Predicting the Functional Effects of Human Short Variations Using Hidden Markov Models

Mingming Liu

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Computer Science and Applications

Liqing Zhang, Chair
Lenwood S. Heath
Layne T. Watson
Xiaowei Wu
Jianjun Hu

May, 2015
Blacksburg, Virginia

Keywords: Genetic variation, Indel, SNP, Hidden Markov Model

Predicting the Functional Effects of Human Short Variations Using Hidden Markov Models

Mingming Liu

(ABSTRACT)

With the development of sequencing technologies, more and more sequence variants are available for investigation. Different types of variants in the human genome have been identified, including single nucleotide polymorphisms (SNPs), short insertions and deletions (indels), and large structural variations such as large duplications and deletions. Of great research interest is the functional effects of these variants. Although many programs have been developed to predict the effect of SNPs, few can be used to predict the effect of indels or multiple variants, such as multiple SNPs, multiple indels, or a combination of both. Moreover, fine grained prediction of the functional outcome of variants is not available. To address these limitations, we developed a prediction framework, HMMvar, to predict the functional effects of coding variants (SNPs or indels), using profile hidden Markov models (HMMs). Based on HMMvar, we proposed HMMvar-multi to explore the joint effects of multiple variants in the same gene. For fine grained functional outcome prediction, we developed HMMvar-func to computationally define and predict four types of functional outcome of a variant: gain, loss, switch, and conservation of function.

This work was supported by National Institutes of Health via grant AI085091.

Dedication

To my parents, Jianhua Liu and Guizhen Kang;

To my husband, Zhibing Xu;

To my little girl, Tina.

Acknowledgments

First of all, I would like to give my most sincere thanks to my advisor, Dr. Liqing Zhang, for her invaluable guidance, advice, patience, and help during the development of this work. She is not only a good advisor in academic research, but also a good friend who has enriched my Ph.D. life.

I would like to thank my committee members: Drs. Lenwood Heath, Jianjun Hu, Layne Watson, and Xiaowei Wu, for their kindness, insight and suggestions along the way, especially Dr. Watson, who was very patient, correcting all my grammar mistakes, helping me improve my English writing greatly. I also thank Dr. Zach Adelman and Dr. David Bevan, who gave me useful suggestions on our collaboration projects.

I also thank my friends in Torgersen Hall: Hong Tran, Mohammad Shabbir Hasan, Zhiye Li, Fei Li, Shuo Wang, and Md Ahsanur Rahman for being there with me and having interesting conversations. I am glad I have known them and I enjoyed their company.

Last but not the least, I thank my parents-in-law for coming to Blacksburg to support my study and life. You have left your work and everything to come here. Without your encouragement and supports, I would have never gone so far.

Contents

1 Introduction **1**

1.1 Background 2

1.1.1 Prediction Methods for the Effects of Variants 4

1.1.2 Genetic Variants in Cancer Genomes 10

1.2 Problems and Motivation 11

1.3 Notation and Preliminaries 13

1.3.1 Introduction of HMM 13

1.3.2 Profile HMM 20

1.4 Summary 21

2 Quantitative Prediction of the Effect of Genetic Variants **24**

2.1 Introduction 24

2.2 HMMvar: Functional Effects of Variants in Coding Regions 26

2.3 Results 30

2.3.1 Predictions on Indels 30

2.3.2 Comparison with Other Tools 35

2.3.3 A Case Study of Individual Protein: TP53 37

3 Functional Effects of Multiple Variants **42**

3.1 Introduction 42

3.2 HMMvar-multi: Joint Effect of Multiple Variants 44

3.2.1 Determine the Haplotypes 44

3.2.2 Compensatory Indel Sets 46

3.3 Results 49

3.3.1 Multiple Variants in the 1000 Genomes Project Data 49

3.3.2 Compensatory Indels in TP53 and PTEN 51

3.3.3 A Case Study: Cardiovascular Disease 52

4 Predicting the Functional Outcome of Variants **56**

4.1 Introduction 56

4.2 HMMvar-func: Predicting the Functional Outcome of Variants 59

4.2.1 Building Multiple HMMs 59

4.2.2 Clustering of MSA 60

4.2.3 Classification by HMMvar Scores 64

4.3 Results 67

4.3.1 Thyrotropin Receptor Gene Mutations 67

4.3.2 Application in Cancer Mutations 71

4.3.3 Predictions on SoF Mutations 76

4.3.4 Mutagenesis Analysis 78

4.3.5 The Conversion Between HPES and TEAS 79

5 Conclusion **82**

References **84**

A The Cumulative Conjecture **102**

B A Prediction System for Interpretation of Genomic Variants **107**

List of Figures

1.1 The workflow of genome projects. 3

1.2 An example of an HMM. 15

1.3 General structure of a profile HMM. D: delete state, M: match state, I: insert state. 20

1.4 An example of building profile HMM. (a) A small DNA multiple alignment. The match states are marked with numbers corresponding to the four positions in the model shown in (c) and the insertion states are shown in the box. (b) The observed count of emission states and transition states. (c) The profile HMM structure for the MSA in (a). 22

2.1 Pipeline of HMMvar prediction for variants in coding regions. 28

2.2 HMMvar score distribution of the dbSNP data set. (a) Histogram of HMMvar scores for disease associated indels and nondisease associated indels. (b) Distribution of sample means of HMMvar scores from the two categories (LSDB and nonLSDB). 32

2.3 Distributions of HMMvar scores for different types of variants. 33

2.4 The relationship between the HMMvar score and the position of an artificially introduced variant. 34

2.5 Compare HMMvar prediction with SIFT-indel prediction on dbSNP indel data set. Distributions of HMMvar of indels that are predicted as damaging (left) and neutral (right) by SIFT-indel. 36

2.6 HMMvar and Provean score distributions and mean/error bars of TP53 mutations binned into 15 classes in terms of transactivity level. (a) HMMvar score distribution of the 15 classes (x-axis represents the 15 classes based on the median of transactivity levels). (b) Provean score distribution of the 15 classes. (c) Mean along with error bar of HMMvar scores in each class. (d) Mean along with error bar of Provean scores in each class. 38

2.7 ROC curve and standard error of the HMMvar score and the Provean score. (a) ROC curve of the Provean score and the HMMvar score to distinguish "nonfunctional" and "partly functional" classes from "functional" and "supertrans" classes. (b) Standard error of the mean of Provean and HMMvar scores in the 15 transactivity level classes. 39

2.8 The HMMvar score of TP53 variants grouped by SIFT SNP prediction. . . . 40

3.1 An example of variant classification in terms of genotypes. The colored sticks on the gene represent variants at different locations. Colors represent different classes of variants. The format $v1 : T; A|A$ means variant $v1$'s ancestral allele is T and the genotype is $A|A$, the same as other variants. The boxes on the transcripts represent exon regions. The gene and the transcripts share the same coordinate system. 46

3.2 Zygosity of disease causing single (a–b) or multiple mutations (c–e). The cross and circle indicate different mutations; the different thicknesses indicate different alleles of a mutation and the thicker one indicates a disease-causing allele; the boxes represent a gene. 48

3.3 Comparison between variant set score (black) and single variant score (red). 50

3.4 Allele frequency distribution of SNP variants in CM sets and nonCM sets. . 51

3.5 Scatter plot of HMMvar score of a single variant versus the median HMMvar score of the corresponding compensatory indel sets for the TP53 gene and the PTEN gene. The red line is $y = x$. (a) TP53 compensatory indels. (b) PTEN compensatory indels. The red solid circle marks the COSMIC variant with ID 428080. 53

4.1 The consequences of LoF, SoF, GoF and CoF mutations (M). The normal gene is indicated by blue box and mutated gene is indicated by orange box. The original functions are represented by blue circles and the new functions are represented by green circles. 58

4.2 Flowchart of the classification procedure (the dashed line represents the wild type sequence). 61

4.3 Decision tree for predicting the functional outcome of variants with hard classification. 65

4.4 The probability combination rule for the classification of mutations 67

4.5 Receiver operating curve (ROC) for prediction of TSHR mutations (sensitivity is with respect to GoF; the AUC is 0.613.). 69

4.6 The cost function with respect to the number of clusters for K-means clustering with 100 runs for each k, $n = 162$. 71

4.7 The performance of HMMvar-func based on K-means clustering or CEO clustering. (100 random initial guesses are evaluated for K-means clustering on the TSHR data set with $k = 4$ and $u = 2.7$. The red diamond points represent the corresponding performance of the CEO clustering.) 72

4.8 Confidence score distribution for different predicted mutation types: (a) confidence score for EGFR mutations, (b) confidence score for BRAF mutations. 73

4.9 Several selected BRAF subfamilies from the clustering result. The last column is the position of the mutation. The wild type query sequence is in the target cluster C_0 and indicated in the dashed red box. Subfamilies are separated by blue borders. 75

4.10 The transactivity level of the gene TP53 in different predicted mutation groups. 76

4.11 Distance tree of the MAP2K subfamilies. Colors indicate different subfamilies. The minimum score S_i^x is calculated from C_{19}. C_0 is the target cluster. C_{28} is an example subfamily that the mutant protein could switch to. The leaves are protein sequences. Two sequences are merged according to the BLOSUM62 matrix by averaging the substitution distance over all the positions in the MSA. 78

4.12 Predictions based on SWISS-PROT mutagenesis data set. 79

4.13 The target cluster of TEAS. The blue shaded columns are the mutation positions and the target sequence is indicated by the red dashed box. 80

5.1 A hypothetical system for trained classifier based method for predicting the functional effect of variants in noncoding regions (HGMD: Human Gene Mutation Database; 1KG: 1000 Genomes Project; MAF: minor allele frequency; FunSeq: [52]; GWAVA: [82]. The red curve may represent deleterious mutations and the blue curve neutral). 83

A.1 A section of the profile HMM of the homologous sequence of βMHC protein.Two variants (D906G and L908L) are marked. 106

B.1 The system architecture 108

List of Tables

1.1 Some example programs for predicting the effect of SNPs and indels for coding and noncoding regions. 5

1.2 Three basic HMM problems. 16

2.1 Data Set from dbSNP 31

2.2 Comparison between HMMvar prediction and SIFT-indel prediction with dbSNP indel data set 35

2.3 Data Set from ENSEMBL 37

3.1 Variants sets related to gene ABCB5 47

3.2 Data Description 52

3.3 Scoring multiple mutations in βMHC and MyBP-C genes 54

4.1 The confusion matrix of the prediction results for the TSHr data set 69

4.2 The comparison of CEO and K-means 70

4.3 Prediction of oncogenic mutations. 72

4.4 SoF Mutations 77

4.5 The mutations that convert TEAS to HPES 80

Chapter 1

Introduction

The DNA sequence of the entire human genome was deciphered by the Human Genome Project (HGP) in 2004 [15]. Since then, a few thousand human genomes have been sequenced and a large number of genetic variations have been identified. According to the 1000 Genomes Project, the 2535 sequenced genomes harbor approximately 80 million genetic variants, of which about 77.4 million are SNPs and 2.2 million indels. Genetic variations are the molecular substrates of evolutionary processes. Moreover, they are critical to the understanding of the relationship between phenotypes and genotypes. However, not all variants are decisive factors for phenotypes or diseases. As it is both expensive and time consuming to investigate the effect of each variant empirically, computational prediction of the effect of variants is usually performed first to filter the variants and identify possible targets for downstream empirical studies. The next section reviews available computational resources for predicting the effect of variants and limitations of the tools.

1.1 Background

Sequencing technology allows sequencing of the whole genome at increasingly lower cost. The ultimate goal of genome sequencing is to characterize the individual genomic landscape, identify mutations relevant to disease diagnosis and therapy, and provide insight into the connection between genotypes and phenotypes [72].

The basic workflow of genome projects, from data generation to prediction, is shown in Figure 1.1. First, whole genome sequencing produces hundreds of millions of short sequences, known as "short reads". These reads are then aligned to and compared with a reference genome. Next, differences between reads and the reference genome are identified as genetic variants. Various tools are applied to predict the effect of the variants (mostly limited to SNPs and short indels). Based on the prediction, variants are filtered, prioritized, and selected for downstream empirical validation. Different computational methods and tools are developed at each step [72]. Our work focuses on the functional prediction module as shown in Figure 1.1.

There are different types of mutations. The two most common types are single nucleotide polymorphisms (SNPs) and insertions and deletions (indels). A SNP is a variation at a single nucleotide position that occurs with some frequency in a population. An indel is an insertion or deletion of a segment of the DNA sequence. From the Phase I data of the 1000 Genomes Project, 96% of the identified variants are SNPs, and 3% indels. The Phase III data uncovered over 79 million variants, of which over 77 million are SNPs and over 2 million indels. Most of the earlier studies focus on annotation of SNPs, since they are relatively easy to identify and analyze. Indels are analyzed by a relatively small number of tools. The common way of annotation in most of the tools is to provide links to the public variant databases that are relevant to the gene where the variant occurred. Besides database links, they also employ different approaches, ranging from simple sequence-based analysis to the evaluation of the structural impact on proteins. The result is a classification of the mutations into either the neutral or the deleterious class. The programs sometimes provide more fine grained

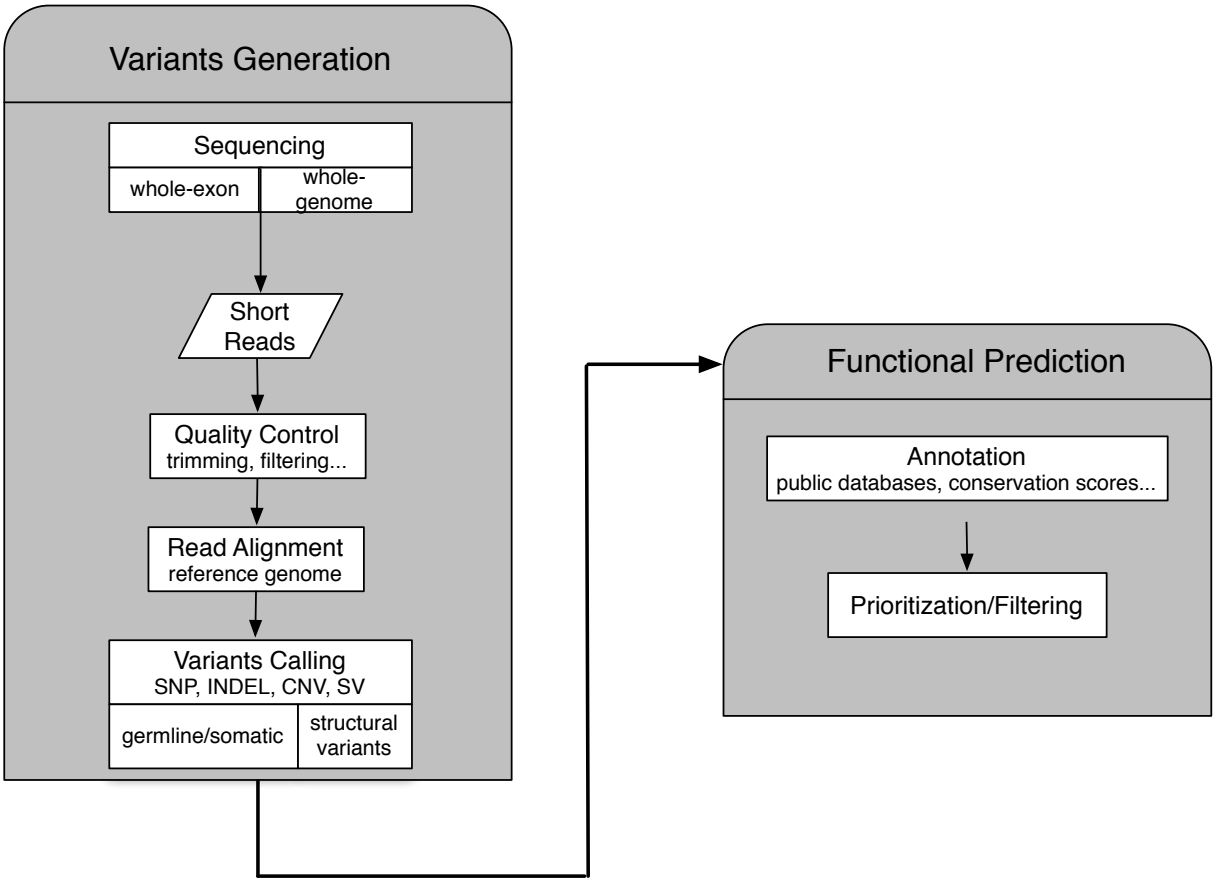


Figure 1.1: The workflow of genome projects.

risk classes or quantitative scores to reflect the likelihood of a deleterious effect. The effect prediction of variants in protein coding regions has been extensively studied, because the effect is straightforward to interpret considering the formation of the corresponding proteins. Comparatively, interpreting variants in noncoding regions is more difficult and fewer tools are available. The protein sequence based prediction methods are typically grouped into two categories [19]: constraint based predictor and trained classifier.

Evolutionary conservation is usually considered as the best measurement of harmfulness of variants [19]. Many computational methods for predicting the harm of variants exploit the fact that sequences observed in living organisms are those that have not been removed by natural selection. Conserved positions evolve very slowly because mutations in these sites

tend to be removed by purifying selection. Mutations in these sites are therefore likely to be deleterious. Constraint based approaches define a metric, such as entropy, variance, or Jensen-Shannon divergence, to quantitatively measure the conservation of the positions and make predictions based on the metric.

By contrast, a trained classifier identifies many potentially relevant properties of the variants, and heuristically combines them to train a classifier to optimally differentiate a set of true positives and negatives. In contrast to constraint based methods, it is not easy to interpret the results using a trained classifier. In addition, they are likely to be biased by the so-called "gold standard" data sets, which are contaminated with erroneous annotations and are not representative of the general population of true positives and negatives [19]. However, the trained classifier methods have the advantages of being specifically tunable to the desired task (e.g., predicting disease causality) and incorporating many sources of information without requiring a detailed understanding of how that information is relevant.

1.1.1 Prediction Methods for the Effects of Variants

Acquiring data is no longer the top priority as in earlier days. Sequencing of the first full human genome took over a decade and roughly three billion dollars [66]. Now, it is possible to sequence a human genome for only $1000 in less than two weeks [66]. However, as getting data becomes easier, new problems arise. Scientists are struggling to figure out the best way to extract useful information from the sea of data. Once variants are identified, the real challenge is interpreting what effect the variants have biologically. A great number of annotation tools have been developed and have already made great contribution to the prioritization of variants that may have relevant functional effects.

Table 1.1 shows typical methods for predicting the harmfulness of SNPs and indels with respect to coding and noncoding regions. Methods for predicting the effects of SNPs in coding regions are much more abundant than others. Methods fall into two groups: constraint based methods and trained classifiers. Constraint based methods are typically based on

Table 1.1: Some example programs for predicting the effect of SNPs and indels for coding and noncoding regions.

	Coding	Noncoding
SNP	SIFT [75], MAPP [96], Polyphen2 [1] Provean [14], FATHMM [87], PhastCons [89] GERP [20]	RegulomeDB [8], FunSeq [52] GWAVA [82], CADD [53] FATHMM-MKL [88]
Indel	SIFT-indel [45], Provean [14], CADD [53]	FunSeq [52], CADD [53]

evolutionary conservation theory and assign a score to the variant indicating the degree of harm. Trained classifier methods classify a variant as neutral or deleterious. The methods listed in Table 1.1 are discussed below.

Sorting Intolerant From Tolerant (SIFT) [75] is a sequence homology-based tool that sorts amino acid substitutions to identify those that may potentially have a phenotypic effect. SIFT works as follows: (1) take a protein sequence of interest as query; (2) search protein databases for homologous sequences; (3) build a multiple sequence alignment (MSA); (4) calculate a conservation value and scaled probability for each position; (5) classify the substitution as a neutral or deleterious mutation based on a predefined cutoff.

In step (4), the probability of amino acid a appearing at position c is estimated by the formula,

$$P_{ca} = \frac{N_c}{N_c + B_c} * g_{ca} + \frac{B_c}{N_c + B_c} * f_{ca}, \tag{1.1}$$

where N_c is the total number of sequences, B_c the number of pseudocounts based on amino acid frequencies and a predetermined matrix (BLOSUM62) [75], g_{ca} the frequency of amino acid a at position c, and f_{ca} the frequency of pseudocounts from a 13-component Dirichlet mixture model [91]. SIFT achieves good performance predicting the effects of amino acid substitutions. However, it can only be applied to SNPs or to indels that preserve the open reading frame of coding regions.

Multivariate Analysis of Protein Polymorphism (MAPP) [96] is another constraint based method. In MAPP, an MSA of closely related protein sequences and their phylogenetic tree is used to derive six matrices that reflect the evolutionary constraints on the 20 amino acids. Each matrix is built on the basis of a single physicochemical property of the amino acids. The results for the six physicochemical properties are then de-correlated to compute a single score that measures the violation of constraints across all properties. MAPP applies to only coding variants similar to SIFT.

Another popular method is the trained classifier Polymorphism Phenotyping v2 (Polyphen2) [1], which predicts possible impacts of an amino acid substitution on the structure and function of a protein by considering straightforward physical and chemical properties. The naïve Bayes classifier is trained on two data sets that contain both deleterious and neutral amino acid changes. Eight sequence based and three structure based features, most of which compare a given property of the wild type amino acid and its mutant, are used to build the classifier.

Protein variation effect analyzer (Provean) [14] is a recently proposed evolutionary conservation based method for predicting the functional effects of both indels and SNPs. Provean collects a set of sequences homologous to the gene or protein of interest and clusters them into different supporting sets to calculate the Provean score based on the delta alignment score. The delta alignment score of a protein sequence Q with its variation v is defined by the change in semiglobal alignment score from the wild type to the mutant type protein sequence with respect to a subject sequence S obtained from homologous sequence search. Specifically,

$$\Delta(Q, v, S) \quad = \quad A(Q', S) - A(Q, S), \tag{1.2}$$

where Q' is the variant sequence of Q caused by v and $A(P_1, P_2)$ the semiglobal alignment score between two protein sequences P_1 and P_2, computed based on a given amino acid substitution matrix (e.g., BLOSUM62) and gap penalties.

Functional Analysis Through Hidden Markov Models (FATHMM) [87] is a program and

server for functional analysis of genetic variants using HMMs. FATHMM uses profile HMMs [28] to capture the position information in an MSA. The HMMs used include an *ad hoc* initial HMM built from a set of homologous sequences to a query protein sequence of interest. Related protein domain HMMs from Pfam [78] and Superfamily [40] are also used to include domain specific information that might be missed by the *ad hoc* HMM. The most informative HMM is then selected and used to score the SNPs. FATHMM claims that it outperforms other traditional algorithms, such as SIFT [75], Polyphen2 [1], and Panther [100]. However, FATHMM is restricted to the functional prediction of SNPs.

The prediction methods discussed above are based on protein sequences. PhastCons [89] and Genomic Evolutionary Rate Profiling (GERP) [20] are two methods based on nucleotide sequences. PhastCons identifies conserved elements and/or computes conservation scores, given an MSA and a phylogenetic hidden Markov model (phylo-HMM). By default, a phylo-HMM is assumed to have two states: a "conserved" state and a "nonconserved" state. GERP estimates a constraint for each aligned column and identifies constrained elements in MSAs by quantifying substitution deficits. These deficits represent the substitutions that did not occur due to the functional constraint of the elements, but would have occurred if the elements were neutral.

Compared to the few dozens of prediction programs for SNPs, methods for evaluating the harmfulness of indels are rare. To our knowledge, only the methods described in [45], [109], and [14] can be used to predict the effects of indels in coding regions. SIFT-indel [45] predicts the effect of frameshift indels by a decision tree classifier. This method extracts 20 features for the prediction of frameshift indels via a feature selection procedure. The algorithm achieves maximum accuracy (84%) when four features are selected, including (1) fraction of affected conserved DNA bases, (2) maximum indel location relative to all the transcripts, (3) maximum fraction of affected conserved amino acids across all transcripts, and (4) minimum distance of the indel to the exon boundary of all affected transcripts. Though results from the decision tree classifier are easy to interpret, the predictive power is limited because the classifier only applies to frameshift indels, which account for a tiny proportion (about

0.05%) of all indels. In addition, SIFT-indel only provides a qualitative prediction, either "damaging" or "neutral", rather than a quantitative measurement. Another indel functional effect prediction method is an evolutionary conservation based approach for both coding and noncoding regions [109]. It follows the constraint based method framework, using an MSA and calculating the information change at each column to define the conservation score. Provean [14] is also capable of predicting the effects of small indels in coding regions.

Annotating variants outside gene regions is difficult, although many genetic variants associated with complex traits or diseases lie in noncoding regions of the genome. Researchers are starting to shift their attention to noncoding regions. The Encyclopedia of DNA elements (ENCODE) [16] project launched in 2003 has produced a large amount of data that helps to annotate the noncoding functional elements in the human genome. Much progress has been made towards predicting or annotating noncoding variants. Because evolutionary conservation based methods are not as effective for noncoding regions as coding regions [84], current algorithms are mostly based on machine learning methods using a set of properties extracted from various databases.

RegulomeDB [8] is a database that annotates SNPs with known and predicted regulatory elements in the intergenic regions of the human genome. Known and predicted regulatory DNA elements include regions of deoxyribonuclease (DNase) hypersensitivity, binding sites of transcription factors, and promoter regions that have been biochemically characterized. Data sources in RegulomeDB include public data sets from Gene Expression Omnibus (GEO) [7], the ENCODE [16] project, and the literature. The input to RegulomeDB is a set of SNPs (dbSNP IDs or SNP locations) or a chromosome region, and the output includes the RegulomeDB scores, related annotations, and other external links. Therefore, RegulomeDB is an integrated annotation database used to identify noncoding functional elements that might be affected by the variant, based on a score indicating the harmfulness of the variant. The method is also limited to only predicting the functional effects of SNPs.

Funseq [52] is a workflow pipeline that identifies candidate driver mutations (mutations caus-

ing cancer) by whether or not the variants are in the so-called "sensitive" or "ultrasensitive" regions. Using the 1000 Genomes data, Funseq defines "sensitive" and "ultrasensitive" regions. A variant in an "ultrasensitive" region has a high probability to be a candidate driver variant. A variant is assigned a score between 0 and 6, indicating at which stage the variant is filtered out, with 6 corresponding to the most deleterious effect. However, the granularity of the scores is coarse because many variants may be filtered out at the same stage with the same scores. Although the authors claim that the pipeline can handle indels and structural variants, the indels discussed in the study are mainly in exon regions.

Genome-wide annotation of variants (GWAVA) [82] uses a modified random forest classifier to integrate various genomic and epigenomic annotations and assigns a score measuring the functional effects of variants. Because the random forest classifier is robust to features that are not predictive, the feature selection step is not necessary. In addition, the random forest classifier is able to identify relatively important features. The authors claim that annotations across the genome, such as G+C content, evolutionary conservation, DNase I hypersensitivity, histone modifications, and distance to the nearest transcription start site, were among the most informative features for the classification. GWAVA is mainly for SNPs in noncoding regions.

Combined Annotation Dependent Depletion (CADD) [53] is a tool for scoring the harm of SNPs and indels in the human genome. CADD integrates many diverse annotations into a single score (C score) using a support vector machine. CADD is the most comprehensive method so far, and can score the functional effects of both SNPs and indels in both coding and noncoding regions.

FATHMM-MKL [88] is the most recently developed method that predicts the functional effects of SNPs in both coding and noncoding regions. An extension of FATHMM [87] that uses a constraint based method to predict SNPs in coding regions, FATHMM-MKL uses machine learning instead. With various types of features, a multiple kernel learning method is used and the weights are assigned to different kernels (group of features). Then a

kernel based classifier (support vector machine) is implemented. The results show that the conservation information is the most informative feature and FATHMM-MKL achieves an area under curve (AUC) over 0.9 with this feature alone.

As we can see, prediction of the effect of variants in noncoding regions is done mainly through machine learning strategies and based on the available annotation of variants.

1.1.2 Genetic Variants in Cancer Genomes

Mutations can be divided into two categories, germline mutations and somatic mutations, according to the types of cells where mutations occur. Germline mutations are mutations that occur in sperms and/or eggs, so they may not only affect the individual that carries them, but also its offspring. Somatic mutations are mutations that occur in other types of cells, and therefore may affect only the individual that carries them. For cancer studies, identifying somatic mutations that may trigger cancer cell growth, also known as driver mutations, is often the main focus.

Prediction methods for driver mutations share commonality with prediction methods for germline mutations. Some methods are designed specifically to prioritize cancer driver mutations, including CHASM [9], CRAVAT [25], and transFIC [38]. For CHASM, a random forest classifier is trained on a curated set of driver mutations derived from COSMIC [34] and randomly simulated passenger mutations. CHASM uses 86 diverse features (available at the SNVBox database [107]), including physiochemical properties of amino acid residues, scores derived from MSAs of protein or DNA sequences, region based amino acid sequence composition, predicted properties of local protein structure, and annotations from the UniProtKB [18] feature tables. CRAVAT is a toolkit that combines three methods, CHASM [9], VEST [10], and SnvGet [107], to rank germline variants, somatic mutations, and relative genes in terms of their importance. TransFIC [38] improves other prediction methods by taking into account the baseline tolerance of genes. It has been shown that the transFIC scores perform better than the original scores in distinguishing driver variants from passen-

ger variants in various data sets [38]. The major difference between the cancer specific driver mutation prediction and germline variant effect prediction lies in the feature sets they use, many of which are in common.

Databases built specifically for cancer somatic mutations are also important resources. Example of these databases include COSMIC [34], IntOGen-mutations [39], DCC-ICGC [46], and TCGA [99]. COSMIC curates millions of somatic mutations in various cancers. IntOGen is an integrative oncogenomics cancer browser. IntOGen-mutations provides a pipeline for cancer researchers to identify mutations in genes and pathways. The DCC-ICGC data portal provides tools for visualizing, querying, and downloading data from associated projects. TCGA contains both germline and somatic mutations collected from large-scale genome sequencing projects, supported by the National Cancer Institute (NCI) and the National Human Genome Research Institute (NHGRI).

1.2 Problems and Motivation

Although many tools have been developed to predict the effect of genetic variants, there are limitations to these methods which prevent us from further interpreting the variants. In the following, we identified four major problems.

Problem 1: Predict the functional effects of various types of variants.

Most of the previous studies focus on SNPs rather than on indels. Indels are the second most common type of genetic variation in humans. Indels can occur in coding or noncoding regions of the genome. Indels in coding regions, especially frameshift ones, are expected to have large effects on protein functions, since they may change the reading frame of a gene, leading to changes in amino acids and protein functions.

Problem 2: Predict the joint effects of multiple variants.

The functional effects of a single variant, especially a SNP, have been studied extensively,

from sequence analysis to structure analysis. However, the effects of multiple variants have been largely ignored in the previous studies. It becomes much more complex when considering multiple variants together, which may involve genotype analysis, haplotype prediction, and linkage disequilibrium (LD) calculation. Complex diseases or cancers are likely to be associated with multiple mutations in one gene or multiple genes. Therefore it is important to be able to predict the aggregate effect of multiple variants.

Problem 3: Predict the fine grained functional outcome.

Many tools and databases predict and annotate the effects of variants in terms of whether they are neutral or deleterious to protein function. However, there is a need for methods that not only identify neutral or deleterious mutations, but also provide fine grained prediction of the cellular outcome resulting from the mutations, such as loss, switch, gain, or conservation of function. Earlier work addressed prediction of the functional types of variants [57, 67, 81] by identifying activating variants, but did not provide a precise computational definition for all the four types. Fine grained prediction of the functional outcome will contribute to better understanding of the molecular mechanisms of disease and cancer causing mutations.

Problem 4: Predict the effect of noncoding variants.

Studies reveal that the majority of genetic variants associated with diseases and complex traits lie in noncoding regions of the genome, and many of them some distance away from the nearest protein-coding loci [42]. Therefore many variants that affect the risk of common and complex diseases are likely to exert their effects by altering the regulation of genes rather than by directly affecting gene and protein functions. However, to date, much effort for annotating functional variants has been focused on variants that directly affect coding regions, such as missense and nonsense mutations, and those that affect transcript splicing signals [19]. In September 2003, the research consortium ENCODE [16] was launched, aiming to identify all functional elements in the human genome. The ENCODE project enables researchers to get relevant properties of variants in noncoding regions. However, it is still not clear which elements or combination of elements should be used to evaluate the harm of the variants in

noncoding regions.

For variants in coding regions, constraint based methods can be applied easily and effectively because protein sequences tend to be rather conserved throughout evolution across different species. Regulatory elements are known to have large interspecies differences [84], which implies that conservation might be a less important criterion when interpreting variants in regulatory regions. Thus, it might not be sufficient to use the conservation information alone for predicting the effects of variants in noncoding regions. Existing methods use different combinations of annotations, such as transcription factor binding sites, cytogenetic bands, regions of enhancers, and repressors or promoters. Identification of critical factors affected by mutations is a major concern. Effects of regulatory variants are also harder to interpret. Moreover, the same variant may have different effects in different tissues, developmental stages, or individuals with different genetic backgrounds. Incomplete annotation in noncoding regions is also an obstacle for developing reliable methods.

Human genetic variation is the engine for human evolution. It is important to understand the origin, development, and effects of genetic variants to associate the genotype with phenotype, which is one of the fundamental questions in genetic studies. We aim to provide solutions to the first three problems, while the last problem will be addressed in future studies.

1.3 Notation and Preliminaries

In this section, we introduce the essential concepts and algorithms for HMMs used in our studies for functional prediction of genetic variants.

1.3.1 Introduction of HMM

A discrete Markov stochastic process is a sequence of random variables $\xi(t_i)$ indexed by times $t_0 < t_1 < t_2 < \cdots$ with values in the set of states $X = \{x_0, x_1, x_2, \ldots\}$. The state

transition probability

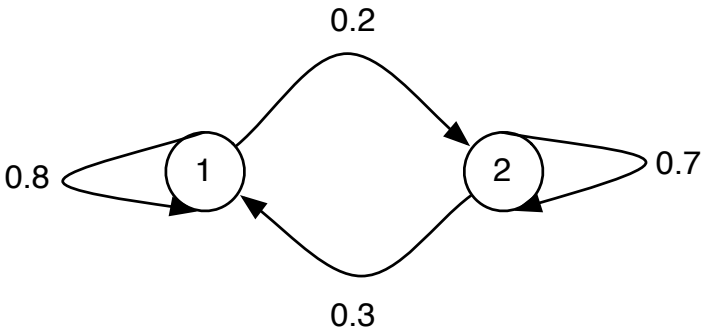
$$a_{ij}(t_k) = P\big(\xi(t_{k+1}) = x_j \mid \xi(t_k) = x_i\big)$$

has the (Markov) property

$$P\big(\xi(t_{k+1}) = x_j \mid \xi(t_k)\big) = P\big(\xi(t_{k+1}) = x_j \mid \xi(t_0), \ldots, \xi(t_k)\big).$$

The process is called *stationary* if the $a_{ij}(t)$ are independent of time t (i.e., constants). The state set X can be finite or infinite. The vector π defines the probability distribution of the initial state $\xi(t_0)$. A hidden Markov model (HMM) is a stationary discrete Markov stochastic process where the states $q_k = \xi(t_k)$ are not directly observable (hidden), but each state q_k produces (emits) an observation $O_k = \eta(q_k)$ with values in the (finite or infinite) set $V = \{0,$ 1, 2, $\ldots\}$, and $b_{jv} \equiv b_j(v)$ is the probability that the state x_j produces (emits) observation $v \in V$.

To understand the basic idea of HMM, let us start with a coin toss experiment with two biased coins. One only observes a sequence of results of coin tosses, e.g., $O_0, O_1, \ldots, O_{T-1} =$ H, T,. . . ,H, where 'H' stands for head and 'T' stands for tail. A graphical representation of this model is shown in Figure 1.2. There are two hidden states, '1' and '2', representing Coin 1 and Coin 2, respectively. Each hidden state determines the probability distribution of the two sides of a coin, which are observable.



State	1	2
P(H)	0.6	0.45
P(T)	0.4	0.55

Figure 1.2: An example of an HMM.

Let

$$
\begin{aligned}
T &= \text{length of the observed sequence,} \\
N &= \text{number of states in the model,} \\
M &= \text{number of observed symbols,} \\
X &= \{x_0, x_1, \ldots, x_{N-1}\} = \text{distinct hidden states of the Markov process,} \\
V &= \{0, 1, \ldots, M-1\} = \text{the set of possible observations,} \\
A &= \{a_{ij}\} = \text{state transition probabilities,} \\
B &= \{b_j(o)\} = \text{emission probability matrix,} \\
\pi &= [\pi_0, \pi_1, \ldots, \pi_{N-1}] = \text{the initial state distribution,} \\
O &= (O_0, O_1, \ldots, O_{T-1}) = \text{the observed sequence, and} \\
Q &= (q_0, q_1, \ldots, q_{T-1}) = \text{the hidden states sequence.}
\end{aligned}
$$

Each element a_{ij} of the transition probability matrix A is the transition probability from

Table 1.2: Three basic HMM problems.

Problem	Algorithm	Complexity
Q1: Evaluaton $P(O_0O_1,\ldots,O_{T-1} \mid \lambda)$	Forward-backward	$O(TN^2)$
Q2: Inference $\arg\max_Q P(Q = q_0q_1,\ldots,q_{T-1} \mid O_0O_2,\ldots,O_{T-1})$	Viterbi decoding	$O(TN^2)$
Q3: Learning $\arg\max_\lambda P(O_0O_1,\ldots,O_{T-1} \mid \lambda)$	Baum-Welch	$O(TN^2)^*$

N: the number of states; T: the number of time steps; *: for one iteration.

state x_i to state x_j, and $a_{ij} = P(q_{t+1} = x_j \mid q_t = x_i)$. Each element $b_j(o)$ of the emission probability matrix B is the probability of observing o at the hidden state x_j. The observation at time t is $O_t \in V$, for $t = 0, 1, \ldots, T-1$, and the hidden state at time t is $q_t \in X$, for $t = 0, 1, \ldots, T-1$. The vector π represents the a priori probability of each hidden state. Formally, an HMM λ can be expressed as $\lambda = (X, V, A, B, \pi)$.

There are three basic HMM problems (Table 1.2): (1). What is the probability of observing a sequence of observations $O_0O_1,\ldots,O_{T-1}$? (2). Given a sequence of observations $O_0O_1,\ldots,O_{T-1}$, what is the optimal hidden state sequence and the associated probability? (3). Given $O_0O_1,\ldots,O_{T-1}$, what is the maximum likelihood HMM generating the sequence of observations? Algorithms have been developed to solve each of these questions. Considering the time complexity, dynamic programming is usually used in these algorithms.

For the first problem, to calculate the conditional probability $P(O_0O_2,\ldots,O_{T-1} \mid \lambda)$, we need to sum over all possible state sequences Q that can generate the observation sequence:

$$\begin{aligned} P(O_0O_1,\ldots,O_{T-1} \mid \lambda) &= \sum_Q P(O, Q \mid \lambda) \\ &= \sum_Q P(O \mid Q, \lambda)P(Q) \\ &= \sum_Q \pi_{\overline{0}} b_{\overline{0}}(O_0) a_{\overline{0}\,\overline{1}} b_{\overline{1}}(O_1) \cdots a_{\overline{T-2},\overline{T-1}} b_{\overline{T-1}}(O_{T-1}), \end{aligned} \tag{1.3}$$

where $x_{\overline{k}}$ denotes the state q_k in a given sequence Q. The joint probability $P(O, Q \mid \lambda)$

of the observation sequence and a specific path is easy to calculate, however, the time complexity for explicitly calculating the total probability using each of the possible paths is $O(TN^T)$, which is generally infeasible for real problems. To eliminate repeated calculations, for $t = 0, 1, \ldots, T-1$ and $i = 0, 1, \ldots, N-1$, define

$$\alpha_t(i) \;=\; P(O_0, O_1, \ldots, O_t, q_t = x_i \mid \lambda), \tag{1.4}$$

the probability of the partial observation sequence up to time t, where the hidden state q_t is x_i at time t. The total probability can be calculated using dynamic programming as follows.

1. Let $\alpha_0(i) = \pi_i b_i(O_0)$, for $i = 0, 1, \ldots, N-1$.

2. For $t = 1, 2, \ldots, T-1$ and $i = 0, 1, \ldots, N-1$, compute

$$\alpha_t(i) \;=\; \big(\sum_{j=0}^{N-1} \alpha_{t-1}(j) a_{ji}\big) b_i(O_t). \tag{1.5}$$

3. The total probability is

$$P(O \mid \lambda) \;=\; \sum_{i=0}^{N-1} \alpha_{T-1}(i). \tag{1.6}$$

For the second problem, the goal is to find the optimal highest scoring path that generates the observation sequence. A brute force solution

$$\begin{aligned} & \arg\max_Q P(Q \mid O_0, O_1, \ldots, O_{T-1}) \\ = \;& \arg\max_Q \frac{P(Q) P(O_0, O_1, \ldots, O_{T-1} \mid Q)}{P(O_0, O_1, \ldots, O_{T-1})} \\ = \;& \arg\max_Q P(Q) P(O_0, O_1, \ldots, O_{T-1} \mid Q) \end{aligned} \tag{1.7}$$

enumerates all possible paths, but usually the Viterbi algorithm [35] is used for this problem. For $t = 1, \ldots, T-1$ and $i = 0, 1, \ldots, N-1$, define

$$\sigma_t(i) \;=\; \max_{Q_{t-1}} P(q_t = x_i, O_0, O_1, \ldots, O_t \mid \lambda), \tag{1.8}$$

$$Q_t(i) \;=\; \big(\arg\max_{Q_{t-1}} P(q_t = x_i, O_0, O_1, \ldots, O_t \mid \lambda), x_i\big). \tag{1.9}$$

where $\sigma_t(i)$ is the maximum probability of any path with length t and having the state x_i at time t, $Q_t(i)$ a path that archives the maximum probability. An optimal path is computed by:

1. Let $\sigma_0(i) = \pi_i b_i(O_0)$ and $Q_0(i) = (x_i)$ for $i = 0, 1, \ldots, N-1$.

2. For $t = 1, \ldots, T-1$ and $j = 0, 1, \ldots, N-1$, calculate

$$\begin{aligned} \sigma_t(j) &= \max_{0 \le i \le N-1} \sigma_{t-1}(i) a_{ij} b_j(O_t), && (1.10) \\ i^* &= \arg\max_{0 \le i \le N-1} \sigma_{t-1}(i) a_{ij} b_j(O_t), && (1.11) \\ Q_t(j) &= \big(Q_{t-1}(i^*), x_j\big). && (1.12) \end{aligned}$$

3. Let $j^* = \arg\max_{0 \le j \le N-1} \sigma_{T-1}(j)$. The optimal probability is $\sigma_{T-1}(j^*)$ and a corresponding path is $Q_{T-1}(j^*)$.

The third problem is to train the HMM to best fit the observations. The sizes of the matrices (N and M) are fixed but the elements of A, B, and π are to be determined, subject to the row stochastic condition. Usually the expectation maximization (EM) algorithm [24] is used to iteratively estimate the parameters. A more detailed description can be found in [93]. For $t = 0, 1, \ldots, T-1$ and $i = 0, 1, \ldots, N-1$, define

$$\beta_t(i) = P(O_{t+1}, O_{t+2}, \ldots, O_{T-1} \mid q_t = x_i, \lambda). \quad (1.13)$$

Similar to the calculation of $\alpha_t(i)$, $\beta_t(i)$ can be computed recursively as follows.

1. Let $\beta_{T-1}(i) = 1$, for $i = 0, 1, \ldots, N-1$.

2. For $t = T-2, T-3, \ldots, 0$ and $i = 0, 1, \ldots, N-1$, compute

$$\beta_t(i) = \sum_{j=0}^{N-1} a_{ij} b_j(O_{t+1}) \beta_{t+1}(j). \quad (1.14)$$

For $t = 0, 1, \ldots, T-2$ and $i = 0, 1, \ldots, N-1$, define

$$\begin{aligned} \gamma_t(i) &= P(q_t = x_i \mid O, \lambda) \\ &= \frac{\alpha_t(i)\beta_t(i)}{P(O \mid \lambda)}, \end{aligned} \tag{1.15}$$

$$\begin{aligned} \delta_t(i,j) &= P(q_t = x_i, q_{t+1} = x_j \mid O, \lambda) \\ &= \frac{\alpha_t(i)a_{ij}b_j(O_{t+1})\beta_{t+1}(j)}{P(O \mid \lambda)}. \end{aligned} \tag{1.16}$$

$\alpha_t(i)$ can be obtained from Equation 1.4 (the forward algorithm) and $\beta_t(i)$ from 1.13 (the backward algorithm). Then the elements in A and B can be estimated as follows.

1. For $i = 0, 1, \ldots, N-1$, let $\pi_i = \gamma_0(i)$.

2. For $i = 0, 1, \ldots, N-1$ and $j = 0, 1, \ldots, N-1$, compute the transition probability,

$$a_{ij} = \sum_{t=0}^{T-2} \delta_t(i,j) \Big/ \sum_{t=0}^{T-2} \gamma_t(i). \tag{1.17}$$

3. For $j = 0, 1, \ldots, N-1$ and $k = 0, 1, \ldots, M-1$, compute the emission probability,

$$b_j(k) = \sum_{O_t = k, t \neq T-1} \gamma_t(j) \Big/ \sum_{t=0}^{T-2} \gamma_t(j). \tag{1.18}$$

Then the problem is fitted into the EM algorithm until the parameters converge by the following steps.

1. Randomly initialize π, A, and B.

2. Calculate $\alpha_t(i)$, $\beta_t(i)$, $\gamma_t(i)$, and $\delta_t(i,j)$ according to equations 1.4, 1.13,1.15, and 1.16, respectively.

3. Re-estimate the model $\lambda = (X, V, A, B, \pi)$ (X and V are fixed).

4. If the likelihood $P(O \mid \lambda)$ is increased, go to step 2.

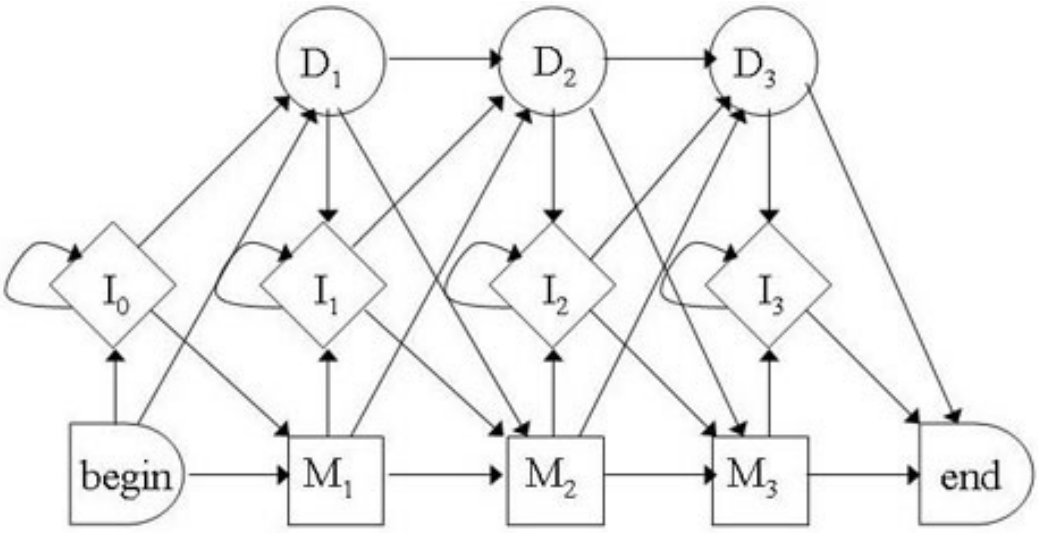


Figure 1.3: General structure of a profile HMM. D: delete state, M: match state, I: insert state.

1.3.2 Profile HMM

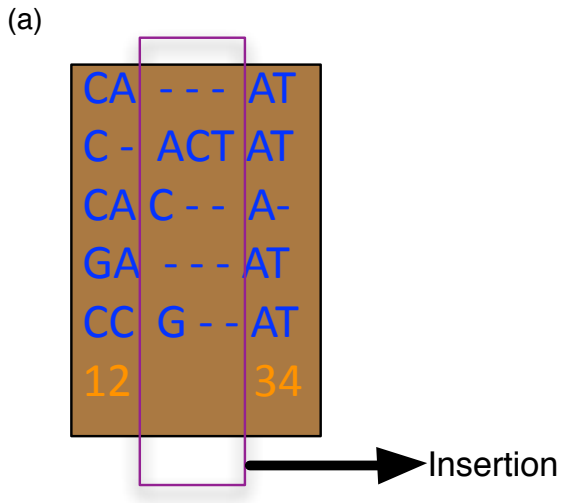
There are many applications of HMMs in bioinformatics. Profile HMMs are probably the most common application of HMMs in molecular biology [28]. Unlike a general HMM using the EM algorithm to estimate the parameters, a profile HMM can be built from a multiple sequence alignment (MSA). A profile HMM is a nondeterministic finite state machine consisting of a series of states, each of which corresponds roughly to a position (column) in the MSA from which the HMM was built. It models an MSA using a special structure as shown in Figure 1.3. Except for the "begin" and "end" states, each column in an MSA is modeled by three states in the model: insertion (I), deletion (D), and match (M). Figure 1.3 displays a model for a three-base consensus sequence. The match states are at the bottom, which model the contiguous blocks in the MSA. Insertion and deletion states are used to consider the possibility that gaps might occur when a sequence is aligned to the model. Insertions and deletions are treated separately and asymmetrically. For example, the insertion state I_i will be used to match insertions after the residue matching the ith column of the MSA. Insertion states have emission distributions that are normally set to the background

distribution. Observe that there is a transition from match state M_i to insertion state I_i, a loop transition from I_i to itself used to enable multiple insertions, and a transition from I_i to M_{i+1}. The deletion states do not emit any observations. Deletion state D_i can move to I_i, D_{i+1}, and M_{i+1}. Because the deletion states are silent, it is possible to use a sequence of them to get from any match state to any subsequent one, between two residues in the sequence.

The size of the model is determined by the decision on which columns in the MSA should be assigned to match states and which columns to insertion states. Then the transition probability matrix and the emission probability matrix can be calculated by counting the frequencies of state transitions and observations. Usually if the number of amino acids in a column is greater than $(n+1)/2$, the column is assigned to a match state. Therefore, there are four match states in Figure 1.4 (a), and the three columns in the middle are assigned to insertion states. Figure 1.4 (b) shows the observed count of emission states and transition states, based on which the emission probability matrix (A) and transition probability matrix (B) are calculated under the background distribution. Note that there are many zeros in 1.4 (b), corresponding to a probability of zero in Equation 1.3. A pseudocount is usually used to avoid zero entries based on a mixture of Dirichlet distributions [47]. More details about estimating the parameters in a profile HMM from an MSA can be found in [27].

1.4 Summary

In this chapter, we first briefly introduced the pipeline of genome projects and highlighted the stage our method targets. Various existing methods for predicting the functional effect of variants were described. We also discussed methods specifically for cancer genomes because they are different from general prediction methods. The emphasis is put on reviewing the general prediction methods since these are fundamental to our work. According to the types or the locations of genetic variants, different methods are developed. Traditional methods



(b)

	0	1	2	3	4
A	-	0	3	5	0
T	-	0	0	0	4
G	-	1	0	0	0
C	-	4	1	0	0
A	0	0	1	0	0
T	0	0	1	0	0
G	0	0	1	0	0
C	0	0	2	0	0
M-M	5	4	2	4	4
M-I	0	0	2	0	0
M-D	0	1	0	1	0
I-M	0	0	3	0	0
I-I	0	0	2	0	0
I-D	0	0	0	0	0
D-M	-	0	0	0	1
D-I	-	0	1	0	0
D-D	0	0	0	0	0

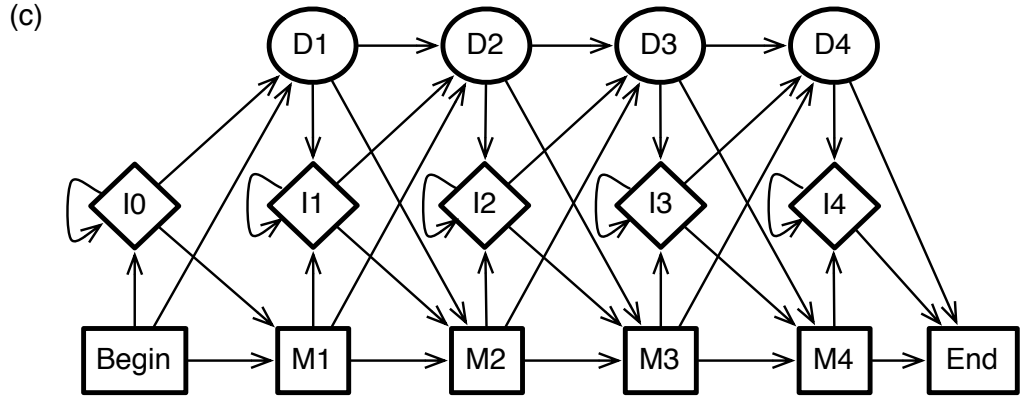


Figure 1.4: An example of building profile HMM. (a) A small DNA multiple alignment. The match states are marked with numbers corresponding to the four positions in the model shown in (c) and the insertion states are shown in the box. (b) The observed count of emission states and transition states. (c) The profile HMM structure for the MSA in (a).

such as SIFT [75] and Polyphen2 [1] predict the functional consequences of variants at the protein level by evaluating the changes in a single amino acid using evolutionary conservation or trained classier based method. These methods are limited to single nucleotide variants and ignore other types such as indels. In addition, a large number of variants in noncoding

regions, which account for more than 99% of the total number of variants identified so far, are far from fully annotation.The state-of-art methods, such as CADD [53], GWAVE [82], and FATHMM-MKL [88], predict the functional effects of variants across different regions in the genome, as well as different types of variants (SNPs and indels). Although the trained classifier method behaves like a black box, where it is difficult to biologically explain the results, people tend to use it for the prediction in noncoding regions, because of its advantage of incorporating many properties of the variants. The collection and preprocessing of the features, as well as data sampling are the important parts in this strategy. Overall, most of these methods are designed for SNPs. They only predict the effects of a single mutation. Besides, these methods predict the mutation consequences only by the fitness effect, which are categorized as neutral, deleterious, or beneficial.

The rest of the dissertation is organized as follows. In Chapter 2, we introduce the novel functional prediction method, HMMvar, and explore its properties. We evaluate the performance of HMMvar and compare it with other existing tools. In Chapter 3, we extend the functionality of HMMvar to predict the joint effects of multiple variants in the same gene. In Chapter 4, we explore the functional outcome of variants based on multiple HMMs generated from clusters of protein sequences. We test the sensitivity of different clustering methods and evaluate their effects to the final prediction results. The results are presented by evaluating various data sets. Chapter 5 concludes the current work and presents a framework for predicting the functional effects of variants in noncoding regions.

Appendix A shows the proof of the Cumulative Conjecture proposed in Chapter 3. Appendix B presents the blueprint of the final comprehensive annotation/prediction system for genetic variants.

Chapter 2

Quantitative Prediction of the Effect of Genetic Variants

2.1 Introduction

Small indels account for the second largest portion of human variants, however, available methods for indel functional predictions, no matter in coding or noncoding regions, are many fewer compared to those for SNPs. It has been shown that indels cause more severe functional changes in proteins than SNPs [85] and also have significant influence on protein-protein interaction interfaces [44]. As revealed by the Human Gene Mutation Database (HGMD) [95], approximately half (57%) of the human disease variations (gene sequence level) are associated with single nucleotide substitutions, and about a quarter (22%) are associated with small indels [94,95]. Mill et al. [65] have shown that 42% of the nearly two millions indels they identified are mapped to human genes and more than 2,000 indels affect coding exons and likely disrupt protein function and cause phenotypic changes in humans. Moreover, many of the identified indels had a high level of linkage disequilibrium (LD) with SNPs, indicating that the indels might be essential factors that cause diseases. Furthermore, indels have profound functional impact in human specific evolution and adaptation [12,13,105]. Despite

their significance, only a handful of quantitative prediction methods, including FunSeq [52], CADD [53], SIFT-indel [45], and Provean [14] (Table 1.1), were developed for predicting the functional effects of indels.

Sequence alignment is usually used to visualize the association between residues in a set of evolutionarily or structurally related proteins. An MSA provides an overview for the underlying evolutionary, structural, or functional constraints of a protein family, which is often used to predict the functional impacts of mutations. According to the theory of natural selection, different regions of a functional sequence are subject to different selective pressures. MSA reveals this by amino acid/nucleiotide conservation in certain positions. Some positions are more conserved than others, and some regions are more tolerant to indels than others. Mutants occurring at highly conserved residuals are more likely to be deleterious, whereas mutants occurring at lowly conserved residuals are more likely to be neutral or less deleterious. This is exactly the feature of profile HMMs. Basically, a profile HMM is a probabilistic description of the consensus of an MSA. Thus it is reasonable to consider profile HMMs as a tool for predicting functional effects of variants. Scoring (computing the probability of generation by a given Markov process) a wild type sequence or mutated sequence with the profile HMM gives one an idea how far the given sequence is away from the original population. A profile HMM captures the characteristics of an MSA, from which quantitative conservation information (a probability) is obtained. A high score means good fitness of the sequence with the protein family represented by a profile HMM. Thus, a high score of the probability of generation from the profile HMM for the wild type sequence, and a low score for the mutant sequence probably mean that the mutation has deleterious effects.

2.2 HMMvar: Functional Effects of Variants in Coding Regions

We developed HMM based variant prediction (HMMvar) [61], a tool for predicting the functional effects of SNPs or indels in coding regions of sequences. The five-step prediction pipeline of HMMvar (Figure 2.1) receives a set of small indels or SNPs in related genes as input, then completes the following steps: (1) find seed proteins that are affected by the indels or SNPs; (2) find homologous sequences from a database for each seed protein; (3) generate an MSA for each set of homologous sequences; (4) build a profile HMM based on each MSA; (5) predict the functional effects of indels using the profile HMMs. The first step identifies all unique proteins affected by these indels or SNPs as wild type sequences (seeds). Because it is possible that multiple indels affect one protein and multiple proteins are affected by one indel due to alternative splicing, it is more computationally efficient to first identify all the proteins involved in parallel. The mutant sequences for a given wild type sequence are obtained by inserting the indels into the wild type sequence. The input can be a nucleotide or protein sequence along with the variants in the sequence. In this case, step 1 is ignored as the seed protein is provided by the user as input. The second step, using the identified proteins as seeds, invokes PSI-BLAST [2] on a protein sequence database to find a set of homologous sequences for each seed protein. Both the NCBI non redundant (nr) [77] and uniprot90 [17] databases are used for comparison. The e-value and iteration limits were 0.01 and 5, respectively. The homologous sequences include both paralogous and orthologous sequences. The third step invokes ClustalW2 [55] with the BLOSUM62 matrix and the word size three for MSA for each homologous sequence set. The next step builds profile HMMs with HMMER3 [32] using the MSAs as training data (one HMM per seed protein). All mutant type sequences derived from the same seed sequence will use the same HMM for functional effect prediction. The last step uses all the constructed HMMs for predicting the functional effects. Precisely, given an input indel (mutant type) corresponding to seed protein (wild type), the ith profile HMM is used to compute the HMMvar score S,

as defined below.

The bit score from HMMER3 measures the fitness of a query sequence with the set of homologous sequences used to define the profile HMM. The HMMER3 bit score is a base 2 logarithm of ratio of probabilities (homology hypothesis over the null hypothesis),

$$B = \log \frac{P(O_0O_2, \ldots, O_{n-1} \mid HMM)}{P(O_0O_2, \ldots, O_{n-1} \mid NULL)}, \tag{2.1}$$

where n is the length of the observed protein sequence; $O_0O_2, \ldots, O_{n-1}$ is the observed protein sequence and "HMM" is the trained profile HMM. "NULL" is the "null model", which is a one-state HMM configured to generate "random" sequences of the same length as the target sequence, with each residue drawn from a background frequency distribution. For proteins, the frequencies of the 20 amino acids are set to the amino acid composition of SWISS-PROT 34 [5]. Another parameter in the null model P_1 controls the expected length of a randomly generated sequence. P_1 is set to 350/351 in [6], which implies that the expected mean length of a protein is about 350 residues. The constituent probabilities are derived from the bit score and used to define the HMMvar score as the odds ratio

$$S = \frac{P_w/(1-P_w)}{P_m/(1-P_m)}, \tag{2.2}$$

where P_w (P_m) is the probability that the wild type (mutated type) protein sequence could have been generated by the profile HMM trained on a homologous sequence set to the seed protein. Usually, this probability is calculated by the forward algorithm as we discussed in Chapter 1 in Equation 1.3. Here, this probability is derived from the bit score obtained from the HMMER3 package. Given a protein sequence, the probability that it was generated under the null model is,

$$P_{null} = exp(l * \log P_1 + \log(1 - P_1)) \tag{2.3}$$

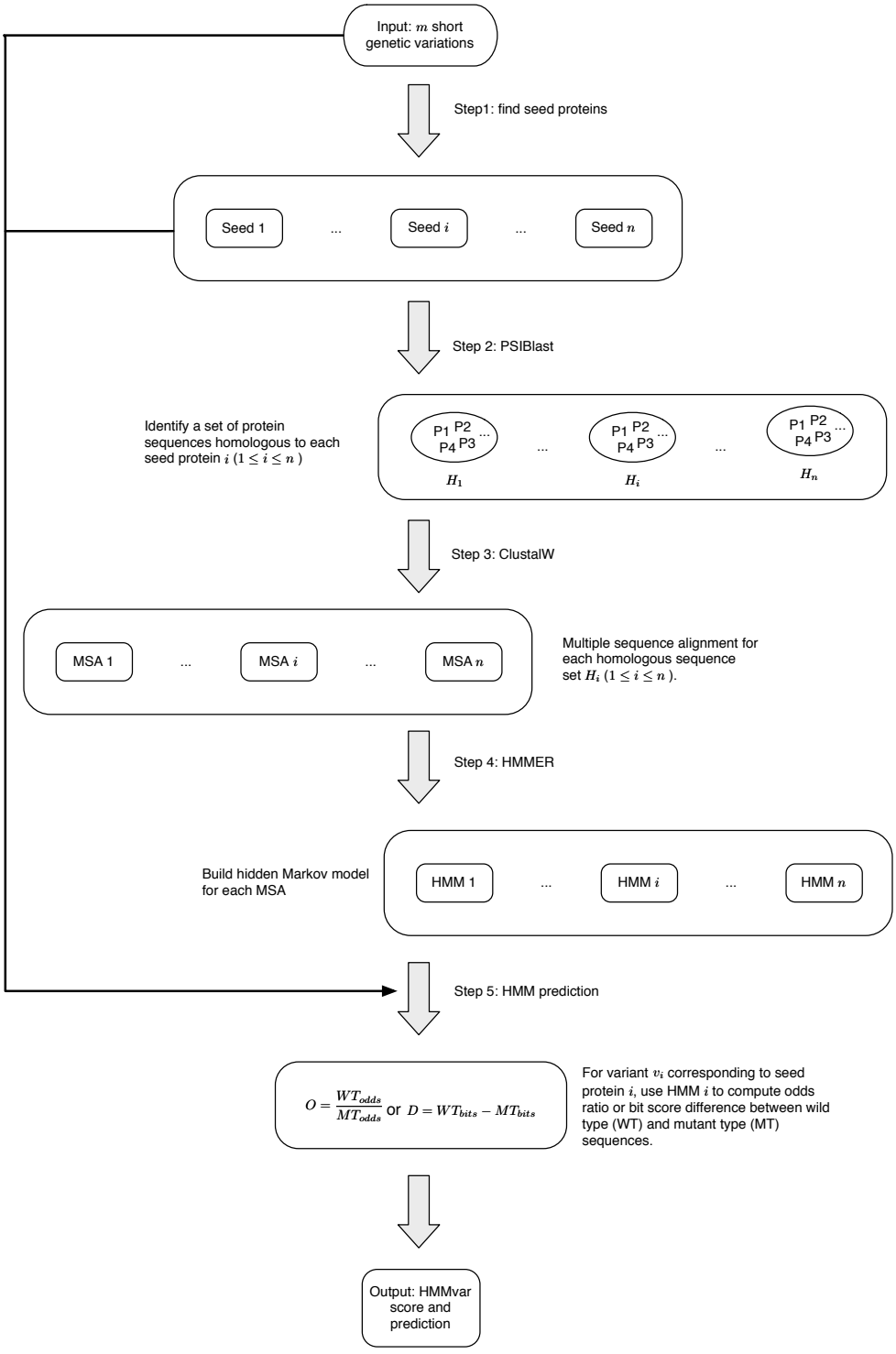


Figure 2.1: Pipeline of HMMvar prediction for variants in coding regions.

where l is the length of the sequence and P_1 the parameter in the null model ($P_1 = 350/351$). From the null model and bit score equation, the probability P_w or P_m can be derived as $P = P_{null} * e^B$ given a wild type sequence or mutated type sequence.

Each wild type sequence (or seed protein) corresponds to one HMM model. We consider only the HMMs whose wild type sequence bit scores are greater than zero and compute the odds ratio for mutant type sequences that derive from these wild type sequences. The odds ratio is expected to be greater than 1, indicating the wild type sequence is more likely to occur in the HMM presented family. However, in practice, this is not always the case, which indicates that the mutant type sequence better fits the homology set profile. This situation may result from the nucleotide level mutation causing the amino acid level changes to be more compatible [5] with the homologous sequences than the wild type protein.

If the HMMvar score S is less than a threshold u, the indel is considered neutral, otherwise deleterious. Fisher's exact test was used to choose the threshold, using SIFT-indel prediction as the reference method, by minimizing the exact test p-value, giving the threshold $u = 2.0$ for the data sets used.

Instead of the odds ratio S, one could use the HMMER3 bit scores directly in the difference

$$D = B_w - B_m, \tag{2.4}$$

where B_w is the bit score of the wild type sequence and B_m the bit score of the mutant type sequence. The bit score difference D represents the base 2 logarithm of the relative risk (probability of generating the wild type sequence over the probability of generating the mutant type sequence). For SNP variants, the S score results in extremely low variance, so the D score is used for the TP53 SNPs in the result section.

The selection of homologous sequences is the key for building a high quality profile HMM. The NCBI non-redundant protein sequence (nr) database was created by integrating several other databases, so it is highly redundant. Comparatively, the uniprot90 database is less redundant as it is generated from a consensus of a group of sequences with identity percentage over 90%.

For this reason, the uniprot90 database is also tested for homologous sequence search. Results for the two databases show no apparent difference. Therefore, the experiments only based on the nr database are presented. PSI-BLAST [2] is used to collect homologous sequences for each seed protein, using e-value 0.01, and iteration limit five. All sequences above 10% identity were selected as homologous sequences for a certain seed protein. Attempts to improve diversity in the homologous sequence set by including the sequences below 10% identity or using instead all sequences from 60% identity to 95% identity did not produce better HMMvar score distributions.

2.3 Results

The performance of HMMvar is validated via various data sets, including indels, SNPs, and mutations in an individual protein. HMMvar is also compared with other traditional tools. The results demonstrate that a scoring strategy based on HMM profiles can achieve good performance in identifying deleterious or neutral variants for different data sets.

2.3.1 Predictions on Indels

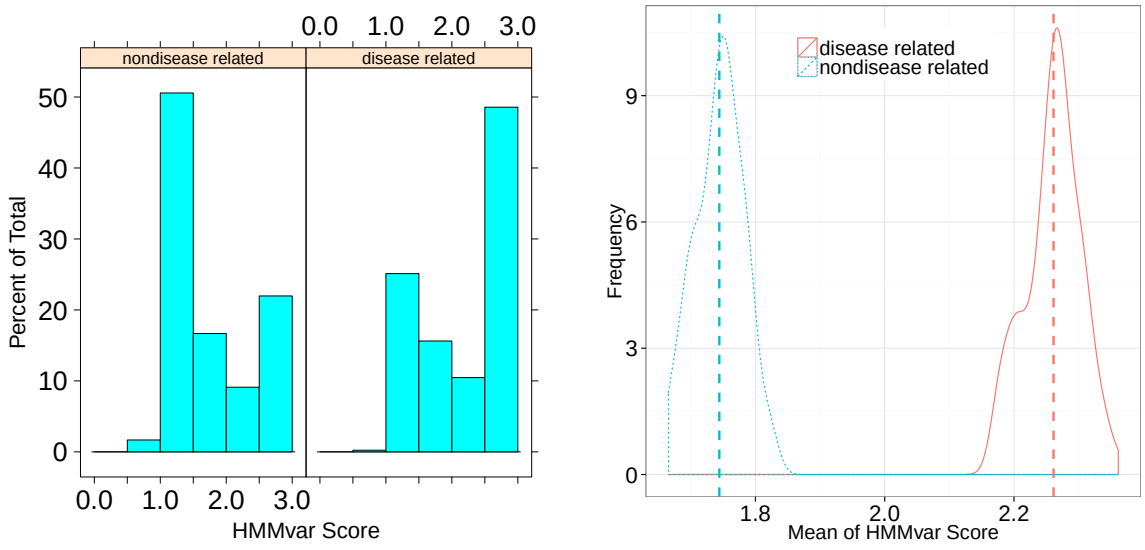
Indels were obtained from the database dbSNP [86], including human coding nonsynonymous mutations, such as nonsense, missense, and frameshift indels. Nonsense means the mutation introduces a stop codon, for example, the codon TCA changes to TGA. Missense means the indels that add or remove amino acids to or from the original protein sequence, for example, the codon ACT changes to GCT, which alters threonine (Thr) to alanine (Ala). The length of a missense indel is always divisible by three, which means the sequence is still in frame with the variants. A missense SNP is a SNP that leads to the replacement of the original amino acid with a new one. Frameshift means the mutation changes the open reading frame of protein translation. The data is then classified into two groups: variants that have Locus-specific Mutation Database (LSDB) [92] annotation, which are expected to

be disease associated and have more harmful effects, and variants that do not have LSDB annotation, which are expected to be nondisease (or unknown) associated and have less harmful effects. Since the numbers of LSDB indels and nonLSDB indels in the database are highly unbalanced, we randomly sampled the same number of proteins that have indel mutations in both categories. Table 2.1 lists the indel categories of the data set. The fractions (4% and 95.7%) of nonsense and frameshift mutations in the LSDB group are higher than those (1% and 95.1%) in the nonLSDB group, while there are no missense indels in the LSDB group but 56 in the nonLSDB group, suggesting that nonsense and frameshift indels are more likely to cause diseases.

Table 2.1: Data Set from dbSNP

	LSDB	nonLSDB	Total
Nonsense	112	15	127
Missense	0	56	56
Frameshift	2519	1387	3906
Total	2631	1458	4089

The effects of indels in these two groups (LSDB and nonLSDB) were quantified by HMMvar. Figure 2.2(a) shows the distributions of the HMMvar scores (the odds ratio, S, described in the Methods section) in the disease associated and nondisease associated groups. When the score is small (typically $S < 2$), nondisease associated variants dominate, while disease associated variants significantly dominate the right side of the distributions ($S \geq 2$). The Kolmogorov-Smirnov test ($p < 2.2e-16$) indicates a significant difference between the HMMvar score distributions of the two groups. The mean scores in the two groups were compared by a one-sided two-sample t-test, where 200 variants from each group were randomly sampled with replacement and the means of the sampled data from the two groups were compared. This process was repeated 100 times, yielding two distributions of the sample means as shown in Figure 2.2(b). The two vertical dashed lines represent the means of these two distributions, which are significantly different (t test, $p < 2.2e-16$).



(a) (b)

Figure 2.2: HMMvar score distribution of the dbSNP data set. (a) Histogram of HMMvar scores for disease associated indels and nondisease associated indels. (b) Distribution of sample means of HMMvar scores from the two categories (LSDB and nonLSDB).

Different functional types of indels (nonsense, missense, and frameshift) were combined to give an overview of the distributions of the HMMvar scores for different groups (Figure 2.3). The most remarkable feature is that the score of missense indels is much lower than the scores of the other two types, consistent with the notion that missense mutations tend to have less deleterious effects than frameshift indels and nonsense mutations. In each type of indel, the median of the nondisease associated group is lower than the median of the disease associated group, demonstrating that the HMMvar score is effective in measuring the harm of indel mutations.

To test the consistency of HMMvar scores with a genome wide analysis, the indels with minor allele frequency (MAF) in dbSNP were extracted, resulting in 447 indels to be scored. The less the allele frequency is in a certain position of a genome, the more conserved the site and the more deleterious the effects of a mutation at this site, in terms of evolutionary theory. In this experiment, the MAF shows a negative Pearson correlation with the HMMvar score

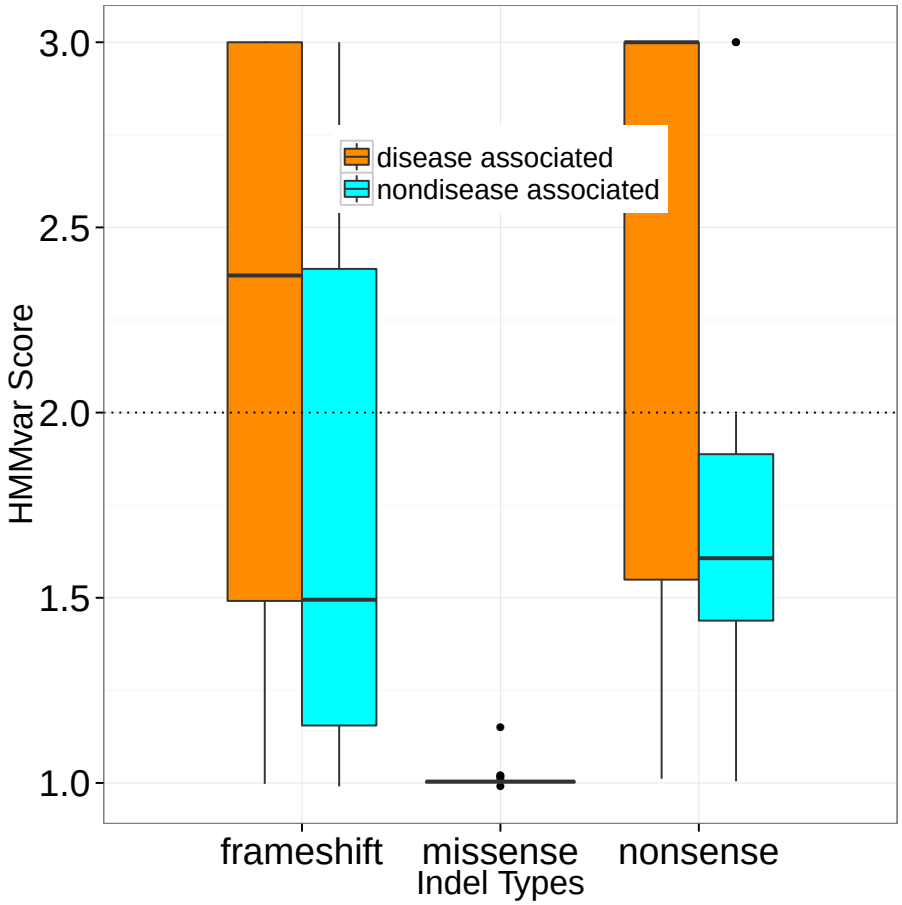


Figure 2.3: Distributions of HMMvar scores for different types of variants.

($r = -0.03$), which is consistent with the common indication of MAF (the lower the MAF, the higher the significance of the site), though the correlation is not significant.

The experiments show several factors that affect the prediction score, such as the location of indels in the protein (Figure 2.4), and different types of indels (nonsense, missense, or frameshift, Figure 2.3). It is expected that frameshift indels close to the 5' end of the sequence are more likely to have deleterious effects than indels occurring close to the 3' end of the sequence as the former may affect a larger number of amino acids. Figure 2.4 displays the relationship between the HMMvar score and the position of an artificially introduced stop codon to a random protein. Nonsense variants introduce a stop codon at the mutation

resulting in the termination of mRNA translation, which brings greatly deleterious effects if occurring close to the 5' end of the sequence. A missense mutation may change some amino acids locally, thus may have a relatively smaller effect compared to frameshift or nonsense variants.

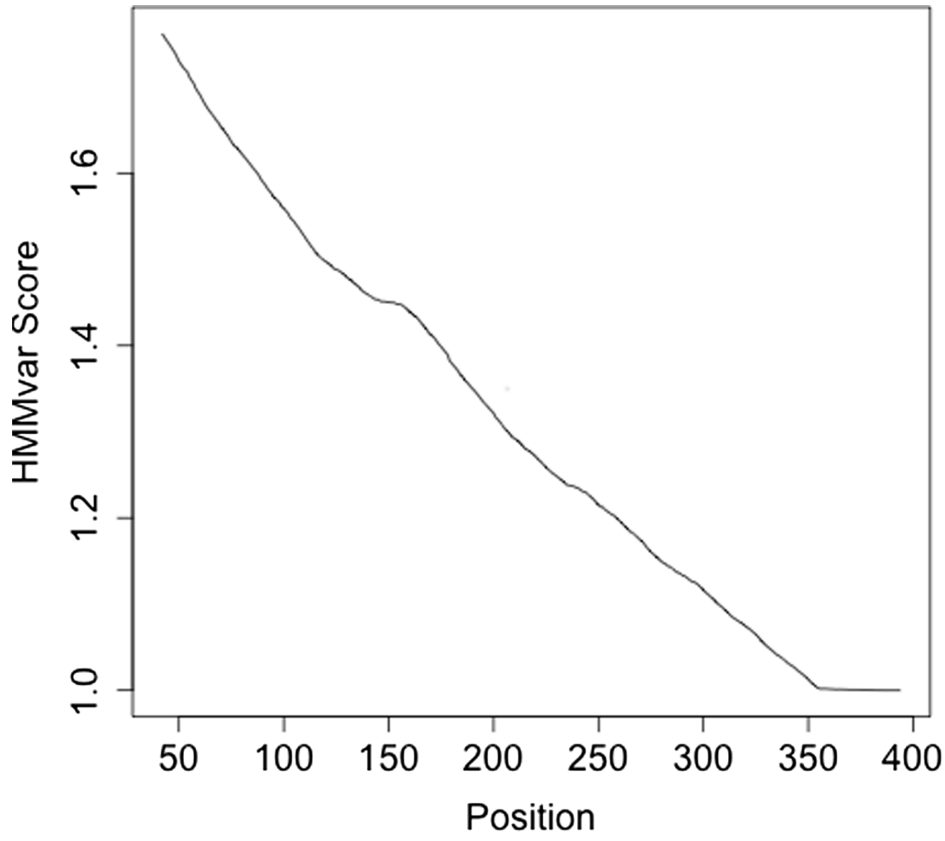


Figure 2.4: The relationship between the HMMvar score and the position of an artificially introduced variant.

It is expected that the quality of the MSA is another factor that can potentially affect the prediction of indel effects. Comparing the HMMvar scores based on different MSA programs, ClustalW [55] and MUSCLE [29], for the TP53 transitivity level data set, showed that HMMvar scores based on the MUSCLE sequence alignment decreases more smoothly and shows lower variance within the same functional classes than scores based on the ClustalW sequence alignment. This suggests that having high quality sequence alignment is important

for accurate indel effect prediction.

2.3.2 Comparison with Other Tools

HMMvar is compared with SIFT-indel [45], a tool recently proposed for predicting indel effects, as well as two traditional effect prediction tools for SNPs only, SIFT SNP [75] and PolyPhen [79]. SIFT-indel uses a trained classifier (decision tree) method to predict the effects of indels using four extracted features, which are considered to be the most informative properties in the paper [45]. Figure 2.5 shows the distributions of HMMvar scores of two groups, damaging and neutral, predicted by SIFT-indel on all the frameshift indels shown in Table 2.1. They have significantly different distributions (Kolmogorov-Smirnov test, $p = 2.273e-09$), indicating that the HMMvar score is able to predict the two different functional effects using SIFT-indel prediction as a reference. When the score is small (typically $S < 2$), the frequency of neutral indels is higher than the frequency of damaging indels. On the other hand, when the score is large ($S \geq 2$), the frequency of damaging indels dominates. Three Fisher's exact tests were done: (1) HMMvar prediction versus SIFT-indel prediction, (2) HMMvar prediction versus database annotation, and (3) SIFT-indel prediction versus database annotation. The p-values are $7.778e-05$, $3.456e-12$, and 0.4863, respectively, showing that HMMvar prediction has higher correlation with database annotation. The sensitivity, specificity, and accuracy comparisons between HMMvar and SIFT-indel are shown in Table 2.2. SIFT-indel prediction has higher sensitivity but much lower specificity than HMMvar prediction.

Table 2.2: Comparison between HMMvar prediction and SIFT-indel prediction with dbSNP indel data set

	Sensitivity	Specificity	Accuracy
HMMvar	77.8%	68.6%	77.7%
SIFT-indel	95.7%	5.9%	94.0%

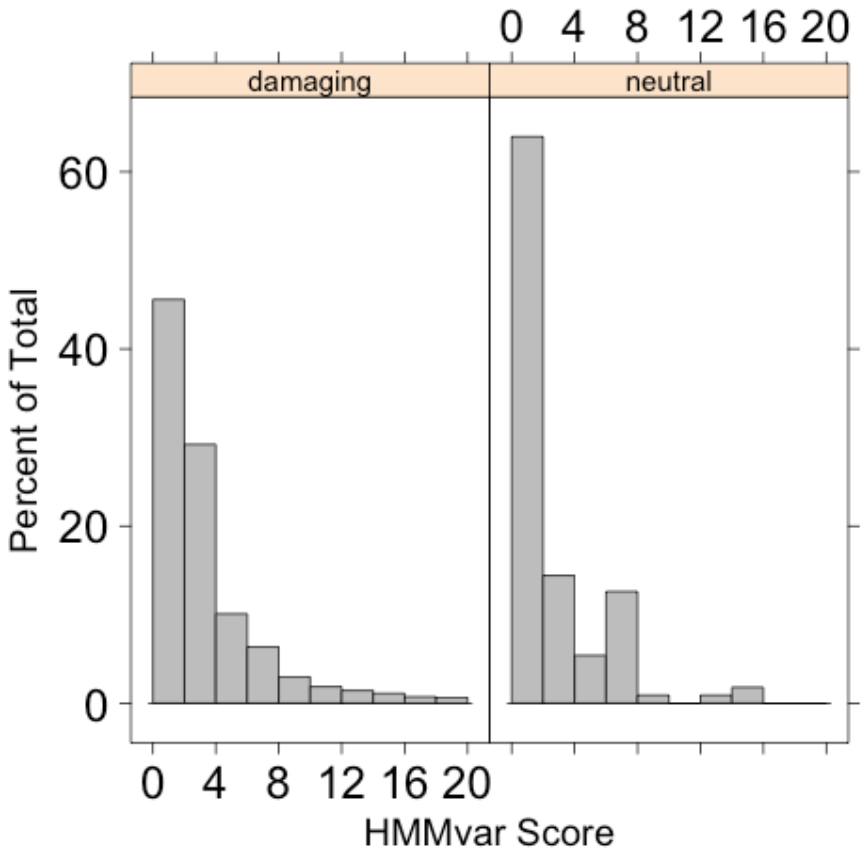


Figure 2.5: Compare HMMvar prediction with SIFT-indel prediction on dbSNP indel data set. Distributions of HMMvar of indels that are predicted as damaging (left) and neutral (right) by SIFT-indel.

Both SIFT SNP and PolyPhen are prediction tools for nonsynonymous SNPs only. To compare with these two programs, SNPs were downloaded from the database ENSEMBL (version: Variation 69, GRCh37.p8) [33], along with precomputed scores and predictions. Among the more than one million SNPs downloaded, only about 80,000 SNPs have Polyphen and/or SIFT predictions. There are two SIFT SNP prediction categories, deleterious and tolerated, and three PolyPhen prediction categories, benign, possibly damaging, and probably damaging. Since prediction for SNPs is very time consuming due to PSI-BLAST database

Table 2.3: Data Set from ENSEMBL

		SIFT		
		Deleterious	Tolerated	Total
Polyphen	Probably damaging	91	87	178
	Benigh+Possibly damaging	107	108	215
	Total	198	195	393

searching, 393 SNPs were randomly selected as shown in Table 2.3. To balance the data, PolyPhens possibly damaging and benign categories are combined together. Fisher's exact test for the HMMvar prediction (cutoff 1.002) versus the SIFT SNP prediction has p-value $5.626e-05$, HMMvar prediction versus PolyPhen prediction has p-value 0.2285, and SIFT SNP prediction versus PolyPhen prediction has p-value 0.8788. The HMMvar prediction has a high correlation with the SIFT SNP prediction, but the HMMvar and SIFT SNP predictions both have a weak correlation with the PolyPhen prediction, based on this data set.

2.3.3 A Case Study of Individual Protein: TP53

This section addresses whether the HMMvar score can reflect the degree of mutation effects on one extensively studied disease related protein, TP53. TP53 (known as tumor protein 53) acts as a tumor suppressor, and regulates cell division by preventing cells from growing and dividing too fast or in an uncontrolled way. A set of 2,565 SNP mutants and corresponding biological activity levels were obtained from the database IARC TP53 [76]. The mutants associated with TP53 were partitioned into four classes: nonfunctional, partially functional, functional (wildtype), and supertrans (higher activity than wildtype). The classification was made in terms of the median transactivation level of eight different promoters as measured in [49]. For each mutant, the median of the eight promoter-specific activities (expressed as a percent of the wild type protein) is calculated and mutations are classified as "nonfunctional"

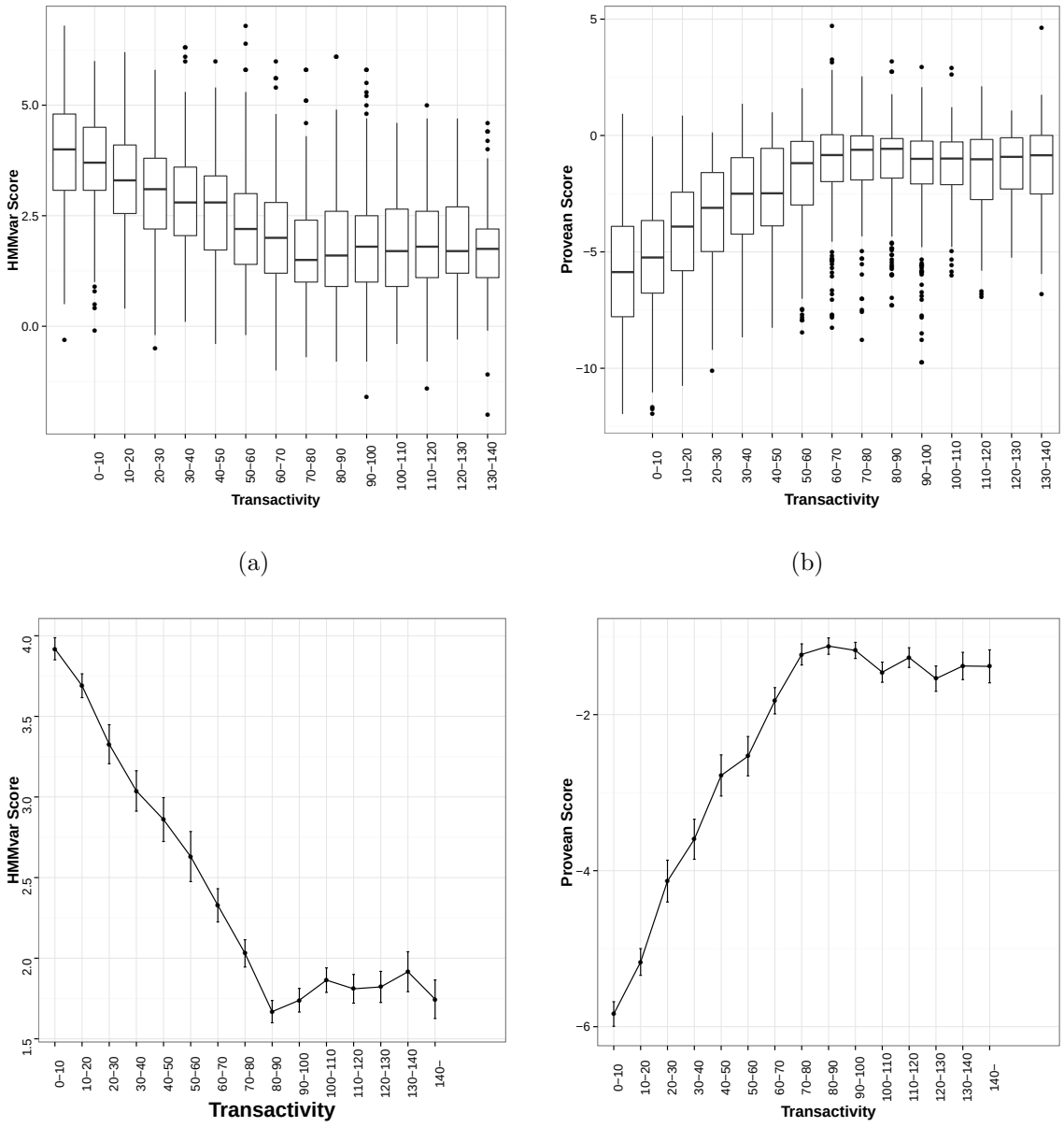
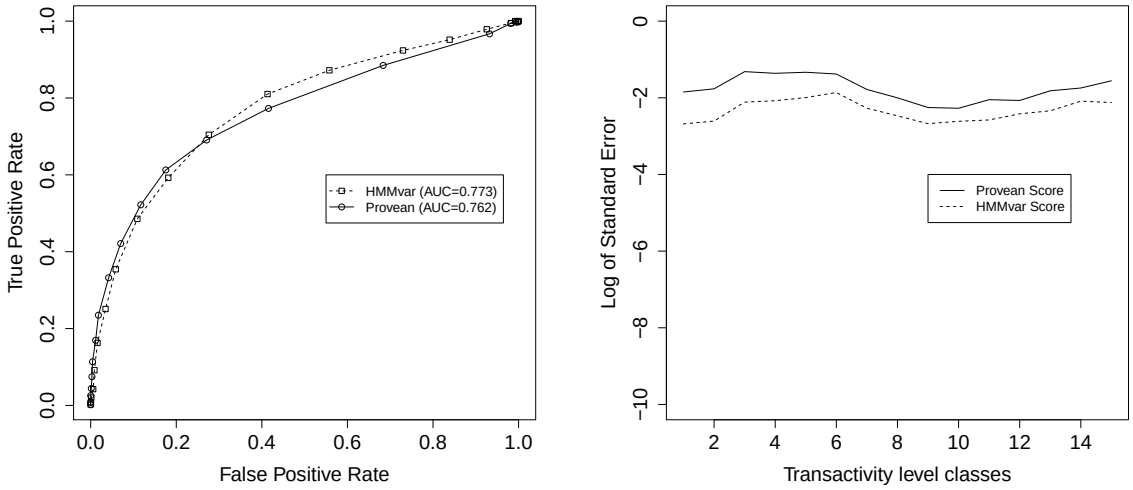


Figure 2.6: HMMvar and Provean score distributions and mean/error bars of TP53 mutations binned into 15 classes in terms of transactivity level. (a) HMMvar score distribution of the 15 classes (x-axis represents the 15 classes based on the median of transactivity levels). (b) Provean score distribution of the 15 classes. (c) Mean along with error bar of HMMvar scores in each class. (d) Mean along with error bar of Provean scores in each class.

if the median is ≤ 20, "partially functional" if the median is > 20 and ≤ 75, "functional" if the median is > 75 and ≤ 140, and "supertrans" if the median is > 140. The SNPs can also be more finely separated into 15 classes in terms of the median values with an increment of 10.



(a) (b)

Figure 2.7: ROC curve and standard error of the HMMvar score and the Provean score. (a) ROC curve of the Provean score and the HMMvar score to distinguish "nonfunctional" and "partly functional" classes from "functional" and "supertrans" classes. (b) Standard error of the mean of Provean and HMMvar scores in the 15 transactivity level classes.

The results are compared with Provean [14] (introduced in Chapter 1), an evolutionary conservation based indel and SNP effect prediction method. Figure 2.6(a) and 2.6(b) show the HMMvar scores and Provean scores versus the transactivity level, respectively. With respect to the transactivity level, the HMMvar score shows a negative relationship, and the Provean score has a positive relationship, especially in the nonfunctional and partially functional regions. Figure 2.6(c) and 2.6(d) show the average scores and error bars for each functional class for the similarity trending HMMvar and Provean scores, respectively. As expected, the HMMvar score shows a strong linear relationship with the Provean score (Pearson cor-

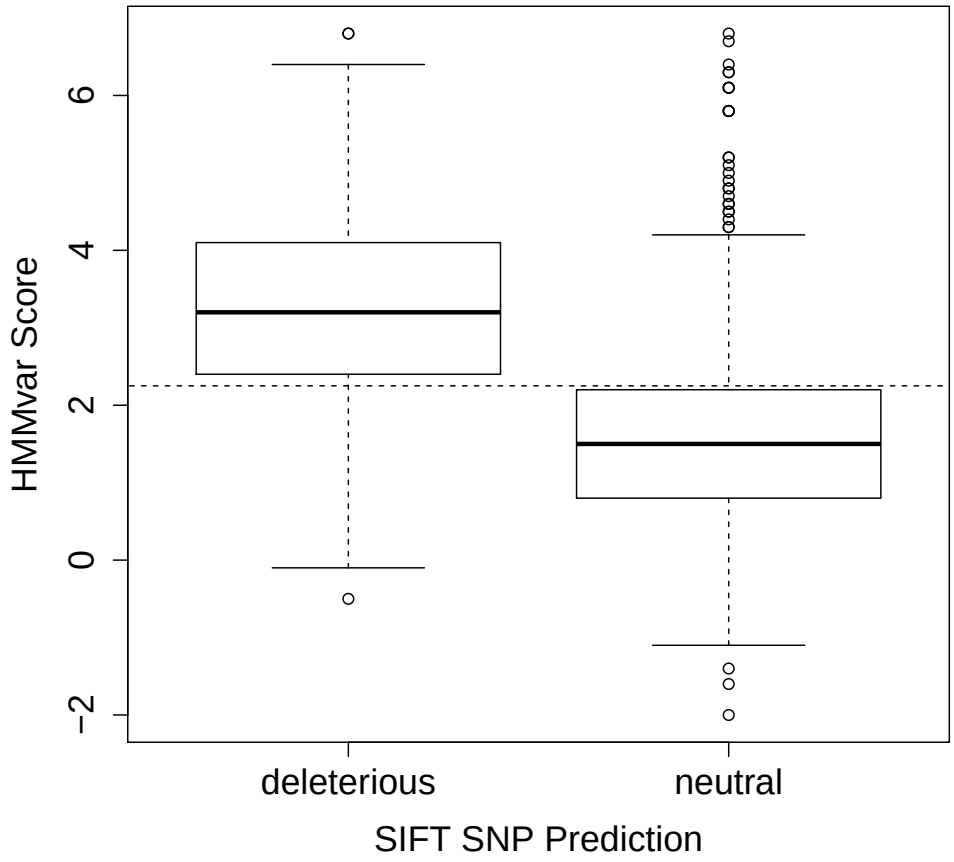


Figure 2.8: The HMMvar score of TP53 variants grouped by SIFT SNP prediction.

relation coefficient $r = -0.733$). The HMMvar score has a slightly lower correlation with the transactivity level ($r = -0.523$) than the Provean score ($r = -0.552$) but a slightly higher correlation than the SIFT SNP score ($r = -0.493$). Figure 2.7(a) shows the receiver operating characteristic (ROC) curve for the comparison between HMMvar and Provean in distinguishing "nonfunctional" and "partly functional" classes from "functional" and "super-trans" classes. HMMvar obtained higher AUC than Provean. To better distinguish between different functional classes, it is highly desirable that a prediction metric exhibits small variance for mutations within the same functional class. Hence consider the variance of HMMvar

and Provean scores within each functional class. The standard error of the mean for each functional class is $SE = \frac{S}{\sqrt{n}}$, where S is the standard deviation of the scores for a functional class and n is the size of the class. The HMMvar score has less variance for each functional class as shown in Figure 2.7(b), indicating that the HMMvar prediction is more stable than the Provean prediction. There are also SIFT SNP predictions for TP53 variants available in the data set; comparing the HMMvar score with the SIFT SNP prediction shows that the medians of the HMMvar scores in the two SIFT SNP predicted groups are significantly different (Figure 2.8).

Most existing methods for variant effect prediction are based on evolutionary conservation theory, which predicts that highly conserved sites experience strong purifying selection and mutations in these sites are most likely to be deleterious to protein function. However, these methods address each site independently of other sites and do not consider the impact of surrounding sites. Moreover, most of these methods are designed only for SNP variants. In contrast, a profile HMM serves as a representation of a set of homologous sequences, relating all sites through a Markov process. FATHMM [87] (introduced in Chapter 1) also uses a profile HMM to predict the functional effects of SNPs in coding regions in a similar way as HMMvar. However, unlike HMMvar, FATHMM only predicts the functional impacts for SNPs. Consequently, the present method HMMvar can provide functional predictions for the effects of indels besides SNPs and can predict the effects of multiple variants simultaneously. The latter is especially useful when multiple variants occur in a protein, each one of them may have deleterious effects on protein function, but the combination of them may have less harmful effects due to the possibility of a compensatory effect. Profile HMMs, used as proposed, have the capability to predict the total effects of multiple mutations along the gene given a specific haplotype, which is discussed in detail in the next chapter.

Chapter 3

Functional Effects of Multiple Variants

3.1 Introduction

Many genetic variants have been identified in the human genome. The functional effects of a single variant have been intensively studied. However, the joint effects of multiple variants have been largely ignored due to complexity or lack of data. Complex diseases are likely to be caused by multiple genes and/or multiple mutations on the same genes [63], so quantitatively measuring the effects of multiple variants together should be helpful for detecting causal genes/mutations for diseases. For example, it has been shown that the correlation between breast cancer and multiple SNPs of the ORAI1 gene is more significant than that with single SNPs [11]. The authors use a genetic algorithm to find combinations of SNPs that are significantly different between the genotypes of the case group and the control group. The results reveal that new insights in cancer studies are possible by considering the cumulative effects of multiple variants or the associations among genetic variants. Genetic causes of autism have been studied for years. Recently, researchers found that multiple rare mutations within a single gene may increase the risk of autism [97]. Also about 20 rare

variants are very likely related to autism, of which four are in the coding regions and 15 in the noncoding regions of the 5-HT transporter (SERT) gene. Cystic fibrosis is a disease that is associated with the CFTR gene and is triggered by multiple mutations [36] in the gene. In the methylenetetrahydrofolate reductase (MTHFR) gene, the variants C677T (alanine to valine) in the catalytic domain and A1298C (glutamate to alanine) in the regulatory domain together are known to decrease the activity of the MTHFR gene [103].

This chapter focuses on predicting the joint effects of variants in coding regions in the same gene using HMMvar [61] discussed in Chapter 2. Because the HMM is built from the MSA of homologous proteins from different species, it reflects the extent of evolutionary conservation naturally by its probabilistic profile. The probabilistic profile can be used to compute and compare the likelihood of generating mutant bearing sequences given the HMM with the likelihood of generating mutant free sequences, i.e., wild type sequences, given the HMM. The lower the former compared with the latter, the more deleterious the mutants are likely to be. Similarly, by evaluating the fitness of the wild type sequence versus the mutant sequence that contains multiple variants, we can measure the effects of multiple variants as a whole. Therefore, HMMvar is able to predict the functional effects of a single mutation, as well as the joint effects of multiple mutations in coding regions.

To demonstrate the effectiveness of HMMvar, data from the 1000 Genomes Project is used to identify genes that have multiple mutations, and HMMvar is used to predict the effects of multiple mutations on the genes identified. In addition, indels from two tumor suppressor genes, TP53 and PTEN, are also used to investigate the effects of multiple indels from a single gene. If a frameshift indel occurs, it is possible that a nearby second indel rescues the gene by restoring the reading frame. There is very limited knowledge about this kind of compensatory indel, but these are important because the deleterious effects of frameshift indels could be minimized by nearby compensatory indels. In [45], the author claims that frameshift indels near each other are more likely to restore the translation frame. The present work found compensatory indel sets for TP53 and PTEN and measured the functional effects of individual indels and compensatory indel sets using HMMvar. Compound mutations causing

severe cardiovascular disease in two genes, compared to a single mutation, are also validated. This chapter extends the functionality of HMMvar, a tool for assigning a quantitative score to a variant, to not only measure the effects of a single variant, but also the joint effects of multiple variants. HMMvar-multi is the first tool that can predict the functional effects of both single and general multiple variations on proteins.

3.2 HMMvar-multi: Joint Effect of Multiple Variants

3.2.1 Determine the Haplotypes

All the variants from the Phase I data of 1000 Genomes Project along with their genotypes and ancestry alleles are collected to find the definite haplotype in the genes for each individual. To quantitatively measure the effects of multiple variants in the same gene, the variant sets are generated in terms of their genotypes and the corresponding ancestral alleles that are inferred from the phylogenetic tree built from the multiple sequence alignment of four primate species [31, 73, 74]. Given a certain gene and an individual sample, the variants are grouped into four classes based on their locations and genotypes. Figure 3.1 illustrates an example of generating SNP sets from eight variants in a gene.

- Class 1: variants that are in the coding regions and the genotypes are homozygous and different from the ancestry allele, as the red variants shown in Figure 3.1.
- Class 2: variants that are in the coding regions and the genotypes are homozygous and the same as the ancestry allele, as the green variants shown in Figure 3.1.
- Class 3: variants that are in the coding regions and the genotypes are heterozygous, as the blue variants shown in Figure 3.1.
- Class 4: variants that are not in the coding regions, such as $3'$-utr, $5'$-utr, or intron regions, as the orange variants shown in Figure 3.1.

Only the variants in Class 1 are kept as a set to be scored, because all the variants in Class 1 are homozygous and are mutants compared to the ancestral alleles. They can form a determined haplotype for a sample individual. A single gene can have multiple transcripts due to alternative splicing, so a variant may be present in multiple transcripts. Figure 3.1 shows a gene with three transcripts. Consequently, a variant set is generated for each transcript: Transcript 1 contains variant set $\{v_1, v_3\}$; Transcript 2 contains variant set $\{v_3, v_6, v_8\}$; Transcript 3 contains variant set $\{v_6, v_8\}$. Finally, these sets will be scored against the corresponding transcripts by HMMvar. The homozygotes detected in individual samples along with the set score are available in the database for further analysis [60].

For each gene, all homologous genotype variants that are different from the ancestry allele are identified based on an individual sample. A transcript related to a certain gene might be associated with multiple variant sets due to the difference of genotypes among samples, and a variant set can also be associated with multiple transcripts due to alternative splicing. Table 3.1 shows an example illustrating the relationship between individual, gene, and variant sets. Only the records related to two individuals are shown here as an example (there are actually 2566 records related to gene ABCB5). As shown, gene ABCB5 is associated with multiple variant sets and even the same transcript (NM_178559.5) is associated with multiple variant sets due to the difference of genotypes of different individuals. The same variant set corresponds to multiple transcripts and multiple individuals. Finally, processing all genes that contain at least one variant set with size greater than one yielded 67,109 variant sets from 8,021 genes (14,917 transcripts) involving 1092 individual samples.

The variants in the 1000 Genomes Project are identified by next generation sequencing technology where short reads are generated and compared to the human reference genome. Therefore, the genotype of the individual is unknown, as is whether multiple mutations exist on the same allele or different alleles. To circumvent this problem, only those variants that are in a homozygous state are scored. Figure 3.2 shows the zygosity of disease-causing mutations or any mutations in general. Single variants could be in a heterozygous (a) or homozygous (b) state. For multiple variants on the same gene (Figure 3.2 (c–e) shows two

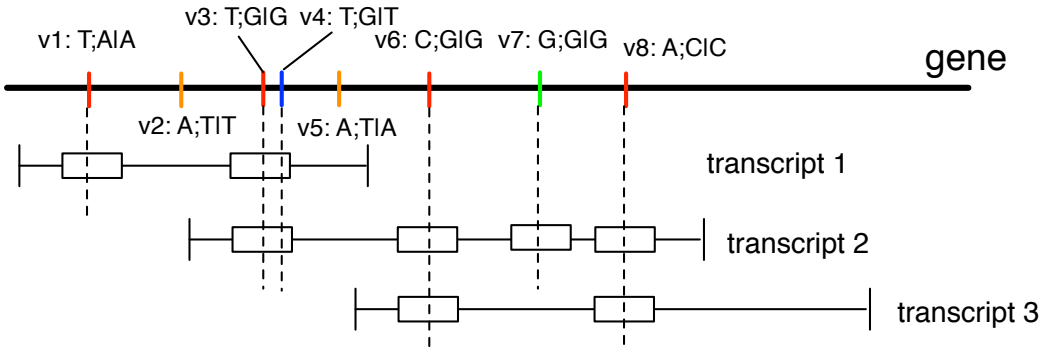


Figure 3.1: An example of variant classification in terms of genotypes. The colored sticks on the gene represent variants at different locations. Colors represent different classes of variants. The format $v1 : T; A|A$ means variant $v1$'s ancestral allele is T and the genotype is $A|A$, the same as other variants. The boxes on the transcripts represent exon regions. The gene and the transcripts share the same coordinate system.

mutations as an example), there are three possible scenarios: trans compound heterozygous (c, the two mutations occur in each copy of a gene, respectively), cis compound heterozygous (d, the two mutations occur in the same copy of a gene), or compound homozygous (e, the two mutations occur in both copies of a gene). This study scored compound mutations as scenario (e) so the two mutations are definite linked on the same allele.

3.2.2 Compensatory Indel Sets

The effects of a deleterious mutation at the sequence level could be compensated for or alleviated by another mutation. For example, a frameshift caused by a one base pair deletion could be recovered by a one base pair insertion nearby. A compensatory indel set is defined as two or more indels that combine to preserve the open reading frame [106]. To simplify the search for compensatory indels, we restrict the consideration of compensatory indel sets, preserving the open reading frame, to those satisfying four conditions: (1) the number of nucleotides inserted or deleted per indel is less than five (≤ 4); (2) the length of each indel is

Table 3.1: Variants sets related to gene ABCB5

Individual ID	Transcript ID	Set ID
NA20805	NM_001163941.1	7619
NA20805	NM_001163942.1	3062
NA20805	NM_001163993.2	3062
NA20805	NM_178559.5	7619
NA20806	NM_178559.5	2807
NA20806	NM_001163993.2	3062
NA20806	NM_001163942.1	3062
NA20806	NM_001163941.1	2807
...	...	...

not divisible by three; (3) the combined length of all indels is divisible by three; (4) all indels in the set occur within 20 base pairs. A single variant in a compensatory indel set is corrected (preserves the reading frame) by combining all other variants in the set. The compensatory indel sets that satisfy the above four conditions for each of the TP53 variants and PTEN variants are considered. The indels are coded with an integer in terms of their lengths and types (insertion or deletion). For example, 'AC/-' represents the deletion of bases A and C at a certain position of a sequence, which is coded with -2; '-/A' represents the insertion of base A at certain position of a sequence, which is coded with 1. With this coding rule, dynamic programming was used to find compensatory indel sets using each single variant as a seed, which is similar to a subset sum problem [64], but with three different sums (-3, 0, and 3). To bound the computational effort, the maximum size of a compensatory indel set is bounded at 10, and the maximum number of compensatory indel sets for each valid length (sums -3, 0 and 3) is bounded at 20. The effects of compensatory indels are evaluated by comparing the HMMvar score of a single variant (as the mutant type) with the HMMvar score of a compensatory indel set (as the mutant type).

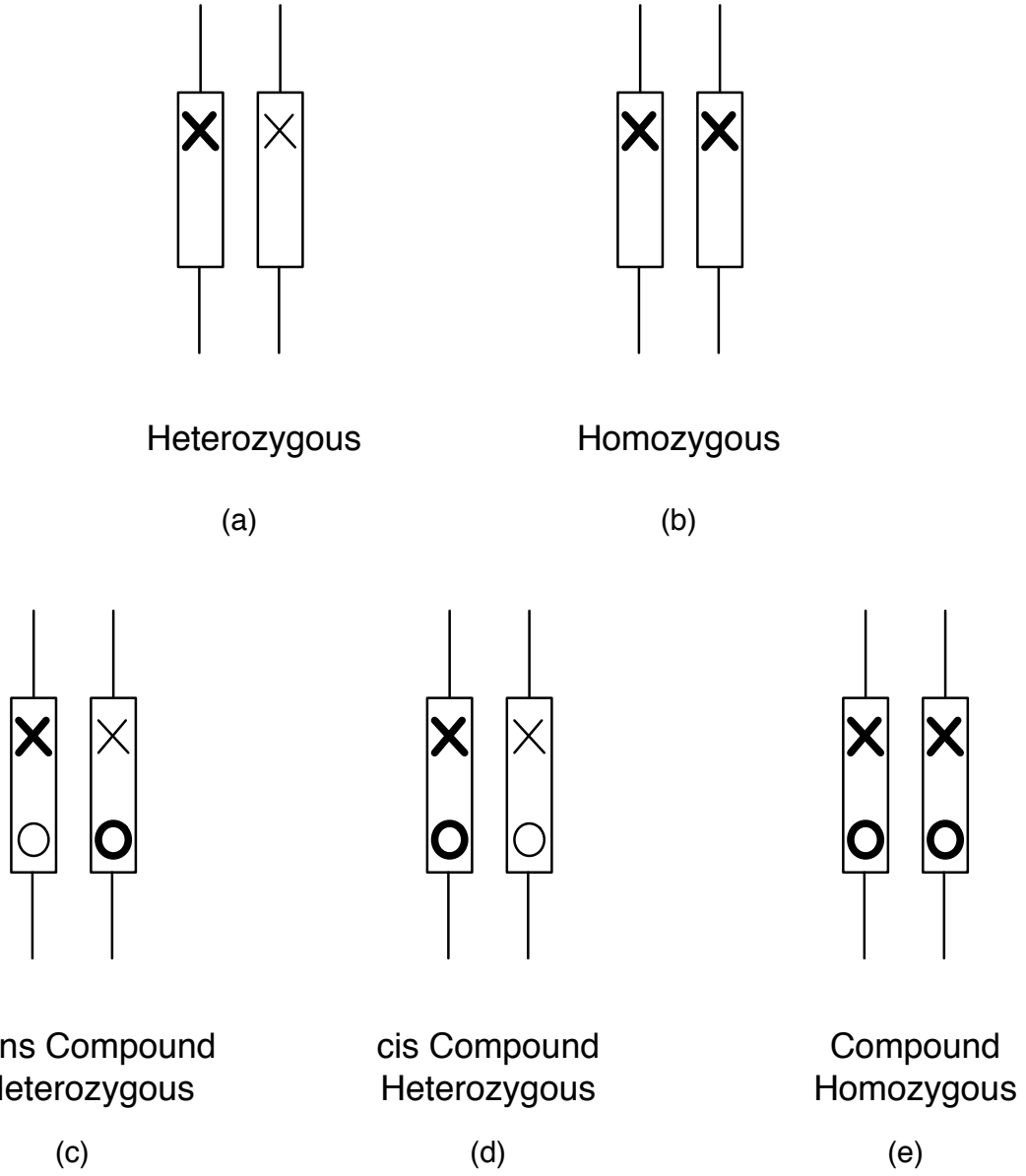


Figure 3.2: Zygosity of disease causing single (a–b) or multiple mutations (c–e). The cross and circle indicate different mutations; the different thicknesses indicate different alleles of a mutation and the thicker one indicates a disease-causing allele; the boxes represent a gene.

3.3 Results

3.3.1 Multiple Variants in the 1000 Genomes Project Data

Only SNPs remain after processing the variants from the 1000 Genomes Project, according to the rules discussed in Section 3.2.1. 67,109 SNP sets are formed and scored. A SNP set may be formed from different transcripts, which results in multiple scores for a set (there are 91,970 set scores in total). Given a SNP set and a transcript sequence, HMMvar measures the deleterious effects of the SNP set using the transcript sequence as the wild type sequence. 291,662 single variants from those SNP sets were gathered and scored. The mean set score distribution is significantly different from the single variant score distribution (one tailed Wilcoxon rank-sum test, $p < 2.2 \times 10^{-16}$). 1000 SNP set scores and 1000 single SNP scores are repeatedly sampled from 91,970 set scores and 275,840 single SNP scores. The cumulative distribution functions of the means of the set scores and single scores are shown in Figure 3.3. Apparently, the HMMvar scores of multiple SNPs tend to be larger than those of the single SNP variants.

Let $V = \{v_1, v_2, \ldots, v_n\}$ be a set of variants v_i $(1 \leq i \leq n)$, S denote the HMMvar score of the set V, and s_1, s_2, ..., s_n be the corresponding single variant scores of v_1, v_2, ..., v_n, respectively. Define V as a compensatory mutation (CM) set if

$$S \leq \min\{s_1, s_2, \ldots, s_n\} - 1.5 * \big(\max\{s_1, s_2, \ldots, s_n\} - \min\{s_1, s_2, \ldots, s_n\}\big).$$

It ensures that the set score is low enough compared to the scores of every single variant in the set. 118 CM sets were obtained from the data set. The CM sets indicate that the deleterious effects of a single variant is compensated by combining it with other variants.

Define V as a noncompensatory mutation (nonCM) set if

$$S \geq \max\{s_1, s_2, \ldots, s_n\} + 1.5 * \big(\max\{s_1, s_2, \ldots, s_n\} - \min\{s_1, s_2, \ldots, s_n\}\big).$$

It ensures that the set score is high enough compared to the scores of every single variant

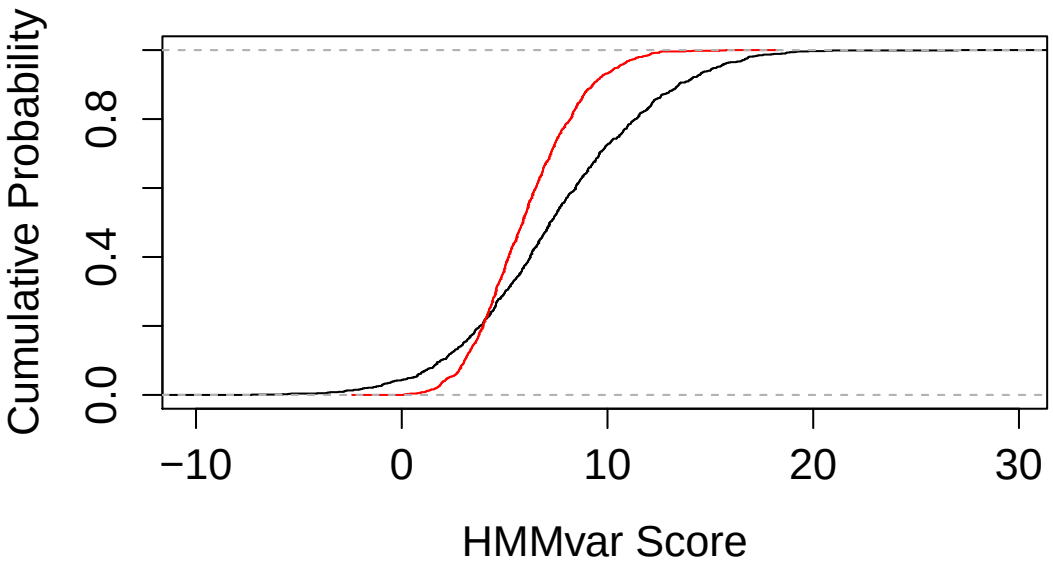


Figure 3.3: Comparison between variant set score (black) and single variant score (red).

in the set. 2392 nonCM sets were obtained from the data set. The nonCM sets indicate the joint effects of multiple neutral variants could possibly result in deleterious effects.

To investigate the single variants in the CM and nonCM sets, all the single variants from all the CM sets and all the nonCM sets are gathered, respectively. The allele frequency distributions from these two groups are compared in Figure 3.4. When the allele frequency is less than 0.1, the proportion of the nonCM variants is greater than that of the CM variants. This is probably because the single variants are so deleterious that in most cases, the joint effects of these deleterious variants is still deleterious. However, when the allele frequency is in the range of 0.1 to 0.3, the signal of the compensatory mutation effect is boosted.

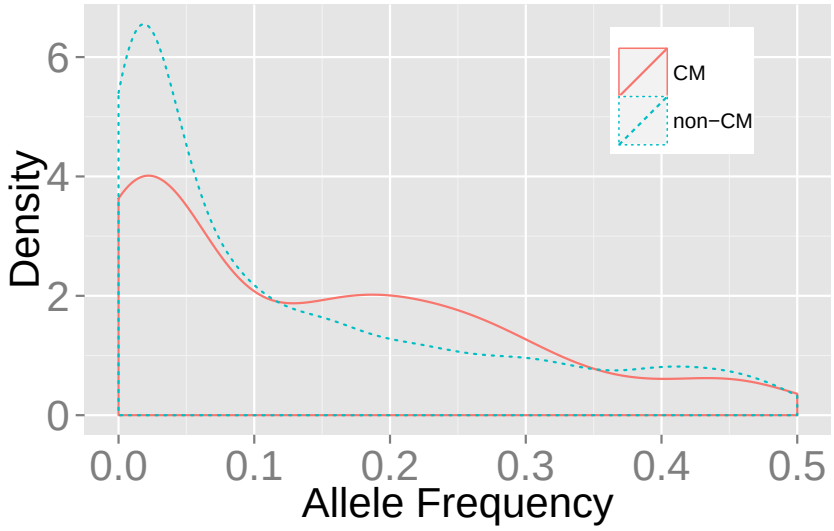


Figure 3.4: Allele frequency distribution of SNP variants in CM sets and nonCM sets.

3.3.2 Compensatory Indels in TP53 and PTEN

The indels in two tumor suppressor genes, TP53 and PTEN, are collected from IARC [76] and COSMIC [34] database, respectively. The 4,736 variations (3,565 for TP53 and 1,171 for PTEN) include frameshift or in-frame insertions, deletions, and complexes (both insertion and deletion take place simultaneously in one location) in coding regions (Table 3.2).

In TP53, 1,039 variants were found that met the criterion for belonging to a compensatory indel set, out of 3,565 variants. The deleterious functional effects caused by these variants can be greatly weakened by compensatory indels as measured by HMMvar scores. A single variant may be different compensatory indel sets due to different combinations. Figure 3.5(a) shows the HMMvar score of a single variant versus the median of the HMMvar scores of the corresponding compensatory indel sets. It is obvious that the effects of a single variant (high HMMvar score) is neutralized by the compensatory indel sets (low HMMvar score). It is likely that single variants nearby share the same compensatory indel set, so many set scores shown in Figure 3.5 are approximately the same (≈ 1).

PTEN is also an intensively studied tumor suppressor gene. Figure 3.5(b) shows the HMMvar

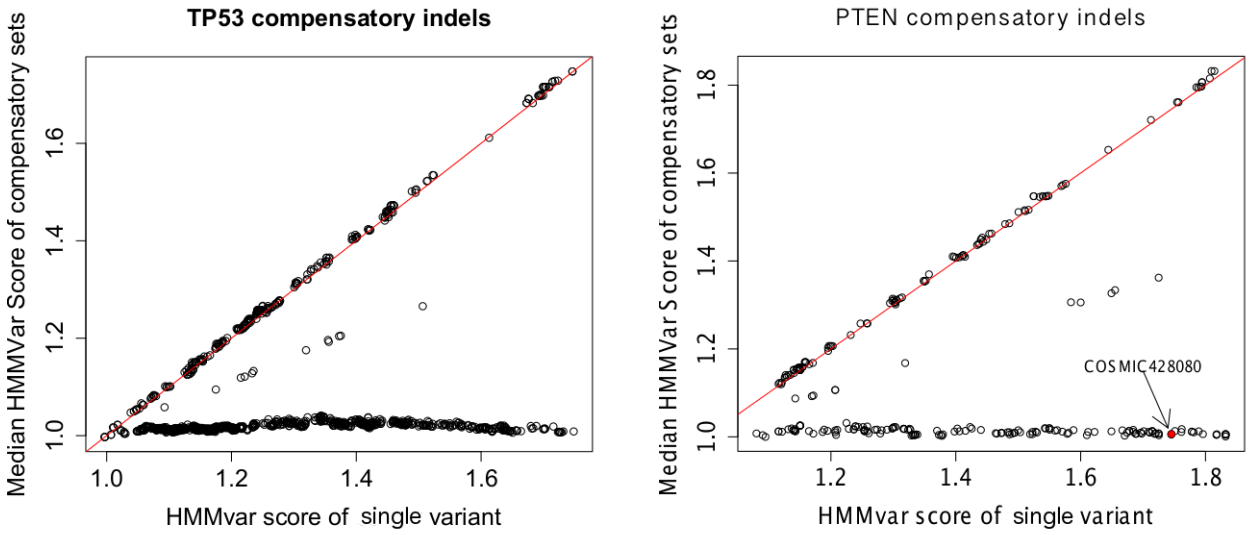
Table 3.2: Data Description

Type \ Database	IARC (TP53)	COSMIC (PTEN)
Insertion (in-frame)	90	7
Insertion (frameshift)	419	116
Deletion (in-frame)	364	43
Deletion (frameshift)	1016	283
Complex (in-frame)	94	8
Complex (frameshift)	53	19
Total	3565	1171

score of 246 variants versus the median HMMvar score of the corresponding compensatory indel sets, which shows the same trend as the TP53 variants. This scoring procedure provides candidate compensatory indel sets, which, when substituted for the indel, ameliorate the deleterious effects of that single mutation. For instance, the deleterious variant c.142delA (COSMIC428080) associated with skin cancer [54] has HMMvar score 1.75; however, with compensatory indels, the deleterious effects can be decreased to an HMMvar score of 1.07. At the same time, the results here demonstrate the importance of scoring multiple variants together, instead of individually, to understand their joint effects.

3.3.3 A Case Study: Cardiovascular Disease

Studies [51] have shown that single mutations in two genes, β-myosin heavy chain (βMHC) and myosin-binding protein C (MyBP-C), can lead to genetic cardiovascular disease, and multiple mutations on these same genes can lead to more severe cardiovascular disorders and even death. As a test case for HMMvar's capability in predicting the effects of multiple variants compared to the effects of single variants, the multiple mutations that have been shown to increase the severity of cardiovascular disease from single mutations are scored



(a) (b)

Figure 3.5: Scatter plot of HMMvar score of a single variant versus the median HMMvar score of the corresponding compensatory indel sets for the TP53 gene and the PTEN gene. The red line is $y = x$. (a) TP53 compensatory indels. (b) PTEN compensatory indels. The red solid circle marks the COSMIC variant with ID 428080.

in these two genes. As shown in Table 3.3 for both genes, compound mutations all have higher scores than single mutations, consistent with the notion that compound mutations in these genes cause more severe cardiovascular disease than single mutations. The set score effectively reflects the cumulative effects of the single mutations. Except for the compound mutations involving nonsense mutation (replaced by a stop codon, 'Ter'), the maximum score for compound mutations in the βMHC gene is the combination of Arg719Trp and Met349Thr, which has been reported causing sudden death [51]. It is noted that in Table 3.3, the set score ($S_{1,2}$) is always approximately the sum of the single scores ($S_1 + S_2$), indicating the cumulative effect of compound mutations. We propose the Cumulative Conjecture to show the capability of HMMvar-multi of capturing the cumulative effect of compound mutations (See Appendix A for detailed illustration).

Conjecture 1 *If one path is dominant in the profile HMM in terms of transition probabilities, the HMMvar score of the compound SNPs is approximately the sum of the HMMvar scores of individual SNPs.*

Table 3.3: Scoring multiple mutations in βMHC and MyBP-C genes

Gene	Mutation1	Score1 (S_1)	Mutation2	Score2 (S_2)	Set Score($S_{1,2}$)
βMHC	Val39Met	1.7	Arg723Cys	3.4	**5.0**
βMHC	Arg54Ter	15.8	Arg870His	2.2	**17.9**
βMHC	Pro211Leu	2.4	Arg663HIs	2.2	**4.5**
βMHC	Met349Thr	2.2	Arg719Trp	3.1	**5.3**
βMHC	Arg663His	2.2	Val763Met	2.3	**4.4**
βMHC	Arg719Gln	1.7	Thr1513Ser	0.0	**1.6**
βMHC	Asp906Gly	2.7	Leu908Val	2.0	**4.6**
MyBP-C	Gly5Arg	1.7	Arg502Trp	3.9	**5.7**
MyBP-C	Gln76Ter	15.6	His257Pro	4.0	**19.6**
MyBP-C	Arg502Trp	3.9	Ser858Asn	2.4	**6.4**
MyBP-C	Glu542Gln	2.2	Ala851Val	2.2	**4.4**
MyBP-C	Asp745Gly	3.9	Pro873His	4.0	**7.9**
MyBP-C	Arg810His	2.9	Arg820Gln	2.6	**5.5**
MyBP-C	Gln1233Ter	0.0	Arg326Gln	1.0	**1.0**

When multiple mutations occur and accumulate on the same gene, it is possible that though deleterious by themselves, they come together and become less deleterious or even beneficial to the carrier due to either recovery of the original gene function or gain of new function. This type of mutation, known as compensatory mutation, has been documented in the literature with many of the cases found in bacteria and viruses [37, 71, 108]. Potential compensatory indels were identified in two tumor suppressor genes, TP53 and PTEN, where compensatory indels are composed of frameshift indels that can recover the original reading frame. Results show that the HMMvar scores for the effects of compensatory indels are indeed much lower

than the scores of the frameshift indels, with many of them close to one (Figure 3.5), suggesting that compensatory indels can rescue the deleterious effects of frameshift indels. Similarly, Figure 3.4 shows that SNPs with putative compensatory effect (CM) tend to have higher frequencies in the 1000 Genomes data than those SNPs predicted to be noncompensatory (nonCM, Figure 3.4).

HMMvar can predict the effects of a set of multiple variants in its entirety. This is especially useful when multiple variants occur in a protein, each of which may have deleterious effects on the protein function, but the combination of them may be less deleterious due to a compensatory effect. Profile HMMs, used as proposed, have the capability to predict the joint effect of multiple mutations along the gene given a specific haplotype. Due to current technological limitations, inferring genotypes of a gene is still a challenge, and little data exists that can be used for understanding the effects of multiple variations on the same gene. With future sequencing technology, long sequences may be generated and genotypes of a gene may be determined with certainty, in which case the HMMvar method will be of great use in understanding the joint effect of multiple mutations, in addition to single mutations, and better identification of disease contributing/causing variations.

Chapter 4

Predicting the Functional Outcome of Variants

4.1 Introduction

Over 79 million genetic variants have been identified in 2535 humans from 26 populations around the world (the 1000 Genomes Project, 06/2014). The sheer numbers of these variants poses a significant challenge for researchers to empirically examine their individual or collective phenotypic or pathological effects and to identify the ones that are important determinants for phenotypes or diseases. Consequently, to help narrow down target variants that may have a phenotypic and/or pathological effect, various computational tools (e.g., [4, 14, 19, 53, 75, 79]) have been introduced to predict the effects of genetic variants. Specifically, these tools provide either a quantitative score indicating the degree of harm of the variant (e.g., [14, 53, 75, 79]), or a qualitative statement of whether the variant is deleterious or neutral (e.g., [45]). These tools have been developed to predict the fitness effects (i.e., neutral, deleterious, or beneficial) of genetic variants on the corresponding proteins. However, prediction in terms of whether a variant causes the variant bearing protein to lose the original function or gain new function is also needed for better understanding how the

variant contributes to disease or cancer. To address this problem, we introduce and computationally define four types of functional outcome of a variant: gain, loss, switch, and conservation of function. Biologically, loss of function (LoF) mutations cause the gene product to have reduced activity or complete loss of function; gain of function (GoF) mutations change the gene product to have a new and possibly abnormal function; switch of function (SoF) mutations cause the gene product to switch from one set of functions to another set of functions [81], thus may involve both loss of the original functions and gain of new functions; conservation of function (CoF) mutations, coined in this study, refer to mutations that are neutral and do not alter gene functions. A graphic view of these definitions is shown in Figure 4.1.

Fine grained prediction of the functional outcome of variants on these four types can be used to help elucidate the molecular mechanism of disease or cancer causing mutations. For instance, two important classes of genes, oncogenes and tumor suppressor genes, when mutated, can both lead to cancer. However, the effects that mutations have on these cancer causing genes are almost the opposite. Mutations in oncogenes can keep the genes stuck in a state of constant activity. A proto-oncogene converted into an oncogene generally involves a GoF. For example, BRAF is a proto-oncogene, and there is a well-known GoF mutation that replaces the amino acid valine with the amino acid glutamic acid at position 600 (V600E). The V600E mutation can cause 500-fold increased activation, stimulating the constant activation of the mitogen-activated protein kinase (MEK) signaling, which leads to tumor cells [30]. This mutation has been frequently found in the skin cancer called melanoma [3]. On the other hand, mutations of a tumor suppressor gene result in the gene losing the ability to prevent or suppress abnormal cells from developing into full-blown tumors, which are essentially LoF mutations. An example can be seen in the PTEN gene, one of the most common down-regulated tumor suppressor gene in a cancer genome. Substitutions of some of its important residues, such as D92 and H93, result in significantly reduced PTEN function [83]. Therefore, identifying different types of mutations in terms of functional impacts helps one understand the driven event and the identification of novel targets, which

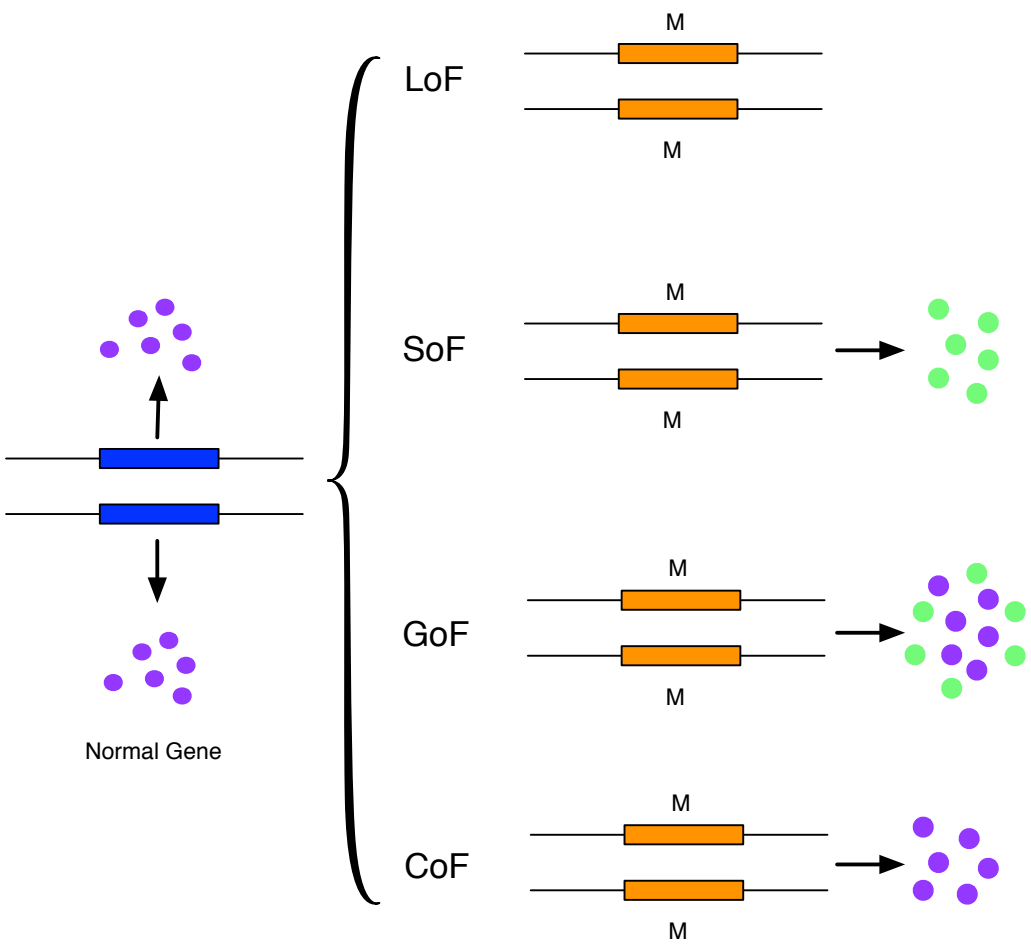


Figure 4.1: The consequences of LoF, SoF, GoF and CoF mutations (M). The normal gene is indicated by blue box and mutated gene is indicated by orange box. The original functions are represented by blue circles and the new functions are represented by green circles.

is crucial for the development of targeted disease and cancer therapeutics.

An MSA captures the evolutionary information within homologous sequences. Evolutionary analysis provides a powerful tool for predicting the functional impact of mutations. Presumably, a profile HMM built from the MSA is an implicit representative of a set of functions of the protein family or subfamily. Based on the fitness of a sequence within a family or across subfamilies, different types of mutations are defined. The LoF mutations weaken the fitness of the wild type sequence within the protein family, whereas the GoF mutations make

the mutant type sequence fit better than the wild type sequence in one of the subfamilies. The SoF is in the middle of LoF and GoF, which loses functions from the original protein subfamily but gains functions from other subfamilies. CoF means the mutation does not cause any functional changes.

Earlier work addressed prediction of the functional type of variants [57, 67] by trying to identify activating variants, but none provides a precise computational classification definition for all these types: LoF, SoF, GoF and CoF mutations. In this chapter, we propose HMMvar-func to computationally classify coding variants into four types on the basis of HMMvar [61].

4.2 HMMvar-func: Predicting the Functional Outcome of Variants

4.2.1 Building Multiple HMMs

HMMvar [61] quantitatively predicts the functional effects of variants. It builds an HMM based on the MSA of a set of sequences homologous to the wild type sequence. Then the wild type protein sequence and mutant type protein sequence are matched against the HMM, respectively. HMM can calculate the fitness or similarity between the sequence and the protein family represented by a profile HMM. If the mutant type sequence scores almost the same as the wild type sequence, the mutation has little effect on the protein function. To identify different types of mutations, the MSA of homologous sequences are clustered and each cluster is viewed as a "subfamily", which captures specific functions. If a mutant sequence fits better than the corresponding wild type sequence in one of the subfamilies, then it is likely the variant enables the protein to "acquire" new functions. With this assumption, we identify subfamilies by clustering the MSA generated from homologous sequences, including the query sequence. Each of the subfamilies represents a functional profile.

The pipeline is shown in Figure 4.2. First, homologous sequences to the protein with mutations are identified by PSI-BLAST [2] against the UniProt90 [98] database. Then the homologous sequences with identity percentage over 50% to the query sequence are aligned by the algorithm MUSCLE [29] with parameters "-maxiters 1 -diags -sv -distance1 kbit20_3". To ensure the quality of the MSA, further processing was performed. First, redundant sequences are removed. If the identity percentage between the aligned positions of any two sequences in the alignment exceeds a threshold (95%), the shorter sequence is discarded. Then low quality columns (those with the number of gaps exceeding a threshold (99%)) are discarded. Given a variant, a region of the MSA is selected by left and right extension from the position of the variant, keeping the query sequence consecutive in the MSA. Finally, empty rows are removed (rows with all gaps).

With the post-processed MSA, the combinatorial entropy optimization algorithm [80] is used to perform the clustering. This algorithm minimizes the combinatorial entropy across different clusters over all the positions in the MSA. By minimizing the combinatorial entropy, we can find a partition of the MSA such that the columns are conserved in a subfamily (cluster) but differ between subfamilies. The detailed clustering algorithms are discussed in the next section. For each of the clusters, a profile HMM is built, which represents a subfamily or specific functions that differ from those of the target cluster; then HMMvar can be used to score the variants. Denote these subfamilies by C_0, C_1, ..., C_{k-1}, where C_0 is the target cluster that contains the wild type sequence, and the corresponding HMMs as H_0, H_1, ..., H_{k-1}. Only the clusters with size greater than one are used for prediction in the pipeline.

4.2.2 Clustering of MSA

A protein family is a group of proteins that share functions but that have diverged from a common evolutionary origin and thus have similar functions and structures. Because the sequence members in a protein family share similar functions, it is reasonable to identify the

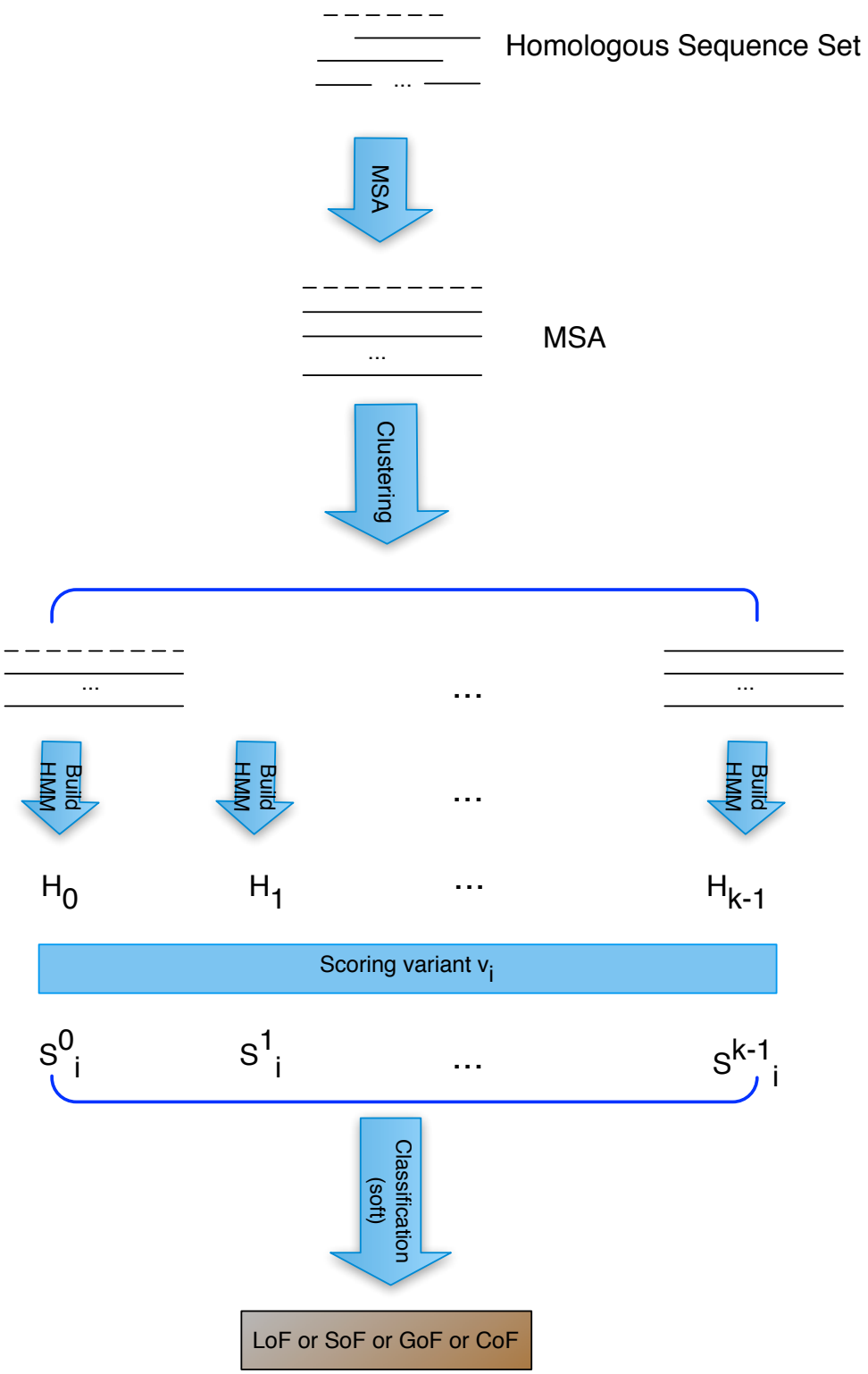


Figure 4.2: Flowchart of the classification procedure (the dashed line represents the wild type sequence).

subfamilies within a protein family and predict the functional transition among subfamilies due to the variants. Two algorithms, K-means [48] clustering and combinatorial entropy optimization (CEO) [80], are used for the clustering step in the pipeline.

K-means clustering is a traditional unsupervised learning algorithm for clustering. Given a set of points $(s_1, s_2, \ldots, s_n)$, K-means clustering aims to partition the n points into k ($\leq n$) sets, so as to minimize the objective function,

$$J = \sum_{j=1}^{k} \sum_{i \in I_j} \rho(s_i, c_j), \tag{4.1}$$

where I_j is the index set of the points s_i in the jth cluster with representative c_j, and $\rho(s_i, c_j)$ is a measurement of the distance between s_i and c_j. The sets I_j, $j = 1, \ldots, k$, partition the set $\{1, \ldots, n\}$.

The algorithm assigns a given data point to one of the k clusters according to the distance between the data point s_i and the cluster representative c_j. The k representatives, one for each cluster, are initially defined. Once all points are assigned, the k representatives c_j are replaced by the centroids of the clusters. The above steps are iteratively implemented until no more changes occur in the k representatives. It can be proved that the iterative procedure will always converge. However, the K-means algorithm does not guarantee an optimal configuration, corresponding to the globally minimized objective function. Besides, the algorithm is very sensitive to the initial randomly selected representatives.

There are two uncertain factors in the clustering of MSA using the K-means algorithm. One is the number of clusters k, which needs to be predetermined or at least well-estimated, and the other is the method for generating new representatives. There are several methods to generate the cluster representatives. For example, one can pick the new representative of a cluster as the medoid, which is a member of a cluster whose average dissimilarity to all the members in the cluster is minimal. However, it is time consuming to calculate the medoid of a big cluster (e.g., $n \geq 50$). We can also encode the amino acid sequence using real values and calculate the mean of the sequences (centroid) in a cluster as the new

representative. However, it is difficult to encode the amino acids efficiently. For simplicity, the most frequently occurring amino acid (and gap) at each column of the MSA is used to form the representative sequence. The distance between sequences is defined as edit distance.

The combinatorial entropy optimization (CEO) algorithm finds the clusters by (locally) minimizing the combinatorial entropy of the MSA. The combinatorial entropy of an MSA is

$$\Delta S = \sum_{m=1}^{M} \sum_{i=1}^{L} \Delta S_i^{(m)}, \tag{4.2}$$

where L is the length of the MSA, M the number of clusters, and $\Delta S_i^{(m)}$ the entropy difference of cluster m at position i. $\Delta S_i^{(m)}$ is computed as

$$\Delta S_i^{(m)} = \ln \frac{N_m!}{\prod_\alpha n_i^{(m)}(\alpha)!} - \ln \frac{N_m!}{\prod_\alpha \Gamma(n_i(\alpha) N_m / N + 1)} \tag{4.3}$$

where N_m is the number of sequences in cluster m, α one of the 20 amino acids (plus the gap), $n_i^{(m)}(\alpha)$ the number of amino acids α (or the gap) at position i in cluster m, $n_i(\alpha)$ the number of amino acids α (or the gap) at position i in the entire MSA, and $n_i(\alpha) N_m / N$ ($N = \sum_{m=1}^{M} N_m$) the expected number of amino acids α (or the gap) at position i in cluster m in terms of a uniform distribution. By minimizing the combinatorial entropy, the CEO algorithm finds the clusters considering not only the conservation in the entire MSA, but also the conservation in subfamilies.

A straightforward solution to this optimization problem would be to enumerate all possible partitions of the sequence set into subfamilies, compute the combinatorial entropy, and then choose the partition with the lowest combinatorial entropy. However, this is computationally infeasible for all but very small M and N. A simple agglomerative hierarchical clustering is used to heuristically find a locally optimal partition. Each sequence is in a cluster by itself at the initial state. Two clusters are merged at each step until no clusters can be merged. The two clusters chosen for merging, a and b, are the two clusters closest to each other, the distance between two clusters a and b being defined as

$$\Delta Q_{a,b} = A \Delta S_{a,b} + (1 - A) \Delta S'_{a,b}, \tag{4.4}$$

where $0 \leq A \leq 1$, $\Delta S_{a,b}$ is the entropy difference of the new cluster resulting from merging clusters a and b, $\Delta S'_{a,b} = \ln(N_a + N_b)$ is a penalty term, N_a is the number of sequences in cluster a, and N_b the number of sequences in cluster b. The granularity parameter $A \in [0, 1]$ controls the merging step in the hierarchical clustering. If $A = 1$, the contribution of merging two clusters comes from the entropy difference, whereas if A is very close to 0, the contribution comes from the sizes of the two clusters, that is, the smaller the two clusters are, the more likely they are merged. By tuning the granularity parameter A, we can avoid favoring the merging of two big clusters in the hierarchical clustering.

The final solution is chosen as the minimum of ΔS (Equation (4.2)) over all clustering steps and penalty weights ($A \in [0, 1]$). A ranging from 0.6 to 0.9 was suggested by the paper [80]. In our implementation, A is fixed at 0.75, as the minimum is only over the hierarchical clustering.

4.2.3 Classification by HMMvar Scores

HMMvar [61] predicts the harmfulness of variants and only one HMM is built from the MSA of all the homologous sequences. In this chapter, multiple HMMs are built, one HMM for each of the k clusters. For a given variant v_i, let S_i^m ($0 \leq m \leq k-1$) denote the quantitative HMMvar score of variant v_i obtained from H_m. H_0 is the HMM built from the target cluster C_0 that contains the wild type sequence (Figure 4.2), and S_i^0 is the score of variant v_i calculated from H_0. With all the scores calculated from multiple HMMs, we assign a functional type to each of the variants using two classification strategies, hard classification and soft classification. Soft classifiers estimate a conditional probability for each of the class labels and perform classification based on the estimated probabilities. By contrast, hard classifiers directly use the decision boundary of different classes without producing any probability estimations.

The hard classification method was used in the previous work [59]. The decision tree for hard classification of different types of mutations is shown in Figure 4.3. $S_i^0 > u$ indicates that in

the target subfamily, the wild type sequence fits better than the mutant type sequence, which is an indication that the variant is a LoF mutation. Further, if for all other subfamilies, the wild type sequence fits better than the mutant type sequence, this variant is classified as a LoF mutation. Otherwise, it is classified as SoF, because, although the variant may cause the protein to lose the function represented by C_0, it may enable the protein to obtain the specific function represented by some C_m. On the other hand, if $S_i^0 \leq t$ and if there exists at least one other subfamily that the mutant type sequence fits better than the wild type sequence, the variant is classified as GoF, otherwise, CoF.

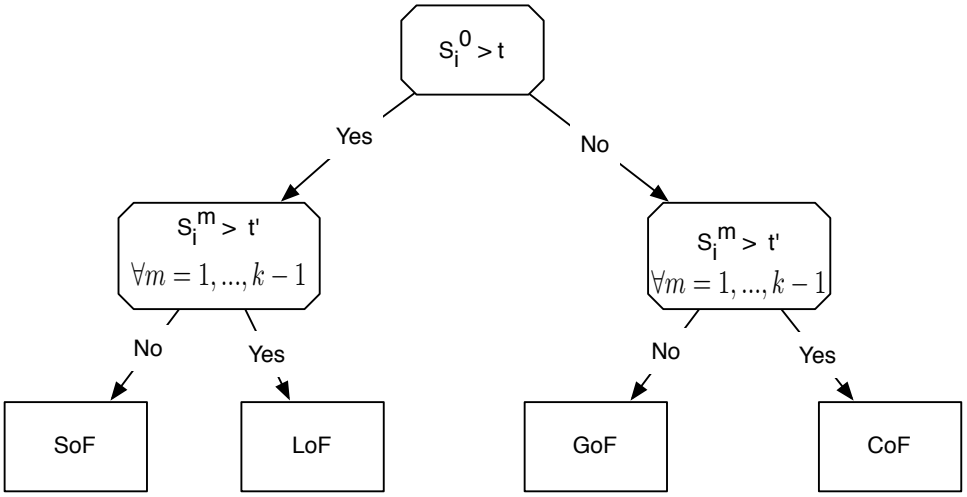


Figure 4.3: Decision tree for predicting the functional outcome of variants with hard classification.

Soft classification is based on a different approach compared to hard classification and has its merits. Because the HMMvar scores are sensitive to the quality of the MSA, a soft classification strategy is more appropriate. Given a variant v_i, the probability L_i^0 of losing the original functions from C_0 and the probability A_i^x of acquiring new functions from C_x are defined by

$$L_i^0 = \frac{1}{1 + e^{-(S_i^0 - u)}}, \tag{4.5}$$

$$A_i^x = \frac{1}{1 + e^{-(u - S_i^x)}}, \tag{4.6}$$

where S_i^0 is the score calculated from H_0, $S_i^x = \min_{1 \le j \le k-1} S_i^j$, and u is the user defined cutoff. The logistic functions correspond to assuming that the odds ratios for L_i^0 and A_i^x are linear in the threshold u. Assuming the independence of losing the original function and acquiring new function, the confidence scores are $L_i^0 * (1 - A_i^x)$, $L_i^0 * A_i^x$, $(1 - L_i^0) * A_i^x$, and $(1 - L_i^0) * (1 - A_i^x)$ for LoF, SoF, GoF, and CoF mutation respectively.

The binary tree in Figure 4.4 demonstrates how the confidence score for different types is calculated. The mutation type corresponding to the maximum probability (confidence score) is taken as the predicted type. If there is a tie for the maximum probability, the tie is broken by the order LoF, SoF, CoF, GoF. For a given variant v_i and predefined cutoff u, $S_i^0 > u$ indicates that in the target subfamily, the wild type sequence fits better than the mutant type sequence, so it has a higher probability of losing the original function. Further, if for the subfamilies x, from which the minimum HMMvar score is calculated, the wild type sequence fits better than the mutant type sequence, then it indicates no new function is acquired and results in LoF ($L_i^0 > 0.5$ and $A_i^x < 0.5$). Otherwise, v_i is classified as SoF ($L_i^0 > 0.5$ and $A_i^x > 0.5$) with higher confidence score because although the variant probably causes the protein loss the function in subfamily C_0, v_i obtains the specific function in some C_m. On the other hand, if $S_i^0 \le u$, the variant could potentially cause GoF. Then if the mutant type sequence fits better in subfamily x ($S_i^x < u$), which means there exists at least one other subfamily that the mutant type sequence fits better than the wild type sequence, the variant v_i is classified as GoF ($L_i^0 \le 0.5$ and $A_i^x > 0.5$) with higher confidence score, otherwise, v_i is classified with CoF ($L_i^0 \le 0.5$ and $A_i^x < 0.5$).

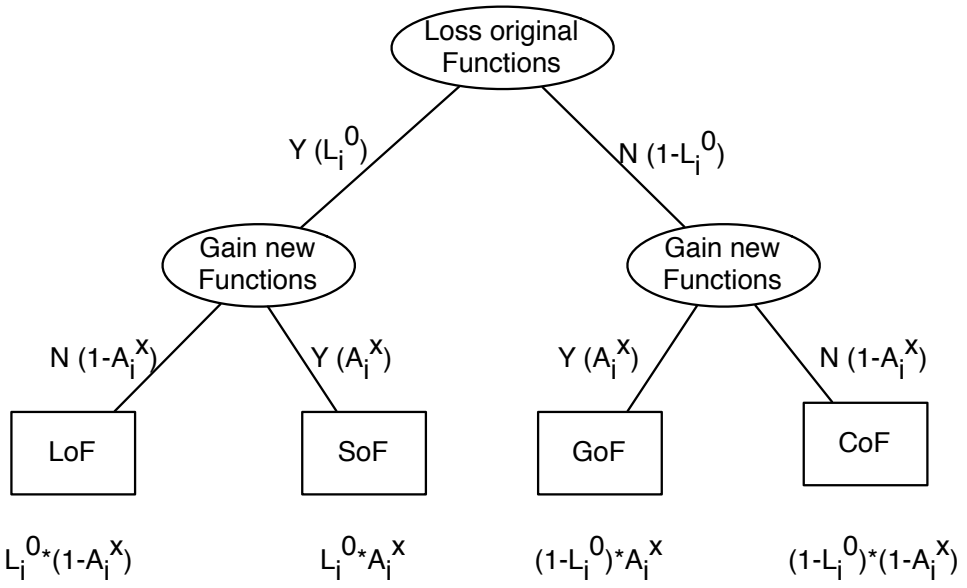


Figure 4.4: The probability combination rule for the classification of mutations

4.3 Results

4.3.1 Thyrotropin Receptor Gene Mutations

Thyroid stimulating hormone (TSH, thyrotropin) and its receptor TSHr together play a key role in controlling thyroid function. The TSHr gene provides instructions for making a receptor that serves as a customized binding site for TSH. Several TSHr gene mutations have been identified in people who are insensitive (or resistant) to TSH. In some cases, this resistance causes congenital hypothyroidism. Hypothyroidism is caused by too little thyroid hormone secreted by the thyroid, which results in the level of thyroid-stimulating hormone in the blood increasing. Somatic mutations in the TSHr gene have been identified in thyroid tumors, which are found in tumor cells but not in the cells from normal tissues. TSHr gene mutations can cause disorders in which the thyroid gland is overactive (hyperthyroidism). Hyperthyroidism mutations change one of the building blocks (amino acids) used to make the thyroid stimulating hormone receptor. As a result, the receptor is continuously activated

and overstimulates the production of thyroid hormones.

Mutations of the TSHr gene can be LoF or GoF depending on their nature, leading to hypo or hyperthyroidism, respectively. The discovery of large serial GoF mutations in the TSHr gene is of great interest, revealing a new disease mechanism of mutations that constantly increase the basal activity of a receptor [26].

111 TSHr mutations are extracted from the TSH Receptor Mutation Database II [101]. These mutations include sporadic, family, or nodule mutations. They are all nonsynonymous SNPs. 61 out of 111 are GoF that constitutionally activate the receptor independently of TSH; the remaining 50 are LoF mutations that result in the loss of TSH sensitivity. The confusion matrix is shown in Table 4.1. Three variants are not available for the prediction, because the built HMMs are not significant for scoring. Only the predicted LoF (25) and GoF (39) are used to calculate the performance metrics (accuracy, sensitivity, and specificity), since there are only two types (LoF and GoF) in the data set. Figure 4.5 shows the ROC with respect to u for HMMvar-func based on CEO clustering. The best performance is achieved at $u = 2.7$ with sensitivity 78.9%, specificity 65.4%, and accuracy 73.4%. The predicted types with high confidence scores are more reliable, thus it is reasonable to focus on these variants, which also avoids the ambiguity of confidence score ties. Considering only the variants with maximum confidence score greater than 0.5 (43 in total, 21 GoF and 22 LoF), the sensitivity, specificity, and accuracy are 85.7%, 68.2%, and 76.7%, respectively. The detailed scores can be accessed at [62]. The CEO algorithm finds the optimal number of clusters to minimize the combinatorial entropy [80]. Due to the processing of the MSA, the MSA used for the clustering step is a segment of the original MSA, and this segment is possibly different with regard to a specific variant. As a result, the number of clusters generated by the CEO algorithm is not fixed for all the variants. The average number of clusters generated in this date set is 19 from 162 sequences in the original MSA (exclude the clusters with size 1).

Two aspects of the HMMvar-func prediction method merit investigation, the clustering method and the cutoff score u set in Figure 4.5. The present work uses the CEO algo-

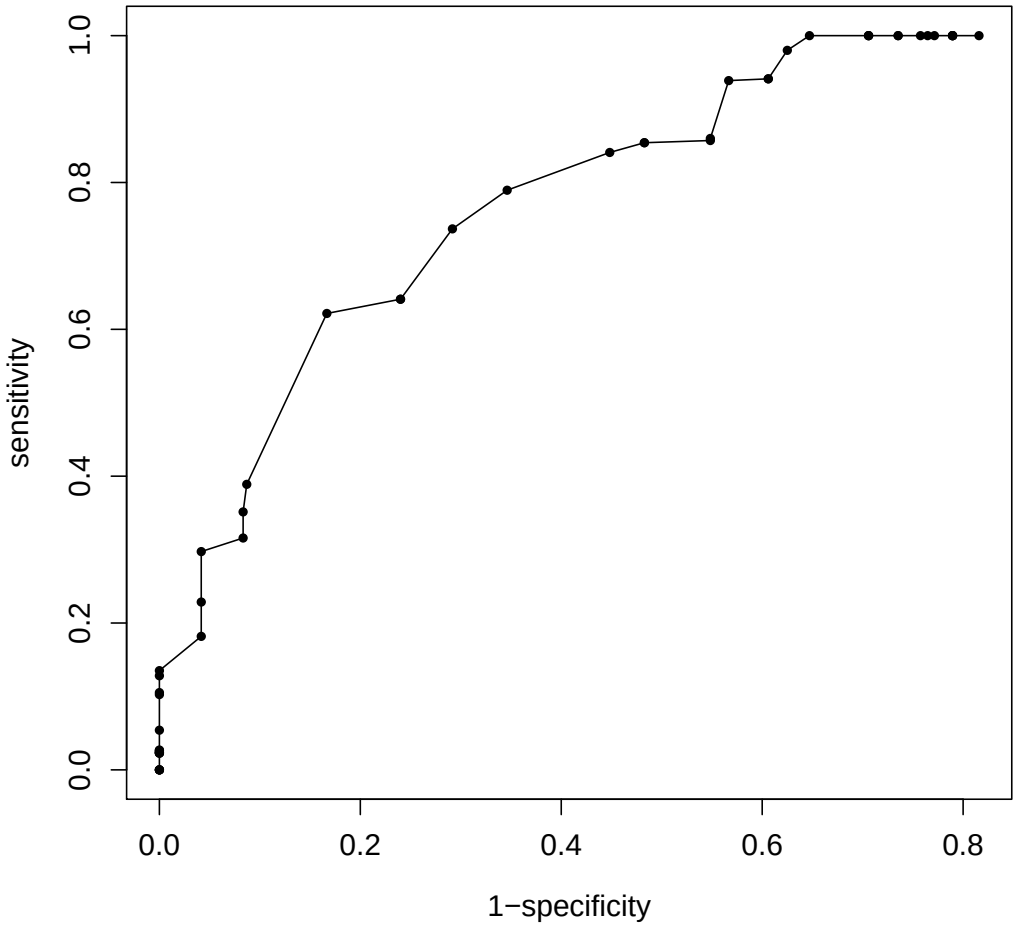


Figure 4.5: Receiver operating curve (ROC) for prediction of TSHR mutations (sensitivity is with respect to GoF; the AUC is 0.613.).

Table 4.1: The confusion matrix of the prediction results for the TSHr data set

	GoF	LoF	SoF	CoF
GoF	30	8	23	0
LoF	9	17	19	2

rithm suggested in [80]. The K-means clustering method, used in previous work [59], was compared with the CEO algorithm. Figure 4.6 shows the cost function 4.1 with respect to the number of clusters; as the number of clusters increases, the cost function decreases, more

Table 4.2: The comparison of CEO and K-means

	Dunn	Davies-Bouldin	accuracy	sensitivity	specificity
CEO	0.429	0.838	0.654	0.667	0.638
median K-means	0.378	0.973	0.574	0.569	0.560
best K-means	0.513	0.839	0.679	0.742	0.600

dramatically from $k = 1$ to $k = 4$, then levels off as $k > 4$. Next, $k = 4$ are selected in terms of the "elbow criterion", and K-means clustering is compared with the CEO clustering method with the cutoff scores $u = 2.7$ as shown in Figure 4.7. The K-means clustering is extremely sensitive to the initial guesses, so 100 runs with random initial guesses were performed to reduce this effect. The number of clusters generated by the CEO method was controlled to be the same as the K-means clustering ($k = 4$) for fair comparison in this case. Figure 4.7 shows that the CEO statistics are much better than what would be expected from using K-means, but that the CEO clusters are not optimal, and a lucky K-means clustering can do much better than CEO.

The inner coherence of the clusters generated by CEO and K-means is also compared in Table 4.2. The "median" and "best" K-means are defined in terms of the median and best accuracy shown in Figure 4.7, respectively. The Dunn index and Davies-Bouldin index are consistent with the accuracy metrics. Better cluster quality corresponds to a higher Dunn index and a lower Davies-Bouldin index.

As expected, results here demonstrate that both the clustering method and the cutoff score u can affect the prediction results. The better the cluster quality is, the more accurate the prediction is. Since there is no consensus on which clustering method works best, it is advisable to compare the quality of the clusters from several different methods and possibly use consensus clusters for downstream prediction.

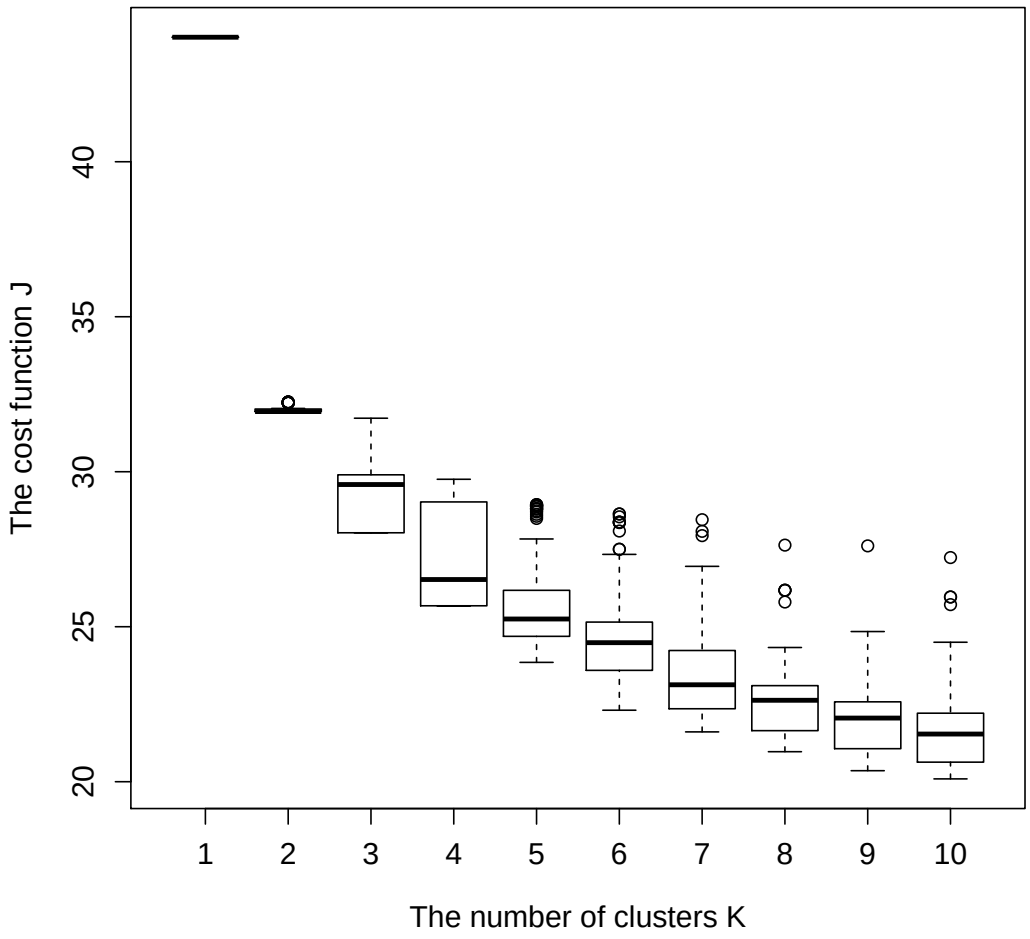


Figure 4.6: The cost function with respect to the number of clusters for K-means clustering with 100 runs for each k, $n = 162$.

4.3.2 Application in Cancer Mutations

Activating mutations [102] in two oncogenes, epidermal growth factor receptor (EGFR) gene and B-Raf proto-oncogene, serine/threonine kinase (BRAF) gene are predicted. The activating variants related to these two genes can be accessed at [62]. In addition, the mutations in an extensively studied suppressor genes, tumor protein p53 (TP53), are also evaluated. The variants related to TP53 are from the IARC TP53 database [76]. The detailed description of this data set can be found in Chapter 2.

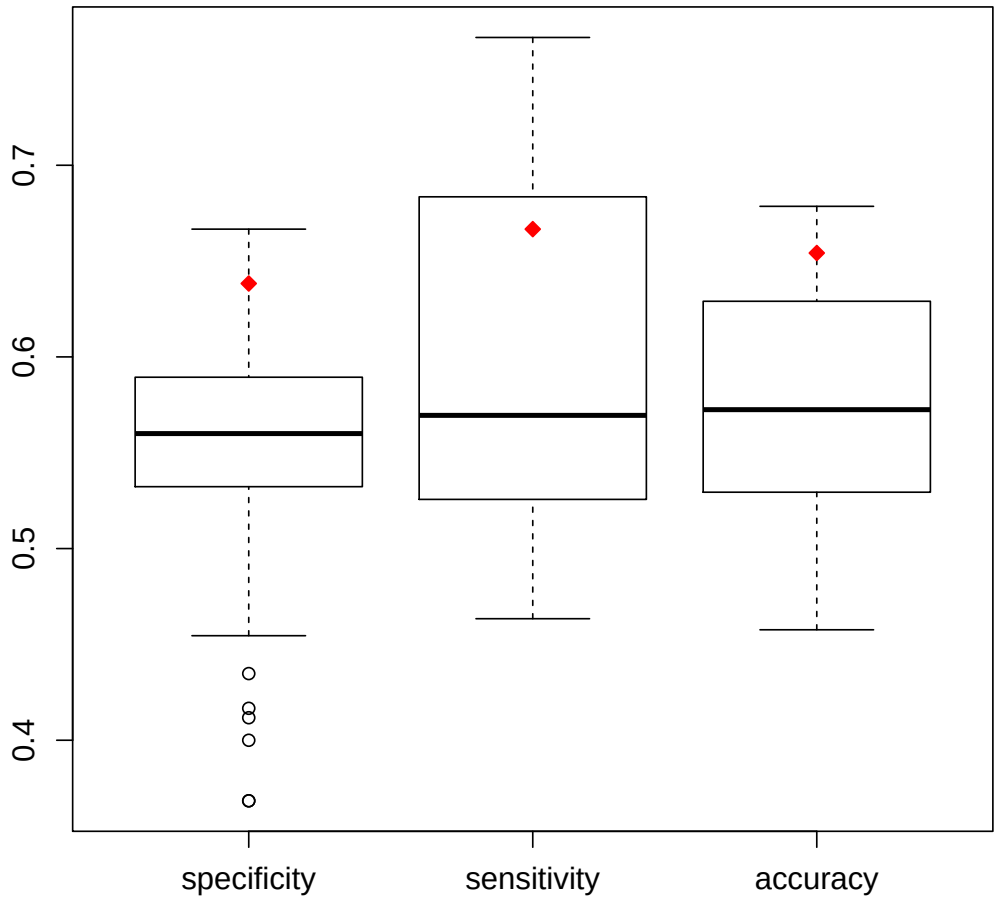
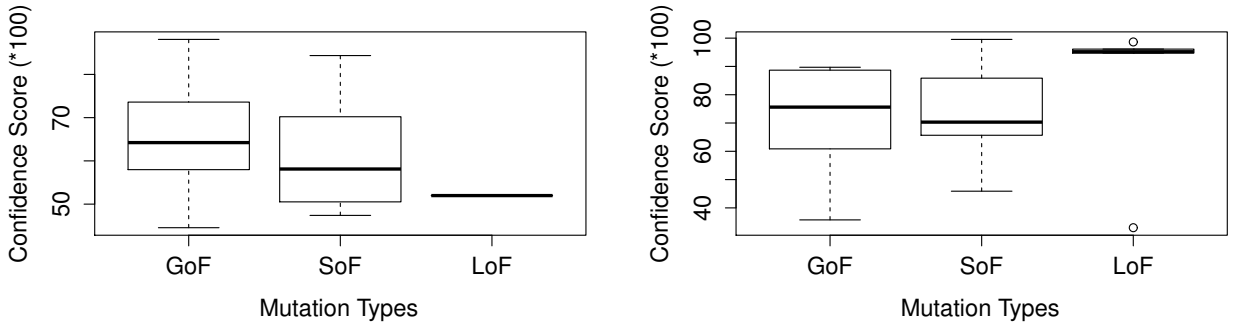


Figure 4.7: The performance of HMMvar-func based on K-means clustering or CEO clustering. (100 random initial guesses are evaluated for K-means clustering on the TSHR data set with $k = 4$ and $u = 2.7$. The red diamond points represent the corresponding performance of the CEO clustering.)

Table 4.3: Prediction of oncogenic mutations.

Gene	Total	GoF	SoF	LoF	CoF
EGFR	78	31	44	1	0
BRAF	46	13	27	5	0



(a) (b)

Figure 4.8: Confidence score distribution for different predicted mutation types: (a) confidence score for EGFR mutations, (b) confidence score for BRAF mutations.

EGFR is a cell surface protein that binds to epidermal growth factor. The BRAF gene makes a protein called B-RAF, which is used to transmit chemical signals from outside the cell into the cell's nucleus. Activating mutations in EGFR and BRAF are frequently found to be associated with cancer [22, 56, 68, 69]. Improper activation results in increased malignant cell survival, proliferation, invasion, and metastasis. Table 4.3 shows the total number of activating mutations evaluated and the corresponding number of predicted GoF, SoF, LoF, and CoF classifications for each gene. The predicted types are dominated by GoF and SoF mutations as expected, because the GoF and SoF mutations are both expected to have the protein acquiring new functions. The median confidence score for GoF is greater than that for SoF, which means the mutant gene is more likely to keep the original functions. Distribution details of the confidence scores for both genes are in Figure 4.8.

BRAF is commonly activated by somatic point mutations in human cancers, most frequently by mutations located within the kinase domain at amino acid positions G466, G469, L597, and V600. The most prevalent mutation is a missense mutation, which results in a replacement of valine at codon 600 (V600) with other amino acids and occurs in 90% of all BRAF mutations [3]. Figure 4.9 shows the target cluster C_0 along with other subfamilies. In the

last column of the MSA, where the substitution occurs, the amino acids are conserved within each subfamily but are different across subfamilies, which may indicate the valine acid at this residue could be easily replaced by the corresponding amino acids from other subfamilies. For example, the BRAF V600A mutation, caused by a single nucleotide change (c.1799T>C) that results in the replacement of valine at position 600 by alanine, has been shown to be a rare mutation in skin cancer [58].

The TP53 SNPs are classified into four classes as shown in Figure 4.10 in terms of the transactivity level. Three dotted lines ($y = 20$, $y = 75$, and $y = 140$) separate the plot vertically into four regions, which represent "nonfunctional", "partially functional", "functional", and "supertrans" regions, respectively, from bottom to top. In Figure 4.10, the median of the transactivity level in the GoF group is the highest among these four groups as the GoF group is enriched by "functional" or "supertrans" variants. In contrast, the LoF group is dominated by "partially functional" or "nonfunctional" variants. The medians of the transactivity level in the GoF and CoF groups are higher than those in the SoF and LoF groups, as 'loss of function' mutations inactivate tumor suppressor genes and the genes are likely losing the original functions as a result of LoF or SoF.

The 273rd codon of TP53 is one of the 'hot spots' for cancer related mutations. In [50], the authors concluded that the mutants of TP53 on the 273rd codon show growth modulation activities regardless of its specific transactivation. Specifically, the R273H mutation enhances cell growth in spite of keeping the TP53 specific transactivation activity, whereas the R273L mutation suppresses cell growth in spite of its complete loss of the TP53 specific transactivation. HMMvar-func predicts R273H to be a GoF mutation with confidence score of 0.49 and R273L to be an SoF mutation with confidence score of 0.71. Therefore, the HMMvar-func prediction of the functional outcome of these two mutations is indeed consistent with the findings in [50].

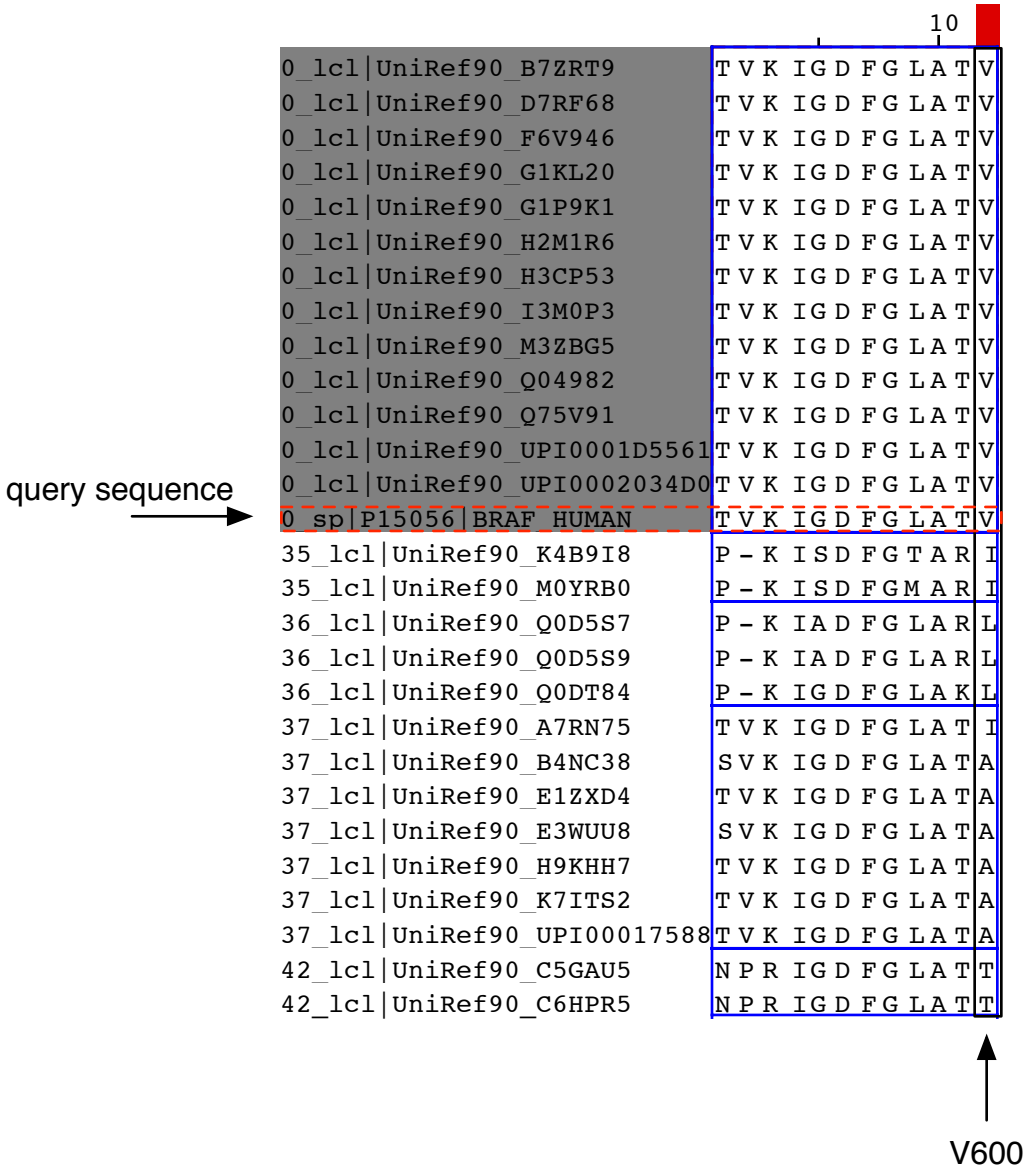


Figure 4.9: Several selected BRAF subfamilies from the clustering result. The last column is the position of the mutation. The wild type query sequence is in the target cluster C_0 and indicated in the dashed red box. Subfamilies are separated by blue borders.

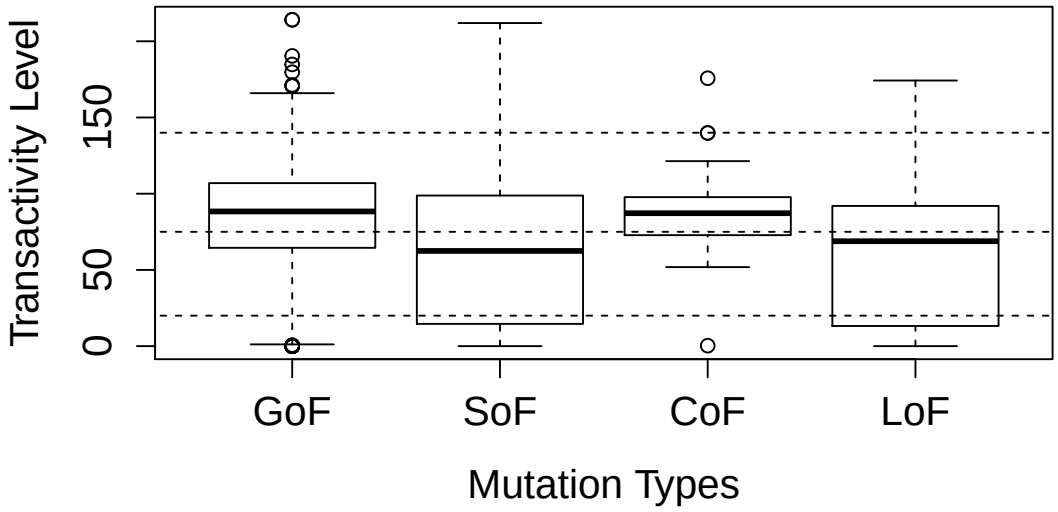


Figure 4.10: The transactivity level of the gene TP53 in different predicted mutation groups.

4.3.3 Predictions on SoF Mutations

The SoF mutations reported in [81] are tested. The R132H mutation in IDH1, shown experimentally [21] to lead to loss of the original function but gain of new function, essentially falls into the category of "SoF" defined in the current study, and is also investigated here. As shown in Table 4.5, three mutations (in PTPRD, MAP2K4, and CDH1) are predicted as switch of function with confidence score over 0.6. As an example, Figure 4.11 shows the tree generated by Jalview [104] from the processed alignment of homologous sequences of the MAP2K4 protein. The tree is built according to the average distance using BLOSUM62 and based on sum of scores for the residue pairs at each aligned position. The tree shows three clusters, C_{19}, C_{28}, and C_0, with C_0 being the target cluster. The minimum score S_i^x is calculated from C_{19}. According to the HMMvar scores, C_{19} and C_{28} are the potential subfamilies that the protein MAP2K4 might switch to due to the variation Q142L (not all the potential subfamilies are listed). Q142L, a missense mutation in MAP2K4, has been identified as one

of the major somatic mutations in human lung cancer samples [23]. However, it is predicted by the two commonly used programs SIFT [75] to be "tolerant" and PolyPhen [79] "benign", respectively. Our prediction result together with [81] suggests an alternative hypothesis for the functional impact of the variant, that is it leads to "SoF" in MAP2K4, which seems to be more likely considering its common occurrence in lung cancer samples [23].

Similarly, the two mutations in PTPRD and CDH1 are likely to lead to SoF with high probabilities. PTPRD has been found to be somatically mutated in colorectal carcinoma with R28Q mutation [90]. H233Q in CDH1 was found to be associated with breast cancer [43].

The prediction for A95E in RAC1 gene is SoF. However, the confidence score is only slightly greater than 0.5, because the probability L_i^0 (Figure 4.4) of losing the original functions is low (0.55) whereas the probability A_i^x of acquiring new functions is high (0.997), making a SoF classification unreliable. Previous studies are more agreed on GoF. As discussed before, the cutoff u is an important factor in determining the final prediction. We examined the cutoff and found that, if $u = 3.0$ instead of 2.7, A95E is predicted as GoF with confidence score 0.524. A similar result is obtained for the R132H mutation in IDH1, which is predicted as GoF with low confidence score ($L_i^0 = 0.40$, $A_i^x = 0.89$). We assume the independence of losing the original functions and gaining new functions. As a result, for those predictions with low confidence scores, the users should be aware of the probability of losing the original functions (L_i^0) and the probability of acquiring new functions (A_i^x).

Table 4.4: SoF Mutations

Gene	Variant	Predicted Type	Confidence Score
RAC1	A95E	SoF	0.548
PTPRD	R28Q	SoF	0.728
MAP2K4	Q142L	SoF	0.800
CDH1	H233Q	SoF	0.651
IDH1	R132H	GoF	0.533

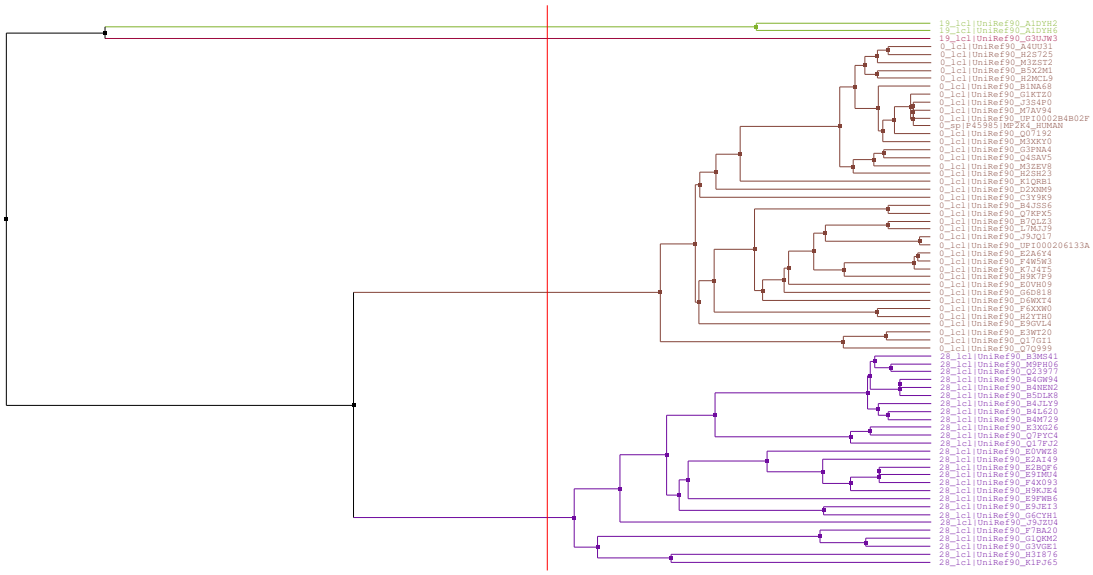


Figure 4.11: Distance tree of the MAP2K subfamilies. Colors indicate different subfamilies. The minimum score S_i^x is calculated from C_{19}. C_0 is the target cluster. C_{28} is an example subfamily that the mutant protein could switch to. The leaves are protein sequences. Two sequences are merged according to the BLOSUM62 matrix by averaging the substitution distance over all the positions in the MSA.

4.3.4 Mutagenesis Analysis

The SWISS-PROT mutagenesis data obtained from [57] consists of 6922 variants from 4061 proteins, including 225 activating, 1222 neutral, and 5475 deleterious variants (labeled in the data set).

Figure 4.12 displays the prediction results. The deleterious variants contribute the most to the predicted LoF group and this contribution decreases as it goes from LoF to SoF to neutral to GoF, while the activating variants contribute the least to the predicted LoF group and this contribution increases as it goes from LoF to SoF to neutral to GoF. The neutral variants have the same trend as the activating variants. Although only 3.3% of the data set is activating, 8.7% of the mutations in the gain of function group are activating. Compared to other groups, the activating variants are enriched in the gain of function group, and the

deleterious variants are enriched in the LoF group.

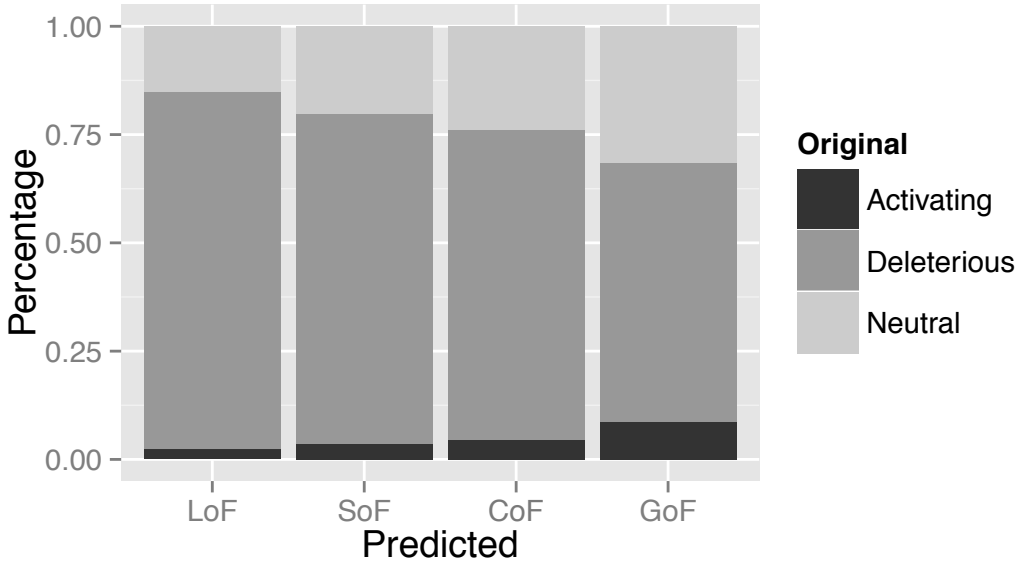


Figure 4.12: Predictions based on SWISS-PROT mutagenesis data set.

4.3.5 The Conversion Between HPES and TEAS

Studies have shown that two closely related proteins (75% amino acid identity), henbane premnaspirodiene synthase (HPES) and 5-epi-aristolochene synthase (TEAS), can convert to each other by nine amino acid mutations [41, 70], shown in Table 4.5. HMMvar-func is used to predict these mutations. First, the single scores of these mutations calculated from the entire family (all the homologous sequences) are very low (Table 4.5), suggesting that these mutations are neutral. Second, these mutations are all predicted as GoF with high confidence scores (over 90%). Finally, the alternative alleles are dominant in the target subfamily (Figure 4.13, the blue shaded columns), further indicating the conversion between TEAS and HPES through these mutations.

We explored the factors that influence the performance of HMMvar-func. Two clustering

Table 4.5: The mutations that convert TEAS to HPES

	Variant	Position	HMMvar Score	Predicted Type	Confidence Score
1	A/T	274	-1.3	GoF	0.952
2	V/A	291	-1.1	GoF	0.961
3	V/I	372	-1.1	GoF	0.941
4	T/S	402	-1.2	GoF	0.957
5	Y/L	406	-0.8	GoF	0.966
6	S/N	436	-0.1	GoF	0.947
7	I/T	438	-0.1	GoF	0.936
8	I/L	439	1.1	GoF	0.929
9	V/I	516	-0.6	GoF	0.703

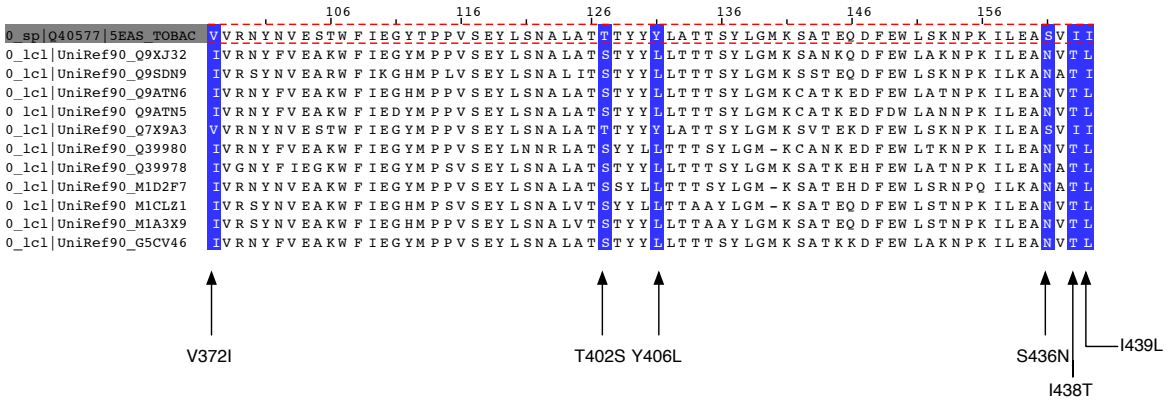


Figure 4.13: The target cluster of TEAS. The blue shaded columns are the mutation positions and the target sequence is indicated by the red dashed box.

algorithms, the K-means clustering and the CEO algorithm, were compared. The cutoff u was determined by the prediction performance in the TSHr data set. The quality of the MSA is an important factor that may affect the final prediction. To ensure high quality, the MSA processing step removes redundant sequences that are above an alignment similarity threshold and the columns that have low alignment quality. Finally, the proper region is

selected by left and right extension from the position of the variant. Prediction of the fitness impact of variants, such as deleterious or neutral, is important, but computationally predicting the fine grained functional outcome is equally crucial, especially in cancer studies. The fine grained predictions can be used to identity mutations that may play an important role in the resistance to certain therapeutic agents.

Chapter 5

Conclusion

Functional prediction of variants using conservation based methods has been done extensively in the literature. An HMM integrates conservation information into a rigorous probabilistic framework. In this dissertation, based on HMMs, a suite of programs, HMMvar, HMMvar-multi, and HMMvar-func, was developed to predict the functional effect of both SNPs and short indels, the joint effects of multiple variants, and the fine grained functional outcome of SNPs, respectively.

Although the dissertation addressed the three limitations raised in Chapter 1, the suite of tools have a major caveat: they are only applicable to coding variants. As the majority of variations identified in the 1000 Genomes Project reside in noncoding regions, future work needs to focus on predicting the effect of noncoding variants.

To this end, I conjecture a pipeline that can predict the functional effects of different types of mutations in noncoding regions (Figure 5.1). The pipeline includes these components: (1) disease set curation, (2) control set curation, (3) feature set curation, and (4) training and classification. Disease set curation will require collecting disease causing mutations and/or cancer driver mutations from various databases. Control set curation can be done through simulation and/or collecting the variants that have minor allele frequency greater than 1%

(MAF> 1%) in the 1000 Genomes data. Feature set curation will involve extensive collection of the annotation data from various programs and databases. Finally, different classifiers can be explored to train and classify/predict the effect of noncoding variants. Generally, the main concern of a trained classifier is feature set curation. In this pipeline, a feature learning procedure can be developed using deep learning technologies to automatically collect and update the features.

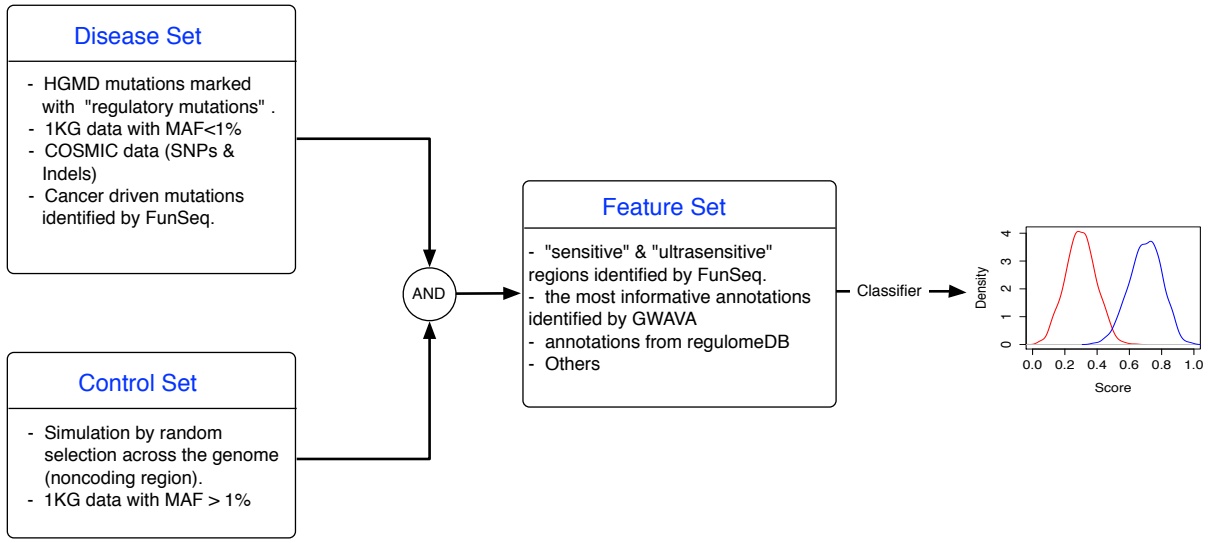


Figure 5.1: A hypothetical system for trained classifier based method for predicting the functional effect of variants in noncoding regions (HGMD: Human Gene Mutation Database; 1KG: 1000 Genomes Project; MAF: minor allele frequency; FunSeq: [52]; GWAVA: [82]. The red curve may represent deleterious mutations and the blue curve neutral).

It is challenging to interpret the molecular mechanisms of disease-related variants in noncoding regions, given the diverse functions of noncoding regions, incomplete annotation of regulatory elements, and the potential existence of unknown regulatory control mechanisms. However, scientists are making efforts to overcome the obstacles and build reliable pipelines to discover the consequences of variants in noncoding regions.

Bibliography

[1] I. A. Adzhubei, S. Schmidt, L. Peshkin, V. E. Ramensky, A. Gerasimova, P. Bork, A. S. Kondrashov, and S. R. Sunyaev. A method and server for predicting damaging missense mutations. *Nature Methods*, 7:248–249, 2010.

[2] S. F. Altschul, T. L. Madden, A. A. Schäffer, J. Zhang, Z. Zhang, W. Miller, and D. J. Lipman. Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. *Nucleic Acids Research*, 25:3389–3420, 1997.

[3] P. A. Ascierto, J. M. Kirkwood, J. J. Grob, E. Simeone, A. M. Grimaldi, M. Maio, G. Palmieri, A. Testori, F. M. Marincola, and N.l Mozzillo. The role of V600 mutation in melanoma. *Journal of Translational Medicine*, 10:85, 2012.

[4] S. Asthana, M. Roytberg, and J. Stamatoyannopoulos. Analysis of sequence conservation at nucleotide resolution. *PLOS Computational Biology*, 3:e254, 2007.

[5] A. Bairoch and R. Apweiler. The SWISS-PROT protein sequence data bank and its supplement TrEMBL. *Nucleic Acids Research*, 25:31–36, 1997.

[6] C. Barrett, R. Hughey, and K. Karplus. Scoring hidden Markov models. *Computer Application in the Biosciences*, 13:191–199, 1997.

[7] T. Barrett, S. E. Wilhite, P. Ledoux, C. Evangelista, I. F. Kim, M. Tomashevsky, K. A. Marshall, K. H. Phillippy, P. M. Sherman, M. Holko, A. Yefanov, H. Lee, N. Zhang,

C. L. Robertson, N. Serova, S. Davis, and A. Soboleva. NCBI GEO: Archive for functional genomics data sets–update. *Nucleic Acids Research*, 41:D991–995, 2013.

[8] A. P. Boyle, E. L. Hong, M. Hariharan, Y. Cheng, M. A. Schaub, M. Kasowski, K. J. Karczewski, J. Park, B. C. Hitz, S. Weng, J. M. Cherry, and M. Snyder. Annotation of functional variation in personal genomes using RegulomeDB. *Genome Research*, 22(9):1790–1797, 2012.

[9] H. Carter, S. Chen, L. Isik, S. Tyekucheva, V. E. Velculescu, K. W. Kinzler, B. Vogelstein, and R. Karchin. Cancer-specific high-throughput annotation of somatic mutations: Computational prediction of driver missense mutations. *Cancer Research*, 69:6660–6667, 2009.

[10] H. Carter, C. Douville, and P. D. Stenson. Identifying mendelian disease genes with the variant effect scoring tool. *BMC Genomics*, 14:S3, 2013.

[11] W. C. Chang, Y. Y. Fang, H. W. Chang, L. Y. Chuang, Y. D. Lin, M. F. Hou, and C. H. Yang. Identifying association model for single-nucleotide polymorphisms of ORAI1 gene for breast cancer. *Cancer Cell International*, 14:29, 2014.

[12] C. H. Chen, T. J. Chuang, B. Y. Liao, and F. C. Chen. Scanning for the signatures of positive selection for human-specific insertions and deletions. *Genome Biology and Evolution*, 1:415–419, 2009.

[13] F. C. Chen, C. J. Chen, W. H. Li, and T. J. Chuang. Human-specific insertions and deletions inferred from mammalian genome sequences. *Genome Research*, 17(1):16–22, 2007.

[14] Y. Choi, G. Sims, and S. Murphy. Predicting the functional effect of amino acid substitutions and indels. *PLOS ONE*, 7:e46688, 2012.

[15] International Human Genome Sequencing Consortium. Finishing the euchromatic sequence of the human genome. *Nature*, 431:931–945, 2004.

[16] The ENCODE Project Consortium. An integrated encyclopedia of DNA elements in the human genome. *Nature*, 489:57–74, 2012.

[17] The UniProt Consortium. Update on activities at the universal protein resource (UniProt) in 2013. *Nucleic Acids Research*, 41:D43–D47, 2013.

[18] The Uniprot Consortium. Activities at the universal protein resource (UniProt). *Nucleic Acids Research*, 42:D191–D198, 2014.

[19] G. Cooper and J. Shendure. Needles in stacks of needles: Finding disease-causal variants in a wealth of genomic data. *Nature Reviews Genetics*, 12:628–640, 2011.

[20] G. Cooper, E. Stone, and G. Asimenon. Distribution and intensity of constraint in mammalian genomic sequence. *Genome Research*, 15:901–913, 2005.

[21] L. Dang, D. W. White, S. Gross, B. D. Bennett, M. A. Bittinger, E. M. Driggers, V. R. Fantin, H. G. Jang, S. Jin, M. C. Keenan, K. M. Marks, R. M. Prins, P. S. Ward, K. E. Yen, L. M. Liau, J. D. Rabinowitz, L. C. Cantley, C. B. Thompson, M. G. Vander Heiden, and S. M. Su. Cancer-associated IDH1 mutations produce 2-hydroxyglutarate. *Nature*, 462:739–744, 2009.

[22] H. Davies, G. R. Bignell, C. Cox, P. Stephens, S. Edkins, S. Clegg, J. Teague, H. Woffendin, M. J. Garnett, W. Bottomley, N. Davis, E. Dicks, R. Ewing, Y. Floyd, K. Gray, S. Hall, R. Hawes, J. Hughes, V. Kosmidou, A. Menzies, C. Mould, A. Parker, C. Stevens, S. Watt, S. Hooper, R. Wilson, H. Jayatilake, B. A. Gusterson, C. Cooper, J. Shipley, D. Hargrave, K. Pritchard-Jones, N. Maitland, G. Chenevix-Trench, G. J. Riggins, D. D. Bigner, G. Palmieri, A. Cossu, A. Flanagan, A. Nicholson, J. W. Ho, S. Y. Leung, S. T. Yuen, B. L. Weber, H. F. Seigler, T. L. Darrow, H. Paterson, R. Marais, C. J. Marshall, R. Wooster, M. R. Stratton, and P. A. Futreal. Mutations of the BRAF gene in human cancer. *Nature*, 417:949–954, 2002.

[23] H. Davies, C. Hunter, R. Smith, P. Stephens, C. Greenman, G. Bignell, J. Teague, A. Butler, S. Edkins, C. Stevens, A. Parker, S. O'Meara, T. Avis, S. Barthorpe,

L. Brackenbury, G. Buck, J. Clements, J. Cole, E. Dicks, K. Edwards, S. Forbes, M. Gorton, K. Gray, K. Halliday, R. Harrison, K. Hills, J. Hinton, D. Jones, V. Kosmidou, R. Laman, R. Lugg, A. Menzies, J. Perry, R. Petty, K. Raine, R. Shepherd, A. Small, H. Solomon, Y. Stephens, C. Tofts, J. Varian, A. Webb, S. West, S. Widaa, A. Yates, F. Brasseur, C. S. Cooper, A. M. Flanagan, A. Green, M. Knowles, S. Y. Leung, L. H. Looijenga, B. Malkowicz, M. A. Pierotti, B. T. Teh, S. T. Yuen, S. R. Lakhani, D. F. Easton, B. L. Weber, P. Goldstraw, A. G. Nicholson, R. Wooster, M. R. Stratton, and P. A. Futreal. Somatic mutations of the protein kinase gene family in human lung cancer. *Cancer Research*, 65(17):7591–7595, 2005.

[24] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39(1):1–38, 1977.

[25] C. Douville, H. Carter, R. Kim, N. Niknafs, M. Diekhans, P. D. Stenson, D. N. Cooper, M. Ryan, and R. Karchin. CRAVAT: Cancer-related analysis of variants toolkit. *Bioinformatics*, 29(5):647–648, 2013.

[26] L. Duprez, J. Parma, J. V. Sande, P Rodien, J. E. Dumont, G Vassart, and M Abramowicz. TSH receptor mutations and thyroid disease. *Trends in Endocrinology and Metabolism*, 9(4):133–140, 1998.

[27] R Durbin, S. R. Eddy, A. Krogh, and G Mitchison. Profile HMMs for sequence families. In *Biological Sequence Analysis*, pages 101–133. Cambridge University Press, 1998.

[28] S. Eddy. Profile hidden Markov models. *Bioinformatics*, 14(9):755–763, 1998.

[29] R. C. Edgar. MUSCLE: Multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Research*, 32:1792–1797, 2004.

[30] R. C. Emma, J. O. John, and M. S. Orla. $BRAF^{V600E}$: Implication for carcinogenesis and molecular therapy. *Molecular Cancer Therapeutics*, 10:385, 2011.

[31] Four way Enredo-Pecan-Ortheus (EPO) multiple alignments. `ftp://ftp.ensembl.org/pub/release-54/emf/ensembl-compara/epo_4_catarrhini/README.epo_4_way`.

[32] R. Finn, J. Clements, and S. Eddy. HMMER web server: Interactive sequence similarity searching. *Nucleic Acids Research*, 39:W29–W37, 2011.

[33] P. Flicek, M. R. Amode, D. Barrell, K. Beal, K. Billis, S. Brent, D. Carvalho-Silva, P. Clapham, G. Coates, S. Fitzgerald, L. Gil, C. G. Giron, L. Gordon, T. Hourlier, S. Hunt, N. Johnson, T. Juettemann, A. K. Kahari, S. Keenan, E. Kulesha, F. J. Martin, T. Maurel, W. M. McLaren, D. N. Murphy, R. Nag, B. Overduin, M. Pignatelli, B. Pritchard, E. Pritchard, H. S. Riat, M. Ruffier, D. Sheppard, K. Taylor, A. Thormann, S. J. Trevanion, A. Vullo, S. P. Wilder, M. Wilson, A. Zadissa, B. L. Aken, E. Birney, F. Cunningham, J. Harrow, J. Herrero, T. J. Hubbard, R. Kinsella, M. Muffato, A. Parker, G. Spudich, A. Yates, D. R. Zerbino, and S. M. Searle. Ensembl 2014. *Nucleic Acids Research*, 42:D749–D755, 2014.

[34] S. A. Forbes, N. Bindal, S. Bamford, C. Cole, C. Y. Kok, D. Beare, M. Jia, R. Shepherd, K. Leung, A. Menzies, J. W. Teague, P. J. Campbell, M. R. Stratton, and P. A. Futreal. Cosmic: Mining complete cancer genomes in the catalogue of somatic mutations in cancer. *Nucleic Acids Research*, 39 (suppl 1):D945–D950, 2011.

[35] G. D. Forney. The Viterbi algorithm. In *Porceedings of the IEEE*, pages 268–278. IEEE, 1973.

[36] K. J. Friedman, W. E. Highsmith, and L. M. Silverman. Detecting multiple cystic fibrosis mutations by polymerase chain reaction-mediated site- directed mutagenesis. *Clinical Chemistry*, 37(5):753–755, 1991.

[37] E. Gonzalez-Ortega, E. Ballana, R. Badia, B. Clotet, and J. A. Este. Compensatory mutations rescue the virus replicative capacity of VIRIP-resistant HIV-1. *Antiviral Research*, 92(3):479–483, 2011.

[38] A. Gonzalez-Perez, J. Deu-Pons, and N. Lopez-Bigas. Improving the prediction of the functional impact of cancer mutations by baseline tolerance transformation. *Genome Medicina*, 4:89, 2012.

[39] A. Gonzalez-Perez, C. Perez-Llamas, J. Deu-Pons, D. Tamborero, M. P. Schroeder, A. Jene-Sanz, A. Santos, and N. Lopez-Bigas. IntOGen-mutations identifies cancer drivers across tumor types. *Nature Methods*, 10:1081–1082, 2013.

[40] J. Gough, K. Karplus, R. Hughey, and C. Chothia. Assignment of homology to genome sequences using a library of hidden Markov models that represent all proteins of known structure. *Journal of Molecular Biology*, 313:903–919, 2001.

[41] B. T. Greenhagen, P. E. O'Maille, J. P. Noel, and J. Chappell. Identifying and manipulating structural determinates linking catalytic specificities in terpene synthases. *Proceedings of the National Academy of Sciences of the United States of America*, 103(26):9826–9831, 2006.

[42] L. A. Hindorff, P. Sethupathy, H. A. Junkins, E. M. Ramos, J. P. Mehta, F. S. Collins, and T. A. Manolio. Potential etiologic and functional implications of genome-wide association loci for human diseases and traits. *Proceedings of the National Academy of Sciences of the United States of America*, 106(23):9362–7, 2009.

[43] A. Hollestelle, J. H. Nagel, M. Smid, S. Lam, F. Elstrodt, M. Wasielewski, S. S. Ng, P. J. French, J. K. Peeters, M. J. Rozendaal, M. Riaz, D. G. Koopman, T. L. Ten Hagen, B. H. de Leeuw, E. C. Zwarthoff, A. Teunisse, P. J. van der Spek, J. G. Klijn, W. N. Dinjens, S. P. Ethier, H. Clevers, A. G. Jochemsen, M. A. den Bakker, J. A. Foekens, J. W. Martens, and M. Schutte. Distinct gene mutation profile among luminal-type and basal-type breast cancer cell lines. *Breast Cancer Research and Treatment*, 121(1):53–64, 2010.

[44] F. Hormozdiari, R. Salari, M. Hsing, A. Schönhuth, S. K. Chan, S. C. Sahinalp, and A. Cherkasov. The effect of insertions and deletions on wirings in protein-protein inter-

action networks: A large-scale study. *Journal of Computational Biology*, 16(2):159–167, 2009.

[45] J. Hu and C. Pauline. Predicting the effects of frame shifting indels. *Genome Biology*, 13:2, 2012.

[46] International Cancer Genome Consortium. `http://dcc.icgc.org/`, 2012.

[47] H. A. James and A. K. Gupta. Mixtures of dirichlet distributions and estimation in contingency tables. *The Annals of Statistics*, 10:1261–1268, 1982.

[48] T. Kanungo, D. M. Mount, Nathan S. N., C. Piatko, R. Silverman, and A. Y. Wu. An efficient k-means clustering algorithm: Analysis and implementation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24:881–892, 2002.

[49] S. Kato, S. Han, and W. Liu. Understanding the function-structure and function-mutation relationships of p53 tumor suppressor protein by high-resolution missense mutation analysis. *Proceedings of the National Academy of Sciences of the United States of America*, 100:8424–8429, 2003.

[50] M. Kawamura, T. Yamashita, K. Segawa, M. Kaneuchi, M. Shindoh, and K. Fujinaga. The 273rd codon mutants of p53 show growth modulation activities not correlated with p53-specific transactivation activity. *Oncogene*, 12(11):2361–2367, 1996.

[51] M. Kelly and C. Seminarian. Multiple mutations in genetic cardiovascular disease a marker of disease severity? *Circulation: Cardiovascular Genetics*, 2:182–190, 2009.

[52] E. Khurana, Y. Fu, V. Colonna, X. J. Mu, H. M. Kang, T. Lappalainen, A. Sboner, L. Lochovsky, J. Chen, A. Harmanci, J. Das, A. Abyzov, S. Balasubramanian, K. Beal, D. Chakravarty, D. Challis, Y. Chen, D. Clarke, L. Clarke, F. Cunningham, U. S. Evani, P. Flicek, R. Fragoza, E. Garrison, R. Gibbs, Z. H. Gumus, J. Herrero, N. Kitabayashi, Y. Kong, K. Lage, V. Liluashvili, S. M. Lipkin, D. G. MacArthur, G. Marth, D. Muzny, T. H. Pers, G. R. Ritchie, J. A. Rosenfeld, C. Sisu, X. Wei, M. Wilson, Y. Xue, F. Yu,

E. T. Dermitzakis, H. Yu, M. A. Rubin, C. Tyler-Smith, M. Gerstein, D. M. Altshuler, R. M. Durbin, G. R. Abecasis, D. R. Bentley, A. Chakravarti, A. G. Clark, P. Donnelly, E. E. Eichler, P. Flicek, S. B. Gabriel, R. A. Gibbs, E. D. Green, M. E. Hurles, B. M. Knoppers, J. O. Korbel, E. S. Lander, C. Lee, H. Lehrach, E. R. Mardis, G. T. Marth, G. A. McVean, D. A. Nickerson, J. P. Schmidt, S. T. Sherry, J. Wang, R. K. Wilson, R. A. Gibbs, H. Dinh, C. Kovar, S. Lee, L. Lewis, D. Muzny, J. Reid, M. Wang, J. Wang, X. Fang, X. Guo, M. Jian, H. Jiang, X. Jin, G. Li, J. Li, Y. Li, Z. Li, X. Liu, Y. Lu, X. Ma, Z. Su, S. Tai, M. Tang, B. Wang, G. Wang, H. Wu, R. Wu, Y. Yin, W. Zhang, J. Zhao, M. Zhao, X. Zheng, Y. Zhou, E. S. Lander, D. M. Altshuler, S. B. Gabriel, N. Gupta, P. Flicek, L. Clarke, R. Leinonen, R. E. Smith, X. Zheng-Bradley, D. R. Bentley, R. Grocock, S. Humphray, T. James, Z. Kingsbury, H. Lehrach, R. Sudbrak, M. W. Albrecht, V. S. Amstislavskiy, T. A. Borodina, M. Lienhard, F. Mertes, M. Sultan, B. Timmermann, M. L. Yaspo, S. T. Sherry, G. A. McVean, E. R. Mardis, R. K. Wilson, L. Fulton, R. Fulton, G. M. Weinstock, R. M. Durbin, S. Balasubramaniam, J. Burton, P. Danecek, T. M. Keane, A. Kolb-Kokocinski, S. McCarthy, J. Stalker, M. Quail, J. P. Schmidt, C. J. Davies, J. Gollub, T. Webster, B. Wong, Y. Zhan, A. Auton, R. A. Gibbs, F. Yu, M. Bainbridge, D. Challis, U. S. Evani, J. Lu, D. Muzny, U. Nagaswamy, J. Reid, A. Sabo, Y. Wang, J. Yu, J. Wang, L. J. Coin, L. Fang, X. Guo, X. Jin, G. Li, Q. Li, Y. Li, Z. Li, H. Lin, B. Liu, R. Luo, N. Qin, H. Shao, B. Wang, Y. Xie, C. Ye, C. Yu, F. Zhang, H. Zheng, H. Zhu, G. T. Marth, E. P. Garrison, D. Kural, W. P. Lee, W. F. Leong, A. N. Ward, J. Wu, M. Zhang, C. Lee, L. Griffin, C. H. Hsieh, R. E. Mills, X. Shi, M. von Grotthuss, C. Zhang, M. J. Daly, M. A. DePristo, D. M. Altshuler, E. Banks, G. Bhatia, M. O. Carneiro, G. del Angel, S. B. Gabriel, G. Genovese, N. Gupta, R. E. Handsaker, C. Hartl, E. S. Lander, S. A. McCarroll, J. C. Nemesh, R. E. Poplin, S. F. Schaffner, K. Shakir, S. C. Yoon, J. Lihm, V. Makarov, H. Jin, W. Kim, K. C. Kim, J. O. Korbel, T. Rausch, P. Flicek, K. Beal, L. Clarke, F. Cunningham, J. Herrero, W. M. McLaren, G. R. Ritchie, R. E. Smith, X. Zheng-Bradley, A. G. Clark, S. Gottipati, A. Keinan, J. L. Rodriguez-Flores, P. C.

Sabeti, S. R. Grossman, S. Tabrizi, R. Tariyal, D. N. Cooper, E. V. Ball, P. D. Stenson, D. R. Bentley, B. Barnes, M. Bauer, R. Cheetham, T. Cox, M. Eberle, S. Humphray, S. Kahn, L. Murray, J. Peden, R. Shaw, K. Ye, M. A. Batzer, M. K. Konkel, J. A. Walker, D. G. MacArthur, M. Lek, R. Sudbrak, V. S. Amstislavskiy, R. Herwig, M. D. Shriver, C. D. Bustamante, J. K. Byrnes, M. De La Vega, S. Gravel, E. E. Kenny, J. M. Kidd, P. Lacroute, B. K. Maples, A. Moreno-Estrada, F. Zakharia, E. Halperin, Y. Baran, D. W. Craig, A. Christoforides, N. Homer, T. Izatt, A. A. Kurdoglu, S. A. Sinari, K. Squire, S. T. Sherry, C. Xiao, J. Sebat, V. Bafna, K. Ye, E. G. Burchard, R. D. Hernandez, C. R. Gignoux, D. Haussler, S. J. Katzman, W. Kent, B. Howie, A. Ruiz-Linares, E. T. Dermitzakis, T. Lappalainen, S. E. Devine, X. Liu, A. Maroo, L. J. Tallon, J. A. Rosenfeld, L. P. Michelson, G. R. Abecasis, H. M. Kang, P. Anderson, A. Angius, A. Bigham, T. Blackwell, F. Busonero, F. Cucca, C. Fuchsberger, C. Jones, G. Jun, Y. Li, R. Lyons, A. Maschio, E. Porcu, F. Reinier, S. Sanna, D. Schlessinger, C. Sidore, A. Tan, M. K. Trost, P. Awadalla, A. Hodgkinson, G. Lunter, G. A. McVean, J. L. Marchini, S. Myers, C. Churchhouse, O. Delaneau, A. Gupta-Hinch, Z. Iqbal, I. Mathieson, A. Rimmer, D. K. Xifara, T. K. Oleksyk, Y. Fu, X. Liu, M. Xiong, L. Jorde, D. Witherspoon, J. Xing, E. E. Eichler, B. L. Browning, C. Alkan, I. Hajirasouliha, F. Hormozdiari, A. Ko, P. H. Sudmant, E. R. Mardis, K. Chen, A. Chinwalla, L. Ding, D. Dooling, D. C. Koboldt, M. D. McLellan, J. W. Wallis, M. C. Wendl, Q. Zhang, R. M. Durbin, M. E. Hurles, C. Tyler-Smith, C. A. Albers, Q. Ayub, S. Balasubramaniam, Y. Chen, A. J. Coffey, V. Colonna, P. Danecek, N. Huang, L. Jostins, T. M. Keane, H. Li, S. McCarthy, A. Scally, J. Stalker, K. Walter, Y. Xue, Y. Zhang, M. B. Gerstein, A. Abyzov, S. Balasubramanian, J. Chen, D. Clarke, Y. Fu, L. Habegger, A. O. Harmanci, M. Jin, E. Khurana, X. J. Mu, C. Sisu, Y. Li, R. Luo, H. Zhu, C. Lee, L. Griffin, C. H. Hsieh, R. E. Mills, X. Shi, M. von Grotthuss, C. Zhang, G. T. Marth, E. P. Garrison, D. Kural, W. P. Lee, A. N. Ward, J. Wu, M. Zhang, S. A. McCarroll, D. M. Altshuler, E. Banks, G. del Angel, G. Genovese, R. E. Handsaker, C. Hartl, J. C. Nemesh, K. Shakir, S. C. Yoon, J. Lihm,

V. Makarov, J. Degenhardt, P. Flicek, L. Clarke, R. E. Smith, X. Zheng-Bradley, J. O. Korbel, T. Rausch, A. M. Stutz, D. R. Bentley, B. Barnes, R. Cheetham, M. Eberle, S. Humphray, S. Kahn, L. Murray, R. Shaw, K. Ye, M. A. Batzer, M. K. Konkel, J. A. Walker, P. Lacroute, D. W. Craig, N. Homer, D. Church, C. Xiao, J. Sebat, V. Bafna, J. J. Michaelson, K. Ye, S. E. Devine, X. Liu, A. Maroo, L. J. Tallon, G. Lunter, G. A. McVean, Z. Iqbal, D. Witherspoon, J. Xing, E. E. Eichler, C. Alkan, I. Hajirasouliha, F. Hormozdiari, A. Ko, P. H. Sudmant, K. Chen, A. Chinwalla, L. Ding, M. D. McLellan, J. W. Wallis, M. E. Hurles, B. Blackburne, H. Li, S. J. Lindsay, Z. Ning, A. Scally, K. Walter, Y. Zhang, M. B. Gerstein, A. Abyzov, J. Chen, D. Clarke, E. Khurana, X. J. Mu, C. Sisu, R. A. Gibbs, F. Yu, M. Bainbridge, D. Challis, U. S. Evani, C. Kovar, L. Lewis, J. Lu, D. Muzny, U. Nagaswamy, J. Reid, A. Sabo, J. Yu, X. Guo, Y. Li, R. Wu, G. T. Marth, E. P. Garrison, W. F. Leong, A. N. Ward, G. del Angel, M. A. DePristo, S. B. Gabriel, N. Gupta, C. Hartl, R. E. Poplin, A. G. Clark, J. L. Rodriguez-Flores, P. Flicek, L. Clarke, R. E. Smith, X. Zheng-Bradley, D. G. MacArthur, C. D. Bustamante, S. Gravel, D. W. Craig, A. Christoforides, N. Homer, T. Izatt, S. T. Sherry, C. Xiao, E. T. Dermitzakis, G. R. Abecasis, H. M. Kang, G. A. McVean, E. R. Mardis, D. Dooling, L. Fulton, R. Fulton, D. C. Koboldt, R. M. Durbin, S. Balasubramaniam, T. M. Keane, S. McCarthy, J. Stalker, M. B. Gerstein, S. Balasubramanian, L. Habegger, E. P. Garrison, R. A. Gibbs, M. Bainbridge, D. Muzny, F. Yu, J. Yu, G. del Angel, R. E. Handsaker, V. Makarov, J. L. Rodriguez-Flores, H. Jin, W. Kim, K. C. Kim, P. Flicek, K. Beal, L. Clarke, F. Cunningham, J. Herrero, W. M. McLaren, G. R. Ritchie, X. Zheng-Bradley, S. Tabrizi, D. G. MacArthur, M. Lek, C. D. Bustamante, F. M. De La Vega, D. W. Craig, A. A. Kurdoglu, T. Lappalainen, J. A. Rosenfeld, L. P. Michelson, P. Awadalla, A. Hodgkinson, G. A. McVean, K. Chen, C. Tyler-Smith, Y. Chen, V. Colonna, A. Frankish, J. Harrow, Y. Xue, M. B. Gerstein, A. Abyzov, S. Balasubramanian, J. Chen, D. Clarke, Y. Fu, A. O. Harmanci, M. Jin, E. Khurana, X. J. Mu, C. Sisu, R. A. Gibbs, C. Kovar, D. Kalra, W. Hale, G. Fowler, D. Muzny, J. Reid, J. Wang, X. Guo, G. Li, Y. Li, X. Zheng, D. M. Alt-

shuler, P. Flicek, L. Clarke, J. Barker, G. Kelman, E. Kulesha, R. Leinonen, W. M. McLaren, R. Radhakrishnan, A. Roa, D. Smirnov, R. E. Smith, I. Streeter, I. Toneva, B. Vaughan, X. Zheng-Bradley, D. R. Bentley, T. Cox, S. Humphray, S. Kahn, R. Sudbrak, M. W. Albrecht, M. Lienhard, D. W. Craig, T. Izatt, A. A. Kurdoglu, S. T. Sherry, V. Ananiev, Z. Belaia, D. Beloslyudtsev, N. Bouk, C. Chen, D. Church, R. Cohen, C. Cook, J. Garner, T. Hefferon, M. Kimelman, C. Liu, J. Lopez, P. Meric, C. O'Sullivan, Y. Ostapchuk, L. Phan, S. Ponomarov, V. Schneider, E. Shekhtman, K. Sirotkin, D. Slotta, C. Xiao, H. Zhang, D. Haussler, G. R. Abecasis, G. A. McVean, C. Alkan, A. Ko, D. Dooling, R. M. Durbin, S. Balasubramaniam, T. M. Keane, S. McCarthy, J. Stalker, A. Chakravarti, B. M. Knoppers, G. R. Abecasis, K. C. Barnes, C. Beiswanger, E. Burchard, C. D. Bustamante, H. Cai, H. Cao, R. M. Durbin, N. Gharani, R. A. Gibbs, C. R. Gignoux, S. Gravel, B. Henn, D. Jones, L. Jorde, J. S. Kaye, A. Keinan, A. Kent, A. Kerasidou, Y. Li, R. Mathias, G. McVean, A. Moreno-Estrada, P. N. Ossorio, M. Parker, D. Reich, C. N. Rotimi, C. D. Royal, K. Sandoval, Y. Su, R. Sudbrak, Z. Tian, B. Timmermann, S. Tishkoff, L. H. Toji, C. Tyler-Smith, M. Via, Y. Wang, H. Yang, L. Yang, J. Zhu, W. Bodmer, G. Bedoya, A. Ruiz-Linares, C. Z. Ming, G. Yang, C. J. You, L. Peltonen, A. Garcia-Montero, A. Orfao, J. Dutil, J. C. Martinez-Cruzado, T. K. Oleksyk, L. D. Brooks, A. L. Felsenfeld, J. E. McEwen, N. C. Clemm, A. Duncanson, M. Dunn, E. D. Green, M. S. Guyer, J. L. Peterson, G. R. Abecasis, A. Auton, L. D. Brooks, M. A. DePristo, R. M. Durbin, R. E. Handsaker, H. M. Kang, G. T. Marth, and G. A. McVean. Integrative annotation of variants from 1092 humans: Application to cancer genomics. *Science*, 342:1235587, 2013.

[53] M. Kircher, D. M. Witten, P. Jain, B. J. O'Roak, G. M. Cooper, and J. Shendure. A general framework for estimating the relative pathogenicity of human genetic variants. *Nature Genetics*, 46:310–315, 2014.

[54] B. Konopka, Z. Paszko, A. Janiec-Jankowska, and M. Goluda. Assesement of the quality and frequency of mutations occurrence in PTEN gene in endometrial carcinomas

and hyperplasias. *Cancer Letters*, 178:43–51, 2002.

[55] M. Larkin, G. Blackshields, and N. Brown. Clustal W and Clustal X version 2.0. *Bioinformatics*, 23:2947–2948, 2007.

[56] S. H. Lee, J. W. Lee, Y. H. Soung, H. S. Kim, W. S. Park, S. Y. Kim, J. H. Lee, J. Y. Park, Y. G. Cho, C. J. Kim, S. W. Nam, S. H. Kim, J. Y. Lee, and N. J. Yoo. BRAF and KRAS mutations in stomach cancer. *Oncogene*, 22:6942–6945, 2003.

[57] W. Lee, Y. Zhang, K. Mukhyala, R. A. Lazarus, and Z. Zhang. Bi-directional SIFT predicts a subset of activating mutations. *PLOS ONE*, 4:e8311, 2009.

[58] J. Lin, M. Takata, H. Murata, Y. Goto, K. Kido, S. Ferrone, and T. Saida. Polyclonality of BRAF mutations in acquired melanocytic nevi. *Journal of the National Cancer Institute*, 101:1423–1427, 2009.

[59] M. Liu, L. T. Watson, and L. Zhang. Classification of mutations by functional impact type: Gain of function, loss of function, and switch of function. In Mitra Basu, Yi Pan, and Jianxin Wang, editors, *Bioinformatics Research and Applications - 10th International Symposium, ISBRA*, volume 8492 of *Lecture Notes in Computer Science*, pages 236–242. Springer, 2014.

[60] M. Liu, L. T. Watson, and L. Zhang. Combined effect of multiple genetic variants. `http://bioinformatics.cs.vt.edu/zhanglab/multivar/`, 2014.

[61] M. Liu, L. T. Watson, and L. Zhang. Quantitative prediction of the effect of genetic variation using hidden Markov models. *BMC Bioinformatics*, 15:5, 2014.

[62] M. Liu, L. T. Watson, and L. Zhang. HMMvar-func: A new method for predicting the functional outcome of genetic variants. `http://bioinformatics.cs.vt.edu/zhanglab/hmmvar-func/data/supplement1.xlsx`, 2015.

[63] K. R. Loeb and L. A. Loeb. Significance of multiple mutations in cancer. *Carcinogenesis*, 21(3):379–385, 2000.

[64] S. Martello and P. Toth. 4 subset-sum problem. In *Knapsack problems: Algorithms and computer interpretations*, pages 105–136. Wiley Interscience, 1990.

[65] R. Mills, W. Pittard, and J. Mullaney. Natural genetic variation caused by small insertions and deletions in the human genome. *Genome Research*, 21:830–839, 2011.

[66] DNA sequencing & the Human Genome Project. `http://www.lehigh.edu/~inbios21/PDF/Fall2013/Marzillier_11132013.pdf`, 2013.

[67] S. Ng, E. A. Collisson, A. Sokolov, T. Goldstein, A. Gonzalez-Perez, N. Lopez-Bigas, C. Benz, D. Haussler, and J. M. Stuart. PARADIGM-SHIFT predicts the function of mutations in multiple cancers using pathway impact analysis. *Bioinformatics*, 28:i640–i646, 2012.

[68] R. I. Nicholson, J. M. Gee, and M. E. Harper. EGFR and cancer prognosis. *European Journal of Cancer*, 34 Suppl 4:S9–15, 2001.

[69] N. Normanno, A. De Luca, C. Bianco, L. Strizzi, M. Mancino, M. R. Maiello, A. Carotenuto, G. De Feo, F. Caponigro, and D. S. Salomon. Epidermal growth factor receptor (EGFR) signaling in cancer. *Gene*, 366(1):2–16, 2006.

[70] P. E. O'Maille, A. Malone, N. Dellas, B. Andes Hess, L. Smentek, I. Sheehan, B. T. Greenhagen, J. Chappell, G. Manning, and J. P. Noel. Quantitative exploration of the catalytic landscape separating divergent plant sesquiterpene synthases. *Nature Chemical Biology*, 4(10):617–623, 2008.

[71] Art P. and Lin C. The rate of compensatory mutation in the DNA bacteriophage ϕx174. *Genetics*, 170(3):989–999, 2005.

[72] S. Pabinger, A. Dander, M. Fischer, R. Snajder, M. Sperk, M. Efremova, B. Krabichler, M. R. Speicher, J. Zschocke, and Z. Trajanoski. A survey of tools for variant analysis of next-generation genome sequencing data. *Brief Bioinformatics*, 15:256–278, 2014.

[73] B. Paten, J. Herrero, K. Beal, S. Fitzgerald, and E. Birney. Enredo and Pecan: Genome-wide mammalian consistency-based multiple alignment with paralogs. *Genome Research*, 18:1814–1828, 2008.

[74] B. Paten, J. Herrero, S. Fitzgerald, K. Beal, P. Flicek, I. Holmes, and E. Birney. Genome-wide nucleotide-level mammalian ancestor reconstruction. *Genome Research*, 18:1829–1843, 2008.

[75] C. Pauline and S. Henikoff. Predicting deleterious amino acid substitutions. *Genome Research*, 11:863–874, 2011.

[76] A. Petitjean, E. Mathe, S. Kato, C. Ishioka, S. V. Tavtigian, P. Hainaut, and M. Olivier. Impact of mutant p53 functional properties on TP53 mutation patterns and tumor phenotype: lessons from recent developments in the IARC TP53 database. *Human Mutation*, 28:622–629, 2007.

[77] K. D. Pruitt, T. Tatusova, and D. R. Maglott. NCBI reference sequence (RefSeq): A curated non-redundant sequence database of genomes, transcripts and proteins. *Nucleic Acids Research*, 33:D501–D504, 2005.

[78] M. Punta, P. C. Coggill, R. Y. Eberhardt, J. Mistry, J. Tate, C. Boursnell, N. Pang, K. Forslund, G. Ceric, J. Clements, A. Heger, L. Holm, E. L. Sonnhammer, S. R. Eddy, A. Bateman, and R. D. Finn. The Pfam protein families database. *Nucleic Acids Research*, 40:D290–D301, 2012.

[79] V. Ramensky, P. Bork, and S. Sunyaev. Human non-synonymous SNPs: Server and survey. *Nucleic Acids Research*, 30:3894–3900, 2002.

[80] B. Reva, Y. Antipin, and C. Sander. Determinants of protein function revealed by combinatorial entropy optimization. *Genome Biology*, 8:R232, 2007.

[81] B. Reva, Y. Antipin, and C. Sander. Predicting the functional impact of protein mutations: Application to cancer genomics. *Nucleic Acids Research*, 39:e118, 2011.

[82] G. R. Ritchie, I. Dunham, E. Zeggini, and P. Flicek. Functional annotation of non-coding sequence variants. *Nature Methods*, 11:294–296, 2014.

[83] I. Rodriguez-Escudero, M. D. Oliver, A. Andres-Pons, M. Molina, V. J. Cid, and R. Pulido. A comprehensive functional analysis of PTEN mutations: Implications in tumor- and autism-related syndromes. *Human Molecular Genetics*, 20(21):4132–4142, 2011.

[84] D. Schmidt, M. D. Wilson, B. Ballester, P. C. Schwalie, G. D. Brown, A. Marshall, C. Kutter, S. Watt, C. P. Martinez-Jimenez, S. Mackay, I. Talianidis, P. Flicek, and D. T. Odom. Five-vertebrate ChIP-seq reveals the evolutionary dynamics of transcription factor binding. *Science*, 328(5981):1036–1040, 2010.

[85] A. Schönhuth, R. Salari, F. Hormozdiari, A. Cherkasov, and S. C. Sahinalp. Towards improved assessment of functional similarity in large-scale screens: A study on indel length. *Journal of Computational Biology*, 17(1):1–20, 2010.

[86] S. Sherry, M. Ward, and M. Kholodov. dbSNP: The ncbi database of genetic variation. *Nucleic Acids Research*, 29(1):308–311, 2001.

[87] H. A. Shihab, J. Gough, D. N. Cooper, P. D. Stenson, G. L. Barker, K. J. Edwards, I. N. Day, and T. R. Gaunt. Predicting the functional, molecular and phenotypic consequences of amino acid substitutions using hidden Markov models. *Human Mutation*, 34:57–65, 2013.

[88] H. A. Shihab, M. F. Rogers, J. Gough, M. Mort, D. N. Cooper, I. N. Day, T. R. Gaunt, and C. Campbell. An integrative approach to predicting the functional effects of non-coding and coding sequence variation. *Bioinformatics*, 31(10):1536–1543, 2015.

[89] A. Siepel, G. Bejerano, J. S. Pedersen, A. S. Hinrichs, M. Hou, K. Rosenbloom, H. Clawson, J. Spieth, L. W. Hillier, S. Richards, G. M. Weinstock, R. K. Wilson, R. A. Gibbs, W. J. Kent, W. Miller, and D. Haussler. Evolutionarily conserved elements in vertebrate, insect, worm, and yeast genomes. *Genome Research*, 15:1034–1050, 2005.

[90] T. Sjöblom, S. Jones, L. D. Wood, D. W. Parsons, J. Lin, T. D. Barber, D. Mandelker, R. J. Leary, J. Ptak, N. Silliman, S. Szabo, P. Buckhaults, C. Farrell, P. Meeh, S. D. Markowitz, J. Willis, D. Dawson, J. K. Willson, A. F. Gazdar, J. Hartigan, L. Wu, C. Liu, G. Parmigiani, B. H. Park, K. E. Bachman, N. Papadopoulos, B. Vogelstein, K. W. Kinzler, and V. E. Velculescu. The consensus coding sequences of human breast and colorectal cancers. *Science*, 314(5797):268–274, 2006.

[91] K. Sjolander, K. Karplus, M. Brown, R. Hughey, A. Krogh, I. S. Mian, and D. Haussler. Dirichlet mixtures: A method for improved detection of weak but significant protein sequence homology. *Computer Applications in the Biosciences*, 12:327–345, 1996.

[92] T. Soussi, C. Ishioka, and M. Claustres. Locus-specific mutation databases: Pitfalls and good practice based on the p53 experience. *Nat. Rev. Cancer*, 6:83–90, 2006.

[93] M. Stamp. A revealing introduction to hidden Markov models. `http://www.cs.sjsu.edu/faculty/stamp/RUA/HMM.pdf`, 2004.

[94] P. Stenson, E. Ball, and M. Mort. Human gene mutation database (HGMD): 2003 update. *Human Mutation*, 21(6):577–581, 2003.

[95] P. Stenson, M. Mort, and E. Ball. The human gene mutation database: 2008 update. *Genome Medicine*, 22(1):13, 2009.

[96] E. A. Stone and A. Sidow. Physicochemical constraint violation by missense substitutions mediates impairment of protein function and disease severity. *Genome Research*, 15:978–986, 2005.

[97] J. S. Sutcliffe, R. J. Delahanty, H. C. Prasad, J. L. McCauley, Q. Han, L. Jiang, C. Li, S. E. Folstein, and R. D. Blakely. Allelic heterogeneity at the serotonin transporter locus (SLC6A4) confers susceptibility to autism and rigid-compulsive behaviors. *The American Journal of Human Genetics*, 77(2):265–279, 2005.

[98] B. E. Suzek, H. Huang, P. McGarvey, R. Mazumder, and C. H. Wu. UniRef: Comprehensive and non-redundant UniProt reference clusters. *Bioinformatics*, 23(10):1282–1288, 2007.

[99] The Cancer Genome Atlas. `http://cancergenome.nih.gov/`, 2006.

[100] P. D. Thomas, M. J. Campbell, A. Kejariwal, H. Mi, B. Karlak, R. Daverman, K. Diemer, A. Muruganujan, and A. Narechania. PANTHER: A library of protein families and subfamilies indexed by function. *Genome Research*, 13:2129–2141, 2003.

[101] TSH receptor mutation database II. `http://endokrinologie.uniklinikum-leipzig.de/tsh/`, 2013.

[102] M. Tuna and I. C. Amos. Activating mutations and targeted therapy in cancer. In David Cooper, editor, *Mutations in Human Genetic Disease*. InTech, New York, 2012.

[103] E. A. Varga, A. C. Sturm, C. P. Misita, and S. Moll. Homocysteine and MTHFR mutations. *Circulation*, 111:e289–e293, 2005.

[104] A. M. Waterhouse, J. B. Procter, D. M. Martin, M. Clamp, and G. J. Barton. Jalview version 2 - a multiple sequence alignment editor and analysis workbench. *Bioinformatics*, 25(9):1189–1191, 2009.

[105] A. Wetterbom, M. Sevov, L. Cavelier, and T. F. Bergstrom. Comparative genomic analysis of human and chimpanzee indicates a key role for indels in primate evolution. *Journal of Molecular Evolution*, 63:682–690, 2006.

[106] L. E. Williams and J. J. Wernegreen. Sequence context of indel mutations and their effect on protein evolution in a bacterial endosymbiont. *Bioinformatics*, 5:599–605, 2013.

[107] W. C. Wong, D. Kim, H. Carter, M. Diekhans, M. C. Ryan, and R. Karchin. CHASM and SNVBox: Toolkit for detecting biologically important single nucleotide mutations in cancer. *Bioinformatics*, 27:2147–2148, 2011.

[108] N. C. Wu, A. P. Young, and Dandekar S. Systematic identification of H274Y compensatory mutations in influenza a virus neuraminidase by high-throughput screening. *Journal of Virology*, 87(2):1193–1199, 2013.

[109] A. Zia and A. Moses. Ranking insertion, deletion and nonsense mutations based on their effect on genetic information. *BMC Bioinformatics*, 12:299, 2011.

Appendix A

The Cumulative Conjecture

As shown in Table 3.3, the set score is always approximately equal to the sum of the two single scores (i.e., $S_{1,2} \approx S_1 + S_2$). We propose Conjecture 1 and theoretically prove it as follows. Let B_w be the bit score for the wild type sequence, B_m^1 the bit score for the mutant sequence with variant v_1, B_m^2 the bit score for the mutant sequence with variant v_2, and $B_m^{1,2}$ the bit score for the mutant sequence with v_1 and v_2. Then the corresponding HMMvar scores (bit score difference) are

$$\begin{aligned} S_1 &= B_w - B_m^1, \\ S_2 &= B_w - B_m^2, \\ \text{and } S_{1,2} &= B_w - B_m^{1,2}. \end{aligned}$$

Assume $S_{1,2} = S_1 + S_2$, we have $B_w = B_m^1 + B_m^2 - B_m^{1,2}$. Therefore, we only need to show $B_w = B_m^1 + B_m^2 - B_m^{1,2}$. The bit score is defined in Equation 2.1. The numerator in Equation 2.1 is the total probability shown in Equation 1.3. If the profile HMM is dominated by one path(e.g., $Q = q_0 q_1, \ldots, q_{T-1}$), there is a very low probability (approximately to zero) of any branches other than those in the dominant path. Thus, the total probability (Equation 1.3) is approximately equal to $P(O, Q \mid \lambda)$. Let the length of the wild type sequence be n ($T = n$

in the HMM model). The logarithm of the joint probability is

$$\begin{aligned}
& \log P(O_0 O_2, \ldots, O_{n-1} q_0 q_2, \ldots, q_{n-1} \mid \lambda) \\
= \; & \log\big(P(O_0 O_2, \ldots, O_{n-1} \mid q_0 q_2, \ldots, q_{n-1}, \lambda) * p(q_0 q_2, \ldots, q_{n-1}, \lambda)\big) \\
= \; & \log\big(P(q_0) \prod_{i=1}^{n-1} P(q_i \mid q_{i-1}) \prod_{i=0}^{n-1} P(O_i \mid q_i)\big) \\
= \; & \log P(q_0) + \sum_{i=1}^{n-1} \log P(q_i \mid q_{i-1}) + \sum_{i=0}^{n-1} \log P(O_i \mid q_i).
\end{aligned}$$

Let us ignore the 'NULL' model in Equation 2.1 for now, or consider it as a constant. Let O_l and O_k denote the normal amino acids at position l and k, and O'_l and O'_k the alternative allele (mutant ones) at position l and k. The bit score of the wild type sequence and the corresponding three kind of mutant sequences are B_w, B^1_m, B^2_m, and $B^{1,2}_m$, respectively.

Calculate

$$
\begin{aligned}
B_w &= \log P(O_0O_2,\ldots,O_l,\ldots,O_k,\ldots,O_{n-1}q_0q_2,\ldots,q_{n-1} \mid \lambda) \\
&= \log P(q_0) + \sum_{i=1}^{n-1} \log P(q_i \mid q_{i-1}) + \sum_{i=0}^{l-1} \log P(O_i \mid q_i) \\
&\quad + \log P(O_l \mid q_l) + \sum_{l+1}^{k-1} P(O_i \mid q_i) + \log P(O_k \mid q_k) + \sum_{k+1}^{n-1} P(O_i \mid q_i), \\
B_m^1 &= \log P(O_0O_2,\ldots,O'_l,\ldots,O_k,\ldots,O_{n-1}q_0q_2,\ldots,q_{n-1} \mid \lambda) \\
&= \log P(q_0) + \sum_{i=1}^{n-1} \log P(q_i \mid q_{i-1}) + \sum_{i=0}^{l-1} \log P(O_i \mid q_i) \\
&\quad + \log P(O'_l \mid q_l) + \sum_{l+1}^{k-1} P(O_i \mid q_i) + \log P(O_k \mid q_k) + \sum_{k+1}^{n-1} P(O_i \mid q_i), \\
B_m^2 &= \log P(O_0O_2,\ldots,O_l,\ldots,O'_k,\ldots,O_{n-1}q_0q_2,\ldots,q_{n-1} \mid \lambda) \\
&= \log P(q_0) + \sum_{i=1}^{n-1} \log P(q_i \mid q_{i-1}) + \sum_{i=0}^{l-1} \log P(O_i \mid q_i) \\
&\quad + \log P(O_l \mid q_l) + \sum_{l+1}^{k-1} P(O_i \mid q_i) + \log P(O'_k \mid q_k) + \sum_{k+1}^{n-1} P(O_i \mid q_i), \\
B_m^{1,2} &= \log P(O_0O_2,\ldots,O'_l,\ldots,O'_k,\ldots,O_{n-1}q_0q_2,\ldots,q_{n-1} \mid \lambda) \\
&= \log P(q_0) + \sum_{i=1}^{n-1} \log P(q_i \mid q_{i-1}) + \sum_{i=0}^{l-1} \log P(O_i \mid q_i) \\
&\quad + \log P(O'_l \mid q_l) + \sum_{l+1}^{k-1} P(O_i \mid q_i) + \log P(O'_k \mid q_k) + \sum_{k+1}^{n-1} P(O_i \mid q_i).
\end{aligned}
$$

Due to the existence of the dominant path, it is very likely that the bit scores of a wild type sequence and the corresponding mutant type sequences are mainly contributed by the dominant path. As a result, if we only consider the dominant path, we can derive

$$B_w = \ B_m^1 + B_m^2 - B_m^{1,2}.$$

Because the probability of going through other path is not exactly zero and there is a background distribution (the null model), $S_{1,2}$ is not strictly equal to $S_1 + S_2$. Figure A.1 shows a part of the profile HMM built from the homologous sequence of βMHC protein. The

path going through all the match states is dominant in the HMM in terms of the transition probabilities. The purple coded states consists of the optimal path for the wild type sequence, as well as for the three mutant sequences. As a result, the set score of these two variants is approximately equal to the sum of the two single scores.

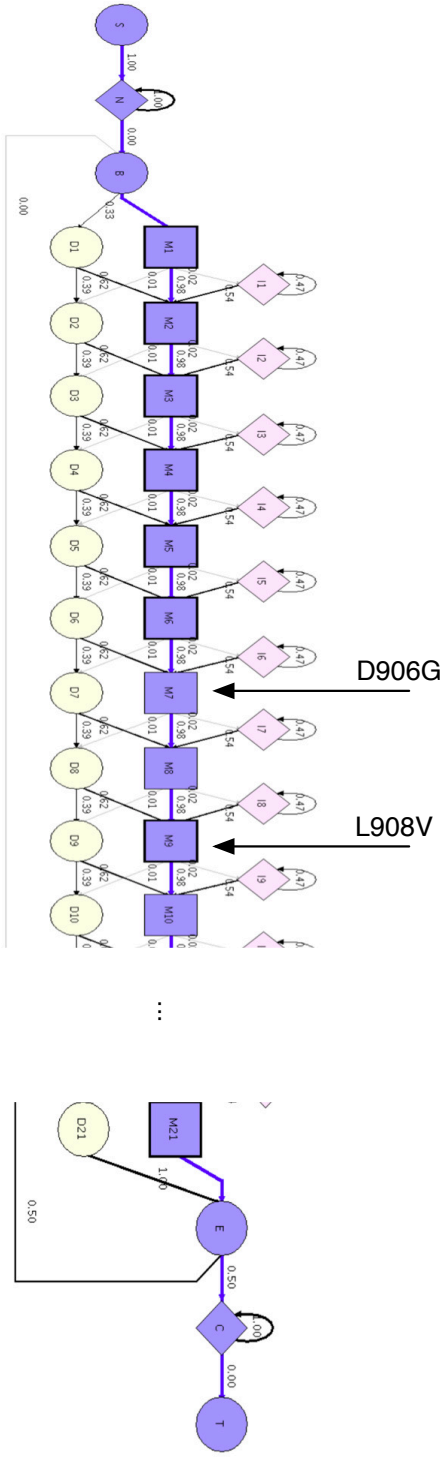


Figure A.1: A section of the profile HMM of the homologous sequence of βMHC protein.Two variants (D906G and L908L) are marked.

Appendix B

A Prediction System for Interpretation of Genomic Variants

Putting all the pieces together, the ultimate goal is to build a comprehensive system that can predict and interpret the functional effects of genomic variants. We present a comprehensive variant prediction system from a theoretical point of view in this appendix.

The inputs are genomic variants, which could be SNPs or indels in coding or noncoding regions. In the prediction step, a quantitative score is calculated to measure the harmfulness of variants, followed by the interpretation step, where annotations, related experiments, literature, and cross references are provided for a complete interpretation of the scores.

Users will be able to not only download the precomputed predictions for possible functional impact of SNPs or indels, but also upload their own variants to the system for functional annotation. The prediction for the likely functional effects of variants will enable biologists to prioritize variants for downstream empirical studies. Moreover, according to the prediction, variants can be selected and incorporated into assays to study their association with various traits or diseases. The high-level design of the system, shown in Figure B.1, consists of four modules: query, validation, core, and storage and visualization. The core module

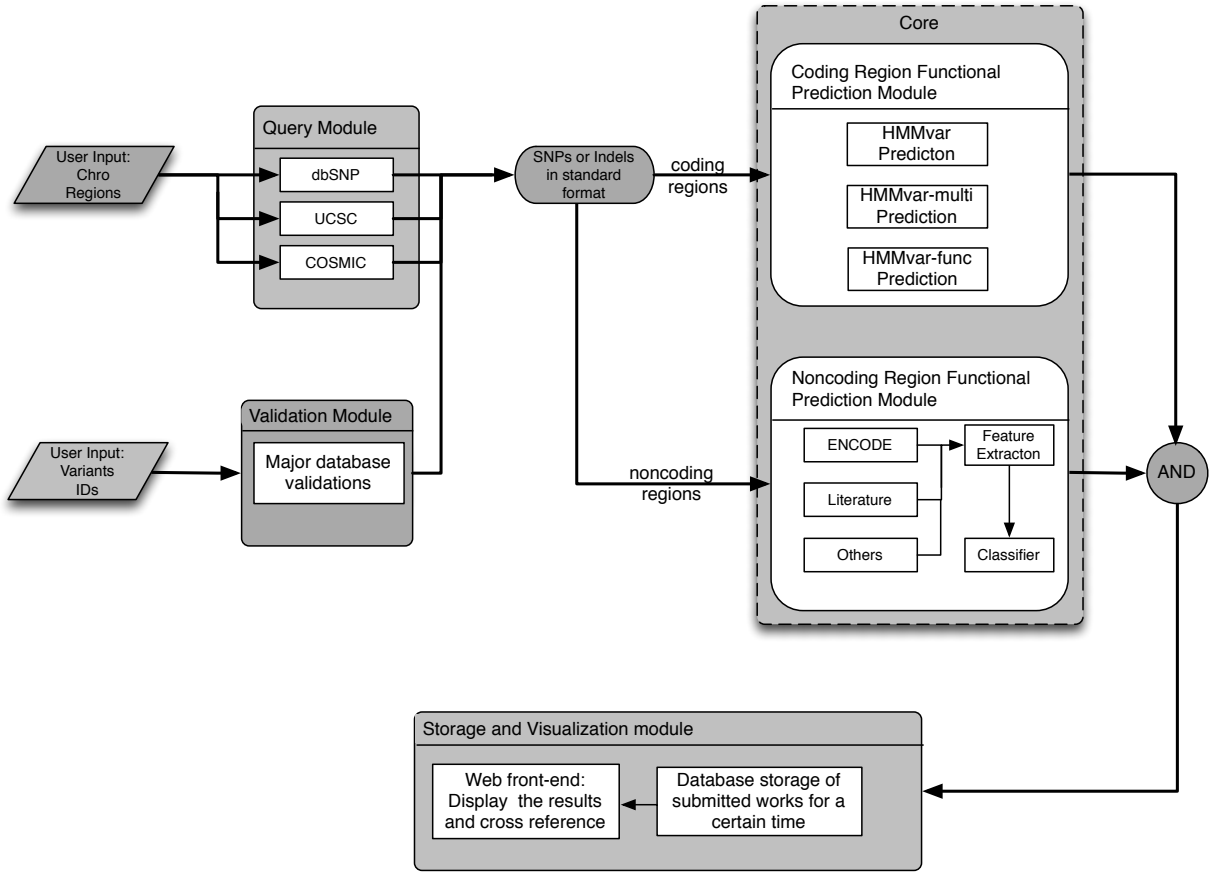


Figure B.1: The system architecture

makes functional predictions. The input can be either the specified chromosome regions or variant IDs from different databases. If the input is a chromosome region, the system queries databases to extract detailed information about the region, including all the variants in this region, lengths of the variants, associated gene/protein sequences, and relative positions of the variants to protein sequences. If the input is variant IDs from major databases, the validation module first confirms that the submitted IDs are valid in major databases. The variants are then transformed to standard format and fed to the core module for functional prediction in coding or noncoding regions. For variants in coding regions, the suite of programs HMMvar, HMMvar-multi, and HMMvar-func perform the predictions. For variants in noncoding regions, the functional prediction for noncoding regions described in Chapter

5 will be conducted. The storage and visualization module aims to store and display the results.