundance data is a useful information in genomic data analysis. This data is used to estimate parameters for genome assembly. In k -mer abundance histogram, each data point is a value f_i which is the number of distinct k -mers that appears i times in sequencing data. In order to compute k -mer abundance, one popular approach is to use k -mer counting tools like Jellyfish, DSK, KMC2, Squeakr, Tallymer, Khmer etc. An example k -mer spectrum is shown in Figure 3.3, which was generated using the k -mer counts found in a genome.

Repeating the same experiment for the sequencing reads, not only the genome would

generate a similar plot, with a very sharp peak near zero. This is because when sequencing technologies generate sequenced reads, they almost always outputs erroneous bases. Consequently, counting all k -mers in a set of sequenced reads will result in a lot of sequences that appear only once or twice, but do not appear in the genome. This, therefore, results in a very high peak in the histogram data near zero. The next peak appears at the value “coverage”, which is approximates the number of times each base in the genome is covered by the sequencing technology. For example, let us assume that a sequencing technology has coverage λ while sequencing a certain genome. Then, generating a k -spectrum from the sequenced reads would result in an abundance plot with a peak at λ .

3.5 Maximum Likelihood Estimation

Maximum likelihood estimation (MLE) is a method of estimating the parameters of a probability distribution by maximizing a likelihood function, so that under the assumed statistical model the observed data is most probable [119]. The point in the parameter space that maximizes the likelihood function is called the maximum likelihood estimate. If the likelihood function is differentiable, the derivative test for determining maxima can be applied. However, in practical circumstances, numerical methods are often used to find the maximum of the likelihood function as the calculation of the derivatives may be analytically difficult, and even impossible to perform.

If we are to determine the parameter θ that explains a set of observation $O_1, O_2, O_3, \dots, O_n$, then MLE defines likelihood as follows:

$$L_n(\theta) = \prod_{i=1}^n P(O_i|\theta)$$

The goal of maximum likelihood estimation is to determine the θ in the parameter space Θ that maximizes this likelihood.

$$\hat{\theta} = \arg \max_{\theta \in \Theta} \hat{L}_n(\theta)$$

Most of the times, the product of the sums underflow in a computer program. Therefore, it is common to use log-likelihood and maximize that.

$$\ell(\theta) = \ln L_n(\theta) = \sum_{i=1}^n \ln P(O_i|\theta)$$

3.6 Expectation Maximization algorithm

The Expectation-Maximization (EM) algorithm is a way to find maximum-likelihood estimates for model parameters when data is incomplete, has missing data points, or has unobserved latent variables. It is an iterative way to approximate the maximum likelihood function. While maximum likelihood estimation can find the best fit model for a set of data, it does not work particularly well for incomplete data sets. The more complex EM algorithm can find model parameters even if there is missing data. It works by choosing random values for the missing data points, and using those guesses to estimate a second set of data. The new values are used to create a better guess for the first set, and the process continues until the algorithm converges on a fixed point [120]. Although Maximum Likelihood Estimation (MLE) and EM can both find best-fit parameters, how they find the models are very different. MLE accumulates all of the data first and then uses that data to construct the most likely model. EM takes a guess at the parameters first — accounting for the missing data — then tweaks the model to fit the guesses and the observed data.

An initial guess is made for the model's parameters and a probability distribution is created. This is the E-Step for the expected distribution. Newly observed data is then fed into the model. The probability distribution from the E-step is tweaked to include the new data. This is called the M-step. These steps are repeated until stability is reached.

The EM Algorithm always improves a parameter's estimation through this multi-step process. However, it sometimes needs a few random starts to find the best model because the algorithm can hone in on a local maxima that is not that close to the (optimal) global maxima. In other words, it can perform better if it is forced to restart and take that initial guess from Step 1 over again.

The EM algorithm can be very slow, even on the fastest computer. It works best when you only have a small percentage of missing data and the dimensionality of the data is not too big. The higher the dimensionality, the slower the E-step; for data with larger dimensionality, it is possible that the E-step runs extremely slow as the procedure approaches a local maximum.

3.7 The Poisson distribution to model sequenced reads

The Poisson distribution is popular for modeling the number of times an event occurs in an interval of time or space [121]. A discrete random variable X is said to have a Poisson distribution with parameter $\lambda > 0$, if, for $k = 0, 1, 2, \dots$, the probability mass

function of X is given by:

$$f(k; \lambda) = \Pr(X = k) = \frac{\lambda^k e^{-\lambda}}{k!}$$

The k -mer spectrum of a set of sequenced reads can be modeled using a set of Poisson distributions. The idea behind using a Poisson distribution is that, Poisson is a special case of a Binomial distribution. Given that the parameters of a Binomial distribution are n and p , where n is the number of trials and p is the probability of success in each trial; if $n \rightarrow \infty$ and $p \rightarrow 0$, the Binomial appears as a Poisson distribution with mean np . Likewise, when we are counting k -mers in a set of sequenced reads, one particular sequence showing up during the counting is a very low probability event (considering a sequence of a considerable length). Yet, with a huge number of base-pair positions in the genome, we do observe the sequence a number of times. Consequently, trying to find a certain k -mer in millions of base-pair positions in a genome, and therefore sequenced reads, can be modeled using Poisson distribution.

The use of a mixture of multiple Poisson distributions is more intuitive. A genome can be thought of as a sequence of repetitive regions. Some regions repeat more than one time, others appear only once in the genome. Therefore, when a DNA sequencing technology attempts to sequence a genome with read-coverage λ and generates λ number of sequencing reads on average covering every base position – the sequenced reads generated from a genomic region that appears i times can be modeled using a Poisson distribution with mean λi . When a k -mer appears in the read set due to the presence of one or more copies of the sequence in the genome, Poisson distributions have been observed to model the counts well in genome sequencing data [122].

3.8 EM in mixture of Poisson distributions

The EM algorithm finds maximum-likelihood estimates for model parameters when we have incomplete data. The “E-Step” finds probabilities for the assignment of data points, based on a set of hypothesized probability density functions; The “M-Step” updates the original hypothesis with new data. The cycle repeats until the parameters stabilize [4]. The process is well accepted for continuous cases such as a mixture of Gaussian distributions [123, 124]. A sketch of the approach to determine the parameters of Gaussian Mixture Models (GMMs) is shown in Figure 3.4 [4].

Similarly, parameters of a mixture of Poisson distributions can also be determined for the discrete case using the EM algorithm. EM has been used for discrete case of Gaussian in literature [125] and therefore, is a good choice to determine the parameters

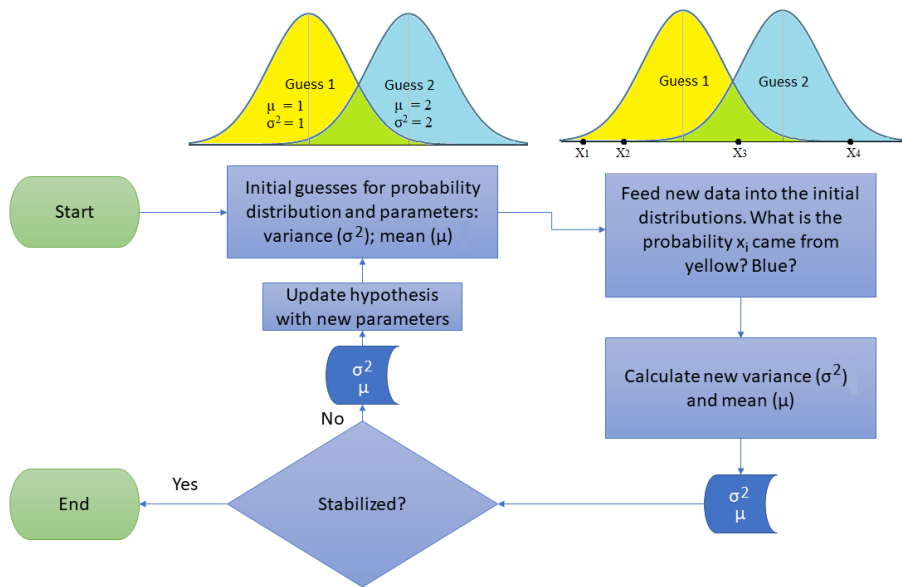


Figure 3.4: Flow of EM algorithm in determining the parameters of a very simple Gaussian Mixture Model (This image was taken from [4]).

of a mixture of Poisson distributions. The steps in the process are similar to that shown in Figure 3.4, only done for a discrete distribution with only the location parameter.

3.9 Bayes decomposition: priors and posteriors

In probability theory, the probability of an event B occurring from the precondition that an event A has triggered event B is given by $P(A|B)$, and can be elaborated using the Bayes formula as follows.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Here, the event A is one of the many preconditions that can result in triggering the event B . It is usually difficult to determine the probability of a cause from the effect. The Bayes formula solves this by reversing the role of the events. After breaking the probability formula using the Bayes rule, we are left to determine the probability of an effect from a cause, which is easier in specific application scenarios.

Additionally, the $P(A)$ and $P(B)$ parts are the prior and posterior probabilities respectively. From prior knowledge, we know the probability of the many triggering causes. The posterior is usually calculated using the evidence observed.

Chapter 4

Methods

This chapter describes the mathematics applied in kRISP-meR to list the guide RNAs. Figure 4.1 shows an overview of the methods. kRISP-meR takes as input a target sequence and sequenced reads from the organism. It starts by aligning the sequenced reads to the target sequence to detect sample-specific variations. Then, the candidate guide RNAs are identified from the personalized target sequence. Next, scores to indicate efficiency and specificity of each candidate are determined using the copy-number of k-mers in the sequences reads with the use of probabilistic models.

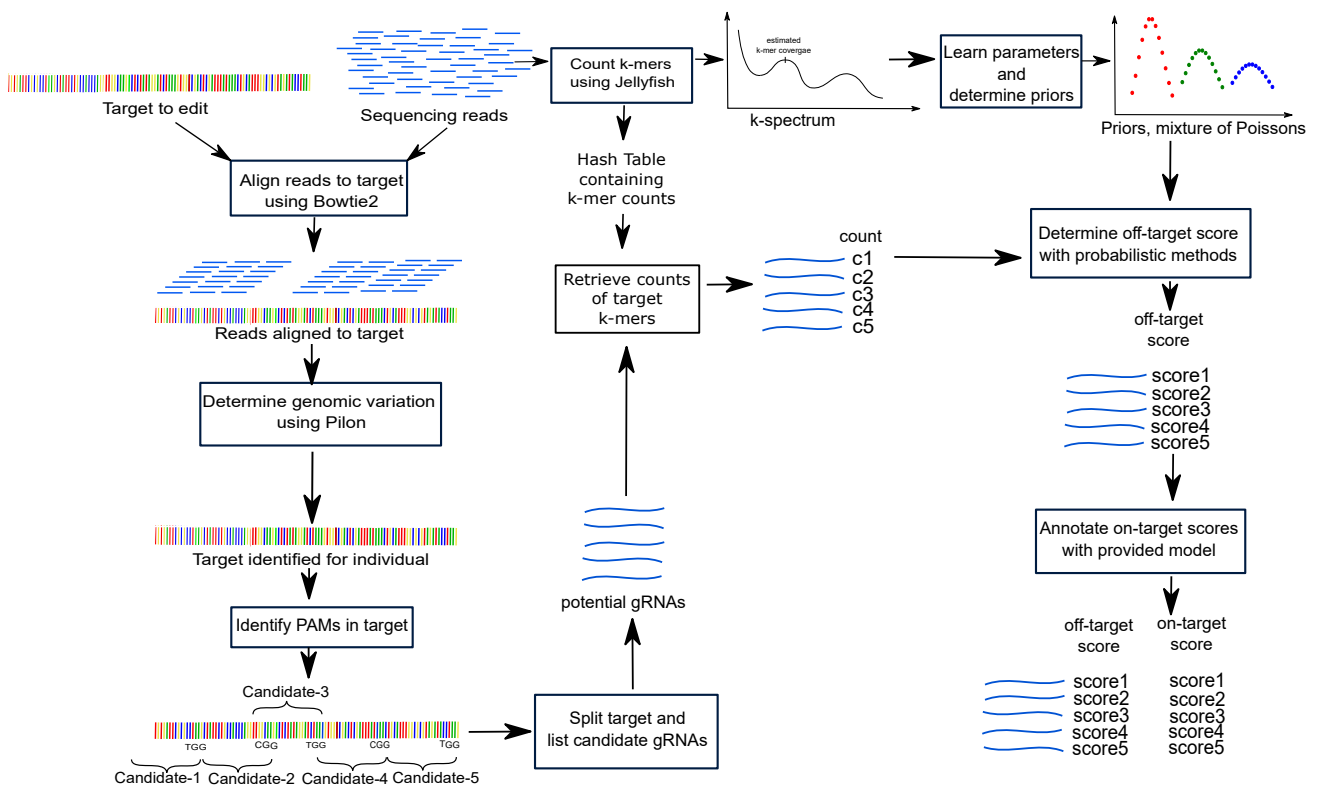


Figure 4.1: Overview of kRISP-meR.

4.1 Overview

The steps of kRISP-meR are shown in Figure 4.1 briefly. It takes as input a set of sequenced reads from the genome of the individual to be edited and a target sequence that corresponds to the region in the genome where CRISPR cleavage is intended. The target sequence can come from a pre-assembled reference genome of the same organism or a similar organism. Bowtie2 and Pilon are used to customize the target sequence personalized for the individual. After that, the personalized target sequence is scanned for the protospacer adjacent motif (PAM) sequences. Next, the target sequence is split at the PAMs, and candidate guide RNAs are listed. These candidate gRNAs are then scored and sorted, which heavily engages k-mer counting in the sequencing reads. These scores are finally used to drop some of these guide RNAs, and then sort and output the list of remaining ones.

4.2 Computing personalized target sequence

The target region is taken as a snap from the reference of a similar genome, chromosome, or an exon. As the the individual may have genetic variations in the target region, it is important to first determine the target sequence for the individual. This is done using sequenced reads of the individual as follows:

- **Aligning sequenced reads to the target:** The sequenced reads are aligned to the input target sequence using Bowtie2 [38], which aligns the sequencing reads to the target sequence. In doing so, the reads that fall below threshold score are discarded. The sequenced reads aligned to the target are recorded in a sorted SAM file.
- **Converting to binary file:** SAM file format is a text file. In the next step, Samtools [39] is then used to convert the SAM file into a BAM file, the corresponding binary counterpart. Then, the BAM file is sorted and indexed using Samtools once more.
- **Polishing the target sequence:** The final BAM file, which contains the sorted and aligned sequenced reads, along with the input target sequence are processed by Pilon [40]. Pilon attempts to better the input target sequence using the aligned sequenced reads. The output generated by Pilon is the target sequence personalized for the individual, and thus the target sequence is polished to generate a personalized target sequence.

This personalized target sequence is then scanned for candidate sgRNAs.

4.3 Identifying candidate gRNAs

From the target sequence, kRISP-meR determines the list of potential guide RNAs by looking for a short (3 nucleotides long for CRISPR/Cas9) protospacer adjacent motif (PAM) sequence. As introduced in Section 2.3, guide RNAs must be followed by this short PAM sequence to result in an effective cleavage (cut). The length of the guide is usually fixed for a particular genome editing technology (20 nucleotides for CRISPR/Cas9).

Candidate guides are searched for in two ways:

1. Searching for PAM sequences directly: kRISP-meR finds candidate guide RNAs by locating the PAM sequences in the target.
 - (a) kRISP-meR searches for the PAM sequence in the target sequence in the 5'-3' direction. Wherever the PAM is located, kRISP-meR takes the preceding sequence consisting of 20 nts, and lists the 23 nts long sequence as a potential guide RNA. Technically, only the 20 nts long sequence is to be considered as a guide RNA. However, the entire 23 nts long sequence make an effective cleavage. Therefore, kRISP-meR does not distinguish the PAM separately.
 - (b) A quite similar search is done on the 3'-5' direction by computing the reverse complement of the target sequence.
2. Searching for PAM sequences considering mismatches: kRISP-meR also considers one mismatch in the PAM sequence and lists the corresponding 23 nts long sequence as a potential guide RNA. This is not as straightforward, as the models used for cutting frequency determination strictly require 'NGG' PAMs. Therefore, the sequences with one mismatch (computed from the substring of the target) are listed as the potential guide RNAs, marking the original 23-nts substring sequence as the target site where cleavage is expected.

In both cases, the strand information is stored. The mostly used PAM for CRISPR/Cas9 is NGG, and therefore, NGG is kept as default in kRISP-meR. However, kRISP-meR can be configured to work with alternate PAMs (must have GG in the 3' end) with the '-a' flag.

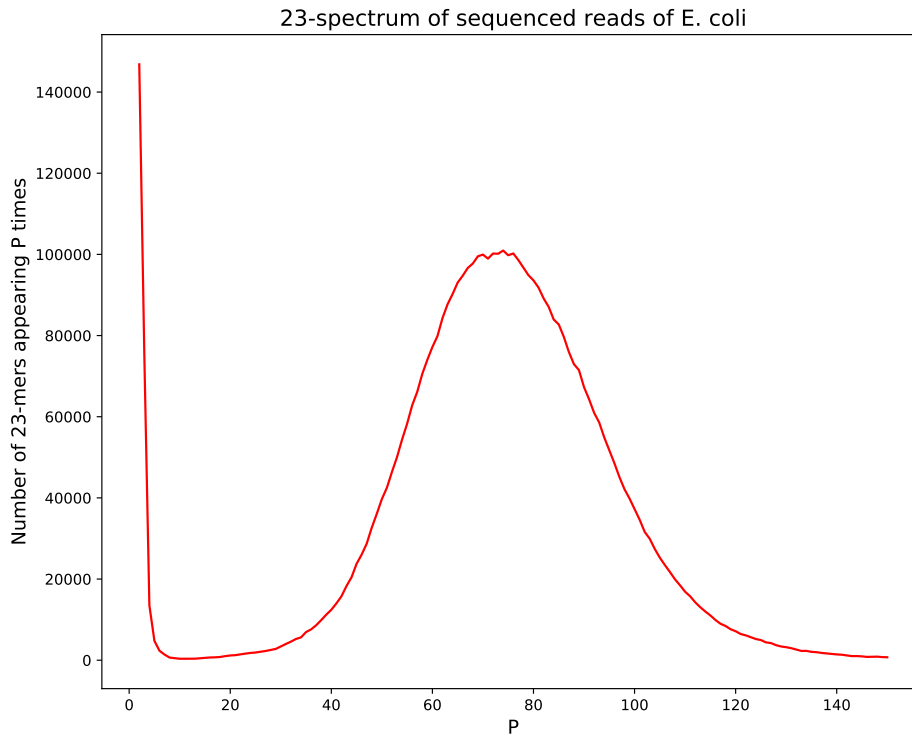


Figure 4.2: 23-spectrum of *E. coli* sequenced reads (read coverage: 74).

4.4 Counting k -mers in sequenced reads

4.4.1 Storing the k -mer counts

Since kRISP-meR is a reference free tool for guide RNA design, it cannot directly determine the number of times a sequence, where a guide RNA may bind to, is present in the genome. Instead, it first counts the number of times sequences of length 23 are present in the set of sequenced reads using Jellyfish 2 [37]. The counts are stored in a hash table as a binary file, which is later used to quickly retrieve the copy-number of sequences in sequenced reads. These counts are required to estimate the number of times a sequence is present in the reference, and in turn, to estimate the number of times a potential guide RNA can make a cut in genome. Detailed steps of this estimation are elaborated in Section 4.5.

4.4.2 Determining the k -spectrum of sequenced reads

The k -mer counting makes a way to determine the k -mer coverage λ of the sequencing technology, which approximates the read-coverage. The k -spectrum shown in Figure 4.1

is generated by counting all possible k -mers in the sequencing reads and determining the number of i -mers in the reads $H(i)$ for all i . A 23-spectrum of the sequenced reads collected from an experiment (NCBI SRA experiment ID: SRX4680200) is shown in Figure 4.2.

The first peak in the spectrum, appearing near zero, is due to large number of k -mers that arise due to sequencing errors and appear very few times. The next peak appears at the k -mer coverage λ , which is required in the steps to score the candidate gRNAs.

4.5 Scoring candidate gRNAs

kRISP-meR assigns a score to each guide RNA indicating its off-target activity. The major steps in determining and assigning this score are as follows:

- Determining expected number of cuts made by a guide RNA in the entire genome
- Determining expected number of cuts made by a guide RNA in the target site
- Determining “most likely” number of appearances of the target in reference
- Calculating the score using the former three

4.5.1 Determining expected number of cuts in the genome

For all the identified potential guide RNAs, kRISP-meR determines the expected number of cuts made by those: both in the whole genome and the target site.

A particular guide RNA, when deployed in CRISPR and is asked to guide the Cas9 enzyme, makes a cut where the exact match of the RNA (including PAM) is located. However, in addition to making a cut where it finds a perfect matching sequence in the genome, the guide RNA may make cuts at sites with number of mismatches as well, albeit with lower probabilities. If the set of all k -mers within a specified Hamming distance of the complementary sequence of the guide RNA is \mathbb{K} , we define the expected number of cuts in the genome as follows:

$$\text{Expected number of cuts in genome, } E[x_G] = \sum_{k \in \mathbb{K}} c_{G,k} \times \psi_k \quad (4.1)$$

where ψ_k is the cutting probability of the guide RNA at the k -mer site determined by an experimentally determined mutation matrix [41], and $c_{G,k}$ is the number of the times the k -mer is present in the genome.

Determining $c_{G,k}$, and thereby $E[x_G]$, allows us to accurately determine the behavior of a guide RNA when deployed in CRISPR. However, in the absence of a reference genome to work with, we have no direct way to calculate these. Therefore, we take the expectation and estimate $E[x_G]$ from the number of times it appears in the read set $c_{R,k}$ using the Bayes rule.

Therefore, we can express $c_{G,k}$, copy-number of a particular k -mer in genome using expectation as follows.

$$c_{G,k} = \sum_{c_{G,k}=1}^{\infty} P(c_{G,k}|c_{R,k}) \times c_{G,k} \quad (4.2)$$

Here, $P(c_{G,k}|c_{R,k})$ is the conditional probability of the k -mer appearing $c_{G,k}$ times in the genome, given that, this particular k -mer is present $c_{R,k}$ times in the sequenced reads.

The value $c_{R,k}$ is the evidence that we observe in the sequenced reads, which is the direct consequence of $c_{G,k}$, the number of times a particular k -mer is present in genome. Thus, we are to calculate the probability of the cause from the occurrence of a consequence, which is naturally complicated. This is a classic case of applying Bayes Theorem to reverse the role of the two, which solves this complication by introducing the priors and the posteriors.

After applying Bayes, we have:

$$P(c_{G,k}|c_{R,k}) = \frac{P(c_{R,k}|c_{G,k})P(c_{G,k})}{P(c_{R,k})} \quad (4.3)$$

Now that the two conditional-probability elements are ordered the “easier” way (probability of effect given a cause), we can use some modeling and calculate this probability. As discussed before in Section 3.7, Poisson distributions can be used to model a set of sequenced reads. If a sequenced read set has been obtained using a sequencing technology with coverage λ , and if a particular k -mer is present a times in the genome, then the number of times this particular k -mer appears in the sequenced reads is Poisson distributed with mean λa . Therefore, this conditional probability in Equation (4.2) can be written as follows:

$$P(c_{R,k}|c_{G,k}) = \frac{e^{(-\lambda c_{G,k})}(\lambda c_{G,k})^{c_{R,k}}}{(c_{R,k})!} \quad (4.4)$$

Where λ is the estimated coverage of the sequenced technology. Plugging everything back to Equation (4.2) and Equation (4.1), we have the following.

$$\begin{aligned}
E[x_G] &\approx \sum_{k \in \mathbb{K}} \left[\sum_{c_{G,k}=1}^{\infty} P(c_{G,k}|c_{R,k}) \times c_{G,k} \right] \times \psi_k \\
&= \sum_{k \in \mathbb{K}} \left[\sum_{c_{G,k}=1}^{\infty} \frac{P(c_{R,k}|c_{G,k})P(c_{G,k})}{P(c_{R,k})} \times c_{G,k} \right] \times \psi_k \\
&= \sum_{k \in \mathbb{K}} \left[\sum_{c_{G,k}=1}^{\infty} \frac{e^{(-\lambda c_{G,k})} (\lambda c_{G,k})^{c_{R,k}}}{(c_{R,k})!} \times \frac{P(c_{G,k})}{P(c_{R,k})} \times c_{G,k} \right] \times \psi_k
\end{aligned}$$

Reorganizing a little to avoid calculating the same thing repeatedly, we have the expected number of cuts made by a guide RNA, $E[x_G]$ as follows.

$$E[x_G] = \sum_{k \in \mathbb{K}} \left[\frac{\psi_k}{P(c_{R,k})(c_{R,k})!} \sum_{c_{G,k}=1}^{\infty} \left\{ e^{(-\lambda c_{G,k})} (\lambda c_{G,k})^{c_{R,k}} \times P(c_{G,k}) \times c_{G,k} \right\} \right] \quad (4.5)$$

Here, $P(c_{G,k})$ is the prior probability that there are $c_{G,k}$ copies of k in the genome; where k represents all k -mers in \mathbb{K} , whereas $P(c_{R,k})$ is the posterior probability that k appears $c_{R,k}$ times in the sequenced read set. Steps to calculate these are elaborated in Sections 4.5.5 and 4.5.6.

4.5.2 Determining expected number of cuts in the target

Similar to the expected number of cuts in the genome, we define the expected number of cuts in the target as follows:

$$\text{Expected number of cuts in target, } E[x_t] = \sum_{k \in \mathbb{K}} c_{t,k} \times \psi_k \quad (4.6)$$

Again, \mathbb{K} is the set of all k -mers within a specified Hamming distance of the complementary sequence of the gRNA which is being examined. The definition of ψ_k , the cutting probability of the guide RNA at the k -mer site also remains the same. Here, $c_{t,k}$ is the number of the times the k -mer is present in the target site. In this case, determining the number of expected cuts resulted from a particular guide RNA

is easier, because computation of $c_{t,k}$ is not so complicated as $c_{G,k}$. Therefore, without relying on the sequenced reads, we can simply count the number of appearances of a sequence in the target sequence.

4.5.3 Determining $C_{G,t}$: number of appearances of the target in reference

The k -spectrum introduced in Section 3.4 can be modeled using a mixture of Poisson distributions, with mean $\lambda \times i$ for $i = 1$ to above; where λ is the read-coverage. The use of a mixture of Poisson distributions to model a set of sequenced reads was introduced earlier in Section 3.7.

Now, if we restrict ourselves within the sequences in the target region while counting the k -mers, we would still get a similar k -spectrum – only a distorted view focusing the sequences found within the target region. Most of the copy numbers illustrated in the original k -spectrum comes from the Poisson with mean λ . On the other hand, most of the copy numbers in this distorted k -spectrum comes from the Poisson with mean $c_{G,t} \times \lambda$.

Therefore, we can calculate the value of $c_{G,t}$ maximizing likelihood. Let the copy numbers of the k -mers within the target sequence be k_1, k_2, k_3 and so on. Then, we define likelihood $L(i, j)$ as:

$$L(i, j) = P(\lambda i, k_j) \quad (4.7)$$

which is the likelihood of observing k_j from the Poisson with mean λi . Hence, the likelihood of the Poisson with mean λi is:

$$L(i) = \prod_{\text{all } j} P(\lambda i, k_j) \quad (4.8)$$

where $P(\lambda i, k_j)$ is the Poisson probability of observing k_j where the mean of the Poisson is λi . As discussed already in the previous section (while putting together Equation (4.4)), this probability is as follows.

$$P(\lambda i, k_j) = \frac{e^{-\lambda i} \times (\lambda i)^{k_j}}{(k_j)!} \quad (4.9)$$

The i that maximizes $L(i)$ is the estimated value of $c_{G,t}$. kRISP-meR works by computing the log-likelihoods to avoid underflow. By plugging in the values back to

Equation (4.8) and taking logarithm, we have log-likelihood of the Poisson with mean λi , $l(i)$ as follows:

$$l(i) = \sum_{\text{all } j} \ln \frac{e^{(-\lambda i)} \times (\lambda i)^{k_j}}{(k_j)!} \quad (4.10)$$

Elaborating this log-likelihood mathematically gives us the following:

$$\begin{aligned} l(i) &= \sum_j \ln e^{-\lambda i} + \sum_j \ln (\lambda i)^{k_j} - \sum_j \ln k_j \\ &= -\lambda i \sum_j 1 + \ln (\lambda i) \sum_j k_j - \sum_j \ln k_j \\ &= -\lambda i \sum_j 1 + \ln \lambda \sum_j k_j + \ln i \sum_j k_j - \sum_j \ln k_j \end{aligned}$$

Here, the parts $\sum_j \ln k_j$ and $\ln \lambda \sum_j k_j$ are going to be the same for all i 's. Also, $\sum_j 1$ is the count of the copy-numbers, and $\sum_j k_j$ is the summation of the copy-numbers. Therefore, maximizing the following suffices:

$$f(i) = -\lambda i \times (\text{count of copy-numbers}) + \ln i \times (\text{sum of copy-numbers}) \quad (4.11)$$

$$= c_1 \ln(i) - c_2 i \quad (4.12)$$

Here, $c_1 = \text{count of copy-numbers}$ and $c_2 = \lambda \times (\text{sum of copy-numbers})$. These are two positive constants with respect to i .

The i that maximizes the $f(i)$ is the estimated value of $C_{G,t}$. Taking derivative with respect to i , we have:

$$f'(i) = \frac{c_1}{i} - c_2 \quad (4.13)$$

To maximize this, we would need to calculate the i for which $f'(i) = 0$. This results in the following:

$$i = \frac{c_1}{c_2} \quad (4.14)$$

Verifying that this local optima is indeed a maxima, we take the second derivative and plug-in this value of i as follows.

$$\begin{aligned}
f''(i) &= -\frac{c_1}{i^2} \\
&= -\frac{c_1}{\left(\frac{c_1}{c_2}\right)^2} \\
&= -\frac{c_2^2}{c_1}
\end{aligned}$$

Since c_1 and c_2 both are always positive, the second derivative is negative for this value of i , indicating that this i will maximize $f(i)$, and consequently, the likelihood.

4.5.4 Calculating the score

Finally, having introduced the pieces to profile the off-target activities, the score assigned to a gRNA, which can be interpreted as the inverse of the specificity, is defined as follows:

$$s \approx \frac{E[x_G]}{E[x_t] \times c_{G,t}} \quad (4.15)$$

4.5.5 Learning probability distributions and estimating priors

To calculate the scores using the equations mentioned above, the prior probability distribution as well as the posterior in Equation (4.5) need to be defined. First, we are going to establish the method of calculating the priors.

Just as we have been using this for quite a number of analyses: if a genomic region is present i times, then the counts of the k -mers within that region are assumed to be Poisson distributed with mean λi , where λ is the k -mer coverage of the dataset. However, k -mers may also appear due to sequencing errors even though it is not present in the genome. This results in large number k -mers that appear one or few times. This is also evident in Figure 4.2, where a very large number of 23-mers are noticed that appear only a few times in the sequenced reads.

Therefore, it is important to model these error-reads (referred to as the “error-component” of the spectrum henceforth) accurately. In our experiments, we tried the following distributions to model this error-component:

- Poisson: with mean of Poisson set to very low, the Poisson “thinning” can model such sharp-rate data
- ZTP (Zero Truncated Poisson): A zero truncated Poisson is a Poisson who never yields zero. The PMF and CDF are revised accordingly [126].

- Empirical: We also converted the k -spectrum into a histogram and treated values below a threshold directly as probabilities.
- Geometric: Geometric distribution with small mean has a very sharp falling rate, exactly like ours.

We found that the geometric distribution works the best in modeling this error-component.

Since organisms differ in their repeat structure, we estimate priors on copy numbers in the genome from the data. The steps of the EM are detailed further in broken down section.

◦ The input

The input to the EM algorithm is the k -mer spectrum of the sequenced reads, like the example showed in Figure 4.2. Let the copy number in this spectrum (the values plotted along the y-axis) be k_1, k_2, k_3, \dots etc.

◦ The model

This input data is modeled using $N + 1$ distribution: D_0, D_1, D_2, \dots upto D_N . Here, D_0 is a geometric distribution, and the others are Poissons.

◦ Initialization

1. We initialize read-coverage, λ as the first peak after the trough in the spectrum
2. We initialize the mean of the distributions as follows:

$$D_i.\text{mean} = \lambda_i = \begin{cases} 1.0 + \epsilon & \text{if } i = 0, \epsilon \text{ is a small number} \\ i \times \lambda & \text{otherwise} \end{cases}$$

3. We initialize the priors as: $\pi[i] = 1.0/(N + 1)$ for all i

◦ The ‘E’-step

1. Probability of observing a copy-number k_j from the distribution $D_i, P_i(k_j)$:

$$P_i(k_j) = \begin{cases} \frac{1}{\lambda_i} \times \left(1 - \frac{1}{\lambda_i}\right)^{(k_j-1)} & \text{if } i = 0 \\ \frac{e^{-\lambda_i} \times \lambda_i^{k_j}}{k_j!} & \text{otherwise} \end{cases}$$

2. Membership of the copy-number k_j to the distribution D_i , $M_i(k_j)$:

$$M_i(k_j) = \frac{\pi[i]P_i(k_j)}{\sum_i \pi[i]P_i(k_j)}$$

◦ The ‘M’-step

1. The means of the distributions, λ_i :

$$\lambda_i = \frac{\sum_j M_i(k_j) \times k_j}{\sum_j M_i(k_j)}$$

2. The priors, $\pi[i]$:

$$\pi[i] = \frac{\sum_j M_i(k_j)}{\sum_j k_j}$$

3. The estimated read coverage, λ :

$$\lambda = \sum_i \pi[i] \times \lambda_i \text{ for } i > 0$$

Just as the traditional EM-algorithm, these ‘E’ and ‘M’ steps are repeated until convergence. When the algorithm finishes, the read-coverage λ as well as the priors are ready to be used in Equation (4.5).

4.5.6 Calculating the posterior probabilities

The last piece, which puts everything together, is the posterior probability $P(c_{R,k})$, the probability of observing a certain sequence $c_{R,k}$ times in the set of sequenced reads. This is simply determined using the priors as follows:

$$P(c_{R,k}) = \sum_{i=1}^{\infty} P(c_{R,k}|c_{G,i}) \times P(c_{G,i}) \quad (4.16)$$

Here, $P(c_{R,k}|c_{G,i})$ is the probability of observing a 23-bp long k -mer $c_{R,k}$ times, when we know that the sequence appears $c_{G,i}$ times in the genome; which is calculated just as Equation (4.4).

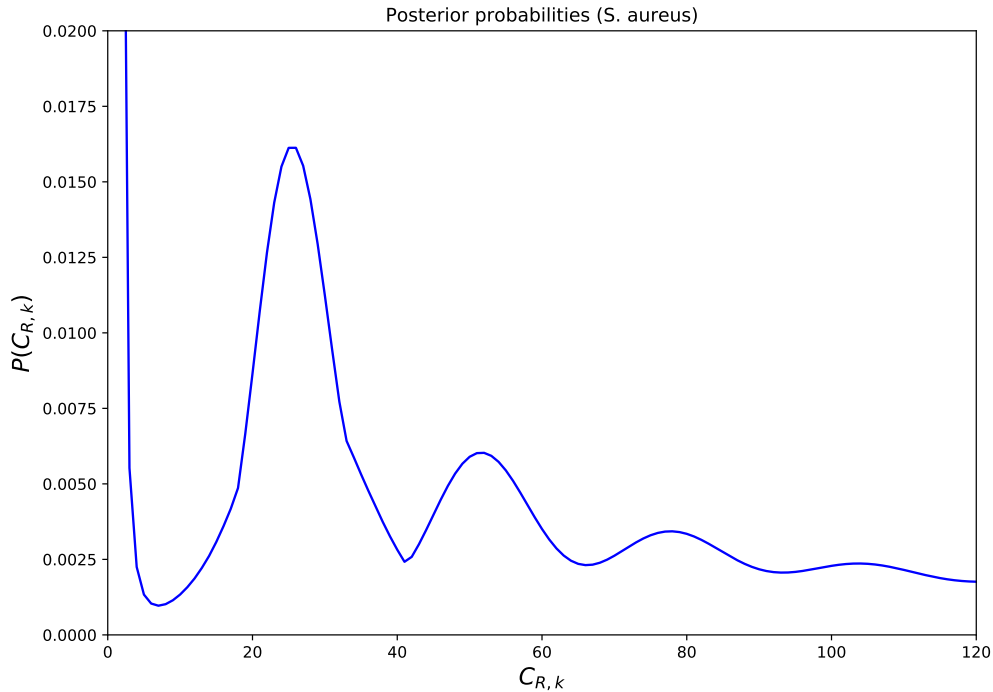


Figure 4.3: Posterior probabilities determined by kRISP-meR working with sequenced reads of *Staphylococcus aureus* (taken from GAGE [5]).

4.6 Determining on-target scores

So far, we have discussed the methods to determine a score profiling the off-target effects of a particular guide RNA. It is also important to profile on-target activity of the guides as well.

On-target, or the “cutting efficiency” scores for each candidate gRNA is determined using the scheme suggested by Doench et al. [41] As in GuideScan [103], we use Rule Set 2 [41] which takes as input a sequence of length 30 at the target site and the 20-mer from the candidate gRNA and calculates cutting efficiency of the candidate gRNA using a boosted regression tree model.

As calculating this model requires the reference genome information, kRISP-meR does not attempt in calculating this model. Instead, this model must be delivered to kRISP-meR as an argument. If no such model is given, then kRISP-meR will simply annotate the off-target score according to Equation (4.15).

4.7 Importance of the error component

The crucial role played by the posterior probabilities is explained with the help of Figure 4.3. Here, the read-coverage is estimated to be a little over 25. The periodic local maximas capture the repetitive regions, and the likelihood of observing a k -mer from those regions.

While counting k -mers in the \mathbb{K} of a candidate guide RNA, if we find a particular sequence present very small number of times (compared to the read-coverage), then it is crucial that we treat that copy-number as an error resulting from sequenced reads. Otherwise, this small number would contribute largely to the value $E[x_G]$ in Equation (4.5), and the overall score in Equation (4.15) would no longer be accurate.

The initial sharp fall-off serves the purpose of neglecting the small copy-numbers. That is, if a copy-number is too small, then the posterior for that copy-number would be very large. As the posterior will divide the other values, that particular copy-number would not be able to contribute significantly.

4.8 Interpretation of the score

It is particularly important to establish an interpretation of the score defined in Equation (4.15) which is assigned to each of the guide RNAs. We present two interpretations of the score, a direct interpretation, and a relation with the “specificity” scores.

4.8.1 Direct interpretation

The score is the ratio of the number of cuts made by a guide RNA in the whole genome to the number of cuts made by the guide RNA in the target site. Therefore, a score of 2.0 means that this particular guide RNA is likely to make two cuts in the whole genome, given that it makes only one cut in the target site (and, of course, that the target site appears only once in the genome). Accordingly, the number of off-target cuts made by this guide RNA should be 1.

Thus, the score in Equation (4.15) has two interpretations:

- The number of cuts made in genome by a guide RNA with score s will be s times the number of cuts made in the target site by the same guide RNA
- This follows from the previous: the number of off-target cuts made by a guide RNA with score s will be: $s - 1$

It is important to note that, by definition, the score must be in the range $[1, \infty)$.

4.8.2 Relation with specificity score

As discussed before in literature review, specificity scores are popular metrics in profiling the off-target effects of guide RNAs. A high specificity guide RNA is less likely to make off-target cuts, and vice versa. Specificity scores are given in the range $[0,1]$. Therefore, the score defined in Equation (4.15) can be thought of as the inverse of the specificity score for that guide RNA.

Due to the direct interpretation, kRISP-meR annotates each guide RNA with the inverse-specificity score. However, at times, it is easier (and more meaningful) to analyze certain things with the specificity scores. Hence, in our subsequent discussions (experiments and results), we will specifically mention whether the score being discussed is specificity or inverse-specificity.

Chapter 5

Experiments and results

In this chapter, we compare kRISP-meR with other popular guide RNA designing tools that require a reference genome. The experiments in this chapter show that, without having to use a reference genome, kRISP-meR is able to generate lists of guide RNAs in par with the other tools.

5.1 Datasets

Comparing kRISP-meR with other tools means that we would have to build genome-wide guide RNA libraries for other tools “using a reference genome” and extract guide RNAs for a certain genomic target site; and generate a list of guide RNAs from kRISP-meR using sequenced reads. That is, we need both the reference genome and a set of sequenced reads. Table 5.1 lists all organisms whose genome was used for various experiments, and the sources from which the sequenced reads were taken.

5.1.1 The genome assemblies

The genome assemblies that we use are selected from various sources, as shown in Table 5.1. For *S. aureus*, *R. sphaeroides* and *H. sapiens* chr-14, we collected the assembly from the GAGE dataset. GAGE is considered to be the gold standard dataset which are used to critically evaluate the genome assemblers and assembly related bioinformatics tools. These tools often compare themselves and the results with GAGE assembly using the sequenced reads available in the GAGE dataset [127].

For the other three organisms, we used the consensus assembly converged after multiple iterations done by the researchers. For *S. cerevisiae*, we used the S288C strand assembly available in the yeast-genome database popularly known as *Saccharomyces* Genome

Database (SGD). This database is developed with a view to combine all sequenced reads of *S. cerevisiae* from experiments performed all over the world. With time, the database polishes available assembly and publishes new releases. In our experiments, we used the S288C assembly in particular (out of multiple available assemblies) because this is widely known and accepted by default by many tools.

Next, for *C. elegans* and *D. melanogaster*, just as *S. cerevisiae*, there are multiple assembly releases available. Out of those, we used the ce11 release and dm6 release [128] for our experiments respectively. Again, this is because it is important to compare our results with other tools, and most web interfaces of bioinformatics tools use the ce11 as well as dm6 releases by default.

5.1.2 The sequenced reads

Just as the genome assembly, we again used the sequenced reads available in the GAGE dataset for *S. aureus*, *R. sphaeroides* and *H. sapiens* chr-14. Again, the reason to select this set of reads is because GAGE provides a gold standard for genome assemblers and assembly related bioinformatics tools. The sequenced reads available in GAGE often are used to determine what read coverage is determined even when designing a sequencing experiments and analyzing the requirements. Therefore, it is natural to use these reads and generate results.

In GAGE, there are two libraries available for each of these three dataset. For our experiments, we used the fragmented dataset with paired-end reads. The coverage is greater than 15 for all cases, which clearly indicates that the data is good enough to be used in experiments (the error components will not merge them with the other poisson components).

For the other three organisms, we searched the NCBI database and sorted out sequenced reads. All of these sequenced reads were collected using the Illumina technology, which is less expensive and more available compared to other sequencers, and at the same time, is able to render a significantly less rate of sequencing error. All these datasets have a minimum coverage of 15. The reads are collected as FASTQ files, which is the expected input format for kRISP-meR.

5.1.3 Using paired-end reads

kRISP-meR has been developed and implemented considering that the sequenced reads are unpaired reads. Therefore, when we are using paired-end sequenced reads collected from a sequencing experiment, how we accommodate this is that we either use only one

Table 5.1: Datasets used to perform various experiments and their sources

Organism name	Genome source	Sequenced reads source	#bases (reads)
Staphylococcus aureus	GAGE	GAGE	1.3G
Rhodobacter sphaeroides	GAGE	GAGE	2.1G
Homo sapiens (Chr 14)	GAGE	GAGE	3.6G
Saccharomyces cerevisiae	link (S288C)	ncbi: SRR12442616	2.9G
Caenorhabditis elegans	link (ce11)	ncbi: ERX1161562	4.0G
Drosophila melanogaster	link (dm6) [128]	ncbi: SRR7167961	4.0G

Table 5.2: Genomic co-ordinates of the target sites

Organism name	Genomic coordinates
Staphylococcus aureus	genome:3221-7700
Homo sapiens (Chr 14)	chr14:33322521-33327770
Saccharomyces cerevisiae	chrI:67141-68940

of the two pair of reads and simply drop the other file, or we merge both the files into a single FASTQ file and simply consider two paired-end reads as independent unpaired reads. Our experiments reveal that doing so does not harm the outcome of the tool.

5.2 Comparing inverse of specificity scores: with and without a reference

For the organisms *Staphylococcus aureus*, *Saccharomyces cerevisiae* and the Chromosome-14 of *Homo sapiens*, we picked random segments from the reference genome as target sites and generated guide RNAs with kRISP-meR from sequenced reads. The exact genomic locations of these targets are listed in Table 5.2.

After generating the guide RNAs with kRISP-meR, we listed the inverse-specificity scores attributed to each gRNA according to Equation (4.15). Then, to assess that these scores are, in fact, meaningful and significant, we calculated the same scores; but this time, using the reference genome. That is, instead of calculating the number of expected copy numbers of a particular sequence, we directly calculated the number of times this sequence exists in the reference. These two versions of the same scores, one calculated by kRISP-meR without the reference, and the other with the reference – are contrasted in Figure 5.1.

The genome of *Staphylococcus aureus* and *Saccharomyces cerevisiae* have low repeat

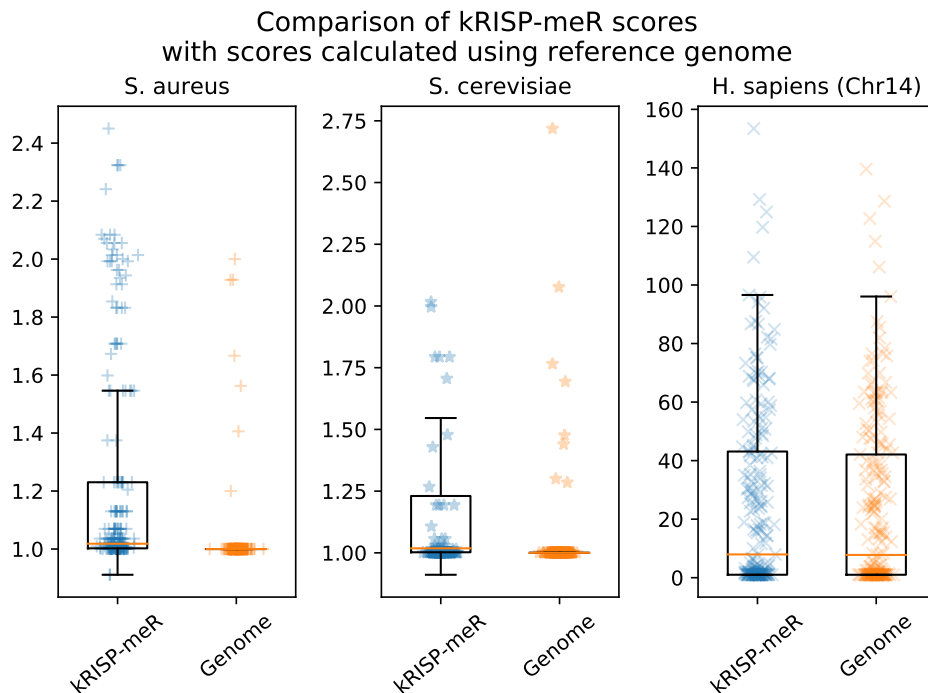


Figure 5.1: Comparison of inverse-specificity scores calculated by kRISP-meR and separately calculated using a reference (each point in the figure denotes a gRNA).

complexity. For these two, we found that the majority of the gRNAs have scores close to 1.0 according to both the schemes. The scores calculated without the reference genome have greater spread compared to the ones obtained using the reference. This is expected because of the variation in sequencing depth across the genome caused by stochasticity and biases in the sequencing process as well as the presence of extraneous k -mers arising because of sequencing errors. However, we observe that almost half the gRNAs have scores close to 1.2 or less, and almost three-fourths are below 1.5.

For the repeat-rich Chromosome-14 of *Homo sapiens*, there are many gRNAs with high scores indicating a lack of specificity and highlighting the importance of appropriately scoring and avoiding these as guides. For this dataset, we observe that the scores computed using both approaches have similar median and inter-quartile ranges.

Another popular way to contrast the specificity scores is to put their CDFs calculated by different methods in the same plot. Since the inverse of the specificity scores are not limited in a range, we plotted the CDFs of the specificity scores in Figure 5.2. Paralleling the boxplots of Figure 5.1, the CDFs for *Staphylococcus aureus* and *Saccharomyces cerevisiae* disagree a little, whereas for *Homo sapiens*, the CDFs are more aligned and agreeing owing to the same reasons.

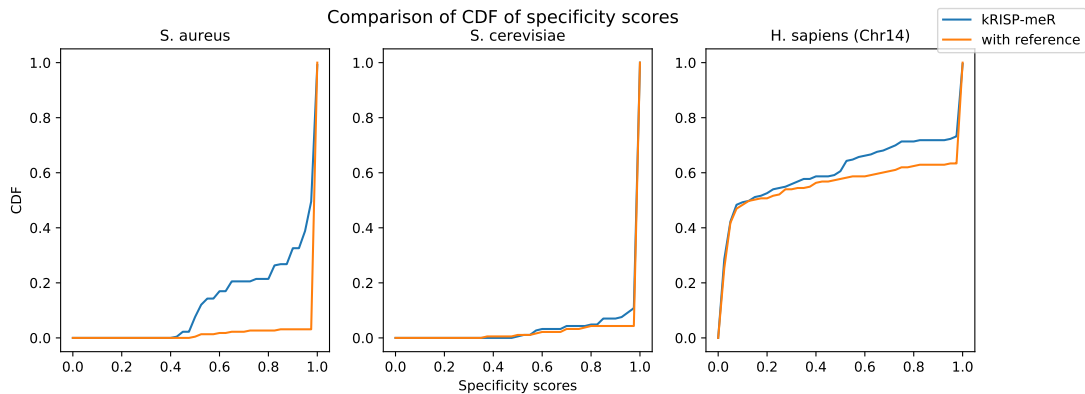


Figure 5.2: CDFs of specificity scores calculated by kRISP-meR and separately calculated using a reference.

5.3 Determining the default cut-off score and pruning guide RNAs

As evident from Figure 5.1, particularly the Homo sapiens scores, the inverse of specificity scores for a lot of the guide RNAs are very high, which indicates very low specificity. These high scores, in turn, mean that these guide RNAs are likely to result in cleavages in sites other than the target site. Ergo, it is crucial to identify such low-specificity guide RNAs and drop them. Therefore, we have to determine a particular cut-off specificity score for the guide RNAs.

To do so, we look at the interpretations of the scores discussed in Section 4.8. The number of off-target cuts made by a guide RNA is profiled by the inverse-specificity score, which is determined by subtracting 1.0.

We wanted to investigate and establish a range in which the predicted number of off-target cuts is low and agrees well with the actual number of off-target cuts. The exact number of off-target cuts made by a particular guide RNA is calculated in silico, following the schemes introduced by Doench et. al [41]. Having determined the predicted and actual number of off-target cuts for all the guide RNAs, we plotted specificity in the x-axis and the number of off-target cuts made by guide RNAs with a specificity lower than that in the y-axis. The plot is shown in the top-left of Figure 5.3. This plot shows that for an extensive range of specificity scores (0.33-1.0), the two curves are horizontal. This means that the guide RNAs with a specificity higher than 0.33 (inverse of specificity lower than 3.0) produce a small number of off-target cuts.

The top-left figure showed the number of off-target cuts resulted by guide RNAs with specificity lower than a specific value. The same figure with the number of off-target cuts caused by guide RNAs with a specificity higher than a particular value is shown

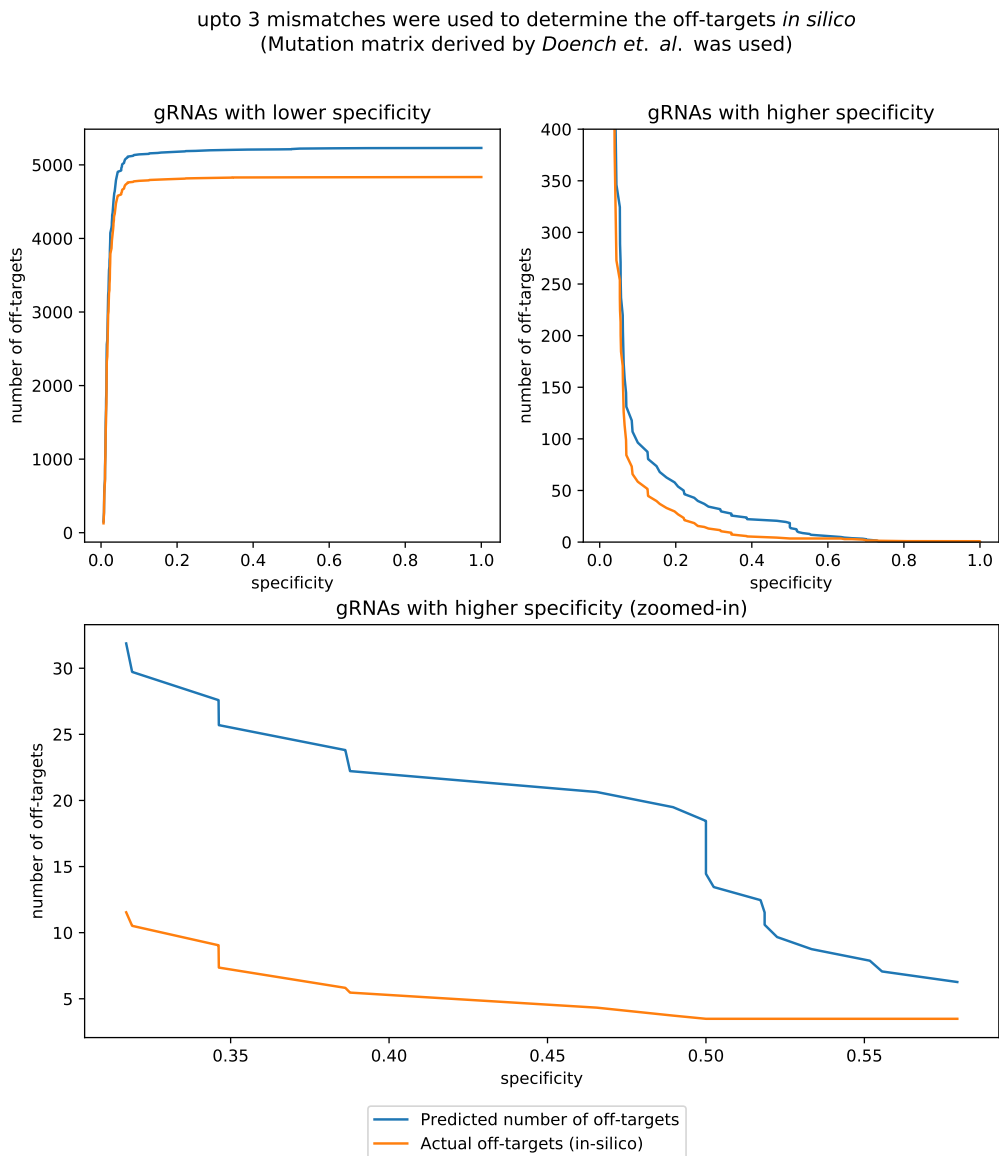


Figure 5.3: Number of off-target cuts (predicted and actual) made by guide RNAs with a specificity score lower/higher than a certain value.

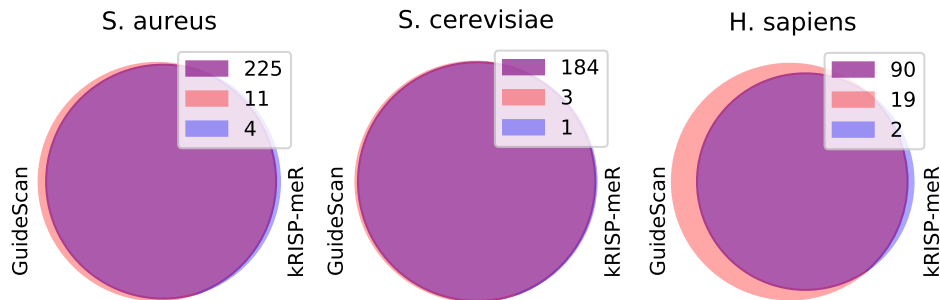


Figure 5.4: Venn diagrams showing numbers of candidate guide RNAs found by only kRISP-meR, only GuideScan and both tools for randomly chosen regions in *S. aureus* genome, *S. cerevisiae* genome and *H. sapiens* chromosome 14.

in top-right. This somewhat zoomed-in view urges us to look even closer. The plot at the bottom allows us to entertain that interest, which reveals that to consider a guide RNA to hold “very high specificity”, we are to drop all guides with specificity scores below 0.5 (inverse of specificity higher than 2.0).

Finally, we designed kRISP-meR to output guide RNAs with an inverse-specificity score less than or equals to 3.0. kRISP-meR can be asked to output only the guide RNAs with “very high” specificity, in which case, kRISP-meR outputs the guides with inverse-specificity less than or equals to 2.0. kRISP-meR can also run with any cut-off score with the argument ‘-c’ to facilitate keener investigations.

5.4 Comparison with GuideScan to validate the performance

Having determined the cut-off score, we listed the output guide RNAs from kRISP-meR using this cut-off, and compared them with the guide RNAs generated by GuideScan [103], a popular and widely used guide RNA designing tool which requires a reference genome.

We ran kRISP-meR on the randomly chosen target sites from the genomes of *Staphylococcus aureus*, *Saccharomyces cerevisiae* and the Chromosome-14 of Homo

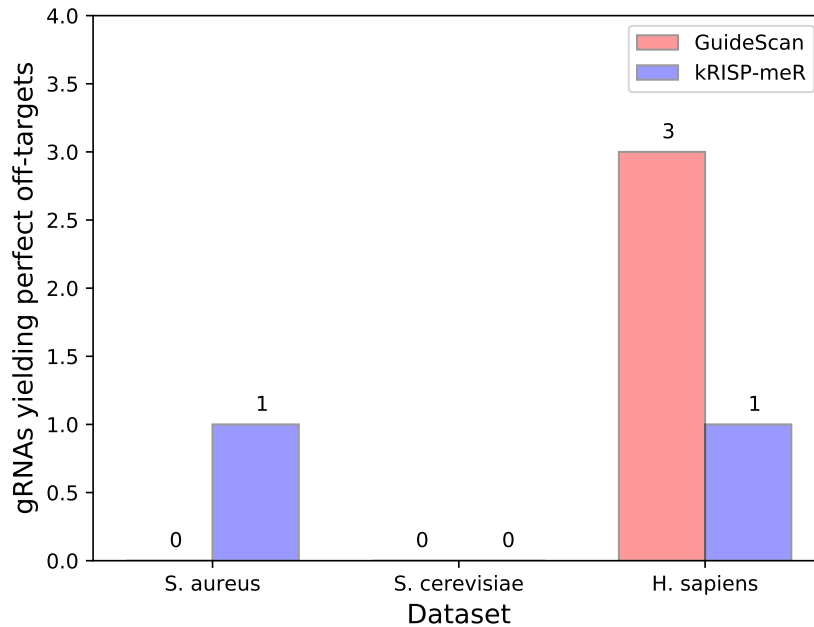


Figure 5.5: Numbers of guide RNAs predicted by kRISP-meR and GuideScan with off-target effects with no mismatches.

sapiens. The exact location of the target sites are shown separately in Table 5.2. We also ran GuideScan on the three reference genomes and built genome-wide sgRNA library with a maximum of 2 mismatches. Next, we identified the sgRNAs from the library for the same target region (which was used from kRISP-meR). Figure 5.4 shows the extent of agreement between the sets of sgRNAs generated by kRISP-meR and GuideScan. It reveals that kRISP-meR is able to recognize the majority of the sgRNAs identified by GuideScan while reporting very few additional ones for all three the datasets.

Finally, we investigated potential off-target effects of the highly specific guide RNAs generated by kRISP-meR. For all the guides identified by GuideScan, and by kRISP-meR, we determined whether they match any site outside of the target region considering no mismatches i.e. have perfect matches. The results are shown in Figure 5.5. We observe that for the *S. aureus* dataset, kRISP-meR generates one guide RNA with off-target effect whereas GuideScan has none, for the *S. cerevisiae* dataset, no off-target cut was made by any guide RNA, and for the human chromosome 14 dataset, kRISP-meR and GuideScan generate 1 and 3 guide RNAs with off-target effects respectively.

This indicates that, for the human chromosome 14, kRISP-meR outperforms GuideScan in terms of the number of off-target producing guide RNAs. At the same time, this also indicates that for a simple genome such as *S. aureus*, kRISP-meR is performing

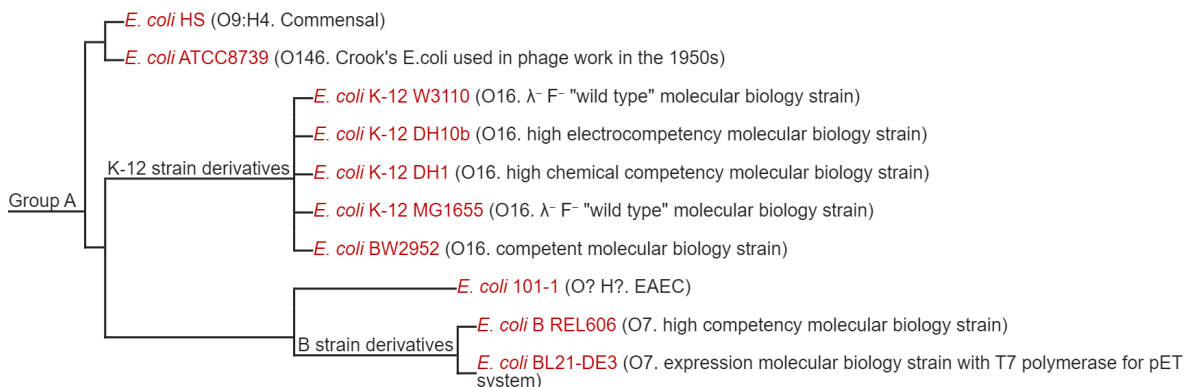


Figure 5.6: Partial phylogenetic tree constructed using the strains of *E. coli*. [6]

worse. This is expected, as we have already seen in Section 5.3, the scores work the best when there are some repetitions in the genome. As the methods of kRISP-meR take into account the repeated regions very carefully, it was able to throw out two of the off-target producing guide RNAs; which GuideScan fails to do.

These investigations from Figure 5.4 and Figure 5.5 reveal that the determined cut-off scores work well in capturing the guide RNAs with high specificity.

5.5 Comparison with GuideScan in terms of variant-aware gRNA design

Having realized the potential for kRISP-meR to outperform reference-based tools in terms of number of off-target cuts by working in a variant-aware fashion, we wanted to investigate this further. In order to do that, we selected a simple and short genome: *Escherichia coli* to do our experiment.

E. coli is a bacterium of the genus *Escherichia*. This is commonly found in the lower intestine of warm-blooded organisms. *E. coli* has various strains, most of them has been well characterized. For our experiment, we chose the strains of K-12 (MG1655 variant), which is the standard used when considering *E. coli*, and ATTC8739. These two are quite similar in their genome structure and are close relatives in the phylogenetic tree of the strains of *E. coli* (see Figure 5.6), and yet, they have enough dissimilarities to consider one of the assemblies an incomplete representative genome for the other and do our experiment.

We collected sequencing reads of the ATTC8739 strain from a real experiment [129] for our analysis. There are 441.3M bases in this dataset of sequencing reads, with a read-coverage of 80. The sequence technology that was used was Illumina MiSeq. We collected the assembly of both the strains from NCBI standard releases [130, 131].

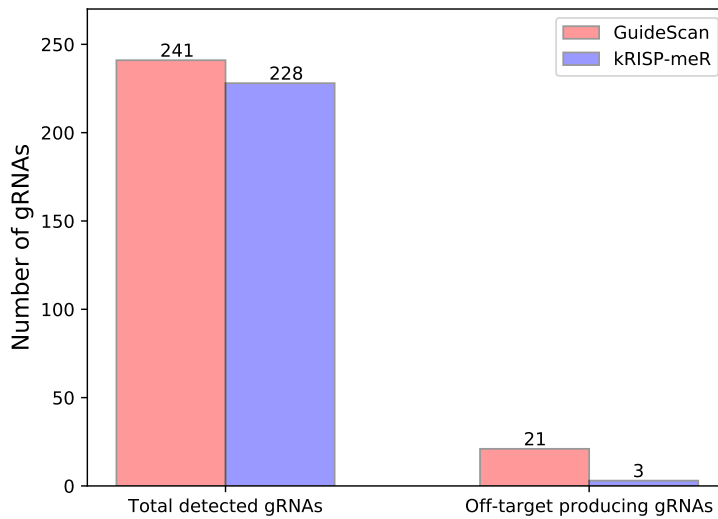


Figure 5.7: Number of guide RNAs resulting in perfect off-targets identified by kRISP-meR and GuideScan. GuideScan uses K-12 assembly and kRISP-meR uses ATTC8739 reads. The off-targets are counted in ATTC8739 assembly.

For this experiment, what we did is, we chose this target sequence ([link](#)) from the genome assembly of the K-12 strain. Then, we used the sequencing reads of the ATTC8739 strain to design guide RNAs for this target sequence using kRISP-meR. At the same time, we also used the standard K-12 assembly for populating the genome-wide guide RNA library for K-12 using GuideScan. Then, we looked for perfect off-target cuts (considering zero hamming distance) in the genome of the ATTC8739 assembly. The results are shown in Figure 5.7, which shows that the number of guide RNAs that result in one or more perfect off-target cuts is 21 for GuideScan, and only 3 for kRISP-meR.

The idea behind doing this experiment is: we have K-12 and ATTC8739 assemblies as two individuals of the same species with different genomes. In other words, K-12 genome is the incomplete genome which represents the K-12 individual perfectly and not able to represent the ATTC8739 individual accurately. Using the sequencing reads of the ATTC8739 strain individual, we may be able to design a better set of guide RNAs in contrast to using the K-12 assembly. Hence, considering K-12 assembly as the reference genome for this species, the outcome of our experiment is to show that the guide RNAs designed using this reference will result in a higher number of off-target cuts when used in a CRISPR experiment on the ATTC8739 strain. The results shown in Figure 5.7 reveal exactly that.

Table 5.3: Genomic co-ordinates of the target sites of *Saccharomyces cerevisiae*, and number of guide RNAs recognized by different tools for these targets

Target sequence	GuideScan	CRISPRSCAN	CRISPOR	kRISP-meR
ChrI:67141-68940	187	97	191	185
ChrII:18001-19800	190	98	189	191
ChrIII:79201-81000	191	96	180	191
ChrIV:86521-88320	188	94	184	185
ChrV:16321-18120	185	97	186	186
ChrVI:88501-90300	188	95	187	188
ChrVII:85381-87180	192	94	189	188
ChrVIII:279121-280920	191	94	186	192
ChrIX:154981-156780	192	96	181	188
ChrX:231661-233460	185	95	183	187
ChrXI:83941-85740	189	98	189	185
ChrXII:255901-257700	187	94	187	186
ChrXIII:56701-58500	187	96	189	188
ChrXIV:272641-274440	193	95	180	188
ChrXV:329881-331680	186	95	192	185
ChrXVI:490681-492480	194	96	198	193

5.6 Comparison with other gRNA designing tools

Finally, we went on to compare kRISP-meR with other popular guide RNA designing tools, namely GuideScan [103], CRISPRScan [81] and CRISPOR [105]. All these three require a reference genome to operate.

We experimented on the genome of three different organisms: *Saccharomyces cerevisiae* (release 3), *Caenorhabditis elegans* (release 11) and *Drosophila melanogaster* (release 6). We built genome wide guide RNA library for the three organisms for all of the three reference-using tools. Then, for each of the chromosomes of these organisms (16 for *S. cerevisiae*, 6 for *C. elegans* and 8 for *D. melanogaster*), we located a random target site. For the chosen target site, we retrieved the guide RNAs from the genome wide library. Next, we located sequenced reads for various strains of all three of the organisms from real experiments by browsing the Short Reads Archive (SRA). Using the target sites mentioned in the last paragraph and the located sequenced reads from publicly available experiments, we generated guide RNAs for all of the target sites considering the corresponding sequenced reads.

Table 5.4: Genomic co-ordinates of the target sites of *Caenorhabditis elegans*, and number of guide RNAs recognized by different tools for these targets

Target sequence coordinates	GuideScan	CRISPRSCAN	CRISPOR	KRISP-mER
ChrI:47451-49450	113	41	124	116
chrII:37801-39800	112	39	128	114
chrIII:44401-46400	112	45	130	117
chrIV:61701-63700	117	43	118	121
chrV:119351-121350	119	45	117	121
chrX:436651-438650	111	38	130	114

Table 5.5: Genomic co-ordinates of the target sites of *Drosophila melanogaster*, and number of guide RNAs recognized by different tools for these targets

Target sequence coordinates	GuideScan	CRISPRSCAN	CRISPOR	KRISP-mER
armX:44401-64600	188	102	199	193
arm2L:37801-39800	187	104	203	191
arm2R:44401-46400	186	101	195	191
arm3L:61701-63700	194	104	203	196
arm3R:119351-121350	192	107	203	194
arm4:436651-438650	192	99	193	198
armU:119351-121350	187	100	203	190
armXHet:436651-438650	185	98	193	187

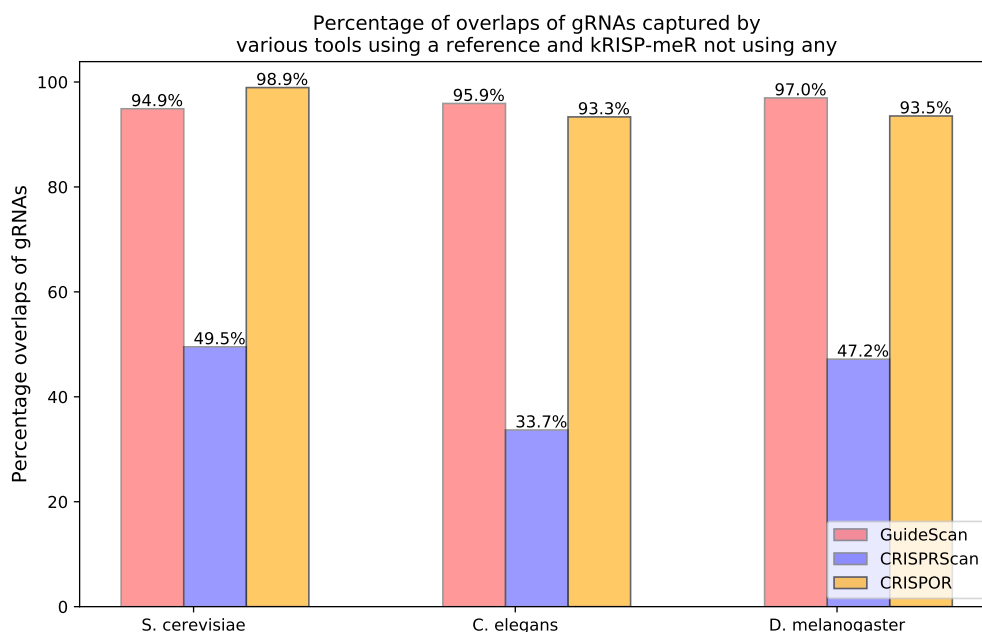


Figure 5.8: Overlap of the guide RNAs recognized by CRISPR and other tools (aggregated over a number of experiments).

5.6.1 Number of guide RNAs recognized

The target sites for the three organisms, along with the number of guide RNAs recognized by the four tools (kRISP-meR and the other three) are shown in Tables 5.3, 5.4 and 5.5.

Immediately, it is evident that the number of guide RNAs identified by CRISPRScan is smaller compared to the other three (including kRISP-meR). This is expected, because of the way CRISPRScan work. CRISPRScan drops quite a few number of guide RNAs based on the Adenine and Guanine contents. Likewise, it is expected that the level of agreement between kRISP-meR and CRISPRScan is going to be less than what we observe for the other two tools (GuideScan and CRISPOR).

5.6.2 Overlap of the guide RNAs recognized by kRISP-meR and other tools

Figure 5.8 shows the overlap of the guide RNAs generated by the other three tools with the guide RNAs generated by kRISP-meR. The percentages shown in the y-axis are calculated as follows: for a certain experiment, we determined the size of the the sets $A \cap B$ and $A \cup B$ where A and B stand for the set of guide RNAs recognized by kRISP-meR and the other contrasting tool; and then we converted $\frac{|A \cap B|}{|A \cup B|}$ into percentages. We

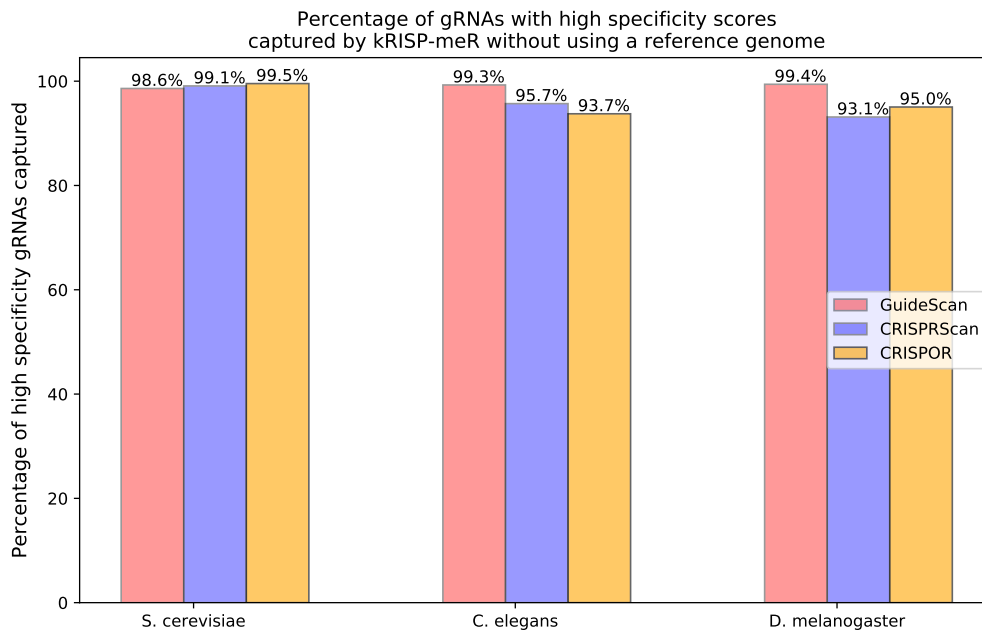


Figure 5.9: Percentage of high-specificity guide RNAs captured by kRISP-meR.

averaged the percentages for all the experiments for a certain organism and plotted that in Figure 5.8. While this plot shows a quite high degree of agreement for kRISP-meR with GuideScan and CRISPOR, the figure is not too impressive with CRISPRScan. This is because CRISPRScan drops some of the guide RNAs by examining the sequence and looking for adenine and guanine content. This is also evident from the number of guide RNAs recognized by CRISPRScan, listed in Tables 5.3, 5.4 and 5.5.

Next, we show the percentage of guide RNAs with high specificity score (generated by the contrasting tool) captured by kRISP-meR's list of guide RNAs with high specificity. For all the experiments of a certain organism, we aggregated the percentages and plotted them in Figure 5.9. This shows that kRISP-meR was able to capture almost all of the high-specificity guide RNAs in its own list of high-specificity guide RNAs.

The results shown in Figure 5.8 and Figure 5.9 simply support our validation made by the experiments detailed in Section 5.4, in which we saw that kRISP-meR is able to produce results that are directly comparable to those of GuideScan. The results in Figure 5.8 and Figure 5.9 just support the same for two other reference-based tools.

5.6.3 Off-target cuts made by the guide RNAs

Having investigated the overlap of the identified guide RNAs, we next calculated the number of guide RNAs that result in perfect-off targets. A perfect off-target, as

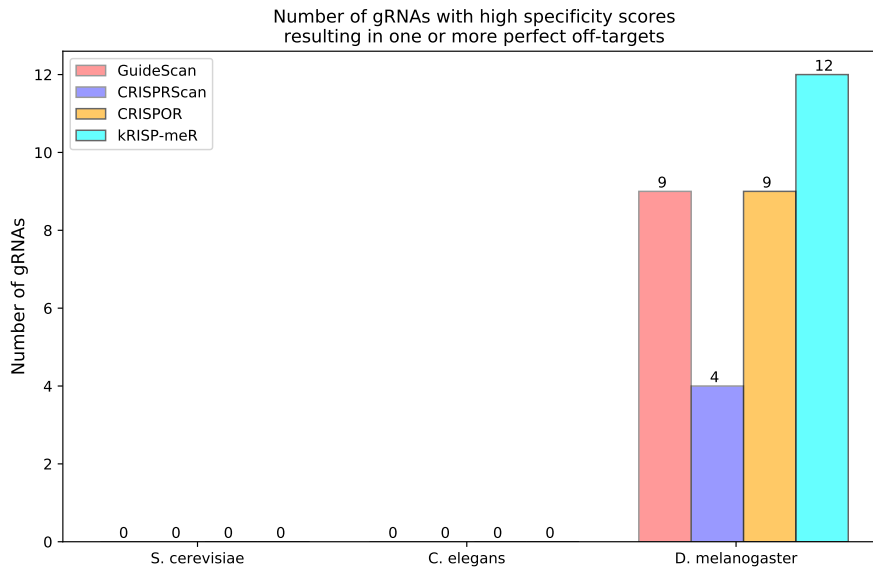


Figure 5.10: Number of guide RNAs that resulted in perfect off-target cleavages.

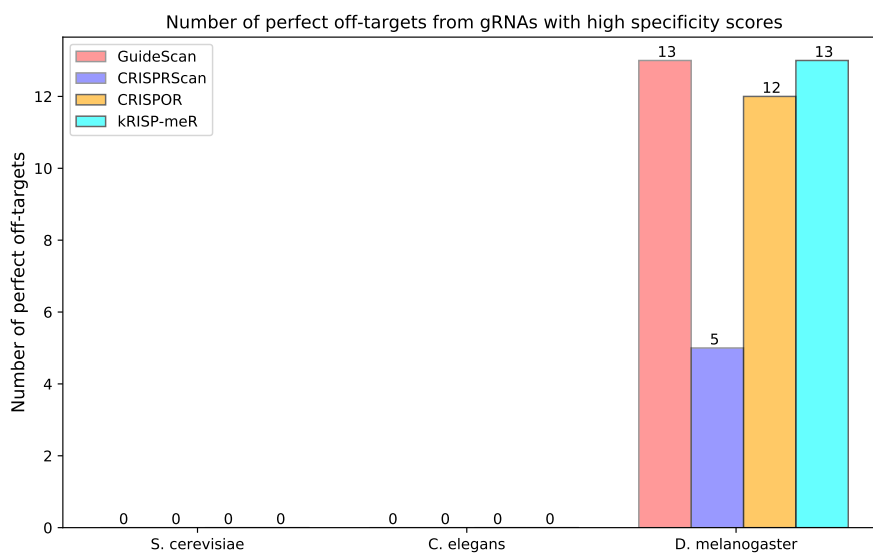


Figure 5.11: Number of off-target cleavages.

explained before, means a genomic location other than the target site where a guide RNA matches perfectly (with no mismatch). The number of such guide RNAs are shown in Figure 5.10. The actual number of off-target cuts (one guide RNA can make multiple off-target cuts) are shown in Figure 5.11.

These figures show that kRISP-meR recognizes 12 guide RNAs that result in one or more perfect off-target cuts. At the same time, this shows that this number for GuideScan is only 9. Considering the fact that one single guide RNA can result in one or more off-target cuts, it is therefore important to not only count the number of such guide RNAs, but also to count the actual number of off-target target cuts. When we do that, we see that for both kRISP-meR and GuideScan, this number is 12. This, therefore, means that there are some guide RNAs identified by GuideScan that result in more than one off-target cuts that kRISP-meR was able to throw out.

This success of kRISP-meR effectively comes from the evidence we saw in Section 5.3. As kRISP-meR accounts for the repeated regions very carefully, kRISP-meR was able to figure out and throw away the guide RNAs that resulted in more than one off-target cuts.

These investigations presented in Section 5.6 show that kRISP-meR, without using a reference genome, is able to capture guide RNAs with high specificity as recognized by other competing tools, and generate guide RNAs that result in tolerable number of mismatches compared to the huge number of gRNAs recognized.

5.7 Benchmarking the running time of kRISP-meR

In order to demonstrate that kRISP-meR will not starve for resources, we wanted to benchmark the running time of the tool. In order to do so, we ran the tool for all the target sites mentioned in Section 5.1 earlier. As the tool operates, we recorded the running time for the major components.

As we have discussed in Chapter 4, kRISP-meR first counts the k -mers in all the sequenced reads. Next, kRISP-meR polishes the target sequence to determine the genetic variations of the individual. After that, kRISP-meR has to go through a number of analyses before processing each one of the candidate guide RNAs. Finally, having all the information ready to process the potential gRNAs, kRISP-meR iterates over all the guide RNAs and assigns the off-target indicating inverse-specificity score to the guide RNAs.

We recorded the running time of all of these components. The averages of the running time are presented in Figure 5.12. We performed the experiment by varying number of mismatches that we consider when we are listing the potential target sites. The

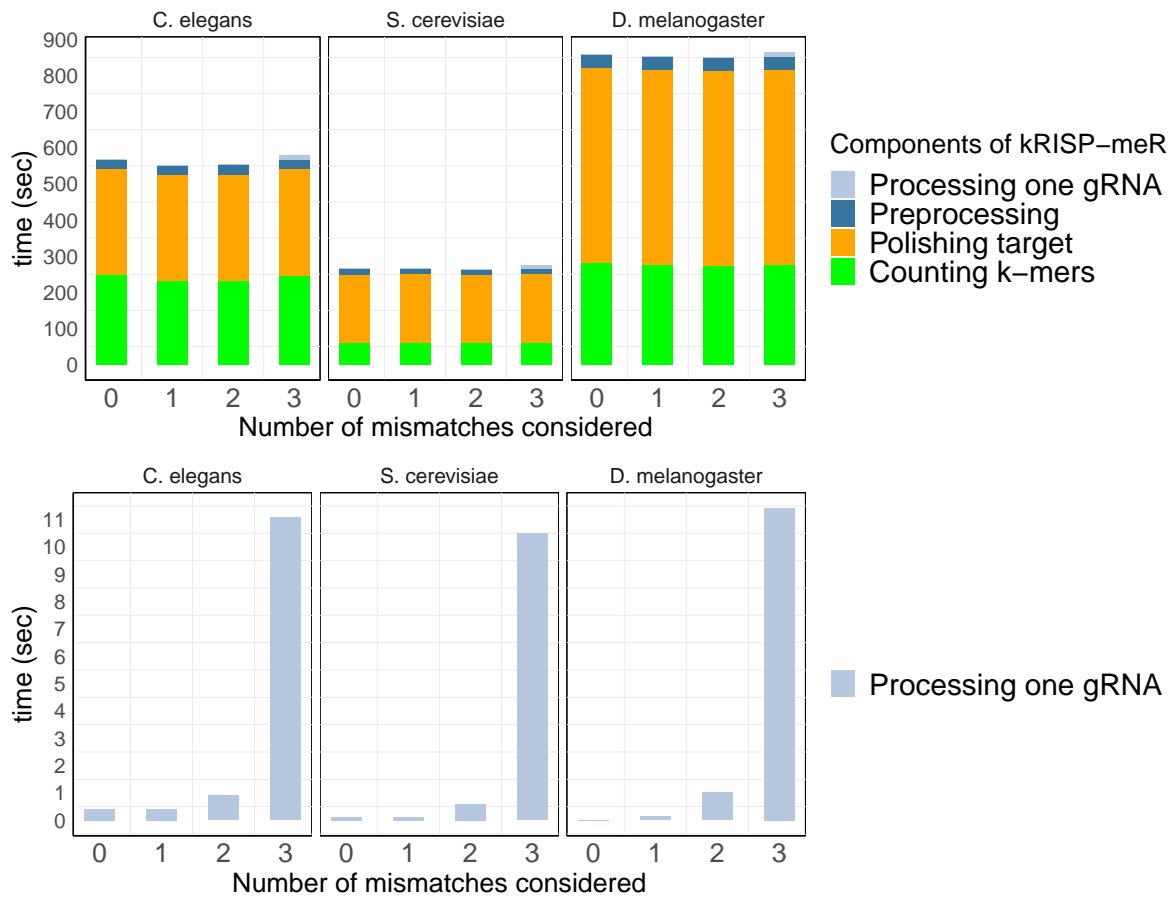


Figure 5.12: Running time of the various components of kRISP-meR. The running times presented here are the average of multiple experiments for a particular number of mismatches considered.

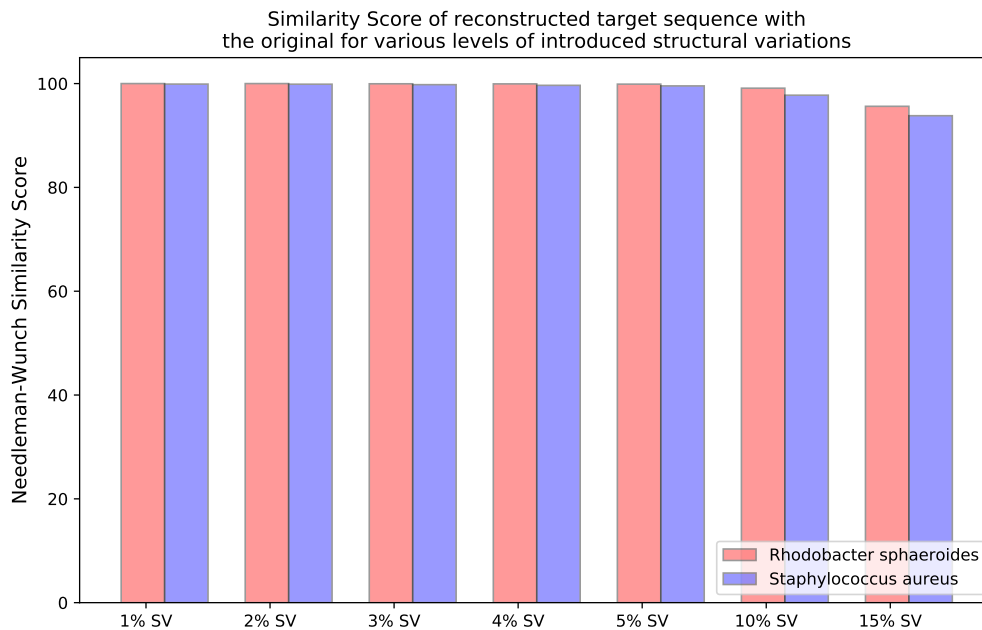


Figure 5.13: Capturing individual genetic variations and determining personalized target sequence by kRISP-meR

machine where we performed these runs is an everyday use desktop computer with only 8GB of RAM and 2.7GHz Intel i7 processor with two cores.

Figure 5.12 shows that kRISP-meR is able to identify the gRNAs in just 10-15 minutes of time. We have also made runs on the complete human genome as well by collecting and using the whole-genome shotgun sequences from credible experiments. Our records show that it takes about 104 minutes (less than 2 hours) to generate a list of guide RNAs for a human target site of 2000 nts, if the Jellyfish run and the personalized target-detection are done before running the experiment.

5.8 Experimenting the accuracy of the personalized target site

Before concluding, we wanted to confirm that kRISP-meR can identify individual genetic variations for an intended CRISPR target accurately. In our attempt to do so, we selected arbitrarily long segments from the reference genome of two organisms, namely *Staphylococcus aureus* and *Rhodobacter sphaeroides* (both the reference and the sequenced reads for these two were taken from GAGE [5]). After that, to mimic genetic variation, we perturbed these sequences using wgsim [132] and introduced structural variations (indels and SNPs). We wanted to examine how well the initial pipeline of kRISP-meR can reproduce the original sequence from this perturbed sequence. We

aligned the final sequence with the original sequence selected from the reference genome. We performed a global alignment using the Needleman-Wunch algorithm and calculated the similarity score.

For a particular organism, we picked a random sequence and repeated this experiment 20 times for a certain amount of introduced structural variation. The average similarity scores are shown in Figure 5.13, which shows that for up to 10% of structural variations, kRISP-meR can detect individual genetic variations and determine the personalized target-sequence for the individual from which the sequenced reads have been collected. This particular experiment shows that it is possible to accurately reconstruct a genome sequence of a short length (typical of the lengths of CRISPR guide RNAs) from the sequencing reads, which in turn enables kRISP-meR to be successfully claimed “variant-aware”. Besides making this claim, it is also important to note and point out that if the sequenced reads do not have any similarity at all with the input genome sequence, then we are in trouble. These results shown in Figure 5.13 were obtained using the default parameters of bowtie2 and pilon. If the sequenced reads come from an organism whose genome is a complete stranger to the reference genome sequence, then the parameters used in bowtie2 and pilon may need to be tweaked around to get an expected result. To enable that, kRISP-meR allows the user to set the parameters of these tools as well.

Chapter 6

Conclusions

In this thesis, we presented kRISP-meR, a bioinformatics tool that integrates other tools along with a novel definition of gRNA specificity score. This tool is able to generate a list of guide RNAs suitable for CRISPR/Cas9 genome editing technology. Most other tools that exist in the literature require a reference genome to work with. As this requirement is dropped in kRISP-meR, it is possible to design guide RNAs for organisms with incomplete reference genome using this tool. Besides that, kRISP-meR directly utilizes sequencing reads which contain genetic information of an individual. Therefore, kRISP-meR is also able to design gRNAs personalized for a particular individual.

Because of the fact that kRISP-meR directly uses sequencing reads and works in absence of any reference genome, it was crucial to establish a link between the two. We did so by modeling the sequenced reads with a mixture of Poisson distribution coupled with an additional geometric distribution accounting for the sequencing errors. With an Expectation-Maximization algorithm, we determine the parameters of this model and use these parameters to assign a score to all gRNAs indicating their off-target effects. To calculate this score, we list all potential target sites for a particular guide RNA, determine the cutting probability of the gRNA to each target site, and estimate the number of occurrences of the target sites in the genome (by utilizing the parameters of the mixture of Poisson distributions model). These, in turn, enable us to estimate the number of cleavages introduced by a guide RNA without having to use a reference genome.

We defined this score in such a way that this has a direct relation with the traditional specificity score, and we can get an upper bound of the number of off-target cuts made by a particular gRNA using the score as well. Since kRISP-meR lists the gRNAs using this score, it was vital to verify that the score, defined for sequencing reads, works

well. We did this verification by calculating the same score, only by using the complete reference genome. Then, we contrasted our score with the score calculates using the genome. Our analyses show that the copy number of a particular sequence within the genome can be accurately estimated by using only the sequenced reads. We also find that it is possible to predict the number of off-target cuts made by a particular gRNA without a reference genome.

Moreover, we also determined a cut-off for our score experimentally. kRISP-meR drops all potential gRNAs with a score crossing this cut-off. At the same time, we kept the option open for a user to set her own cut-off score as well.

After establishing a threshold, initially, we compared kRISP-meR with GuideScan firstly because our score definition reflects that of GuideScan a lot, and secondly because GuideScan needs a reference genome to generate a list of guide RNAs. Finding a promising agreement with GuideScan, we compared ourselves with two other tools, namely CRISPRScan and CRISPOR, both of which require a reference genome. These comparisons were done in the following manner: we took a random target sequence from the reference genome. Then, these three mentioned tools were run with the reference genome and this particular target sequence. Next, the same experiment was done for kRISP-meR as well with the same target sequence, only without using a reference genome, rather, a set of sequencing reads.

Finally, we compared the list of guide RNAs generated by kRISP-meR with the lists generated by the other three tools. Our analyses reveal that the lists are quite similar and the numbers of guide RNAs that result in one or more off-target cuts are almost the same.

6.1 Discussions

The experiments show that without using a reference genome, simply by using the information available in the sequenced reads, high-specificity guide RNAs can be identified that would perform well in real CRISPR experiments. With sequencing technologies becoming cheaper and more available everyday, decoupling the need to use a reference may turn out to be highly impactful. This holds true both for CRISPR and other application scenarios. The specificity of the guide RNAs identified by kRISP-meR also shows the potential of solving other computational problems relating to a reference genome without using a reference.

Genome editing has been massively successful in changing traits of various crops and thus was able to greatly influence agriculture. The genome of various important crops related to agriculture are still incomplete. This is owing to the fact that oftent the genome is very long (e.g. wheat) and contains multiple repeated regions. On top of that, the genome assembly literature has so many important organisms to deal with (including the humans) that the efforts to assemble to entire reference genome shifted away from these important plants. kRISP-meR is expected to solve this problem in terms of designing a CRISPR experiment and generating good guide RNAs by decoupling the use of reference genome.

Besides solving the problem of having to work with incomplete reference genome, our experiments also revealed that kRISP-meR performs better for some cases over a reference-based guide RNA designer tool, especially when there are repeated regions in the genome. With so many genome with so many repeats, this is a highlighted upside of using kRISP-meR.

kRISP-meR is also designed to generate variant-aware guide RNAs by utilizing the information content available with in the sequenced reads about a particular individual. This is a very important aspect because using the guide RNAs generated using GRCh38 may be potentially harmful for a random person. The ability to design variant-aware guide RNAs has the potential to effectively make CRISPR experiments safer and more robust.

In our experiments, we always used the sequencing reads collected from the Illumina technology, rather than using long read technology such as PacBio or MinION. This is because of two reasons. Firstly, the sequencing error rates for PacBio and MinION are higher compared to Illumina. Secondly, the coverage for Illumina is higher compared to the other two. At the same time, the Illumina reads are cheaper to generate and more available than other long read generating technologies. Therefore, it makes sense to prepare the tool for Illumina reads. Moreover, the motivation to use longer reads is not to capture the variant information, rather to collapse the gaps in the existing assembly of a genome. In any case, if the long read generating technology can generate reads with higher depth and reduce the error rate, in principle, kRISP-meR will be able to process those reads as well.

In implementation, kRISP-meR does not use the reference genome at all. We implemented kRISP-meR this way to ensure that when we are polishing the target string in a variant aware fashion, we are not incorporating any bias from the original reference assembly. At the same time, this implementation strengthens the novelty and contributions of kRISP-meR in operating at the absence of the reference genome.

6.2 Future works

We plan to test the effectiveness of kRISP-meR by experimenting with de novo assemblies of humans. Our plan is to take a sequence from the reference human genome (GRCh38) as the target, but use the Illumina sequencing reads collected from a bio-sample. At the same time, we plan to run GuideScan (or some other reference based guide RNA designer) using the reference genome (GRCh38). After that, we plan to show that the list generated by kRISP-meR performs better in the de novo assembly of the bio-sample than does the guide RNAs identified by GuideScan (which uses the information available only with in GRCh38).

Besides doing these experiments, we also plan to extend kRISP-meR from its current state in multiple directions. First, we plan to implement the tool so that it supports all other cutting enzymes besides Cas9. This involves handling a variable length guide RNA (instead of a fixed 20-nt sequence for CRISPR/Cas9), determining the cutting probability for the variable length, and handling variable length PAM sequences. kRISP-meR already handles multiple alternate PAM sequences, the other mentioned features are to be implemented in future.

We also plan to analyze the pipeline that identifies the personalized target sequence. This is important because kRISP-meR needs a target sequence to work with, and this target sequence is usually taken from the reference genome of the same organism or a related organism. Therefore, it is crucial that this pipeline functions accurately.

Another major direction to explore is working with the sequencing reads in the target site as well. Currently, we utilize the sequencing reads of the genome, and take the target sequence as an input. Methods can be developed to drop the requirement of taking a target sequence as input. In order to do so, we have to model the sequencing reads coming from a particular location in genome and incorporate those accordingly in the definition of the score.

6.3 Availability

kRISP-meR is publicly available <https://github.com/mahmudhera/kRISP-meR>, with an installation guide and usage instructions. The preprint of the work is available here [133] for a quicker read.

References

- [1] R. Barrangou and L. A. Marraffini, “CRISPR-Cas systems: prokaryotes upgrade to adaptive immunity,” *Molecular cell*, vol. 54, no. 2, pp. 234–244, 2014.
- [2] P. Horvath and R. Barrangou, “CRISPR/Cas, the immune system of bacteria and archaea,” *Science*, vol. 327, no. 5962, pp. 167–170, 2010.
- [3] “CRISPR: a game changing genetic engineering technique.” [Online; accessed 21-February-2020].
- [4] “EM Algorithm Explained in One Picture.” <https://www.datasciencecentral.com/profiles/blogs/em-algorithm-explained-in-one-picture>. [Online; accessed 4-January-2020].
- [5] S. L. Salzberg, A. M. Phillippy, A. Zimin, D. Puiu, T. Magoc, S. Koren, T. J. Treangen, M. C. Schatz, A. L. Delcher, M. Roberts, et al., “GAGE: A critical evaluation of genome assemblies and assembly algorithms,” *Genome research*, vol. 22, no. 3, pp. 557–567, 2012.
- [6] Wikipedia contributors, “Escherichia coli — Wikipedia, the free encyclopedia,” 2020. [Online; accessed 19-August-2020].
- [7] “What is DNA? - Genetics.” <https://ghr.nlm.nih.gov/primer/basics/dna>. [Online; accessed 21-February-2020].
- [8] . G. P. Consortium et al., “An integrated map of genetic variation from 1,092 human genomes,” *Nature*, vol. 491, no. 7422, p. 56, 2012.
- [9] . G. P. Consortium et al., “A global reference for human genetic variation,” *Nature*, vol. 526, no. 7571, pp. 68–74, 2015.
- [10] S. Alwin, M. B. Gere, E. Guhl, K. Effertz, C. F. Barbas III, D. J. Segal, M. D. Weitzman, and T. Cathomen, “Custom zinc-finger nucleases for use in human cells,” *Molecular Therapy*, vol. 12, no. 4, pp. 610–617, 2005.

- [11] T. Cermak, E. L. Doyle, M. Christian, L. Wang, Y. Zhang, C. Schmidt, J. A. Baller, N. V. Somia, A. J. Bogdanove, and D. F. Voytas, “Efficient design and assembly of custom TALEN and other TAL effector-based constructs for DNA targeting,” *Nucleic acids research*, vol. 39, no. 12, pp. e82–e82, 2011.
- [12] M. Jinek, K. Chylinski, I. Fonfara, M. Hauer, J. A. Doudna, and E. Charpentier, “A programmable dual-RNA-guided DNA endonuclease in adaptive bacterial immunity,” *science*, vol. 337, no. 6096, pp. 816–821, 2012.
- [13] J. A. Doudna and E. Charpentier, “The new frontier of genome engineering with CRISPR-Cas9,” *Science*, vol. 346, no. 6213, pp. 1077–1087, 2014.
- [14] “Genome Editing.” <https://www.genome.gov/about-genomics/policy-issues/what-is-Genome-Editing>. [Online; accessed 21-February-2020].
- [15] E. H. Kaji and J. M. Leiden, “Gene and stem cell therapies,” *Jama*, vol. 285, no. 5, pp. 545–550, 2001.
- [16] Wikipedia contributors, “Gene therapy — Wikipedia, the free encyclopedia,” 2020. [Online; accessed 21-February-2020].
- [17] “A Cell Therapy Untested in Humans Saves a Baby With Cancer.” [Online; accessed 21-February-2020].
- [18] J. Couzin-Frankel, “Baby’s leukemia recedes after novel cell therapy,” 2015.
- [19] A. Hammond, R. Galizi, K. Kyrou, A. Simoni, C. Siniscalchi, D. Katsanos, M. Gribble, D. Baker, E. Marois, S. Russell, et al., “A CRISPR-Cas9 gene drive system targeting female reproduction in the malaria mosquito vector *Anopheles gambiae*,” *Nature biotechnology*, vol. 34, no. 1, p. 78, 2016.
- [20] “Mutant mosquitoes: Can gene editing kill off malaria?.” [Online; accessed 21-February-2020].
- [21] R. Barrangou and J. A. Doudna, “Applications of CRISPR technologies in research and beyond,” *Nature biotechnology*, vol. 34, no. 9, p. 933, 2016.
- [22] J. R. Prado, G. Segers, T. Voelker, D. Carson, R. Dobert, J. Phillips, K. Cook, C. Cornejo, J. Monken, L. Grapes, et al., “Genetically engineered crops: from idea to product,” *Annual review of plant biology*, vol. 65, 2014.
- [23] Y. Zhang, K. Massel, I. D. Godwin, and C. Gao, “Applications and potential of genome editing in crop improvement,” *Genome biology*, vol. 19, no. 1, p. 210, 2018.

- [24] W. M. Ainley, L. Sastry-Dent, M. E. Welter, M. G. Murray, B. Zeitler, R. Amora, D. R. Corbin, R. R. Miles, N. L. Arnold, T. L. Strange, et al., "Trait stacking via targeted genome editing," *Plant Biotechnology Journal*, vol. 11, no. 9, pp. 1126–1134, 2013.
- [25] C. Cantos, P. Francisco, K. R. Trijatmiko, I. Slamet-Loedin, and P. K. Chadha-Mohanty, "Identification of "safe harbor" loci in indica rice genome by harnessing the property of zinc-finger nucleases to induce DNA damage and repair," *Frontiers in plant science*, vol. 5, p. 302, 2014.
- [26] T. Li, B. Liu, M. H. Spalding, D. P. Weeks, and B. Yang, "High-efficiency TALEN-based gene editing produces disease-resistant rice," *Nature biotechnology*, vol. 30, no. 5, p. 390, 2012.
- [27] Q. Shan, Y. Zhang, K. Chen, K. Zhang, and C. Gao, "Creation of fragrant rice by targeted knockout of the Os BADH 2 gene using TALEN technology," *Plant biotechnology journal*, vol. 13, no. 6, pp. 791–800, 2015.
- [28] W. Haun, A. Coffman, B. M. Clasen, Z. L. Demorest, A. Lowy, E. Ray, A. Retterath, T. Stoddard, A. Juillerat, F. Cedrone, et al., "Improved soybean oil quality by targeted mutagenesis of the fatty acid desaturase 2 gene family," *Plant biotechnology journal*, vol. 12, no. 7, pp. 934–940, 2014.
- [29] B. Kannan, J. H. Jung, G. W. Moxley, S.-M. Lee, and F. Altpeter, "TALEN-mediated targeted mutagenesis of more than 100 COMT copies/alleles in highly polyploid sugarcane improves saccharification efficiency without compromising biomass yield," *Plant biotechnology journal*, vol. 16, no. 4, pp. 856–866, 2018.
- [30] M. Li, X. Li, Z. Zhou, P. Wu, M. Fang, X. Pan, Q. Lin, W. Luo, G. Wu, and H. Li, "Reassessment of the four yield-related genes Gn1a, DEP1, GS3, and IPA1 in rice using a CRISPR/Cas9 system," *Frontiers in plant science*, vol. 7, p. 377, 2016.
- [31] Y. Zhang, D. Li, D. Zhang, X. Zhao, X. Cao, L. Dong, J. Liu, K. Chen, H. Zhang, C. Gao, et al., "Analysis of the functions of Ta GW 2 homoeologs in wheat grain weight and protein content traits," *The Plant Journal*, vol. 94, no. 5, pp. 857–866, 2018.
- [32] M. Andersson, H. Turesson, A. Nicolia, A.-S. Fält, M. Samuelsson, and P. Hofvander, "Efficient targeted multiallelic mutagenesis in tetraploid potato (*Solanum tuberosum*) by transient CRISPR-Cas9 expression in protoplasts," *Plant cell reports*, vol. 36, no. 1, pp. 117–128, 2017.

- [33] A. Ortigosa, S. Gimenez-Ibanez, N. Leonhardt, and R. Solano, “Design of a bacterial speck resistant tomato by CRISPR/Cas9-mediated editing of Sl JAZ 2,” *Plant biotechnology journal*, vol. 17, no. 3, pp. 665–673, 2019.
- [34] J. Chandrasekaran, M. Brumin, D. Wolf, D. Leibman, C. Klap, M. Pearlsman, A. Sherman, T. Arazi, and A. Gal-On, “Development of broad virus resistance in non-transgenic cucumber using CRISPR/Cas9 technology,” *Molecular plant pathology*, vol. 17, no. 7, pp. 1140–1153, 2016.
- [35] J. Martínez-Fortún, D. W. Phillips, and H. D. Jones, “Potential impact of genome editing in world agriculture,” *Emerging Topics in Life Sciences*, vol. 1, no. 2, pp. 117–133, 2017.
- [36] Y. Ran, Z. Liang, and C. Gao, “Current and future editing reagent delivery systems for plant genome editing,” *Science China Life Sciences*, vol. 60, no. 5, pp. 490–505, 2017.
- [37] G. Marçais and C. Kingsford, “A fast, lock-free approach for efficient parallel counting of occurrences of k-mers,” *Bioinformatics*, vol. 27, no. 6, pp. 764–770, 2011.
- [38] B. Langmead and S. L. Salzberg, “Fast gapped-read alignment with Bowtie 2,” *Nature Methods*, vol. 9, pp. 357–359, Mar. 2012.
- [39] H. Li, B. Handsaker, A. Wysoker, T. Fennell, J. Ruan, N. Homer, G. Marth, G. Abecasis, and R. Durbin, “The sequence alignment/map format and SAMtools,” *Bioinformatics*, vol. 25, no. 16, pp. 2078–2079, 2009.
- [40] B. J. Walker, T. Abeel, T. Shea, M. Priest, A. Abouelliel, S. Sakthikumar, C. A. Cuomo, Q. Zeng, J. Wortman, S. K. Young, et al., “Pilon: an integrated tool for comprehensive microbial variant detection and genome assembly improvement,” *PloS one*, vol. 9, no. 11, p. e112963, 2014.
- [41] J. G. Doench, N. Fusi, M. Sullender, M. Hegde, E. W. Vaimberg, K. F. Donovan, I. Smith, Z. Tothova, C. Wilen, R. Orchard, et al., “Optimized sgRNA design to maximize activity and minimize off-target effects of CRISPR-Cas9,” *Nature biotechnology*, vol. 34, no. 2, p. 184, 2016.
- [42] R. Barrangou, “The roles of CRISPR–Cas systems in adaptive immunity and beyond,” *Current opinion in immunology*, vol. 32, pp. 36–41, 2015.

- [43] F. Zhang, Y. Wen, and X. Guo, "CRISPR/Cas9 for genome editing: progress, implications and challenges," *Human molecular genetics*, vol. 23, no. R1, pp. R40–R46, 2014.
- [44] E. Kits, A. E. Kits, and R. Santa Cruz, "CRISPR-Cas9, TALENs and ZFNs-the battle in gene editing,"
- [45] P. D. Hsu, E. S. Lander, and F. Zhang, "Development and applications of CRISPR-Cas9 for genome engineering," *Cell*, vol. 157, no. 6, pp. 1262–1278, 2014.
- [46] R. Barrangou, C. Fremaux, H. Deveau, M. Richards, P. Boyaval, S. Moineau, D. A. Romero, and P. Horvath, "CRISPR provides acquired resistance against viruses in prokaryotes," *Science*, vol. 315, no. 5819, pp. 1709–1712, 2007.
- [47] L. A. Marraffini and E. J. Sontheimer, "CRISPR interference limits horizontal gene transfer in staphylococci by targeting DNA," *science*, vol. 322, no. 5909, pp. 1843–1845, 2008.
- [48] P. Mohanraju, K. S. Makarova, B. Zetsche, F. Zhang, E. V. Koonin, and J. Van der Oost, "Diverse evolutionary roots and mechanistic variations of the CRISPR-Cas systems," *Science*, vol. 353, no. 6299, p. aad5147, 2016.
- [49] F. Hille, H. Richter, S. P. Wong, M. Bratovič, S. Ressel, and E. Charpentier, "The biology of CRISPR-Cas: backward and forward," *Cell*, vol. 172, no. 6, pp. 1239–1259, 2018.
- [50] D. Carroll, "Progress and prospects: zinc-finger nucleases as gene therapy agents," *Gene therapy*, vol. 15, no. 22, pp. 1463–1468, 2008.
- [51] M. Mushtaq, J. A. Bhat, Z. A. Mir, A. Sakina, S. Ali, A. K. Singh, A. Tyagi, R. K. Salgotra, A. A. Dar, and R. Bhat, "CRISPR/Cas approach: A new way of looking at plant-abiotic interactions," *Journal of plant physiology*, vol. 224, pp. 156–162, 2018.
- [52] "'breakthrough of the year: Crispr makes the cut'" <https://www.sciencemag.org/news/2015/12/and-science-s-2015-breakthrough-year>. [Online; accessed 25-Jan-2020].
- [53] H. Ledford, "CRISPR: gene editing is just the beginning," *Nature News*, vol. 531, no. 7593, p. 156, 2016.
- [54] J. Travis, "Breakthrough of the Year: CRISPR makes the cut," *Science Magazine*, 2015.

- [55] H. Ledford, “CRISPR, the disruptor,” *Nature*, vol. 522, no. 7554, p. 20, 2015.
- [56] L. Cong, F. A. Ran, D. Cox, S. Lin, R. Barretto, N. Habib, P. D. Hsu, X. Wu, W. Jiang, L. A. Marraffini, et al., “Multiplex genome engineering using CRISPR/Cas systems,” *Science*, vol. 339, no. 6121, pp. 819–823, 2013.
- [57] P. Mali, L. Yang, K. M. Esvelt, J. Aach, M. Guell, J. E. DiCarlo, J. E. Norville, and G. M. Church, “RNA-guided human genome engineering via Cas9,” *Science*, vol. 339, no. 6121, pp. 823–826, 2013.
- [58] E. Hartenian and J. G. Doench, “Genetic screens and functional genomics using CRISPR/Cas9 technology,” *The FEBS journal*, vol. 282, no. 8, pp. 1383–1393, 2015.
- [59] O. Shalem, N. E. Sanjana, E. Hartenian, X. Shi, D. A. Scott, T. S. Mikkelsen, D. Heckl, B. L. Ebert, D. E. Root, J. G. Doench, et al., “Genome-scale CRISPR-Cas9 knockout screening in human cells,” *Science*, vol. 343, no. 6166, pp. 84–87, 2014.
- [60] T. Wang, J. J. Wei, D. M. Sabatini, and E. S. Lander, “Genetic screens in human cells using the CRISPR-Cas9 system,” *Science*, vol. 343, no. 6166, pp. 80–84, 2014.
- [61] H. Koike-Yusa, Y. Li, E.-P. Tan, M. D. C. Velasco-Herrera, and K. Yusa, “Genome-wide recessive genetic screening in mammalian cells with a lentiviral CRISPR-guide RNA library,” *Nature biotechnology*, vol. 32, no. 3, p. 267, 2014.
- [62] R. Kaminski, R. Bella, C. Yin, J. Otte, P. Ferrante, H. E. Gendelman, H. Li, R. Booze, J. Gordon, W. Hu, et al., “Excision of HIV-1 DNA by gene editing: a proof-of-concept in vivo study,” *Gene therapy*, vol. 23, no. 8, pp. 690–695, 2016.
- [63] R. Kaminski, Y. Chen, T. Fischer, E. Tedaldi, A. Napoli, Y. Zhang, J. Karn, W. Hu, and K. Khalili, “Elimination of HIV-1 genomes from human T-lymphoid cells by CRISPR/Cas9 gene editing,” *Scientific Reports*, vol. 6, no. 22555, pp. 1–15, 2016.
- [64] A. Jain, G. Zode, R. B. Kasetti, F. A. Ran, W. Yan, T. P. Sharma, K. Bugge, C. C. Searby, J. H. Fingert, F. Zhang, et al., “CRISPR-Cas9-based treatment of myocilin-associated glaucoma,” *Proceedings of the National Academy of Sciences*, vol. 114, no. 42, pp. 11199–11204, 2017.

- [65] P. A. Papatianos and N. Windbichler, “Redkmer: An assembly-free pipeline for the identification of abundant and specific X-chromosome target sequences for X-shredding by CRISPR endonucleases,” *The CRISPR journal*, vol. 1, no. 1, pp. 88–98, 2018.
- [66] C. M. Margulies, F. Buquicchio, G. Schiroli, V. Vavassori, J. Gori, K. Gogi, F. Harbinski, L. Naldini, P. Genovese, C. Albright, et al., “Efficient Targeted Integration in Human T Cells with CRISPR-Cas9 for the Treatment of X-Linked Hyper-IgM Syndrome,” in *Molecular Therapy*, vol. 26(5), pp. 237–237, 2018.
- [67] S. A. Shah, S. Erdmann, F. J. Mojica, and R. A. Garrett, “Protospacer recognition motifs: mixed identities and functional diversity,” *RNA biology*, vol. 10, no. 5, pp. 891–899, 2013.
- [68] F. J. Mojica, C. Díez-Villaseñor, J. García-Martínez, and C. Almendros, “Short motif sequences determine the targets of the prokaryotic CRISPR defence system,” *Microbiology*, vol. 155, no. 3, pp. 733–740, 2009.
- [69] S. H. Sternberg, S. Redding, M. Jinek, E. C. Greene, and J. A. Doudna, “DNA interrogation by the CRISPR RNA-guided endonuclease Cas9,” *Nature*, vol. 507, no. 7490, pp. 62–67, 2014.
- [70] B. Farboud and B. J. Meyer, “Dramatic enhancement of genome editing by CRISPR/Cas9 through improved guide RNA design,” *Genetics*, vol. 199, no. 4, pp. 959–971, 2015.
- [71] K. E. Garbison, B. A. Heinz, M. E. Lajiness, J. R. Weidner, G. S. Sittampalam, M. E. Lajiness, S. P. Protocol, S. D. Lamore, C. W. Scott, M. F. Peters, et al., “Assay Guidance Manual,” Eli Lilly & Company and the National Center for Advancing Translational Sciences, 2004.
- [72] L. Cong and F. Zhang, “Genome engineering using CRISPR-Cas9 system,” in *Chromosomal Mutagenesis*, pp. 197–217, Springer, 2015.
- [73] Y. Fu, J. A. Foden, C. Khayter, M. L. Maeder, D. Reyon, J. K. Joung, and J. D. Sander, “High-frequency off-target mutagenesis induced by CRISPR-Cas nucleases in human cells,” *Nature biotechnology*, vol. 31, no. 9, pp. 822–826, 2013.
- [74] S. W. Cho, S. Kim, Y. Kim, J. Kweon, H. S. Kim, S. Bae, and J.-S. Kim, “Analysis of off-target effects of CRISPR/Cas-derived RNA-guided endonucleases and nickases,” *Genome research*, vol. 24, no. 1, pp. 132–141, 2014.

- [75] A. Veres, B. S. Gosis, Q. Ding, R. Collins, A. Ragavendran, H. Brand, S. Erdin, C. A. Cowan, M. E. Talkowski, and K. Musunuru, “Low incidence of off-target mutations in individual CRISPR-Cas9 and TALEN targeted human stem cell clones detected by whole-genome sequencing,” *Cell stem cell*, vol. 15, no. 1, pp. 27–30, 2014.
- [76] S. E. Mohr, Y. Hu, B. Ewen-Campen, B. E. Housden, R. Viswanatha, and N. Perrimon, “CRISPR guide RNA design for research applications,” *The FEBS journal*, vol. 283, no. 17, pp. 3232–3238, 2016.
- [77] J. G. Doench, E. Hartenian, D. B. Graham, Z. Tothova, M. Hegde, I. Smith, M. Sullender, B. L. Ebert, R. J. Xavier, and D. E. Root, “Rational design of highly active sgRNAs for CRISPR-Cas9-mediated gene inactivation,” *Nature biotechnology*, vol. 32, no. 12, p. 1262, 2014.
- [78] M. K. Rahman and M. S. Rahman, “CRISPRpred: A flexible and efficient tool for sgRNAs on-target activity prediction in CRISPR/Cas9 systems,” *PloS one*, vol. 12, no. 8, p. e0181943, 2017.
- [79] H. Xu, T. Xiao, C.-H. Chen, W. Li, C. A. Meyer, Q. Wu, D. Wu, L. Cong, F. Zhang, J. S. Liu, et al., “Sequence determinants of improved CRISPR sgRNA design,” *Genome research*, vol. 25, no. 8, pp. 1147–1157, 2015.
- [80] H. Zou and T. Hastie, “Regularization and variable selection via the elastic net,” *Journal of the royal statistical society: series B (statistical methodology)*, vol. 67, no. 2, pp. 301–320, 2005.
- [81] M. A. Moreno-Mateos, C. E. Vejnar, J.-D. Beaudoin, J. P. Fernandez, E. K. Mis, M. K. Khokha, and A. J. Giraldez, “CRISPRscan: designing highly efficient sgRNAs for CRISPR/Cas9 targeting in vivo,” *Nature methods*, vol. 12, no. 10, p. 982, 2015.
- [82] R. Chari, P. Mali, M. Moosburner, and G. M. Church, “Unraveling CRISPR-Cas9 genome engineering parameters via a library-on-library approach,” *Nature methods*, vol. 12, no. 9, pp. 823–826, 2015.
- [83] N. Wong, W. Liu, and X. Wang, “WU-CRISPR: characteristics of functional guide RNAs for the CRISPR/Cas9 system,” *Genome biology*, vol. 16, no. 1, p. 218, 2015.
- [84] P. F. Kuan, S. Powers, S. He, K. Li, X. Zhao, and B. Huang, “A systematic evaluation of nucleotide properties for CRISPR sgRNA design,” *BMC bioinformatics*, vol. 18, no. 1, p. 297, 2017.

- [85] M. Labuhn, F. F. Adams, M. Ng, S. Knoess, A. Schambach, E. M. Charpentier, A. Schwarzer, J. L. Mateo, J.-H. Klusmann, and D. Heckl, “Refined sgRNA efficacy prediction improves large-and small-scale CRISPR–Cas9 applications,” *Nucleic acids research*, vol. 46, no. 3, pp. 1375–1385, 2018.
- [86] P. D. Hsu, D. A. Scott, J. A. Weinstein, F. A. Ran, S. Konermann, V. Agarwala, Y. Li, E. J. Fine, X. Wu, O. Shalem, et al., “DNA targeting specificity of RNA-guided Cas9 nucleases,” *Nature biotechnology*, vol. 31, no. 9, pp. 827–832, 2013.
- [87] Y. Cui, J. Xu, M. Cheng, X. Liao, and S. Peng, “Review of CRISPR/Cas9 sgRNA designing tools,” *Interdisciplinary Sciences: Computational Life Sciences*, vol. 10, no. 2, pp. 455–465, 2018.
- [88] M. Stemmer, T. Thumberger, M. del Sol Keyer, J. Wittbrodt, and J. L. Mateo, “CCTop: an intuitive, flexible and reliable CRISPR/Cas9 target prediction tool,” *PloS one*, vol. 10, no. 4, 2015.
- [89] B. J. Mendoza and C. T. Trinh, “Enhanced guide-RNA design and targeting analysis for precise CRISPR genome editing of single and consortia of industrially relevant and non-model organisms,” *Bioinformatics*, vol. 34, no. 1, pp. 16–23, 2018.
- [90] R. Singh, C. Kuscu, A. Quinlan, Y. Qi, and M. Adli, “Cas9-chromatin binding information enables more accurate CRISPR off-target prediction,” *Nucleic acids research*, vol. 43, no. 18, pp. e118–e118, 2015.
- [91] F. Heigwer, G. Kerr, and M. Boutros, “E-CRISP: fast CRISPR target site identification,” *Nature methods*, vol. 11, no. 2, pp. 122–123, 2014.
- [92] T. G. Montague, J. M. Cruz, J. A. Gagnon, G. M. Church, and E. Valen, “CHOPCHOP: a CRISPR/Cas9 and TALEN web tool for genome editing,” *Nucleic acids research*, vol. 42, no. W1, pp. W401–W407, 2014.
- [93] T. J. Cradick, P. Qiu, C. M. Lee, E. J. Fine, and G. Bao, “COSMID: a web-based tool for identifying and validating CRISPR/Cas off-target sites,” *Molecular Therapy-Nucleic Acids*, vol. 3, p. e214, 2014.
- [94] Y. Naito, K. Hino, H. Bono, and K. Ui-Tei, “CRISPRdirect: software for designing CRISPR/Cas guide RNA with reduced off-target sites,” *Bioinformatics*, vol. 31, no. 7, pp. 1120–1123, 2015.

- [95] I. Grissa, G. Vergnaud, and C. Pourcel, “CRISPRFinder: a web tool to identify clustered regularly interspaced short palindromic repeats,” *Nucleic acids research*, vol. 35, no. suppl_2, pp. W52–W57, 2007.
- [96] Y. Lei, L. Lu, H.-Y. Liu, S. Li, F. Xing, and L.-L. Chen, “CRISPR-P: a web tool for synthetic single-guide RNA design of CRISPR-system in plants,” *Molecular plant*, vol. 7, no. 9, pp. 1494–1496, 2014.
- [97] L. J. Zhu, B. R. Holmes, N. Aronin, and M. H. Brodsky, “CRISPRseek: a bioconductor package to identify target-specific guide RNAs for CRISPR-Cas9 genome-editing systems,” *PloS one*, vol. 9, no. 9, 2014.
- [98] S. Xie, B. Shen, C. Zhang, X. Huang, and Y. Zhang, “sgRNAs9: a software package for designing CRISPR sgRNA and evaluating potential off-target cleavage sites,” *PloS one*, vol. 9, no. 6, 2014.
- [99] S. Bae, J. Park, and J.-S. Kim, “Cas-OFFinder: a fast and versatile algorithm that searches for potential off-target sites of Cas9 RNA-guided endonucleases,” *Bioinformatics*, vol. 30, no. 10, pp. 1473–1475, 2014.
- [100] L. J. Zhu, “Overview of guide RNA design tools for CRISPR-Cas9 genome editing technology,” *Frontiers in Biology*, vol. 10, no. 4, pp. 289–296, 2015.
- [101] J. Listgarten, M. Weinstein, B. P. Kleinstiver, A. A. Sousa, J. K. Joung, J. Crawford, K. Gao, L. Hoang, M. Elibol, J. G. Doench, et al., “Prediction of off-target activities for the end-to-end design of CRISPR guide RNAs,” *Nature biomedical engineering*, vol. 2, no. 1, pp. 38–47, 2018.
- [102] G. Chuai, H. Ma, J. Yan, M. Chen, N. Hong, D. Xue, C. Zhou, C. Zhu, K. Chen, B. Duan, et al., “DeepCRISPR: optimized CRISPR guide RNA design by deep learning,” *Genome biology*, vol. 19, no. 1, p. 80, 2018.
- [103] A. R. Perez, Y. Pritykin, J. A. Vidigal, S. Chhangawala, L. Zamparo, C. S. Leslie, and A. Ventura, “GuideScan software for improved single and paired CRISPR guide RNA design,” *Nature Biotechnology*, vol. 35, no. 4, pp. 347–349, 2017.
- [104] T. Li, S. Wang, F. Luo, F.-X. Wu, and J. Wang, “MultiGuideScan: a multi-processing tool for designing CRISPR guide RNA libraries,” *Bioinformatics*, 2019.
- [105] M. Haeussler, K. Schönig, H. Eckert, A. Eschstruth, J. Mianné, J.-B. Renaud, S. Schneider-Maunoury, A. Shkumatava, L. Teboul, J. Kent, et al., “Evaluation of off-target and on-target scoring algorithms and integration into the guide RNA selection tool CRISPOR,” *Genome biology*, vol. 17, no. 1, p. 148, 2016.

- [106] J. Sun, H. Liu, J. Liu, S. Cheng, Y. Peng, Q. Zhang, J. Yan, H.-J. Liu, and L.-L. Chen, “CRISPR-Local: a local single-guide RNA (sgRNA) design tool for non-reference plant genomes,” *Bioinformatics*, 2018.
- [107] M. R. Vollger, P. C. Dishuck, M. Sorensen, A. E. Welch, V. Dang, M. L. Dougherty, T. A. Graves-Lindsay, R. K. Wilson, M. J. Chaisson, and E. E. Eichler, “Long-read sequence and assembly of segmental duplications,” *Nature methods*, vol. 16, no. 1, p. 88, 2019.
- [108] R. Sachidanandam, D. Weissman, S. C. Schmidt, J. M. Kakol, L. D. Stein, G. Marth, S. Sherry, J. C. Mullikin, B. J. Mortimore, D. L. Willey, et al., “A map of human genome sequence variation containing 1.42 million single nucleotide polymorphisms,” *Nature*, vol. 409, no. 6822, pp. 928–934, 2001.
- [109] K. C. Keough, S. Lyalina, M. P. Olvera, S. Whalen, B. R. Conklin, and K. S. Pollard, “AlleleAnalyzer: a tool for personalized and allele-specific sgRNA design,” *Genome biology*, vol. 20, no. 1, pp. 1–9, 2019.
- [110] A. L. Jacquin, D. T. Odom, and M. Lukk, “Crisflash: open-source software to generate CRISPR guide RNAs against genomes annotated with individual variation,” *Bioinformatics*, 2019.
- [111] S. Cancellieri, M. C. Canver, N. Bombieri, R. Giugno, and L. Pinello, “CRISPRitz: rapid, high-throughput, and variant-aware in silico off-target site identification for CRISPR genome editing,” *Bioinformatics*, 2019.
- [112] Wikipedia contributors, “Read (biology) — Wikipedia, the free encyclopedia,” 2019. [Online; accessed 2-January-2020].
- [113] S. Chhangawala, G. Rudy, C. E. Mason, and J. A. Rosenfeld, “The impact of read length on quantification of differentially expressed genes and splice junction detection,” *Genome biology*, vol. 16, no. 1, p. 131, 2015.
- [114] A. Conesa, P. Madrigal, S. Tarazona, D. Gomez-Cabrero, A. Cervera, A. McPherson, M. W. Szczesniak, D. J. Gaffney, L. L. Elo, X. Zhang, et al., “A survey of best practices for RNA-seq data analysis,” *Genome biology*, vol. 17, no. 1, p. 13, 2016.
- [115] Wikipedia contributors, “K-mer — Wikipedia, the free encyclopedia.” <https://en.wikipedia.org/w/index.php?title=K-mer&oldid=932249716>, 2019. [Online; accessed 27-December-2019].

- [116] R. De La Briandais, “File searching using variable length keys,” in Papers presented at the the March 3-5, 1959, western joint computer conference, pp. 295–298, 1959.
- [117] Wikipedia contributors, “Trie — Wikipedia, the free encyclopedia,” 2019. [Online; accessed 2-January-2020].
- [118] Geeks for geeks, 2019. [Online; accessed 2-January-2020].
- [119] Wikipedia contributors, “Maximum likelihood estimation — Wikipedia, the free encyclopedia,” 2019. [Online; accessed 2-January-2020].
- [120] “EM Algorithm (Expectation-maximization): Simple Definition.” <https://www.statisticshowto.datasciencecentral.com/em-algorithm-expectation-maximization>. [Online; accessed 4-January-2020].
- [121] Wikipedia contributors, “Poisson distribution — Wikipedia, the free encyclopedia.” https://en.wikipedia.org/w/index.php?title=Poisson_distribution&oldid=933271323, 2019. [Online; accessed 2-January-2020].
- [122] A. Rahman, I. Hallgrímsson, M. Eisen, and L. Pachter, “Association mapping from sequencing reads using k-mers,” *Elife*, vol. 7, p. e32920, 2018.
- [123] G. Xuan, W. Zhang, and P. Chai, “EM algorithms of Gaussian mixture model and hidden Markov model,” in *Proceedings 2001 International Conference on Image Processing (Cat. No. 01CH37205)*, vol. 1, pp. 145–148, IEEE, 2001.
- [124] M. R. Gupta, Y. Chen, et al., “Theory and use of the EM algorithm,” *Foundations and Trends® in Signal Processing*, vol. 4, no. 3, pp. 223–296, 2011.
- [125] A. El-Baz and G. Gimel’farb, “EM based approximation of empirical distributions with linear combinations of discrete Gaussians,” in *2007 IEEE International Conference on Image Processing*, vol. 4, pp. IV–373, IEEE, 2007.
- [126] Wikipedia contributors, “Zero-truncated poisson distribution — Wikipedia, the free encyclopedia,” 2019. [Online; accessed 13-February-2020].
- [127] S. L. Salzberg, A. M. Phillippy, A. Zimin, D. Puiu, T. Magoc, S. Koren, T. J. Treangen, M. C. Schatz, A. L. Delcher, M. Roberts, et al., “GAGE: A critical evaluation of genome assemblies and assembly algorithms,” *Genome research*, vol. 22, no. 3, pp. 557–567, 2012.

-
- [128] R. A. Hoskins, J. W. Carlson, K. H. Wan, S. Park, I. Mendez, S. E. Galle, B. W. Booth, B. D. Pfeiffer, R. A. George, R. Svirskas, et al., “The release 6 reference sequence of the drosophila melanogaster genome,” *Genome research*, vol. 25, no. 3, pp. 445–458, 2015.
- [129] B. Muller, P. Mollon, E. Santiago-Allexant, F. Javerliat, and G. Kaneko, “In-depth comparison of library pooling strategies for multiplexing bacterial species in ngs,” *Diagnostic microbiology and infectious disease*, vol. 95, no. 1, pp. 28–33, 2019.
- [130] “Escherichia coli str. k-12 substr. mg1655, complete genome,” 2020. [Online; accessed 19-August-2020].
- [131] “Escherichia coli atcc 8739 (e. coli),” 2020. [Online; accessed 19-August-2020].
- [132] H. Li, “wgsim-Read simulator for next generation sequencing,” Github Repository, 2011.
- [133] M. R. Hera, A. Rahman, and A. Rahman, “kRISP-meR: A Reference-free Guide-RNA Design Tool for CRISPR/Cas9,” *bioRxiv*, p. 869115, 2019.

Generated using Postgraduate Thesis L^AT_EX Template, Version 1.02. Department of
Computer Science and Engineering, Bangladesh University of Engineering and
Technology, Dhaka, Bangladesh.

This thesis was generated on Thursday 20th August, 2020 at 1:49pm.