

PREDICTION OF THE EFFECTS OF SINGLE AMINO ACID VARIATIONS
ON PROTEIN FUNCTIONALITY
WITH STRUCTURAL AND ANNOTATION CENTRIC MODELING

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF INFORMATICS OF
THE MIDDLE EAST TECHNICAL UNIVERSITY
BY

FATMA CANKARA

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE
IN
THE DEPARTMENT OF BIOINFORMATICS

JANUARY 2020

Approval of the thesis:

PREDICTION OF THE EFFECTS OF SINGLE AMINO ACID VARIATIONS
ON PROTEIN FUNCTIONALITY
WITH STRUCTURAL AND ANNOTATION CENTRIC MODELING

Submitted by Fatma Cankara in partial fulfillment of the requirements for the degree of **Master of Science in Health Informatics, Middle East Technical University** by,

Prof. Dr. Deniz Zeyrek Bozşahin
Dean, **Graduate School of Informatics**

Assoc. Prof. Dr. Yeşim Aydın Son
Head of Department, **Health Informatics**

Assoc. Prof. Dr. Nurcan Tunçbağ
Supervisor, **Health Informatics, METU**

Assoc. Prof. Dr. Tunca Doğan
Co-Supervisor, **Computer Engineering Dept.,
Hacettepe University**

Examining Committee Members:

Assoc. Prof. Dr. Yeşim Aydın Son
Health Informatics, METU

Assoc. Prof. Dr. Nurcan Tunçbağ
Health Informatics, METU

Assoc. Prof. Dr. Tunca Doğan
Computer Engineering Dept.,
Hacettepe University

Assist. Prof. Dr. Aybar Can Acar
Health Informatics, METU

Assoc. Prof. Dr. Özlen Konu Karakayalı
Molecular Biology and Genetics Dept., Bilkent University

Date:

30.01.2020



I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name: FATMA CANKARA

Signature : _____

ABSTRACT

PREDICTION OF THE EFFECTS OF SINGLE AMINO ACID VARIATIONS ON PROTEIN FUNCTIONALITY WITH STRUCTURAL AND ANNOTATION CENTRIC MODELING

Cankara, Fatma

MSc., Department of Bioinformatics

Supervisor: Assoc. Prof. Dr. Nurcan Tunçbağ

Co-Supervisor: Assoc. Prof. Dr. Tunca Doğan

January 2020, 142 pages

Whole-genome and exome sequencing studies have indicated that genomic variations may cause deleterious effects on protein functionality via various mechanisms. Single nucleotide variations that alter the protein sequence, and thus, the structure and the function, namely non-synonymous SNPs (nsSNP), are associated with many genetic diseases in human. The current rate of manually annotating the reported nsSNPs cannot catch up with the rate of producing new sequencing data. To aid this process, automated computational approaches are being developed and applied on the unknown data. In this study, we propose a new methodology to collect and organize the information related to the effects of nsSNPs at the amino acid sequence level from various biological databases and to utilize this information in a supervised machine-learning based system to predict the function disrupting capacities of mutations with unknown consequences. For this, 157,138 annotated mutation data points (89,363 deleterious and 67,775 neutral) were collected from multiple resources such as UniProt, ClinVar and Protein Mutant Database. For each mutation data point, a feature vector was constructed using protein 3-D structure information and site-specific feature annotations in the UniProt database. The information about the spatial proximity of the reported mutations to these protein features were also incorporated to the feature vector. The system was trained with these feature vectors and their respective labels in a supervised fashion using random forest, where the ultimate aim was to construct a model that classifies unknown mutations either as deleterious or neutral.

The prediction model was evaluated in detail to observe the contribution of different feature types to the prediction success. The finalized model displayed a satisfactory performance (AUROC:0.86, precision: 0.77, recall 0:90, accuracy: 0.78, F1-score: 0.83 and MCC: 0.54) on the independent test dataset. Besides, the performance of the proposed model was compared to the widely used variant effect predictors in the literature, over standard benchmark datasets. As future work, we plan to conduct a case study over interesting prediction examples and to validate our results via literature-based information. Finally, we plan to construct a ready-to-use command line based variant effect prediction tool and to share it with the research community over an open access data repository. We believe that this system will be complementary to the well-known methods in the literature and its incorporation to ensemble-based tools will increase the performance of the state-of-the-art in variant effect prediction.

Keywords: Single amino acid variations, variant effect prediction, protein sequence annotations, machine learning, random forest.

ÖZ

TEKİL AMİNO ASİT MUTASYONLARININ PROTEİN İŞLEVLERİ ÜZERİNDEKİ ETKİSİNİN YAPISAL VE ANOTASYON ODAKLI YAKLAŞIMLA TAHMİNİ

Cankara, Fatma

Yüksek Lisans, Biyoenformatik Bölümü

Tez Yöneticisi: Doç. Dr. Nurcan Tunçbağ

Eş Tez Yöneticisi: Doç. Dr. Tunca Doğan

Ocak 2020, 142 sayfa

Genom üzerindeki tekli nükleotid değişiklikleri protein dizisi, yapısı ve kararlılığı üzerinde yarattığı etkiler aracılığıyla proteinlerin işlevlerinde önemli değişikliklere sebep olabilir. Bu sinonim olmayan tek nükleotid polimorfizmleri, insanda pek çok hastalığın oluşumundan sorumludur. Uzmanların bu mutasyonları anote etme (etiketleme) hızı, günümüzde yeni dizi verisi üretme hızının çok gerisinde kalmaktadır. Bu süreci hızlandırmak için hesaplamalı yöntemler geliştirilmekte ve otomatize şekilde bilinmeyen veri üzerinde uygulanmaktadır. Bu çalışmada, sinonim olmayan tekli nükleotid değişikliklerinin amino asit seviyesinde gösterdikleri etkiler hakkındaki bilgilerin çeşitli veri tabanlarından toplanması ve organize edilmesi, bunun yanında bu bilginin etkisi bilinmeyen tekli nükleotid değişikliklerinin proteinin işlevine zarar verme potansiyellerinin gözetimli makine öğrenmesi yaklaşımı kullanarak tahmini için bir metodoloji sunulmuştur. Bu amaçla, UniProt, ClinVar ve PMD gibi çeşitli veri tabanlarından anote edilmiş 157,138 mutasyon (89,363 zarar gösteren ve 67,775 zarar göstermeyen) toplanmıştır. Her mutasyon veri noktası için, ilgili genin ürünü olan proteinin 3 boyutlu yapı bilgisi ve bölgesel UniProt dizi anotasyonları kullanılarak bir öznitelik vektörü oluşturulmuştur. Ayrıca, her mutasyon öznitelik vektörüne o mutasyonun, üzerinde bulunduğu genin ürünü olan proteinin bölgesel dizi anotasyonlarına olan uzaysal uzaklığı eklenmiştir. Bu öznitelik vektörleri ve bunların etiketleri kullanılarak, amacı mutasyonları protein işlevine zarar verenler ve zarar vermeyenler şeklinde sınıflandırmak olan ve rastgele orman algoritmasını kullanan bir makine öğrenmesi modeli geliştirilmiştir. Bu model çeşitli öznitelik alt gruplarının tahmin

başarısına etkisini ölçmek üzere detaylı bir şekilde değerlendirilmiştir ve nihai model bağımsız bir test seti üzerinde tatmin edici bir başarıya ulaşmıştır (AUROC:0.86, kesinlik: 0.77, duyarlılık 0:90, doğruluk: 0.78, F1-puanı: 0.83 ve MCC: 0.54). Ayrıca, modelin performansının, standart bir veri seti üzerinden mutasyon etki tahmini yapan yaygın yöntemlerin sonuçlarıyla kıyaslaması gerçekleştirilmiştir. Gelecekte yapılacak çalışmalar olarak, bir vaka çalışması yürütülerek, yeni mutasyon etki tahmin sonuçlarının literatür bazlı bilgi ile doğrulanması planlanmaktadır. Ayrıca, geliştirilen yöntemin kullanıma hazır bir komut satırı aracı haline getirilerek açık kaynaklı bir veri deposu vasıtasıyla araştırma topluluğuyla paylaşılması amaçlanmaktadır. Geliştirilen yöntemin literatürde sıkça kullanılmakta olan mutasyon etki tahmini araçlarıyla beraber olarak kullanılmasının tamamlayıcı bir etki yaratacağı ve bu yöntemlerin tahmin performanslarını arttıracığı düşünülmektedir.

Anahtar Sözcükler: Tek amino asit değişimleri, varyasyon etki tahmini, protein dizi anotasyonları, makina öğrenmesi, rastgele orman



To My Family

ACKNOWLEDGMENTS

First and foremost, I would like to express my gratitude to my advisor Assoc. Prof. Dr. Tunca Dođan for his endless support. I deem myself lucky to start my research journey under his supervision. Throughout my time as a master's student, he was generous to share his knowledge and time with me that helped me a lot to build a solid base for my future studies. This work has been accomplished thanks to his guidance. I sincerely thank him for his patience and encouragement.

I also would like to thank Prof. Dr. Rengül Çetin-Atalay and Assoc. Dr. Nurcan Tunçbađ for their academic and administrative contributions that helped me complete my studies.

Furthermore, I would like to thank my thesis committee members Assoc. Prof. Dr. Yeşim Aydın Son, Assoc. Prof. Dr. Nurcan Tunçbađ, Assoc. Prof. Dr. Özlen Konu Karakayalı and Assist. Prof. Dr. Aybar Can Acar for their valuable time, feedback and consideration to be in my thesis committee.

I would also like to thank my dear colleague Heval Ataş for being there for me whenever I needed. Her knowledge and consulting have helped me a lot to accomplish this work. Besides, I was lucky enough to spend some memorable years during my studies as a master's student thanks to my friends. For this reason, I would like to thank Gökçe Abay, Cansu Demirel, Muazzez Çelebi Çınar, Elif Bozlak, Meriç Kınalı and Evrim Fer for their enjoyable friendship. I also would like to thank Abdulrahman Alabrash for his valuable time, knowledge, patience and for being there whenever I needed. Moreover, I would like to thank my dear friends Fatma İstanbullu and Merve Haksever for bearing with me for more than 10 years and for their ever-lasting support, kindness and friendship. My life wouldn't be the same without them.

Lastly, I owe my deepest gratitude to my beloved family for their encouragement, unconditional love and care. My mother, Nazan Cankara, and my father, Mustafa Cankara, has never stopped to support me to follow my dreams and be the person I've imagined. I also would like to thank my lovely siblings, Furkan Cankara and Feyza Cankara for making my life a lot fun and showing me the true meaning of love. This work could not be done without any of them.

TABLE OF CONTENTS

ABSTRACT	v
ÖZ	vii
DEDICATION	viii
ACKNOWLEDGMENTS.....	x
TABLE OF CONTENTS	xi
LIST OF TABLES	xiv
LIST OF FIGURES	xvi
LIST OF ABBREVIATIONS.....	xviii
CHAPTERS	
1. INTRODUCTION & BACKGROUND INFORMATION	1
1.1 Proteins and Their Functions.....	1
1.2 DNA Variations.....	2
1.2.1. Single Base Pair Substitutions	2
1.2.2. Structural Variations	4
1.3 Efforts for Characterizing DNA Variations	5
1.4 Machine-Learning Applications.....	6
1.4.1. Unsupervised Methods	7
1.4.2. Supervised Methods.....	8
1.5 Aim of The Study	9
1.6 Overview of the Thesis.....	10
2. LITERATURE REVIEW	13
2.1 Traditional vs. Predictive Methods on the Study of Effects of SAVs on Proteins.....	13
2.2 Variant Effect Assessment Using Predictive Approaches.....	15
2.3 Classification of Variant Effect Prediction Methods Based on the Modelling Approach.....	15
2.3.1. Probabilistic and Statistical Approaches	15
2.3.2. Machine-Learning Approaches.....	19

2.4. Classification of Variant Effect Prediction Methods Based on the Input Features They Utilize	21
2.4.1. Methods That Use Sequence Information.....	21
2.4.2. Methods That Use Structural Information	24
2.4.3. Ensemble and Consenses Based Methods	26
3. MATERIALS & METHOD.....	31
3.1. Data Retrieval.....	31
3.1.1. The Universal Protein Knowledgebase (UniProt)	31
3.1.2. ClinVar.....	32
3.1.3. Protein Mutant Database (PMD)	32
3.2. Feature Vector Construction.....	33
3.2.1. Domain Data	33
3.2.2. Structure Data	33
3.2.3. Physicochemical Properties	35
3.2.4. UniProt Sequence Annotations	35
3.2.5. Retrieving Mutation’s Location on the Structure	36
3.3. Classification Algorithm.....	37
3.3.1. Decision Trees	37
3.3.2. Random Forests	39
3.3.3. Model Assessment	41
3.3.4. Performance Metrics	42
3.4. Modelling Approach.....	44
3.4.1. Data Pre-processing	45
3.4.2. Feature Vector Construction	47
3.4.3. Model Implementation.....	53
4. RESULTS.....	55
4.1. Summary of The Data.....	55
4.2. Distribution of Data Points Among Domain Regions	56
4.3. Identification of Significant Protein Domains	57
4.4. Physicochemical Properties of Mutations	60
4.5. Family-Based Evaluation of Mutations	62

4.6. UniProt Annotations and Their Distribution in The Mutation Dataset	64
4.7. Distribution of Data Points In Terms of Relative Accessibility	66
4.8. Mutation Classification With Random Forest Classifier	67
4.9. Benchmark Analysis.....	84
5. DISCUSSION & CONCLUSION	93
APPENDICES	125
APPENDIX A	125
APPENDIX B	133
APPENDIX C	136
APPENDIX D.....	141

LIST OF TABLES

Table 2.1. Summary of some of the well-known methods for variant effect prediction....	16
Table 3.1. Grantham matrix scores.....	36
Table 3.2. A summary for the counts of mutation	47
Table 3.3. Annotation types and classes retrieved from UniProt	50
Table 4.1. A summary for the counts of mutations in the dataset.	56
Table 4.2. Most informative 10 domains after Fisher’s exact test analysis.....	59
Table 4.3. Significance of association (p-values) between physicochemical property and protein regions.....	62
Table 4.4. Significance of association (p-values) between physicochemical property and Protein Families.....	64
Table 4.5. Confusion matrix for the hold-out validation set of all-domains model	68
Table 4.6. Confusion matrix for the hold-out validation set of significant domains model	69
Table 4.7. Performance metrics for all-domains and significant-domains models	69
Table 4.8. Confusion matrix for the hold-out validation set of model without physicochemical properties	72
Table 4.9. Confusion matrix for the hold-out validation set of model with physicochemical properties	72
Table 4.10. Performance metrics for models with and without physicochemical properties	72
Table 4.11. Confusion matrix for the hold-out validation set of model without 3D sequence annotations	75
Table 4.12. Confusion matrix for the hold-out validation set of model with 3D sequence annotations.....	75
Table 4.13. Performance metrics for models without and with 3D sequence annotations	75
Table 4.14. Parameters for grid search for the finalized model.....	78
Table 4.15. Confusion matrix for the hold-out validation set of the finalized model.....	78
Table 4.16. Performance metrics for the best model of grid search for the finalized model.....	79
Table 4.17. Confusion matrix for the hold-out validation set of the model trained with UniProt and PMD data	82
Table 4.18. Confusion matrix for the hold-out validation set of the model trained with ClinVar data	82
Table 4.19. Performance metrics for models without and with 3D sequence annotations.....	82
Table 4.20. Data point counts for the datasets downloaded from PredictSNP.....	85

Table 4.21. Data points counts for the benchmarking analysis	85
Table 4.22. Confusion matrix for the hold-out validation set for PredictSNP benchmark data	86
Table 4.23. Performance metrics for the best model for PredictSNP benchmark set	86
Table 4.24. Confusion matrix for the hold-out validation set for PredictSNP PMD test data	87
Table 4.25. Performance metrics for the best model for PredictSNP PMD test set	88
Table 4.26. Confusion matrix for the hold-out validation set for PredictSNP MMP test data	89
Table 4.27. Performance metrics for the best model for PredictSNP MMP test set	89
Table 4.28. Overall comparison of different classifiers.....	91

LIST OF FIGURES

Figure 1.1. Ribbon diagrams of example protein architectures.....	2
Figure 1.2. Some commonly used machine learning algorithms.....	7
Figure 1.3. An example workflow for machine learning methods.....	7
Figure 2.1. Workflow of SIFT.....	17
Figure 2.2. MAPP analysis steps.....	18
Figure 2.3. Summary of the MutationAssessor method.....	24
Figure 2.4. PolyPhen-2 pipeline.....	27
Figure 2.5. SNPs&GO input schema.....	28
Figure 2.6. Workflow diagram of PredictSNP.	30
Figure 3.1. Example cross-validation scheme.....	41
Figure 3.2. An example confusion matrix.....	42
Figure 3.3. An example ROC curve.....	44
Figure 3.4. Overview of the method.....	45
Figure 3.5. Data distribution in data sources.....	47
Figure 3.6. Feature vector representation of the considered features.....	48
Figure 3.7. 3D structures are searched for data points.....	49
Figure 3.8. Mapping o mutations and annotations on structure and distance calculations.....	52
Figure 4.1. Distribution of data points for their presence within or out of the domain boundaries.	57
Figure 4.2. Distribution of Domains for Their Abundance After Fisher’s Exact Test....	58
Figure 4.3. Distribution of data points for the physicochemical properties considered....	61
Figure 4.4. Percentage of each annotation category in the dataset.....	65
Figure 4.5. Percentage of neutral and deleterious mutations per annotation class for mutations found to occupy an annotation region.....	66
Figure 4.6. Distribution of data points for accessibility group.....	67
Figure 4.7. ROC curves for the hold-out validation sets of the models that are trained with all domain and significant domains.....	70
Figure 4.8. Feature importance for all-domains and significant-domain models.....	71
Figure 4.9. ROC curves for the hold-out validation sets of the models that are trained with and without physicochemical properties.....	73
Figure 4.10. Feature importance models with and without physicochemical properties...74	
Figure 4.11. ROC curves for the hold-out validation sets of the models that are trained with and without 3D distance information.....	76
Figure 4.12. Feature importance models with and without physicochemical properties...77	
Figure 4.13. ROC curve for trained model with after grid search.....	79

Figure 4.14. Scaled variable importance for the features for the best model of finalized vector.....	80
Figure 4.15. Log-loss graph for the best model after the grid search.....	81
Figure 4.16. ROC curves for the hold-out validation sets of models trained and validated on Humsavar -PMD data and ClinVar data	83
Figure 4.17. Feature importance models trained and validated on Humsavar-PMD data and ClinVar data	84
Figure 4.18. ROC curve for the hold-out validation set of the model generated using PredictSNP benchmark dataset	87
Figure 4.19. ROC curve for the hold-out validation set of the model generated using PMD test dataset.....	88
Figure 4.20. ROC curve for the hold-out validation set of the model generated using MMP test dataset.....	90

LIST OF ABBREVIATIONS

3D	Three Dimensional
A GV-GD	Alignment Grantham-Variation, Grantham-Deviation
ANN	Artificial Neural Networks
AUC	Area Under the Receiver Operating Characteristics Curve
bioDBnet	The Biological Database Network
CART	Classification and Regression Trees
CHASM	Cancer-Specific High-throughput Annotation of Somatic Mutations
CNV	Copy Number Variations
COSMIC	Catalogue of Somatic Mutations in Cancer
DBSCAN	Density-Based Spatial Clustering of Applications with Noise
DNA	Deoxyribonucleic Acid
éclair	Ensemble Classifier Learning Algorithm to predict Interface Residues
FIS	Functional Impact Score
FN	False Negative
FP	False Positive
HGMD	Human Gene Mutation Database
HMM	Hidden Markov Model
GWAS	Genome-Wide Association Studies
KNN	k-Nearest Neighbors
OMIM	Online Mendelian Inheritance in Man
NCBI	National Center for Biotechnology Information
NMR	Nuclear Magnetic Resonance
PDB	The Protein Data Bank
PMD	Protein Mutant Database
PolyPhen-2	Polymorphism Phenotyping v2
PSSM	Position Specific Scoring Matrix
ROC	Receiver Operating Characteristics
SASA	Solvent Accessible Surface Area
SIFT	Sorting Intolerant from Tolerant
SNV	Single Nucleotide Variant
SuSPect	Disease-Susceptibility-based SAV Phenotype Prediction
SNV	Single Amino Acid Variations
SVM	Support Vector Machine
TN	True Negative
TP	True Positive

UniParc The Universal Protein Resource Archive
UniProt The Universal Protein Resource
UniProtKB The Universal Protein Resource Knowledgebase
UniRef The Universal Protein Resource Reference Clusters





CHAPTER 1

1. INTRODUCTION & BACKGROUND INFORMATION

1.1 Proteins and Their Functions

Proteins are critical molecules in all organisms, and they are responsible for a variety of tasks in the cell; including but not limited to biological process regulation, structural assembly, cellular organization, storage, transport, defense, regulation of DNA replication and messaging. The amino acid sequence that makes up the protein defines its function as well as its folding state, binding properties, stability and localization in the cell. A protein's sequence also defines its final structure; individual structural elements and their organization in the protein. There are well-characterized elements that constitute the protein 3D structure such as alpha-helices, beta sheets, coils and turns. Their organization, number and spatial presence with respect to the other ones determine the unique quaternary structure and thus the function of the protein. An example figure for some protein structures is given in Figure 1.1 (Abe *et al.*, 2009). Protein structures and their folded states reflect the function of the proteins better than the sequence information. A protein's function can be deduced from its 3D structure; because certain folds and motifs can define certain functions (Berg *et al.*, 2002). In addition to that, since structure is better conserved than the sequence, protein function can be inferred from homologous organisms in more accurate manner (Illergård *et al.*, 2009). For these reasons focusing on the structure gives a more in-depth and accurate information when the question that is being asked is about the protein function. When changes occur in protein sequence, this may affect the processes that proteins are involved via various mechanisms. These changes that result in faulty protein products or faulty pathways primarily occur in DNA sequence and they are successively inherited to the product itself; changing its native structure or its stability. Given the relationship between the structure and the function, changes in the protein sequence that alter the structure may cause serious deleterious effects on the protein's function. Recent advances in sequencing technologies have revealed a large data in terms of sequence variations whose effects on proteins are open to interpretation. Changes that result in sequence variations may or may not alter the structure and

phenotype, ultimately, they may have beneficial, neutral or deleterious effects on the survival and fitness of the organism.

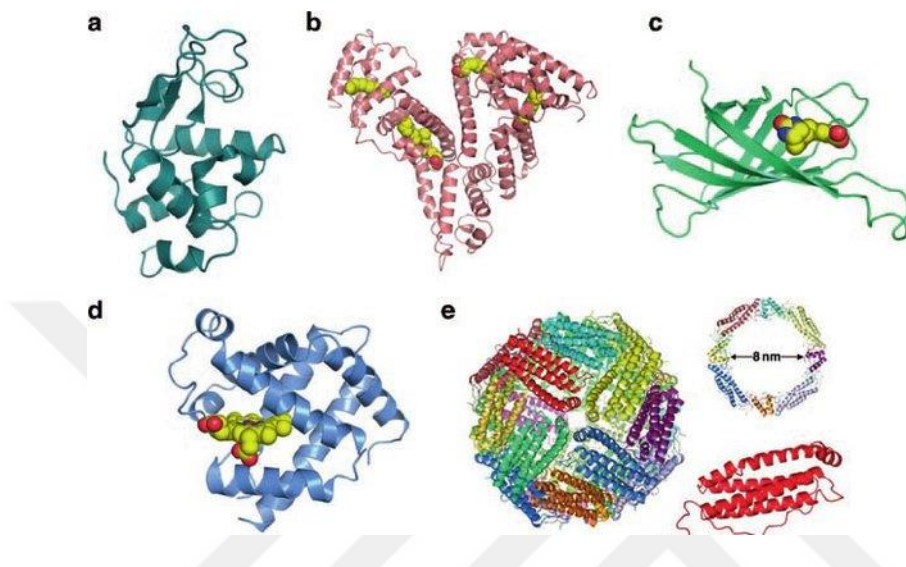


Figure 1.1. Ribbon diagrams of protein architectures: a lysozyme, b serum albumin, c avidin, d myoglobin, and e ferritin taken from PDB ID: 2VB1, 1BJ5, 1AVD, 4MBN, and 1DAT, respectively. Reprinted from ‘*Artificial Metalloproteins Exploiting Vacant Space: Preparation, Structures, and Functions*’ by Abe *et al.*, 2009, Topics in Organometallic Chemistry.

1.2. DNA Variations

1.2.1. Single Base Pair Substitutions

Changes in DNA sequence are results of mutations or structural reorganizations. A mutation is defined as any change in a DNA sequence that results in a different, abnormal or rare allele from normal one that is prevalent in the population. A mutation may or may not be significant depending on the consequences it introduces. In the most common form, mutations may alter a single nucleotide on the DNA sequence. Resulting variations are called Single Nucleotide Variants (SNVs) and they constitute the majority of sequence variations (Bromberg *et al.*, 2013). Studies show that a typical overall genome differs from the reference human genome around 4.1 million to 5.0 million sites (The 1000 Genomes Project Consortium, 2012).

When two alternative bases occur at appreciable frequency in the human population, which at least 1% of the population, they are named as Single Nucleotide Polymorphisms (SNPs). In other words, a SNP is when a change is expected at the position for any member in the species, on the other hand an SNV is when there is a variation at a position that hasn't been well characterized. Polymorphisms are common in the populations. In the case

of polymorphisms, no single allele is considered as the standard one; rather two or more alleles are equally acceptable options. Roughly 10 million such sites, on average about 1 site per 300 bases, are estimated to exist in the human population such that both alleles have a frequency of at least 1% (Belmont *et al.*, 2003). Constituting the most frequent type of DNA variation in human population, many SNPs are believed to express the most of human genetic diversity, as well as to cause phenotypic differences among individuals (Sachidanandam *et al.*, 2001; Ramensky, 2002).

Genome-wide association studies (GWAS) have identified thousands of single nucleotide polymorphisms (SNPs) associated with a large number of phenotypes in order to understand the relationship between SNPs and complex and/or monogenic diseases (Hindorff *et al.*, 2009; Manolio *et al.*, 2008). These studies showed that the majority of disease-associated SNPs are located in noncoding regions of the genome, *non-coding SNPs (ncSNPs)*, where they do not show their effects on the phenotype but rather are located on regulatory regions. These ncSNPs are believed to have effects on promoters, enhancers and non-coding RNA genes (Halushka *et al.*, 1999; Hindorff *et al.*, 2009). ncSNPs may also alter transcription factor motifs or change nearby residues and in turn alter the way of activation (Farh *et al.*, 2015; Khurana *et al.*, 2016). As they do not show any observable phenotypical outcome, these findings show that they exert their effects by changing regulatory codes in the genome. Although association studies showed a very strong connection between diseases and ncSNPs, research to understand functional consequences of sequence variations mainly focused on coding SNPs, non-synonymous SNPs in particular, due to the difficulties in interpreting non-coding mutations given the incomplete annotation of regulatory elements, diversity of non-coding functions and potentially still unknown mechanisms of regulatory control (Ward & Kellis, 2012).

SNPs that are located in coding regions of the genome (cSNPs) can be categorized as synonymous (sSNPs) and non-synonymous SNPs (nsSNPs). Synonymous SNPs are results of alterations of a single base in the DNA sequence; however, they do not affect the sequence of protein product due the phenomenon called codon-degeneracy. Since an amino acid can be recognized by more than one codon, when changes of that sort introduce another codon for the same amino acid in the mRNA sequence, product remains unchanged. Although synonymous mutations do not show a phenotypic consequence even though they occur in the coding segments, research showed that sSNPs share a similar likelihood of human disease association compared to nsSNPs (Chen *et al.*, 2010). These silent mutations can affect gene expression, transcription, splicing, protein folding and mRNA stability; thus, they can cause significant effects on protein function, and changes in cellular response to different agents (Chamary *et al.*, 2006; Edwards *et al.*, 2012; Hunt *et al.*, 2009; Pagani *et al.*, 2005; Presnyak *et al.*, 2015; Stergachis *et al.*, 2013; Zwart *et al.*, 2018). As a result of these effects, they are also shown to have associations with many diseases including pulmonary sarcoidosis, attention deficit/hyperactivity disorder, and cancer (Sauna & Kimchi-Sarfaty, 2011; Supek *et al.*, 2014). Non-synonymous mutations, on the other hand, change the sequence of the protein product as a result of single nucleotide alteration in the genome. nsSNPs, together with SNPs in regulatory regions, are believed to have the highest impact on phenotype because most of the known genetic

variation associated with inherited human diseases occurs in protein-coding regions of the genome (Datta *et al.*, 2015; Ramensky, 2002; Thomas & Kejariwal, 2004). As mentioned, nsSNPs change the amino acid sequence. Thus, they also show their effects on the protein level. These changes are expressed as single amino acid variations (SAVs) that indicates a change in the amino acid sequence due to a mutation on DNA sequence. Because the change in one nucleotide of DNA causes a change in the amino acid sequence that is to be translated, this switch, in turn, produces a new amino acid which will affect the nature of the protein itself. The human population is estimated to have 67,000–200,000 common nsSNPs and due to their disease associations, they provide a valuable information for variant interpretation (Ng & Henikoff, 2006).

One of the major goals in human genetics is to understand the role of common genetic variants in susceptibility to common diseases. Given the challenges in evaluating the ncSNPs, long-assumed silent behavior of sSNPs and more readily observable wealthy outcomes of nsSNPs, studies to infer meanings from genomic changes focused on non-synonymous mutations (Bromberg & Rost, 2007; Karchin *et al.*, 2005; Kumar *et al.*, 2009; Ramensky, 2002; Zhu *et al.*, 2008). Most of the Mendelian diseases, diseases that are caused by a mutation in a single gene, arise from a single amino acid change in an encoded protein. Examples include phenylketonuria, cystic fibrosis, sickle-cell anemia and Huntington's diseases. Databases such as Human Gene Mutation Database (HGMD) (Stenson *et al.*, 2003) and Online Mendelian Inheritance in Man (OMIM) (Hamosh *et al.*, 2005) contain disease-causing nsSNPs that are responsible for almost half of the genetic changes known to cause a disease. Although the data in these databases mainly focus on Mendelian diseases, it is likely that these mutations also play important roles in many of the complex diseases (Barrett *et al.*, 2011; Bishop *et al.*, 2009; Botstein & Risch, 2003; Bush & Moore, 2012; Datta *et al.*, 2015; Gorski *et al.*, 2015; Hepp *et al.*, 2015; Liu *et al.*, 2011; Ng & Henikoff, 2006; Thomas & Kejariwal, 2004)

1.2.2. Structural Variations

Other than single nucleotide changes on the DNA sequence, proteins and DNA may also comprise structural changes such as changes in copy number of genes (deletions, insertions and duplications), inversions and chromosomal location rearrangements (translocations, fusions). Structural variations are defined as changes in DNA regions that are greater than 1 kb in size (Stankiewicz & Lupski, 2010). Typical human genome contains estimated 2,100 to 2,500 structural variants, affecting around 20 million bases of sequence (The 1000 Genomes Project Consortium, 2012). These changes can affect a range of protein properties, such as stability, catalytic activity or the ability to interact with other molecules.

Copy Number Variations (CNVs) are type of structural variations that result in changes in the number of copies of specific DNA segments. They can be stemmed from duplications, deletions or insertions. As they occur along a long segment in DNA, they may encompass more than one gene and cause genetic defects in individuals. CNVs may

account for 13% of the human genome; however, it is anticipated that with emerging techniques to detect CNVs, a larger proportion of them will be revealed (Stankiewicz & Lupski, 2010). Studies show that CNV's are responsible for a number of disorders including but not limited to DiGeorge/velocardiofacial, Smith-Magenis, Williams-Beuren and Prader-Willi syndromes (Bishop *et al.*, 2009; Koolen *et al.*, 2006; Lee & Scherer, 2010; Shaw-Smith *et al.*, 2006). Additionally, autism, schizophrenia, epilepsy, Parkinson disease, Alzheimer disease and many more complex traits are shown to be resulting from CNVs in some fraction of patient DNAs (Stankiewicz & Lupski, 2010). Another structural variation, translocation, is observed in genomes when a segment from a chromosome is transferred to a different chromosome or to another region of the same chromosome without changing the DNA content of the segment. This type of variation results in a novel chromosome that did not exist in the native genome. When translocation joins two segments together, it is called a fusion. As translocations drastically change the organization in the genome, most of the times they result in disease conditions. One common example to such diseases can be Familial Down Syndrome which mostly is observed as a translocation between chromosomes 14 and 21 (O'Connor, 2008). Translocations may also be playing roles in certain types of cancers as they disrupt the gene function and cause a faulty product (Balmain, 2001). Final category of structural variants to cover here occurs as a result of inversions. Inversions are observed when two chromosomal segments join together after a breakage, but when one of them is inverted 180 degrees before rejoining. Although overall DNA content remains the same, inversions may cause different diseases such as hemophilia A and Hunter syndrome; or common disease such as prostate cancer (Bondeson *et al.*, 1995; Lakich *et al.*, 1993).

1.3. Efforts for Characterizing DNA Variations

Up to this point, the importance of variations in the genome and how they may have implications in a lot of diseases or malfunctions in the organisms is highlighted with some background information. A considerable number of studies have been conducted to understand the consequences of human variation and most of these studies have been revolving around extrapolating the significance of SNPs and understanding the pathogenic results of them. However, given the abundance of uncharacterized SNPs and difficulties in identifying real causal SNPs due to the biases in experimental setups like linkage disequilibrium; it would be time-consuming, difficult, labor-intensive and expensive to try to identify functional consequences of SNPs on proteins and their disease relations by traditional experiments. As a result of these challenges and endeavor to interpret the underlying meaning of mutations, a number of computational algorithms and tools have been developed for automatically annotating these sequence variations, prioritization of amino acid changes or predicting their significance. These methods that are built for variant prioritization and effect prediction helped to gain insight into how they affect th' gene's regulation and/or function of its protein products (Mooney & Klein, 2002). Each of these methods differ in the type of features they utilize in the data, their approach to evaluate the significance or their prediction mechanism. Among all, the most well-known tools are SIFT (Kumar *et al.*, 2009) and PolyPhen-2 (Adzhubei *et al.*, 2010). Other well-

established methods include SNAP2 (Hecht *et al.*, 2015), PROVEAN (Choi *et al.*, 2012), MutationAssessor (Reva *et al.*, 2011), PANTHER (Thomas *et al.*, 2003) and SNPs3D (Yue *et al.*, 2005).

1.4. Machine-Learning Applications

Machine learning is the study and usage of algorithms and statistical models to provide computers the ability to perform specific tasks without being explicitly programmed. Machine learning applications rely on the ability of the system to learn from the inherent nature of the data and on the inference made using patterns that data holds. The process of learning begins with examining and understanding the data, continues with building appropriate mathematical models to represent it. Then the model aims to make predictions on new data and answers the questions asked. Machine learning applications are used in a wide range of fields such as finance, computer vision, image recognition, data analysis and robotics.

Machine learning applications are also used in bioinformatics where understanding the underlying mechanisms in biological processes and inferring meanings from biological data matters. Many of the methods that are developed to characterize mutations and prioritize variants use machine learning models, as well. They provide robust, fast and generalized interpretations for analysis of DNA variations and their effects on the biological processes. Machine learning methods implement a number of statistical and computational algorithms and make predictions on biological data by extracting relevant information mathematically. Machine learning methods identify the data via its features. Relevance or redundancy of these features affect the performance of the algorithm. They can be categorized as supervised learning and unsupervised learning. In unsupervised models, the system is not provided with labels for the output data, rather it is expected to infer natural structure present within data itself. Cluster analysis (e.g. hierarchical clustering and k-means clustering) and principal component analysis can be given as examples to unsupervised learning methods. On the other hand, in supervised methods, the system is provided with the output labels that gives a prior knowledge for partition of data. One of the major types of supervised learning is classification. In classification, the aim is to predict a categorical response value or class label which is introduced to the system in the training phase. Separation can be binary classification where the class labels only have two categories or multiclass classification where there are more than two classes. The other major type of supervised learning is regression where the response value or prediction output is a continuous value. Figure 1.2 is added to show examples to supervised methods which include regression-based methods (e.g. logistic regression, generalized linear models), Naïve Bayes, Support Vector Machines (SVM), Artificial Neural Networks (ANN) and tree-based methods (e.g., decision trees and random forests).

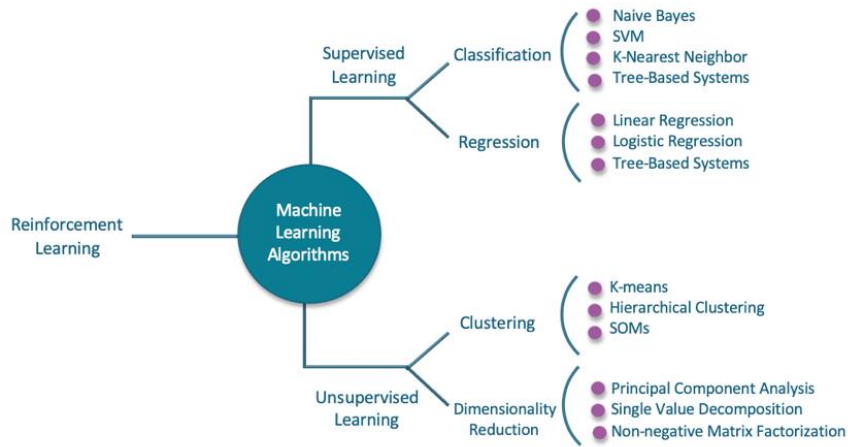


Figure 1.2. Some commonly used machine learning algorithms are shown. Scope of this study includes supervised learning only.

A typical machine learning approach to any sort of problem includes data pre-processing, splitting the data into test and training sets, feature selection on training data set, generating the model, running the model and getting the predictions, assessing the accuracy measurements and fine tuning the model (Figure 1.3). The model is then used to obtain predictions for new coming data.

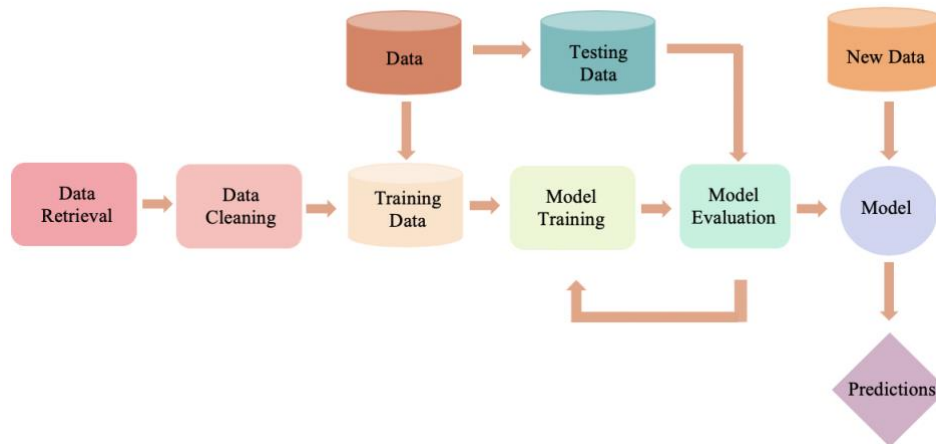


Figure 1.3. An example workflow for machine learning methods.

1.4.1. Unsupervised Methods

One of the well-known and widely used examples of unsupervised learning is clustering. Clustering is a learning method where similar data is partitioned into groups in order to reveal hidden internal structure of data. Since it is an unsupervised learning method cluster

analysis do not use any labels, rather it creates groups from scratch. It aims to partition data in a way that data points are very similar within groups, but as distinct as possible in different groups (Müller & Guido, 2015). There is no absolute best criterion on the best clusters produced as the nature of the question defines the homogeneity and separation of clusters.

One of the widely used clustering methods is k-means clustering where data is clustered into k mutually exclusive groups. It works by iteratively assigning data points to clusters where the distance between data points and the mean or median location of the cluster is minimum. The algorithm terminates when the assignment of clusters remains unchanged (Müller & Guido, 2016; Shalev-Shwartz & Ben-David, 2014).

Another widely used clustering method is hierarchical clustering where data is grouped into multi-level tree or dendrogram. Hierarchical clustering is also an example of agglomerative clustering which refers to a type of algorithm that works by merging similar clusters from their single point clusters until some stopping criterion is met (Müller & Guido, 2015; Shalev-Shwartz & Ben-David, 2013)

DBSCAN is also a clustering algorithm that works by identifying points that are in dense regions of the feature space, where many data points are close together. It does not require a pre-set cluster number, and it can capture clusters of complex shapes. It also can identify points that are not part of any cluster (Müller & Guido, 2015; Shalev-Shwartz & Ben-David, 2013).

1.4.2. Supervised Methods

Examples for supervised methods include a wide range of algorithms. A very simple example is k-nearest neighbors where the algorithm finds the closest data points in the training set (nearest neighbor). Then the data point is grouped with similar data points whose distances are also closest to the same point. All of them is classified based on the class label of its nearest neighbor (Müller & Guido, 2015). New data points are assigned to one of these groups.

Another example for supervised learning methods is simple linear regression. In linear regression algorithm, the aim is to find the parameters within a function that minimizes the mean squared error (the sum of the squared differences) between predictions and true target values (Müller & Guido, 2015).

Naïve Bayes classifiers are also classified as supervised methods as they make use of pre-defined labels. In Naïve Bayes method, a family of classifiers based on Bayes' Theorem where in feature statistics is calculated for each one individually. It assumes independence among them meaning that the presence of a particular feature in a class is unrelated to the presence of any other feature (Müller & Guido, 2015).

Another category of widely used supervised learning algorithms include tree-based methods. Decision trees where a decision is reached by splitting the data into nodes based on rules generated during the classification (James *et al.*, 2000; Mitchell, 1999), and random forests where multiple decision trees are generated for the same problem in order to obtain a consensus result (Breiman, 2001) are representatives of this class. Tree-based algorithms predict the label of an instance by following if/else statements until a final decision is reached. Final decision can be a continuous value or some category that makes decision tree a suitable method for both classification and regression tasks (Kotsiantis, 2013)

Support Vector Machines (SVM) (Cortes & Vapnik, 1995; Cristianini & Shawe-Taylor, 2000) are also an example of supervised learning algorithms where the data is labelled as a result of a separation by an optimal hyperplane that is used as the discriminative classifier. SVMs use linear models to implement non-linear regression by mapping the input space to a higher dimensional feature space using kernel functions. In other words, it aims to find an N dimensional (N being the number of features) hyperplane that separates data in the best possible way. If the data is two dimensional, this hyperplane is a line that divides a planar surface to partition data. Among all possible planes, the one that maximizes the margin, the minimum distance between data points of both classes, is selected as the classifier. SVMs can be used for both classification and regression tasks (Müller & Guido, 2015; Shalev-Shwartz & Ben-David, 2013).

Last but not least, Artificial Neural Networks (ANN) can be given as an example algorithm for supervised learning methods. ANNs are computational systems that are inspired by human brain and its capability to deliver information through its neurons - nodes in computational terms. ANNs learn by examining labelled examples and automatically generating features, thus they are supervised learning techniques. ANNs take inputs and process it within some hidden layers by some non-linear function of the sum of its inputs (Schmidhuber, 2015).

1.5. Aim of The Study

It has been shown that genomic variations have the capacity to cause deleterious effects on protein functionality, and because of that, they are considered to be responsible for the occurrence of many different genetic diseases. These variations are important to characterize, as they can help to understand the underlying mechanisms of these changes, and further down the road, may help to develop treatments to cure these diseases by easing clinical interpretations. However, the current rate of manually annotating the reported nsSNPs cannot catch up with the rate of producing new sequencing data. To aid this process, automated computational approaches are being developed and applied on the unknown data.

In this study, we aimed to develop a methodology to accurately predict the effect of mutations on protein functionality, as either deleterious or neutral, by utilizing protein and

mutation specific features, and by examining the contribution of different sets of structure and sequence-based features to the prediction performance. Objectives of the study can be listed as retrieving relevant data from different data sources, combining this data into a final clean data set, generating mutation feature vectors to be used as input to the prediction model, constructing and training a variant effect prediction model using random forest classifier, evaluating the importance of considered feature types and optimizing the final model to assess the performance for further use with newly introduced data. In our study, we assumed that quantizing the nature of the mutation, i.e. the change from wild type to mutant residue in terms of physicochemical properties such as polarity or volume, as well as mutation's correspondence with the functionally annotated regions on the protein, or the proximity of the mutation to these functional regions in the 3-D space, play an important role on the observed effect of that particular mutation. Thus, incorporating structure-based and sequence-based features together into the prediction model is expected to result in the accurate binary prediction of the effect of variations as deleterious or neutral.

In this study, we proposed a new methodology to collect and organize the information related to the effects of nsSNPs at the amino acid sequence level from various biological databases and to utilize this information in a supervised machine-learning based system to predict the function disrupting capacities of mutations with unknown consequences. Our hypothesis was that, the incorporation of the information about the correspondence between the mutation and the site specific functional features (e.g. DNA binding regions, active sites, disulfide bridge forming residues and etc.), and the information regarding spatial proximity in-between when there is no direct correspondence, will be informative in terms of estimating the function disrupting capacity of the reported mutation on the respective protein. Thus, to test our hypothesis, we constructed 68-dimensional mutation feature vectors for each data point using information from protein 3-D structures, physicochemical properties of mutations and site-specific functional annotations in the UniProt database and trained a random forest classifier to classify unknown mutations either as deleterious or neutral. In order to confirm the validity of our hypothesis, we checked the performance of different subsets of features presented in the feature vector and made a comparison. Combining structure and sequence information showed better performance than any of them used in a single form. We highlighted the importance of sequence annotations and showed their potential importance in variant effect prediction. We hope that this system will be complementary to the well-known variant effect predictors in the literature and will increase the performance of the state-of-the-art via its incorporation to ensemble based variant effect predictors in the future.

1.6. Overview of the Thesis

In Chapter 1, background information regarding sequence variations; their nature, types and importance is given in order to present a clear picture of the problem. We have also provided a brief description of the potential methodologies that are being used for addressing these problems.

In Chapter 2, we compared traditional and computation-based methods for prediction of sequence variations. We also gave a brief concept-wise explanation on machine learning methods as most of the computational methods use machine learning algorithms for their predictions. Finally, we classified the computational methods for the methodology they employ, input features they utilize and the underlying principles they carry.

In Chapter 3, we explained our methodology from data collection to model building. We elaborated on the sources we gathered the data used in our study, how all the information is incorporated together and finally how our model is built.

In Chapter 4, we showed the analysis of the data and our results upon training our model with different parameters. Models selected as best ones are shown in the results.

And finally, in Chapter 5, we discussed our results and how they could be interpreted in the context of our data selection and features used. We also discussed how these findings may be used and further developed in future studies.



CHAPTER 2

2. LITERATURE REVIEW

2.1. Traditional vs. Predictive Methods on the Study of Effects of SAVs on Proteins

The ability to predict the disease-causing capacity of mutations holds a great importance for early detection of individuals that possess a high risk of carrying a particular disease and promises hope for personalized medicine applications (Kucukkal *et al.*, 2014). Especially after the revelation of the diversity in the human genome, medical research has shifted to a more individual centric approach. This shift in methodology have been facilitated by the amount of SNP data available from research. After the completion of Human Genome Project, decreasing costs of sequencing individual genomes have generated a mass of individual-specific omics data that harbors a wealth of information (Ho *et al.*, 2019). Personalized medicine approaches vastly target mining this data and find differences among individuals and understand each the genetic make-up in an individual basis. DNA mutations are the cause of many human diseases and they are the reason for natural differences among individuals by affecting the structure, function, interactions, and other properties of DNA and expressed proteins. The ability to predict whether a given mutation is deleterious or harmless is of great importance for the early detection of patients with a high risk of developing a particular disease and would pave the way for personalized medicine and diagnostics.

Traditional methods considered single simple mutations as causatives of diseases and tried to associate them with observed phenotypes in order to understand the roots of diseases. Gene mapping is used to relate diseases to genotypes by identifying co-segregating markers among individuals. This was a valid assumption for single-gene or Mendelian diseases; however, it is also found out that most of the diseases carry a complex nature and single mutations cannot explain their causalities. In order to untangle this and find causatives of diseases that cannot be explained by single mutations, another approach, GWAS, had been developed. Genome-wide association studies (GWAS) are used in order to relate phenotypic traits to relevant SNPs for complex diseases (Tawfik & Spruit, 2018). GWA studies are carried out with hundreds of thousands of SNPs that are believed to capture the variation that causes certain diseases (Hirschhorn & Daly, 2005; W. Y. S.

Wang *et al.*, 2005). It relies on the idea that genetic variations with alleles that are common in the population can shed light to the heritability of common diseases (Reich & Lander, 2001). Although GWA studies helped the discovery of various SNPs that are associated with complex diseases such as Crohn's disease, type II diabetes, cardiovascular and autoimmune diseases as well as cancer (Burton *et al.*, 2007; Easton & Eeles, 2008; Lettre & Rioux, 2008; McCarthy *et al.*, 2008; Mohlke *et al.*, 2008; Samani *et al.*, 2007), researches show that GWA studies actually are not as powerful as expected in terms of associating variants to phenotypes. Identified SNPs usually do not have a direct effect on the condition under study or have a small fraction in the causality (Frazer *et al.*, 2009). However, they are located in the vicinity of the real causal SNPs since genotyped SNPs are chosen over the whole genome to increase the coverage (Ahmed *et al.*, 2009; Amundadottir *et al.*, 2009). This linkage disequilibrium situation can affect the interpretation of SNPs (Goddard *et al.*, 2009). In addition to possible over estimations, GWA studies may also miss out some well-known genetic risk factors as the phenotypes are mostly considered complex and multiple factors are expected to contribute their presence. Together with linkage disequilibrium, contribution of multiple genetic and environmental factors and also the fact that number of SNPs are very higher than the number of samples (high-dimension, low-sample-size problem) introduce other challenges to GWA studies (Szymczak *et al.*, 2009). GWA studies aim to find causal factors of diseases in genotype level. In other words, they try to link mutations to their induced diseases. However, given the complexity and challenges introduced throughout the experimentation process, one important aspect unfolds which is prioritizing the mutations to be tested in GWA studies. In addition to GWA studies, in the clinical part of understanding disease causalities, unraveling the potential effects of mutations after genetic testing or similar practices carries a great importance. Thus, instead of only relying on experimental results that require a tedious and time-consuming work to link mutations to diseases directly, implementing computational methods to model possible effects of mutations from the mutation specific or protein specific information is very significant.

Given the challenges in experimental strategies that are just mentioned above, quantity of available variation data, the importance of readily characterizing variants without long experimentation processes and the need for prioritization of mutations for other downstream experiments, developing computational predictive methods have been a focus for variant analysis. Computational methods are not accurate enough to replace wet-lab experiments however, they may help in selecting and prioritizing a small number of likely and tractable candidates from the pool of available data. Parametric statistical models accompanying traditional experimental approaches have limitations in terms of the analysis of the data as they have limited power for modeling high dimensional, non-linear samples. However, most of the times biological data has a lot of dimensions and inherently very complex; and it encompasses a rich pool of information to extract meaningful relationships and infer patterns. Implementation of effective data mining strategies and machine learning algorithms on biological data have increased the ability to predict possible causalities (Ho *et al.*, 2019). This comes from the ability of machine learning algorithms to handle multi-dimensional data. Methods utilizing machine learning

algorithms have shown to be explanatory on such high dimensional data that is otherwise very hard to explain.

2.2. Variant Effect Assessment Using Predictive Approaches

Given the magnitude and complexity of biological data, in order to catch the pace of improvements in science, predictive approaches come into play when it comes to evaluating the effects of sequence variations. Experimental approaches are more accurate, however; they fail to cover all possible mutation space. This fact shows us the importance of using predictive approaches for the purpose of variant effect assessment. Predictive approaches include machine learning and statistical methods. Their ability to interpret complex relations made machine-learning based and statistical methods a focus on evaluating variants and many other problems, including variant analysis, understanding biological data, inferring disease relations, revealing network patterns, making risk predictions or to making a more accurate diagnosis for certain diseases (Okser *et al.*, 2014; Singh & Samavedham, 2015; Szymczak *et al.*, 2009; Wei *et al.*, 2009; Worachartcheewan *et al.*, 2015). Table 2.1 provides a brief summary on the well-known methods.

2.3. Classification of Variant Effect Prediction Methods Based on the Modelling Approach

2.3.1. Probabilistic and Statistical Approaches

Models developed to predict the effects of amino acid mutations can be grouped according to the approaches they take in terms of the way they make their predictions. One approach takes probabilistic or statistical models. In these methods, predictions and the performance of the method is calculated through a set of mathematical and probability calculations.

A very widely accepted method SIFT (Ng & Henikoff, 2001) makes probability calculations in order to obtain its predictions. Workflow of SIFT can be seen in Figure 2.1. In the methodology it follows, after searching and aligning related sequences, SIFT calculates normalized probabilities that an amino acid is tolerated in a protein sequence conditional on the most frequent amino acid being tolerated. It calculates these values for all possible mutations at each position from the alignment. If normalized probabilities are less than a cutoff, which is 0.05 by default, mutations are predicted to be deleterious; otherwise they are predicted to be tolerated. Probabilities are calculated through a Position Specific Scoring Matrix (PSSM) (Gribskov *et al.*, 1987) that was constructed as a result of the multiple sequence alignment generated. A PSSM here is a $l \times 20$ matrix where l is the length of the protein sequence. Each matrix entry p_{ca} is the probability of occurrence for amino acid a , at position c . Range for c goes up to l , and a represents any one of 20 amino acids.

Table 2.1. Summary of some of the well-known methods for variant effect prediction.

Method	Features used	Prediction Mechanism & Algorithm	Input	Output
SIFT	Sequence-conservation	PSSM scores with Dirichlet priors	Protein sequence and substitution, MSA and substitution dbSNP ID, protein ID	Score ranging from 0 to 1 [0: damaging, 1: neutral]
Polyphen-2	Sequence-conservation & structural information	Naive Bayes	Protein sequence and variation, dbSNP ID, Protein ID and substitution	Score ranges from 0 to a R ⁺ [higher scores imply damaging capacity]
SNPs3D	Sequence-conservation & structural information	SVM	dbSNP ID, Protein accession number, Literature search, Gene ontology term	SVM score [R ⁺ : neutral, R ⁻ : deleterious] [High Confidence > 0.5]
CanPredict	Sequence conservation & GO annotations	Random Forest	Protein ID and variation, Protein sequence variation	4 3 categories [Likely cancer, likely non-cancer or not determined]
PMUT	Sequence conservation & predicted physico-chemical properties	Neural Networks	Protein ID, Protein sequence, Multiple Sequence Alignment	Score ranging from 0 to 1 [Lower scores implying neutral cases, higher scores implying damage]
SNAP	Sequence-conservation & sequence-derived structural information	Neural Networks	Protein sequence	Score ranges from -100 (strong neutral prediction) to +100 (strong effect prediction)
SNPs&GO	Sequence-conservation & structural information	SVM	Protein sequence and GO Terms and substitution, Protein ID and GO Terms and substitution	Binary prediction (disease or neutral) and reliability index
MutPred2	Sequence-conservation & structural information	Neural Networks	Protein sequence and substitution	Score reflecting the probability of pathogenicity
MutationAssessor	Sequence-conservation	Statistical analysis	Protein ID and substitution	Functional impact score

Following formula is used for probability calculation

$$p_{ca} = \frac{N_c}{N_c + B_c} * g_{ca} + \frac{B_c}{N_c + B_c} * f_{ca} \quad (1)$$

where N_c is the total number of sequences in the multiple sequence alignment and g_{ca} is the sequence-weighted frequency that amino acid a appears at position c in the alignment (Henikoff & Henikoff, 1992). f_{ca} represents the pseudocounts that is calculated from Dirichlet mixture (Sjolander *et al.*, 1996) and B_c is the total number of pseudocounts. Pseudocounts are added because the multiple sequence alignment does not represent all similar sequences since observed sequences are taken from a database search and they compensate for the space of all sequences (Henikoff & Henikoff, 1992). In case there are gaps in the alignment, the frequency of observing a gap in the position of interest is used to increment the count g_{ca} for each amino acid by 1/20 of gap frequency.

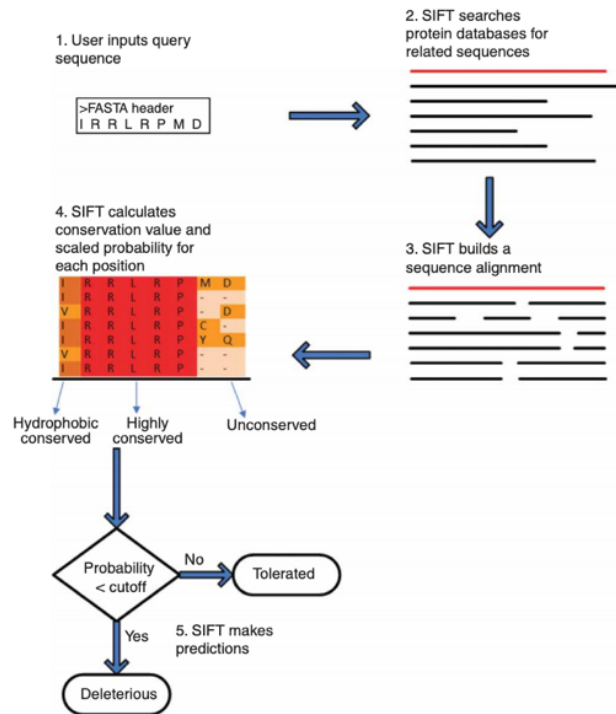


Figure 2.1. Workflow of SIFT. Reprinted from ‘Predicting the effects of coding non-synonymous variants on protein function using the SIFT algorithm’ by P. Kumar, S. Henikoff & P.C. Ng, 2009, Nature Protocols

Another method that makes use of probability is MAPP (Stone & Sidow, 2005). MAPP calculates quantitative scales for six physicochemical properties derived from sequence alignment and uses them to build weighted matrices to capture relevant information. After obtaining sequence alignment profile, it creates a vector that represents the occurrence of each amino acid throughout the alignment. Then, it creates a weighted matrix of physicochemical properties in which all the values are normalized to measure these properties in compatible units. Using this matrix, a 6 x 20 summary matrix is created that captures column means for each property. Later on, deviation of an amino acid from the alignment columns is calculated for each property and stored in a 6 x 1 vector of deviations. Finally, a correlation matrix is built for each property with respect to other properties. Using these matrices in a series of mathematical formulas, MAPP devises a probability-based rule and impact score thresholds are calculated for each mutation. Above this threshold, variants are predicted to be deleterious; and below they are predicted to be neutral. Seven analysis steps of MAPP are shown in Figure 2.2.

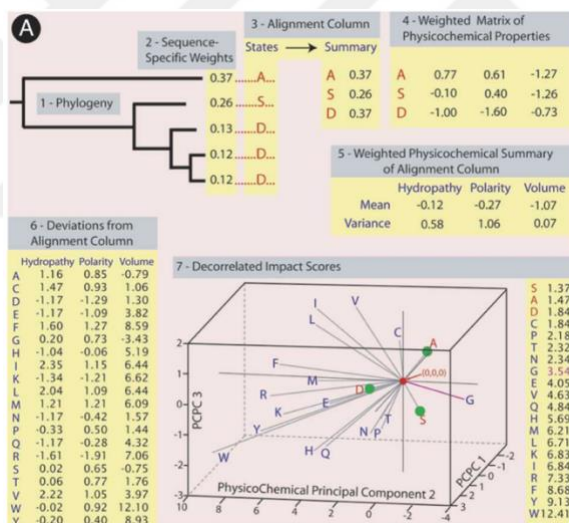


Figure 2.2. MAPP analysis steps. Reprinted from ‘*Physicochemical constraint violation by missense mutations mediates impairment of protein function and disease severity*’ by E. A. Stone, & A. Sidow, 2005, Genome Research.

There are a number of other methods that uses statistical or probabilistic analysis in different forms (Chasman & Adams, 2001; Lau & Chasman, 2004; Mathe *et al.*, 2006; Saunders & Baker, 2002; Verzilli *et al.*, 2005). For example, Verzilli *et al.* developed a hierarchical Bayesian multivariate adaptive regression spline (BMARS) model and tested their predictive performance by using data from mutagenesis experiments on lac repressor and lysozyme. Their methods showed to yield lower out-of-sample misclassification rates compared to other methods tested. In another example, Mathe *et al.* (2006) uses a co-occurrence analysis that calculates the likelihood of a variant being deleterious versus neutral. They test their method with a specific range of data that only includes BRCA1 variation.

2.3.2. Machine-Learning Approaches

Another approach to study the impact of single amino acid variation includes the use of machine learning models. These models predict the effects of mutations using known data and give an outcome with varying prediction confidences. Depending on the nature of the problem, different machine learning algorithms can be used such as support vector machines (SVMs), tree-based algorithms, neural networks and Naïve Bayes.

SVMs are widely used to obtain predictions of disease-causing capacities of mutations as they can extract information from multi-dimensional data that is very hard to handle with other methods. Yue *et al.* (2005) employed SVMs in their method SNPs3D to partition the 15-dimensional feature space into two classes as disease-causing or neutral. They have used a radial based kernel to allow for complex surface topology to accommodate data in a better way. In another method Yue & Moulton (2006) used SVMs this time to make predictions on features obtained from sequence only. Using a different kernel type, linear kernel, that fits the needs of the nature of features, they have trained their SVM model with five parameters mentioned above. For both models, they assigned weights to datasets in order to have them equally contribute to determination of partition space. Krishnan & Westhead (2003) has also tested an SVM model in order to distinguish neutral variants. In their paper, they have compared results from SVMs to results from decision trees and found out that SVM method shows a better generalized result for more realistic cases and it is less susceptible to protein-specific effects in the small learning set associated with a single protein than the decision tree. SNPs&GO (Calabrese *et al.*, 2009) is another method that uses SVMs. An input vector of 52 values are fed into the classifier to obtain binary predictions. Capriotti *et al.* (2005), Tian *et al.* (2007), Bao & Cui (2005), Karchin *et al.* (2005), Kulkarni *et al.* (2008) are other methods developed using SVMs.

Another widely used machine learning method for making predictions is tree-based methods. Decision Trees and random forests are implemented in variety of studies given their practicality in interpreting the outcome and relatively simplicity of underlying mechanism. When comparing decision trees to SVMs in their paper, Krishnan & Westhead (2003) observed that decision trees are able to provide predictions with significantly lower error rates for certain data sets, however they are more susceptible to learning protein specific rules, making them error prone in heterogenous datasets. Random Forests are another type of tree-based systems where multiple trees are generated to obtain a consensus prediction. Since it relies on different separation criteria in each tree, it is more robust compared to a single decision tree. Bao & Cui (2005), employed a random forest model, along with an SVM model to compare two models like Krishnan & Westhead (2003). They built 1000 trees and showed that random forest was superior to SVM method in their analysis. Another method, Cancer-specific High-throughput Annotation of Somatic Mutations (CHASM) (Carter *et al.*, 2009) have been developed by training a random forest classifier with the aim of discriminating between driver missense mutations and synthetically generated passenger missense mutations. 49 predictive features are used to train the classifier. Can-Predict (Kaminker *et al.*, 2007) is also a well-assessed method that uses random forests to predict cancer-associated mutations.

In Can-Predict, the classifier is trained on pathogenic mutation obtained from Catalogue of Somatic Mutations in Cancer (COSMIC) (Forbes *et al.*, 2017) and neutral mutations from dbSNP (Sherry, 2001) databases. Instead of giving prediction using protein features, Can-Predict measures the impact of changes using three metrics; SIFT scores (Ng & Henikoff, 2001), Pfam-based LogR.E-value metric (Clifford *et al.*, 2004) that predicts the impact of variation by characterizing the difference in fit of a wild-type version of the protein to a particular Pfam model and Gene Ontology Similarity Score (GOSS) that provides a measure of closeness of the gene in which the variant is found to other known cancer-causing genes. Three outcomes of these methods are fed into a random forest classifier to obtain final prediction on the consequences of mutations. An interactive webserver that does the same task with other methods is MutD (Wainreb *et al.*, 2010). MutD distinguishes itself in that protein specific structural and functional information can be added while making predictions. MutD also uses a random forest-based classifier that is generated from 650 trees.

Neural Networks is another important machine-learning algorithm that has been implemented in various methods with the goal of prediction of disease-capacity of mutations. One well-known example is SNAP (Bromberg & Rost, 2007; Bromberg *et al.*, 2008). Deriving features from sequence alone, SNAP uses a standard feed-forward neural network model with an input vector of 195 nodes, along with 50 hidden units. Features in the input vector are both categorical and continuous values depending on the feature represented. For example, changes in hydrophobicity, charge, size are represented by the severity of the change; an input of 100 representing a change from a positive charged residue to a negative charged residue, an input of 50 representing a change from positive residue to a neutral residue and an input of 0 representing a change from a positive charge residue to another positively charged residue. Some other features such as the presence of buried charge or an introduction of proline into an alpha helix are represented by a single binary node; with values of either 0 or 100, while predicted flexibility, SIFT and PolyPhen-2 scores are incorporated as their actual value. PMut (Carles Ferrer-Costa *et al.*, 2005) also uses neural networks as a way of obtaining predictions. It uses two neural network architectures, a large one that is used as the default one that includes 1 hidden layer, 20 nodes and 15 descriptors (C. Ferrer-Costa *et al.*, 2004; Carles Ferrer-Costa *et al.*, 2002) and a small one that includes 20 nodes and no hidden layer with 3 parameters. After training the classifier, a pathogenicity index ranging from 0 to 1 (values > 0.5 implying pathogenicity) and a confidence index ranging from 0 (low) to 9 (high) is reported. Capriotti *et al.* (2004) also employed neural networks in their method and by that they tried to reveal possible effects of mutation on protein's stability, and through that understanding the capacity of mutation to cause aberrations in the protein. Problem again addressed via implementing a use standard feed-forward neural networks, with the back-propagation algorithm as a learning procedure. Three different architectures are developed with increasing complexity to understand features that contribute the most to destabilization of the protein. N1 included 22 input neurons, 2 of accounts for temperature and the pH at which the stability of the mutated protein was measured, while the other 20 nodes represent 20 amino acids. Residue corresponding to the wild type residue is set to -1, mutant residue is set to 1 and all other are set to 0. In N2 architecture, one more node

that represents the relative accessibility of the mutated residue computed with the DSSP program (Kabsch & Sander, 1983) is added. Finally, in N3 architecture, 20 more input neurons (43 in total) are used to characterize 3D residue environment for each of 20 amino acids. Distance values are calculated for amino acids that are found within a certain radius. As expected, N3 performed better than the other two simpler methods.

Final method to mention in this section that is used for the same purpose is Naïve Bayes. A well-known method PolyPhen-2 uses a probabilistic classifier and calculates the Naïve Bayes posterior probability that a given mutation is damaging for its prediction mechanism. A mutation is classified as benign, possibly damaging or probably damaging. Another example to methods using Naïve Bayes approach is MutationTaster (Schwarz *et al.*, 2014, 2010). Similar to PolyPhen-2 it predicts the disease potential of the mutation using a Bayesian classifier to calculate probabilities for each case; namely disease. Naïve Bayes have some advantages in the sense that both discrete and continuous valued features can be incorporated into Naïve Bayes model. Additionally, it is a rather simple method as it does not contain any parameter to fine tune except for representing factored probabilities and smoothing, which is done by Laplace estimators.

Each method mentioned above uses a different set of features, takes a different approach to give answers to the same problem using different algorithms and reasoning. All of them are applicable to certain data with certain limitations. This makes it inevitable to conduct a thorough investigation before using any of them for prioritization of variants. Despite the abundance of computational models that addressed the question of disease or functional effect prediction of mutations, the challenge resides in the biological interpretations of these effects. All of the methods above, predicts the effect of variations to some degree and helps to prioritize variants for clinical testing, drug development or personalized treatment options. However, they do not provide a direct association do diseases in most of the cases. Even though these methods can help in selecting certain variants over others, prevention and treatment strategies without interpreting the results in the context of human biology is not possible.

2.4. Classification of Variant Effect Prediction Methods Based on the Input Features They Utilize

2.4.1. Methods That Use Sequence Information

Researches show that mutations that occur at evolutionarily conserved regions of the genome often have malign effects on the protein's structure or function (Kumar *et al.*, 2009). These conserved regions are shown to possess important roles in protein's function as they have been selected to survive through generations. As a result, mutations occurring in these regions are expected to have more severe effects than those occurring in non-conserved regions. Conserved sequences in a protein's structure are characterized through multiple sequence alignment of the protein across multiple organisms; either via alignment of consensus sequence of the same protein across organisms or via alignment

with its homologues (Kucukkal *et al.*, 2014). This approach is particularly advantageous because it does not require any structure data. Given the gap between the number of sequences that are being characterized and the number of experimentally assessed structures, it does not seem close to validate structure information for all proteins for which sequence data is available (Schwede, 2013). As a result of this gap, and also due to the availability of sequence data, giving variant effect predictions through sequence information carries a great importance. The scope of methods that use sequence information is expected to cover a wider range of mutations as new sequence data is added to the existing ones (Kumar *et al.*, 2009). As well as being a strength, relying only on sequence has its drawbacks, as well. Since the outcome of these methods are highly dependent on the multiple sequence alignment carried out, quality and depth of the alignment can affect the predictions (Kucukkal *et al.*, 2014). There are a number of methods and servers that use sequence-derived information as their methodology including Sorting Intolerant from Tolerant (SIFT) (Ng & Henikoff, 2001), Alignment Grantham-Variation, Grantham-Deviation (Align-GVGD) (Tavtigian *et al.*, 2008), Mutation Assessor (Reva *et al.*, 2011) and Multivariate Analysis of Protein Polymorphism (MAPP) (Stone & Sidow, 2005).

A very well-known tool SIFT (Ng & Henikoff, 2001), for example incorporates position-specific information by considering only the position of mutation on the protein sequence and type of amino acid change in order to classify variants. Based on the generated alignment across selected sequences by SIFT, a normalized probability of a mutation being pathogenic or tolerable is calculated. If this value is less than a predefined cut-off, the mutation is predicted to be deleterious. For example, a single amino acid is observed throughout the alignment, any variant that occurs in that position is predicted to be deleterious; because it is hypothesized that this single amino acid is crucial to the function of the protein. On the other hand, if an amino acid of a certain class, i.e. hydrophobic, polar etc., is observed at a particular position, any change to an amino acid from the same class is predicted to be tolerant, and any change to an amino acid from another class is predicted to be deleterious. One important feature of SIFT is that it calculates a score called median conservation score. This is particularly important because as the outcome of the prediction is highly dependent on the quality of the alignment; the quality of the alignment is dependent on the sequences selected. If the sequences are selected from closely related organisms, they may not show enough diversity and a normally tolerant mutation may be predicted as deleterious. Median conservation score measures the diversity among alignments that helps to assess the reliability of sequences selected (Ng & Henikoff, 2003).

Another method that is developed by Yue & Moulton (2006) also makes use of a similar approach that only considers sequence related information. After constructing alignment profiles, five sequence features have been extracted: probability of accepting the amino acid mutation of interest taken from a generated Position Specific Scoring Matrix (PSSM) (Gribskov *et al.*, 1987) in the same way that was done in the SIFT method, entropy at each position of the alignment that is calculated by Shannon's entropy formula (Shannon, 1948) and summed over the 20 possible amino acids, mean entropy over the entire sequence,

standard deviation of the entropy over all positions and entropy at each position as expressed as Z score. All these features are extracted from sequence alone and used as the input set for classifier. This method showed a higher performance over a structure-based model developed by the same group (Yue *et al.*, 2005).

More examples on the methods that use sequence conservation alone includes Multivariate Analysis of Protein Polymorphism (MAPP) (Stone & Sidow, 2005). In MAPP method, the hypothesis that degree of protein impairment and disease severity might be correlated with the difference between the original and mutated residue's physicochemical properties. Thus, MAPP makes its predictions by merging sequence alignments with the physicochemical characteristics in each position of the protein, based on observed evolutionary variation. It quantifies the physicochemical variation in each column of a multiple sequence alignment and calculates the deviation of candidate amino acid replacements from this variation. The greater the deviation, the higher is the probability that a replacement impairs the function of the protein, and the greater is its predicted effect on the function of the protein (Stone & Sidow, 2005). Physicochemical properties involved includes quantitative scales of hydropathy (Kyte & Doolittle, 1982), polarity (Stryer, 1995), charge (Stryer, 1995), side-chain volume (Zamyatnin, 1972), free energy in helical conformation (Muñoz & Serrano, 1994); and free energy in sheet conformation (Muñoz & Serrano, 1994). MAPP method has shown that physicochemical properties are very important in the degree of impairment and by this contributed to the study of amino acid variants by providing an explanatory mechanism underlying detrimental effects.

One other method A-GVGD (Tavtigian *et al.*, 2008) also uses sequence alignment and with a similar approach to MAPP. This method uses Grantham matrix (Grantham, 1974) scores between the wild type and mutated amino acids are used to score missense mutations against the range of variation present at their position in a multiple sequence alignment. Sequence alignments and Grantham analyses are measures of evolutionary fitness that are indirectly tied to disease susceptibility. Thus, incorporating this feature into the analysis helped to distinguish between neutral and deleterious missense mutations.

Last but not least, another method that uses sequence conservation information is MutationAssessor (Reva *et al.*, 2011). MutationAssessor introduces a measure called functional impact score (FIS) that is calculated between the wild type and mutant amino acids using evolutionary conservation patterns in order to prioritize functional effects of mutations. As a new angle to look at the problem, this method involves the conserved residues among sequences called specificity residues into the analysis. Strong selection patterns across an entire protein family or within protein subfamilies are very likely the result of strong selection that disfavors amino acid residues not consistent with the conservation pattern, no matter their separate contributing factors, such as effects on protein stability or protein-protein interactions be. As a basis of the analysis, this method calculates FIS from two different scores, namely conservation and specificity scores where conservation score takes conservation across the entire family into account, while specificity score considers conservation within subfamily and variation between

subfamilies. Entropy throughout the alignments are calculated for both situations and these values are combined in order to get final values for impact assessment. A schematic representation of the method is given in Figure 2.3, showing derivation of functional impact score from multiple sequence alignments. The score is based on the evolutionary conservation of mutated residues.

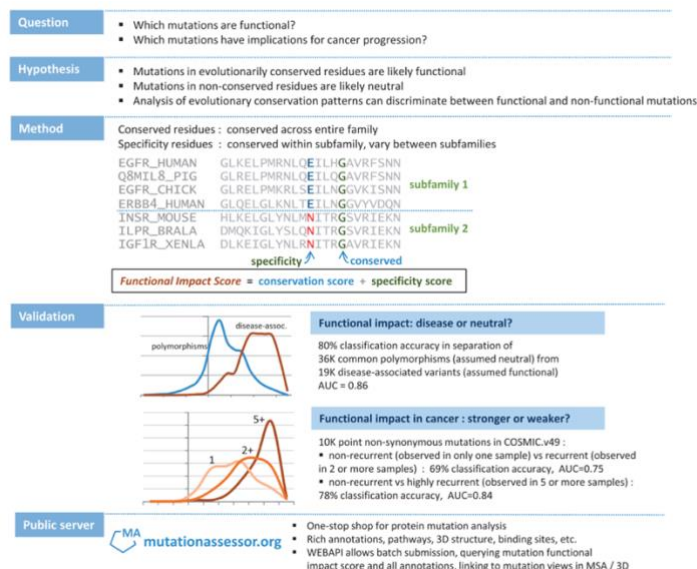


Figure 2.3. Summary of the MutationAssessor method. Reprinted from ‘*Predicting the functional impact of protein mutations: application to cancer genomics*’ by B. Reva et al., 2011, Nucleic Acids Research

2.4.2. Methods That Use Structural Information

There has been a continuous effort and progress in experimentally or theoretically discovering the 3D structures of proteins. Publicly available structure data is deposited in Protein Data Bank (PDB) (“The Protein Data Bank,” 2003). When experimental methods are unable to reveal 3D structure of a protein, modeling approaches are used to predict the structure using existing similar proteins or using ab-initio techniques which utilizes protein’s bio-physical properties. Despite being far from characterizing structure for all proteins whose sequences are available, the models deposited so far has helped research community use structure information in a variety of studies. One of these categories of studies is the prediction of functional or structural effects of variants through 3D structures (Glusman *et al.*, 2017). Although, predicting the functional effects of proteins from sequence conservation helps to characterize variants, it does not elaborate on the mechanism of the change. Mapping mutations to protein’s three-dimensional (3D) structure may shed light into the mechanism by which that mutation disrupts function of the protein (Glusman *et al.*, 2017). There are only few methods that solely use structure information and most of them do not give direct meanings into the disease capacity of

mutations, rather they explain mutation's effects on protein's stability or structure. Level of impairment as a result of the mutation of interest can be inferred from the outcome of these tools. Several tools integrate 3D visualization of proteins and map variants on the protein; however, they do not provide a prediction of their consequences (Cerami *et al.*, 2012; Douville *et al.*, 2013; Mooney & Altman, 2003; Niknafs *et al.*, 2013). Some tools on the other hand, in addition to making visualization on 3D structure possible, allows analysis of variants through structure-derived calculations. STRUM, for example, uses folding energy calculations and the difference of values in this feature upon mutation introduction to give a glimpse of the effect of new variant (Quan *et al.*, 2016). Another such tool is a pipeline called VIPUR that allows automatic interpretation of the effect of the mutation on the protein structure by using Rosetta energy terms (Baugh *et al.*, 2016; Leaver-Fay *et al.*, 2011). A well-known method that uses purely structure information to predict the change in protein stability, folding or protein-protein binding dynamics, not the disease-causing capacity, is FoldX (Schymkowitz *et al.*, 2005). FoldX uses an empirical force field scoring function to calculate the free energy of proteins based on the 3D structure. Depending on the 3D structure alone, FoldX gives its predictions when well-characterized structures are present. Another method, BindProfX, assesses protein-protein binding free-energy changes upon introduction of a variant to the protein (Xiong *et al.*, 2017). Since mutations that affect protein-protein binding regions tend to disrupt the formation of necessary complexes or interactions between proteins, they are considered to be changing the function. Changes in the free energy is associated with the conformation of the protein, thus BindProfX helps to understand roles of disease related mutations associated with protein-protein interactions.

One last method to give an example to methods that use structure information is developed by Yue *et al.* (2005) complementary to another method for the same purpose, however, uses only sequence information. This method makes use of a set of structural calculations such as reduction in hydrophobic area, overpacking, backbone strain, and loss of electrostatic interactions in order to understand the impact of single residue mutations on protein stability. Their model included values for the mutated state and/or values for the wild type state as well as differences for selected features between these two where applicable. Features used include difference in electrostatic energy, overpacking that means the atomic distance between the mutant residue and its nearest neighbor, relative accessible surface area, hydrophobic burial area change and crystallographic temperatures factors as continuous values; introduction of a cavity, introduction of electrostatic repulsion which is characterized by the presence of two residues with two like charged groups within a close atomic contact, presence of a mutation that results in a residue with zero solvent accessibility (buried charge), presence of a mutation that results in a polar group with zero solvent accessibility (buried polar) and breakage of disulfide bond upon mutation occurrence. All these features are derived from structure only.

As seen from the different methodologies used by different tools, the use of structural information varies from method to method. While FoldX uses the 3D atomic coordinates of the protein, last method mentioned by Yue *et al.* (2005) extract structural features that characterize changes in the local environment around a mutated residue.

2.4.3. Ensemble and Consenses Based Methods

Sequence-only or structure-only methods have advantages and disadvantages in different terms. While structure-only methods are applicable to a wide range of proteins whose sequences are available, they fail to reveal underlying mechanisms of action. On the other hand, structure-based methods can provide means to explain mechanism of action, however, they are only applicable to proteins whose 3D structures are characterized which limits its scalability. For this reason, most methods use a combination of structural and sequence features and then formulate a regression problem to predict scalar values, or a classification problem to predict a mutation as probably deleterious or neutral. A number of methods fall into this category including but not limited to PolyPhen-2 (Adzhubei *et al.*, 2010), SNAP (Bromberg & Rost, 2007), SNPs&GO (Calabrese *et al.*, 2009), LS-SNP/PDB (Ryan *et al.*, 2009), SNPeffect (De Baets *et al.*, 2012), MutPred (Li *et al.*, 2009), NetDiseaseSNP (Johansen *et al.*, 2013) and Mutation Taster (Schwarz *et al.*, 2010).

A very well-known and accepted method PolyPhen-2 (Adzhubei *et al.*, 2010) makes use of eight sequence-based and three structure-based predictive features. These features are selected from a larger set of candidate features by an iterative greedy algorithm. Since other features are shown to decrease or not affect the model performance, final analysis is done with eleven remaining features. Majority of these features are calculated as the difference of values between the wild type residue and mutant residue. In order to obtain sequence information, PolyPhen-2 employs clustering and multiple sequence alignment steps of closely related sequences to the query protein. From the alignment generated, PolyPhen-2 calculates profile scores using Position-Specific Independent Counts (PSIC) (Adzhubei *et al.*, 2010) that shows how likely it is for a particular amino acid to occur at a specific position in the protein sequence, given the pattern of amino acid mutations throughout the alignment profile. Another sequence-derived feature is the sequence identity to the closest homologue that carries a different amino acid from the wild-type allele at the mutation site. Congruency of the mutant allele, which means the sequence identity between the analyzed protein and its closest homologue in which this amino acid is observed, is also used as a feature that is calculated through sequence alignment. As for structural features that are derived from 3D structures of query protein, the accessible surface area of the wild-type amino acid residue, the change in the hydrophobic propensity in the form of knowledge-based potential, crystallographic B-factor reflecting conformational mobility of the wild-type amino acid residue are considered. Change in the amino acid volume between wild type and mutant amino acids, and whether the site of the mutation resides within an annotated Pfam (Finn *et al.*, 2014) domain are also used as features for PolyPhen-2 predictions. Using a fine-tuned set of predictive features from a wider range of possible features, PolyPhen-2 performs well compared to many other methods (Adzhubei *et al.*, 2010). Pipeline for PolyPhen-2 method is given Figure 2.4.

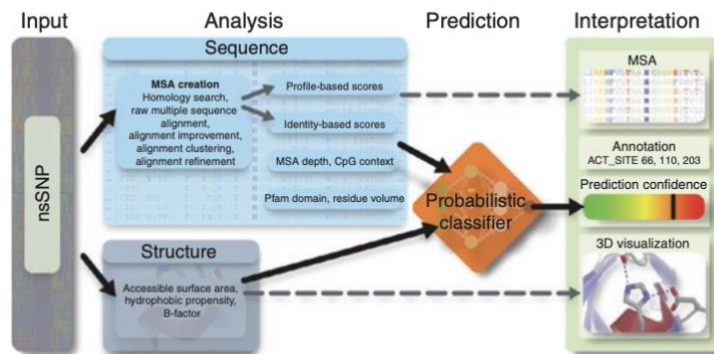


Figure 2.4. PolyPhen-2 pipeline. Reprinted from ‘A method and server for predicting damaging missense mutations’ by I. A. Adzhubei et al., 2010, Nature Protocols

Another powerful method is Screening for Non-Acceptable Polymorphisms or SNAP (Bromberg & Rost, 2007). SNAP requires only sequence information; however, it also calculates structure related features in order to make more powerful predictions. Biochemical features such as hydrophobicity, charge and size changes, the presence of buried charge, change in C β -branching, or an introduction of proline into an alpha-helix are included into the prediction mechanism. From sequence alignment, likelihood of observing certain mutations in a position and the degree to which a residue is conserved in a family of related proteins are calculated. In order to calculate the latter, PSSM (Gribskov *et al.*, 1987) constructions that are derived from PSI-BLAST (Altschul *et al.*, 1997) alignments are used. In addition to these two, PSIC are used to compile compiling position-specific weights that considers the overall level of sequence similarity between the proteins aligned. Above all, some predictions are made in order to incorporate more structure related information. Using (Rost & Sander, 1994; “Proteomics Protoc. Handb.,” 2005) the relative solvent accessibility of each residue is predicted. PROFsec (Rost, 1996; Rost & Sander, 1993; “Proteomics Protoc. Handb.,” 2005) is used to predict secondary structures that the protein of interest holds, and PROFbval (Schlessinger *et al.*, 2006) is employed to predict chain flexibility. Besides biochemical properties of the protein, family information is also taken into consideration that includes presence or absence of domain boundaries in the residue stretch, the model score of this domain, indication of whether the position is conserved and whether the mutant is a better match (according to the BLOSUM62 mutation matrix) to the consensus than the wild type from Pfam domains (Finn *et al.*, 2014). Last but not least, 5 selected SWISS-PROT (Bairoch, 2000) annotations are introduced to the model as input as binary features to explain whether annotation is present at the position of mutation. These mentioned features along with some other binary or continuous features are fed into a classifier to make predictions. Given the variety of features that SNAP includes, it is shown that it outperformed many other similar methods (Bromberg & Rost, 2007).

SNPs&GO (Calabrese *et al.*, 2009) is another example to combined methods. In addition to using structure and sequence derived information, for the first time SNPs&GO integrates Gene Ontology (GO) (Ashburner *et al.*, 2000) score into its predictors. GO database provides tree-structured and controlled vocabularies (ontology) that describe gene products in terms of their associated biological processes, cellular components, and molecular functions. As for other features, SNPs&GO method utilizes the local sequence environment of the mutation of interest, features derived from sequence alignment, prediction data provided by the PANTHER (Thomas *et al.*, 2003) classification system and a functional based log-odds score calculated considering the GO classification. Each of the features are encoded in feature vectors in accordance with the classification method used. For instance, transition from wild type residue to mutant residue is encoded in a 20-dimensional vector where wild type residues are assigned -1, mutant residues are assigned 1 and the rest is assigned 0. Another 20-dimensional vector is created for structural features that included the occurrence of the residue in a radius of 6 Å around the C α atom and relative accessible surface area of the mutated residue in 3D. Four outputs (the predicted probability of deleterious mutation, the frequencies of the wild-type and mutated residue and the number of independent counts) of the PANTHER algorithm are also fed into the prediction in the form of a 4-dimensional vector. Sequence profile features are incorporated as the frequencies of both wild type and mutated residues at the position of interest along with a conservation index. Input schema for SNPs&GO is given in Figure 2.5 where different features and their encoding are shown.

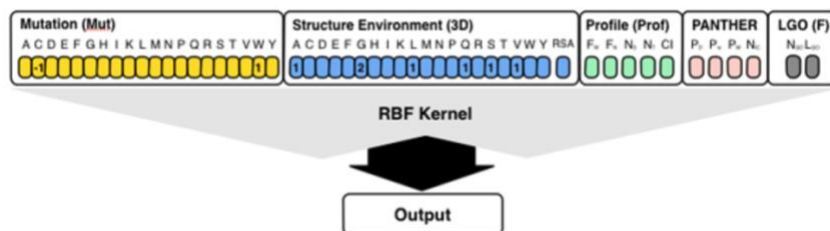


Figure 2.5. SNPs&GO input schema. Reprinted from ‘*Functional Annotations Improve the Predictive Score of Human Disease-Related Mutations in Proteins*’ by R. Calabrese *et al.*, 2009, Human Variation

One more method to review in this section is a method called Disease-Susceptibility-based SAV Phenotype Prediction (SuSPect) (Yates *et al.*, 2014). This is the first method to incorporate network information into the features used in order to make predictions. It has been shown that certain proteins and domains are susceptible to bear disease-associated variant compared to others (Yates & Sternberg, 2013). However, using this information alone can cause wrong predictions due to inherent biases in the data such as the abundance of information about well-studied proteins and lack of information for the remaining ones. It is also shown that susceptibility of proteins and domains to contain disease-associated

variants is also related to other features including the location in the interactome network of the protein or domain and the function of the protein. For this reason, SuSPect incorporates network information together with some structural and sequence-derived features in order to make its predictions. Some of the features calculated for the input vector includes degree centrality in the protein-protein network, number of Universal Protein Resource (The UniProt Consortium, 2019) annotations at the mutation position, score for the wild type and mutant residues from PSSM as well as score difference between two, difference between Pfam HMM emission probabilities for the wild type and mutant residues, a measure for sequence conservation, percentage sequence identity with the first sequence in the MSA to have the mutant amino acid at the mutation position and relative accessible surface area value for the residues. Network related features have shown to improve the performance of the method when compared to evaluations made without including them, showing the promising potential for this approach.

In addition to individual classifiers that uses a certain method of prediction and certain features from the proteins, there also exists consensus classifiers, in other words meta-predictors, which give predictions on the effects of mutations by combining the results of multiple individual classifiers. Owing to the fact that each of the individual classifiers that make up the consensus have different set of data, underlying principles and methodology, a combination of them is likely to give more accurate predictions. Some examples to consensus classifiers include CONDEL (González-Pérez & López-Bigas, 2011), PON-P (Olatubosun *et al.*, 2012), Meta-SNP (Capriotti *et al.*, 2013) and PredictSNP (Bendl *et al.*, 2014). Each of these classifiers have shown to outperform their constituting individual classifiers. PredictSNP, for instance, uses three independent datasets that are constructed in a very elaborative way to eliminate the bias as much as possible. It combines the predictions of MAPP, PhD-SNP, PolyPhen-1, PolyPhen-2, SIFT and SNAP considering their confidence level with their own predictions and calculates its consensus result for the mutations. Figure 2.6 shows the workflow diagram of PredictSNP. Output from this ensemble classifier is shown to outperform all of its individual predictors, thus it is considered to be more accurate. Adding the ability to interpret a wider range of mutations to its improved performance, consensus classifiers are considered as good alternatives to individual classifiers.

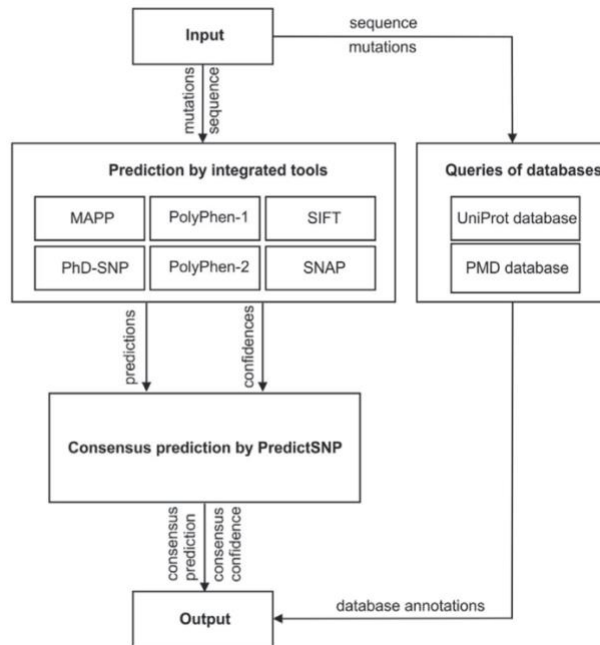


Figure 2.6. Workflow diagram of PredictSNP. Reprinted from ‘*PredictSNP: Robust and Accurate Consensus Classifier for Prediction of Disease-Related Mutations*’ by Bendl et al., 2014, PLoS Computational Biology.

CHAPTER 3

3. MATERIALS & METHOD

This section describes the data sources, procedures for data pre-processing and methodology followed in detail. Three sections describe data retrieval, feature vector construction and development of the classifier respectively.

3.1. Data Retrieval

In order to train a machine-learning classifier in a supervised model, data with known outcome labels are needed to train the model. For this purpose, we have retrieved data from 3 different sources; UniProt (The UniProt Consortium, 2019), ClinVar (Landrum *et al.*, 2018) and Protein Mutant Database (PMD) (Kawabata *et al.*, 1999). These sources include mutations causing either a disease or an impairment in the protein function in common. Terminology to describe non-neutral mutations differ in each of them. For the sake of simplicity, throughout this study ‘deleterious and ‘non-neutral’ terms are used interchangeably in order to describe mutations that cause a disease or an effect on the protein’s structure, stability or function. Likewise, ‘non-disease or ‘neutral’ terms are used interchangeably to describe benign mutations that does not show any effects on the protein’s native function. From each of three data sources, fields describing a ‘*data point*’ for our study are picked out. A data point is characterized by UniProtID, wild type residue, mutant (substitute) residue and position of mutation on UniProt sequence.

3.1.1. *The Universal Protein Knowledgebase (UniProt)*

The Universal Protein Knowledgebase (UniProt) is a database that contains protein sequences, sequence variations, family information and functional annotations as well as other relevant data from cross-referenced external databases. UniProt hosts three different databases, namely UniProt Knowledgebase (UniProtKB) (The UniProt Consortium, 2019), UniProt Reference Clusters (UniRef) (Suzek *et al.*, 2007) and UniProt Archive (UniParc) (Leinonen *et al.*, 2004), each of which holds a different sort of information.

UniProtKB contains functional information of proteins, their description, taxonomic classification, annotations along with many other protein specific features. It is composed of two sections; one section, Swiss-Prot, contains manually annotated proteins. Annotations are derived from literature and evaluated by curators. Another section, TrEMBL, contains automatically annotated information that is derived from computational analysis. Entries in TrEMBL are added to Swiss-Prot after curator evaluation. Currently Swiss-Prot contains 561,568 entries while TrEMBL contains 179,250,561 entries. Another UniProt database UniParc, contains most of the publicly available protein sequences in the world. It aims to decrease redundancy that is caused by multiple presence of the same protein in different databases or within the same database. It stores each sequence only once and updates it when new data is available. Other database UniRef contains clustered sets of protein sequences from both UniProtKB and UniParc with different sequence similarities. Sequences and sub-fragments are merged together in order to store sequences more effectively. From UniProt, we have retrieved sequence variations listed for human proteins (HUMSAVAR, release 2019_01) (Yip *et al.*, 2008). HUMSAVAR file contains all missense variants annotated in human UniProtKB/Swiss-Prot entries. 69,580 data points are retrieved from UniProt database.

3.1.2. ClinVar

ClinVar is a freely accessible, public archive that holds the information regarding relationships among human variations and phenotypes, with supporting evidence. It combines variants discovered by clinical laboratories, researchers, expert panels, and others as well as their interpretations and share collected data for further use by the science community. The database holds 600,000 submitted records from 1,000 submitters, representing 430,000 unique variants (Landrum & Kattman, 2018). By combining different sources together. ClinVar allows comparison of interpretations, providing transparency into the concordance or discordance of interpretations. It also allows users to access supporting evidence for a better evaluation and comparison of the variants. Variant submissions come either from clinical providers who provide their own interpretation of the variant or from groups that primarily provide phenotypic information from patients. ClinVar processes submitted clinical data and presents in an understandable way for anyone who wants to use this data. ClinVar data is available in FTP site where a range of information from genomic coordinates to variant summaries are stored. From FTP archive of ClinVar, variant summary data is retrieved for variant effect analysis. Complete report includes all variants at a location on the genome for which data have been submitted to ClinVar. May 2019 release of variant summary file is used for the retrieval of 59,375 data points from ClinVar.

3.1.3. Protein Mutant Database (PMD)

Protein Mutant Database (PMD) contains manually curated information regarding a variation and its consequences from scientific papers. Each entry corresponds to an article that describes the related experiment for that entry and contains several or a number of

protein mutants. Being a manually curated database of variants, PMD carries reliable variation data. Mutation's effect on protein structure, stability or function in comparison with the wild-type protein are described as the severity of effect, including loss of activity. [++], [+] signs denoting an increase in the activity/stability, [--], [-] signs denoting a decrease in the activity/stability, [0] sign denoting complete loss of function and [=] sign denoting same behavior are used to describe an observed effect in PMD. PMD covers over 81,000 mutants including artificial as well as natural mutants of various proteins extracted from about 10,000 articles. It includes all proteins except members of the globin and immunoglobulin families. PMD uses keywords to describe and summarize the relevant information. "CHANGE" keyword refers to the position and kind of mutations, such as amino acid substitution, insertion and deletion. Other keywords such as "FUNCTION", "STRUCTURE", "STABILITY" describes the observed effect on the protein. Data is retrieved on April 2019 from PMD, and data retrieval resulted in 51,296 data points.

3.2. Feature Vector Construction

After constructing the final data set, features that will be used by the model are added to create the final feature vector. This section describes the data sources consulted to retrieve relevant information.

3.2.1. Domain Data

Domains are conserved functional and structures regions present in the proteins that are responsible for a particular function. They are distinct entities that evolve independently from the rest of the protein. The presence of a domain can define the role of protein in biological processes.

InterPro (Finn *et al.*, 2017) is a database that contains information about protein families, domains and functional site annotations. InterPro classifies proteins and performs functional analysis on them using predictive models from member databases. These models are built from already available protein families or domains and used to annotate new entries. UniprotKB proteins are matched with models to provide annotations. Interpro entries are classified into different classes such as families, domains, repeats depending on their signature types. These classes are then grouped into non-overlapping hierarchies or subclasses that contains related domains in a hierarchical manner. In this study, hierarchy built v64 is used for selecting parent domains.

3.2.2. Structure Data

a. Protein Data Bank (PDB)

PDB (Berman *et al.*, 2002) is a portal that holds experimentally determined 3D structures of proteins, nucleic acids, and complex assemblies. Experimental methods used to

determine structure include different strategies including X-Ray crystallography, NMR and electron microscopy. PDB website currently accommodates around 160,000 experimentally validated structures for a variety of organisms. One protein is matched with multiple structures, each from different experiments, with different resolutions or representing different regions of the protein.

b. SWISS-MODEL

When experimentally characterized 3D structure data is not available in PDB, homology modelled structures from different sources are utilized. Homology modelling is a method to computationally determine 3D structures of proteins where experimental evidence is missing. It relies on the fact that when two protein sequences are similar to each other, their structures are also likely to be similar. In order to obtain a 3D model for a protein, generally following steps are applied. Firstly, similar sequences are found by BLAST (Altschul *et al.*, 1990). Then, sequences are aligned using multiple sequence alignment to align important regions that can give meaningful insights towards the structure of interest. After that, a structure backbone is generated. This step is followed by some fine-tuning steps to model certain regions such as side chains, loops and turns. Finally, the model is optimized using energy minimization and stereochemical evaluation by Ramachandran plot (Ramachandran *et al.*, 1963).

SWISS-MODEL (Waterhouse *et al.*, 2018) is a fully automated server and database for protein modelling. SWISS-MODEL allows its users to build a model for their protein of interest using homology modelling. In addition to that, it deposits data for already modelled sequences. It is regularly updated to accommodate updates from newly included sequences. Current repository holds 1,640,595 models from SWISS-MODEL for UniProtKB targets.

c. MODBASE

As another source of homology modelled structures, a widely used homology modelling database called MODBASE (Pieper *et al.*, 2014) is used. MODBASE is a queryable database that contains annotated protein structure models generated via ModPipe pipeline (Eswar *et al.*, 2003). ModPipe pipeline uses PSI-BLAST (Altschul *et al.*, 1997) for the alignment of similar sequences and MODELLER algorithm (Webb & Sali, 2016) for fold assignment, sequence–structure alignment, model building and model assessment. ModPipe calculates a number of scores for assessing the quality of the model. Among these, ModPipe Quality Score (MPQS) is considered while scoring the models in our study as it is a composite model quality score calculated from other scores including the coverage of the modeled sequence, sequence identity, the fraction of gaps in the alignment, the compactness of the model and various statistical potential Z-scores.

3.2.3. *Physicochemical Properties*

Researches have shown that physicochemical properties are very important in mutation effect predictions (Chasman & Adams, 2001; Saunders & Baker, 2002; Z. Wang & Moulton, 2001; Yue *et al.*, 2005). There are a number of different physicochemical properties and methods that include these properties, each of which is used with different combinations in different studies. One of such properties is Grantham Matrix (Grantham, 1974) which is calculated from three physicochemical value differences between the wild type and substitute residues. Grantham scores are calculated from three physicochemical property values, composition, polarity and molecular volume. These three are selected because they showed the best correlation with protein residue mutation frequencies. Volume and polarity values are taken from published data (Aboderin, 1971; Goldsack & Chalifoux, 1973). Composition is calculated as the ratio between atomic weight of non-carbon atoms in the side chains to the total weight of carbon atoms in the side chain. For instance, a side chain for lysine residue consists of -CCCCNH₂ atoms. Composition value for this residue is calculated as 16/48. For each of these three properties difference between native and mutant amino acid is calculated. After assigning appropriate weights to each property, a distance matrix for 20 amino acids is constructed. Distance scores range from 5 to 215, the closest ones being leucine and isoleucine and the most distant ones being cysteine and tryptophan. As the matrix value increases, the effect of replacement gets more dramatic; because a higher matrix value implies a higher degree of difference between residues. In other words, it is a measure of exchangeability. Calculated Grantham Matrix scores for different mutations can be found in Table 3.1.

3.2.4. *UniProt Sequence Annotations*

In addition to being a comprehensive database for protein sequences and related information, UniProt also provides curated and automated position-specific annotations for Swiss-Prot and TrEMBL entries respectively, in order to describe important regions for protein function. Sequence annotations characterize important regions in protein sequence such as glycosylation, disulfide bonding, binding sites and repeats. Since they are responsible for certain tasks or found in structural key points, mutations occurring in these regions may contribute the overall impairment capacity of mutation of interest.

Table 3.1. Grantham matrix scores between amino acid pairs. Each score is calculated by taking the difference between wild type residue and the substitute residue.

Arg	Leu	Pro	Thr	Ala	Val	Gly	Ile	Phe	Tyr	Cys	His	Gln	Asn	Lys	Asp	Glu	Met	Trp	
110	145	74	58	99	124	56	142	155	144	112	89	68	46	121	65	80	135	177	Ser
	102	103	71	112	96	125	97	97	77	180	29	43	86	26	96	54	91	101	Arg
		98	92	96	32	138	5	22	36	198	99	113	153	107	172	138	15	61	Leu
			38	27	68	42	95	114	110	169	77	76	91	103	108	93	87	147	Pro
				58	69	59	89	103	92	149	47	42	65	78	85	65	81	128	Thr
					64	60	94	113	112	195	86	91	111	106	126	107	84	148	Ala
						109	29	50	55	192	84	96	133	97	152	121	21	88	Val
							135	153	147	159	98	87	80	127	94	98	127	184	Gly
								21	33	198	94	109	149	102	168	134	10	61	Ile
									22	205	100	116	158	102	177	140	28	40	Phe
										194	83	99	143	85	160	122	36	37	Tyr
											174	154	139	202	154	170	196	215	Cys
												24	68	32	81	40	87	115	His
													46	53	61	29	101	130	Gln
														94	23	42	142	174	Asn
															101	56	95	110	Lys
																45	160	181	Asp
																	126	152	Glu
																		67	Met

3.2.5. Retrieving Mutation's Location on the Structure

3.2.5.a. Relative Accessible Surface Area

Residues in the protein structure can be classified into different groups depending on their solvent accessible surface area (SASA). If a single threshold value is selected, they can be categorized as residues residing on the surface of the protein, i.e. surface or exposed residues; and residues residing in the inner and not as-easily accessible parts of the protein, i.e. core or buried residues. However, another information to this categorization can be added via incorporation of interface residues. Interface residues are located in the protein-protein interaction regions and are shown to be important for a variety of aspects regarding protein's stability, specificity and recognition by other proteins (Jayashree *et al.*, 2019). Generally buried residues maintain the structural integrity while surface residues contribute to protein function (Gong *et al.*, 2017). They are involved in interaction with other proteins and ligands and serve as active sites, thus contributes to protein's stability more than core residues (Malleshappa Gowder *et al.*, 2014).

Solvent accessible surface area values are calculated using FreeSASA program (Mitternacht, 2016) that calculates residue level solvent accessible surface area values from atomic coordinates.

3.2.5.b Interface Residues

Interactome Insider (Meyer *et al.*, 2018) is a center that brings structural and genomic information together. Besides allowing exploration of mutations in the context of structural regions such as interface domains, residues, or 3D atomic clusters; it also collects interaction interfaces from various other databases for both co-crystallized structures and homology models. In addition to collecting validated data, it also employs a method called Ensemble Classifier Learning Algorithm to predict interface residues (ECLAIR) to enrich the prediction for the ones that are not present in the searched databases. ECLAIR predicts interfaces from protein interactions in eight organisms including humans. It uses 8 random forest classifiers to predict the interface residues.

In this study, Interactome Insider is used to add the third category, i.e. the interface category, for residues which were already separated as buried and surface.

3.3. Classification Algorithm

In this machine-learning based study, random forest algorithm is implemented to generate models which are used to understand impairment association of missense mutations. This section explains the principles of random forest algorithm and the tools used to obtain models.

3.3.1. Decision Trees

Decision tree is an algorithm that predicts the label of an instance by following if/else statements until a final decision is reached. Final decision can be a continuous value or some category that makes decision tree a suitable method for both classification and regression tasks. In decision trees, data is split in a hierarchical manner using the rules defined by the algorithm used. Three well-known algorithms are ID3 (Quinlan, 1986), CART (Gordon *et al.*, 1984) and C4.5 (Quinlan, 1996). Most algorithms developed for decision trees revolve around a core algorithm that makes a top-down greedy search through the space of possible decision trees. To build a tree, the algorithm searches over all possible attributes and finds the one that is most informative about the target variable. ID3, the first one among these algorithms for instance, uses Shannon's Information Theory (Shannon, 1948) to measure the information content in each node after splitting. It calculates the entropy in each class and measures the purity of nodes. The attribute that provides the largest change in entropy, thus gain in information, is selected for the split. As a successor of ID3, C4.5 provides the ability to include continuous variables by partitioning them into discrete set of intervals. Then it converts the trees into a set of if-then rules. Classification and Regression Trees (CART) on the other hand, improves the algorithm further and includes continuous variables without discretizing them.

Splits aim to make output values as distinct as possible. For this purpose, it searches through all the attributes and find out which attribute splits the purest or best among all other options. This search is repeated at each step of the tree building. A quantitative measure called information gain, that evaluates how well an attribute separates the samples, is calculated. The term ‘entropy’ defines the impurity of samples and used to calculate the total information gain. Mathematically expressed, given a collection S , containing positive and negative labelled samples, entropy of S relative to this binary classification is

$$\text{Entropy}(S) = -p_1 \log_2 p_1 - p_0 \log_2 p_0 \quad (2)$$

where p_1 is the proportion of positive examples of S and p_0 is the proportion of negative examples of S .

Entropy is 0 if a node is perfectly split into two categories and all members belong to the same class; and 1 if it contains the same number of samples from each class. For a classifier where the target value can take up to c different values, the entropy of S is calculated as

$$\text{Entropy}(S) = \sum_{i=1}^c p_i \log_2 p_i \quad (3)$$

where p_i is the proportion of S belonging to class i .

After calculating the impurity of a node, i.e. entropy, information gain can be calculated using these values. Information gain is the expected reduction in entropy caused by partitioning the samples according to the attribute of interest and it is defined as

$$\text{Gain}(S, A) = \text{Entropy}(S) - \sum_{v \in \text{Values}(A)} \frac{|S_v|}{|S|} \text{Entropy}(S_v) \quad (4)$$

where $\text{Values}(A)$ is the set of all possible values for an attribute A , and S_v is the subset of S for which attribute A has value v (i.e. $S_v = \{s \in S / A(s) = v\}$). First term in the Equation (4) is the entropy of the original collection S , and the second term is the expected value of entropy after S is partitioned using attribute A ; in other words, it refers to the sum of entropies for each subset of S_v , weighted by the fraction of examples $\frac{|S_v|}{|S|}$ that belong to S_v . Thus, $\text{Gain}(S, A)$ of Equation (3) means expected reduction in entropy upon split by that attribute. Information gain is the measure used by ID3 algorithm (HSSINA *et al.*, 2014).

One of the major advantages of decision trees includes their easy interpretability to human eye. Decisions can be traced and the logic behind is splits can be observed by examining the trees. In addition to that, decision trees can split binary, categorical, and continuous

predictors in the same model. One drawback of decision trees can be counted as their high variability which causes the decisions and the outcome to change with the changing data (Dasgupta *et al.*, 2011). They may also overfit which means they can memorize the training data and fail on the newly introduced data points.

When using decision trees, or random forest, as a classifier, algorithm takes some hyperparameters in order to optimize the algorithm like any other machine learning algorithm. Many parameters can be listed, however; only the ones that generally have the most effect on the results will be explained here. One of the most important parameters is maximum depth that defines the maximum number of splits that a tree can have. Increases in this parameter can lead to overfitting of the tree as it will make memorization easier. In other words, as the value of this parameter increases, the tree will have splits until each node contains a pure class of samples. One other hyperparameter that needs to be tuned for a better tree is number of trees generated. More trees usually led to better results. The reason for this is because random forest uses a technique called bagging which means selecting a subset of data points rather than using all of them at the same time and another technique called random feature selection that makes the algorithm select a random subset of attributes while building each tree. If the number of data points is large and the number of trees is small, some of these data points may be included once, if anytime. The same logic goes for the attributes, as well. For this reason, as the number of trees increase, the possibility to include all data and all features increases. However, after a certain point, the increase in performance gets lower as the number of trees increase. At this point, performance of the classifier and the cost in computation should be considered before increasing tree number further. Minimum numbers at each split is also another important parameter. It specifies the minimum number of samples that is required at a node to be to be considered for splitting. It controls the overfitting as high numbers prevent the model from learning relations which may be specific to a particular sample. However, if the value is too high this time it may cause under-fitting. As mentioned above, random forests use random feature selection. Thus, in order to specify the number of selected features at each split, another hyperparameter is used. Maximum number of features is given to the model to limit the number of features included. As a thumb-rule, square root of the total number of features is used. Higher values can lead to overfitting; however, the best number changes from one case to another.

3.3.2. *Random Forests*

Random forest method is an extension of decision trees where multiple trees are built instead of one; and an ensemble of these trees are used to make a decision (Breiman, 2001). Randomly selected subset of given size is drawn with replacement from the original data and trees are built with each data set separately (Dasgupta *et al.*, 2011). This is called bagging (Quinlan, 1996). The samples that are left unused are named as out-of-bag (OOB) data which is used to obtain an unbiased estimate of classification error during model training. Another technique that Random Forests use is called random feature selection. By employing this technique, models randomly select features to be used in for splitting

in each model. This decreases the chances of over-fitting. Over-fitting occurs when the model memorizes the data. When it happens, the model gives very good evaluation scores with the training set, however it fails with the new coming data. Randomness in feature selection decreases over-fitting risks. An agreed output from the created forest is then selected as the final decision.

Random forests are ensembles of multiple trees where each tree actually depend on a different set of variables as explained. Mathematically represented, for a p dimensional random vector $X = (X_1, \dots, X_p)$ representing the real-valued input vector and a random variable Y representing the response variable, an unknown distribution $P_{xy}(X, Y)$ is assumed. The aim of the classifier is to find a prediction function $f(x)$ that predicts the response variable Y . The prediction function is determined by a loss function $L(Y, f(X))$ and defined to minimize the expected value of the loss

$$E_{XY}(L(Y, f(X))) \quad (5)$$

where the subscripts denote expectation with respect to the joint distribution of X and Y (Cutler et al., 2012). Loss function is the measure of the closeness between $f(X)$ and Y . Choice of loss function may differ but common functions are squared error loss $L(Y, f(X)) = (Y - f(X))^2$ for regression and zero-one-loss for classification.

$$L(Y, f(X)) = I(Y \neq f(X)) = \begin{cases} 0 & \text{if } Y = f(X) \\ 1 & \text{otherwise.} \end{cases} \quad (6)$$

Ensembl trees construct f in terms of a collection of so-called base-learners $h_1(x), \dots, h_J(x)$ which are then combined to obtain final ensembl prediction. For regression problems, the output of the learners are averaged

$$f(x) = \frac{1}{J} \sum_{j=1}^J h_j(x) \quad (7)$$

while in classification, $f(x)$ is the most frequently predicted class, in other words by the common vote

$$f(x) = \underset{y \in Y}{\operatorname{argmax}} \sum_{j=1}^J I(y = h_j(x)) \quad (8)$$

In random forest models j th base learner is denoted as $h_j(X, \Theta_j)$ where Θ_j is a collection of independent random variables (Cutler *et al.*, 2012). A variety of functions can be used for error minimization and regression or classification analyses, however deeper explanation of these algorithms is out of the scope of this study.

Random Forests are shown to perform well compared to both single decision trees and other machine learning techniques in many occasions (Gunther *et al.*, 2003; Svetnik *et al.*, 2003; Wu *et al.*, 2003). Low-bias nature of the trees in the forest and the random sampling gives random forest method the ability to provide low-bias, low-variance predictions (Dasgupta *et al.*, 2011).

3.3.3. Model Assessment

Every model must be evaluated for its performance before it is applied to a new dataset to see if the model is over-fitting, under-fitting or well-generalized. As data that is used to build the model cannot be used for validation of the model, because this would cause model to memorize labels and perform accidentally well, another strategy must be followed. One such method is called train-test-split in which data is randomly separated into two sets. On the first set, the model is trained and on the remaining set performance is assessed. However, this method has some drawbacks when applied to data with a small size as the diversity may not be represented in the training or test sets. One other method is a statistical resampling procedure called k-fold cross-validation where the entire data is split into k random groups (Müller & Guido, 2015). For each group, train test split method is applied where one portion of the model is left for training and the other portion is used for testing, and the model is run to obtain performance measures. Iterations are continued until the model is run for every group. Figure 3.1 depicts a typical cross-validation for data chunks. In this way, every sub-sample of data will have served as the test set. Average from different round of iterations is given as the overall performance of the model. This method has less bias than the simpler train-test-split method, however selection of k is of major importance as it creates a trade-off between variance and bias. The fold value k should be selected in a way that it splits data to create groups that are statistically representative. Ask gets smaller, the bias of the method gets smaller as well as the difference between the training set and the resampling subsets gets smaller, as well (Kuhn & Johnson, 2013). Empirically validated values for k is either 5 or 10 since they showed moderate results in terms of bias and variance relationship (Ziegler, 2016). When k is selected to be the same as the size of the dataset, this results in representation of each and every data point in the test set. This special case is called leave-one-out cross-validation.

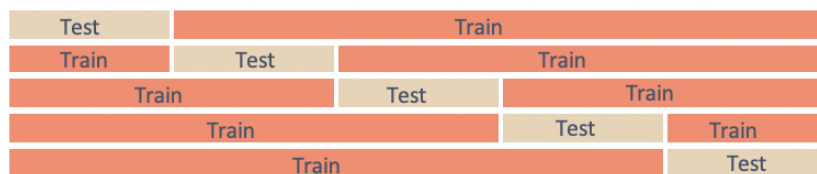


Figure 3.1. Cross-validation separates data into k folds and does performance evaluation for each of the sets. It then, combines each output to give a common performance for the model.

3.3.4. Performance Metrics

In machine learning models, after building the model, the performance needs to be evaluated in order to understand the validity of the results. Depending on the data type and the problem addressed, best measures may change, however there are certain measures that are commonly accepted for assessing model quality.

One of the ways to represent the results is confusion matrices in which true positive, true negative, false positive and false negative counts are shown to give a brief summary of the outcome. An example confusion matrix can be seen in Figure 3.2. Cells are filled after the comparison between the data with known labels and the prediction outcome.

	Actual Positive Condition	Actual Negative Condition
Predicted Positive Condition	True Positive	False Positive (<i>Type I Error</i>)
Predicted Negative Condition	False Negative (<i>Type II Error</i>)	True Negative

Figure 3.2. An example confusion matrix.

Calculating true positive (TP), true negative (TN), false positive (FP) and false negative (FN) values helps to derive other performance measures such as accuracy, precision, recall and F-score (Müller & Guido, 2015).

Accuracy can be calculated with the following formula

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FN} + \text{FP}} \quad (9)$$

Another widely accepted performance measure is precision that measures how many of the samples predicted as positive are actually positive. It stands out as a descriptive measure when eliminating false positives are important.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (10)$$

Recall or sensitivity or true positive rate (TPR), on the other hand, measures how many of the positive samples are captured by the positive predictions. Recall is an important metric when avoiding false negatives are important.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (11)$$

However, because of the nature of the way precision and recall are calculated, there is a trade-off between them. Higher precision leads to lower recall or vice versa. In order to see a better picture, F1-score is calculated from these two values. As it takes both values into account, it can be a better measure than accuracy on imbalanced binary classification datasets (Müller & Guido, 2016).

$$\text{F1} = 2 \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \quad (12)$$

Receiver operating characteristics (ROC) and AUC are another way to evaluate classifiers at different thresholds. A plot of the false positive rate (FPR) against the true positive rate (TPR) is drawn over all possible thresholds for a given classifier gives Receiver Operating Characteristics (ROC) curve. True positive rate is recall, while the false positive rate is the fraction of false positives out of all negative samples:

$$\text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}} \quad (13)$$

An ideal behavior for a ROC curve is when the recall is high while false positive rate is low. Figure 3.3 depicts an example of a good scoring, high discriminatory ROC curve. It is often that ROC curve is explained as area under the ROC curve (AUC). AUC values are good indicators of the model performance. Predicting randomly always produces an AUC of 0.5, regardless of the imbalance in data classes (Müller & Guido, 2015).

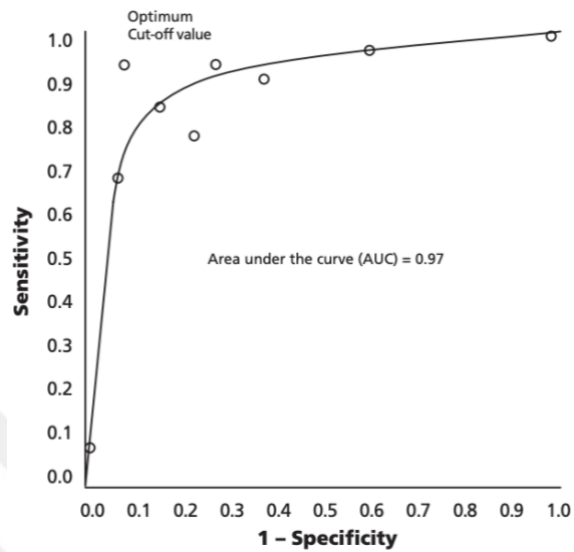


Figure 3.3. An example ROC curve illustrating high discriminatory power (Fan *et al.*, 2006).

3.4. Modelling Approach

In this study, we have followed an integrative approach where we combined features from multiple databases and sources. In this section, details about pre-processing, feature extraction and model generation are explained. An overview of the method can be observed in Figure 3.4.

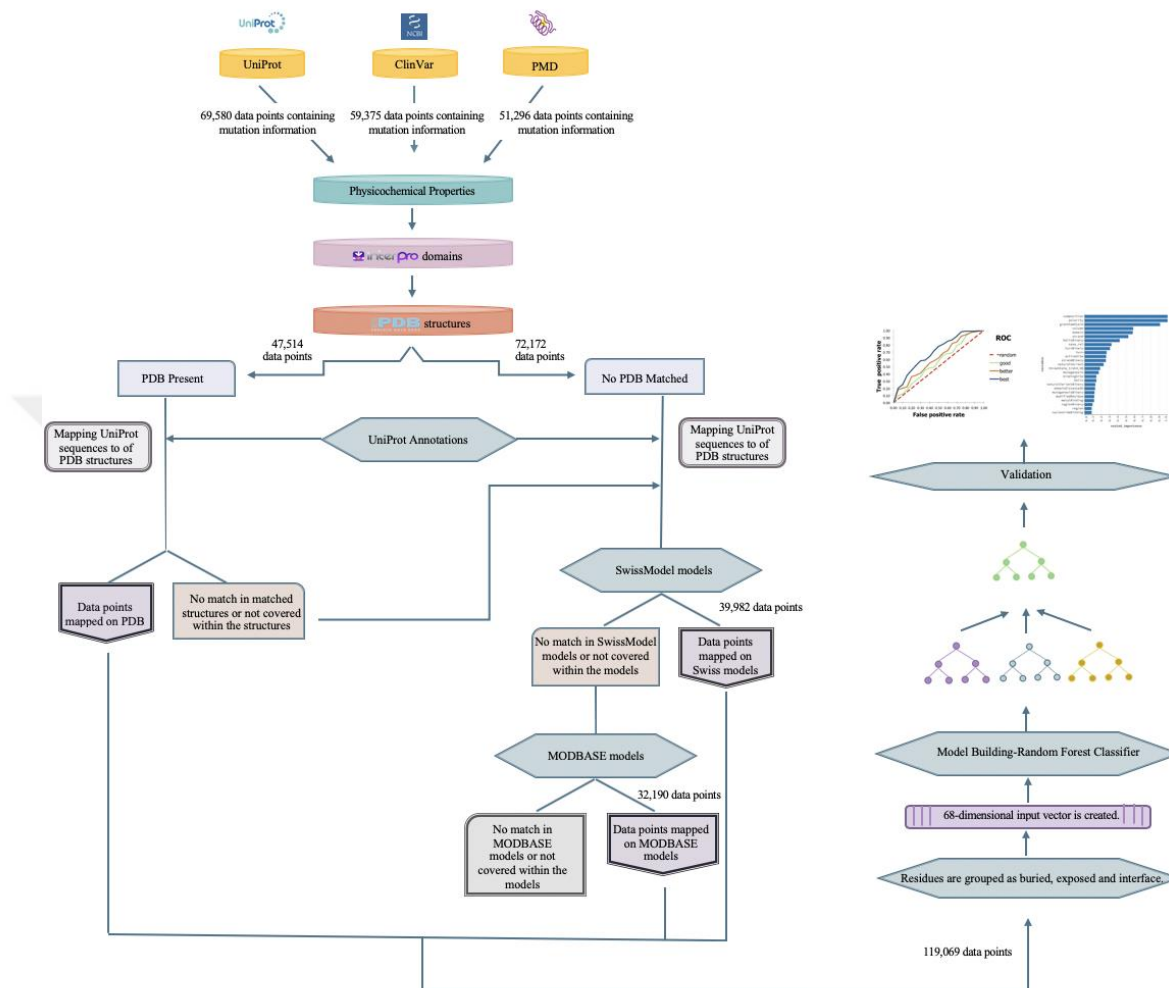


Figure 3.4. Overview of the method. Data is retrieved from three sources. Feature vector is constructed. Finally, random forest model is built.

3.4.1. Data Pre-processing

Data is retrieved from three different databases (see Methods). After retrieval, it is pre-processed in order to obtain necessary information, as well as to clean the data.

HUMSAVAR file from UniProt involves gene name, Swiss-Prot accession number, variation ID, amino acid mutations, UniProt sequence position of the mutation, variation type as polymorphism or disease-causing, dbSNP ID and the associated disease, if present. Disease labels are given if the variant is implicated in a disease. If not, variation is labelled as polymorphism. Variations with conflicting implications of disease are labelled as unclassified. Unclassified variants are excluded from data set. Additionally, 31 mutations

are also excluded as they showed conflicting labels (i.e. both polymorphism and disease for the same protein). We have ended up with 69,580 data points from UniProt. Proportion of UniProt data per effect class is shown in Figure 3.5A. According to this, 51.7% of data contained in HUMSAVAR entry belongs to polymorphism, i.e. neutral, class while 38.2% belongs to deleterious class. 10.1 % remains unclassified due to either conflicting interpretations or lack of evidence.

From ClinVar variant summary file, we have filtered relevant information that characterizes variants; National Center for Biotechnology Information (NCBI) RefSeq (Geer *et al.*, 2009) accession number, amino acid substitution and clinical significance. In the downstream analysis, we will make use of UniProt annotations and sequences. For this purpose, NCBI accession numbers are converted to UniProt accession numbers via The Biological Database Network (bioDBnet) (Mudunuri *et al.*, 2009). After conversion to UniProt IDs, outdated IDs are replaced with current IDs stored in UniProt. ClinVar contains a wide range of clinical significance categories some of which are pathogenic, likely benign, drug response, risk factor, conflicting interpretations of pathogenicity and so on. In order to avoid any misleading interpretation of disease or neutral variations, we only included a subset of these clinical significance terms. Variants labelled as likely benign and benign are included as neutral variants; whereas variants labelled as likely pathogenic and pathogenic are included as deleterious variants. Conflicting interpretations of pathogenicity and other categories are excluded. As a final step data is cleaned from repeats resulting from ID changes, synonymous changes and variations that contain missing information. In the final count 59,375 data points are remained. Figure 3.5B shows the distribution of ClinVar data per class. It can be observed that, 21.4 % of the data deposited in ClinVar are disease variants, while 30.4% are benign or neutral variants. Other categories which constitutes 48.3% of the data are composed of all remaining categories that are not included in the data.

A final database PMD is again filtered to retrieve only relevant information. An increase or a decrease in the activity, no matter the magnitude is, has a potential to impair protein's native state. For this reason, increase or decrease in activity and/or stability, along with loss of functions are recorded as deleterious, whereas no-effect cases are recorded as neutral class. Data contained variation information from 176,992 mutations, however, this data is filtered to exclude missing and conflicting information as well as repeats. Same procedure to update protein IDs as it was done for ClinVar. At the end of cleaning process, PMD introduced 51,296 data points to the final data set. As can be seen in Figure 3.5C, 55.3% of the mutations are pathogenic, or causing an effect in PMD terms, while 32.3% are benign and 12.4% are non-quantitative.

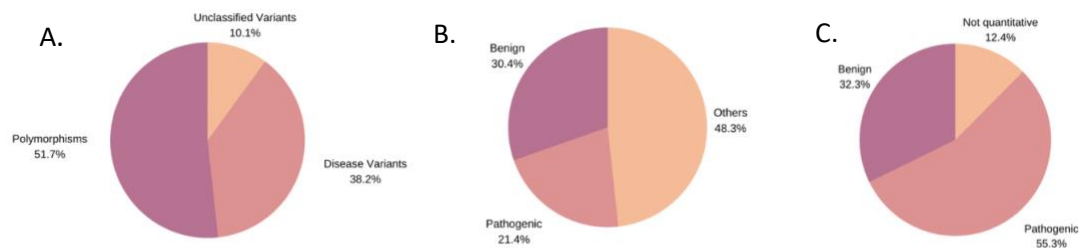


Figure 3.5. (A) Proportion of different classes of variant in 2019_01 release of HUMSAVAR. (B) Proportion of different classes of variant in ClinVar variant summary [2019_05]. (C) Proportion of different classes of variant in PMD [2019_04].

Final data is a combination of these three data sources. However, there are some cases where the label from two data sources do not match each other. We have eliminated all such cases in order to introduce a clean data to our model. We have found 207 conflicts between Uniprot and PMD, 1458 conflicts between UniProt and ClinVar and 81 conflicts between ClinVar and PMD. Table 3.2 shows the number of common and conflicting data points from 3 sources.

After excluding among and within conflicts, duplicates, or missing valued data for any reason, we have constructed out final data set for downstream feature incorporation. Data generation process yielded a total of 157,138 data points.

Table 3.2. Counts from three data sources.

	Common	Conflict
Uniprot-PMD	734	207
Uniprot-ClinVar	16989	1458
ClinVar-PMD	492	81

3.4.2. Feature Vector Construction

After creating the data set, we have constructed the feature vector to be fed into the classifier using different databases (see Methods). This section describes the methodology applied to process and include them into the final feature vector which consists of 68 input dimensions (Figure 3.6).

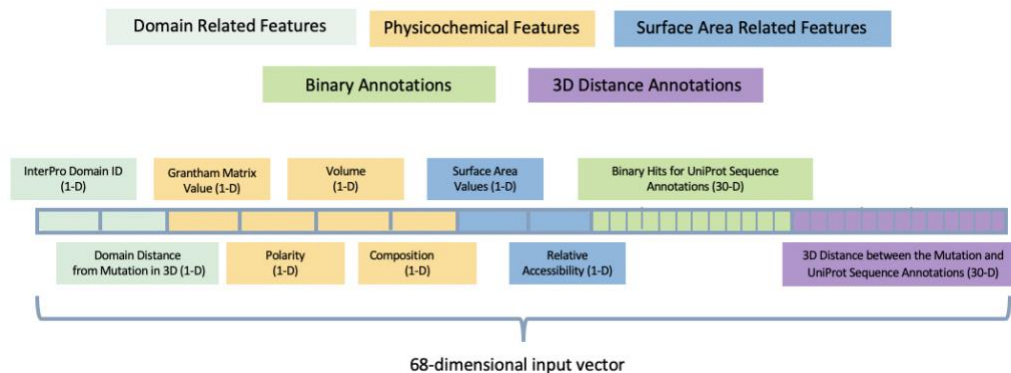


Figure 3.6. Feature vector representation of the considered features.

3.4.2.a. Domains

In order to incorporate domain information, domains that are associated with proteins in our data set are retrieved from InterPro. Majority of proteins contain more than one domain. In our model, we wanted to represent each protein and mutation with a single domain for performance reasons. For this purpose, domains are filtered with respect to their coverage of the mutation of interest. According to this, if the mutation of interest is found to occur within the domain boundaries, these domains are retained. When multiple domains remained for a data point, hierarchically top-level domain is saved. If a mutation is not found to occur within the sequence boundaries of any domain, the one with the minimum distance to the position of mutation is recorded. For some proteins no Interpro domains were present, thus they are not matched to any domains.

As a result, three groups of data points are remained with respect to InterPro domains; data points with domains that possess mutation within its boundaries, data points with domains that does not possess the mutation within its boundaries and data points that do not match to any domains. Domain information is included as a single feature that is constituted by 2,159 values.

In addition to adding domains as a categorical entity as belonging to a certain protein, minimum distance between the domain's location and the mutation is also recorded as a separate continuous and binary feature. For this purpose, domain positions from the sequence are mapped on associated structures (see below).

3.4.2.b. 3D Structures and Structure Models

Data points are mapped to their corresponding structures deposited in Protein Data Bank (PDB) where applicable. Only structures whose 3D shapes are determined via X-Ray crystallography are included. All possible PDB structures available for a protein are retained as a selection round will be applied to them based their resolution. Search space

included 46,493 structures, however; these structures are filtered for their coverage of the mutation position and resolution as will be mentioned below. 3D structures for 88,565 data points are found to be present in PDB. For the remaining 68,573 of them, modelled structures are collected from SWISS-MODEL and MODBASE databases.

We have downloaded models from SWISS-MODEL repository for proteins in our data that did not match to any structure in PDB, as well as proteins that did match to PDB 3D structures, but their mutation position is not found on the structure. 83,807 models are downloaded for 10,601 proteins. This is because a single protein has more than models; each with different ranges of coverage and quality scores (QMEAN) (Benkert *et al.*, 2009). All models are considered as candidates for proteins in our data and thus retained. Their selection is again done by filtering the ones which contain mutation of interest on their structure and also among them filtering for the quality score to get the highest scoring one.

After retrieving model data from SWISS-MODEL repository, we still had proteins that are not matched to any SWISS-MODEL models or not covered within the matched models. In the same manner that was applied in the previous structure retrieval steps, models are matched to their corresponding protein from the database. Only models that enclose the mutation of interest on the modelled structure for a particular protein; and among them the ones with the highest score are retained.

Structure data retrieved from these sources, experimental and computational, are merged with the relevant proteins (Figure 3.7). At the end of this step, each data point is represented with only one structure of its. This was necessary to reduce complexity and to be able to correctly identify data points in the model. Obtained information is used to calculate 3D distances between different sequence annotations and substitution to understand the importance of spatial arrangement in impairment capacity of mutations.

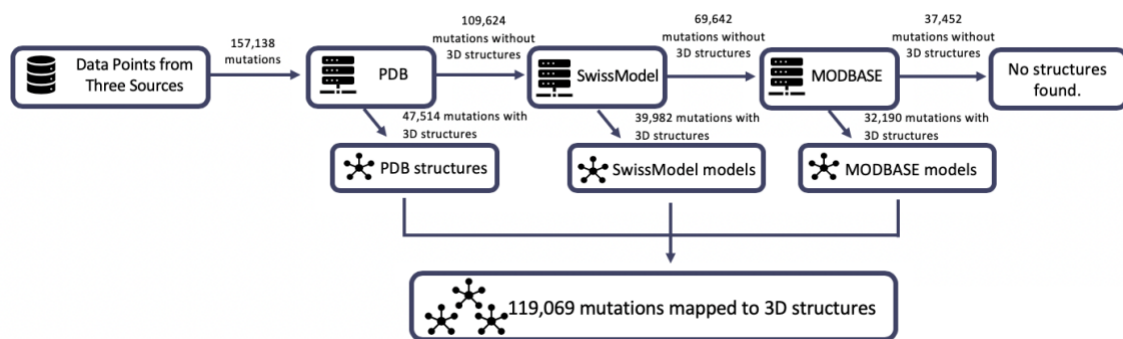


Figure 3.7. 3D structures are searched for data points. Firstly, PDB is checked. If a representative structure is not found in PDB Swiss-Model and MODBASE are consulted, respectively.

3.4.2.c. Physicochemical Property Values

For this study, amino acid substitution scores from Grantham Matrix are included into the final feature vector for each data point, along with individual values of three physicochemical property classes that are used to generate Grantham Matrix scores; namely polarity, composition and volume. Collectively, these properties added four dimensions with continuous values to the feature vector.

3.4.2.d. UniProt Annotations

UniProt sequence annotations are downloaded for proteins in the data set. We have included 30 sequence annotations to our data vector (Table 3.3). These annotations are included to the feature vector both as continuous real-valued input and as binary input as will be explained in the following section.

Table 3.3. Annotation types and classes retrieved from UniProt.

Annotation Class	Annotation Type	Description
Region	Coiled Coil	Positions of regions of coiled coil within the protein
	Motif	Short sequence motif of biological interest
	Region	Region of interest in the sequence
	Repeat	Positions of repeated sequence motifs or repeated domains
	Zinc Finger	Position(s) and type(s) of zinc fingers within the protein
	Calcium Binding	Position(s) of calcium binding region(s) within the protein
	DNA Binding	Position and type of a DNA-binding domain
	Nucleotide Binding	Nucleotide phosphate binding region
	Intramembrane	Extent of a region located in a membrane without crossing it
	Transmembrane	Extent of a membrane-spanning region
Topological Domain	Location of non-membrane regions of membrane-spanning proteins	
Sites	Active Site	Amino acid(s) directly involved in the activity of an enzyme
	Binding Site	Binding site for any chemical group
	Metal Binding	Binding site for a metal ion
	Site	Any interesting single amino acid site on the sequence
Experimental Info	Mutagenesis	Site which has been experimentally altered by mutagenesis
Amino Acid Modification	Cross-link	Residues participating in covalent linkage(s) between proteins
	Disulfide Bond	Cysteine residues participating in disulfide bonds
	Glycosylation	Covalently attached glycan group(s)
	Lipidation	Covalently attached lipid group(s)
	Modified Residue	Modified residues excluding lipids, glycans and protein cross-links
Natural Variation	Natural Variant	Description of a natural variant of the protein
Secondary Structure	Beta Strand	Beta strand regions within the experimentally determined protein structure
	Helix	Helical regions within the experimentally determined protein structure
	Turn	Turns within the experimentally determined protein structure
Molecule Processing	Peptide	Extent of an active peptide in the mature protein
	Pro-peptide	Part of a protein that is cleaved during maturation or activation
	Signal	Sequence targeting proteins to the secretory pathway
	Initiator methionine	Cleavage of the initiator methionine
	Transit Peptide	Extent of a transit peptide for organelle targeting

3.4.2.e. Mapping sequence to structure

Our method includes 3D distance values between UniProt annotations, InterPro domains and the mutation. Any residue number indicating a UniProt sequence number should be converted to sequence indices on PDB structure. For this reason, in order to obtain 3D coordinate information from PDB files (and also from other structures obtained by homology modelling), we have performed local alignment between protein's selected sequence and PDB FASTA file of its corresponding structure. In order to select which UniProt sequence to align with PDB FASTA file, we have searched UniProt database for sequences which possess same residue as wild type residue of the data point at mutation position from data set. Matched sequences are used in alignment with the matched PDB FASTA sequence. This step identified 70,392 of the proteins as matching to UniProt canonical sequence; 2,132 of them as matching to one of the isoform sequences and 16,041 of them not matching to any of the sequences available in the sequence collection for that protein. The latter group was excluded from the analysis as it did not show any sequence for alignment. For the rest, appropriate sequences are aligned with FASTA files for PDB structures. Alignment is done using BLOSUM62 as the substitution matrix. Gap scores are assigned as -11 for opening and -1 for extension. At the end of alignment procedure, we have mapped sequence annotations and domain's start and end boundaries on the PDB structure.

In addition to that, alignment results also revealed that some of the mutations were not covered in any of the structures characterized for that data point. These data points (25,010), along with the ones for which there was not hit in PDB database (68,573), are searched in SWISS-MODEL database to retrieve homology models when present.

SWISS-MODEL search was positive for 53,947 data points. For those the same procedure from finding sequences to alignment with these sequences and modelled PDB files are performed. Among 78,353 data points that entered this step, 74,219 of them are found to match with UniProt canonical sequence while 2,265 matched to some isoform sequence. Remaining 1474 did not return any hits from UniProt, thus eliminated from the data set.

Following the same procedure, alignment is performed with the same set-up as in the previous step. Since there are multiple models available for a single protein, models with the highest score, covering the mutation of interest and also covering the largest set of UniProt sequence annotations are retained. Again, some data points are found to not covered within their modeled structures (36,502). These data points along with the ones for which no SWISS-MODEL models were present (14,626) are collected for another round of homology modelled structure search in MODBASE. For 39,982 data points, SWISS-MODEL search successfully returned models that covers the mutation of interest.

Finally, the same procedure is applied for remaining 51,732 data points. Search in MODBASE resulted in 32,190 data points to use in the following steps.

Rest of the data is excluded for the same reasons above. They were either not in the range of MODBASE models, or they did not return any hits in MODBASE data base. Data points that do not match any UniProt sequence are also excluded the same way it was done in the previous steps.

3.4.2.f. Calculating 3D distances between annotations, domains and the mutations

As mentioned above, sequence annotations are incorporated into the final feature vector in two ways; both as continuous real-valued input and as binary input. Continuous input holds the distance between annotation of interest and the mutation in 3D. For this purpose, both mutation position and annotation positions are mapped to structure in order to get correct coordinates for distance calculation, as residue numbering in UniProt does not always align well with the one in PDB (Figure 3.8). For each annotation type, Euclidian distance between the mutation position and annotation position is calculated by using the following equation:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} \quad (14)$$

where

x_2, y_2, z_2 : coordinates for the mutant residue

x_1, y_1, z_1 : coordinates for the wild type residue

Some annotations are found spread throughout a range of amino acids. Also, some of the annotation types are annotated multiple times in a single protein. For such cases, Euclidian distance between the mutation position and residue constituting the minimum distance among all the residues that make up the annotation is considered.

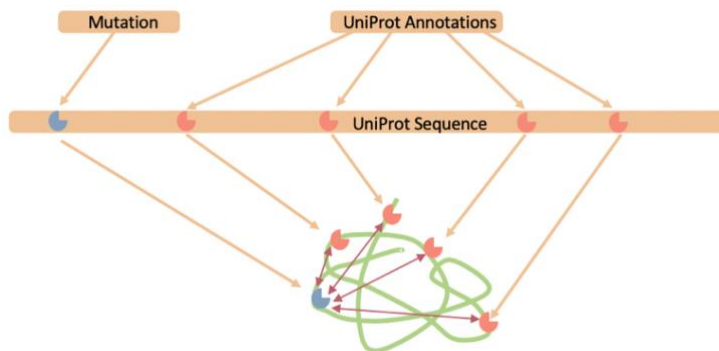


Figure 3.8. Mutations and UniProt annotations are mapped on protein's corresponding 3D structure. Distance between each annotation type and the mutation is calculated from their 3D coordinates.

Not every protein has a structure associated with it in the final data set. For this reason, binary inputs are calculated on the UniProt sequence instead of checking for hits on the structure. In other words, a mutation is labelled as hit if its mutation position is found to hit the annotation position on the UniProt sequence. This has saved data points that may lack a structure, thus distance information, to the final data by considering annotation relation in a different manner.

Domains are treated the same as annotations covering a range in the protein sequence. Distance between the mutation position and domain start and end position are calculated. Minimum of these values are retained as the distance between them, if mutation does not fall inside the domain. If it does, distance is recorded as 0.

Including UniProt annotations added 60-dimensions to the final feature vector; 30 being continuous distance information and 30 being binary hits. Extra 2 dimensions (1-continuous, 1 binary) are added by including distance from the domain.

3.4.2.g. Relative Solvent Accessibility

As a final step, different classes of residues based on their accessibility are included into the feature vector. We separated residues into three as surface, buried and interface. A stringent threshold of 4% is selected to discriminate between surface and core residues (Dincer, 2019; Dincer *et al.*, 2019; Momen-Roknabadi *et al.*, 2008). According to this, an amino acid with relative surface accessible less than 4% is considered as a buried residue, while a residue with a relative accessible surface are more than 4% is considered as surface residue. In order to include interface information validated interfaces from databases along with high quality ECLAIR predictions are merged with the other two categories. If residues previously labelled as surface are present in the interface residue data, they are relocated to the interface group. If buried-labelled residues are found in interface data, they are labelled as conflict.

As a result of FreeSASA and Interactome Insider analysis, three groups are generated for datapoints. This feature is included as a single dimension to the final feature vector, with 3 possible values; interface, surface and buried.

3.4.3. Model Implementation

This section describes the tool used to generate random forest model. Random Forest models can be implemented by various ways. Common implementations utilize Python's sci-kit learn package (Pedregosa *et al.*, 2011) or R programming language (Liaw & Wiener, 2002). However, they both fail to handle categorical data in its natural form. Encoding options are out of the scope of this study. However, to mention briefly, Python implementations convert categorical data into some numeric or matrix form and treats the feature accordingly. This can include converting categories into some ordinally sorted numbers. However, this is not applicable for all types of categorical data such as colors.

Colors do not have an inherent order in them, that's why they cannot be ordinally encoded. One other transformation can be converting the categorical data into a matrix in which each category is represented in a different column and data points including the feature of interest is labelled as 1 in that particular column. This is called one-hot encoding and is an alternative for handling categorical data in sci-kit learn implementations (Hastie *et al.*, 2009). However, it is not good for data with high cardinality, meaning data containing a lot of values, for the categorical feature. For example, in our data one of the categorical features is the domains. We have 2,159 domains for 119,069 data points. This would create a matrix for 119,069 x 2,159 dimensions matrix for encoding only one column. In addition to that, each feature is represented with one domain, adding another level of difficulty in terms of matrix sparsity. Using one-hot encoding for such a problem would cause problems more than it offers a solution. One-hot encoding is also not recommended for tree-based algorithms. Sparsity of the matrix causes algorithm to read the variables wrongly by decreasing the importance of the categorical variable and not incorporating in the early splits if they don't meet a high response rate. On the other hand, R implementations can treat categorical variables in their natural form, as categorical. However, it has limitations to be able to handle up to 53 levels. It is again incapable of accommodating the data we have for domains.

For all these reasons, in this study random forest algorithm is implemented on H2O.ai platform (The H2O Team, 2015). It is an open source software for artificial intelligence and machine learning solutions. It has interfaces for R, Python Scala, Java, JSON, and CoffeeScript/JavaScript, as well as a built-in web interface, Flow. H2O includes a number of algorithms such as Naïve Bayes, PCA, k-means, random forest and logistic regression. Although it has some options for different encoding options, H2O also has the ability to handle categorical categories in their native form. This is valuable for our study as domains are included as a categorical feature with more than 2,000 levels in the final feature vector.

For this study, H2O's built-in web interface is used to obtain predictions. Different parameters are searched over a grid space in order to find best performing parameters.

CHAPTER 4

4. RESULTS

4.1. Summary of The Data

Data is collected and cleaned as explained in the previous sections (see Methods.). After the pre-processing steps, initial data contains 157,138 data points for 15,481 distinct proteins from three data sources. A data point is defined by protein ID, wild type residue, variation position and mutant residue. First of all, InterPro domains are matched for each data point. As a result of hierarchy analysis, 2,159 distinct domains are remained for all of the data points. Later on, structures from PDB are matched with data points as explained in section 3.4.2. 88,565 data points are found to match a structure from PDB, while 68,573 of them did not match to any structures. 16,041 of 88,565 data points are excluded because their wild type residue did not show any matches in any of the UniProt canonical or isoform sequences. Remaining data points are taken for alignment with their corresponding structures to map them on a 3D space. 47,514 of them are found to be represented on the matched structures, while 25,010 data points are not represented on the crystallized 3D structure boundaries. Thus, they are taken for the search in the homology modelled structures' space. This search is firstly done in SWISS-MODEL database. These 25,010 data points along with 68,573 data points from the first split are searched in SWISS-MODEL for homology modelled structures. Only 78,353 of them are found to have models. 1,474 of them are again excluded for not containing the same residue as the wild type in UniProt sequence. After repeating the same steps that was done for PDB structures, i.e. alignment, 39,982 of them are found to be represented on the matched models while 36,502 are not. These not-represented ones, along with the ones that do not have any models in SWISS-MODEL are then taken to MODBASE. A total of 51,732 data points is searched for modelled structures from MODBASE. This search yielded 32,190 data points having MODBASE models which contains mutation position on the modelled structure range. A summary of the data count can be seen in Table 4.1.

Eliminating those data points without any mapped 3D structures, either experimentally or computationally determined, we have ended up with 119,069 data points to continue our analysis.

Table 4.1. A summary for the counts of mutations in the dataset.

Initial Number of Mutations	157,138
Number of Unique Proteins	15,481
Number of Unique Interpro Domains	2,159
Total Number of Mutations with Structural Information	119,069
Number of Mutations that correspond with PDB Structure Regions	47,514
Number of Mutations that correspond with SwissModel Structure Regions	39,982
Number of Mutations that correspond with MODBASE Structure Regions	32,190
Number of Neutral Mutations in The Final Dataset	50,238
Number of Deleterious Mutations in The Final Data set	68,831

4.2. Distribution of Data Points Among Domain Regions

Firstly, the data is analyzed as whether data points fall within the boundaries of the associated domains or not. UniProt sequences are considered while exploring this aspect. According to this, if a given mutation is found to fall within the UniProt sequence boundaries (sequence indices) of their domain, these mutations are considered as range; and if not, they are considered as not-range. From our 119,069 data points, 88,251 of them are mapped to a domain from InterPro. Remaining 30,818 are treated as proteins without any domain information. Although they did not show any match to any domains for our data set, this is related to the Interpro version used in our study and domains might be found for these data points either from a different and more recent InterPro build or from other domain databases.

Figure 4.1 shows the distribution of the mutation locations with respect to their associated domains. As can be seen, mutations found within domain boundaries shows a higher proportion for deleterious ones compared to the neutral ones. On the other hand, for mutations found in the vicinity of domains, the proportion favors neutral effect. As explained before, domains are functional and distinct regions within the proteins, and they are responsible for the protein tasks. From this information, we expected to observe mutations that cause an effect on the protein function to be located within the domain boundaries more abundantly compared to the ones that do not cause an effect. Findings from our data backs up this claim about the placement of neutral and deleterious mutations. According to this, when a mutation occurs within a domain, it is more likely to cause an impairment in the protein as it directly affects a functional site by causing cause changes in the domain architecture.

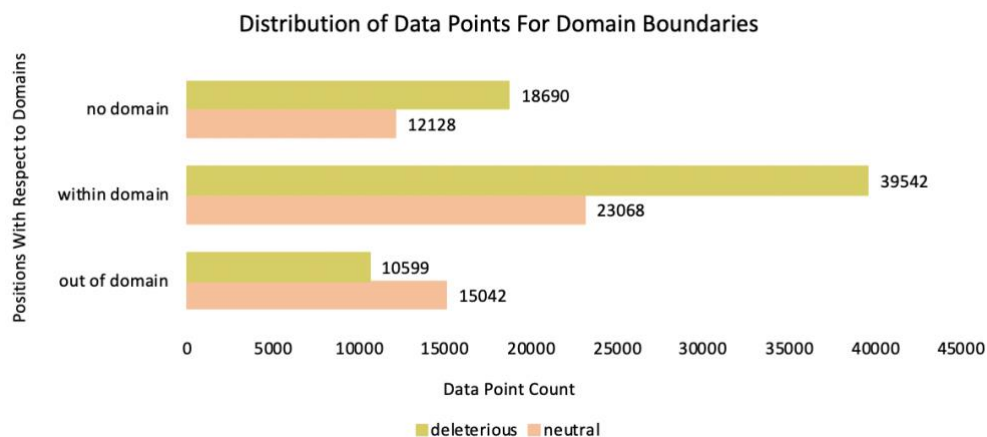


Figure 4.1. Distribution of data points for their presence within or out of the domain boundaries.

4.3. Identification of Significant Protein Domains

In the dataset, we had a total of 2,159 distinct domains mapped to 119,069 data points. Even though we have merged domains belonging to the same hierarchical family into one group and replaced hierarchically lower ones with the superior ones (see Methods), 2,159 is still a high number for the classifier to handle. It also carries the risk to overfit when this many distinct domains are used. For this reason, Fisher's exact test (Fisher, 1956) is applied to the domains in order to select the ones with the highest importance. Fisher's exact test is a method to assess the significance of difference between two groups. It shows the proportion of one class of variables over the values of the other class. A contingency table of $m \times n$ size, m and n being the number of searched categories for each condition, is created for calculations in which data falling in each category is recorded in the related cell. Statistical measures such as p-value is calculated through the values in this table. It is an exact test because the deviation from the null hypothesis can be calculated exactly, without needing an approximation. For this study, a contingency table of size 2×2 is created for deleterious and neutral mutations vs. domain presence and absence.

Fisher's exact test resulted in 327 distinct domains when p is set to 0.01. According to this, the most informative domains are listed in Appendix A. Their importance is defined by the separation power they hold for each category. Total number of data points that a particular domain is associated with and their proportion for neutral and deleterious classes can also be seen in the Appendix A. Figure 4.2 shows the most crowded domains among the remaining ones after the analysis. Being the most crowded domain does not mean to be the most informative domain. As can be traced from the Appendix A, the domain that tops the list is IPR011162; however, in Figure 4.2 it can be seen that the most crowded domain is IPR027417. This is a result of the proportion of each domain holds for

neutral and deleterious class. The bigger the gap is between two classes, the more informative a domain is.

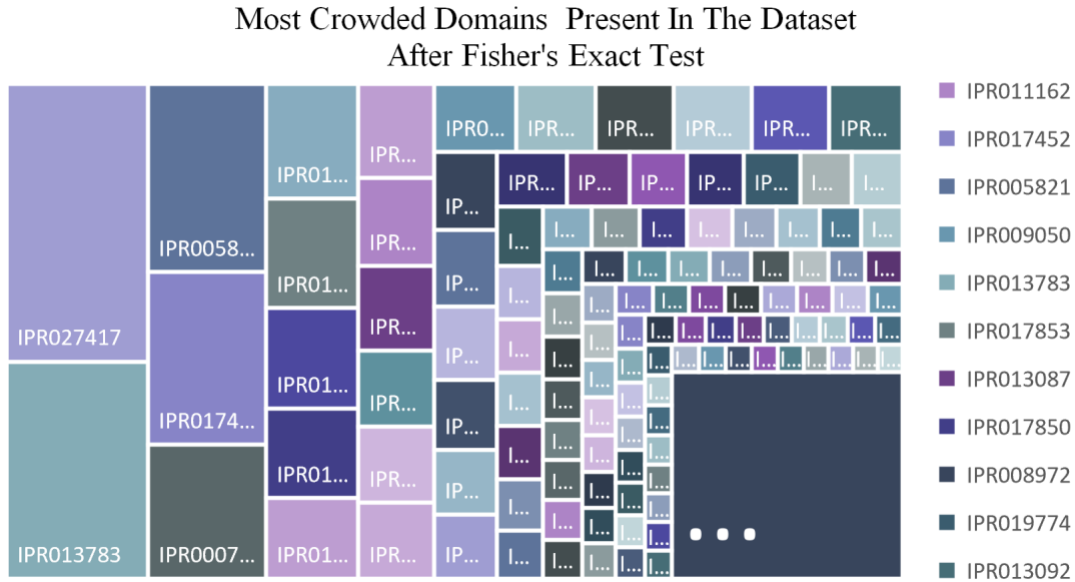


Figure 4.2. Distribution of domains for their abundance after Fisher's exact test.

When top resulting 10 domains after Fisher's exact test are inspected for their biological role, it can be seen that domains are responsible for a variety of protein functions including cell surface ligand recognition, ion channel activity, enzymatic activity and cellular transportation (Table 4.2). Each of these functions are important in cellular messaging and signal conduction within the cell. Thus, when an impairment occurs within these regions that disrupts the domain, it is very likely to affect downstream mechanisms, and ultimately cause aberrations in the cell.

A total set of 2,159 domains, as well as a reduced version containing 327 domains are used to train different classifiers to observe the effect and separability capacity of domains.

Table 4.2. 10 most informative domains after Fisher's exact test analysis

Domain ID	Domain Name	Domain Description	Total Cases	Non-neutral	Neutral
IPR011162	MHC classes I/II-like antigen recognition protein	Cell surface receptors that function to present antigen peptide fragments to T cells responsible for cell-mediated immune responses	632	10	622
IPR017452	GPCR, rhodopsin-like, 7TM	Transduce extracellular signals through interaction with guanine nucleotide binding (G) proteins	1918	550	1368
IPR005821	Ion transport domain	Found in sodium, potassium and calcium ion channels proteins	2095	1905	190
IPR009050	Globin-like superfamily	Heme-containing proteins involved in binding and/or transporting oxygen	530	48	482
IPR013783	Immunoglobulin-like fold	Contains immunoglobulin-like (Ig-like) fold. Involved in interactions, commonly with other Ig-like domains via their beta sheets.	2908	1290	1618
IPR017853	Glycoside hydrolase superfamily	Enzymes that hydrolyze the glycosidic bond between two or more carbohydrates, or between a carbohydrate and a non-carbohydrate moiety.	956	840	116
IPR013087	Zinc finger C2H2-type	DNA binding motif. Transcription factors usually contain several Znfs capable of making multiple contacts along DNA	607	181	426
IPR017850	Alkaline-phosphatase-like, core domain superfamily	These domains form the core domain of alkaline phosphatases	780	687	93
IPR008972	Cupredoxin	Stabilizes copper (I) ion from which is created after reduction of toxic copper (II)	462	430	32
IPR019774	Aromatic amino acid hydroxylase, C-terminal	Hydroxylation of the aromatic amino acids phenylalanine, tyrosine, tryptophan	293	285	8

4.4. Physicochemical Properties of Mutations

Literature evidence that physicochemical properties have the potential to guide algorithms differentiate between distinct classes. Since they define the protein's structure, it is no surprise that they are determinants in the final functioning of the proteins.

In our model, we have included three physicochemical properties as polarity, volume and composition, along with a Grantham matrix scores. All values are calculated as the difference between the wild type and the substituted residue. In our data, we had some mutations that resulted in a termination codon incorporation as the substitute residue. No values of physicochemical properties or Grantham scores is available for such cases as termination codons do not code for any amino acid. 11,371 data points are found to constitute this class. These data points are also included in the data because this itself is a property to infer meanings from for the classifier.

Distribution of considered physicochemical values across the data set can be seen in Figure 4.3. For the three properties, namely polarity, composition and volume, the difference between the two classes is not as notable as the Grantham score case. However, when Grantham score graph is examined, it can be observed that there is a significant shift towards bigger values in the non-neutral graph. A study conducted by Huang *et al.* (2010) showed that higher scores are observed more in disease labelled mutations; whereas lower scores are more frequently observed in neutral cases. This is expected as a greater value in the Grantham Matrix indicates a greater difference between the wild type and its substitute. The greater the difference, the bigger the effect is on the protein's functionality.

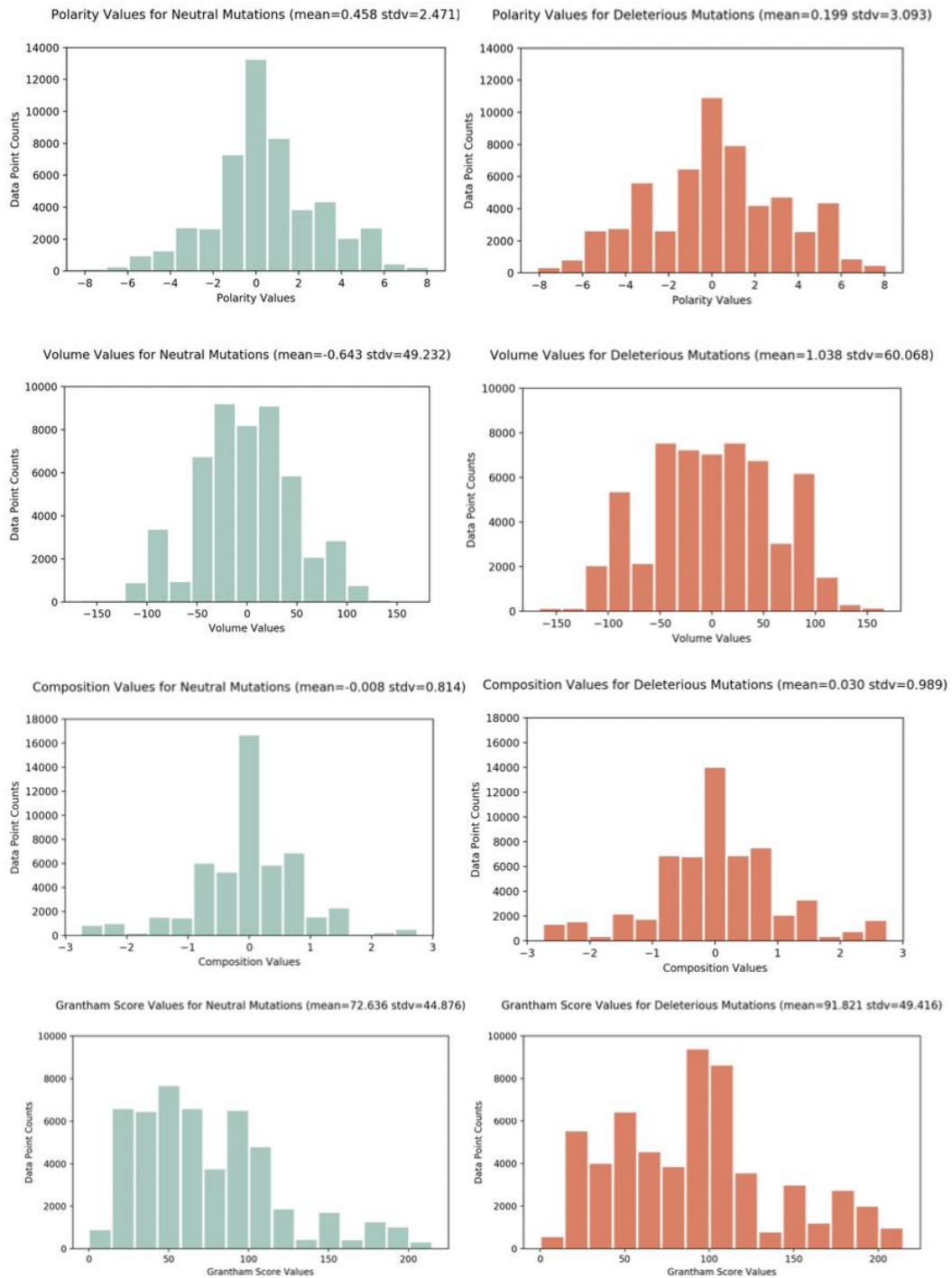


Figure 4.3. Distribution of data points for the physicochemical properties considered.

In order to observe the effect of each physicochemical property on different regions of the protein structure, we have applied Fisher's exact test for each property in three significant regions; core, interface and surface. Firstly, we have separated the data into these three categories. For each category, we labelled each data point as 'significant change' and 'not significant change' for different thresholds for different properties. Thresholds are chosen to leave approximately same number of data points per condition, i.e. deleterious and neutral. For polarity property, data points having polarity values higher than 1.6 or lower than -1.6 are considered in the 'significant change' class while data points lying in between these values are considered in the 'not significant change' class. For volume property, this value is set to 38, i.e. values lower than -38 and greater than 38 are considered as 'significant change', while values in between are considered as 'not significant change'. The threshold for composition was set to 0.52 and the selection is done in accordance with former two. Finally, for Grantham Matrix scores, we have counted values lower than 81 as not significant, while values higher than 81 were recorded as significant. Data points in each group are also further divided into two groups as deleterious and neutral. Counts for each sub-group, for example deleterious mutations in significant polarity change group for core region, are obtained and used to calculate p-values for the significance of association between groups. Related contingency tables can be found in Appendix B for a detailed investigation for the counts per classes. A confidence interval of 99% is selected for evaluations. According to this, p-values smaller than the threshold are considered to be significant and interpreted as an important association between certain physicochemical property and the effect of mutation. Calculated p-values for each condition is given in Table 4.3. Region-wise analysis shows that for all of the regions, the highest significance is observed for Grantham Score values. This is expected as Grantham values are calculated by taking all three others into consideration and thus show a better distinction in values per effect class.

Table 4.3. The significance (i.e. p-values) of associations between the changes in physicochemical properties at the mutated site and the effect of the mutation (i.e. deleterious or neutral), evaluated independently for different protein regions.

	Composition	Polarity	Volume	Grantham Score
Core	1.97e-25	1.04e-91	1.96e-116	3.27e-159
Interface	0.00072	4.73e-22	7.13e-30	2.06e-34
Surface	3.93e-88	6.18e-250	1.44e-274	0.0

4.5. Family-Based Evaluation of Mutations

In order to understand the association between mutation effect and physicochemical properties in protein families, like in the previous section, we've applied Fisher's exact test to our data. This time, instead of separating the data as core, interface or surface regions; we have grouped the proteins in our data set into five protein families as enzymes,

ion channels, membrane, transcription factors and others. For each family, we have again used the same thresholds to group physicochemical changes and further divided the dataset as deleterious and neutral. A confidence interval of 99% is chosen for evaluation. Tables having the individual counts per group can be found in Appendix C. Individual values for each protein family is given in Table 4.4. When the table is examined familywise, we can observe that the highest significance is observed in the Grantham Score property for enzyme, ion channel and membrane families, whereas for transcription factor family and other category volume was the most significant property that showed a difference between neutral and deleterious classes. For enzyme family, the magnitude of significance is much higher than any other families, showing a significance value of $3.70e-163$. Other families are observed in comparable ranges. For families except membrane family, least significance is observed in composition. For the membrane family, however, lowest significance is recorded in the polarity property. For transcription factor family and other category, the p-value is not significant for composition with values very far from the selected significance threshold of 0.01. Polarity and Grantham Matrix score is also not found to be significantly associated with the mutation effect in the transcription factor family. We observed that Grantham Value shows the most significant association for most of the families; however, Grantham Value is a combined value from three actual properties. Significance in this context refers to the association of the change in property values upon a mutation introduction and its resulting consequence. Thus, when we inspect these individual properties alone, we can see that a change in volume has the most significant association with the mutation effect in all families. This can be explained by the importance of the 3D spatial organizations of proteins in protein function. Changes in the volume of the residue in a certain position can change the overall 3D structure of the protein due to the size constraints. When significant volume changes are seen in a residue position, it is likely that it will alter the protein's structure thus function. When the inspection is done property-wise, we observe that the most significant values are observed for the enzyme category, suggesting that in enzyme proteins physicochemical properties are good indication for distinguishing between disease and neutral mutations. For properties other than composition, the least significance is found in transcription factor family which holds the highest p-values for the physicochemical properties. For composition highest p-value is observed for the others category which is actually not a significant value. From these observations, we can deduce that the physicochemical properties are mostly significant for families and they are good indicator of separation between deleterious and neutral classes.

Table 4.4: The significance (i.e. p-values) of associations between the changes in physicochemical properties at the mutated site and the effect of the mutation (i.e. deleterious or neutral), evaluated independently for different protein families. * shows non-significant ones.

	Composition	Polarity	Volume	Grantham Score
Enzyme	4.12e-28	4.37e-72	6.22e-102	3.70e-163
Ion Channel	4.80e-05	3.67e-08	1.65e-14	8.71e-21
Membrane	3.60e-12	3.28e-10	9.16e-18	1.82e-28
Transcription Factor	0.57*	0.19*	2.80e-08	0.002*
Others	0.71*	9.71e-23	6.65e-32	3.84e-23

4.6. UniProt Annotations and Their Distribution in The Mutation Dataset

Our model is annotation centric which means we want to investigate the role of sequence annotations on the impairment capacity of mutations. For this reason, we have mapped 30 UniProt sequence annotations to the data. Details on the selected annotations can be found in the Methods section. Here, we analyzed the abundance of each annotation type for our two final classes.

Figure 4.4 shows the proportion of annotation for each annotation category in our data set. As can be seen from the numbers, most of the annotation types are found to be annotated only for a small number of data points. A majority of them, however, are not found annotated for a bigger number of cases. A protein does not necessarily contain all the annotation types selected. In addition to that, even when present it is not as easy to identify all functionally important regions. Annotation entries are updated with the ongoing efforts, but currently our access for annotation data is limited. Thus, this gap between the annotated case and not-annotated case for each annotation category is an expected observation.

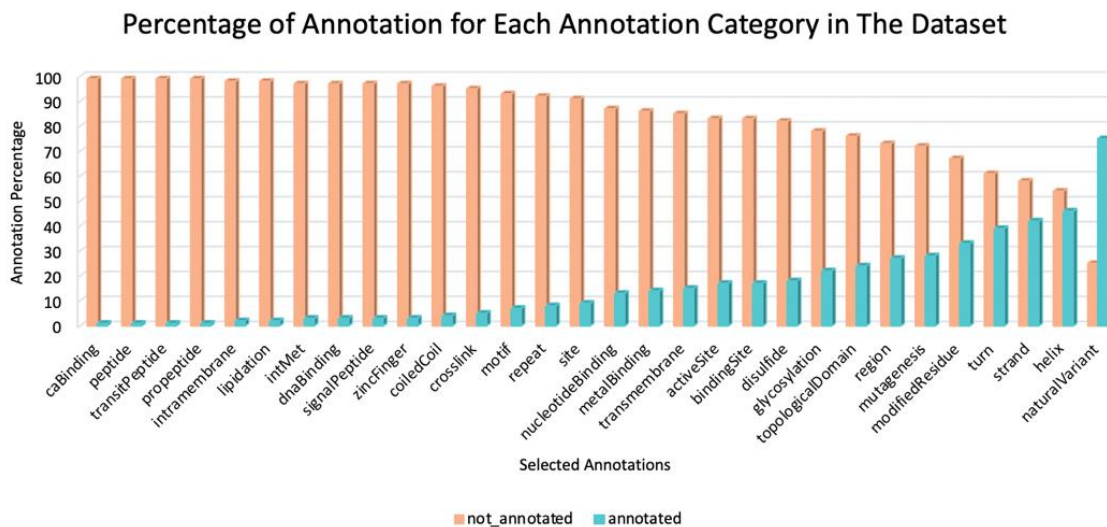


Figure 4.4. Percentage of each annotation category in the dataset. Each annotation category is annotated in a small number of data points. Only exception is natural variant category. This is expected as we are specifically dealing with data points with variations.

It is also important to investigate annotations with respect to their impairment capacity and whether the mutation is found within the annotation boundaries. Figure 4.5 shows the observation frequencies of each annotation category for two response classes when the mutation is found within the annotation boundaries or hit the annotation position. Some annotations cover a long stretch in the protein sequence while some other are only represented in a single amino acid. For both cases, if the mutation is found within this sequence stretch or occupies the same sequence position as the annotation, it is considered as a hit. As can be seen from the graph, when the mutation is found within the annotation range, a majority of the annotation types tend to cause some sort of impairment and show non-neutral effects. Initial methionine shows to affect the protein function for all data points. This annotation is added when a mutation changes the starting methionine in the sequence, thus affects the translation of the protein. This means, when an initiator methionine is changed, it is almost certain that there will be a problem in the functioning of the protein.

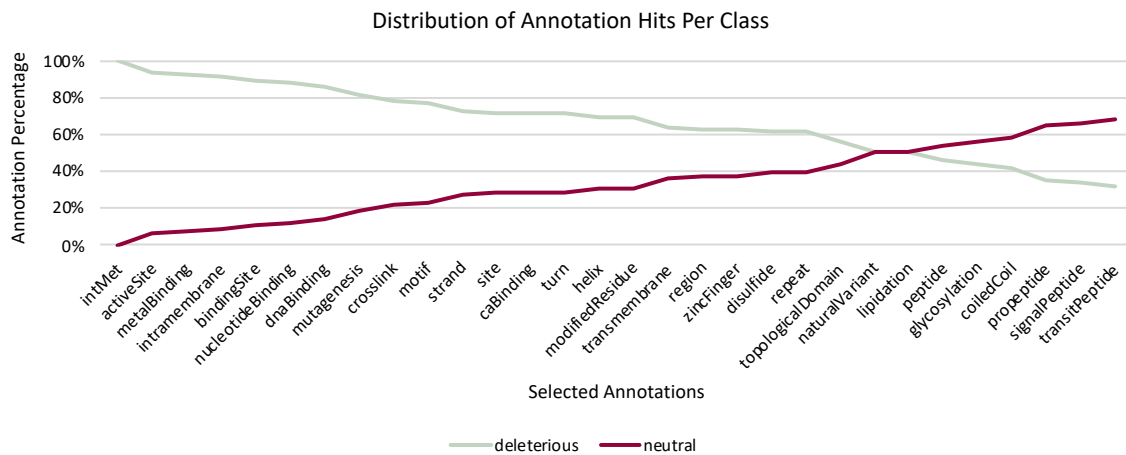


Figure 4.5. Percentage of neutral and deleterious mutations per annotation class for mutations found to occupy an annotation region.

4.7. Distribution of Data Points In Terms of Relative Accessibility

One more feature that is included in the final feature vector is relative solvent accessible area; and three groups, buried, surface and interface, defined from the outcome of area values. In order to determine which threshold to select for group separation, we performed the analysis with different thresholds. For these analysis a stringent 4% and more relaxed 16% thresholds are selected (Dincer, 2019; Dincer *et al.*, 2019; Momen-Roknabadi *et al.*, 2008). 4% yielded 18,822 data points for the buried class, while 16% yielded 38,386 data points for the same class. The rest was assigned to surface class for both cases. However, when interface information is incorporated, 4% threshold yielded 1,884 conflict cases for high quality selection, i.e. labelled as both interface and buried, while 16% yielded 3,912 conflicts. High quality refers to the confidence of prediction by ECLAIR algorithm of Interactome Insider. When the interface data includes medium confidence predictions it is referred as low quality. Conflict numbers are found to be 2,847 for 4% threshold and 6,072 for 16% threshold. According to this, to minimize the conflicts we continued our analysis with 4% threshold and high-quality interface residues. Resulting three groups are examined for their possession of deleterious and neutral mutations. In Figure 4.6, counts for our data set is given as a result of this grouping. As shown, in interface and core regions, non-neutral mutations are observed more than two-fold compared to neutral mutations. For interface class, 68% of interface mutations are shown to be deleterious, while 32% are neutral. On the other hand, for core class, 70% of the mutations are deleterious, while 30% are classified as neutral.

For surface residues, this ratio is not that drastic, however, since surface regions can also contribute to the function of the protein, it is expectable to observe more deleterious mutations in these regions, as well. For this class, 49% of surface mutations are classified

as neutral, while 51% are classified as deleterious. As shown in the literature, core and interface regions are responsible for protein's stability and interaction with other proteins, respectively. Given this information, we expect to observe more non-neutral mutations in these regions. Results from our analysis supports this hypothesis and show a higher fraction of non-neutral mutations in these regions.

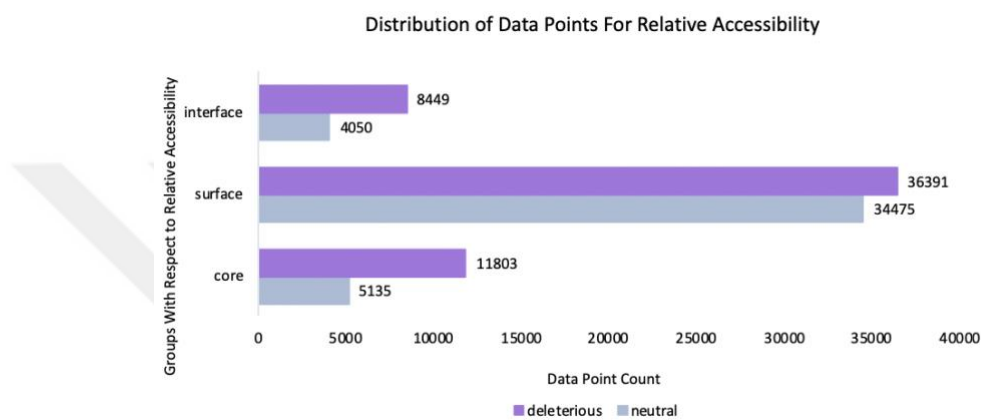


Figure 4.6. Distribution of data points for accessibility group.

4.8. Mutation Classification With Random Forest Classifier

In this study, random forest classifier is used in order to obtain predictions for 109,069 data points of which 50,238 belong to the neutral class and 68,831 belong to the deleterious class. A feature vector of 68 dimensions is fed into the classifier that consists of protein and mutation level features such as physicochemical properties of the residues, UniProt sequence annotations and surface accessibility results. The output of the classifier is a binary classification where given data points are separated as deleterious or neutral. Random forest classifier merges multiple trees, each of which considers a different subset of data points and features and makes the decision as an agreed output from generated trees. In order to build the model, we have tested the performance of different subsets of the feature vector and observed the contribution of selected features. First of all, we checked the contribution of the domain attribute which consists of 2,159 distinct domains for all data points. This is a very crowded attribute for a random forest classifier to make decisions, thus we wanted to see how the complexity affects the performance. Later on, we have tested the contribution of physicochemical properties. Physicochemical properties are reported in the literature as features that provide a good indication for separation, thus we wanted to observe their sole effects for our classifier. Following that, we checked the effect of annotations as we want to build our main model annotation based. Examining the contribution of annotations is particularly important to validate our hypothesis that functionally important regions contribute to the overall function of the protein and changes in these regions are more likely to help us infer meanings from them.

Finally, after exploring the individual contributions we have constructed our final vector with the informative attributes and made a grid search over a space of hyperparameters to optimize our model.

4.8.1. Evaluating the Effect of Domains

Domains are distinct functional units in proteins which can carry meaningful information in terms protein function, and disruptions in domain regions are expected to cause deleterious effects upon non-synonymous mutation introduction. In our feature vector, we have 2,159 distinct domains that constitute to a categorical-valued vector. However, this many values in a feature vector causes high cardinality and many of the machine-learning implementation tools cannot handle high cardinality. H2O platform is developed to be able to process high cardinality values, thus we chose to use it in our analysis. However, we still wanted to understand the effect of a complex feature on the prediction outcome, and for this reason we have compared two different models. In one of the models, we used all of the 2,159 domains assigned for all data points. In the other one, we decrease the complexity of this vector by doing a Fisher’s exact test on the domain vector and selecting only the significant domains that show a high power for discriminating two output classes. We have used default parameters in order to provide an equal hyperparameter space. 50 trees are generated with 25 maximum depth. Full parameter list can be found in Appendix D. 10-fold cross-validation is performed on the data that is split 75% to 25% as training and independent hold-out validation set, respectively.

Confusion matrix values are given in Table 4.5 for all domain analysis and in Table 4.6 for reduced domain complexity analysis. Neutral output is evaluated as the negative class, while deleterious output is considered as positive class. According to this, in the analysis with all domains included we have 6,943 true negatives, 5,574 false positives, 1,790 false negatives and 15,452 true positives. Recall is calculated as 0.86, while the precision is 0.89 and specificity is 0.55. F1 score is reported as 0.87. For the second analysis where we only included statistically significant domains, we have 7,947 true negatives, 4,720 false positives, 1,942 false negatives and 1,524 true positives. Recall is calculated as 0.89, while the precision is 0.76 and specificity is 0.63. F1 score is reported as 0.82. Other metrics derived from precision and recall values are reported in Table 4.7 for both analyses.

Table 4.5. Confusion matrix for the hold-out validation set of all-domains model.
Rows: Actual Class; Columns: Predicted Class

	predicted neutral class	predicted deleterious class
actual neutral class	6943	5574
actual deleterious class	1790	15452
Total	8733	21026

Table 4.6. Confusion matrix for the hold-out validation set of significant-domains model.

Rows: Actual Class; Columns: Predicted Class

	Predicted neutral class	Predicted deleterious class
Actual neutral class	7947	4720
Actual deleterious class	1942	15244
Total	9889	19964

Table 4.7. Performance metrics for all-domains and significant-domains models.

Measure	All Domains	Significant Domains
Sensitivity	0.8565	0.8870
Specificity	0.5547	0.6274
Precision	0.8880	0.7636
Negative Predictive Value	0.4840	0.8036
False Positive Rate	0.4453	0.3726
False Discovery Rate	0.1120	0.2364
False Negative Rate	0.1435	0.1130
Accuracy	0.7976	0.7768
F1 Score	0.8720	0.8207
Matthews Correlation Coefficient	0.3911	0.5401
AUC	0.8400	0.8600

ROC curves for validation sets are shown in **Figure 4.7** for both analyses. According to this, validation AUC value is reported as 0.84 for all-domains model, and as 0.86 for significant domains.

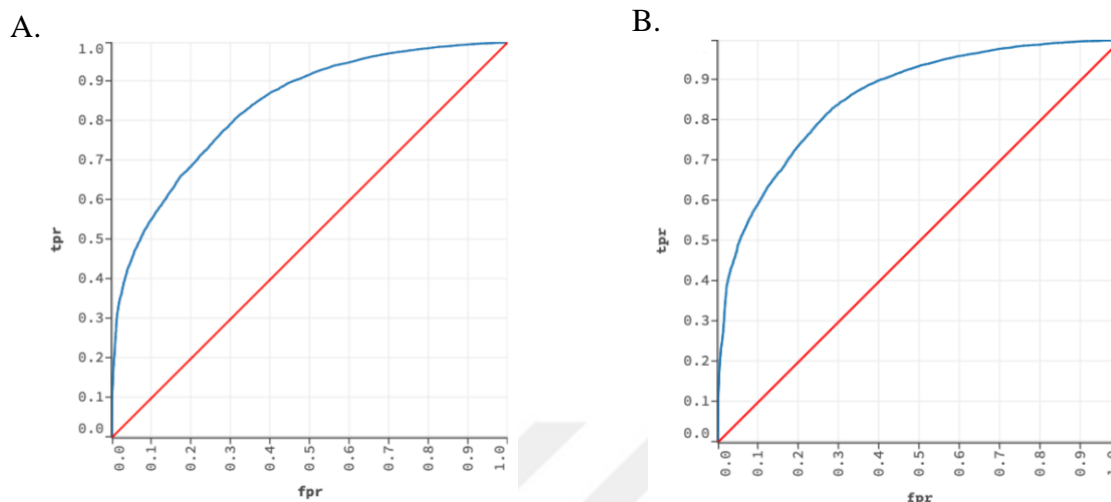


Figure 4.7. ROC curves for the hold-out validation sets of the models (A) where all domains are excluded, and (B) where only significant domains are included. AUC values are 0.84 and 0.86 respectively.

One more thing to consider as a result of random forest analysis is the feature importance. Random Forest models rank the importance of each feature according to their power for separation of the data. Different trees are generated with different feature subsets and different samples during a random forest implementation. Algorithm calculates decrease in the error upon each split for different features and determines the importance and relevance of each feature based on calculated error values. Figure 4.8 shows two figures of feature importance for two analysis. According to this, in the first model, the rank of the domain feature is 4. However, when statistically redundant domains are eliminated from the model, feature importance for domains raises to the first rank. This is because, this elimination only leaves the most informative and discriminative domains and the information obtained from remaining domains becomes more significance. In the second figure, we can also observe that the scaled importance for other features decreases as domains get more importance. For instance, scaled importance of the second feature decreases from 0.79 to 0.56 when domain feature gets to the first rank. In the first rows, other than domains we can also see the dominance of physicochemical features. These findings are in accordance with the literature as many studies highlight the importance of physicochemical features for variant effect prediction.

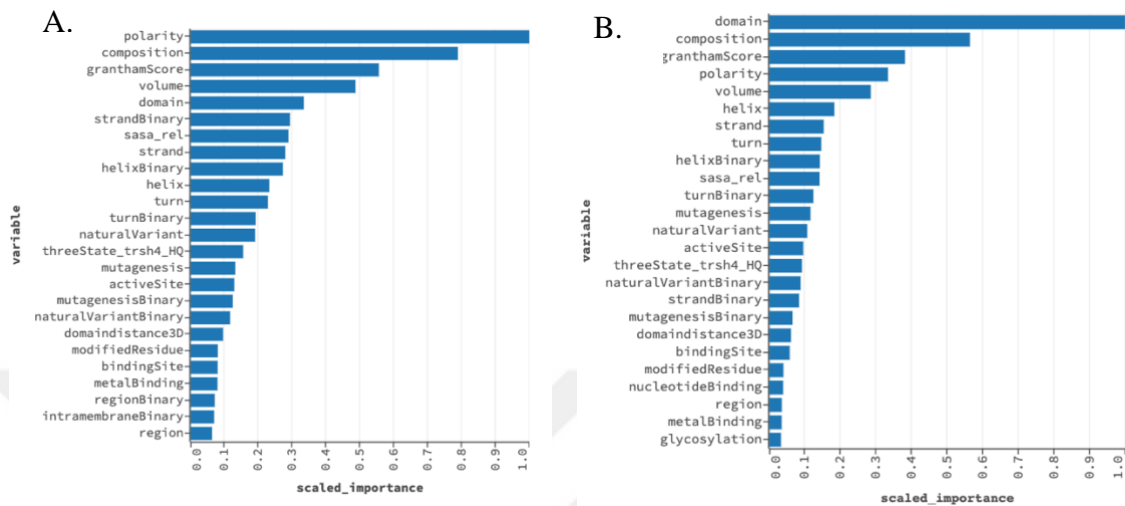


Figure 4.8. Feature importance for all-domains and significant-domain models. (A) shows the feature importance ranking for all-domains case. Domain feature is located at the 4th rank. (B) shows the feature importance ranking for significant-domains case. Domain feature is raised to the first rank.

When two models are compared, we observe an increase in the performance metrics of the second model. Also, we can observe the increased importance of domains as the complexity is reduced for domain feature and redundant domains are eliminated, leaving only most informative domains. For the further analysis, we used statistically significant domains for our models due to the increased performance.

4.8.2. Evaluating the Effect of Physicochemical Features

Physicochemical features are shown to increase the predictive performance of the classifiers that aim to distinguish between neutral and deleterious mutations. In our preliminary analyses, we also observed that physicochemical features rank in the higher rows of feature importance. Thus, we wanted to compare two models with and without physicochemical features from our feature vector and see their effect on an individual basis. Models are built with default parameters again. 50 trees generated with 25 set as their maximum depth. 10-fold cross-validation is applied on the data which is split as training and independent hold-out validation sets, respectively.

Confusion matrix results are shown in Table 4.8 and Table 4.9 for model without physicochemical properties and with physicochemical properties, respectively. Neutral output is evaluated as the negative class. In the first analysis where we excluded physicochemical features from the feature vector, we have 6,875 true negatives, 5,712 false positives, 1,985 false negatives and 15,172 true positives. Recall is calculated as 0.88, while the precision is 0.73 and specificity is 0.55. F1 score is reported as 0.80. In the

second analysis where we included physicochemical properties, we have recorded 7,947 true negatives, 4,720 false positives, 1,942 false negatives and 15,244 true positives. Recall is calculated as 0.89, while the precision is 0.76 and specificity is 0.63. F1 score is reported as 0.82. Other metrics derived from precision and recall values are reported in Table 4.10 for both analyses.

Table 4.8. Confusion matrix for the hold-out validation set of the model without physicochemical properties. Rows: Actual Class; Columns: Predicted Class

	Predicted neutral class	Predicted deleterious class
Actual neutral class	6875	5712
Actual deleterious class	1985	15172
Total	8860	20884

Table 4.9. Confusion matrix for the hold-out validation set of the model with physicochemical properties. Rows: Actual Class; Columns: Predicted Class

	Predicted neutral class	Predicted deleterious class
Actual neutral class	7947	4720
Actual deleterious class	1942	15244
Total	9889	19964

Table 4.10. Performance metrics for models with and without physicochemical properties.

Measure	Model without Physicochemical Properties	Model with Physicochemical Properties
Sensitivity	0.8843	0.8870
Specificity	0.5462	0.6274
Precision	0.7265	0.7636
Negative Predictive Value	0.7760	0.8036
False Positive Rate	0.4538	0.3726
False Discovery Rate	0.2735	0.2364
False Negative Rate	0.1157	0.1130
Accuracy	0.7412	0.7768
F1 Score	0.7977	0.8207
Matthews Correlation Coefficient	0.4651	0.5401
AUC	0.8100	0.8600

ROC curves for validation sets for both analyses are shown in Figure 4.9. According to this, validation AUC value is reported as 0.81 for the model where physicochemical properties are excluded, and 0.86 as for the model where physicochemical properties are included.

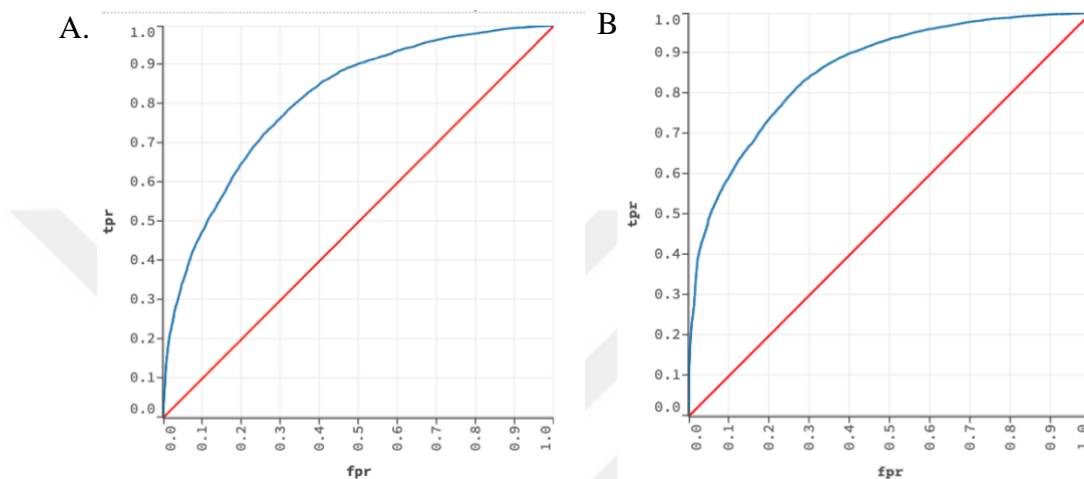


Figure 4.9. ROC curves for the hold-out validation sets of the models (A) where physicochemical properties are excluded, and (B) where physicochemical properties are included. AUC values are 0.81 and 0.86 respectively.

Feature importance figures can be found in Figure 4.10. According to this, in the first model that is built by excluding physicochemical features, first ranks are occupied by domain and other sequence features (Figure 4.10A). In the other model that is built by using all physicochemical properties, i.e. polarity, volume, composition and Grantham Matrix score, the rank of these features is the highest after the domain feature (Figure 4.10B). This is expected as physicochemical properties are shown to be strong determinant in all of the analysis. When the figures are examined, it is observed that in the case of model without physicochemical features, the scaled importance of other features increases for most of them compared to the first model. Thus, we can conclude that when present, physicochemical domains dominate the other features as their predictive power is higher and decreases the importance of other features such as sequence features.

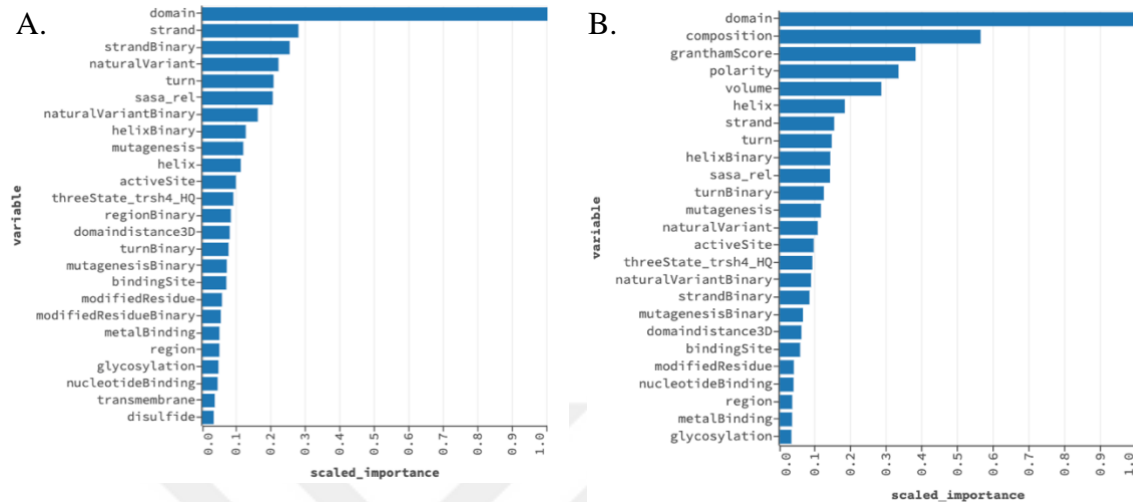


Figure 4.10. Feature importance models with and without physicochemical properties. (A) shows the feature importance ranking for model with physicochemical properties are excluded. All 4 properties are located in the highest ranks. (B) shows the feature importance ranking model with physicochemical features.

When two models are compared, we can observe a significant difference in the performance between two models. The model in which physicochemical properties are included shows a greater performance compared to the model that does not use these features. These results show the importance of incorporation of physicochemical features into the predictive models in accordance with the literature. Mutations cause deleterious changes because they alter their environment, and the scale of the effect is proportional to the degree of change they show. Thus, the reason why these properties carry such a big importance could be attributed to the fact that they directly characterize the mutations.

4.8.3. Evaluating the Effect of Spatial Distances to Protein Feature Annotations

Finally, we wanted to examine the effect of sequence annotations on the performance of predictors. For this reason, we again built different two models. In one of them, we included all the features and in the second one, we excluded 3D distance features for sequence annotations. Since our purpose is to compare two models, we used default parameters. 50 trees generated with 25 set as their maximum depth. 10-fold cross-validation is applied on the data which is split as 75%-25% as training and independent hold-out validation sets, respectively.

Confusion matrix results are shown in Table 4.11 and Table 4.12 for model without 3D distance values and with 3D distance values for sequence annotations, respectively. Neutral output is evaluated as the negative class. According to this, in the analysis we performed without 3D distance values for sequence annotations from the feature-vector, we have 7,131 true negatives, 5,456 false positives, 1,803 false negatives and 15,354 true

positives. Recall is calculated as 0.89, while the precision is 0.74 and specificity is 0.57. F1 score is reported as 0.81. On the other hand, for the second analysis where we included physicochemical properties we have 7,947 true negatives, 4,720 false positives, 1,942 false negatives and 15,244 true positives. Recall is calculated as 0.89, while the precision is 0.76 and specificity is 0.63. F1 score is reported as 0.82. Other metrics derived from precision and recall values are reported in Table 4.13 for both analyses.

Table 4.11. Confusion matrix for the hold-out validation set of the model without 3D sequence annotations. Rows: Actual Class; Columns: Predicted Class

	Predicted neutral class	Predicted deleterious class
Actual neutral class	7131	5456
Actual deleterious class	1803	15354
Total	8934	20810

Table 4.12. Confusion matrix for the hold-out validation set of the model with 3D sequence annotations. Rows: Actual Class; Columns: Predicted Class

	Predicted neutral class	Predicted deleterious class
Actual neutral class	7947	4720
Actual deleterious class	1942	15244
Total	9889	19964

Table 4.13. Performance metrics for models without and with 3D sequence annotations.

Measure	Model without 3D Sequence Annotation distances	Model with 3D Sequence Annotation distances
Sensitivity	0.8949	0.8870
Specificity	0.5665	0.6274
Precision	0.7378	0.7636
Negative Predictive Value	0.7982	0.8036
False Positive Rate	0.4335	0.3726
False Discovery Rate	0.2622	0.2364
False Negative Rate	0.1051	0.1130
Accuracy	0.7560	0.7768
F1 Score	0.8088	0.8207
Matthews Correlation Coefficient	0.4973	0.5401
AUC	0.8400	0.86

ROC curves for validation sets for both analyses are shown in Figure 4.11. According to this, AUC value is reported as 0.84 for the model where 3D distances of annotations are excluded, and as 0.86 for the where 3D distances of annotations are included.

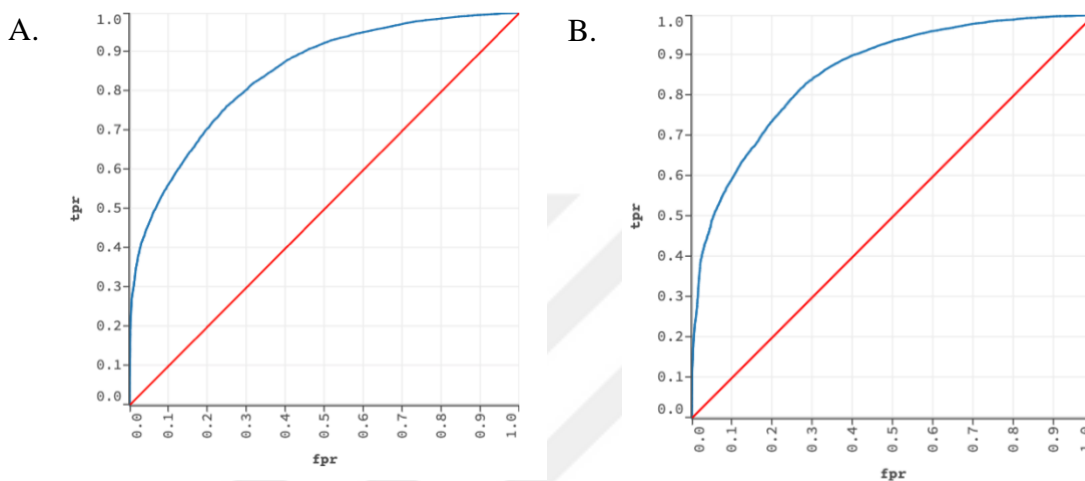


Figure 4.11. ROC curves for the hold-out validation sets of the models (A) where 3D distances of annotations are excluded, and (B) where 3D distances of annotations are included. AUC values are 0.84 and 0.86 respectively.

As for feature importance figures, Figure 4.12 shows feature importance for two analysis. According to this, in the first model that is built by excluding 3D distances of annotations, we can observe that higher ranks are occupied by physicochemical feature as expected. Binary annotations follow physicochemical properties. Helix, strand and turn annotations occupy higher ranks compared to other annotation types. As for the model where 3D distances are included, the same three annotations, helix, strand and turn, are shown to occupy higher ranks with their 3D distance values; again, after physicochemical properties. Although 3D distance annotation values are scattered in terms of feature importance, from the output results we can conclude that they play an important role in the predictive performance of the model.

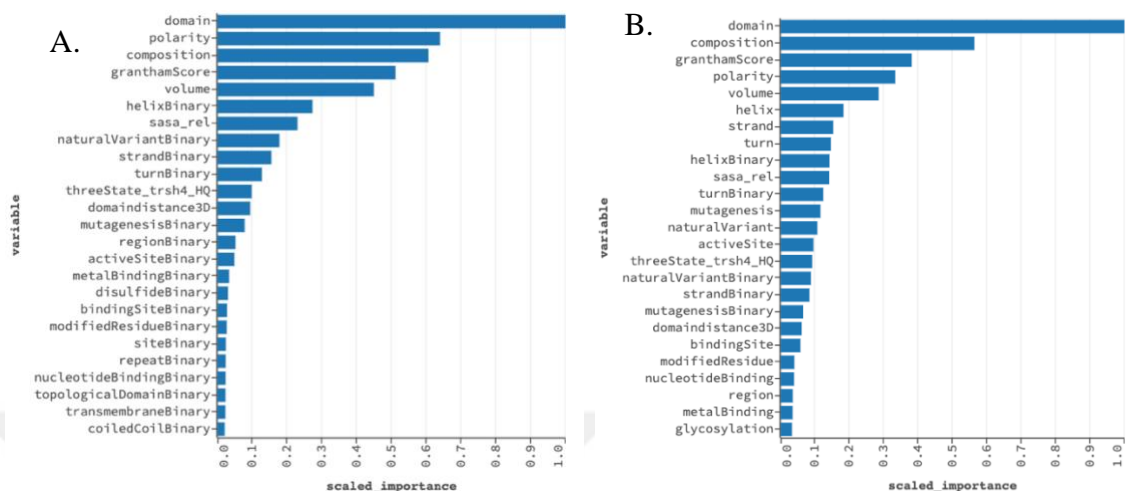


Figure 4.12. Feature importance models with and without physicochemical properties. (A) shows the feature importance ranking for model without physicochemical properties are excluded. All 4 properties are located in the highest ranks. (B) shows the feature importance ranking model with physicochemical features.

These results show that, 3D annotations are valuable parts of the feature vector and their inclusion can enhance the predictive performance. With a similar logic to the domain case, we can conclude that mutations occurring in the functionally important regions are distinctive in terms of determining the possible effect of the mutation as they alter their surroundings as well.

4.8.4. Performance of The Finalized Classifier

After measuring the individual contributions of different subsets of the feature vector and observing the importance of each subset alone, we constructed a model with the final feature vector that contains all of the features tested. Since using only statistically significant domains increased the performance slightly and also reduced the complexity, we decided to use this reduced data set for further analysis. Final feature vector that is used to build the model contains 68-features. In order to determine the best parameters and optimize our model we performed a grid search over the space of possible hyperparameter values. We did not tune all the parameters due to computational limitations, rather we adjusted a subset of hyperparameters that could be important in the tree generation and model performance. Tested hyperparameters are given in Table 4.14. Best performing parameters as a result of the grid search are shown in bold. Full list of parameters can be found in Appendix D.

Table 4.14. Parameters for grid search for the finalized model. Best performing ones are shown in bold.

ntrees	25	50	75	100	125	150	175
max_depth	10	15	20	25	30		
balanced_class	yes	no					

ntrees parameter shows the number of trees built for the forest. Final decision is determined as a common decision obtained from 100 trees. max_depth parameter stands for maximum depth for the tree. This means, a tree can be branched up to these many levels. As the depth increases, it generates more splits and contains more information about the data. This hyperparameter is selected to be set to 25. Also, we can observe that a better score is obtained when the classes are balanced. When this parameter is set True, algorithm balances the contribution from two response classes by either oversampling the less abundant one, or under sampling the more abundant one. In the data, we had 68,831 data points belonging to non-neutral case, and 50,238 data points belonging to neutral case. Although the difference between two classes is not very much, grid search resulted in a better model for the balanced case.

Using the parameters that resulted in the best scores from Table 4.14 and default values for the rest of the hyperparameters, we have performed 10-fold cross-validation on our data. 75% of data is split for training; while the rest is saved as an independent hold-out validation set, respectively.

Confusion matrix for the hold-out validation set is given in Table 4.15. According to this, we have 7,586 true negatives, 4,791 false positives, 1,759 false negatives and 15,677 true positives. Recall is calculated as 0.90, while the precision is 0.77 and specificity is 0.61. F1 score is reported as 0.83. Other metrics derived from precision and recall values are reported in Table 4.16.

Table 4.15. Confusion matrix for the hold-out validation set of the finalized model.

Rows: Actual Class; Columns: Predicted Class

	Predicted neutral class	Predicted deleterious class
Actual neutral class	7586	4791
Actual deleterious class	1759	15677
Total	9345	20468

Table 4.16. Performance metrics for the best model of grid search for the finalized model.

Measure	Value
Sensitivity	0.8991
Specificity	0.6129
Precision	0.7659
Negative Predictive Value	0.8118
False Positive Rate	0.3871
False Discovery Rate	0.2341
False Negative Rate	0.1009
Accuracy	0.7803
F1 Score	0.8272
Matthews Correlation Coefficient	0.5439
AUC	0.86

Resulting ROC after building the final model is given in Figure 4.13. ROC curve demonstrated an AUC value of 0.86 for the hold-out validation set.

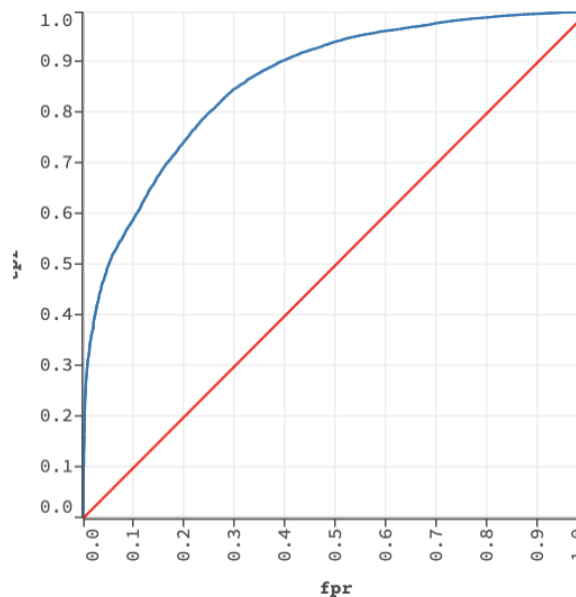


Figure 4.13. ROC curve for the hold-out validation set after grid search. AUC = 0.86. Blue line represents true positive rate and false positive rate at different threshold values, red line represents a random prediction.

As for variable importance, we can see that the highest-ranking variables are again domain and physicochemical properties when all of the features are included. Strand among strand, helix and turn annotations that are seen in higher ranks in the previous analysis is again observed to carry higher importance.

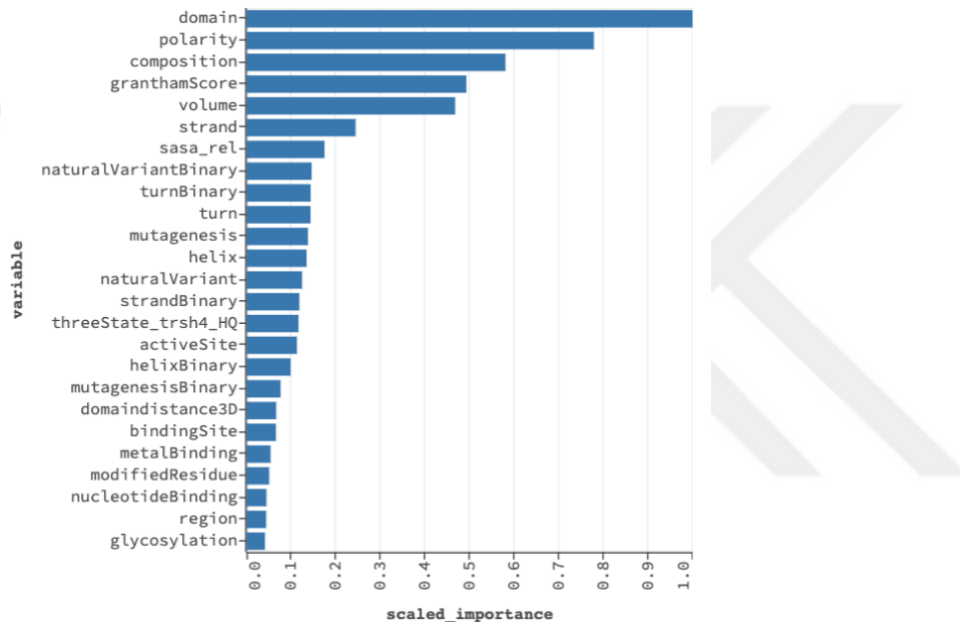


Figure 4.14. Scaled variable importance for the features for the best model of finalized vector. Physicochemical properties are shown to contribute the most.

Finally, Figure 4.15 shows a log-loss curve for the best performing model from the grid search. Blue line represents the training set, while orange line represents validation set. The logarithmic loss curves give a measure of the closeness of a model's predicted values to the actual target value. In other words, it measures the accuracy of the classifier by penalizing false classifications. An algorithm's goal is to minimize this rate. As we can see for our case, as the number of trees increases log-loss value decreases. One can notice that error-rate is lower for the validation case while in fact a lower error is expected for the training set. It could be deceptive at the first glance because the formula for calculating log-loss takes the negative of the value. Since logarithmic formula changes sign between 0 and 1, log-loss formula outputs the higher value as a lower value.

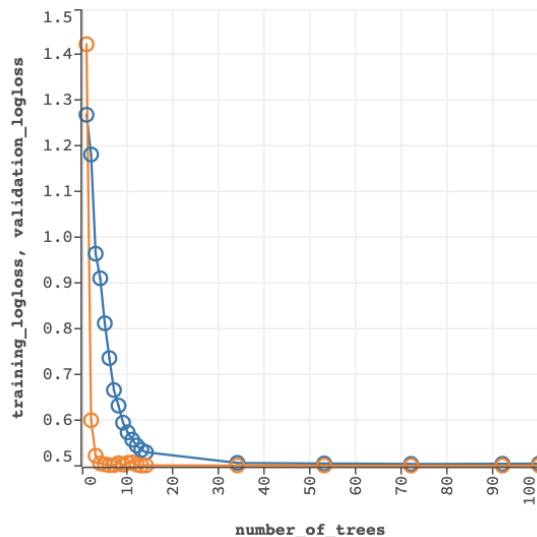


Figure 4.15. Log-loss graph for the best model after the grid search. Orange line shows validation set, blue line shows training set.

When results from four analyses are examined, we can see that a feature vector where all of the features from structure and sequence is included performs the best for optimized parameters. We obtained an AUC value of 0.86 for the final model.

After obtaining the results for the final classifier, we also wanted to observe the performance of our classifier with different subsets from our data. For our model, we have retrieved data from three sources; UniProt, ClinVar and PMD. UniProt and PMD deposits mutations that cause a deleterious effect on the protein’s structure or function. However, ClinVar deposits disease data that are discovered in clinical or experimental studies. For this reason, in order to observe how the classifier would behave in two different data sets with different characteristics we split our data into two sets as one having only ClinVar mutations and the other one having UniProt and PMD mutations. 75% of the data is divided as the training set and the rest is saved as the hold-out validation set. 10-fold cross-validation is applied on the data. In order to obtain a better performance number of trees are set to 150, and maximum depth is set to 25. Confusion matrices for the hold-out validation sets of two datasets are given in Table 4.17 and Table 4.18, respectively. According to this, in the model trained with UniProt and PMD data, we have 5,709 true negatives, 4,501 false positives, 1,246 false negatives and 9,966 true positives. Recall is calculated as 0.89, while the precision 0.69 and specificity is 0.56. F1 score is reported as 0.78. Other metrics are reported in Table 4.19. On the other hand, in the model trained with ClinVar data only, we have 1,972 true negatives, 1,253 false positives, 593 false negatives and 8,282 true positives. Recall is recorded as 0.93, while precision is 0.87 and specificity is 0.61. F1 score is calculated as 0.90. Other metrics can be found in Table 4.19.

Table 4.17. Confusion matrix for the hold-out validation set of the model trained with UniProt and PMD data. Rows: Actual Class; Columns: Predicted Class

	Predicted neutral class	Predicted deleterious class
Actual neutral class	5709	4501
Actual deleterious class	1246	9966
Total	6955	14467

Table 4.18. Confusion matrix for the hold-out validation set of the model trained with ClinVar data. Rows: Actual Class; Columns: Predicted Class

	Predicted neutral class	Predicted deleterious class
Actual neutral class	1972	1253
Actual deleterious class	593	8282
Total	2565	9535

Table 4.19. Performance metrics for models that use UniProt-PMD data and ClinVar data.

Measure	Model without UniProt-PMD data	Model with ClinVar Data
Sensitivity	0.8889	0.9332
Specificity	0.5592	0.6115
Precision	0.6889	0.8686
Negative Predictive Value	0.8208	0.7688
False Positive Rate	0.4408	0.3885
False Discovery Rate	0.3111	0.1314
False Negative Rate	0.1111	0.0668
Accuracy	0.7317	0.8474
F1 Score	0.7762	0.8997
Matthews Correlation Coefficient	0.4779	0.5892
AUC	0.82	0.91

ROC curves for validation sets for both analyses, dataset for ClinVar and dataset for UniProt and PMD, are shown in Figure 4.16. According to this, AUC value is reported as 0.82 for the model with UniProt and PMD data combined, while AUC is 0.91 for the model is trained with ClinVar data.

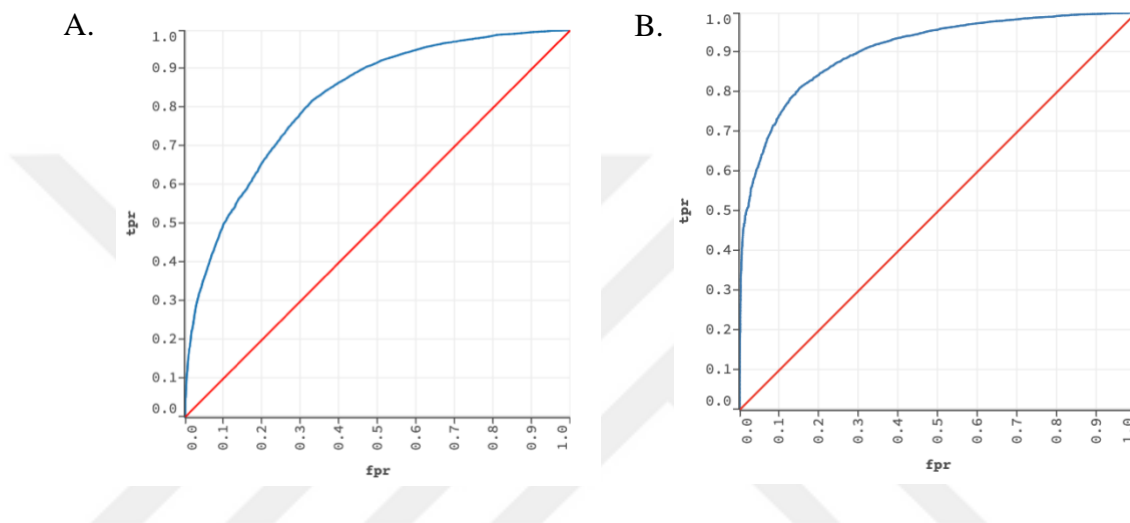


Figure 4.16. ROC curves of the hold-out validation sets of models trained on (A) Humsavar and PMD data (B) ClinVar data. AUC values are 0.82 and 0.91 respectively.

As for feature importance figures, Figure 4.17 shows feature importance for two analysis. In the first model that is built with UniProt and PMD data, surprisingly, contrary to the other analysis physicochemical properties are seen in the lower ranks. Higher ranks are occupied by strand, helix and turn annotations from UniProt annotations and only by Grantham Matrix Score from physicochemical properties. Domains are again observed in the higher ranks. Relative accessible surface area value is also surprisingly in the high ranks. And in the second model where ClinVar data is used for training and validating the model, higher ranks are occupied by physicochemical properties as expected. They carry the most importance as the scaled importance for the other ones are low. On contrary in the first model, scaled importance are distributed more equally compared to this model.

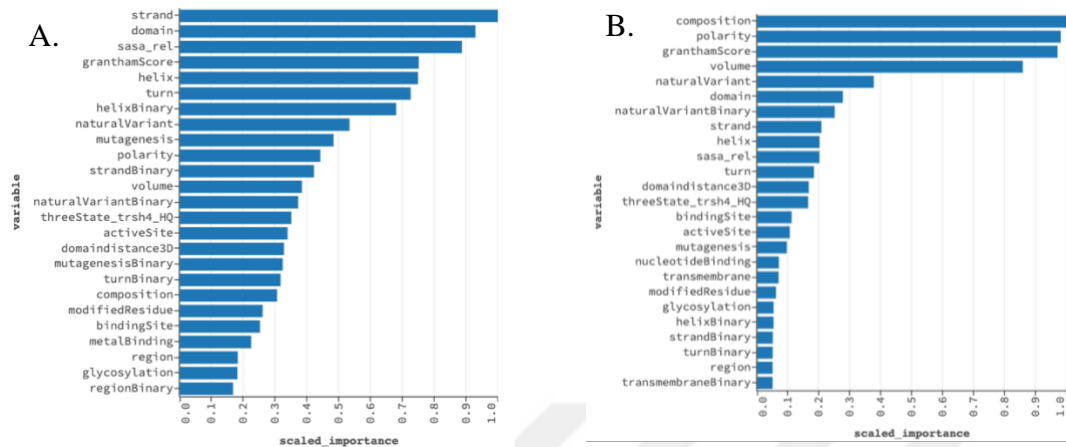


Figure 4.17. Feature importance models trained and validated on (A) Humsavar and PMD data (B) ClinVar data.

4.9. Benchmark Analysis

In order to compare our results with the performances of existing methods, we've done a benchmark analysis. For this purpose, we have used data from PredictSNP (Bendl *et al.*, 2014). PredictSNP is an ensemble classifier that gives a consensus prediction from 6 different classifiers, each use a different set of data points and methods. PredictSNP follows a very elaborative method while constructing its benchmark data set. It collects the data from other sources that are not used as the branches of the ensemble classifier and removes data that are found common in the training or test sets of the selected classifiers. By doing so, it ensures a fully independent data set that is able to measure the performance of each individually. However, one thing prevents PredictSNP from totally reflecting the individual unbiased performances is that classifiers are not able to accommodate all the data points in the benchmark data set. Some classifiers for instance require proteins to be deposited in certain databases, and when proteins from the data is not present in that database, this prevents their evaluation by the mentioned classifier. Despite that, even using some portion of this unbiased data gives a measure of understanding the performances. We have compared the performance of our model with the results from this ensemble classifier, as well as its selected individual classifiers on three different datasets.

Datasets were available to download in PredictSNP. There are three datasets available to download in PredictSNP. The benchmark dataset is used for the evaluation of the selected classifiers and also for training of PredictSNP. It is compiled from five different sources. One of the testing sets were compiled from mutations on Protein Mutant Database. Finally, the second testing test, and the third of three datasets, was compiled from experiments conducted by Yampolsky and Stoltzfus for massively mutated proteins (MMP); and from two patent applications issued by Danisco Inc. describing the effect of

mutations on serine protease from *Bacillus subtilis* and alpha-amylase from *Geobacillus stearothermophilus* (Aehle W., Wolfgang, Cascao-Pereira Luis G., Estell David A., Goedegebuur Frits, Kellis, Jr. James T., Poulouse Ayrookaran J., 2010; Cuevas William A, Lee Sang-Kyu, Ramer Sandra W, Shaw Andrew, Topozada Amr R, Estell David E, 2009; Quan et al., 2016; Yampolsky & Stoltzfus, 2005). Data counts for individual data sets can be found in Table 4.20.

Table 4.20. Data point counts for the datasets downloaded from PredictSNP.

	Neutral	Deleterious	Total
PredictSNP benchmark dataset	24082	19800	43882
PMD dataset	1248	2249	3497
MMP dataset	7538	4452	11990

In order to be able to use these data in our model, we retrieved the common data points that are present both in PredictSNP data sets and our dataset. This has resulted in 15491 data points common with the benchmark set, and 952 data points common with the PMD test set. Later on, to construct the training and test sets for the model, we have extracted these common data points from our data set and created training and test sets. For each evaluation, remaining data points were used as the training set, while the excluded data points, i.e. common between PredictSNP data and the original dataset used in this study, are considered as the validation set. Additionally, in order to give fairer results, we have only used a subset of our data that includes the mutations from the same versions of UniProt, ClinVar and PMD used in PredictSNP data sets. This data set is referred as Dataset I in the following sections. According to this, counts of per training and test sets for each data set are given in Table 4.21.

Table 4.21. Data points counts for the benchmarking analysis.

	Neutral	Deleterious	Total
Dataset I	38874	44177	83051
Benchmark excluded from Dataset I (Train)	29735	40227	69962
Excluded Benchmark Set (Test)	9139	3950	13089
PMD excluded from Dataset I (Train)	38539	43563	82102
Excluded PMD (Test)	335	614	949
MMP excluded from Dataset I (Train)	47553	64434	111987
Excluded MMP (Test)	5	18	23

We have created three models for evaluation of each data set. In the first model, we have excluded data points that are common to PredictSNP’s benchmark dataset and our original dataset. We again used significant domains for the domain feature in our model. This has resulted in a training set of 69,962 mutations and a test set of 13,089. Number of trees are set to 150 and maximum depth is set to 20. Confusion matrix for the hold-out validation set is given in Table 4.22 and performance metrics obtained using these sets are given in Table 4.23. Using this dataset, our model obtained a recall value of 0.77, while the precision is 0.66 and the specificity is 0.83. F1 score is found to be 0.71 while the accuracy is 0.81. We can also see that another measure Matthews Correlation Coefficient is 0.58.

Table 4.22. Confusion matrix for the hold-out validation set for PredictSNP benchmark data. Rows: Actual Class; Columns: Predicted Class

	Predicted neutral class	Predicted deleterious class
Actual neutral class	7583	1556
Actual deleterious class	906	3044
Total	8489	4600

Table 4.23. Performance metrics for the best model for PredictSNP benchmark set.

Measure	Value
Sensitivity	0.7706
Specificity	0.8297
Precision	0.6617
Negative Predictive Value	0.8933
False Positive Rate	0.1703
False Discovery Rate	0.3383
False Negative Rate	0.2294
Accuracy	0.8119
F1 Score	0.7120
Matthews Correlation Coefficient	0.5772
AUC	0.8657

ROC curve generated over all thresholds for this model is given in Figure 4.18. According to this, independent validation set obtained an AUC value of 0.87.

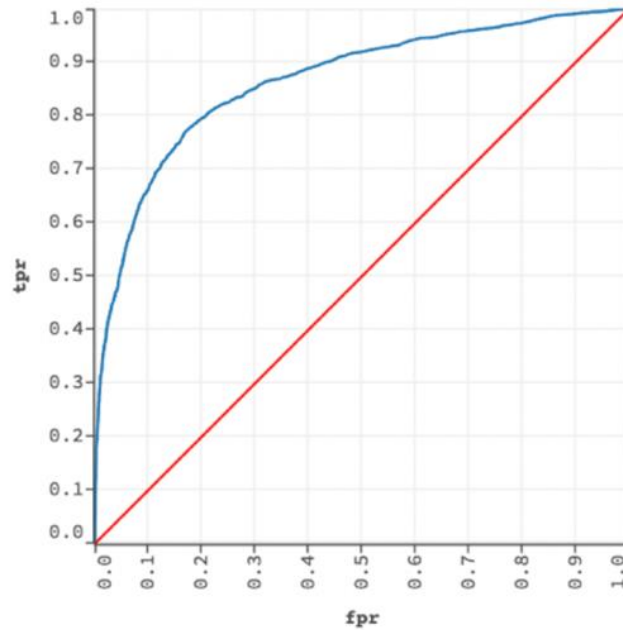


Figure 4.18. ROC curve for the hold-out validation set of the model generated using PredictSNP benchmark dataset. AUC = 0.87. Blue line represents true positive rate and false positive rate at different threshold values, red line represents a random prediction.

Another model is generated with the data set that is constructed by excluding data points that are common with PMD test set of PredictSNP. According to this, we have 82,102 datapoints in the training set and 949 data points in the test set. In order to get better results, we have refined the parameters as number of trees equals to 200 and maximum depth is equal to 30. Confusion matrix and performance metrics are given in Table 4.24 and 4.25, respectively. For this model, we have obtained a recall value of 0.98, while the precision is 0.66 and the specificity is 0.095. F1 score is found to be 0.80 while the accuracy is 0.67. We can also see that another measure Matthews Correlation Coefficient is 0.17.

Table 4.24. Confusion matrix for the hold-out validation set for PredictSNP PMD test data. Rows: Actual Class; Columns: Predicted Class

	Predicted neutral class	Predicted deleterious class
Actual neutral class	32	303
Actual deleterious class	13	601
Total	45	904

Table 4.25. Performance metrics for the best model for PredictSNP PMD test set.

Measure	Value
Sensitivity	0.9788
Specificity	0.0955
Precision	0.6648
Negative Predictive Value	0.7111
False Positive Rate	0.9045
False Discovery Rate	0.3352
False Negative Rate	0.0212
Accuracy	0.6670
F1 Score	0.7918
Matthews Correlation Coefficient	0.1672
AUC	0.7101

ROC curve generated over all thresholds for this model is given in Figure 4.19. According to this, independent validation set obtained an AUC value of 0.71.

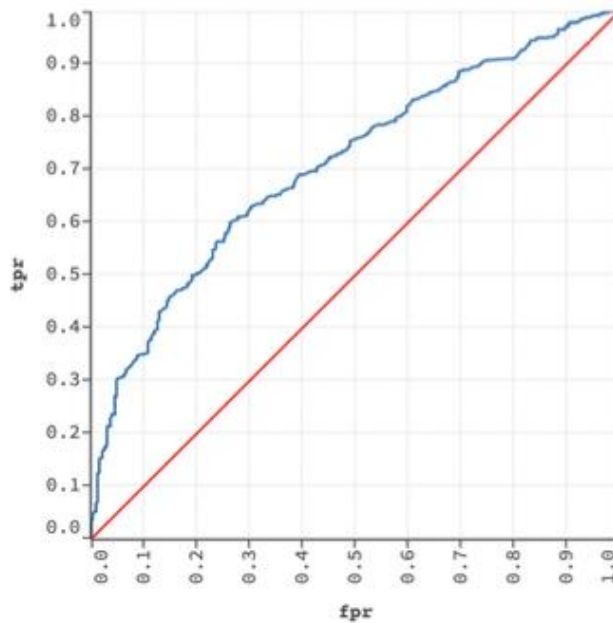


Figure 4.19. ROC curve for the hold-out validation set of the model generated using PMD dataset. AUC = 0.71. Blue line represents true positive rate and false positive rate at different threshold values, red line represents a random prediction.

As we can see, the metrics for this model is not as good as the previous one. This can owe to the fact that the validation set is not enough to reflect the real capacity of the model as it contains really a few number of data points. Most of the data points are predicted as deleterious, suggesting a bias in the data set. When the counts for individual classes are examined, we can observe that counts for deleterious mutations are double as neutral ones. Even though balanced classes parameter is activated in the model, it still seems to affect the performance.

Finally, we have generated the last model using MMP set of PredictSNP. MMP validation set is not reliable as it only contains 23 datapoints. To recall, these data points are the ones that are found in common with our original set of data. Remaining data points that are only present in our data remained as the training set.

Training set for MMP model contains 11,1987 data points and validation set contains 23 data points. Confusion matrix and performance metrics are given in Table 4.26 and Table 4.27, respectively. For this model, we have obtained a recall value of 1.0, while the precision is 0.82 and the specificity is 0.20. F1 score is found to be 0.90 while the accuracy is 0.83. We can also see that another measure Matthews Correlation Coefficient is 0.40.

Table 4.26. Confusion matrix for the hold-out validation set for PredictSNP MMP test data. Rows: Actual Class; Columns: Predicted Class

	Predicted neutral class	Predicted deleterious class
Actual neutral class	1	4
Actual deleterious class	0	18
Total	1	22

Table 4.27. Performance metrics for the best model for PredictSNP MMP test set.

Measure	Value
Sensitivity	1.000
Specificity	0.2000
Precision	0.8182
Negative Predictive Value	1.000
False Positive Rate	0.8000
False Discovery Rate	0.1818
False Negative Rate	0.0000
Accuracy	0.8261
F1 Score	0.9000
Matthews Correlation Coefficient	0.4045
AUC	0.5101

ROC curve generated over all thresholds for this model is given in Figure 4.20. According to this, independent validation set obtained an AUC value of 0.51.

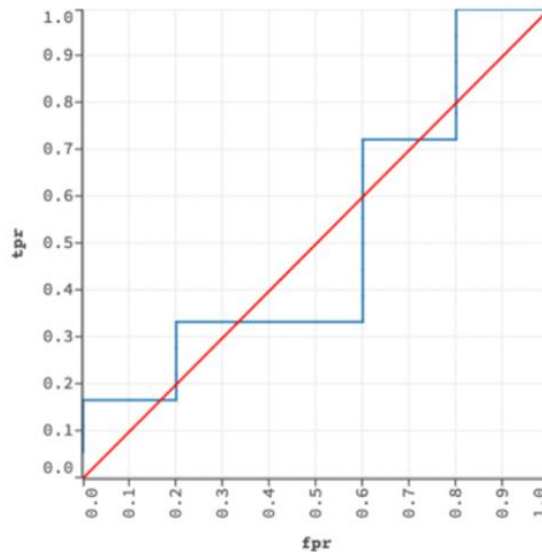


Figure 4.20. ROC curve for the hold-out validation set of the model generated using MMP dataset. AUC = 0.51. Blue line represents true positive rate and false positive rate at different threshold values, red line represents a random prediction.

As can be seen from the ROC curve generated for MMP analysis, it is close to random predictions. This is mainly due to the effect of not having enough data points in the validation set. Only 23 data points cannot show the real evaluation of the classifier, thus we obtain such a result for MMP case. Also, we cannot observe any learning by the classifier which makes the predictions almost randomly assigned. To compare the effect of not having enough data points, we can take a look at the closest example which is PANTHER. In Table 4.25, we can see that PANTHER classifier is able to evaluate 61.9% of the mutations. This ratio is the lowest among all other classifiers. When we compare the metrics for this data set with the other classifiers, it is seen that PANTHER shows the lowest performance metrics for this set. This observation shows the importance of data availability and how a good representation can improve the performance.

Overall comparison of performance metrics with other methods are given in Table 4.28. According to this, as expected consensus classifier outperformed the individual classifiers in many occasions. When the dataset is MMP, our model showed a poor performance due to the reasons mentioned above. However, with PMD test set and the benchmark set is

concerned our model has shown comparable or better results in most of the cases. Our model's AUC value for benchmark set is 0.87 while the best score was given by PredictSNP as 0.81. For PMD test dataset this value is 0.70 for PredictSNP. Our score is similar with a value of 0.71.

Table 4.28. Overall comparison of different classifiers.

Performance Metrics	Dataset	Mapp	nsSNPanalyzer	Panther	PhD-SNP	PPH-1	PPH-2	SIFT	SNAP	PredictSNP	Our Model
Percent of evaluated mutations	PredictSNP	87.8	33.5	54.6	100.0	98.8	100.0	97.1	99.1	100.0	35.3
	PMD	81.1	63.4	38.1	100.0	97.1	98.3	77.6	95.1	100.0	27.0
	MMP	99.8	91.5	61.9	100.0	97.7	97.7	95.4	100.0	100.0	1.98
Accuracy	PredictSNP	0.711	0.632	0.642	0.746	0.682	0.701	0.723	0.670	0.747	0.81
	PMD	0.653	0.629	0.651	0.633	0.654	0.632	0.643	0.631	0.642	0.67
	MMP	0.707	0.618	0.603	0.629	0.684	0.677	0.646	0.709	0.708	0.83
Matthews Correlation Coefficient	PredictSNP	0.423	0.219	0.296	0.494	0.364	0.407	0.447	0.346	0.492	0.58
	PMD	0.327	0.243	0.303	0.258	0.299	0.289	0.312	0.253	0.281	0.17
	MMP	0.400	0.400	0.227	0.255	0.357	0.359	0.308	0.406	0.408	0.40
AUC	PredictSNP	0.773	0.634	0.692	0.812	0.695	0.776	0.784	0.732	0.808	0.87
	PMD	0.695	0.630	0.697	0.676	0.658	0.704	0.685	0.667	0.700	0.71
	MMP	0.759	0.620	0.676	0.685	0.720	0.774	0.710	0.769	0.787	0.50

PPH-1 – PolyPhen-1; PPH-2 – PolyPhen-2; PMD dataset – dataset from Protein Mutant Database; MMP – dataset of massively mutated proteins



CHAPTER 5

5. DISCUSSION & CONCLUSION

Prediction of the effects of single amino acid variations on protein functionality remains an important question in computational biology. Although there are many tools and approaches developed for understanding possible effects, there is still room for improvement and the need for better prediction models continues. In this study, we aimed to develop a methodology for prediction of disease-causing capacity of single amino acid variations by taking a structure and annotation-centric approach where we utilized 30 different UniProt sequence annotations along with three physicochemical properties to enrich our analysis. A total number of 119,069 data points were mapped to relative features and a feature vector of 68 dimensions was created to be fed into the machine learning classifier. Random forest algorithm was used to build the machine-learning model and variant effect predictions were obtained from the trained classifier.

First of all, we observed that mutations occurring in functional regions are more prone to cause deleterious effects. These regions include various UniProt annotation sites, domains, sequence motifs and more. In our study, we selected 30 UniProt annotations and each protein's InterPro domains to be considered as functionally important regions. Firstly, we investigated the relationship between domain regions and the mutation's capacity to cause deleterious effects (Figure 4.1). Our analysis showed that 63% of the mutations occurring within the boundaries of domain regions cause an impairment in protein's function. Remaining 37% is in that regions are found to be neutral mutations. On the other hand, 41% of the mutations that are found in the regions outside of the domain boundaries are shown to cause an effect on the protein. As expected, the ratio is lower for this case as these out-of-domain regions are less likely to be functionally important compared to domain regions. There of course can be other functionally important regions in these out-of-domain regions such as regulatory areas or interacting areas. This is why, we are expecting to see deleterious mutations in these regions, as well; however, with a smaller proportion. Analysis from our data can add to this conclusion. When we look at the graph, we observe another category as no-domains which indicates the data points for which no InterPro domain was found in the version used for this study. Proportions for the nature of mutation shows that 61% of such data points are recorded as having a deleterious effect. These unknown areas that includes deleterious mutations may

be expected to belong to functionally important regions. These findings can help to prioritize sequence and structure annotation efforts for these regions after some more in-depth analysis.

In addition to the domain regions, we have also made use of UniProt sequence annotations. These annotations are included both in the continuous valued form where we calculated the 3D distance between the annotation region and the mutation position, and in the binary form where we checked for the presence or absence of the mutation within the annotated region. In a similar manner we did for the domains, here as well, we expected to observe deleterious mutations more in the annotation regions. Our analysis showed that it was a valid assumption. In the regions of annotation, we observed more deleterious mutations than neutral mutations for most of the annotation types (Figure 4.5). For example, in initiator methionine annotations, all mutations that hit these regions caused a deleterious effect in the protein. Methionine is coded by a single amino acid and it is the initiator codon of protein synthesis for the eukaryotic organisms. Thus, any mutation that creates any product other than methionine distorts the protein synthesis and prevents the protein formation. Therefore, we would expect to see that mutations in this annotation region would be deleterious as confirmed by our analysis. Only few annotation types did not agree with this trend. Signal peptide, coiled coil, peptide, transit peptide, glycosylation and pro-peptide annotations showed a higher proportion for neutral mutations when mutation is found to hit these regions. Coiled coil is normally an important region for protein folding and mutations in this region are expected to impair protein folding; thus function. 16% of difference is observed between two classes for coiled coil. The ratio is not too high and can be a result of the positions of the mutation in the selected data set. Coiled coils are packed alpha helices. Mutations may be located in the inner regions that even when caused distortions their effect may not change the overall behavior of the complex. Glycosylation is again an important secondary protein processing mechanism that affects protein structure and stability. The difference in proportion is 12 %. This is not a dramatically large difference but still the reasons why non-neutral mutations are found to be more present in this annotation type can be investigated. The remaining annotations are related to peptide structures. Same as glycosylation goes for the peptide annotation which differs by 8% from the non-neutral class. Peptide annotation refers to a peptide, as can be understood from the name, and generally carries a biological function. Reasons why, even though small, this 8% excess is present for neutral class can be investigated. Remaining two annotations propeptide and transit peptide shows a bigger difference for neutral and non-neutral classes. For propeptide, we obtained a 30% of difference, while for transit peptide this proportion is 36%. A reasonable conclusion can be made by referring to the functions of these annotations. Pro-peptide is actually cleaved from the protein when it gets to its mature form. For this reason, any functionally important task is not expected from this annotation region. As for transit peptide, their task is to carry proteins to their destined location. This would normally affect the function by not sending the protein to its proper place, however a mutation that affects that batch of proteins is not necessarily found present for another synthesis cycle. These mutations do not directly affect the protein structure, but they affect the carrier part. What this means is proteins can still be carried to their destined locations as long as these mutations are

spontaneous mutations, not hereditary mutations. When the overall picture is examined, mutations are found to cause some sort of impairment for most of the annotations they are found in.

We also investigated the importance and relevance of physicochemical properties in protein regions and in protein families. In order to understand the association of the mutation with the protein region it occurs and its consequence, i.e. disease or neutral case, we applied Fisher's exact test on three protein regions: core, surface and interface (Table 4.3). Our results have shown that in all of these regions change in the Grantham Matrix score is highly associated with the probability of observing deleterious mutations. This means that when a significant change occurs in the Grantham Matrix value, it is very likely that this mutation may cause deleterious effect. As Grantham Matrix is a combination of three other properties, we also wanted to observe the individual effects of these properties. When the results are examined, it can be observed that volume property shows the highest significance in all regions that suggests, again, changes in the volume are highly associated with the deleteriousness capacity of the mutation. This shows us the importance of maintaining protein's 3D structure and spatial organization as drastic changes in the volumes may occur as a result of a substitution by very different residues which changes the overall shape. In the other analysis, we did the same comparison for different protein families. For this purpose, instead of three protein regions we have split the dataset into protein families. Four protein families are selected as ion channel, membrane receptor, enzyme and transcription factor. Other families are grouped as the 'other' category. In each family, the proportion of the deleterious and neutral cases in different significance, i.e. significant change in the property or non-significant change in the property, groups for physicochemical properties are recorded. According to this, in enzyme, ion channel and membrane families, Grantham Score again showed to have the most significant association with the state of the outcome. In transcription factor family volume property surpassed all others with a very high significance. When we examine the individual properties, this time we again observed that volume's significance is higher in all families. These results show the importance of maintenance of structure in proteins. We can hypothesize that in addition to having its own importance, structure can also be important because spatial conformation affects surrounding and its interactions as well.

One other thing we examined was the relative accessibility of a mutation position and the degree of effect it may cause on the protein structure (Figure 4.6). Our analysis for our data has shown that when a mutation is observed in the buried region, it is more than 2-fold likely to be a deleterious mutation. Proportions show that 70% of buried mutations caused an impairment; while remaining 30% is found to show no effect. Interface residues show a similar trend with 68% for non-neutral and 32% for neutral residues. This is expected because these two structure regions are important for protein's ability to function. While buried residues contribute to the stability of the protein, interface regions are responsible for recognizing other proteins and make interactions and complexes with them. Thus, mutations affecting these regions are prone to cause important effects on the protein. Proportions for surface residues are not as dramatic as the other two cases with 49% for neutral and 51% for non-neutral classes.

Finally, we have created our models using random forest algorithm. Models are built in a bottom-up manner in which we evaluated the performance of different subsets of candidate features and determined their relative importance in a full combined vector. Firstly, we evaluated the contribution of domains. For this purpose, we compared two feature vectors. One of them contained a full set of domains that are associated with the data points. The other feature vector only included statistically significant domains that we obtained as a result of Fisher's exact test analysis. These selected domains showed a better association with the resulting outcome, meaning that their presence or absence was strongly linked to the type of outcome. In the original data set, we have 2,159 domains. However, after performing Fisher's exact test 327 significant domains are remained (Appendix A). By doing so, we also wanted to observe the effect of reducing complexity in a high cardinality feature. Normally decision trees are able to handle categorical data; however, most of the computational implementations fail to accurately read categorical data. This fact led us to use H2O platform that can treat categorical data in its natural form. Despite that, high cardinality is not preferred as it may cause model to memorize data. As a result of our analysis, we observed an increase in the performance for reduced domain case. AUC value is increased from 0.84 to 0.86 when significant domains are used (Figure 4.7). This has shown us that using less domains is still informative; additionally, it reduces computational complexity. Thus, it is preferable to build the final model with this set of domains. Another analysis is performed to measure the contribution of physicochemical features. Physicochemical features are shown to be informative in many prediction methods. This is expected as they directly characterize the mutation, thus providing insight about the changes introduced to the affected region. For example, when a large residue is changed with a small one, it is more likely to cause a deleterious effect on the protein because the surroundings will be affected more. In order to observe these effects, we again built two models; one is with physicochemical features included, and the other one is with physicochemical features excluded. The model with the physicochemical models showed a greater performance with an AUC value 0.86 compared to the other model which had an AUC value of 0.81 (Figure 4.9). Finally, we wanted to examine the contribution of 3D distances between sequence annotations and mutation of interest. We built two vectors with one including 3D distance values, and the other one excluding them. Our results have shown us 3D distance values are also contributors to the model performance as AUC value is decreased to 0.84 from 0.86 that was measured with the model that used all of the features (Figure 4.11).

After assessing the individual importance of different feature subsets, we constructed our final vector using all of these candidate features as all of them shown to improve predictive performance. We again used statistically significant domains as their analysis showed comparable results with the case where all of the domains are used. We performed a grid search on the hyperparameter space in order to optimize the model by using best performing parameters. As a result of this, the best performance is observed when 150 trees with maximum depth 20 are generated. A balanced dataset gave better results. This model resulted in an AUC value of 0.86 for the validation set and 0.88 for the training set (Figure 4.13). When feature importance is examined, physicochemical properties are seen

to lead the rankings, followed by 3D distance between strand annotation and mutation which are also shown to rank high in other preliminary analysis (Figure 4.14). Along with strand, helix and turn annotations have also showed to rank in higher degrees in the other preliminary evaluations. All three annotations belong to the secondary structure class which may suggest that mutations occurring in these regions are more likely to be deleterious because these regions are strongly related to protein folding. Thus, interruption of the proper folding may lead to non-functioning protein products; thus, preventing the processes from happening.

After generating the finalized model, we wanted to observe the effects of different data characteristics in our model. For this reason, we divided our data into two groups. One group contained data points from ClinVar whereas the other group was composed of data points from UniProt and PMD. Data deposited in ClinVar is different than the data in UniProt and PMD as ClinVar stores disease mutations whereas the other two store mutations causing a deleterious effect on the protein function, stability or structure. According to the performance metrics, model that is trained and validated on ClinVar data outperformed the model that uses UniProt and PMD data. This is an unexpected observation, because our model is meant to give better predictions on protein functionality and structure due to the feature set it utilizes rather than implicating a disease. However, the results show otherwise (Table 4.19). This observation could be a result of the lack of variation in ClinVar data set. ClinVar holds a more homogenous and more predictable data, whereas for UniProt and PMD is very heterogenous. Thus, this makes the validation set easy to predict. We think that this could lead to a dataset that is harder to predict in the case for Humsavar-PMD data.

Finally, we wanted to compare our model with other classifiers with a benchmark analysis. For this purpose, we have used benchmark and test sets from the PredictSNP consensus classifier study. PredictSNP uses these three independent datasets to compare the performances of its individual constituting classifiers as well as its own performance. However, one thing that prevents PredictSNP analysis from being completely accurate is the fact that individual classifiers that made up the consensus can evaluate only a portion of the benchmarking data sets (Table 4.28). Other parts of the data remain unused by the classifiers, and this restrains a fully fair evaluation. In our study, we evaluated the mutations that were also present in our dataset. Evaluation results suggest that our model showed a better or comparable performance than the ensemble classifier for different datasets. For example, for benchmarking set, our model performed better than the consensus classifier with an AUC value of 0.87; while the same metric measures 0.80 for PredictSNP (Figure 4.18). For PMD test set, we obtained a comparable result with the PredictSNP. However, number of the false positives are higher than expected that suggests we have more room for improvement. Finally, for the last dataset which consists of experimentally massively mutated proteins, our model did not give satisfying results. The reason for this is because we did not have enough data points in the validation set to reflect the learning outcomes of the data. When we examine one other individual classifier of PredictSNP, which is PANTHER, we can see that its coverage as well as its performance for this data set is lower than the other classifiers. The same reason explained for our data

is also valid here and shows the importance of a good coverage. One other reason for low performance could be the difference between data points characteristics between two sets. In order to overcome this, datasets can be further fine-tuned to represent a more similar nature. Overall, our model obtained comparable results with a consensus classifier PredictSNP which gives predictions from multiple classifiers. Considering our model only uses its own methodology for prediction, we can deem our model as a successful one.

As a conclusion, we have developed a methodology to obtain variant effect predictions for single amino acid substitutions. Both protein structure and sequence information are used in the feature vectors. In our performance analyses, we observed that our classifier gave sufficiently well results. Also comparing with some individual and consensus classifiers, we have seen that our model shows comparable or better results when enough data is available. Future works of this study includes the construction of a computational tool that will automatize feature vector generation steps, in order to produce predictions on newly introduced data using our model. We plan to enrich the missing data by incorporating new protein functional annotations as they are added to the newer versions of the UniProt database, to increase the predictive performance further. We will also improve our benchmarking results by comparing our model with different benchmark sets such as the one given in Sarkar *et al* 2019). Finally, we plan to optimize our model to work as a meta-predictor in coherence with other well-established tools such as PolyPhen-2 and SIFT, to further increase the predictive performance of the existing state-of-the-art variant effect predictors. Later, this meta-predictor can be implemented in a webserver with a user interface to produce real-time predictions for newly reported mutations with unknown consequence.

REFERENCES

- Abe, S., Ueno, T., & Watanabe, Y. (2009). Artificial metalloproteins exploiting vacant space: Preparation, structures, and functions. In *Topics in Organometallic Chemistry*. https://doi.org/10.1007/978-3-540-87757-8_2z
- Aboderin, A. A. (1971). An empirical hydrophobicity scale for α -amino-acids and some of its applications. *International Journal of Biochemistry*. [https://doi.org/10.1016/0020-711X\(71\)90023-1](https://doi.org/10.1016/0020-711X(71)90023-1)
- Adzhubei, I. A., Schmidt, S., Peshkin, L., Ramensky, V. E., Gerasimova, A., Bork, P., Kondrashov, A. S., & Sunyaev, S. R. (2010). A method and server for predicting damaging missense mutations. *Nature Methods*, 7(4), 248–249. <https://doi.org/10.1038/nmeth0410-248>
- Aehle W., Wolfgang, Cascao-Pereira Luis G., Estell David A., Goedegebuur Frits, Kellis, Jr. James T., Poulou Ayrookaran J., S. B. F. (2010). *Compositions and methods comprising serine protease variants*.
- Ahmed, S., Thomas, G., Ghossaini, M., Healey, C. S., Humphreys, M. K., Platte, R., Morrison, J., Maranian, M., Pooley, K. A., Luben, R., Eccles, D., Evans, D. G., Fletcher, O., Johnson, N., Dos Santos Silva, I., Peto, J., Stratton, M. R., Rahman, N., Jacobs, K., ... Easton, D. F. (2009). Newly discovered breast cancer susceptibility loci on 3p24 and 17q23.2. *Nature Genetics*. <https://doi.org/10.1038/ng.354>
- Altschul, S. F., Gish, W., Miller, W., Myers, E. W., & Lipman, D. J. (1990). Basic local alignment search tool. *Journal of Molecular Biology*. [https://doi.org/10.1016/S0022-2836\(05\)80360-2](https://doi.org/10.1016/S0022-2836(05)80360-2)

- Altschul, S. F., Madden, T. L., Schäffer, A. A., Zhang, J., Zhang, Z., Miller, W., & Lipman, D. J. (1997). Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. In *Nucleic Acids Research*. <https://doi.org/10.1093/nar/25.17.3389>
- Amundadottir, L., Kraft, P., Stolzenberg-Solomon, R. Z., Fuchs, C. S., Petersen, G. M., Arslan, A. A., Bueno-De-Mesquita, H. B., Gross, M., Helzlsouer, K., Jacobs, E. J., LaCroix, A., Zheng, W., Albanes, D., Bamlet, W., Berg, C. D., Berrino, F., Bingham, S., Buring, J. E., Bracci, P. M., ... Hoover, R. N. (2009). Genome-wide association study identifies variants in the ABO locus associated with susceptibility to pancreatic cancer. *Nature Genetics*. <https://doi.org/10.1038/ng.429>
- Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., Davis, A. P., Dolinski, K., Dwight, S. S., Eppig, J. T., Harris, M. A., Hill, D. P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J. C., Richardson, J. E., Ringwald, M., Rubin, G. M., & Sherlock, G. (2000). Gene ontology: Tool for the unification of biology. In *Nature Genetics*. <https://doi.org/10.1038/75556>
- Bairoch, A. (2000). The SWISS-PROT protein sequence database and its supplement TrEMBL in 2000. *Nucleic Acids Research*. <https://doi.org/10.1093/nar/28.1.45>
- Balmain, A. (2001). Cancer genetics: From Boveri and Mendel to microarrays. *Nature Reviews Cancer*. <https://doi.org/10.1038/35094086>
- Bao, L., & Cui, Y. (2005). Prediction of the phenotypic effects of non-synonymous single nucleotide polymorphisms using structural and evolutionary information. *Bioinformatics*. <https://doi.org/10.1093/bioinformatics/bti365>
- Barrett, J. H., Iles, M. M., Harland, M., Taylor, J. C., Aitken, J. F., Andresen, P. A., Akslen, L. A., Armstrong, B. K., Avril, M. F., Azizi, E., Bakker, B., Bergman, W., Bianchi-Scarrà, G., Bressac-De Paillerets, B., Calista, D., Cannon-Albright, L. A., Corda, E., Cust, A. E., Dębniak, T., ... Bishop, D. T. (2011). Genome-wide association study identifies three new melanoma susceptibility loci. *Nature Genetics*. <https://doi.org/10.1038/ng.959>

- Baugh, E. H., Simmons-Edler, R., Müller, C. L., Alford, R. F., Volfovsky, N., Lash, A. E., & Bonneau, R. (2016). Robust classification of protein variation using structural modelling and large-scale data integration. *Nucleic Acids Research*. <https://doi.org/10.1093/nar/gkw120>
- Belmont, J. W., Hardenbol, P., Willis, T. D., Yu, F., Yang, H., Ch'Ang, L. Y., Huang, W., Liu, B., Shen, Y., Tam, P. K. H., Tsui, L. C., Waye, M. M. Y., Wong, J. T. F., Zeng, C., Zhang, Q., Chee, M. S., Galver, L. M., Kruglyak, S., Murray, S. S., ... Tanaka, T. (2003). The international HapMap project. *Nature*. <https://doi.org/10.1038/nature02168>
- Bendl, J., Stourac, J., Salanda, O., Pavelka, A., Wieben, E. D., Zendulka, J., Brezovsky, J., & Damborsky, J. (2014). PredictSNP: Robust and Accurate Consensus Classifier for Prediction of Disease-Related Mutations. *PLoS Computational Biology*. <https://doi.org/10.1371/journal.pcbi.1003440>
- Benkert, P., Künzli, M., & Schwede, T. (2009). QMEAN server for protein model quality estimation. *Nucleic Acids Research*. <https://doi.org/10.1093/nar/gkp322>
- Berg, J., Tymoczko, J., & Stryer, L. (2002). Biochemistry, 5th edition. In *Biochemistry*.
- Berman, H. M., Battistuz, T., Bhat, T. N., Bluhm, W. F., Bourne, P. E., Burkhardt, K., Feng, Z., Gilliland, G. L., Iype, L., Jain, S., Fagan, P., Marvin, J., Padilla, D., Ravichandran, V., Schneider, B., Thanki, N., Weissig, H., Westbrook, J. D., & Zardecki, C. (2002). The protein data bank. *Acta Crystallographica Section D: Biological Crystallography*. <https://doi.org/10.1107/S0907444902003451>
- Bishop, D. T., Demenais, F., Iles, M. M., Harland, M., Taylor, J. C., Corda, E., Randerson-Moor, J., Aitken, J. F., Avril, M. F., Azizi, E., Bakker, B., Bianchi-Scarr, G., Bressac-De Paillerets, B., Calista, D., Cannon-Albright, L. A., Chin-A-Woeng, T., Dbniak, T., Galore-Haskel, G., Ghiorzo, P., ... Newton Bishop, J. A. (2009). Genome-wide association study identifies three loci associated with melanoma risk. *Nature Genetics*. <https://doi.org/10.1038/ng.411>
- Bondeson, M. L., Dahl, N., Malmgren, H., Kleijer, W. J., Tønnesen, T., Carlberg, B. M., & Pettersson, U. (1995). Inversion of the IDS gene resulting from recombination with IDS-related sequences in a common cause of the hunter syndrome. *Human Molecular Genetics*. <https://doi.org/10.1093/hmg/4.4.615>

- Botstein, D., & Risch, N. (2003). Discovering genotypes underlying human phenotypes: Past successes for mendelian disease, future approaches for complex disease. In *Nature Genetics*. <https://doi.org/10.1038/ng1090>
- Breiman, L. (2001). Random forests. *Machine Learning*. <https://doi.org/10.1023/A:1010933404324>
- Bromberg, Y., Kahn, P. C., & Rost, B. (2013). Neutral and weakly nonneutral sequence variants may define individuality. *Proceedings of the National Academy of Sciences of the United States of America*, *110*(35), 14255–14260. <https://doi.org/10.1073/pnas.1216613110>
- Bromberg, Y., & Rost, B. (2007a). SNAP: Predict effect of non-synonymous polymorphisms on function. *Nucleic Acids Research*, *35*(11), 3823–3835. <https://doi.org/10.1093/nar/gkm238>
- Bromberg, Y., & Rost, B. (2007b). SNAP: Predict effect of non-synonymous polymorphisms on function. *Nucleic Acids Research*. <https://doi.org/10.1093/nar/gkm238>
- Bromberg, Y., Yachdav, G., & Rost, B. (2008). SNAP predicts effect of mutations on protein function. *Bioinformatics*, *24*(20), 2397–2398. <https://doi.org/10.1093/bioinformatics/btn435>
- Burton, P. R., Clayton, D. G., Cardon, L. R., Craddock, N., Deloukas, P., Duncanson, A., Kwiatkowski, D., & Green, E. K. (2007). The Wellcome Trust Case Control Consortium. Genome-wide association study of 14,000 cases of seven common diseases and 3,000 shared controls. *Nature*. <https://doi.org/10.1038/nature05911>
- Bush, W. S., & Moore, J. H. (2012). Chapter 11: Genome-Wide Association Studies. *PLoS Computational Biology*. <https://doi.org/10.1371/journal.pcbi.1002822>
- Calabrese, R., Capriotti, E., Fariselli, P., Martelli, P. L., & Casadio, R. (2009a). Functional annotations improve the predictive score of human disease-related mutations in proteins. *Human Mutation*, *30*(8), 1237–1244. <https://doi.org/10.1002/humu.21047>

- Calabrese, R., Capriotti, E., Fariselli, P., Martelli, P. L., & Casadio, R. (2009b). Functional annotations improve the predictive score of human disease-related mutations in proteins. *Human Mutation*. <https://doi.org/10.1002/humu.21047>
- Capriotti, E., Altman, R. B., & Bromberg, Y. (2013). Collective judgment predicts disease-associated single nucleotide variants. *BMC Genomics*. <https://doi.org/10.1186/1471-2164-14-s3-s2>
- Capriotti, E., Fariselli, P., & Casadio, R. (2004). A neural-network-based method for predicting protein stability changes upon single point mutations. *Bioinformatics*. <https://doi.org/10.1093/bioinformatics/bth928>
- Capriotti, E., Fariselli, P., & Casadio, R. (2005). I-Mutant2.0: Predicting stability changes upon mutation from the protein sequence or structure. *Nucleic Acids Research*. <https://doi.org/10.1093/nar/gki375>
- Carter, H., Chen, S., Isik, L., Tyekucheva, S., Velculescu, V. E., Kinzler, K. W., Vogelstein, B., & Karchin, R. (2009). Cancer-specific high-throughput annotation of somatic mutations: Computational prediction of driver missense mutations. *Cancer Research*. <https://doi.org/10.1158/0008-5472.CAN-09-1133>
- Cerami, E., Gao, J., Dogrusoz, U., Gross, B. E., Sumer, S. O., Aksoy, B. A., Jacobsen, A., Byrne, C. J., Heuer, M. L., Larsson, E., Antipin, Y., Reva, B., Goldberg, A. P., Sander, C., & Schultz, N. (2012). The cBio Cancer Genomics Portal: An open platform for exploring multidimensional cancer genomics data. *Cancer Discovery*. <https://doi.org/10.1158/2159-8290.CD-12-0095>
- Chamary, J. V., Parmley, J. L., & Hurst, L. D. (2006). Hearing silence: Non-neutral evolution at synonymous sites in mammals. In *Nature Reviews Genetics*. <https://doi.org/10.1038/nrg1770>
- Chasman, D., & Adams, R. M. (2001). Predicting the functional consequences of non-synonymous single nucleotide polymorphisms: Structure-based assessment of amino acid variation. *Journal of Molecular Biology*. <https://doi.org/10.1006/jmbi.2001.4510>

- Chen, R., Davydov, E. V., Sirsota, M., & Butte, A. J. (2010). Non-synonymous and synonymous coding SNPS show similar likelihood and effect size of human disease association. *PLoS ONE*. <https://doi.org/10.1371/journal.pone.0013574>
- Choi, Y., Sims, G. E., Murphy, S., Miller, J. R., & Chan, A. P. (2012). Predicting the Functional Effect of Amino Acid Substitutions and Indels. *PLoS ONE*, 7(10). <https://doi.org/10.1371/journal.pone.0046688>
- Clifford, R. J., Edmonson, M. N., Nguyen, C., & Buetow, K. H. (2004). Large-scale analysis of non-synonymous coding region single nucleotide polymorphisms. *Bioinformatics*. <https://doi.org/10.1093/bioinformatics/bth029>
- Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*. <https://doi.org/10.1007/bf00994018>
- Cristianini, N., & Shawe-Taylor, J. (2000). An Introduction to Support Vector Machines and Other Kernel-based Learning Methods. In *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. <https://doi.org/10.1017/cbo9780511801389>
- Cuevas William A, Lee Sang-Kyu, Ramer Sandra W, Shaw Andrew, Topozada Amr R, Estell David E, H. S. H. (2009). *Geobacillus Stearotherophilus Alpha-Amylase (AmyS) Variants with Improved Properties*.
- Cutler, A., Cutler, D. R., & Stevens, J. R. (2012). Random forests. In *Ensemble Machine Learning: Methods and Applications*. https://doi.org/10.1007/9781441993267_5
- Dasgupta, A., Sun, Y. V., König, I. R., Bailey-Wilson, J. E., & Malley, J. D. (2011). Brief review of regression-based and machine learning methods in genetic epidemiology: The Genetic Analysis Workshop 17 experience. *Genetic Epidemiology*. <https://doi.org/10.1002/gepi.20642>
- Datta, A., Mazumder, M. H. H., Chowdhury, A. S., & Hasan, M. A. (2015). Functional and Structural Consequences of Damaging Single Nucleotide Polymorphisms in Human Prostate Cancer Predisposition Gene RNASEL. In *BioMed Research International*. <https://doi.org/10.1155/2015/271458>

- De Baets, G., Van Durme, J., Reumers, J., Maurer-Stroh, S., Vanhee, P., Dopazo, J., Schymkowitz, J., & Rousseau, F. (2012). SNPeff 4.0: On-line prediction of molecular and structural effects of protein-coding variants. *Nucleic Acids Research*. <https://doi.org/10.1093/nar/gkr996>
- Dincer, C. (2019). *3D Spatial organization and network-guided comparison of Mmutation profiles in glioblastoma*. Middle East Technical University.
- Dincer, C., Kaya, T., Keskin, O., Gursoy, A., & Tuncbag, N. (2019). 3D spatial organization and network-guided comparison of mutation profiles in Glioblastoma reveals similarities across patients. *PLoS Computational Biology*. <https://doi.org/10.1371/journal.pcbi.1006789>
- Douville, C., Carter, H., Kim, R., Niknafs, N., Diekhans, M., Stenson, P. D., Cooper, D. N., Ryan, M., & Karchin, R. (2013). CRAVAT: Cancer-related analysis of variants toolkit. *Bioinformatics*. <https://doi.org/10.1093/bioinformatics/btt017>
- Easton, D. F., & Eeles, R. A. (2008). Genome-wide association studies in cancer. *Human Molecular Genetics*. <https://doi.org/10.1093/hmg/ddn287>
- Edwards, N. C., Hing, Z. A., Perry, A., Blaisdell, A., Kopelman, D. B., Fathke, R., Plum, W., Newell, J., Allen, C. E., Geetha, S., Shapiro, A., Okunji, C., Kosti, I., Shomron, N., Grigoryan, V., Przytycka, T. M., Sauna, Z. E., Salari, R., Mandel-Gutfreund, Y., ... Kimchi-Sarfaty, C. (2012). Characterization of coding synonymous and non-synonymous variants in ADAMTS13 using ex vivo and in silico approaches. *PLoS ONE*. <https://doi.org/10.1371/journal.pone.0038864>
- Eswar, N., John, B., Mirkovic, N., Fiser, A., Ilyin, V. A., Pieper, U., Stuart, A. C., Marti-Renom, M. A., Madhusudhan, M. S., Yerkovich, B., & Sali, A. (2003). Tools for comparative protein structure modeling and analysis. *Nucleic Acids Research*. <https://doi.org/10.1093/nar/gkg543>
- Fan, J., Upadhye, S., & Worster, A. (2006). Understanding receiver operating characteristic (ROC) curves. *Canadian Journal of Emergency Medicine*. <https://doi.org/10.1017/S1481803500013336>

Farh, K. K. H., Marson, A., Zhu, J., Kleinewietfeld, M., Housley, W. J., Beik, S., Shores, N., Whitton, H., Ryan, R. J. H., Shishkin, A. A., Hatan, M., Carrasco-Alfonso, M. J., Mayer, D., Luckey, C. J., Patsopoulos, N. A., De Jager, P. L., Kuchroo, V. K., Epstein, C. B., Daly, M. J., ... Bernstein, B. E. (2015). Genetic and epigenetic fine mapping of causal autoimmune disease variants. *Nature*. <https://doi.org/10.1038/nature13835>

Ferrer-Costa, C., Orozco, M., & De La Cruz, X. (2004). Sequence-based prediction of pathological mutations. *Proteins: Structure, Function and Genetics*. <https://doi.org/10.1002/prot.20252>

Ferrer-Costa, Carles, Gelpí, J. L., Zamakola, L., Parraga, I., de la Cruz, X., & Orozco, M. (2005). PMUT: A web-based tool for the annotation of pathological mutations on proteins. *Bioinformatics*. <https://doi.org/10.1093/bioinformatics/bti486>

Ferrer-Costa, Carles, Orozco, M., & De La Cruz, X. (2002). Characterization of disease-associated single amino acid polymorphisms in terms of sequence and structure properties. *Journal of Molecular Biology*. <https://doi.org/10.1006/jmbi.2001.5255>

Finn, R. D., Attwood, T. K., Babbitt, P. C., Bateman, A., Bork, P., Bridge, A. J., Chang, H. Y., Dosztanyi, Z., El-Gebali, S., Fraser, M., Gough, J., Haft, D., Holliday, G. L., Huang, H., Huang, X., Letunic, I., Lopez, R., Lu, S., Marchler-Bauer, A., ... Mitchell, A. L. (2017). InterPro in 2017-beyond protein family and domain annotations. *Nucleic Acids Research*. <https://doi.org/10.1093/nar/gkw1107>

Finn, R. D., Bateman, A., Clements, J., Coghill, P., Eberhardt, R. Y., Eddy, S. R., Heger, A., Hetherington, K., Holm, L., Mistry, J., Sonnhammer, E. L. L., Tate, J., & Punta, M. (2014). Pfam: The protein families database. In *Nucleic Acids Research*. <https://doi.org/10.1093/nar/gkt1223>

Fisher, R. (1956). The mathematics of a lady tasting tea. *The World of Mathematics*.

Forbes, S. A., Beare, D., Boutselakis, H., Bamford, S., Bindal, N., Tate, J., Cole, C. G., Ward, S., Dawson, E., Ponting, L., Stefancsik, R., Harsha, B., YinKok, C., Jia, M., Jubb, H., Sondka, Z., Thompson, S., De, T., & Campbell, P. J. (2017). COSMIC: Somatic cancer genetics at high-resolution. *Nucleic Acids Research*. <https://doi.org/10.1093/nar/gkw1121>

- Frazer, K. A., Murray, S. S., Schork, N. J., & Topol, E. J. (2009). Human genetic variation and its contribution to complex traits. In *Nature Reviews Genetics*. <https://doi.org/10.1038/nrg2554>
- Geer, L. Y., Marchler-Bauer, A., Geer, R. C., Han, L., He, J., He, S., Liu, C., Shi, W., & Bryant, S. H. (2009). The NCBI BioSystems database. *Nucleic Acids Research*. <https://doi.org/10.1093/nar/gkp858>
- Glusman, G., Rose, P. W., Prlić, A., Dougherty, J., Duarte, J. M., Hoffman, A. S., Barton, G. J., Bendixen, E., Bergquist, T., Bock, C., Brunk, E., Buljan, M., Burley, S. K., Cai, B., Carter, H., Gao, J. J., Godzik, A., Heuer, M., Hicks, M., ... Deutsch, E. W. (2017). Mapping genetic variations to three-dimensional protein structures to enhance variant interpretation: A proposed framework. *Genome Medicine*. <https://doi.org/10.1186/s13073-017-0509-y>
- Goddard, M. E., Wray, N. R., Verbyla, K., & Visscher, P. M. (2009). Estimating effects and making predictions from genome-wide marker data. *Statistical Science*. <https://doi.org/10.1214/09-STS306>
- Goldsack, D. E., & Chalifoux, R. C. (1973). Contribution of the free energy of mixing of hydrophobic side chains to the stability of the tertiary structure of proteins. *Journal of Theoretical Biology*. [https://doi.org/10.1016/0022-5193\(73\)90075-1](https://doi.org/10.1016/0022-5193(73)90075-1)
- Gong, H., Zhang, H., Zhu, J., Wang, C., Sun, S., Zheng, W. M., & Bu, D. (2017). Improving prediction of burial state of residues by exploiting correlation among residues. *BMC Bioinformatics*. <https://doi.org/10.1186/s12859-017-1475-5>
- González-Pérez, A., & López-Bigas, N. (2011). Improving the assessment of the outcome of nonsynonymous SNVs with a consensus deleteriousness score, Condel. *American Journal of Human Genetics*. <https://doi.org/10.1016/j.ajhg.2011.03.004>
- Gordon, A. D., Breiman, L., Friedman, J. H., Olshen, R. A., & Stone, C. J. (1984). Classification and Regression Trees. *Biometrics*. <https://doi.org/10.2307/2530946>

Gorski, M., Tin, A., Garnaas, M., McMahon, G. M., Chu, A. Y., Tayo, B. O., Pattaro, C., Teumer, A., Chasman, D. I., Chalmers, J., Hamet, P., Tremblay, J., Woodward, M., Aspelund, T., Eiriksdottir, G., Gudnason, V., Harris, T. B., Launer, L. J., Smith, A. V., ... Böger, C. A. (2015). Genome-wide association study of kidney function decline in individuals of European descent. *Kidney International*. <https://doi.org/10.1038/ki.2014.361>

Grantham, R. (1974). Amino acid difference formula to help explain protein evolution. *Science*. <https://doi.org/10.1126/science.185.4154.862>

Gribskov, M., McLachlan, A. D., & Eisenberg, D. (1987). Profile analysis: detection of distantly related proteins. *Proceedings of the National Academy of Sciences of the United States of America*. <https://doi.org/10.1073/pnas.84.13.4355>

Gunther, E. C., Stone, D. J., Gerwien, R. W., Bento, P., & Heyes, M. P. (2003). Prediction of clinical drug efficacy by classification of drug-induced genomic expression profiles in vitro. *Proceedings of the National Academy of Sciences of the United States of America*. <https://doi.org/10.1073/pnas.1632587100>

Halushka, M. K., Fan, J. B., Bentley, K., Hsie, L., Shen, N., Weder, A., Cooper, R., Lipshutz, R., & Chakravarti, A. (1999). Patterns of single-nucleotide polymorphisms in candidate genes for blood-pressure homeostasis. *Nature Genetics*. <https://doi.org/10.1038/10297>

Hamosh, A., Scott, A. F., Amberger, J. S., Bocchini, C. A., & McKusick, V. A. (2005). Online Mendelian Inheritance in Man (OMIM), a knowledgebase of human genes and genetic disorders. *Nucleic Acids Research*. <https://doi.org/10.1093/nar/gki033>

Hastie, T., Tibshirani, R., & Friedman, J. (2009). Springer Series in Statistics. In *The Elements of Statistical Learning*. <https://doi.org/10.1007/b94608>

Hecht, M., Bromberg, Y., & Rost, B. (2015). Better prediction of functional effects for sequence variants. *BMC Genomics*, *16*(8), S1. <https://doi.org/10.1186/1471-2164-16-S8-S1>

- Henikoff, S., & Henikoff, J. G. (1992). Amino acid substitution matrices from protein blocks. *Proceedings of the National Academy of Sciences of the United States of America*. <https://doi.org/10.1073/pnas.89.22.10915>
- Hepp, D., Gonçalves, G. L., & Ochotorena De Freitas, T. R. (2015). Prediction of the damage-associated non-synonymous single nucleotide polymorphisms in the human MC1R gene e0121812. *PLoS ONE*. <https://doi.org/10.1371/journal.pone.0121812>
- Hindorff, L. A., Sethupathy, P., Junkins, H. A., Ramos, E. M., Mehta, J. P., Collins, F. S., & Manolio, T. A. (2009). 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*. <https://doi.org/10.1073/pnas.0903103106>
- Hirschhorn, J. N., & Daly, M. J. (2005). Genome-wide association studies for common diseases and complex traits. In *Nature Reviews Genetics*. <https://doi.org/10.1038/nrg1521>
- Ho, D. S. W., Schierding, W., Wake, M., Saffery, R., & O'Sullivan, J. (2019). Machine learning SNP based prediction for precision medicine. In *Frontiers in Genetics*. <https://doi.org/10.3389/fgene.2019.00267>
- HSSINA, B., MERBOUHA, A., EZZIKOURI, H., & ERRITALI, M. (2014). A comparative study of decision tree ID3 and C4.5. *International Journal of Advanced Computer Science and Applications*. <https://doi.org/10.14569/specialissue.2014.040203>
- Huang, T., Wang, P., Ye, Z., Xu, H., He, Z., Feng, K., Hu, L., Cui, W., Wang, K., Dong, X., Xie, L., Kong, X., Cai, Y. D., & Li, Y. (2010). Prediction of deleterious non-synonymous SNPs based on protein interaction network and hybrid properties. *PLoS ONE*. <https://doi.org/10.1371/journal.pone.0011900>
- Hunt, R., Sauna, Z. E., Ambudkar, S. V., Gottesman, M. M., & Kimchi-Sarfaty, C. (2009). Silent (synonymous) SNPs: should we care about them? In *Methods in molecular biology (Clifton, N.J.)*. https://doi.org/10.1007/978-1-60327-411-1_2

- Illergård, K., Ardell, D. H., & Elofsson, A. (2009). Structure is three to ten times more conserved than sequence - A study of structural response in protein cores. *Proteins: Structure, Function and Bioinformatics*. <https://doi.org/10.1002/prot.22458>
- James, G., Witten, D., Hastie, T., & Tibshirani, R. (2000). An introduction to Statistical Learning. In *Current medicinal chemistry*. <https://doi.org/10.1007/978-1-4614-7138-7>
- Jayashree, S., Murugavel, P., Sowdhamini, R., & Srinivasan, N. (2019). Interface residues of transient protein-protein complexes have extensive intra-protein interactions apart from inter-protein interactions. *Biology Direct*. <https://doi.org/10.1186/s13062-019-0232-2>
- Johansen, M. B., Izarzugaza, J. M. G., Brunak, S., Petersen, T. N., & Gupta, R. (2013). Prediction of Disease Causing Non-Synonymous SNPs by the Artificial Neural Network Predictor NetDiseaseSNP. *PLoS ONE*. <https://doi.org/10.1371/journal.pone.0068370>
- Kabsch, W., & Sander, C. (1983). Dictionary of protein secondary structure: Pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers*. <https://doi.org/10.1002/bip.360221211>
- Kaminker, J. S., Zhang, Y., Watanabe, C., & Zhang, Z. (2007). CanPredict: A computational tool for predicting cancer-associated missense mutations. *Nucleic Acids Research*, 35(SUPPL.2), 595–598. <https://doi.org/10.1093/nar/gkm405>
- Karchin, R., Diekhans, M., Kelly, L., Thomas, D. J., Pieper, U., Eswar, N., Haussler, D., & Sali, A. (2005). LS-SNP: Large-scale annotation of coding non-synonymous SNPs based on multiple information sources. *Bioinformatics*. <https://doi.org/10.1093/bioinformatics/bti442>
- Kawabata, T., Ota, M., & Nishikawa, K. (1999). The protein mutant database. In *Nucleic Acids Research*. <https://doi.org/10.1093/nar/27.1.355>

- Khurana, E., Fu, Y., Chakravarty, D., Demichelis, F., Rubin, M. A., & Gerstein, M. (2016). Role of non-coding sequence variants in cancer. In *Nature Reviews Genetics*. <https://doi.org/10.1038/nrg.2015.17>
- Koolen, D. A., Vissers, L. E. L. M., Pfuntd, R., De Leeuw, N., Knight, S. J. L., Regan, R., Kooy, R. F., Reyniers, E., Romano, C., Fichera, M., Schinzel, A., Baumer, A., Anderlid, B. M., Schoumans, J., Knoers, N. V., Van Kessel, A. G., Sistermans, E. A., Veltman, J. A., Brunner, H. G., & De Vries, B. B. A. (2006). A new chromosome 17q21.31 microdeletion syndrome associated with a common inversion polymorphism. *Nature Genetics*. <https://doi.org/10.1038/ng1853>
- Kotsiantis, S. B. (2013). Decision trees: A recent overview. In *Artificial Intelligence Review*. <https://doi.org/10.1007/s10462-011-9272-4>
- Krishnan, V. G., & Westhead, D. R. (2003). A comparative study of machine-learning methods to predict the effects of single nucleotide polymorphisms on protein function. *Bioinformatics*. <https://doi.org/10.1093/bioinformatics/btg297>
- Kucukkal, T. G., Yang, Y., Chapman, S. C., Cao, W., & Alexov, E. (2014). Computational and experimental approaches to reveal the effects of single nucleotide polymorphisms with respect to disease diagnostics. In *International Journal of Molecular Sciences*. <https://doi.org/10.3390/ijms15069670>
- Kuhn, M., & Johnson, K. (2013). Applied predictive modeling. In *Applied Predictive Modeling*. <https://doi.org/10.1007/978-1-4614-6849-3>
- Kulkarni, V., Errami, M., Barber, R., & Garner, H. R. (2008). Exhaustive prediction of disease susceptibility to coding base changes in the human genome. *BMC Bioinformatics*. <https://doi.org/10.1186/1471-2105-9-S9-S3>
- Kumar, P., Henikoff, S., & Ng, P. C. (2009a). Predicting the effects of coding non-synonymous variants on protein function using the SIFT algorithm. *Nature Protocols*. <https://doi.org/10.1038/nprot.2009.86>

- Kumar, P., Henikoff, S., & Ng, P. C. (2009b). Predicting the effects of coding non-synonymous variants on protein function using the SIFT algorithm. *Nature Protocols*, 4(7), 1073–1082. <https://doi.org/10.1038/nprot.2009.86>
- Kyte, J., & Doolittle, R. F. (1982). A simple method for displaying the hydropathic character of a protein. *Journal of Molecular Biology*. [https://doi.org/10.1016/0022-2836\(82\)90515-0](https://doi.org/10.1016/0022-2836(82)90515-0)
- Lakich, D., Kazazian, H. H., Antonarakis, S. E., & Gitschier, J. (1993). Inversions disrupting the factor VIII gene are a common cause of severe haemophilia A. *Nature Genetics*. <https://doi.org/10.1038/ng1193-236>
- Landrum, M. J., & Kattman, B. L. (2018). ClinVar at five years: Delivering on the promise. *Human Mutation*. <https://doi.org/10.1002/humu.23641>
- Landrum, M. J., Lee, J. M., Benson, M., Brown, G. R., Chao, C., Chitipiralla, S., Gu, B., Hart, J., Hoffman, D., Jang, W., Karapetyan, K., Katz, K., Liu, C., Maddipatla, Z., Malheiro, A., McDaniel, K., Ovetsky, M., Riley, G., Zhou, G., ... Maglott, D. R. (2018). ClinVar: Improving access to variant interpretations and supporting evidence. *Nucleic Acids Research*. <https://doi.org/10.1093/nar/gkx1153>
- Lau, A. Y., & Chasman, D. I. (2004). Functional classification of proteins and protein variants. *Proceedings of the National Academy of Sciences of the United States of America*. <https://doi.org/10.1073/pnas.0305043101>
- Leaver-Fay, A., Tyka, M., Lewis, S. M., Lange, O. F., Thompson, J., Jacak, R., Kaufman, K., Renfrew, P. D., Smith, C. A., Sheffler, W., Davis, I. W., Cooper, S., Treuille, A., Mandell, D. J., Richter, F., Ban, Y. E. A., Fleishman, S. J., Corn, J. E., Kim, D. E., ... Bradley, P. (2011). Rosetta3: An object-oriented software suite for the simulation and design of macromolecules. In *Methods in Enzymology*. <https://doi.org/10.1016/B978-0-12-381270-4.00019-6>
- Lee, C., & Scherer, S. W. (2010). The clinical context of copy number variation in the human genome. In *Expert Reviews in Molecular Medicine*. <https://doi.org/10.1017/S1462399410001390>

- Leinonen, R., Garcia Diez, F., Binns, D., Fleischmann, W., Lopez, R., & Apweiler, R. (2004). UniProt archive. *Bioinformatics*.
<https://doi.org/10.1093/bioinformatics/bth191>
- Lette, G., & Rioux, J. D. (2008). Autoimmune diseases: Insights from genome-wide association studies. *Human Molecular Genetics*.
<https://doi.org/10.1093/hmg/ddn246>
- Li, B., Krishnan, V. G., Mort, M. E., Xin, F., Kamati, K. K., Cooper, D. N., Mooney, S. D., & Radivojac, P. (2009). Automated inference of molecular mechanisms of disease from amino acid substitutions. *Bioinformatics*.
<https://doi.org/10.1093/bioinformatics/btp528>
- Liaw, A., & Wiener, M. (2002). Classification and Regression by randomForest. *R News*.
- Liu, C. T., Garnaas, M. K., Tin, A., Kottgen, A., Franceschini, N., Peralta, C. A., de Boer, I. H., Lu, X., Atkinson, E., Ding, J., Nalls, M., Shriner, D., Coresh, J., Kutlar, A., Bibbins-Domingo, K., Siscovick, D., Akyzbekova, E., Wyatt, S., Astor, B., ... Fox, C. S. (2011). Genetic association for renal traits among participants of African Ancestry reveals new loci for renal function. *PLoS Genetics*.
<https://doi.org/10.1371/journal.pgen.1002264>
- Malleshappa Gowder, S., Chatterjee, J., Chaudhuri, T., & Paul, K. (2014). Prediction and analysis of surface hydrophobic residues in tertiary structure of proteins. *The Scientific World Journal*. <https://doi.org/10.1155/2014/971258>
- Manolio, T. A., Brooks, L. D., & Collins, F. S. (2008). A HapMap harvest of insights into the genetics of common disease. In *Journal of Clinical Investigation*.
<https://doi.org/10.1172/JCI34772>
- Mathe, E., Olivier, M., Kato, S., Ishioka, C., Hainaut, P., & Tavtigian, S. V. (2006). Computational approaches for predicting the biological effect of p53 missense mutations: A comparison of three sequence analysis based methods. *Nucleic Acids Research*. <https://doi.org/10.1093/nar/gkj518>

- McCarthy, M. I., Abecasis, G. R., Cardon, L. R., Goldstein, D. B., Little, J., Ioannidis, J. P. A., & Hirschhorn, J. N. (2008). Genome-wide association studies for complex traits: Consensus, uncertainty and challenges. In *Nature Reviews Genetics*. <https://doi.org/10.1038/nrg2344>
- Meyer, M. J., Beltrán, J. F., Liang, S., Fragoza, R., Rumack, A., Liang, J., Wei, X., & Yu, H. (2018). Interactome INSIDER: a structural interactome browser for genomic studies. *Nature Methods*. <https://doi.org/10.1038/nmeth.4540>
- Mitchell, T. M. (1999). Machine Learning and Data Mining. *Communications of the ACM*. <https://doi.org/10.1145/319382.319388>
- Mitternacht, S. (2016). FreeSASA: An open source C library for solvent accessible surface area calculations. *F1000Research*. <https://doi.org/10.12688/f1000research.7931.1>
- Mohlke, K. L., Boehnke, M., & Abecasis, G. R. (2008). Metabolic and cardiovascular traits: An abundance of recently identified common genetic variants. *Human Molecular Genetics*. <https://doi.org/10.1093/hmg/ddn275>
- Momen-Roknabadi, A., Sadeghi, M., Pezeshk, H., & Marashi, S. A. (2008). Impact of residue accessible surface area on the prediction of protein secondary structures. *BMC Bioinformatics*. <https://doi.org/10.1186/1471-2105-9-357>
- Mooney, S. D., & Altman, R. B. (2003). MutDB: Annotating human variation with functionally relevant data. *Bioinformatics*. <https://doi.org/10.1093/bioinformatics/btg241>
- Mooney, S. D., & Klein, T. E. (2002). The functional importance of disease-associated mutation. *BMC Bioinformatics*. <https://doi.org/10.1186/1471-2105-3-24>
- Mudunuri, U., Che, A., Yi, M., & Stephens, R. M. (2009). bioDBnet: The biological database network. *Bioinformatics*. <https://doi.org/10.1093/bioinformatics/btn654>
- Müller, A. C., & Guido, S. (2015). Introduction to Machine Learning with Python and Scikit-Learn. In *O'Reilly Media, Inc.*

- Muñoz, V., & Serrano, L. (1994). Intrinsic secondary structure propensities of the amino acids, using statistical ϕ - ψ matrices: Comparison with experimental scales. *Proteins: Structure, Function, and Bioinformatics*. <https://doi.org/10.1002/prot.340200403>
- Ng, P. C., & Henikoff, S. (2001). Predicting deleterious amino acid substitutions. *Genome Research*. <https://doi.org/10.1101/gr.176601>
- Ng, P. C., & Henikoff, S. (2003). SIFT: Predicting amino acid changes that affect protein function. *Nucleic Acids Research*. <https://doi.org/10.1093/nar/gkg509>
- Ng, P. C., & Henikoff, S. (2006). Predicting the Effects of Amino Acid Substitutions on Protein Function. *Annual Review of Genomics and Human Genetics*, 7(1), 61–80. <https://doi.org/10.1146/annurev.genom.7.080505.115630>
- Niknafs, N., Kim, D., Kim, R., Diekhans, M., Ryan, M., Stenson, P. D., Cooper, D. N., & Karchin, R. (2013). MuPIT interactive: Webserver for mapping variant positions to annotated, interactive 3D structures. *Human Genetics*. <https://doi.org/10.1007/s00439-013-1325-0>
- O'Connor, C. (2008). Human Chromosome Translocations and Cancer. *Nature Education*. <https://doi.org/10.1038/35094086>
- Okser, S., Pahikkala, T., Airola, A., Salakoski, T., Ripatti, S., & Aittokallio, T. (2014). Regularized Machine Learning in the Genetic Prediction of Complex Traits. *PLoS Genetics*. <https://doi.org/10.1371/journal.pgen.1004754>
- Olatubosun, A., Väliäho, J., Härkönen, J., Thusberg, J., & Vihinen, M. (2012). PON-P: Integrated predictor for pathogenicity of missense variants. *Human Mutation*. <https://doi.org/10.1002/humu.22102>
- Pagani, F., Raponi, M., & Baralle, F. E. (2005). Synonymous mutations in CFTR exon 12 affect splicing and are not neutral in evolution. *Proceedings of the National Academy of Sciences of the United States of America*. <https://doi.org/10.1073/pnas.0502288102>

Pedregosa, F., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Dubourg, V., Pedregosa, F., Gramfort, A., Michel, V., Thirion, B., Pedregosa, F., & Weiss, R. (2011). Scikit-learn : Machine Learning in Python To cite this version : Scikit-learn : Machine Learning in Python. *Journal of Machine Learning Research*. <https://doi.org/https://dl.acm.org/citation.cfm?id=2078195>

Pieper, U., Webb, B. M., Dong, G. Q., Schneidman-Duhovny, D., Fan, H., Kim, S. J., Khuri, N., Spill, Y. G., Weinkam, P., Hammel, M., Tainer, J. A., Nilges, M., & Sali, A. (2014). ModBase, a database of annotated comparative protein structure models and associated resources. *Nucleic Acids Research*. <https://doi.org/10.1093/nar/gkt1144>

Presnyak, V., Alhusaini, N., Chen, Y. H., Martin, S., Morris, N., Kline, N., Olson, S., Weinberg, D., Baker, K. E., Graveley, B. R., & Collier, J. (2015). Codon optimality is a major determinant of mRNA stability. *Cell*. <https://doi.org/10.1016/j.cell.2015.02.029>

Quan, L., Lv, Q., & Zhang, Y. (2016). STRUM: Structure-based prediction of protein stability changes upon single-point mutation. *Bioinformatics*. <https://doi.org/10.1093/bioinformatics/btw361>

Quinlan, J. R. (1986). Induction of Decision Trees. *Machine Learning*. <https://doi.org/10.1023/A:1022643204877>

Quinlan, J. R. (1996). Bagging, boosting, and C4.5. *Proceedings of the National Conference on Artificial Intelligence*.

Ramachandran, G. N., Ramakrishnan, C., & Sasisekharan, V. (1963). Stereochemistry of polypeptide chain configurations. In *Journal of Molecular Biology*. [https://doi.org/10.1016/S0022-2836\(63\)80023-6](https://doi.org/10.1016/S0022-2836(63)80023-6)

Ramensky, V. (2002). Human non-synonymous SNPs: server and survey. *Nucleic Acids Research*, 30(17), 3894–3900. <https://doi.org/10.1093/nar/gkf493>

Reich, D. E., & Lander, E. S. (2001). On the allelic spectrum of human disease. In *Trends in Genetics*. [https://doi.org/10.1016/S0168-9525\(01\)02410-6](https://doi.org/10.1016/S0168-9525(01)02410-6)

- Reva, B., Antipin, Y., & Sander, C. (2011). Predicting the functional impact of protein mutations: Application to cancer genomics. *Nucleic Acids Research*. <https://doi.org/10.1093/nar/gkr407>
- Rost, B. (1996). [31] PHD: Predicting one-dimensional protein structure by profile-based neural networks. *Methods in Enzymology*. [https://doi.org/10.1016/s0076-6879\(96\)66033-9](https://doi.org/10.1016/s0076-6879(96)66033-9)
- Rost, B., & Sander, C. (1993). Prediction of protein secondary structure at better than 70% accuracy. *Journal of Molecular Biology*. <https://doi.org/10.1006/jmbi.1993.1413>
- Rost, B., & Sander, C. (1994). Conservation and prediction of solvent accessibility in protein families. *Proteins: Structure, Function, and Bioinformatics*. <https://doi.org/10.1002/prot.340200303>
- Ryan, M., Diekhans, M., Lien, S., Liu, Y., & Karchin, R. (2009). LS-SNP/PDB: Annotated non-synonymous SNPs mapped to Protein Data Bank structures. *Bioinformatics*. <https://doi.org/10.1093/bioinformatics/btp242>
- Sachidanandam, R., Weissman, D., Schmidt, S. C., Kakol, J. M., Stein, L. D., Marth, G., Sherry, S., Mullikin, J. C., Mortimore, B. J., Willey, D. L., Hunt, S. E., Cole, C. G., Coggill, P. C., Rice, C. M., Ning, Z., Rogers, J., Bentley, D. R., Kwok, P. Y., Mardis, E. R., ... Altshuler, D. (2001). A map of human genome sequence variation containing 1.42 million single nucleotide polymorphisms. *Nature*. <https://doi.org/10.1038/35057149>
- Samani, N. J., Erdmann, J., Hall, A. S., Hengstenberg, C., Mangino, M., Mayer, B., Dixon, R. J., Meitinger, T., Braund, P., Wichmann, H. E., Barrett, J. H., König, I. R., Stevens, S. E., Szymczak, S., Tregouet, D. A., Iles, M. M., Pahlke, F., Pollard, H., Lieb, W., ... Schunkert, H. (2007). Genomewide association analysis of coronary artery disease. *New England Journal of Medicine*. <https://doi.org/10.1056/NEJMoa072366>
- Sarkar, A., Yang, Y., & Vihinen, M. (2019). Variation Benchmark Datasets: Update, Criteria, Quality and Applications. *BioRxiv*. <https://doi.org/10.1101/634766>

- Sauna, Z. E., & Kimchi-Sarfaty, C. (2011). Understanding the contribution of synonymous mutations to human disease. In *Nature Reviews Genetics*. <https://doi.org/10.1038/nrg3051>
- Saunders, C. T., & Baker, D. (2002). Evaluation of structural and evolutionary contributions to deleterious mutation prediction. *Journal of Molecular Biology*. [https://doi.org/10.1016/S0022-2836\(02\)00813-6](https://doi.org/10.1016/S0022-2836(02)00813-6)
- Schlessinger, A., Yachdav, G., & Rost, B. (2006). PROFbval: Predict flexible and rigid residues in proteins. *Bioinformatics*. <https://doi.org/10.1093/bioinformatics/btl032>
- Schmidhuber, J. (2015). Deep Learning in neural networks: An overview. In *Neural Networks*. <https://doi.org/10.1016/j.neunet.2014.09.003>
- Schwarz, J. M., Cooper, D. N., Schuelke, M., & Seelow, D. (2014). Mutationtaster2: Mutation prediction for the deep-sequencing age. *Nature Methods*, *11*(4), 361–362. <https://doi.org/10.1038/nmeth.2890>
- Schwarz, J. M., Rödelsperger, C., Schuelke, M., & Seelow, D. (2010). MutationTaster evaluates disease-causing potential of sequence alterations. *Nature Methods*, *7*(8), 575–576. <https://doi.org/10.1038/nmeth0810-575>
- Schwede, T. (2013). Protein modeling: What happened to the “protein structure gap”? In *Structure*. <https://doi.org/10.1016/j.str.2013.08.007>
- Schymkowitz, J., Borg, J., Stricher, F., Nys, R., Rousseau, F., & Serrano, L. (2005). The FoldX web server: An online force field. *Nucleic Acids Research*. <https://doi.org/10.1093/nar/gki387>
- Shalev-Shwartz, S., & Ben-David, S. (2013). Understanding machine learning: From theory to algorithms. In *Understanding Machine Learning: From Theory to Algorithms*. <https://doi.org/10.1017/CBO9781107298019>
- Shannon, C. E. (1948). A Mathematical Theory of Communication. *Bell System Technical Journal*. <https://doi.org/10.1002/j.1538-7305.1948.tb01338.x>

- Shaw-Smith, C., Pittman, A. M., Willatt, L., Martin, H., Rickman, L., Gribble, S., Curley, R., Cumming, S., Dunn, C., Kalaitzopoulos, D., Porter, K., Prigmore, E., Krepischi-Santos, A. C. V., Varela, M. C., Koiffmann, C. P., Lees, A. J., Rosenberg, C., Firth, H. V., De Silva, R., & Carter, N. P. (2006). Microdeletion encompassing MAPT at chromosome 17q21.3 is associated with developmental delay and learning disability. *Nature Genetics*. <https://doi.org/10.1038/ng1858>
- Sherry, S. T. (2001). dbSNP: the NCBI database of genetic variation. *Nucleic Acids Research*. <https://doi.org/10.1093/nar/29.1.308>
- Singh, G., & Samavedham, L. (2015). Unsupervised learning based feature extraction for differential diagnosis of neurodegenerative diseases: A case study on early-stage diagnosis of Parkinson disease. *Journal of Neuroscience Methods*. <https://doi.org/10.1016/j.jneumeth.2015.08.011>
- Sjolander, K., Karplus, K., Brown, M., Hughey, R., Krogh, A., Saira Mian, I., & Haussler, D. (1996). Dirichlet mixtures: A method for improved detection of weak but significant protein sequence homology. *Bioinformatics*. <https://doi.org/10.1093/bioinformatics/12.4.327>
- Stankiewicz, P., & Lupski, J. R. (2010). Structural Variation in the Human Genome and its Role in Disease. *Annual Review of Medicine*. <https://doi.org/10.1146/annurev-med-100708-204735>
- Stenson, P. D., Ball, E. V., Mort, M., Phillips, A. D., Shiel, J. A., Thomas, N. S. T., Abeyasinghe, S., Krawczak, M., & Cooper, D. N. (2003). Human Gene Mutation Database (HGMD®): 2003 Update. In *Human Mutation*. <https://doi.org/10.1002/humu.10212>
- Stergachis, A. B., Haugen, E., Shafer, A., Fu, W., Vernot, B., Reynolds, A., Raubitschek, A., Ziegler, S., LeProust, E. M., Akey, J. M., & Stamatoyannopoulos, J. A. (2013). Exonic transcription factor binding directs codon choice and affects protein evolution. *Science*. <https://doi.org/10.1126/science.1243490>
- Stone, E. A., & Sidow, A. (2005). Physicochemical constraint violation by missense substitutions mediates impairment of protein function and disease severity. *Genome Research*. <https://doi.org/10.1101/gr.3804205>

Stryer, L. (1995). Stryer Biochemistry. In *Biochemistry textbook*.

Supek, F., Miñana, B., Valcárcel, J., Gabaldón, T., & Lehner, B. (2014). Synonymous mutations frequently act as driver mutations in human cancers. *Cell*. <https://doi.org/10.1016/j.cell.2014.01.051>

Suzek, B. E., Huang, H., McGarvey, P., Mazumder, R., & Wu, C. H. (2007). UniRef: Comprehensive and non-redundant UniProt reference clusters. *Bioinformatics*. <https://doi.org/10.1093/bioinformatics/btm098>

Svetnik, V., Liaw, A., Tong, C., Christopher Culberson, J., Sheridan, R. P., & Feuston, B. P. (2003). Random Forest: A Classification and Regression Tool for Compound Classification and QSAR Modeling. *Journal of Chemical Information and Computer Sciences*. <https://doi.org/10.1021/ci034160g>

Szymczak, S., Biernacka, J. M., Cordell, H. J., González-Recio, O., König, I. R., Zhang, H., & Sun, Y. V. (2009). Machine learning in genome-wide association studies. *Genetic Epidemiology*. <https://doi.org/10.1002/gepi.20473>

Tavtigian, S. V., Byrnes, G. B., Goldgar, D. E., & Thomas, A. (2008). Classification of rare missense substitutions, using risk surfaces, with genetic- and molecular-epidemiology applications. *Human Mutation*. <https://doi.org/10.1002/humu.20896>

Tawfik, N. S., & Spruit, M. R. (2018). The SNPcurator: Literature mining of enriched SNP-disease associations. *Database*. <https://doi.org/10.1093/database/bay020>

The 1000 Genomes Project Consortium. (2012). An integrated map of genetic variation. *Nature*.

The H2O Team. (2015). *H2O: Scalable Machine Learning. Version 3.1.0.99999*. www.h2o.ai

The Protein Data Bank. (2003). *Methods of Biochemical Analysis*. <https://doi.org/10.4135/9781412994231.n75>

- The Proteomics Protocols Handbook. (2005). In *The Proteomics Protocols Handbook*. <https://doi.org/10.1385/1592598900>
- The UniProt Consortium. (2019). UniProt: a worldwide hub of protein knowledge The UniProt Consortium. *Nucleic Acids Research*. <https://doi.org/10.1093/nar/gky1049>
- Thomas, P. D., Campbell, M. J., Kejariwal, A., Mi, H., Karlak, B., Daverman, R., Diemer, K., Muruganujan, A., & Narechania, A. (2003). PANTHER: A library of protein families and subfamilies indexed by function. *Genome Research*. <https://doi.org/10.1101/gr.772403>
- Thomas, P. D., & Kejariwal, A. (2004). Coding single-nucleotide polymorphisms associated with complex vs. Mendelian disease: Evolutionary evidence for differences in molecular effects. *Proceedings of the National Academy of Sciences of the United States of America*. <https://doi.org/10.1073/pnas.0404380101>
- Tian, J., Wu, N., Guo, X., Guo, J., Zhang, J., & Fan, Y. (2007). Predicting the phenotypic effects of non-synonymous single nucleotide polymorphisms based on support vector machines. *BMC Bioinformatics*. <https://doi.org/10.1186/1471-2105-8-450>
- Verzilli, C. J., Whittaker, J. C., Stallard, N., & Chasman, D. (2005). A hierarchical Bayesian model for predicting the functional consequences of amino-acid polymorphisms. In *Journal of the Royal Statistical Society. Series C: Applied Statistics*. <https://doi.org/10.1111/j.1467-9876.2005.00478.x>
- Wainreb, G., Ashkenazy, H., Bromberg, Y., Starovolsky-Shitrit, A., Haliloglu, T., Ruppin, E., Avraham, K. B., Rost, B., & Ben-Tal, N. (2010). MuD: An interactive web server for the prediction of non-neutral substitutions using protein structural data. *Nucleic Acids Research*, 38(SUPPL. 2), 523–528. <https://doi.org/10.1093/nar/gkq528>
- Wang, W. Y. S., Barratt, B. J., Clayton, D. G., & Todd, J. A. (2005). Genome-wide association studies: Theoretical and practical concerns. In *Nature Reviews Genetics*. <https://doi.org/10.1038/nrg1522>

- Wang, Z., & Moulton, J. (2001). SNPs, protein structure, and disease. *Human Mutation*. <https://doi.org/10.1002/humu.22>
- Ward, L. D., & Kellis, M. (2012). Interpreting noncoding genetic variation in complex traits and human disease. In *Nature Biotechnology*. <https://doi.org/10.1038/nbt.2422>
- Waterhouse, A., Bertoni, M., Bienert, S., Studer, G., Tauriello, G., Gumienny, R., Heer, F. T., De Beer, T. A. P., Rempfer, C., Bordoli, L., Lepore, R., & Schwede, T. (2018). SWISS-MODEL: Homology modelling of protein structures and complexes. *Nucleic Acids Research*. <https://doi.org/10.1093/nar/gky427>
- Webb, B., & Sali, A. (2016). Comparative protein structure modeling using MODELLER. *Current Protocols in Bioinformatics*. <https://doi.org/10.1002/cpbi.3>
- Wei, Z., Wang, K., Qu, H. Q., Zhang, H., Bradfield, J., Kim, C., Frackleton, E., Hou, C., Glessner, J. T., Chiavacci, R., Stanley, C., Monos, D., Grant, S. F. A., Polychronakos, C., & Hakonarson, H. (2009). From disease association to risk assessment: An optimistic view from genome-wide association studies on type 1 diabetes. *PLoS Genetics*. <https://doi.org/10.1371/journal.pgen.1000678>
- Worachartcheewan, A., Shoombuatong, W., Pidetcha, P., Nopnithipat, W., Prachayasittikul, V., & Nantasenamat, C. (2015). Predicting metabolic syndrome using the random forest method. *Scientific World Journal*. <https://doi.org/10.1155/2015/581501>
- Wu, B., Abbott, T., Fishman, D., McMurray, W., Mor, G., Stone, K., Ward, D., Williams, K., & Zhao, H. (2003). Comparison of statistical methods for classification of ovarian cancer using mass spectrometry data. *Bioinformatics*. <https://doi.org/10.1093/bioinformatics/btg210>
- Xiong, P., Zhang, C., Zheng, W., & Zhang, Y. (2017). BindProfX: Assessing Mutation-Induced Binding Affinity Change by Protein Interface Profiles with Pseudo-Counts. *Journal of Molecular Biology*. <https://doi.org/10.1016/j.jmb.2016.11.022>
- Yampolsky, L. Y., & Stoltzfus, A. (2005). The exchangeability of amino acids in proteins. *Genetics*. <https://doi.org/10.1534/genetics.104.039107>

Yates, C. M., Filippis, I., Kelley, L. A., & Sternberg, M. J. E. (2014). SuSPect: Enhanced prediction of single amino acid variant (SAV) phenotype using network features. *Journal of Molecular Biology*. <https://doi.org/10.1016/j.jmb.2014.04.026>

Yates, C. M., & Sternberg, M. J. E. (2013). Proteins and domains vary in their tolerance of non-synonymous single nucleotide polymorphisms (nsSNPs). *Journal of Molecular Biology*. <https://doi.org/10.1016/j.jmb.2013.01.026>

Yue, P., Li, Z., & Moulton, J. (2005). Loss of protein structure stability as a major causative factor in monogenic disease. *Journal of Molecular Biology*, 353(2), 459–473. <https://doi.org/10.1016/j.jmb.2005.08.020>

Yue, P., & Moulton, J. (2006). Identification and analysis of deleterious human SNPs. *Journal of Molecular Biology*, 356(5), 1263–1274. <https://doi.org/10.1016/j.jmb.2005.12.025>

Zamyatnin, A. A. (1972). Protein volume in solution. In *Progress in Biophysics and Molecular Biology*. [https://doi.org/10.1016/0079-6107\(72\)90005-3](https://doi.org/10.1016/0079-6107(72)90005-3)

Zhu, Y., Stevens, R. G., Leaderer, D., Hoffman, A., Holford, T., Zhang, Y., Brown, H. N., & Zheng, T. (2008). Non-synonymous polymorphisms in the circadian gene NPAS2 and breast cancer risk. *Breast Cancer Research and Treatment*. <https://doi.org/10.1007/s10549-007-9565-0>

Ziegler, A. (2016). An Introduction to Statistical Learning with Applications. R. G. James, D. Witten, T. Hastie, and R. Tibshirani (2013). Berlin: Springer. 440 pages, ISBN: 978-1-4614-7138-7. *Biometrical Journal*. <https://doi.org/10.1002/bimj.201500224>

Zwart, M. P., Schenk, M. F., Hwang, S., Koopmanschap, B., de Lange, N., van de Pol, L., Nga, T. T. T., Szendro, I. G., Krug, J., & de Visser, J. A. G. M. (2018). Unraveling the causes of adaptive benefits of synonymous mutations in TEM-1 β -lactamase. *Heredity*. <https://doi.org/10.1038/s41437-018-0104-z>



APPENDICES

APPENDIX A

The table here shows selected domains and their counts for each class after Fisher's exact test analysis along with the significance of association with 99% confidence interval.

Domain Name	Total Number of Observations	Deleterious Observations	Neutral Observations	p-value
IPR011162	632	10	622	2,64E-237
IPR017452	1918	550	1368	1,14E-196
IPR005821	2095	1905	190	5,89E-178
IPR009050	530	48	482	8,26E-137
IPR013783	2908	1290	1618	1,55E-82
IPR017853	956	840	116	4,30E-52
IPR013087	607	181	426	8,36E-48
IPR017850	780	687	93	5,91E-41
IPR008972	462	430	32	3,66E-36
IPR019774	293	285	8	5,36E-31
IPR013092	474	429	45	1,63E-26
IPR008967	505	451	54	7,53E-26
IPR015919	680	265	415	1,23E-21
IPR027417	3726	2687	1039	1,09E-16
IPR011146	221	213	8	1,45E-16
IPR013088	199	194	5	5,83E-17
IPR009057	513	441	72	1,03E-14
IPR008948	194	188	6	5,97E-14
IPR001909	67	0	67	8,32E-15
IPR000536	499	427	72	7,59E-14
IPR017943	79	4	75	6,89E-13
IPR022772	168	164	4	1,25E-13
IPR032675	550	228	322	4,61E-11
IPR000742	1492	1124	368	3,86E-08
IPR013320	547	232	315	6,45E-08
IPR029021	432	363	69	9,67E-08
IPR006132	121	119	2	3,05E-05
IPR014710	230	208	22	4,08E-06
IPR020858	76	9	67	2,07E-04
IPR029063	268	95	173	3,79E-04
IPR016176	121	118	3	4,86E-05

IPR023796	293	108	185	9,74E-05
IPR006131	113	110	3	1,91E-02
IPR029057	112	109	3	3,29E-02
IPR000436	255	93	162	7,08E-03
IPR009100	159	147	12	1,88E-01
IPR011991	546	434	112	1,22E-01
IPR001846	104	26	78	3,52E+00
IPR008922	134	124	10	4,50E-01
IPR023214	381	310	71	1,15E+02
IPR002181	78	16	62	2,84E+02
IPR022417	64	64	0	2,20E+03
IPR013821	85	82	3	3,29E+03
IPR003644	71	15	56	8,33E+02
IPR015424	415	329	86	1,58E+04
IPR032695	77	18	59	1,71E+04
IPR001478	105	31	74	3,28E+03
IPR024041	51	8	43	3,98E+03
IPR008280	131	118	13	4,01E+04
IPR014743	182	157	25	4,57E+04
IPR004273	93	26	67	6,59E+03
IPR008930	70	16	54	7,56E+02
IPR015813	78	75	3	7,62E+03
IPR004841	89	84	5	9,39E+03
IPR003533	118	107	11	1,70E+05
IPR023210	57	11	46	1,92E+04
IPR012674	43	6	37	3,78E+04
IPR027387	34	3	31	5,68E+05
IPR007696	149	130	19	6,38E+04
IPR009071	108	98	10	8,10E+04
IPR001811	23	0	23	1,06E+06
IPR017448	54	11	43	1,74E+06
IPR011042	454	350	104	2,20E+05
IPR024074	57	56	1	2,43E+05
IPR000585	38	5	33	2,67E+05
IPR001926	101	92	9	2,67E+06
IPR028992	49	49	0	3,31E+06
IPR022636	55	54	1	4,17E+05
IPR001424	94	86	8	5,04E+05
IPR009075	151	130	21	5,11E+05
IPR011990	323	149	174	5,91E+05
IPR023298	83	77	6	6,99E+05
IPR022673	141	122	19	7,48E+04
IPR022675	77	72	5	8,00E+04
IPR016137	31	3	28	8,60E+05
IPR010982	71	67	4	1,31E+07
IPR012336	146	56	90	1,43E+07
IPR015943	702	365	337	1,49E+05

IPR018884	51	11	40	1,66E+07
IPR020683	381	183	198	1,84E+07
IPR008984	121	106	15	2,15E+07
IPR016024	999	540	459	2,98E+06
IPR006207	74	69	5	3,34E+07
IPR022672	111	98	13	4,33E+06
IPR008979	295	233	62	4,68E+04
IPR013816	77	71	6	7,65E+05
IPR009045	55	53	2	8,30E+06
IPR013780	87	79	8	8,45E+06
IPR029332	18	0	18	1,56E+08
IPR003594	165	137	28	2,54E+08
IPR015425	21	1	20	2,88E+08
IPR029061	140	118	22	5,13E+08
IPR002912	36	36	0	7,84E+07
IPR001452	151	63	88	1,05E+09
IPR027936	16	0	16	1,15E+09
IPR013680	16	0	16	1,15E+09
IPR000859	116	45	71	1,15E+09
IPR000980	232	184	48	1,18E+09
IPR016185	58	54	4	1,86E+09
IPR023578	30	5	25	2,30E+08
IPR013785	179	145	34	2,31E+09
IPR011764	34	34	0	2,32E+09
IPR008250	175	142	33	2,56E+09
IPR029006	49	13	36	2,73E+08
IPR029047	15	0	15	3,12E+08
IPR000157	37	8	29	3,45E+08
IPR027309	32	6	26	3,47E+09
IPR011992	260	124	136	4,11E+09
IPR013806	63	20	43	5,59E+08
IPR003331	38	37	1	7,41E+08
IPR009051	38	37	1	7,41E+08
IPR024732	31	31	0	7,48E+08
IPR008519	14	0	14	8,47E+07
IPR032455	14	0	14	8,47E+07
IPR028889	47	13	34	9,75E+08
IPR008983	229	179	50	1,27E+10
IPR000472	54	50	4	1,28E+10
IPR000834	35	8	27	1,49E+10
IPR000873	55	17	38	1,68E+10
IPR002035	362	184	178	1,78E+10
IPR006134	13	0	13	2,30E+10
IPR020602	41	39	2	2,48E+10
IPR017878	86	74	12	3,11E+10
IPR003008	91	78	13	3,12E+10
IPR001098	61	55	6	3,20E+10

IPR031162	51	47	4	3,23E+10
IPR000326	51	47	4	3,23E+10
IPR020843	16	1	15	3,27E+10
IPR013847	28	28	0	3,75E+10
IPR029041	40	38	2	4,01E+09
IPR023416	89	76	13	4,55E+09
IPR032189	104	87	17	5,65E+09
IPR016187	231	112	119	5,69E+09
IPR001322	63	56	7	5,71E+09
IPR006158	44	41	3	6,30E+09
IPR000477	27	27	0	6,73E+10
IPR009048	18	2	16	7,51E+10
IPR012677	52	17	35	9,50E+08
IPR025837	38	36	2	1,11E+11
IPR010987	72	27	45	1,24E+11
IPR017981	118	51	67	1,33E+11
IPR001320	166	131	35	1,64E+11
IPR024810	11	0	11	1,70E+10
IPR006612	31	30	1	1,72E+11
IPR009080	32	8	24	2,07E+11
IPR029039	27	6	21	2,14E+11
IPR031160	25	5	20	2,33E+10
IPR022418	30	29	1	2,93E+11
IPR029052	163	128	35	2,93E+10
IPR001007	52	18	34	3,63E+11
IPR003010	105	86	19	4,09E+10
IPR001156	24	5	19	4,47E+10
IPR008916	10	0	10	4,60E+10
IPR014044	10	0	10	4,60E+10
IPR011657	10	0	10	4,60E+10
IPR015798	10	0	10	4,60E+10
IPR021072	10	0	10	4,60E+10
IPR029020	10	0	10	4,60E+10
IPR002190	21	4	17	4,61E+10
IPR000569	61	53	8	5,05E+11
IPR033118	29	28	1	5,08E+10
IPR008274	13	1	12	5,36E+11
IPR011527	873	494	379	6,40E+08
IPR016039	52	46	6	6,97E+10
IPR007860	68	58	10	7,51E+09
IPR015794	27	26	1	8,56E+10
IPR001750	36	11	25	8,75E+10
IPR015252	46	41	5	0,00
IPR003112	75	63	12	1,01E-04
IPR009011	40	13	27	1,04E-04
IPR007110	15	2	13	1,05E-04

IPR002791	9	0	9	1,25E-04
IPR028565	9	0	9	1,25E-04
IPR002502	9	0	9	1,25E-04
IPR024779	9	0	9	1,25E-04
IPR018292	9	0	9	1,25E-04
IPR001932	9	0	9	1,25E-04
IPR001024	60	23	37	1,30E-04
IPR018484	12	1	11	1,35E-04
IPR007111	12	1	11	1,35E-04
IPR000312	41	37	4	1,35E-04
IPR016177	66	56	10	1,65E-04
IPR000257	49	43	6	1,66E-04
IPR014010	33	10	23	1,81E-04
IPR004865	19	19	0	1,92E-04
IPR012932	19	19	0	1,92E-04
IPR002350	43	15	28	2,00E-04
IPR006545	20	20	0	2,04E-04
IPR032200	20	20	0	2,04E-04
IPR011029	153	74	79	2,05E-04
IPR001879	37	12	25	2,18E-04
IPR001296	30	28	2	2,24E-04
IPR010991	48	42	6	2,52E-04
IPR000034	32	10	22	3,24E-04
IPR024571	8	0	8	3,39E-04
IPR029155	8	0	8	3,39E-04
IPR031320	8	0	8	3,39E-04
IPR012308	8	0	8	3,39E-04
IPR027007	8	0	8	3,39E-04
IPR012319	8	0	8	3,39E-04
IPR032431	8	0	8	3,39E-04
IPR032471	8	0	8	3,39E-04
IPR002870	8	0	8	3,39E-04
IPR010630	8	0	8	3,39E-04
IPR013035	8	0	8	3,39E-04
IPR011547	118	93	25	3,63E-04
IPR013057	30	9	21	4,02E-04
IPR001214	103	82	21	4,50E-04
IPR001763	27	8	19	4,84E-04
IPR003191	35	12	23	6,51E-04
IPR014853	20	5	15	6,91E-04
IPR009254	20	5	15	6,91E-04
IPR029030	47	18	29	6,94E-04
IPR003599	18	4	14	7,40E-04

IPR011598	93	74	19	7,86E-04
IPR016040	504	354	150	8,35E-04
IPR010994	10	1	9	8,36E-04
IPR003137	10	1	9	8,36E-04
IPR027357	7	0	7	9,21E-04
IPR010926	7	0	7	9,21E-04
IPR031907	7	0	7	9,21E-04
IPR031474	7	0	7	9,21E-04
IPR017854	7	0	7	9,21E-04
IPR011008	7	0	7	9,21E-04
IPR000922	7	0	7	9,21E-04
IPR003609	63	52	11	9,94E-04
IPR006594	16	16	0	1,01E-03
IPR000197	21	20	1	1,13E-03
IPR026831	21	20	1	1,13E-03
IPR020568	122	94	28	1,31E-03
IPR016093	62	51	11	1,41E-03
IPR013121	25	23	2	1,58E-03
IPR000699	54	45	9	1,68E-03
IPR013158	14	3	11	1,80E-03
IPR032630	14	3	11	1,80E-03
IPR025766	15	15	0	1,83E-03
IPR027841	9	1	8	2,05E-03
IPR021040	6	0	6	2,50E-03
IPR003726	6	0	6	2,50E-03
IPR005302	6	0	6	2,50E-03
IPR016182	6	0	6	2,50E-03
IPR010979	6	0	6	2,50E-03
IPR000674	6	0	6	2,50E-03
IPR000917	6	0	6	2,50E-03
IPR013992	6	0	6	2,50E-03
IPR031437	6	0	6	2,50E-03
IPR024309	6	0	6	2,50E-03
IPR004102	6	0	6	2,50E-03
IPR001194	6	0	6	2,50E-03
IPR001180	6	0	6	2,50E-03
IPR007725	6	0	6	2,50E-03
IPR001599	6	0	6	2,50E-03
IPR008942	6	0	6	2,50E-03
IPR013697	6	0	6	2,50E-03
IPR002999	6	0	6	2,50E-03
IPR000772	16	4	12	2,82E-03
IPR029071	103	50	53	2,86E-03

IPR012675	41	35	6	3,03E-03
IPR008121	13	13	0	3,12E-03
IPR001107	37	32	5	3,14E-03
IPR003619	37	32	5	3,14E-03
IPR029067	14	14	0	3,41E-03
IPR001736	14	14	0	3,41E-03
IPR002219	94	73	21	3,63E-03
IPR000731	47	39	8	3,86E-03
IPR014729	209	152	57	3,98E-03
IPR031701	23	21	2	4,04E-03
IPR033644	23	21	2	4,04E-03
IPR000375	35	30	5	4,62E-03
IPR031481	24	8	16	4,66E-03
IPR002919	28	10	18	4,97E-03
IPR007943	8	1	7	4,99E-03
IPR008253	8	1	7	4,99E-03
IPR027397	8	1	7	4,99E-03
IPR000294	68	54	14	5,31E-03
IPR000315	17	5	12	5,33E-03
IPR024240	12	12	0	5,37E-03
IPR013803	12	12	0	5,37E-03
IPR002100	12	12	0	5,37E-03
IPR034154	12	12	0	5,37E-03
IPR032419	12	12	0	5,37E-03
IPR009061	12	12	0	5,37E-03
IPR010536	12	12	0	5,37E-03
IPR001048	18	17	1	5,40E-03
IPR016035	136	70	66	5,67E-03
IPR016090	13	3	10	6,44E-03
IPR008928	72	34	38	6,78E-03
IPR029017	5	0	5	6,79E-03
IPR010600	5	0	5	6,79E-03
IPR029190	5	0	5	6,79E-03
IPR031688	5	0	5	6,79E-03
IPR025232	5	0	5	6,79E-03
IPR027859	5	0	5	6,79E-03
IPR002668	5	0	5	6,79E-03
IPR029048	5	0	5	6,79E-03
IPR022409	5	0	5	6,79E-03
IPR001303	5	0	5	6,79E-03
IPR006782	5	0	5	6,79E-03
IPR006116	5	0	5	6,79E-03
IPR004021	5	0	5	6,79E-03

IPR022764	5	0	5	6,79E-03
IPR010918	5	0	5	6,79E-03
IPR021989	5	0	5	6,79E-03
IPR002589	5	0	5	6,79E-03
IPR032746	5	0	5	6,79E-03
IPR019162	5	0	5	6,79E-03
IPR029312	5	0	5	6,79E-03
IPR031442	5	0	5	6,79E-03
IPR029388	5	0	5	6,79E-03
IPR006581	5	0	5	6,79E-03
IPR008332	5	0	5	6,79E-03
IPR029064	10	2	8	6,93E-03
IPR027295	10	2	8	6,93E-03
IPR001111	79	38	41	6,99E-03
IPR017946	39	16	23	7,00E-03
IPR002937	26	23	3	7,08E-03
IPR001148	190	102	88	8,23E-03
IPR025659	16	15	1	8,79E-03
IPR005480	16	15	1	8,79E-03
IPR020630	16	15	1	8,79E-03
IPR009078	41	34	7	8,81E-03
IPR009014	17	16	1	9,34E-03
IPR013234	11	11	0	9,45E-03
IPR002474	11	11	0	9,45E-03
IPR024986	11	11	0	9,45E-03
IPR006644	11	11	0	9,45E-03
IPR012429	11	11	0	9,45E-03
IPR011162	632	10	622	0,00

APPENDIX B

Tables in this Appendix shows Fisher's exact test contingency table results for three protein regions per physicochemical property. Data points are classified for the regions their mutations occur as surface, core and interface. Physicochemical properties are divided as significant and non-significant based on the distribution they show in the data. Later on, association between the significance of properties and deleteriousness outcome is investigated.

CORE – VOLUME

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	4548	3149	1.44
Significant	6154	1932	3.19
Total	10702	5081	2.11
p-value	1.96e-116		

CORE – COMPOSITION

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	4915	2785	1.76
Significant	5787	2296	2.52
Total	10702	5081	2.11
p-value	1.97e-25		

CORE – POLARITY

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	4704	3110	1.51
Significant	5998	1971	3.04
Total	10702	5081	2.11
p-value	1.05e-91		

CORE – GRANTHAM SCORE

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	4270	3186	1.34
Significant	6432	1895	3.39
Total	10702	5081	2.11
p-value	3.27e-159		

INTERFACE – VOLUME

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	3096	2163	1.43
Significant	4181	1866	2.24
Total	7277	4029	1.81
p-value	7.13e-30		

INTERFACE – COMPOSITION

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	3417	2026	1.67
Significant	3860	2003	1.93
Total	7277	4029	1.81
p-value	0.00072		

INTERFACE – POLARITY

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	3143	2122	1.48
Significant	4134	1907	2.17
Total	7277	4029	1.81
p-value	4.73e-22		

INTERFACE – GRANTHAM SCORE

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	2970	2127	1.40
Significant	4307	1902	2.26
Total	7277	4029	1.81
p-value	2.06e-34		

SURFACE – VOLUME

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	12198	19349	0.63
Significant	16601	14906	1.11
Total	28799	34255	0.84
p-value	1.44e-3274		

SURFACE – COMPOSITION

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	13254	18489	0.72
Significant	15545	15766	0.98
Total	28799	34255	0.84
p-value	3.93e-88		

SURFACE – POLARITY

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	12829	19875	0.65
Significant	15970	14380	1.11
Total	28799	34255	0.84
p-value	6.18e-250		

SURFACE – GRANTHAM SCORE

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	11757	20777	0.56
Significant	17042	13478	1.26
Total	28799	34255	0.84
p-value	0.0		

APPENDIX C

Tables in this Appendix shows Fisher's exact test contingency table results for five protein families per physicochemical property. Data points are classified for the families that the protein is a member of. Physicochemical properties are divided as significant and non-significant based on the distribution they show in the data. Later on, association between the significance of properties and deleteriousness outcome is investigated.

TRANSCRIPTION FACTOR – VOLUME

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	552	196	2.82
Significant	698	122	5.72
Total	1250	318	3.93
p-value	2.75e-08		

TRANSCRIPTION FACTOR – COMPOSITION

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	608	151	4.03
Significant	642	167	3.84
Total	1250	318	3.93
p-value	0.753		

TRANSCRIPTION FACTOR – POLARITY

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	584	162	3.60
Significant	666	156	4.27
Total	1250	318	3.93
p-value	0.187		

TRANSCRIPTION FACTOR – GRANTHAM SCORE

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	511	159	3.21
Significant	739	159	4.65
Total	1250	318	3.93
p-value	0.0035		

ENZYME – VOLUME

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	6748	4558	1.48
Significant	9072	3378	2.69
Total	15820	7936	1.99
p-value	9.55e-103		

ENZYME – COMPOSITION

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	7283	4286	1.70
Significant	8537	3650	2.34
Total	15820	7936	1.99
p-value	4.45e-31		

ENZYME – POLARITY

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	7168	4575	1.57
Significant	8652	3361	2.57
Total	15820	3361	1.99
p-value	4.37e-72		

ENZYME – GRANTHAM SCORE

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	6491	4701	1.38
Significant	9329	3235	2.88
Total	15820	7936	1.99
p-value	3.27e-155		

ION CHANNEL – VOLUME

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	1792	642	2.79
Significant	2278	485	4.70
Total	4070	1127	3.61
p-value	1.63e-14		

ION CHANNEL – COMPOSITION

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	1990	631	3.15
Significant	2080	496	4.19
Total	4070	1127	3.61
p-value	2.55e-05		

ION CHANNEL – POLARITY

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	2090	683	3.06
Significant	1980	444	4.46
Total	4070	1127	3.61
p-value	3.67e-08		

ION CHANNEL – GRANTHAM SCORE

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	1920	707	2.72
Significant	2150	420	5.12
Total	4070	1127	3.61
p-value	2.04e-20		

MEMBRANE – VOLUME

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	1017	876	1.16
Significant	1373	668	2.06
Total	2390	1544	1.55
p-value	4.14e-18		

MEMBRANE – COMPOSITION

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	957	794	2.21
Significant	1433	750	1.91
Total	2390	1544	1.55
p-value	2.50e-12		

MEMBRANE – POLARITY

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	1030	824	1.25
Significant	1360	720	0.88
Total	2390	1544	1.55
p-value	3.28e-10		

MEMBRANE – GRANTHAM SCORE

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	870	834	1.04
Significant	1520	710	2.14
Total	2390	1544	1.55
p-value	1.58e-27		

OTHERS – VOLUME

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	1033	985	1.05
Significant	1683	775	2.17
Total	2716	1760	1.54
p-value	5.04e-32		

OTHERS – COMPOSITION

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	1273	865	1.47
Significant	1443	895	1.61
Total	2716	1760	1.54
p-value	0.1415		

OTHERS – POLARITY

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	1169	1022	1.14
Significant	1547	738	2.09
Total	2716	1760	1.54
p-value	9.71e-23		

OTHERS – GRANTHAM SCORE

	Deleterious	Neutral	Proportion (del/neut)
Not Significant	1058	952	1.11
Significant	1658	808	2.05
Total	2716	1760	1.54
p-value	3.20e-23		

APPENDIX D

The table below shows the selected parameters for the finalized prediction model.

label	type	level	selected_value	default_value
model_id	Key	critical	grid-40823815-b7a5-4e5f-bbae-9776fc38dc90_model_2	
training_frame	Key	critical	newtrain_fisher	
validation_frame	Key	critical	newtest_fisher	
nfolds	int	critical	10	0
keep_cross_validation_models	boolean	expert	false	true
keep_cross_validation_predictions	boolean	expert	false	false
keep_cross_validation_fold_assignment	boolean	expert	false	false
score_each_iteration	boolean	secondary	false	false
score_tree_interval	int	secondary	0	0
fold_assignment	enum	secondary	AUTO	AUTO
fold_column	VecSpecifier	secondary	.	
response_column	VecSpecifier	critical	disease	
ignored_columns	string[]	critical	uniprotIDwtmutpos	
ignore_const_cols	boolean	critical	true	true
offset_column	VecSpecifier	secondary	.	
weights_column	VecSpecifier	secondary	.	
balance_classes	boolean	secondary	true	false
class_sampling_factors	float[]	expert	.	
max_after_balance_size	float	expert	5	5
max_confusion_matrix_size	int	secondary	20	20
max_hit_ratio_k	int	secondary	0	0
ntrees	int	critical	100	50
max_depth	int	critical	20	20
min_rows	double	critical	1	1
nbins	int	critical	20	20
nbins_top_level	int	secondary	1024	1024
nbins_cats	int	secondary	1024	1024

r2_stopping	double	secondary	1.7976931348623157e+308	1.7976931348623157e+308
stopping_rounds	int	secondary	0	0
stopping_metric	enum	secondary	AUTO	AUTO
stopping_tolerance	double	secondary	0.001	0.001
max_runtime_secs	double	secondary	0	0
seed	long	critical	-3279707484346758770	-1
build_tree_one_node	boolean	expert	false	false
mtries	int	critical	-1	-1
sample_rate	double	critical	0.632	0.632
sample_rate_per_class	double[]	expert	.	
binomial_double_trees	boolean	expert	false	false
checkpoint	Key	secondary	.	
col_sample_rate_change_per_level	double	expert	1	1
col_sample_rate_per_tree	double	secondary	1	1
min_split_improvement	double	secondary	0.00001	0.00001
histogram_type	enum	secondary	AUTO	AUTO
categorical_encoding	enum	secondary	AUTO	AUTO
calibrate_model	boolean	expert	false	false
calibration_frame	Key	expert	.	
distribution	enum	secondary	multinomial	AUTO
custom_metric_func	string	secondary	.	
export_checkpoints_dir	string	secondary	.	