AN INTEGRATIVE FRAMEWORK FOR CLINICAL DIAGNOSIS AND KNOWLEDGE DISCOVERY FROM EXOME SEQUENCING DATA

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

MONA SHOJAEI

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY
IN
MEDICAL INFORMATICS

SEPTEMBER 2023

AN INTEGRATIVE FRAMEWORK FOR CLINICAL DIAGNOSIS AND KNOWLEDGE DISCOVERY FROM EXOME SEQUENCING DATA

Submitted by MONA SHOJAEI in partial fulfillment of the requirements for the degree of **Doctor of Philosophy in Health Informatics, Middle East Technical University** by,

Date: **11.09.2023**

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 : Mona Shojaei

Signature : _____

ABSTRACT

AN INTEGRATIVE FRAMEWORK FOR CLINICAL DIAGNOSIS AND KNOWLEDGE DISCOVERY FROM EXOME SEQUENCING DATA

Shojaei, Mona

Ph.D, Department of Health Informatics

Supervisor: Assist. Prof. Dr. Aybar Can Acar

September 2023, 92 pages

Non-silent single nucleotide genetic variants, like nonsense changes and insertion-deletion variants (indels), that affect protein function and length substantially are prevalent and are frequently misclassified. The low sensitivity and specificity of existing variant effect predictors for nonsense and indel variations restrict their use in clinical applications. We propose the Pathogenic Mutation Prediction (PMPred) method to predict the pathogenicity of single nucleotide variations (SNVs), which impair protein function by prematurely terminating a protein's elongation during its synthesis. The prediction starts by monitoring functional effects (Gene Ontology annotation changes) of the change in sequence, using an existing ensemble machine learning model (UniGOPred). This, in turn, reveals the mutations that significantly deviate functionally from the wild-type sequence. We have identified novel harmful mutations in patient data and present them as motivating case studies. We also show that our method has increased sensitivity and specificity compared to state-of-the-art, especially in SNVs that produce large functional changes in the final protein. As further validation, we have done a comparative docking study on such a variation that is misclassified by existing methods and, using the altered binding affinities, show how PMPred can correctly predict the pathogenicity when other tools miss it. PMPred is freely accessible as a web service at https://pmpred.kansil.org/, and the related code is available at https://github.com/kansil/PMPred.

Keywords: variant pathogenicity prediction, gene ontology, pearson correlation, protein sequence, insertion-deletion variants

ÖZ

KLİNİK TEŞHİS VE BİLGİ KEŞFİ İÇİN EKZOM SEKANSLAMA VERİLERİNE DAYALI BÜTÜNLEŞİK BİR SİSTEM

Shojaei, Mona

Doktora, Sağlık Bilişimi Bölümü

Tez Yöneticisi: Dr. Öğr. Üyesi Aybar Can Acar

Eylül 2023, 92 sayfa

Protein işlevini ve uzunluğunu önemli ölçüde etkileyen nonsense değişiklikler ve insersiyon-delesyon varyantları (indeller) gibi sessiz olmayan tek nükleotid genetik varyantlar yaygındır ve sıklıkla yanlış sınıflandırılır. Nonsense ve indel varyasyonları için mevcut varyant etki tahmincilerinin düşük duyarlılığı ve özgüllüğü, klinik uygulamalarda kullanımlarını kısıtlamaktadır. Bir proteinin sentezi sırasında uzamasını zamanından önce sonlandırarak protein işlevini bozan tek nükleotid varyasyonlarının (SNV'ler) patojenitesini tahmin etmek için Patojenik Mutasyon Tahmini (PMPred) yöntemini öneriyoruz. Tahmin, mevcut bir topluluk makine öğrenme modelini (UniGOPred) kullanarak dizideki değişikliğin işlevsel etkilerini (Gen Ontolojisi ek açıklama değişiklikleri) izleyerek başlar. Bu da işlevsel olarak vahşi tip sekanstan önemli ölçüde sapan mutasyonları ortaya çıkarır. Hasta verilerinde yeni zararlı mutasyonlar tespit ettik ve bunları motive edici vaka çalışmaları olarak sunduk. Ayrıca yöntemimizin, özellikle nihai proteinde büyük işlevsel değişiklikler üreten SNV'lerde, en son teknolojiye kıyasla daha yüksek hassasiyet ve özgüllüğe sahip olduğunu gösteriyoruz. Daha fazla doğrulama olarak, mevcut yöntemlerle yanlış sınıflandırılan böyle bir varyasyon üzerinde karşılaştırmalı bir docking çalışması yaptık ve değişen bağlanma yakınlıklarını kullanarak, PMPred'in diğer araçlar bunu kaçırdığında patojeniteyi nasıl doğru bir şekilde tahmin edebileceğini gösterdik. PMPred'e https://pmpred.kansil.org/ adresinden bir web hizmeti olarak ücretsiz erişilebilir ve ilgili kod https://github.com/kansil/PMPred adresinde mevcuttur.

Anahtar Sözcükler: varyant patojenite tahmini, gen ontolojisi, pearson korelasyonu, protein dizisi, insersiyon-delesyon varyantları

To My Family

ACKNOWLEDGMENTS

To My Beloved Family and Loved Ones

Words cannot express the depth of my gratitude for your constant support, love, and encouragement throughout this incredible journey. You have been my pillars of strength, my guiding stars, and the source of my inspiration. To my parents, thank you for instilling in me the values of perseverance and curiosity from an early age. Your sacrifices and steadfast faith in my abilities have made this achievement possible. To my sister, Afra, your constant encouragement and understanding have been my refuge in times of challenge.

To my partner, your patience, love, and understanding have sustained me through countless late nights and moments of doubt. I deeply appreciate your presence and commitment to my journey. To my extended family, your belief in my abilities and your constant cheering from the sidelines have meant the world to me.

To My Esteemed Professors

I am deeply honored to express my gratitude to my esteemed professors, who have not only shared their knowledge but also shaped my intellectual growth. Your guidance, mentorship, and insightful feedback have been invaluable. To my advisor, Assist. Prof. Dr. Aybar Can Acar, your unwavering support and guidance have been instrumental in shaping the direction of my research. I am grateful for your patience, wisdom, and the countless hours you dedicated to helping me grow as a researcher. To my co-advisor, Assoc. Prof. Dr. Can Alkan, I am profoundly thankful for your collaborative spirit, which have enriched my academic journey and contributed significantly to the completion of this Ph.D. thesis

To my committee members, your expertise, constructive criticism, and encouragement have been instrumental in refining my work. Your belief in the significance of my research has motivated me to push the boundaries of knowledge.

To My Wonderful Friends

To my dear friends, you have been my source of laughter, solace, and inspiration beyond measure. Your friendship has made this journey unforgettable, and your unwavering belief in me has been a constant source of strength. Whether it was late-night study sessions or moments of celebration, you made every step of this journey memorable.

In conclusion, this Ph.D. thesis represents the culmination of years of hard work, dedication, and the unwavering support of my family, loved ones, professors, and friends. Each of you has played an indispensable role in my academic and personal growth, and for that, I am profoundly thankful.

With heartfelt gratitude,

TABLE OF CONTENTS

ABSTRACT iv
ÖZ v
DEDICATION vi
ACKNOWLEDGMENTS vii
TABLE OF CONTENTS ix
LIST OF TABLES xiii
LIST OF FIGURES xiv
LIST OF ABBREVIATIONS xv
CHAPTERS
1. INTRODUCTION 1
1.1. Motivation 1
1.2. Background 3
1.3. Summary of contribution 4
1.4. Outline 5
2. BACKGROUND AND RELATED WORKS 7
2.1. In silico prediction methods 11
2.1.1. Sequence-based prediction methods 11
2.1.1.1. Sift 11
2.1.1.2. Provean 11
2.1.1.3. MutPred 11
2.1.1.4. PolyPhen 12
2.1.1.5. MutationTaster 12
2.1.1.6. DDIG-in 13
2.1.1.7. GERP++ 13
2.1.2. Structure-based prediction methods 14
2.1.2.1. mCSM 14

2.1.2.2. FoldX 14
2.1.2.3. SDM 15
2.1.2.4. I-Mutant Suite 15
2.1.2.5. MuPro 16
2.1.3. Hybrid-based prediction methods 16
2.1.3.1. CADD 16
2.1.3.2. Revel 17
2.1.3.3. Vest 17
2.1.3.4. Meta-SNP 18
2.1.3.5. FATHMM 18
2.1.3.6. MetaLR 19
2.2. Performance evaluation metrics 19
2.2.1. Common performance evaluation metrics for genetic variant prediction tools 19
2.2.1.1. Sensitivity/Recall 19
2.2.1.2. Specificity 20
2.2.1.3. Precision 20
2.2.1.4. F1 score 21
2.2.1.5. Matthews correlation coefficient (MCC) 21
2.2.1.6. Receiver operating characteristic curve (ROC curve) 21
2.2.1.7. Area under the ROC curve (AUC-ROC) 22
2.2.1.8. Confusion matrix 22
3. MATERIALS AND METHODS 23
3.1. Materials 23
3.1.1. Data description 23
3.1.2. Annovar 26
3.1.3. UniGOPred 27
3.1.4. InterProScan 28
3.1.5. Uniprot 30
3.1.6. Alphafold 31
3.1.7. SwissModel 32

3.1.8. AutoDock Vina 33
3.1.9. MGLTools 34
3.1.10. PyMOL 35
3.1.11. Nextflow 36
3.1.12. Flask 37
3.1.13. Mailjet 37
3.2. Method 38
3.2.1. Data collection 39
3.2.2. Variant annotation 41
3.2.3. Format conversion 43
3.2.4. Function prediction 44
3.2.5. Domain prediction 45
3.2.6. Subcellular location prediction 47
3.2.7. Pathogenicity prediction 49
3.3. Performance assessment 51
3.3.1. Benchmarking with ClinVar dataset 51
3.3.2. Case study 51
3.3.2.1. Structure of the wildtype CFH 52
3.3.2.2. Structure of the mutated CFH 52
3.3.2.3. Structure of the binding ligand 53
3.3.2.4. Docking 55
3.4. Pipeline automation 55
3.5. Web interface integration 56
3.6. Sending results to the user 56
3.7. Availability 57
4. RESULTS AND DISCUSSION 59
4.1. Performance evaluation 60
4.1.1. Multi-method benchmark set 60
4.1.1.1. Precision and sensitivity 60
4.1.1.2. ROC curve analysis 61
4.1.2. Docking with CADD's unrecognized pathogenic mutation 62

4.1.3. Case study with clinical exome ssequencing data 64
4.2. Analysis of the impact of prediction features on PMPred performance using the decision tree approach 64
4.3. Evaluation of PMPred on clinical exome and gnomAD data and statistical method selection 65
4.4. Precision and reliability of predictions 66
4.5. Importance of functional measurement feature in accurate categorization of nonsense and frameshift variants 67
4.6. Validation through docking and literature comparison 67
4.7. Normality of benchmark set 69
4.8. Limitations 70
5. CONCLUSION 73
5.1. Research objective 73
5.2. Contribution 74
5.3. Future directions 75
REFERENCES 77
APPENDIX 87
APPENDIX A 87
CURRICULUM VITAE 92

LIST OF TABLES

Table 1: Prediction accuracies of PMPred, CADD, and VEST-indel, in predicting (a) exclusively nonsense mutations and (b) all mutations in the test set. 61

LIST OF FIGURES

Figure 1: Overview of ACMG guidelines for classification of pathogenic and benign variants. 2
Figure 2: Comparison of Protein Products: Normal vs. Mutated Gene. 7
Figure 3: Transcript Structure and Impactful Mutation Regions. 9
Figure 4: Guidelines for Variant Analysis in Next Generation Sequencing (NGS). 10
Figure 5: The workflows in Nextflow are depicted as graphs, with nodes representing individual processes and edges representing the channels that connect them. 36
Figure 6: The PMPred Workflow.....38
Figure 7: Filtered variants for analysis based on CLNSIG, CLNREVSTAT, and GENEINFO criteria.....40
Figure 8: PMPred Web Server and Workflow. 58
Figure 9: TP vs. FP rate at different classification thresholds. 62
Figure 10: The docking of CFH to its ligand ITGAM. 63
Figure 11: Decision tree: 197 samples of benign (Class = B) and pathogenic (Class = P) variants. 65
Figure 12: A) Wildtype Q-Q plot, B) Mutated Q-Q plot, C) Wildtype normality plot, D) Mutated normality plot. 69

LIST OF ABBREVIATIONS

SNVs	Single Nucleotide Variations
ACMG	American College of Medical Genetics and Genomics
P	Pathogenic
LP	Likely Pathogenic
VUS	Uncertain Significance
LB	Likely Benign
B	Benign
TGP	1000 Genomes Project
HGMD	Human Gene Mutation Database
CADD	Combined Annotation-Dependent Depletion
PMPred	Pathogenic Mutation Prediction
MAF	Minor Allele Frequency
DDIG-in	Disease-Defining Genomic Information
FS	Frameshift
NS	Nonsense
SFFS	sequential forward floating selection
GERP	Genomic Evolutionary Rate Profiling
mCSM	Mutation Cutoff Scanning Matrix
SVR	Support Vector Regression
RF	Random Forest
SDM	Site-Directed Mutator
Tm	Melting Temperature
SVM	Support Vector Machines
ESP	Exome Sequencing Project
REVEL	Rare Exome Variant Ensemble Learner
ExAC	Exome Aggregation Consortium
GBM	gradient boosting machine
VEST	Variant Effect Scoring Tool
OMIM	Online Mendelian Inheritance in Man
TP	True Positive
FN	False Negative
FP	False Positives

TN	True Negative
MCC	Matthews correlation coefficient
ROC	Receiver Operating Characteristic
TPR	True Positive Rate
FPR	False Positive Rate
VCF	Variant Call Format
HGVS	Human Genome Variation Society
SNP	Single Nucleotide Polymorphism
ANNOVAR	Annotation of Genetic Variants
SV	Structural Variant
CNV	Copy Number Variation
INDELS	Insertions/Deletions
BED	Browser Extensible Data
UniProtKB	UniProt Knowledgebase
MF	Molecular Function
CC	Cellular Component
BP	Biological Process
GO	Gene Ontology
HMMs	Hidden Markov Models
TSV	Tab-Separated Values
EC	Enzyme Commission
CNN	Convolutional Neural Network
CASP	Critical Assessment of Structure Prediction
MGLTools	Molecular Graphics Laboratory Tools
PQR	Partial Charge Representation
MOL2	Tripos SYBYL Molecular 2
CRG	Centre for Genomic Regulation
WSGI	Web Server Gateway Interface
AMD	Age-related Macular Degeneration

CHAPTER 1

INTRODUCTION

1.1. Motivation

Recent improvements in genome sequencing technologies have drastically increased the detection of genetic variations across a variety of populations. These variations include single nucleotide variations (SNVs), insertions, deletions, and other structural alterations. The Genetic variants in coding regions that can affect protein function and structure lead to a wide range of effects from benign to fatally pathological.

In the field of genetics and genomics, classifying variations as benign or pathogenic is critical for a number of reasons. The growth of knowledge and research in genetics is facilitated by proper variation classification. It supports the development of new diagnostic and treatment strategies, the understanding of disease mechanisms, and the identification and analysis of variations linked to certain diseases. Prior knowledge of inherited disease-causing variants provides the chance for alternative family planning (Marwaha, Knowles, & Ashley, 2022). In this regard, disease diagnosis is an important issue and depends on detecting deleterious variants among the numerous discovered variants. However, it is difficult to estimate the clinical pathogenicity of variants as it requires evaluating their influence on the proteins involved and the body as a whole.

The American College of Medical Genetics and Genomics (ACMG) (Richards et al., 2015) has published guidelines to support a consistent classification system for variants. Overall, the twenty-eight specific criteria in the ACMG recommendations can be used to assess the quality of the evidence supporting variant classification. Numerous factors are covered by these criteria, including population frequency, computational estimations, functional studies, segregation analyses, and other supporting data (Figure 1). Based on these guidelines, the public database ClinVar(Landrum et al., 2018) classifies the variants as pathogenic (P), likely pathogenic (LP), uncertain significance (VUS), likely benign (LB) and benign (B). Strong clinical or functional evidence supports the disease-causing nature of the variations designated as P. Additionally, variations that have been previously classified as pathogenic in databases for particular diseases or in the literature are categorized as LP. In contrary, variants with strong evidence of being benign, such as a high population frequency or functional studies confirming no harmful effect and

benign polymorphisms with substantial population data are designated as B and LB, respectively. The remaining variants are labeled as VUS when there is insufficient or conflicting evidence to determine whether they are pathogenic or benign.

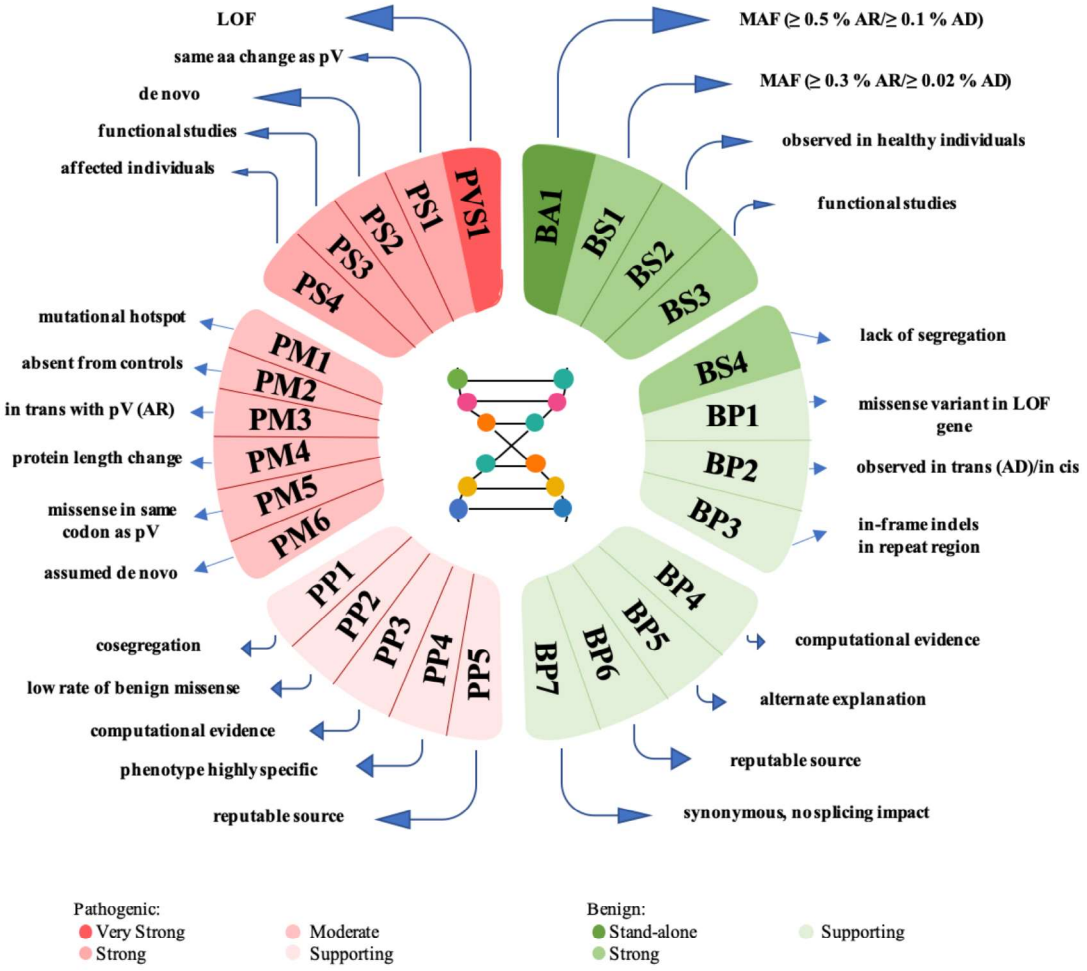

Figure 1: Overview of ACMG guidelines for classification of pathogenic and benign variants. The ACMG guidelines follow a structured process that involves the interpretation of the 28 rules. The rules are divided into 16 pathogenic and 14 benign criteria, providing a comprehensive framework for variant classification. In order to classify a variant, the ACMG guidelines require the interpretation of these rules and the assessment of all available evidence. The rules are evaluated in a structured manner, considering both the pathogenic and benign criteria. The collective weight of the evidence is then used to determine the final classification of the variant as pathogenic, likely pathogenic, uncertain significance, likely benign, or benign.

In addition to the ACMG guidelines, several databases provide valuable resources for variant classification. ClinVar, one such database, assigns confidence levels to variant classifications using a star-based system. The number of stars is proportional to the number of submitters and the reliability of the source, indicating the confidence level associated with each classification. This helps researchers and clinicians assess the robustness of variant classifications reported in ClinVar. Another significant resource

is the 1000 Genomes Project (TGP) (Auton et al., 2015), which serves as a comprehensive repository of human genetic variation. This project has gathered data from a diverse population of healthy individuals, enabling the identification and characterization of common genetic variations in the human genome. The information obtained from the TGP contributes to the understanding of normal genetic variation and serves as a reference for comparison when evaluating disease-associated variants. Furthermore, the Human Gene Mutation Database (HGMD) (Stenson et al., 2014) represents an extensive collection of mutations associated with human inherited diseases. HGMD serves as a valuable resource for researchers and clinicians, providing comprehensive information on disease-related mutations and their clinical significance. By utilizing HGMD, researchers can access a vast repository of knowledge and gain insights into the pathogenicity of specific genetic variants.

Together, ACMG guidelines, along with databases such as ClinVar, TGP and HGMD, play critical roles in variant classification. They provide essential resources, standardized guidelines, and comprehensive databases that enable researchers and clinicians to assess the pathogenicity of variants, contribute to the understanding of genetic variation, and enhance the accuracy of genetic diagnosis and personalized medicine.

1.2. Background

Despite the amount of data available, there is a significant gap between the number of well classified variants and the number of potential variants in the human genome. Furthermore, the rapid progress in high-throughput sequencing technologies and computational methods has expedited the detection of genetic variations, leading to a continuous expansion of genomic data. However, confirming a pathogenic variation and diagnosing an associated disease condition can take a long time and much effort, resulting in treatment delays and added cost for patients. As a result, the challenge of determining the functional impact of sequence alterations is still an open area of research, and efforts to construct more reliable predictor tool are ongoing.

Considering these limitations, various prediction methods have been developed to determine the level of pathogenicity of variations and assist clinical decision-making. These in silico prediction tools have become important resources for evaluating the possible effects of genetic variations on protein structure and function. They use algorithms to estimate the deleteriousness of variations based on a variety of variables, including conservation scores, physicochemical characteristics, and evolutionary data. These computational techniques attempt to categorize variations as benign or harmful by examining the mentioned characteristics, offering insightful information about the functional effects of genetic changes.

In the field of computational prediction, some attempts have been made using machine learning algorithms. REVEL (Ioannidis et al., 2016) employs a random forest model to combine many pathogenicity predictors in order to create an integrated variant

predictor. This method predicts the disease-causing effect of missense variants using eighteen features. SIFT (Sim et al., 2012) is another algorithm that uses statistical methods to determine the likelihood that a mutation at a specific sequence position can be tolerated. CADD (Combined Annotation-Dependent Depletion) (Rentzsch, Witten, Cooper, Shendure, & Kircher, 2019), applies linear regression to integrate different features and assign a single score for ranking genetic variants. The naïve Bayes-based method, POLYPHEN (Ramensky, Bork, & Sunyaev, 2002), uses a substitution matrix score for pathogenicity evaluation. The model is built to predict deleterious effects of missense mutations based on the combination of different features. The newer methods try to contribute to the field by applying sequence-based prediction. Methods that exploit protein sequences are more efficient as they solve the overfitting problem using sequence context. MutPred2 (Pejaver et al., 2020) is one of these sequence-based models and provides a functional score using the Gene Ontology.

Initially, the majority of known approaches for analyzing genetic variations focused solely on missense mutations; however, some groups have since improved these methods to include indels and nonsense alterations. SIFT Indel (Hu & Ng, 2013) is an extension of the SIFT algorithm, which deals with frameshift and non-frameshift. This classifier was trained on two datasets, taken from HGMD and interspecies variations, to distinguish damaging frameshift variants from indels with no harmful effect on gene function. The prediction results of this method are provided based on four features, namely the percentage of affected conserved DNA nucleotides, indel location, percentage of affected conserved amino acids and distance between indel and exon border. DDIG-in (Folkman et al., 2015) and VEST-indel (Douville et al., 2016) are the other methods that handle frameshift and nonsense mutations using an SVM-based model and Random Forest classifiers, respectively. The DDIG-in method employs eight features and actual human variants are used for training. VEST-indel uses twenty-four features including the “PubMed feature” and it was trained on variants taken from HGMD and ESP6500. The CADD algorithm predicts all possible types of genetic variations and covers most of the features in the reference assembly, using more than sixty features. This workflow is trained on over thirty million variants with a linear support vector machine (SVM) and takes the advantage of a considerably larger and less biased training dataset. The training dataset of neutral variants was derived from cross - species comparisons, where the deleterious dataset was compiled from simulated variations.

1.3. Summary of contribution

Sequence variant pathogenicity prediction approaches have advanced significantly over the last decade but they still have critical flaws and new approaches are required to detect misclassified deleterious variations. Among different classes of variants, frameshift and nonsense mutations account for a significant fraction of damaging mutations. Nonsense and frameshift variations, which impair a gene or protein's ability to function normally, are the root cause of numerous genetic disorders. Making appropriate therapeutic decisions and recognizing the biological basis of the disease

depend on accurate identifying of target variations. Nevertheless, the functional impact of frameshift indels and nonsense changes are less studied (in contrast to, e.g. missense variants ((Rogers et al., 2018),(Adzhubei, Jordan, & Sunyaev, 2013),(Dong et al., 2015)) and interpreting the effect of these variants to clinical phenotypes is not always straightforward. These variants often lead to a premature stop codon or a disruption of the reading frame, resulting in a truncated or non-functional protein. Since the causative effects of nonsense and frameshift variants are generally more severe than missense mutations, the features relating to functionality of proteins are more highlighted in predicting the impact of these variants. Furthermore, while a variant's association with a disease may be known, its molecular effect on protein function could still be unknown.

Therefore, we developed a sequence-based method, Pathogenic Mutation Prediction (PMPred), for predicting the pathological nature of variants in the human proteome based on analysis of gene ontologies and protein domains. Indeed, PMPred calculates the pathogenicity rating of frameshift indels and nonsense mutations in terms of the conserved domains and loss of protein function without specific training. PMPred is available as a standalone tool (https://github.com/kansil/PMPred) and a publicly available web server (https://pmpred.kansil.org) that allows users to submit their target variants in VCF format. The program automatically analyzes the effects of variants with respect to protein function and has comparable or higher accuracy in differentiating pathogenic and benign variants, when compared to the state-of-the-art. As a result, it generates an annotated list regarding the consequences of these variants on conserved domains and their pathogenicity rating. In addition, we have shown that PMPred outperforms current indel and nonsense prediction algorithms by creating a benchmark dataset of neutral and harmful variants on which none of the approaches had been trained.

1.4. Outline

This thesis defines a novel approach for identifying pathogenic mutations, presented across five informative chapters. The chapters collectively form a comprehensive framework for comprehending the rationale, methodologies, findings, and implications of the proposed approach in the field of pathogenicity prediction for genetic variants.

The concerns of intended research area are briefly described in the "Introduction" chapter, along with the study's objective, which may be able to address the corresponding issues. The "Background and Literature Review" chapter goes into detail of different variation types and their potential disease-causing impacts. Also, numerous techniques for finding harmful mutations are outlined in depth. In addition, it covers significant studies and discuss features that make them remarkable methods. The "Materials and Methods" section provides a detailed illustration of the developed method for pathogenicity prediction. The prerequisites for using the application, the workflow procedures, and the necessary input format for analysis are all thoroughly

detailed in this part. The "Results" chapter will then describe the outcome of our method and test the effectiveness of it using a test dataset. Also, the same examination is conducted with different methods to evaluate their performance and represent the comparison results. The efficacy of our approach is discussed in the chapter titled as "Discussion", which also reviews the impact of several parameters on the pathogenicity of variants. This section also outlines the extra options and regions that we plan to include in future versions. The "Conclusion" chapter summarizes the research objective and achievements, highlighting the effectiveness of the proposed approach, PMPred, in accurately predicting the disease-causing potential of genetic variants, and outlines future directions for enhancing pathogenicity prediction tools.

CHAPTER 2

BACKGROUND AND RELATED WORKS

Mutation is an alteration in the DNA sequence of a gene or genome. Mutations may happen spontaneously or as a result of mistakes made during DNA replication or response to the external factors like radiation or toxins. A mutation can affect an organism in a variety of ways. While some alterations can be detrimental or even fatal, others may have little or no impact at all. However, under specific circumstances, some mutations can also be advantageous, resulting in evolution and the creation of novel features (Figure 2) (Loewe & Hill, 2010).

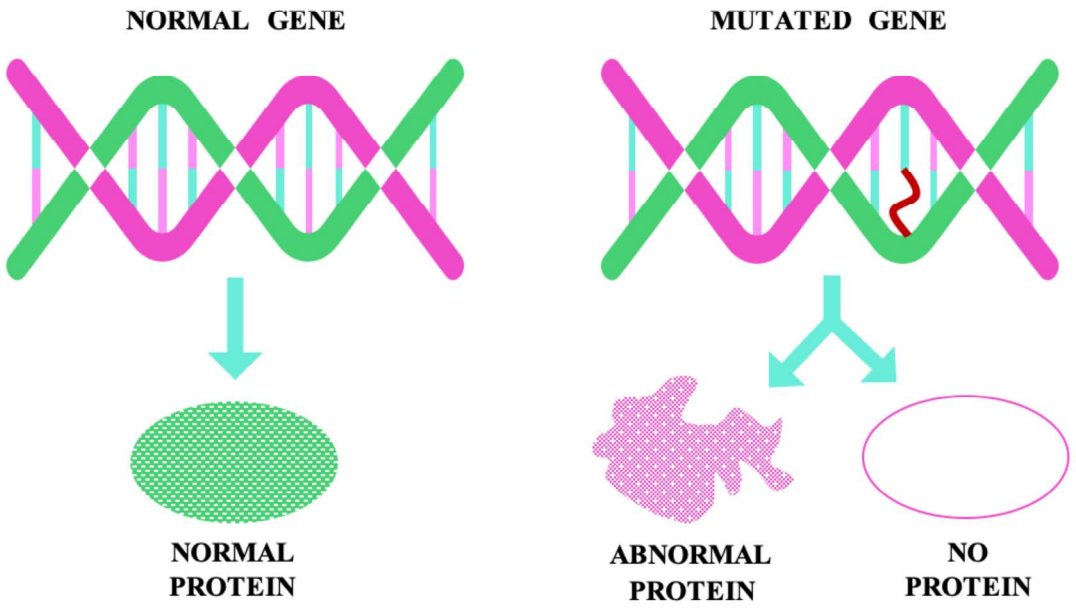

Figure 2: Comparison of Protein Products: Normal vs. Mutated Gene. The figure illustrates a comparison between the protein products of the normal gene and the mutated gene. The normal protein is represented by a green circle, symbolizing a functional and properly shaped protein. In contrast, the product of a mutated gene is depicted by an uncolored circle or an unshaped circle, indicating the absence of a protein product or the presence of an abnormal protein. This visual representation highlights the impact of the mutation on the DNA sequence, which can lead to disruptions in protein production, structure, and function.

Exons, non-coding regions and regulatory regions of DNA are among the parts of the genome where mutations can take place. The function of a protein may be affected by modifications to the amino acid sequence caused by mutations in coding regions. On the other hand, mutations that occur in regulatory or non-coding regions might affect how genes are expressed and have important consequences (Lee & Young, 2013). Mutations can be categorized as missense, nonsense, silent, or frameshift depending on the way they affect gene function. Missense mutations are changes in a single nucleotide that cause the protein to code for a different amino acid, where silent mutations are changes that do not affect the protein's amino acids. Nonsense mutations cause the generation of a shortened amino acid by encoding an early stop codon. Finally, frameshift mutations alter the reading frame of the codons by insertion or deletion of a nucleotide (Figure 3) (Gorlov, Kimmel, & Amos, 2006).

Genetic variations may have a variety of effects on the structure and function of proteins, ranging from deleterious to benign. Classifying a mutation as deleterious or benign is a challenging process that typically includes combining several lines of evidence, such as experimental results and bioinformatics estimations. Mode of inheritance is one of the factors that is usually considered when figuring out how harmful a mutation is. X-linked or dominantly inherited mutations are more likely to be disease-causing than recessive or mutations with an ambiguous mode of inheritance. Population frequency is another important issue that should be considered before labeling a variant as pathogenic. Generally rare mutations have a greater probability of becoming harmful than widespread mutations in the population. For a variant classification, co-segregation with disease should also be considered since mutations that do so within families or communities are more likely to be pathogenic. Functional impact is another consideration since mutations with strong effect on protein function are more likely to be deleterious. Also, conservation is a main element since mutations that alter the protein's or its binding partners' evolutionarily conserved areas are willing to be more dangerous. Another key component in defining a variant's pathogenicity is clinical databases and literary material that links mutations to diseases (Figure 4). Beyond these, the effect of mutations on protein function can be predicted computationally using bioinformatics tools (Liu, Yeung, Chiu, & Cao, 2022). It is more likely that a mutation is pathogenic if it is projected to be harmful by several methods.

The in-silico prediction methods measure the probability of variants' deleteriousness to increase the system's sensitivity because revealing a disease-causing mutation can have long-lasting effects on an individual and their family. Despite the efficiency of these techniques, they should not be utilized alone to categorize a variant. In fact, these tools are a component of a more sophisticated classification system and variants must go through the entire procedure, before being labeled as pathogenic or benign. In the field of computational prediction, methods can be classified based on the approach used to evaluate the variations. Although methods are not necessarily restricted to one technique, they can mainly depend on sequence-based models, structural parameters or hybrid models (Katsonis, Wilhelm, Williams, & Lichtarge, 2022).

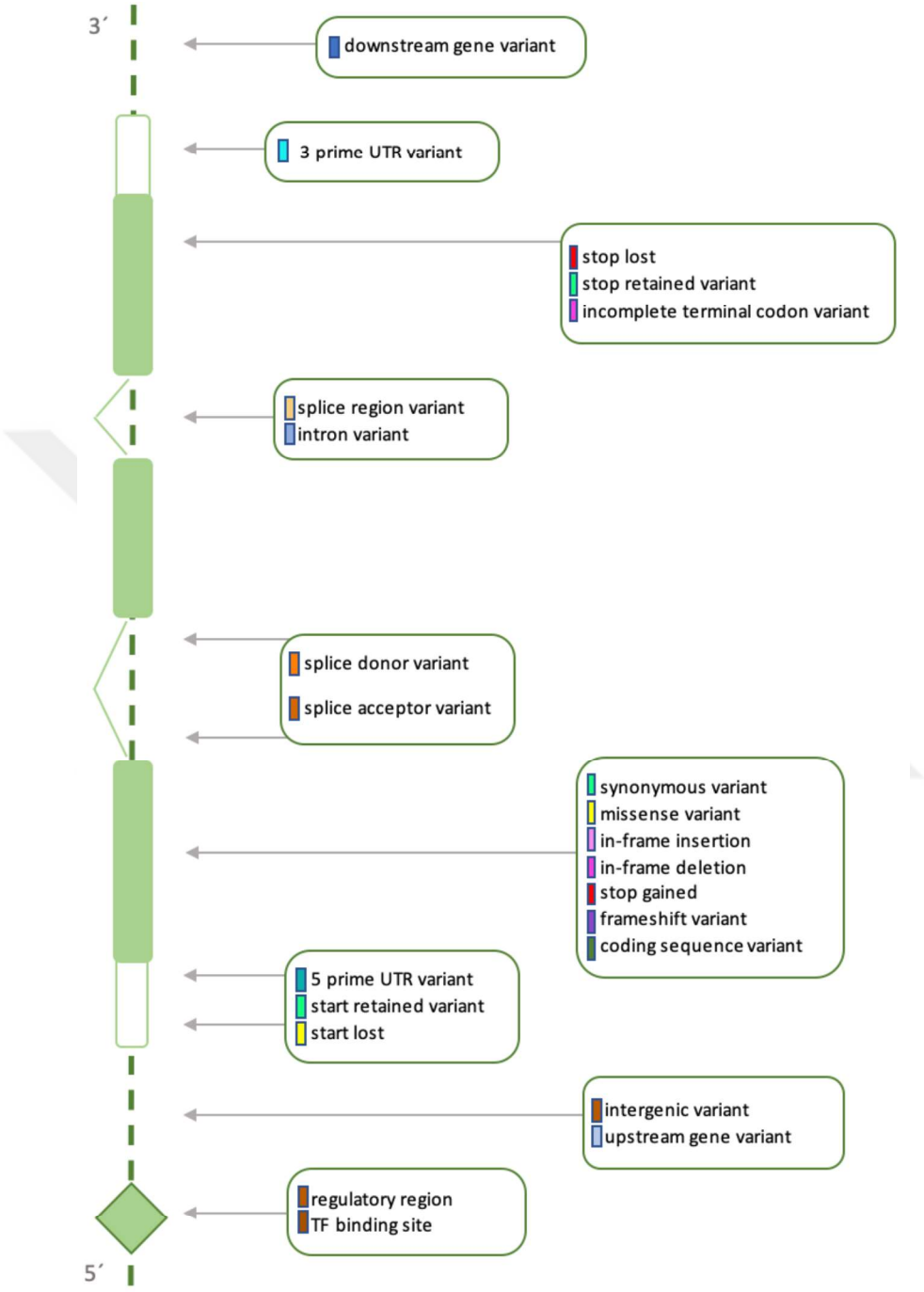

Figure 3: Transcript Structure and Impactful Mutation Regions. This figure platforms the structure of a transcript, highlighting distinct regions. Key regions are labeled to provide context to the transcript structure. Furthermore, the diagram illustrates the locations of various impactful mutations within the transcript. These high-impact regions encompass splice sites, frameshift mutations, stop gain, stop loss, and start loss mutations. Understanding the precise positions of these mutations relative to the transcript structure is crucial in comprehending their potential consequences on gene expression and protein functionality.

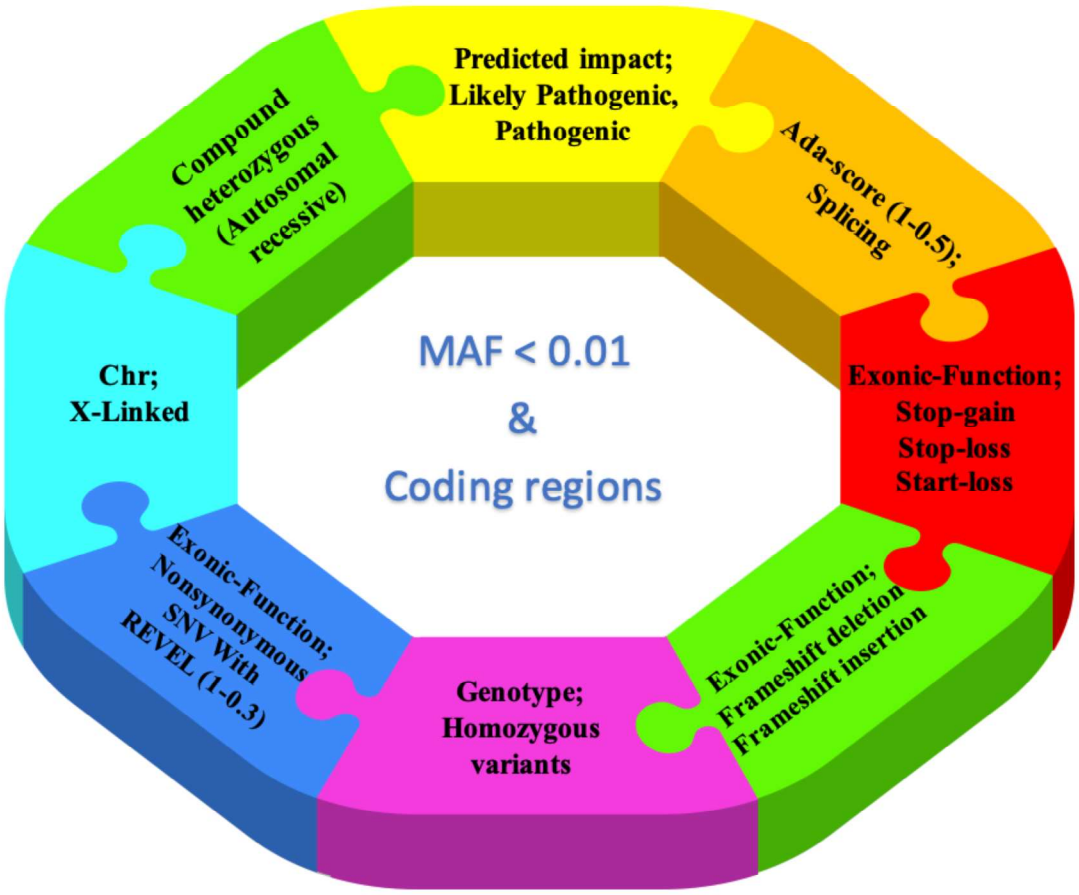

Figure 4: Guidelines for Variant Analysis in Next Generation Sequencing (NGS). It serves as a guideline, highlighting the critical steps and variant categories to be considered during the NGS variant analysis process. The analysis procedure begins with filtering variants based on their Minor Allele Frequency (MAF), selecting variants with MAF below 0.01. Next, the focus shifts to variants located in coding regions, exonic-splicing regions, and splicing sites. After this initial filtering step, the analysis procedure focuses on a specific subset of variants for further investigation. First, variants with pathogenic or likely pathogenic impact are given priority for analysis, as they have known associations with diseases. Additionally, splicing variants with high Ada scores (ranging from 0.5 to 1.0) are examined further to evaluate their potential impact on splicing. Variants resulting in stop gain, stop loss, and start loss consequences are investigated due to their potential effects on protein function. Frameshift deletions and insertions, which cause a shift in the reading frame, are also considered important in the analysis. Homozygous genotypes in the patient, particularly relevant for recessive inheritance patterns, are examined. For male patients, X-linked genes are specifically analyzed. Compound heterozygous variants, where the patient carries two different variants in the same gene, are explored as well. Lastly, nonsynonymous variants with high REVEL values (ranging from 1 to 0.3) are investigated to identify variants that are more likely to cause functional changes in the protein.

2.1. In silico prediction methods

2.1.1. Sequence-based prediction methods

2.1.1.1. Sift

Sorting Intolerant From Tolerant (SIFT) is a widely used method for determining the possible effects of missense mutations on protein structure and function. The approach is exclusively designed for missense mutations and other sorts of mutations, such as frameshift mutations, nonsense mutations, or splice site mutations, which can have various impacts on protein function, are not meant to be predicted by this method. The main principle of SIFT is to find conserved and variable residues by comparing the amino acid sequence of the query protein to a database of homologous protein sequences. The idea is that conserved residues are more likely to be functionally significant, whereas variable residues may be more adaptive to changes in amino acids. After prediction, missense mutations can be categorized as "tolerated" or "deleterious" depending on how much the altered amino acid residue varies among species. The assigned scores for variants, scales from 0 to 1, where the lower values denoting a greater possibility of a functional impact. A SIFT score of less than 0.05 is the standard cutoff for designating a variant as "damaging", albeit this cutoff can be changed depending on the particular application.

2.1.1.2. Provean

The Protein Variation Effect Analyzer (PROVEAN) (Choi & Chan, 2015) is a tool for calculating the effect of amino acid substitutions on proteins. PROVEAN primarily employs two models to predict the effects of amino acid alterations. Pairwise alignment-based model is the first technique that involves comparing the protein's amino acid sequence to similar sequences in other species using pre-calculated position-specific scoring matrix (PSSM). Hidden Markov Model-based model (HMM) is the second technique to predict the impact of a substitution based on the position and characteristics of the substituted amino acid. In fact, this statistical technique considers the correlations between nearby amino acids in the protein sequence to assess the likelihood of a variant for being harmful. PROVEAN integrates the two models to produce a more detailed estimation of the functional impact of missense mutations. This prediction tool was built exclusively for missense mutations and classifies them as neutral or deleterious. The method’s prediction rating scale spans from -10 to +10, with positive values denoting possible neutral mutations and negative ones denoting possibly disruptive mutations.

2.1.1.3. MutPred

MutPred considers a wide range of factors, such as protein stability, evolutionary conservation, residue properties, structural context, and splice site disruption to produce a score that quantifies the risk that a certain mutation will be harmful. The tool additionally incorporates a number of databases to provide further annotation and functional properties on the protein and associated mutations. The capability of

MutPred to precisely predict the molecular mechanism through which a mutation may result in disease is one of its distinctive features. Understanding the molecular mechanism of genetic diseases can be facilitated by this information, which will consequently assist in the creation of individualized treatments. The main purpose of the model is to predict the functional impact of missense mutations, while it can also estimate the effects of nonsense and frameshift variations. A mutation's likelihood of being destructive is shown by the MutPred prediction scores, which range from 0 to 1. Variants with scores close to 0 are probably harmless, while mutations with scores close to 1 are either pathogenic or likely to be pathogenic.

2.1.1.4. PolyPhen

The sequence-based tool PolyPhen is used to evaluate the probable functional impact of an amino acid alteration in a protein. In addition, PolyPhen-2 (Adzhubei et al., 2013), a new version of PolyPhen, makes use of both sequence-based and structure-based aspects to increase sensitivity and specificity. In order to estimate the probable functional impact of a variant, the model employs a naïve Bayes classifier to incorporate data from several sequence-based features, such as conservation scores, amino acid physicochemical properties, and structural annotations. Additionally, statistical significance testing is used to evaluate the prediction's accuracy. As a result, the PolyPhen-2 score, which measures the likelihood that a missense mutation would be deleterious, and the PolyPhen-2 HVAR score, which measures the degree of confidence in the prediction, are both calculated for each missense mutation. With a score ranging from 0 to 1, it can therefore assess the potential consequences of missense mutations. However, some forms of mutations, such as insertions, deletions, or non-coding variations, cannot be predicted by PolyPhen since they do not alter the amino acid sequence of the protein.

2.1.1.5. MutationTaster

MutationTaster (Schwarz, Rödelsperger, Schuelke, & Seelow, 2010) is a popular bioinformatics tool employed to predict the disease-causing potential of genetic mutations. By harnessing a blend of computational algorithms and databases, MutationTaster evaluates the potential functional implications of a genetic variant. Its primary objective is to ascertain whether a specific DNA alteration is prone to affect protein function, potentially resulting in the manifestation of a disease phenotype.

The method employs a multi-step prediction process to assess the impact of genetic variants. Initially, it analyzes the variant's genomic context, taking into account factors like its location within a gene and the presence of regulatory elements. Subsequently, it investigates the conservation of the affected region across diverse species, considering the evolutionary importance of the DNA sequence. Additionally, MutationTaster examines potential modifications in protein characteristics, encompassing structural domains, functional motifs, and sites susceptible to post-translational modifications. It also utilizes existing databases that store information about mutations associated with diseases and their corresponding phenotypic effects.

Through a comparative analysis between the input variant and the entries in these databases, MutationTaster extrapolates the potential pathogenicity of the variant by drawing parallels to previously reported cases with similar characteristics.

MutationTaster utilizes a Bayes classifier to predict the disease potential of genetic alterations. The tool offers three distinct models, each tailored to different types of alterations. The "without_aae" model focuses on silent alterations, including non-synonymous and intronic changes. The "simple_aae" model is designed for alterations that result in the substitution, insertion, or deletion of a single amino acid. Lastly, the "complex_aae" model addresses more intricate alterations, such as mutations introducing premature stop codons or complex changes in the amino acid sequence. All models were trained using a comprehensive dataset of common polymorphisms and disease mutations, and MutationTaster automatically selects the appropriate model for each alteration. The results generated by the tool include a prediction score to classify each variant under analysis. The classification labels the variants as either "disease-causing" or "polymorphism," indicating their potential influence on protein function. Furthermore, MutationTaster supplies supplementary details, such as conservation scores, functional annotations, and references to pertinent literature, which contribute to a comprehensive understanding of the analyzed variants.

2.1.1.6. DDIG-in

An innovative machine learning approach named DDIG-in (Disease-Defining Genomic Information) utilizes a comprehensive dataset of human genetic variations obtained from two distinct sources: The HGMD (containing disease-causing variants) and the TGP (encompassing putatively neutral variants).

Frameshifting (FS) indels and nonsense (NS) variants are types of genetic alterations that significantly impact the protein-coding sequence. FS indels introduce shifts in the reading frame, while NS variants introduce premature termination codons. Differentiating between disease-causing and neutral FS indels and NS variants remains an area of limited understanding as these disruptive mutations have been observed in individuals without any apparent disease phenotype. This method incorporated a comprehensive analysis of thirty six distinct features to distinguish between disease-causing FS indels, neutral FS indels, and NS variants. In order to identify the most effective combination of features, the researchers employed the sequential forward floating selection (SFFS) algorithm. The selected feature set was then utilized to train a robust SVM model, which forms the basis of their classification approach.

2.1.1.7. GERP++

Genomic Evolutionary Rate Profiling (GERP) (Cooper et al., 2005) is a technique that estimates evolutionary constraint at specific positions in the genome by employing maximum likelihood evolutionary rate estimation. Additionally, it identifies "constrained elements" that involve multiple positions exhibiting a collective signal suggesting the presence of a potential functional element. The focus of this method is

on providing position-specific scores, rather than making explicit predictions about the functional elements themselves.

GERP++ (Davydov et al., 2010) operates on the fundamental concept that functional elements within the genome experience selective pressure, leading to their preservation across evolutionary time. Through comprehensive analysis of sequence conservation patterns across diverse species, GERP++ effectively detects genomic regions that are more likely to possess functional significance. This system employs a phylogenetic approach that incorporates the evolutionary relationships and divergence timelines among various species. It compares the observed sequence conservation patterns to the expected outcomes under neutral conditions, while also considering factors such as mutation rates and baseline conservation levels. This comprehensive analysis enables it to discern regions of the genome that deviate from neutral expectations and likely harbor functional elements.

The output is a conservation score for each position within the genomic region under analysis. These scores serve as indicators of the level of evolutionary constraint, with higher values suggesting greater conservation and potential functional significance. Researchers can utilize these scores to prioritize regions for additional experimental exploration or to examine the potential effects of genetic variants. In summary, GERP++ offers valuable insights into the evolutionary conservation within the genome, assisting in the identification of critical functional elements.

2.1.2. Structure-based prediction methods

2.1.2.1. mCSM

Mutation Cutoff Scanning Matrix (mCSM) (Pires, Ascher, & Blundell, 2014) scans the protein structure and predicts the impact of mutations using a matrix-based methodology. This approach measures the effect of mutations on protein function by combining graph-based fingerprints of protein structures with Support Vector Regression (SVR) (Awad & Khanna, 2015) and Random Forest (RF) (Breiman, 2001) machine learning methods. The major focus of the tool is to evaluate the pathogenicity of missense mutations that potentially alter protein stability or protein-protein interactions. It has been shown that the methodology is useful for predicting the effects of single point alterations or multiple mutations on protein. Since its first release, mCSM has undergone updates to incorporate deep learning and ensemble approaches, which offers a variety of capabilities, including the ability to predict protein-ligand and protein-DNA interactions. The supplied score ranges between -3 and 3, with positive and negative values denoting stable mutations and harmful mutations, respectively.

2.1.2.2. FoldX

FoldX (Alford et al., 2017) is a structure-based in silico prediction tool that calculates the effects of mutations on protein stability, interactions, and function by utilizing the protein's three-dimensional structure. In fact, the model relies on physical and

empirical force fields to assess the free energy differences between wild-type and mutant form of the target protein. This tool analyzes a mutation in a protein with a known structure to determine scores relating to the mutation and numerous energy terms, including total energy and van der Waals energy. For each variant, FoldX generates a free energy change (ΔΔG) value, with larger negative values suggesting a greater risk of protein structural destabilization and related functional implications. The threshold of variant classification is flexible and can be altered depending on the usage and context. At the end, the result interpretation can be used to determine how the protein's stability and functionality will be impacted by the mutation.

2.1.2.3. SDM

Site-Directed Mutator (SDM) (Worth, Preissner, & Blundell, 2011) is a tool for determining how amino acid alterations affect the stability of proteins. The changes in protein stability resulting from amino acid substitutions are computed by SDM using a combination of practical and experience-based energy equations. It calculates the energy difference between proteins that are wild-type and mutant using a two-step thermodynamic cycle method. Missense mutations that impact protein stability can be predicted by SDM. Nonetheless, it might be possible to produce predictions for insertions, deletions, or nonsense variations if they take place in a region of the protein that involves in the structure formation used by method's prediction. Additionally, the technique can be used to predict how mutations will affect protein-protein interactions and protein-ligand binding.

2.1.2.4. I-Mutant Suite

I-Mutant Suite (Capriotti, Fariselli, & Casadio, 2005) is a tool designed to computationally forecast the influence of amino acid substitutions on the stability of proteins. Its main objective is to investigate the impact of individual mutations on protein stability, providing insights into the relationship between protein structure and function. Additionally, it enables the evaluation of the potential pathogenicity of variants by analyzing their effects on protein stability.

The I-Mutant Suite comprises several distinct modules, each tailored to deliver specific predictions pertaining to protein stability. These modules encompass I-Mutant 2.0, I-Mutant 3.0, I-Mutant Suite 4.0, and I-Mutant Suite 5.0. I-Mutant 2.0 employs a SVM approach to estimate the thermodynamic stability change (ΔΔG) resulting from a single amino acid substitution. By considering the protein sequence and mutation details as input, it generates predictions regarding whether the mutation will enhance or diminish protein stability. I-Mutant 3.0 expands upon the capabilities of I-Mutant 2.0 by incorporating additional features such as solvent accessibility and secondary structure information. Through an ensemble of SVM classifiers, it enhances the precision of stability predictions by estimating ΔΔG values more accurately. The updated versions, I-Mutant Suite 4.0 and I-Mutant Suite 5.0 offer advanced capabilities for predicting the impact of amino acid substitutions on protein stability. I-Mutant Suite 4.0, incorporating advanced machine learning techniques like SVR and

RF algorithms enhances the prediction accuracy for both ΔΔG values and changes in melting temperature (Tm) caused by mutations. Also, I-Mutant Suite 5.0 delivers improved predictions of mutation effects on protein stability by employing a combination of deep learning and machine learning algorithms.

The typical outcome generated by this method is the prediction of the ΔΔG value or the alteration in melting temperature caused by a specific mutation. A positive ΔΔG value suggests destabilization, whereas a negative value indicates stabilization. It is possible to tailor the specific thresholds used to classify mutations as either stabilizing or destabilizing based on the specific preferences or needs of the user.

2.1.2.5. MuPro

MuPro (Cheng, Randall, & Baldi, 2006) is a computational solution created to assess the influence of amino acid substitutions on the stability of proteins. Its main objective is to explore the impact of mutations on protein folding and stability, allowing for a comprehensive analysis of variant pathogenicity and the intricate relationship between protein structure and function. By accurately predicting the effect of mutations on protein stability, MuPro plays a crucial role in the identification of potentially harmful variants, contributing to a deeper understanding of their functional implications.

This approach employs a machine learning methodology to forecast alterations in protein stability resulting from amino acid substitutions. It takes into account a range of mutation-related features, including the physical and chemical characteristics of both the original and mutated amino acids as input of prediction model. The prediction model has been trained on a dataset comprising experimentally validated information on changes in protein stability. As a result, MuPro generates scores regarding the impact of a mutation on protein stability and expresses how the mutation is likely to affect the protein's stability. In this prediction, the scores greater than zero indicate mutations that have a disrupting effect, whereas scores below zero indicate mutations that have a stabilizing effect. The tool offers a user-friendly interface and allows users to input protein sequences in the commonly used FASTA format. By integrating a range of algorithms and techniques, such as SVM, the tool achieves precise and dependable predictions.

2.1.3. Hybrid-based prediction methods

2.1.3.1. CADD

CADD is a hybrid prediction method as it implements over sixty features such as evolutionary conservation and protein annotations to predict the pathogenicity of variants. This machine learning based method is trained with a list of variants with known pathogenic impact to discriminate between harmful and benign variants. In order to create this dataset, SNVs from the NHLBI Exome Sequencing Project (ESP) and the TGP were extracted and functional information from sources like ENCODE (Luo et al., 2020) was used to annotate them. Here the core methodology is using the train set to learn the relationship between different features and deleteriousness to

predict the pathogenicity of novel variants by computing a CADD score. Raw scores and C-scores are the two sorts of results offered by CADD. The raw scores are on a continuous scale for reflecting the severity of an alteration, where the C-scores provide the scaled and normalized scores of all variants in the human genome. As part of the score normalization procedure, the distribution of raw scores for each potential single nucleotide variation in the human genome is fitted to a logistic regression model. In the context of a normalized metric, the C-score denotes a Phred-like assessment and varies between 1 to 99 with higher scores suggesting that the mutation is more likely to be pathogenic. The C-score in CADD has a proposed threshold between 10 to 20, which indicates that variations with score above it are anticipated to have disease-causing effect. It is crucial to know that the chosen cutoff may vary based on the type of analysis and case in which CADD is being applied. This tool is capable of predicting the impact of various types of mutations including small insertions, small deletions, and missense changes. Apart from that, an exceptional feature of the technique is its ability to forecast the impact of variations in non-coding regions such introns, promoters, and enhancers.

2.1.3.2. Revel

A machine learning-based approach called REVEL (Rare Exome Variant Ensemble Learner) integrates numerous genomic data to calculate the effects of target variants on protein function. Over 20,000 human missense mutations from multiple sources, including the HGMD and the Exome Aggregation Consortium (ExAC) (Karczewski et al., 2017), made up the majority of the dataset used to train REVEL. On the basis of their known association with human disease or their prevalence in the community, these variations were primarily classified as harmful or benign. Consequently, after splitting the dataset into training and validation sets, a gradient boosting machine (GBM) (Friedman, 2001) learning technique was employed to train the algorithm and maximize the separation between the two classes. The accuracy of the prediction made by the final model was then evaluated using a different test set of mutations. This technique can estimate the consequences of splice site alterations, small indels and SNVs. The score generated by REVEL is a number between 0 and 1, and the recommended threshold for differentiating between harmful and neutral effects can vary depending on the gene, disease, population, and other factors. In general, a higher score indicates a greater likelihood that a variant would result in disease, whereas a lower score indicates a greater likelihood that the variant will be benign.

2.1.3.3. Vest

Variant Effect Scoring Tool (VEST) (Carter, Douville, Stenson, Cooper, & Karchin, 2013) is a prediction tool that employs a random forest algorithm to evaluate a variety of criteria, such as evolutionary conservation and protein domain annotations, in order to estimate the functional impact of missense variations in human protein-coding areas. To train the model, approximately 25,000 single nucleotide polymorphisms, (SNPs) are included, of which 2500 variants acquired from databases like HGMD and OMIM (Online Mendelian Inheritance in Man) (Amberger, Bocchini, Schiettecatte,

Scott, & Hamosh, 2015) as pathogenic while the rest of variants obtained from TGP and considered as benign. The Random Forest technique generates the final prediction by merging the predictions from numerous decision trees. To avoid overfitting and maximize model accuracy, the algorithm chooses a random subset of characteristics to use for each decision tree during training. The main goal of VEST is to estimate the effects of missense variations. On the other hand, VEST-indel, an extension of VEST, is developed to predict the functional impact of indels in protein-coding regions. In this method pathogenicity is represented by a number ranging from 0 to 1, with no fixed cutoff between harmful and benign.

2.1.3.4. Meta-SNP

A meta-predictor technique called Meta-SNP (Capriotti, Altman, & Bromberg, 2013) is used to forecast how single nucleotide polymorphisms (SNPs) may affect the stability of proteins. It uses machine learning to aggregate the outcomes of four distinct prediction tools, SNAP (Bromberg & Rost, 2007), PhD-SNP (Capriotti & Fariselli, 2017), PANTHER (Tang & Thomas, 2016) and SIFT. Using random forests, the approach combines the individual scores and provides an estimation of how an SNP will impact protein stability. This method does not produce its own predictions. Instead, it makes use of predictions made by other variation effect prediction tools, which may have been trained using a variety of datasets and techniques. This method is established to calculate the impact of single nucleotide variants (SNVs), but it is unable to anticipate the effects of insertions, deletions, or nonsense variations. Meta-SNP utilizes the results of the aforementioned four predictors, combining them into an eight-component feature vector consisting of two sets, each containing four elements. The prediction result is presented as a binary classification, where a score of 1 indicates a disruptive mutation and a score of 0 proposes a neutral mutation.

2.1.3.5. FATHMM

A hidden Markov model-based tool called FATHMM (Rogers et al., 2018) incorporates data from various sources to evaluate the functional effects of missense variations and to calculate their potential pathogenicity. Inter-species sequence alignment is one of the sources the model uses to examine the mutated amino acid sequence across a number of species and identify conserved and divergent residues. Apart from that, the model makes use of evolutionary conservation data and also considers the anticipated or known structure of the variant-affected protein. Mutations that disrupt the protein structure or occur in conserved areas are more likely to be harmful. The other feature taken into consideration by this model is disease association. The likelihood of a variation being pathogenic is increased when a connection between it and a disease is known. The algorithm then creates a score, which is a continuous measure of the probability that the mutation is deleterious, by integrating the findings from different sources. Variants with higher FATHMM scores have a higher likelihood of being harmful, whereas those with lower scores have a higher probability of being benign.

2.1.3.6. MetaLR

MetaLR (Dong et al., 2015) is a computational prediction approach that combines diverse genomic and functional characteristics to generate an assessment of the potential pathogenicity of a variant. In MetaLR, the prediction procedure entails the utilization of a machine learning model that is trained using an extensive collection of annotated variants. This training dataset comprises variants with known pathogenicity status, accompanied by a range of features including conservation scores, functional annotations, and variant annotations. This method utilizes a meta-learning framework to integrate predictions from several individual classifiers, each trained on distinct subsets of the training dataset. This ensemble strategy enhances the overall prediction accuracy by capturing a wide range of patterns and relationships present in the data. By leveraging diverse classifiers, MetaLR optimizes its ability to make accurate predictions. When making predictions, it considers various features of a specific variant, such as genomic position and functional annotations. These features are utilized to compute a pathogenicity score through the application of the trained machine learning model. The resulting pathogenicity score serves as an indicator of a variant potential for being either pathogenic or benign. The output is typically in the form of a probability or score, which reflects the predicted pathogenicity of the variant. A higher score implies an increased probability of the variant being pathogenic, whereas a lower score suggests a higher probability of it being benign. The determination of the threshold for classifying variants as pathogenic or benign can vary, influenced by factors such as the specific application or the user's personal preference.

It is crucial to notice that a variety of elements, including the particular dataset utilized, the kind of variations being examined, and the threshold selected for classifying variants, could all have an impact on how well the prediction tools function. In fact, the specific needs of the analysis and the characteristics of the variations under study determine which technique is more effective. For an in-depth understanding of variant pathogenicity, it is typically advised to compare the predictions made by different techniques. To get more accurate conclusions, it's also crucial to take into account additional evidence sources, such as functional research and experimental validation.

2.2. Performance evaluation metrics

Metrics for performance assessment are measurements that are used to gauge how well a classifier or prediction model is performing. Researchers can examine a variety of models and algorithms using performance evaluation criteria and decide which one fits best for their particular application. In order to improve the results, these measures are also utilized to identify the regions of concern and refine the parameters of the model (Hicks et al., 2022).

2.2.1. Common performance evaluation metrics for genetic variant prediction tools

2.2.1.1. Sensitivity/Recall

In binary classification, sensitivity is a performance evaluation parameter that quantifies the fraction of real positive cases that a model properly classifies as positive. In fact, sensitivity assesses how likely a model is to correctly pick out the positive instances from among all the true positives (TP). As a result, a model will have a high sensitivity if it accurately identifies the existence of a condition, producing a high proportion of TP and a low proportion of false negatives (FN). When the consequences of not treating the disease are severe or the medication is incredibly beneficial with few adverse effects, high sensitivity is crucial. The range of sensitivity is 0 to 1, with 1 indicating that the model successfully recognizes all positive cases and 0 suggesting that it fails to do so.

$$sensitivity = \frac{True\ Positive}{(True\ Positive + False\ Negative)}$$

2.2.1.2. Specificity

A performance evaluation indicator known as specificity quantifies the percentage of real negative cases that are accurately classified as such by a prediction model. A model will have a high specificity if it can accurately filter out cases which does not satisfy the condition, producing a high proportion of true negatives (TN) and a low proportion of false positives (FP). Specificity is a value between 0 and 1, with specificity of 1 denoting the accurate classification of all TN and specificity of 0 denoting the misclassification of all TN. In general, an improved performance of a model in accurately recognizing negative cases is indicated by a higher specificity value. However, the specificity value can also be impacted by the classification threshold that is selected by the model.

$$specificity = \frac{True\ Negative}{(True\ Negative + False\ Positive)}$$

2.2.1.3. Precision

A performance evaluation metric precision is the percentage of correctly anticipated positive cases among all positively predicted cases. In order to clarify, precision determines the accuracy of positive predictions by taking into account both the total number of positive forecasts made as well as the percentage of those predictions that came true. While a low precision score suggests that the model is producing many FP predictions, a high precision score shows that the model is producing fewer FP predictions. Precision is measured on a scale of 0 to 1, where 0 denotes no correct positive predictions and 1 denotes that all positive predictions are accurate. Additionally, in order to provide a more detailed evaluation of the model's performance, precision is typically utilized in conjunction with recall.

$$precision = \frac{True\ Positive}{(True\ Positive + False\ Positive)}$$

2.2.1.4. F1 score

The performance assessment statistics that combine recall and precision to yield a single score are known as the F-score. Two of the most popular F-scores are the F1 and F-beta scores. When the number of TP and FP are equally relevant, the F1 score is a useful indicator. Furthermore, this measurement score plays a critical role in situations when binary classification mode is used and the objective is to label cases as either positive or negative. On the other hand, when precision or recall is more crucial in a certain context, F-beta score, which employs a beta value to consider them more highly, may be utilized. The F1 score can have a maximum value of 1, which denotes perfect precision and recall, and a minimum value of 0, which occurs when either precision or recall are zero.

$$F1\ score = 2\frac{Precision\ .\ Recall}{Precision + Recall}$$

2.2.1.5. Matthews correlation coefficient (MCC)

A performance evaluation statistic known as the Matthews correlation coefficient (MCC) (Matthews, 1975) is used to assess how well binary classification predictions are made. In this evaluation, the TP, FP, TN, and FN are all taken into consideration. A single score is then generated that goes from -1 to 1. In this evaluation, perfect classification results in an MCC score of 1, while random classification results in a score of 0, and a full confliction between estimation and observation results in a value of -1. The MCC is typically regarded as a balanced indicator as it performs effectively when the sample size is unbalanced and the positive and negative classes are not equal. In these situations, MCC is recommended over accuracy, which can be deceptive due to its poor performance on the minority class.

$$\mathrm{MCC} = \frac{(\mathrm{TP} \times \mathrm{TN}) - (\mathrm{FP} \times \mathrm{FN})}{\sqrt{\big((\mathrm{TP} + \mathrm{FP}) \times (\mathrm{TP} + \mathrm{FN}) \times (\mathrm{TN} + \mathrm{FP}) \times (\mathrm{TN} + \mathrm{FN})\big)}}$$

2.2.1.6. Receiver operating characteristic curve (ROC curve)

The Receiver Operating Characteristic (ROC) curve (Tan, 2009) is a frequently employed statistic to measure the overall efficacy of the model. This metric provides a graphical representation of the approach, with the true positive rate

(sensitivity)(TPR) on the y-axis and the false positive rate (1-specificity) (FPR) on the x-axis. In this way, it shows the balance between TPR and FPR across all feasible threshold choices, and a successful classification model is one that achieves high TPR and low FPR values across all cutoffs. In this plot, the curve indicates perfect classification if it touches the upper left corner, while the diagonal line indicates random classification. Consequently, the model performs better, while the curve gets closer to the top left corner.

2.2.1.7. Area under the ROC curve (AUC-ROC)

The area under the ROC curve (AUC-ROC) (Melo, 2013), a performance evaluation criterion, gauges a binary classifier's performance in differentiating between positive and negative classes at all available thresholds. Calculating the area under the curve yields a value in the range of 0 and 1. The cutoff value of this evaluation metric is 0.5, which represents random categorization, and a classifier performs better as it approaches 1.

2.2.1.8. Confusion matrix

A confusion matrix (Ting, 2017) is a table of TP, FP, TN, and FN outcomes that are provided as four distinct possibilities in binary classification. This tabular evaluation metric locates the actual labels and predicted labels of a dataset on opposite sides of the table. As a consequence, it enables the visualization of an algorithm's performance by displaying examples of the predicted class in each row of the matrix and instances of the actual class in each column, or vice versa. The performance of a method cannot be assessed solely by related confusion matrix. However, through a comparison of the actual and predicted labels, this summarization method aids in determining the effectiveness of the classification model.

CHAPTER 3

MATERIALS AND METHODS

3.1. Materials

This section explains PMPred's analysis procedures and the acceptable input format for pathogenicity measurement. Additionally, this part covers the implemented tools in proposed method and provides a thorough explanation of input preparation.

3.1.1. Data description

The VCF (Variant Call Format) format is a valid input type for PMPred. The VCF is a widely utilized standard file format for collecting variant information acquired through DNA sequencing (Danecek et al., 2011). A comprehensive description of genetic variations, including SNVs, insertions, deletions, and other structural variations, can be covered in the VCF file. A typical VCF file has crucial elements and fields that should be considered in data preparation. Each file has lines with the '##' symbol at the beginning that stand for the metadata. Information regarding the reference genome used, the version of the software, and other important details about the variant calling process are all included in the metadata. Following the metadata section, the header line, marked with the '#' symbol, is presented to define the names and order of columns in the subsequent variant entries. The frequently utilized column titles in this row consist of chromosome (CHROM), position (POS), unique identifier (ID), reference allele (REF), alternative allele(s) (ALT), variation quality score (QUAL), variation filter status (FILTER), supplementary information (INFO), and genotype information format for each sample (FORMAT). Variant entries denote distinctive genetic variations along with relevant information. These tab-delimited entries provide specific information for each of the columns listed in the header. After the variant entries, genotype information for individual samples is arranged in tab-separated columns, aligning with the FORMAT field specified in the header. Subsequent to the variant entries, genotype information for each sample is presented in columns separated by tabs, based on the FORMAT field outlined in the header. This genotype information includes genotype calls, allele depth, genotype quality, and supplementary metrics specific to each sample. The VCF file format makes it possible to store and display a large number of samples along with their individual genotyping data for a particular set of variations. It functions as a comprehensive and reliable

format that makes it easier to share and analyze genetic variation data across many research and therapeutic applications. The fields and annotations contained inside the VCF file may differ based on the variant calling pipeline, the reference genome version, and additional annotations incorporated during the variant annotation process. The information in the relevant file serves as the basis for downstream variant analysis, which includes in silico pathogenicity prediction applying a variety of computational tools and techniques.

To perform pathogenicity prediction analysis, a subset of variations in VCF format is extracted from the freely accessible dataset offered by ClinVar. The publicly available database, ClinVar, gathers and curates data regarding genetic variants and their association with human health and disease. This database is maintained by the National Center for Biotechnology Information (NCBI) (Sayers et al., 2022), which operates under the U.S. National Library of Medicine. Various sources, such as clinical laboratories, researchers, and experts in the field all submit variant interpretations to ClinVar, which acts as a central repository for all of these submissions. With the use of this information, researchers, physicians, and patients will be better able to make decisions on the clinical importance of genetic variants. The database contains a diverse range of variants, encompassing SNVs, insertions, deletions, and structural alterations. Each variant in ClinVar is linked to pertinent clinical claims, such as those about pathogenicity, probable pathogenicity, benignity, or ambiguous significance (Sayers et al., 2023). These claims are substantiated by references to published literature, professional consensus, or other trustworthy sources, all of which rely on the currently available evidence.

Information about each variant is included in the INFO field, which encompasses various sections such as “AF_ESP”, “AF_EXAC”, “AF_TGP”, “ALLELEID”, “CLNDN”, “CLNDNINCL”, “CLNDISDB”, “CLNDISDBINCL”, “CLNHGVS”, “CLNREVSTAT”, “CLNSIG”, “CLNSIGCONF”, “CLNSIGINCL”, “CLNVC”, “CLNVCSO”, “CLNVI”, “DBVARID”, “GENEINFO”, “ORIGIN” and “RS”.

- AF_ESP: This field provides the allele frequency of the variant in the ESP dataset. It displays the percentage of alleles that were found in the ESP population study.

- AF_EXAC: The frequency of the variant alleles identified in a large cohort of exome sequencing data is represented by the AF_EXAC parameter, which stands for allele frequency of the variant in the ExAC dataset.

- AF_TGP: Based on the information provided regarding the frequency of the variant alleles seen in various populations included in the 1000 Genomes Project, this field provides the allele frequency of the variant.

- ALLELEID: Each variant's unique allele ID is displayed in this item that can be used to refer to the variant and get more information from the ClinVar database.

- CLNDN: The disease that the variation is linked to is represented by CLNDN annotation. It lists the name(s) of the illness or condition as reported in the variant submission. This parameter can be used to pinpoint the clinical situation in which the variant is pertinent.

- CLNDNINCL: Any additional clinical disease or condition term that is provided for the variation is identified in the field CLNDNINCL, which may contain several disease names separated by commas.

- CLNDISDB: Disease databases or other resources that report the disease or condition linked to the variation are identified by this annotation. Users can find further details on the disease phenotype linked to the variant by using the database names or accession numbers included.

- CLNDISDBINCL: The field CLNDISDBINCL, which may a contain comma-separated list of different database names, offers further database names or IDs for the clinical condition of the variant.

- CLNHGVS: The Human Genome Variation Society (HGVS) (den Dunnen et al., 2016) nomenclature for the variant is contained in CLNHGVS field. According to the HGVS recommendations, it offers a standardized depiction of the genetic variant, enabling consistent and clear variant description.

- CLNREVSTAT: The field gives information about the variant's review status. It makes clear whether the variant has undergone clinical expert review and offers details on the review procedure. There are a variety of possible values for this field, including "criteria provided, multiple submitters, no conflicts," "criteria provided, single submitter," "reviewed by expert panel," and "no assertion criteria provided".

- CLNSIG: The clinical importance of each mutation and its level of pathogenicity are indicated by this annotation. The phrases P, LP, VUS, LB, B or other particular classifications may be included in the values assigned to this annotation.

- CLNSIGCONF: The level of confidence in the reported clinical significance for the variant is indicated by the field CLNSIGCONF. In fact, it represents the level of certainty in the reported association between the variant and the clinical condition.

- CLNSIGINC: The clinical relevance of the reported mutation is supported by names or identities of additional sources, which are included in this element.

- CLNVC: The variant type is specified by the CLNVC section. It describes the genetic alteration of the variation by one of the "single nucleotide variant", "insertion", "deletion", "indel" or other specific terms.

- CLNVCSO: The VCF and Sequence Ontology (SO) words are used in the CLNVCSO section to offer information about the functional consequences of the variant at the genomic level. In this section, the terms like missense, frameshift, and synonymous are used to describe the precise impact of the variant. This annotation aids in determining how the mutation might affect the structure and function of the protein.

- CLNVI: This section of variant information stores the identifiers of the variant obtained from other databases or sources. This field makes it possible to link and compile different pieces of information from various resources. It may contain variant identifiers or accession numbers from external databases, facilitating cross-referencing and integration of data.

- DBVARID: The related part contains the variant identifier from the Database of Genomic Variants (dbVar) (Church et al., 2010), which lists structural variations in the human genome.

- GENEINFO: The annotation supplies details regarding the gene(s) linked to the variant. It commonly includes the gene symbol(s) or name(s) and may also encompass additional gene-related information like the gene ID or genomic coordinates.

- ORIGIN: The ClinVar ORIGIN annotation provides information on the origin of the submitted variant to specify the organization, source data, or person in charge of sending the variant to this database. This data reveals the origins of the clinical interpretation and aids in tracing the history of the variant data.

- RS: The reference SNP identifier connected to the variant is indicated by the RS annotation. It refers to a particular variation that has been given a distinctive identity in the dbSNP database, which acts as a thorough database of genetic variations. The variant's allele frequencies in various populations, connections with diseases, and other pertinent annotations are all retrieved and referred to using the RS ID.

3.1.2. Annovar

ANNOVAR (Annotation of Genetic Variants) (Wang, Li, & Hakonarson, 2010) is a comprehensive widely used software tool in bioinformatics that facilitates the annotation and interpretation of genetic variants. By providing a command-line experience, ANNOVAR helps researchers and clinicians in assessing the functional implications of genetic variants detected via next-generation sequencing (NGS) technologies. To delve into genetic variant annotation realm, Dr. Kai Wang at the University of Pennsylvania designed ANNOVAR. The first release of ANNOVAR was in 2010 to integrate various databases and prediction algorithms for offering a versatile platform for researchers to interpret genetic variants in a systematic manner.

ANNOVAR enables the annotation of single nucleotide variants (SNVs), copy number variations (CNVs), insertions/deletions (indels), and structural variants (SVs). It provides a robust framework for variant annotation, encompassing coding and non-coding regions, as well as regulatory elements, empowering researchers to explore the functional significance of identified variants.

By using algorithms such as PolyPhen-2, SIFT, and MutationTaster, it provides useful estimations for variants. The so-called estimations offer understandings of the possible effect of amino acid substitutions on protein structure and function. Additionally, ANNOVAR integrates information from conservation scores, such as PhyloP (Siepel et al., 2005) and GERP++, to recognize evolutionarily conserved regions and list variants with probable functional importance. By focusing on human genetics, ANNOVAR excels in annotating genomic variants that are of particular relevance to disease understanding and research. It leverages databases like dbSNP (Sherry et al., 2001), TGP, and ESP to annotate variants, enabling researchers to identify known variants and their frequencies across diverse populations. This information is crucial for distinguishing between rare pathogenic variants and common benign variants.

Together with gene promoters, enhancers, and non-coding RNA genes, ANNOVAR distinguishes itself with its ability to annotate variants in non-coding regions. It simplifies the identification of regulatory variants that may affect gene expression or joining patterns by integrating information from databases like ENCODE and Roadmap Epigenomics. Users can incorporate and highly customize their own annotation sources, to adapt into precise research needs. To analyze variant data in different sequencing platforms ANNOVAR supports numerous input files formats, such as VCF, BED (Browser Extensible Data), and VCF-like formats.

With ANNOVAR, variant annotation serves as a stepping-stone to further analysis. Researchers can utilize downstream tools to uncover candidate genes, pathways, or functional modules linked to specific diseases or phenotypes based on ANNOVAR's output. By seamlessly integrating ANNOVAR with other bioinformatics resources, researchers can gain profound insights into the biological consequences of genetic variants.

3.1.3. UniGOPred

UniGOPred (Rifaioglu et al., 2018) is an automated prediction system for Gene Ontology (GO) annotations, offering comprehensive function predictions for protein sequences in the UniProt Knowledgebase (UniProtKB) (Magrane, 2011). It covers a wide spectrum of GO system functionalities, with its thorough coverage of 514 molecular function (MF) terms, 2909 biological process (BP) terms, and 438 cellular component (CC) terms. Additionally, the approach permits thorough functional annotations by anticipating both general and specific GO phrases. The system utilizes an ensemble approach, combining predictions from three base predictor models as SPMap, BLAST-kNN, and Pepstats-SVM (Saraç, Atalay, & Cetin-Atalay, 2010). These models output prediction scores indicating the association of input proteins with

specific GO terms. Each of the three base predictors underwent training and validation individually. A single dataset was generated from UniProtKB/Swiss-Prot protein sequences and divided randomly into an 80% training dataset and a 20% validation dataset. The training aimed to determine parameter values and optimize the base predictor models, while the validation phase fine-tuned these models and identified the best hyper-parameter values. The construction of the training and validation datasets followed specific constraints. Positive datasets included proteins directly associated with the corresponding GO terms or their offspring, while negative datasets consisted of proteins lacking the specific function. Annotations with manual experimental evidence codes were used in the positive training datasets. The negative training datasets were refined to minimize false negatives, ensuring a higher accuracy of predictions.

The performance of UniGOPred was evaluated and compared to other methods. It achieved excellent results, ranking among the top ten methods in the molecular function category in the CAFA2 challenge (Zhou et al., 2019). UniGOPred outperformed the baseline BLAST classifier (Altschul et al., 1997) across all GO categories, demonstrating its superior predictive capabilities. Comparisons with annotations in the UniProtKB/TrEMBL database further validated the accuracy of UniGOPred's predictions. By leveraging advanced computing power and machine learning techniques, UniGOPred provides researchers with reliable functional annotations for protein sequences, facilitating genomics and proteomics studies.

3.1.4. InterProScan

InterProScan (Quevillon et al., 2005) is a commonly utilized software application that plays a vital role in the field of bioinformatics. It offers a comprehensive analysis of protein sequences, allowing users to explore the functional aspects of their targets. By incorporating information from the InterPro database (Paysan-Lafosse et al., 2023), users gain valuable insights that contribute to a more profound understanding of protein properties. InterProScan serves as a comprehensive tool that combines multiple computational techniques and databases to anticipate and characterize protein domains, functional motifs, and sites. Furthermore, it enriches protein analysis by incorporating Gene Ontology (GO) terms and pathway information. The latest version, InterProScan5, introduces supplementary functionalities like the prediction of signal peptides and transmembrane regions using Phobius (Käll, Krogh, & Sonnhammer, 2004). This tool facilitates the comprehension of proteins' functional attributes and distinctive traits by offering valuable perspectives on their structure, function, and evolutionary connections. InterProScan's principal role involves the identification and annotation of protein domains and sites present within a specific protein sequence. This objective is achieved through the implementation of diverse search algorithms and databases, encompassing BLAST, HMMER, Pfam, PROSITE, and SMART (Mulder & Apweiler, 2007). InterProScan incorporates BLAST functionality to enhance the efficiency of similarity searches against a vast collection of well-known protein sequences. This comparison process involves matching the provided protein sequence with those available in the reference database, enabling the detection of

similar protein domains or regions. Within InterProScan, HMMER (Finn, Clements, & Eddy, 2011) assumes a critical role by employing Hidden Markov Models (HMMs) (Bosch, 2010) to conduct profile-based searches. HMMER efficiently detects conserved protein domains by conducting a thorough comparison between the input sequence and the Pfam-A database, which comprises curated protein domain models. The Pfam database (Mistry et al., 2021) is essential for identifying protein family signatures and domain annotations in the context of InterProScan. Through the use of the extensive collection of protein domain models in Pfam, InterProScan enables the identification of conserved areas and functional elements within protein sequences. Within this framework, the inclusion of the PROSITE database serves as an additional valuable resource. PROSITE (Hulo et al., 2006) encompasses a collection of patterns, profiles, and motifs that effectively distinguishes functional domains or distinct features within proteins. InterProScan skillfully finds and describes particular functional motifs buried within protein sequences by making use of PROSITE's capabilities. InterProScan incorporates the functionality of SMART (Schultz, Copley, Doerks, Ponting, & Bork, 2000), a specialized database focusing in protein domain architectures. By leveraging SMART, InterProScan effectively examines the structural arrangement of protein domains within a provided protein sequence and identifies potential interactions between different domains.

InterProScan's outcomes deliver insightful data regarding the functional annotations affiliated with protein sequences. The tool generates comprehensive reports encompassing an overview of predicted protein domains, sites, motifs, and their associated functional annotations. These reports include diverse presentations, such as statistical summaries, graphical visualizations, and diagrams, illustrating the distribution and distinctive attributes of the identified features. As a result of running the software in the TSV (Tab-Separated Values) format, it presents the results in a structured tabular form, with distinct columns outlining various annotations and corresponding details for each predicted feature. Below are some key elements typically encompassed in the TSV output:

- Protein Identifier: The unique identifier or accession number of the protein sequence being analyzed.

- Protein Length: The reported size of the protein sequence, indicating the count of amino acids it contains.

- Domain Matches: Any matches to well-known domain signatures from databases like Pfam, SMART, or PROSITE are highlighted by the program. The presence of functional motifs or conserved regions within the protein sequence is indicated by these matches.

- Domain Accession: The accession number or identifier of the predicted domain or feature.

- Domain Description: A short explanation of the domain's name or brief description that reveals some of its features or functions.

- Domain Start and End Positions: The precise locations in the protein sequence where the anticipated domain starts and finishes.

- Statistical Analysis: The output contains statistical data, such as E-values or confidence scores, that give an estimation of the significance and reliability of InterProScan's predictions.

- Status: Refers to the annotation status of a particular protein sequence or feature. It provides information about the reliability or confidence of the annotation assigned to that protein or feature.

- Date: It indicates the timestamp of when the InterProScan tool was run on the given protein sequence or dataset.

- InterPro Accession: The accession number or identifier assigned by the InterPro database for the specific domain or feature.

- InterPro Description: A more detailed description of the InterPro domain, elaborating on its functional or structural properties.

- GO Terms: Gene Ontology terms associated with the predicted domain, describing the biological processes, molecular functions, and cellular components it is related to.

- Pathways: Information about metabolic or signaling pathways in which the predicted domain is involved, often provided through database cross-references.

In general, InterProScan is a robust and flexible tool that assists scientists in enhancing their comprehension of protein functionality and evolutionary patterns. Users have the option to utilize the tool either through a command line interface or a graphical user interface, and it is accessible as both standalone software and a web-based service.

3.1.5. Uniprot

UniProtKB, also known as UniProt Knowledgebase, is a comprehensive and precisely curated database of proteins, which provides a wide range of useful details about protein sequences, activities, annotations, and associated information. Protein entries in the database come from a variety of sources and are curated manually by experts in the field as well as automatically by annotation pipelines. These entries undergo a stringent review process to guarantee the precision and uniformity of the provided information. UniProtKB serves as a centralized hub for accessing and delving into protein-related knowledge from various organisms. It encompasses a wide spectrum

of species, including Homo sapiens (human), and is consistently updated with novel data.

Researchers can get an in-depth understanding of human proteins, their functions, genetic information, and sequence properties by examining the UniProtKB database and the related entry table. The composition of the table columns may differ depending on the particular UniProtKB version and the selected display preferences. One of the common columns is the Entry/Accession, which represents the unique identifier assigned to each protein entry in UniProtKB. The Protein Name column displays the name or description of the protein, while the Gene Names column lists the associated gene names. The Organism column specifies the source organism from which the protein is derived, such as "Homo sapiens" for human proteins. The Length column indicates the length of the protein sequence, typically measured in amino acids. Depending on classification schemes like Pfam or InterPro, the Protein Families & Domains column in the table can additionally represent details about the families or groups of proteins to which the protein belongs. The External Resources column includes links or references to other resources that contain additional information about the protein, such as Ensembl (Hubbard et al., 2002), NCBI and PDB (Protein Data Bank) (Young et al., 2018). The Function column describes the known or predicted function of the protein, offering insights into its biological role. UniProtKB Keywords are listed in a specific column, representing important aspects of the protein's biological function, cellular localization, or involvement in diseases. The Subcellular Location column indicates the subcellular localization or compartment where the protein is primarily located within the cell. Enzyme Commission (EC) Numbers, if applicable, are provided in a separate column, indicating the enzymatic activities associated with the protein. These are some of the common columns that can be found in the related table. However, the database continues to evolve, and additional columns or features may be present in specific versions or in specialized tables within UniProtKB. The application of this knowledge may be required to investigate the relationships between biological processes, illnesses, and protein structure-function correlations.

3.1.6. Alphafold

AlphaFold, an algorithm created by DeepMind (“AlphaFold and beyond,” 2023), employs deep learning techniques to achieve precise and efficient predictions of protein structures (Jumper et al., 2021). Its main objective is to address the "protein folding problem," a significant challenge in structural biology, which involves predicting the three-dimensional arrangement of a protein based on its sequence of amino acids.

AlphaFold harnesses the capabilities of advanced deep learning networks, combining convolutional neural networks (CNNs) and attention-based transformers. It draws insights from a vast library of protein structure data, including experimentally obtained structures from the PDB. Through extensive training, the algorithm develops the expertise to anticipate the spatial arrangement of amino acids in a protein, resulting in

highly accurate and dependable predictions of its three-dimensional structure. The application's search function allows users to input queries using various criteria such as protein name, gene name, UniProt accession, or organism name. While direct support for BLAST or sequence-based searching is not available on related site, an alternative option is provided. Users can utilize the EBI Protein Similarity Search tool (Park, Squizzato, Buso, Gur, & Lopez, 2017) to explore the AlphaFold DB by entering a query sequence.

The algorithm can accurately predict the structures of proteins, reaching levels of precision that rival traditional methods like X-ray crystallography and cryo-electron microscopy. Accurate predictions of protein structures enhance our comprehension of their functions, interactions, and underlying mechanisms, offering valuable insights into diverse biological phenomena and disease pathways. It goes beyond just providing a general overview by offering intricate details, such as precise atomic-level coordinates for the protein's backbone and side chains. This allows researchers to explore the intricate architecture of the protein and identify potential regions that are crucial for its function. AlphaFold is versatile, capable of predicting structures for both soluble proteins and membrane proteins, accommodating a wide range of biological systems. It even provides confidence scores alongside its predictions, enabling users to assess the reliability and strength of the generated structures. This empowers researchers to make informed decisions and confidently explore the fascinating world of protein structures. In addition, AlphaFold's potential to accelerate the discovery and design of new drugs is truly remarkable. It aids in pinpointing potential targets for therapeutic interventions and foreseeing how small molecules will interact with proteins, paving the way for innovative approaches to developing life-saving treatments.

DeepMind has actively participated in the Critical Assessment of Structure Prediction (CASP) competitions (Kryshtafovych, Schwede, Topf, Fidelis, & Moult, 2019), providing an opportunity to rigorously evaluate and validate the algorithm's performance. In a spirit of openness and collaboration, DeepMind has also made the source code and models of AlphaFold available to researchers, fostering an environment of continued exploration and advancement in the realm of protein structure prediction.

3.1.7. SwissModel

SwissModel is a widely recognized and user-friendly bioinformatics tool that scientists rely on to predict protein structures (Schwede, Kopp, Guex, & Peitsch, 2003). Accessible through a web-based platform, it empowers researchers to generate accurate and reliable models by utilizing existing structural templates. With its ability to bridge the gap when experimental structures are unavailable, SwissModel has become an invaluable resource for scientists seeking precise and dependable protein models.

The platform employs a methodology known as comparative modeling, which is also referred to as homology modeling. This technique operates on the premise that proteins sharing similar sequences of amino acids tend to exhibit comparable three-dimensional structures. Creating a homology model involves a multi-step process. It begins with the identification of appropriate structural templates that closely resemble the target protein. In comparative modeling, the target protein sequence is compared to a collection of known protein structures obtained through experiments. This comparison seeks out templates that exhibit sequence similarity to the target protein. Once suitable templates are identified, they serve as a foundation for constructing a model that represents the structure of the target protein. Following the selection of appropriate template structure(s), the target protein sequence is aligned with these templates to determine corresponding positions. Based on this alignment, the model is constructed, taking into consideration the arrangement of amino acids and other structural attributes. The model's quality is then assessed to gauge its dependability and precision. Refinement and optimization processes are employed to further enhance the alignment and adjust local structures, ensuring compatibility with the target protein sequence. Through these iterative steps, a final model is generated, ready to provide valuable insights into the structure and function of the target protein.

SwissModel presents researchers with an intuitive online interface, enabling them to submit protein sequences and access a wide range of modeling and analysis tools. Within the platform, researchers can leverage visualization tools to examine the generated models, assess their quality, and delve into potential functional sites or binding regions. In order to enhance the precision and comprehensiveness of its modeling predictions, SwissModel continually updates its database with newly available protein structures from diverse sources, including the PDB. This dynamic integration of data ensures that the platform remains a valuable asset for researchers across various disciplines, spanning structural biology, drug discovery, and the exploration of protein functionality.

3.1.8. AutoDock Vina

AutoDock Vina (Trott & Olson, 2010) serves as a powerful software tool within the realm of computational drug discovery and the refinement of structure-based lead compounds. It empowers scientists to unravel the intricate world of binding interactions between small molecules, known as ligands, and their target proteins. By employing the power of molecular docking, AutoDock Vina enables the accurate prediction and thorough analysis of binding interactions, shedding light on the binding mode and affinity between ligands and receptors. Its specialization in this area positions AutoDock Vina as a reliable and proficient software, catering to the unique demands of molecular docking and facilitating innovative advancements in the field of drug discovery.

The software employs a scoring system to estimate how well the ligand and receptor bind together. This scoring system takes various factors into consideration, including clashes, hydrogen bonds, and electrical interactions. It evaluates different poses of

ligand-receptor binding and ranks them based on their compatibility. AutoDock Vina possesses an impressive capability to handle flexible ligand and receptor structures. Although the docking process itself involves rigid body docking, both the ligand and receptor can have flexibility in their structures. This flexibility enables the software to investigate a diverse array of ligand conformations, as well as account for the limited movement of the receptor during the docking simulation.

In order to employ AutoDock Vina, users must input the ligand and receptor structures into the software. These structures are commonly submitted in the PDB format. AutoDock Vina generates output that encompasses anticipated binding configurations. These configurations deliver details regarding the ligand's coordinates within the binding site, the projected binding strength, and various other significant parameters. The tool provides users with the ability to modify the configuration settings and parameters according to their unique docking simulation needs. Customization options include defining the search area, determining the exhaustiveness (number of generated docking poses), and specifying the energy range. Furthermore, users have the flexibility to set constraints or preferences, such as constraining specific areas of the ligand or receptor during the docking process. This versatility empowers users to optimize the simulation conditions for their specific requirements.

The outcomes of the docking simulations performed with this application can be subjected to additional analysis and visualization using software specialized in molecular visualization. This facilitates the exploration of potential binding interactions, assessment of the binding affinity across different docking poses, and the prioritization of ligands based on their anticipated binding energies. By employing such visualization tools, researchers can gain valuable insights and make informed decisions regarding the ligand-receptor interactions under investigation.

3.1.9. MGLTools

MGLTools (Sanner, 1999), also known as Molecular Graphics Laboratory Tools, is a specialized software package created to facilitate the visualization, analysis, and manipulation of molecular structures and associated data. This all-encompassing package offers an extensive assortment of tools and utilities dedicated to the fields of molecular modeling, docking, visualization, and simulation. Scientists and researchers working in the fields of structural biology and computational chemistry heavily rely on MGLTools to investigate macromolecules, explore the complex interactions between ligands and receptors, and delve into the fascinating realm of drug design.

One of the standout aspects of MGLTools lies in its adeptness at seamlessly working with multiple file formats prevalent in the fields of molecular biology and chemistry. These file formats, such as PDB, MOL2 (Tripos SYBYL Molecular 2), and PQR (Partial Charge Representation), are effortlessly handled by the software. The software empowers users with the ease and convenience of importing, exporting, and manipulating these file formats. The system is designed to present researchers with an intuitive graphical interface that facilitates dynamic visualization and analysis of

molecular structures. Within this user-friendly platform, users can navigate through lots of visualization options. These include the ability to depict molecules in diverse representations, such as ball-and-stick, wireframe, and surface renderings. Additionally, it offers the flexibility to color atoms and residues based on various properties, as well as apply captivating visual effects that accentuate specific features of interest.

Beyond its visualization capabilities, MGLTools encompasses a suite of functionalities dedicated to molecular docking. This entails the intricate task of forecasting the binding conformation and affinity between a ligand and its intended receptor. Notably, this tool seamlessly integrates with AutoDock Vina, harnessing its power to conduct precise docking simulations. Researchers can leverage this integration to execute docking experiments and evaluate the resultant docking poses and binding energies directly within the MGLTools framework. By merging these progressive tools, MGLTools equips researchers with an all-in-one solution for the exploration and analysis of molecular docking phenomena, amplifying their ability to unravel the complex realm of ligand-receptor interactions.

3.1.10. PyMOL

PyMOL is a molecular visualization system developed by Warren Lyford DeLano that offers a flexible solution for visualizing and analyzing complex three-dimensional biomolecular structures (Schrödinger, 2010). With its open-source yet proprietary nature, PyMOL stands out among the limited options of mostly open-source tools dedicated to structural biology. Researchers can leverage PyMOL to gain deep insights into the intricate architecture of biomolecules, including proteins, nucleic acids, and small molecules, enabling comprehensive exploration and analysis of their functional properties and interactions.

In order to further enhance the exploration and analysis of molecular structures, PyMOL provides diverse visualization styles, such as cartoon, surface, and stick representations. It enables the display of electrostatic potentials, molecular surfaces, and molecular dynamics trajectories, facilitating in-depth examination of molecular properties. Furthermore, PyMOL includes tools like Ramachandran plots for assessing protein structure quality and supports the visualization of experimental data, such as electron density maps. The ability to cover multiple structures further facilitates comparative analysis.

In addition to its extensive visualization capabilities, PyMOL provides users with the capability to import molecular structures in diverse file formats, enabling manipulation and personalized customization of visualization settings for comprehensive exploration. The software's extensive array of functionalities includes rendering options that yield visually striking images, measurement tools for precise analysis of distances and angles, and the capacity for molecular editing to facilitate structural modifications as needed. Moreover, the tool offers an extensive range of sophisticated capabilities that facilitate the execution of intricate operations, including molecular

docking, in-depth analysis of protein-ligand interactions, and refinement of protein structures. Additionally, PyMOL's integration of Python scripting empowers users to automate tasks, and personalize workflows to their specific requirements.

3.1.11. Nextflow

While working at the Centre for Genomic Regulation (CRG) in Barcelona, Paolo Di Tommaso, a bioinformatician and software developer decided to create a powerful and scalable framework for orchestrating and executing computational workflows. By his efforts, Nextflow (Di Tommaso et al., 2017), is designed to simplify the development and execution of complex scientific pipelines with data-driven computational workflow at its core. Di Tommaso faced challenges in large-scale genomics data analysis and got motivated to create a well-suited tool for parallelized tasks in the field.

In Nextflow, we can create pipelines and distribute environments, to abstract away the complexity of resource management and assign tasks for interconnected computational processes. With flexibility of writing in different programming languages such as python and R, Nextflow can represent a unit of work with a task that performs a specific operation on input data, and generates output data as a result.

One of the outstanding features of Nextflow is the ability to handle concurrent execution of tasks in different environments, such as local machines, clusters, or cloud infrastructures. By looking at dataflow programming model, Nextflow can abstract away the underlying execution of environments and handle parallelism based on input and output dependencies. The domain-specific language (DSL) is the foundation of Nextflow which gives users the luxury of describing pipeline in a concise and intuitive manner (Figure 5).

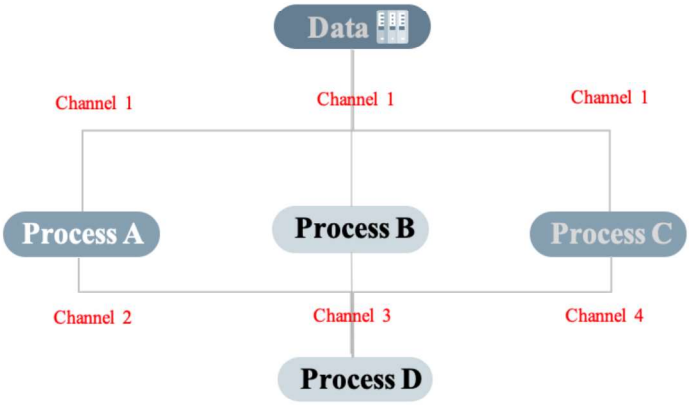

Figure 5: The workflows in Nextflow are depicted as graphs, with nodes representing individual processes and edges representing the channels that connect them. Processes in Nextflow are essentially blocks of executable code, such as scripts or programs, while channels serve as asynchronous queues that facilitate the communication of input and output between processes. One notable characteristic of Nextflow is that processes are independent of each other and can be executed in parallel, depending on the availability of data within the channels. In the given example, processes A, B, and C can be executed concurrently, and only when all of them have completed their tasks, process D will be triggered to start. This parallel execution capability enhances the efficiency and speed of workflow execution in Nextflow.

In 2013, first public version of Nextflow is released, with built-in support for data provenance and traceability which makes capture metadata in the execution. Making it available to the scientific community under an open-source license, Nextflow has gained significant popularity by its strength in reproducibility, auditing and tracking the linage of data and results.

Overall Nextflow simplifies into the integration of external tools and libraries through its extensive support for interoperability. It offers researchers a scalable and reproducible platform for effectively addressing data-intensive and computationally challenging tasks across diverse scientific disciplines by supporting for containerization, data provenance and collaboration.

3.1.12. Flask

With the power of Python programming language at its core, Flask (Grinberg, 2018) is an easy to use and lightweight web framework for developing scalable and robust web applications. Armin Ronacher the creator of flask believed in "micro" frameworks, which give flask the power to focus on the core functionality and lets developers to integrate libraries as needed. While simple and versatile flask provide the foundation for handling HTTP requests and responses at its core. Among key features of flask, routing URLs to specific functions is interesting because of its nature to handle corresponding request. It gives developers the luxury of having a clean and organized code structure while different routes and their associated logic are defined within distinct view functions. By having Werkzeug WSGI (Web Server Gateway Interface) toolkit, at its foundation to create web forms in Python, flask also integrates Jinja2 to render dynamic HTML templates.

Another distinguished feature of Flask is its flexibility which lets developers to adjust several architectural decorations, based on their project requirements and preferences. With Flask's support of extensions for third party libraries, it does not dictate how application should be organized, and gives developers the freedom to design and integrate additional functionalities such as database integration or form validation. Flask-RESTful provides developers the ability to build RESTful APIs and supports various HTTP methods such as GET, POST, PUT, DELETE and so on.

Flask can handle small to medium-sized applications with ensuring performance by its built-in development server. The automatic reloading of code changes, allows developers to test and debug applications instantly and manually at their local machine. All and all, Flask is lightweight, flexible and modular with a solid foundation for web development in Python.

3.1.13. Mailjet

A Mailjet is a cloud-based email marketing software that offers a comprehensive set of tools for managing and sending emails at scale as well as tracking and delivering email campaigns. With its powerful features, business owners and developers are able to generate, send and track specialized emails and transactions effortlessly to

subscribers. There are wide range of tools in Mailjet to be applied and used such as email design tools, segmentation and automation. As of the time this article was written, Mailjet has gained overwhelming admiration because of its user-friendly interface and reliable infrastructure to send emails securely. With robust platform, Mailjet guarantees impressive delivery success rate to intended recipients' inboxes and handles a high volume of email communication for applications and businesses with considerable email demands.

With industry-standard security protocols, such as applying encryption algorithms and maintaining highly secure data centers Mailjet prioritizes data security and privacy. With the help of dashboards like click-through rates, bounce rates and unsubscribe rates, users can measure the success rates of email campaigns. The thrilling aspect of Mailjet is a robust API and SDK infrastructure, which allows developers to integrate Mailjet's marketing functionalities into their websites or applications. For our application purposes, we leverage Mailjet's APIs to produce email solutions, such as custom email and automated output attachment, tailored to our PMPred requirements.

3.2. Method

An overview of the PMPred method is shown in Figure 6. The PMPred tool is a workflow of functional analysis tools that are connected together using the Nextflow framework.

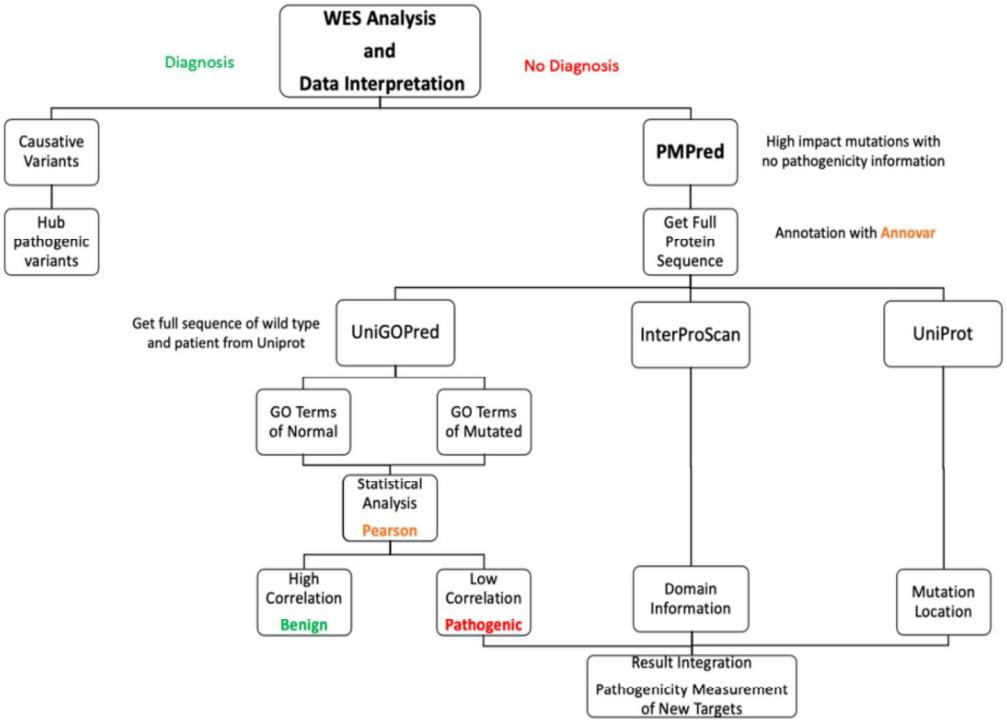

Figure 6: The PMPred Workflow. PMPred is aimed toward cases where no known variant causes the observed symptoms. The pipeline PMPred employs to analyze the pathogenicity of such variants includes UniGOPred pathogenicity prediction from sequence, domain loss analysis using InterProScan, and variant annotation using Annovar.

Provided a list of variants in VCF format, the pipeline performs all the calculation steps required to return the final prediction results to the user. First, variant annotation is required to extract the target sequences and their genetic information. Function prediction with UniGOPred follows this step to evaluate the effect of the changes on protein function. Simultaneously, the workflow applies domain prediction to identify damage in sequences' critical domains. The data acquired from the function and domain prediction phases is then used to estimate the pathogenicity of the target variations. Additionally, information regarding the location of mutations is displayed to assist the user in understanding the data.

3.2.1. Data collection

I retrieved the Variant Call Format (VCF) file from the ClinVar database, which contains a wealth of information about genetic variants and their clinical significance. The VCF file was obtained from the official ClinVar FTP server at the address https://ftp.ncbi.nlm.nih.gov/pub/clinvar/vcf_GRCh37/.

To extract specific variants from the VCF file, I applied certain filtering criteria based on the CLNSIG (Clinical Significance), CLNREVSTAT (ClinVar Review Status), and GENEINFO fields. My goal was to focus on variants that are pathogenic (P) or likely pathogenic (LP) and variants with benign (B) or likely benign (LB) interpretations, as indicated by the CLNSIG field. In order to ensure reliable and robust variant selection, I prioritized variants that met the criteria of "multiple submitters, no conflicts" as stated in the CLNREVSTAT field. This criterion indicates that multiple independent sources have provided consistent interpretations for these variants, enhancing their reliability. However, since the number of variants meeting this criterion was limited, I also included variants with single submitters and no assertion to increase the diversity of the dataset. Moreover, in the GENEINFO field, I focused on selecting variants associated with diverse gene symbols that encompassed frameshift, in-frame, or nonsense mutations. This approach allowed me to cover a broad range of genetic alterations across different genes of interest (Figure 7).

In the data collection phase of this methodology, the attention was given to selecting an approximately equal number of pathogenic and benign variants, ensuring a balanced representation in the dataset. Special consideration was given to stop-gain variants located near the beginning of the protein, as these are known to have more pronounced disruptive effects on protein structure and function. By prioritizing these variants, valuable insights into the underlying pathogenic mechanisms could be obtained. Regarding insertions and deletions, regions that are GC-rich or known to be repetitive were excluded. These regions are susceptible to sequencing errors and technical artifacts, which could introduce biases and inaccuracies into the analysis. By excluding variants in these regions, the dataset was refined to include only those with a higher likelihood of accurate assessment of functional impact. These selection criteria, encompassing the balance between pathogenic and benign variants, the focus on disruptive stop-gain variants near the protein's start, and the exclusion of variants in GC-rich or repetitive regions, collectively contributed to the construction of a robust

and comprehensive dataset for the research. This approach ensured a representative sample of variants suitable for in-depth analysis of their pathogenicity and functional implications.

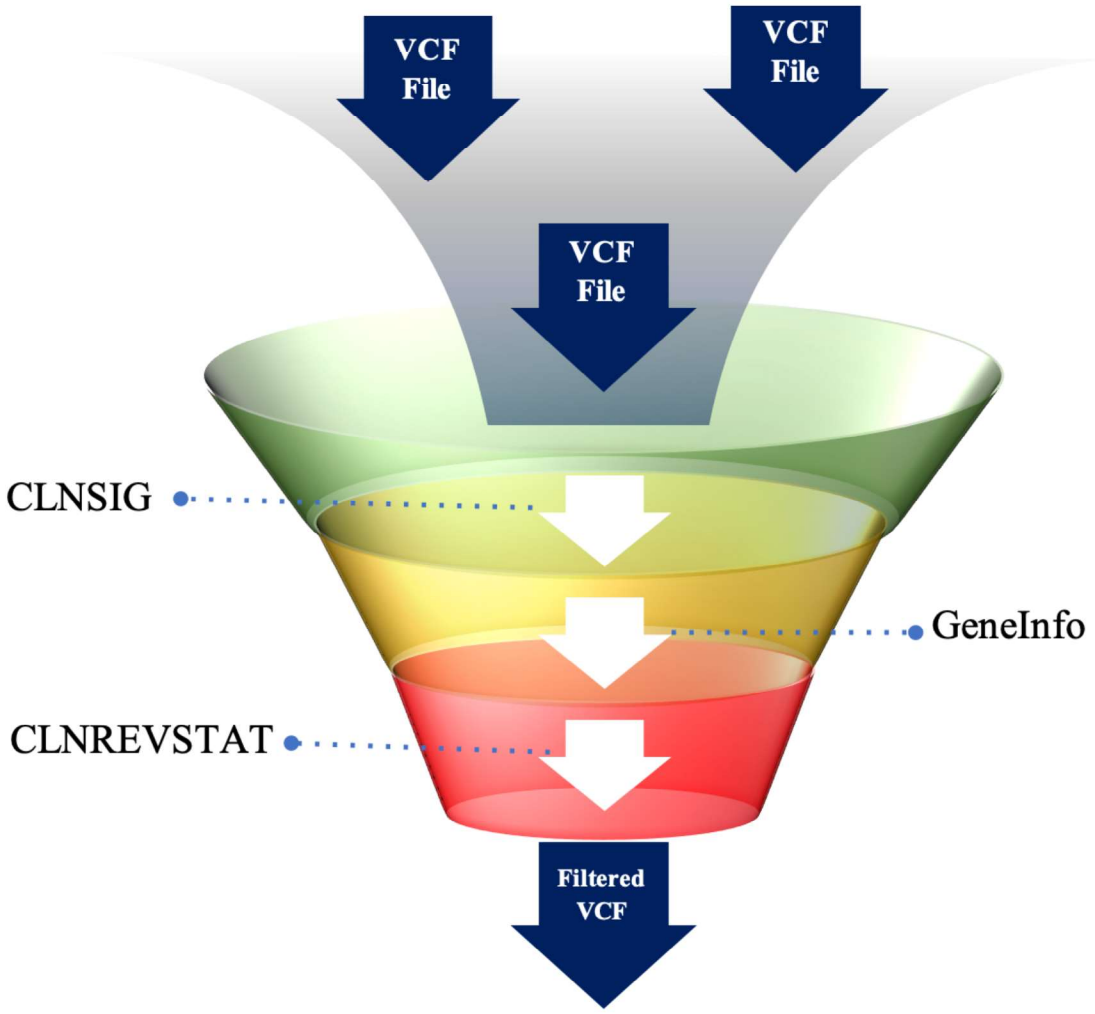

Figure 7: Filtered variants for analysis based on CLNSIG, CLNREVSTAT, and GENEINFO criteria: prioritized variants classified as pathogenic/likely pathogenic or benign/likely benign according to CLNSIG; selected variants with "multiple submitters, no conflicts" in CLNREVSTAT for enhanced reliability; included variants with single submitters and no assertion to increase dataset diversity; focused on diverse gene symbols encompassing frameshift, in-frame, or nonsense mutations in GENEINFO.

Finally, a selective group of in-frame, frameshift and nonsense variants were collected from the NCBI ClinVar database. According to the selection process only, items classified as "benign" or "likely benign" and either "pathogenic" or "likely pathogenic" were included. The number of variants used for this study were 272, out of which 138 variants are predicted to have pathogenic or likely pathogenic effect, and 134 variants

have benign, and likely benign consequences. In total, the used dataset contains 107 nonsense, 40 in-frame, and 125 frameshift mutations.

3.2.2. Variant annotation

Variant annotation is an important step to assign the functional information of variants (Lek et al., 2016). Annovar, SnpEff (Cingolani et al., 2012), and Variant Effect Predictor (VEP) (McLaren et al., 2016) are currently the three most common tools used to characterize variants. All three methods are efficient and widely used in literature to annotate genetic variants detected from diverse genomes. Out of these methods, Annovar is the only one that offers a tab separated text output. For each submitted VCF file, our pipeline can effectively annotate variants and extract related protein sequences using the ANNOVAR packages. This method can identify the intergenic and exonic variants to report them.

ANNOVAR, a widely used tool in genomic research, provides a useful collection of packages designed for efficient variant annotation and classification. Within this comprehensive suite, three key packages were specifically employed in the annotation procedure: 'convert2annovar.pl', 'annotate_variation.pl', and 'coding_change.pl'. Each of these packages plays a crucial role in facilitating the annotation process and generating essential information for subsequent analysis. After downloading it from https://www.openbioinformatics.org/annovar/annovar_download_form.php, I proceeded to utilize the included packages to enhance my analysis pipeline.

The first crucial step in the annotation process involved utilizing the 'convert2annovar.pl' package, a powerful tool offered by ANNOVAR. This package played a pivotal role in transforming the VCF file, obtained from the ClinVar database, into a format that is compatible and seamlessly processed by ANNOVAR. In fact, it facilitated the conversion of the VCF file into an input format specifically designed for ANNOVAR, utilizing the '-format vcf4' argument in the 'convert2annovar.pl' package. Upon executing the mentioned package, I obtained the 'avinput' file, which served as the transformed input format for subsequent analysis. This conversion involves mapping the genomic coordinates of the variants, extracting relevant information such as chromosome number, position, reference allele, alternate allele, and other necessary fields, and organizing them in a structured manner. The used package implemented a well-defined set of rules and procedures to accurately capture and represent the variant data in a required format. This process was essential to ensure that the variants extracted from the input file could be efficiently annotated and classified using the subsequent ANNOVAR packages.

```
convert2annovar.pl -format vcf4 ${PWD##*/}.'vcf' >
${PWD##*/}.'avinput'
```

Following the conversion step, the 'annotate_variation.pl' package took center stage. This package performed the essential task of annotating the variants by assigning functional and genomic attributes to each variant. It provided valuable information

about the nature and potential impact of the variants on gene function and their genomic context. The 'annotate_variation.pl' package implemented a robust and comprehensive annotation algorithm that precisely analyzed the variants. It classified each variant into specific categories based on their location and functional consequences. This classification encompassed a wide range of possibilities, including intergenic variants, intronic variants, non-synonymous SNPs that cause amino acid changes, frameshift deletions resulting in a shift in the reading frame, and many other variant types. During the annotation process, the tool utilized various databases and resources to retrieve critical information about the variants. By integrating these diverse sources of information, the package provided a comprehensive annotation report that elucidated the potential functional impact of the variants.

The 'annotate_variation.pl' package provides the flexibility to customize the annotation process through various arguments. The "geneanno" argument allows users to select the specific type of gene-based annotation they wish to perform, such as RefSeq genes, knownGene, or ensGene. This enables targeted annotation of variants within the desired gene set. The "dbtype" argument is used to specify the annotation database to be used, such as RefSeq (O'Leary et al., 2016), dbSNP, or COSMIC (Tate et al., 2019), ensuring the availability of relevant genomic information during annotation. The "buildver" argument is used to indicate the genome assembly version being used, such as "hg19" or "hg38", ensuring accurate mapping of variants to the correct genomic coordinates. Finally, the "humandb" argument specifies the directory containing the ANNOVAR annotation databases for human genomes, allowing access to a comprehensive set of annotations tailored for human variants. During the annotation process, the "dbtype" was set to "refGene" and the "buildver" was set to "hg19", enabling the annotation to be performed using the RefSeq gene database and the human reference genome assembly, respectively.

During the analysis conducted for data extraction, I ran the 'annotate_variation.pl' package, resulting in the generation of essential output files, including 'avinput.exonic_variant_function', 'avinput.log', and 'avinput.variant_function'. The 'avinput.exonic_variant_function' file contains functional annotations specifically for exonic variants, such as missense, nonsense, frameshift, or synonymous mutations. This file allows for a detailed understanding of the impact of the variants on protein-coding regions. The 'avinput.variant_function' file provides a broader view of the variant annotations, including information on the genomic location and functional impact of the variants. It classifies the variants into categories such as intergenic, intronic, upstream, downstream, or splicing variants. This file helps identify the genomic context and potential regulatory effects of the variants. The 'avinput.log' file serves as a log file, recording any warnings, errors, or messages generated during the annotation process. It provides important diagnostic information that can help troubleshoot any issues and ensure the accuracy and reliability of the annotation results.

```
annotate_variation.pl -geneanno -dbtype refGene -
buildver hg19 ${PWD##*/}.'avinput' humandb/
```

At the final step of annotation procedure, the 'coding_change.pl' package, utilizing the 'avinput.exonic_variant_function' file as input, plays a crucial role in generating the 'avinput.txt' output. To perform this task, it relies on two key arguments: 'hg19_refGene.txt' and 'refGeneMrna.fa'. 'hg19_refGene.txt' is a reference file that contains comprehensive information about the genomic locations and structures of annotated genes in the human genome. It serves as a critical resource for mapping variant annotations to specific genes and their corresponding genomic features. 'refGeneMrna.fa' is a FASTA-formatted file that provides the reference sequences for the coding regions of genes. This file contains the wildtype DNA sequences of the genes, allowing the identification of both the wildtype and altered sequences for each variant. By utilizing these reference files and the information from 'avinput.exonic_variant_function', the 'coding_change.pl' package generates the 'avinput.txt' output file. This file includes the wildtype and altered sequences of each target in the list, providing valuable insights into the coding changes associated with the variants and facilitating further analysis of their potential functional consequences.

```
coding_change.pl
${PWD##*/}.'avinput.exonic_variant_function'
humandb/hg19_refGene.txt humandb/hg19_refGeneMrna.fa >
${PWD##*/}.'txt'
```

3.2.3. Format conversion

The format conversion process is implemented to transform the annotation output file, obtained from ANNOVAR, into a format compatible with the UniGoPred tool. This conversion was necessary to facilitate subsequent Gene Ontology (GO) term enrichment analysis and activity measurement of the provided sequences.

In order to achieve this conversion, I have written a specific code in Python. The conversion code began by importing the required modules and opening the input and output files. It then proceeded to iterate through each line of the provided annotation file, applying a series of transformations and substitutions to obtain the desired format. These transformations included removing unnecessary characters, such as parentheses and spaces, and replacing specific patterns with appropriate identifiers. As an example, the code identified and adjusted wildtype and mutated sequences, added delimiters to separate different parts of the sequences, and modified the headers to comply with the later needed format. Finally, after applying the necessary transformations, the modified sequences were written to the output file in FASTA format. This transformed file could then be readily used as input for function and domain prediction processes.

```
python annovar2unigopred.py ${PWD##*/}.'txt'
${PWD##*/}.'fasta'
```

3.2.4. Function prediction

Following our pipeline, the wildtype and mutated sequences of each variant goes through the functional analysis in the next step. We used UniGOPred, a well performer in functional prediction, to measure the variations' impact on protein function. The UniGOPred method is built on a system made up of predictors, where each GO phrase has a separate predictor. This automated protein annotation algorithm depends on BLAST-kNN for similarity search, SPMap to conduct sequence based functional prediction and Pepstats-SVM (Rice, Longden, & Bleasby, 2000) for statistical measurement of the results. The method covers almost the entire ontology referring to molecular function (MF), biological processes (BP), and cellular components (CC) in the system.

The analysis started by preparing the input files required for UniGoPred. I provided the UniGoPred tool with the GO term file, denoted as "GOterms50MF.txt," which contained the relevant GO terms for the analysis. Additionally, I provided the tool with the transformed FASTA input file, denoted as "UniGOPred_Input.fasta," which contained the wildtype and mutated sequences obtained from the ANNOVAR annotation output. The tool requires specific command-line arguments to run effectively on FASTA input files. I invoked the command below to execute the specified analysis.

```
Ruby runGOPredMF.rb GOterms50MF.txt
UniGOPred_Input.fasta SequencePreds/UniGOPred_Output 6
10
```

The command initiated the targeted analysis by specifying the input GO terms file, the input FASTA file, the output directory, the number of processing units and the batch size. The runGOPredMF.rb script served as the driver script for running the UniGoPred analysis. It performed several crucial tasks to facilitate the analysis. Firstly, it created the necessary directories for storing intermediate and final results. These directories included "spmap," "blast," "pepstats," and "results."

The script then proceeded to split the GO terms file into multiple parts for parallel processing. This step ensured efficient utilization of computing resources and accelerated the analysis. Next, the script processed the input FASTA file in batches. It read the sequences from the FASTA file and organized them into batches, where each batch consisted of a predefined number of sequences (specified by the batch size). For each batch, the script performed the following steps in parallel:

- Blast Query: The sequences in the batch were subjected to a BLAST query against a specified database. This step aimed to identify similar sequences and gather additional information for further analysis.

- Fasta2Pepstats: The batched sequences were processed using the Fasta2Pepstats tool. This tool calculated various protein statistics, such as

molecular weight and amino acid composition, which were essential for the subsequent analysis.

- GO Term Prediction: The batched sequences were analyzed against the GO terms using the "runBatch.rb" script. This script integrated the sequence information with the GO terms to predict the relevant GO terms for each sequence.

After processing all batches, the script combined the results and eliminated any duplicate predictions. It also removed problematic sequences from the analysis, which were identified based on size criteria. The resulting predictions for each variant sequence were stored in separate files, denoted as "#{tmpid}.preds," within the "results" directory. These prediction files contained the GO terms associated with the wildtype and mutated sequences, along with their respective values.

UniGOPred thus provides the GO terms associated with functions that each sequence is involved in. Of these results we only use those in the Molecular Function (MF) tree. The list of GO terms is the same for each sequence given to UniGOPred. What changes is the activity score associated with each term based on whether the input was the wildtype sequence or the mutated sequence. Therefore, as protein activity in a certain function changes in a protein due to mutation, the related score of the mutated sequence will tend to change accordingly. In this regard, we are interested in the UniGOPred activity score differential between the wildtype and mutated sequences of each protein under investigation. For a given molecular function, this differential serves as an indicator of the functional effect of the mutation on said protein.

3.2.5. Domain prediction

Besides functional analysis using UniGOPred, the domain prediction process can reveal damage in critical domains of sequences. A domain is a polypeptide chain that may fold and function biologically even if the rest of the chain is removed. Each protein can be made up of a single or multiple domains that play a role in its essential physiological activities. Thus, identifying the domain composition of proteins is a necessary step for precise assessment of proteins' function. InterProScan22 is one of the tools which can address this requirement as it gives an overview of the families that a protein belongs to and the domains it contains. Our pipeline applies domain analysis on the wildtype sequence and mutated sequence of each protein using InterProScan. Consequently, the conserved domains of sequences are obtained and results' comparison reveal the changes such as domain gain or loss after alteration.

To predict protein domains and functional sites, I installed the InterProScan 5 software and utilized its capabilities. The installation process involved the following steps:

- Obtaining the InterProScan Software: To begin, a dedicated directory was established with the name "my_interproscan". This directory served as the designated location for storing the InterProScan software and related files.

After creating the "my_interproscan" directory, I navigated into it using the appropriate command. This step ensured that all subsequent actions were performed within the intended directory. Once inside the required directory, the next step was to initiate the download of the InterProScan 5 software package. This was accomplished using a specific set of command-line instructions tailored for this purpose. These commands were designed to fetch the necessary software package and its associated md5 checksum. In order to ensure the integrity of the downloaded files, the md5 checksum was utilized. The md5 checksum is a unique identifier that verifies the authenticity and integrity of the downloaded software package. By comparing the calculated md5 checksum of the downloaded file with the provided checksum, any potential errors or discrepancies in the downloaded files could be identified.

– Extracting the Software Package: After successfully downloading the InterProScan software package, the next step involved extracting its contents to obtain a self-contained version of InterProScan 5. This process was carried out using a designated command tailored for extracting "tar" files. By executing this command, the "tar" file was unpacked, revealing the diverse components that constitute the InterProScan 5 software package. These components encompassed member database specific binaries, model/signature files, and other essential elements necessary for the proper functioning of InterProScan. The resulting self-contained version of InterProScan 5 provided a comprehensive and cohesive environment for executing the software. This version was meticulously designed to seamlessly operate on Linux systems, ensuring compatibility and optimal performance. By successfully extracting the software package, I obtained a self-contained version of InterProScan 5, equipped with all the requisite components to carry out subsequent protein domain and functional site predictions. This extraction process set the foundation for leveraging the capabilities of InterProScan and performing accurate and efficient analysis of protein domains and functional sites.

```
tar -pxvzf interproscan-5.59-91.0-*-bit.tar.gz
```

– Indexing HMM Models: Before proceeding with the execution of InterProScan, an essential step involved indexing the Hidden Markov Model (HMM) models. This indexing process was initiated by executing a specific command, carefully designed to prepare the HMM models in a format compatible with hmmscan, which serves as a crucial component of InterProScan. The purpose of indexing the HMM models was to optimize their accessibility and compatibility within the InterProScan framework. By organizing and structuring the HMM models in a suitable format, the subsequent domain prediction process would be characterized by enhanced accuracy and efficiency.

```
python3 setup.py interproscan.properties
```

For domain prediction, I utilized InterProScan by running the command below. In this command, I specified the "Pfam" application, which focuses on predicting protein domains using the Pfam database. The input file, "sequences.fasta," was obtained from the output of the format conversion process. This file contained the protein sequences in FASTA format. Executing the InterProScan command generated a tab-separated values (TSV) file that contained the predicted Pfam domains for each protein sequence. The TSV file provided information about the protein domains, including their names, positions, and associated statistics. By employing InterProScan, I could accurately predict protein domains and obtain crucial insights into the functional characteristics of the protein sequences. The Pfam domain predictions helped in further understanding the protein's structure, function, and potential functional implications of the observed variants.

```
~/my_interproscan/interproscan-5.53-
87.0/interproscan.sh -appl Pfam -i
~/sequence.avinput.fasta -f tsv
```

3.2.6. Subcellular location prediction

The impact of a variant relies on both the type of variant (e.g. indel, nonsense) and its location on the target sequence. A nonsense mutation leads to an early stop codon which typically provides a truncated, ineffective protein product. Besides that, frameshift mutations result in misreading of the rest of sequences by adding or removing bases from the nucleotide strand. Therefore, in addition to the mutation type, its position on the sequence can define different degrees of disease severity. For example, a mutation in splice site region can lead to incorrect editing of the related transcript. Also, a substitution at the beginning of the sequence that leads to a stop gain mutation and terminates translation in its early stages may have a significantly greater impact on protein function than the same substitution at the end. Based on this information, we extract the human protein data from Uniprot23, a database of protein sequence and functional information. Accordingly, to give details relevant to the location feature, we compute the distance of these variants from the N-terminus by considering the length of each protein and location of corresponding variants on sequences.

In order to initiate the process, I navigated to the UniProt Knowledge Base (UniProtKB) website using the provided URL, which is "https://www.uniprot.org/uniprotkb?query=%2A". Within UniProtKB, I refined my search parameters to focus specifically on human organisms known to possess protein products. This filtering step aimed to retrieve relevant data for my analysis. To extract the essential information, I tailored the result table according to my requirements. I

customized it to display crucial details such as the UniProt entry ID, gene symbol, and the corresponding sequence length of the proteins. These specific data points were vital for my subsequent analyses. Once I had configured the result table to include the desired information, I proceeded to download the table as an Excel file. This format facilitated further examination and exploration of the retrieved data, allowing for in-depth analysis and the potential discovery of meaningful insights.

Using the data, obtained from the downloaded file, I devised a code snippet to perform a comparison between each mutation present in the target gene and its corresponding sequence length. The code was designed to operate on prediction files that possessed a "preds.csv" extension, which were acquired from UniGOPred. These prediction files specifically contained the NM number (RefSeq accession number) and mutation information. For the prediction files ending with "ALTERED.preds.csv," the NM number and mutation information were extracted directly from the file name. The NM number served as a unique identifier for the gene in the RefSeq database, while the mutation information was presented in the cDNA format, identified by the "c." prefix. Additionally, the line identifier associated with the UniProt entry linked to the mutation was retrieved. To obtain the gene symbol, the MyGeneInfo Python library (Wu, Macleod, & Su, 2013) was utilized, using the NM number as a reference sequence identifier. By querying the library, the gene symbol was retrieved and subsequently transformed to uppercase for consistency. Moving forward, the gene length information extracted from the downloaded Excel file was iterated through. By comparing the gene symbols, the relevant gene's length was successfully matched with the one of interest. This process allowed for comprehensive analysis and comparison of mutation data with their corresponding gene sequence lengths.

With the gene length information at hand, I performed calculations to determine the percentage of a mutation's occurrence in relation to the N-terminus, which corresponds to the gene's starting codon. This involved a series of steps to accurately derive the desired percentage. To initiate the calculation, I divided the location of the mutation by three since three nucleotides make up one codon in the genetic code. This step was crucial to convert the mutation's position from nucleotides to codon units. Subsequently, the obtained value was divided by the gene's length, enabling a relative comparison of the mutation's location within the gene sequence. The result was presented in a more intuitive manner by multiplying the calculated value by 100 to express it as a percentage. This transformation allowed for a clearer representation of the mutation's position relative to the N-terminus. The percentage value was rounded to the nearest whole number to ensure its conciseness and comprehensibility.

By following this systematic approach, the final outcome provided a precise percentage that indicated the mutation's distance from the N-terminus of the gene. This information proved valuable in assessing the spatial location of the mutation within the gene sequence and gaining insights into its potential functional implications.

3.2.7. Pathogenicity prediction

After getting the prediction results of all targets from UniGOPred and InterProScan, the wildtype and altered sequences are compared according to their GO terms. This comparison step is central to assessing the pathogenicity rating of each target and measuring the significance of changes between them.

The pathogenicity prediction code has been developed by integrating multiple sources of data. The code snippet handles the processing of results obtained from different tools. In order to start, the code expects the directory path to the UniGoPred output as the first command-line argument (sys.argv[1]). An Excel file containing the gene length data is provided as the second command-line argument (sys.argv[2]). The file is read using the pandas function, and the resulting data is stored in a DataFrame. The predictions from the InterProScan tool, which identifies protein domains, are also integrated. The third command-line argument (sys.argv[3]) is expected to be the path to the InterProScan domain prediction result file in a tab-separated format.

1. UniGOPred directory as input: The code segment organizes and categorizes prediction files obtained from UniGoPred. Using os.walk(), it traverses through the specified directory and performs actions on the files. It organizes files based on whether they end with "WILDTYPE.preds" or "ALTERED.preds" by placing both files of each variant into the same corresponding folder. Once the files have been relocated to their appropriate folders based on their names, the code proceeds to handle the prediction files within each category. For files labeled as "ALTERED.preds", the code reads the data, sorts it based on the "GOTERM" column, and renames the score column as "Prediction Score_y". The resulting subset is then saved as a CSV file. The same procedure is applied to files labeled as "WILDTYPE.preds", with the score column renamed as "Prediction Score_x". In summary, this code segment organizes and processes UniGoPred prediction files, generating structured data for subsequent analysis and evaluation in the pathogenicity prediction method.

   Consequently, the code merges two data frames of each target variant based on the "GOTERM" column. It calculates the absolute difference between the "Prediction Score_x" and "Prediction Score_y" columns and adds a new column called "Score_diffr" to store these differences. The data frame is then sorted in descending order based on the "Score_diffr" column. In this way, high activity fluctuations and their associated GO terms are recorded. The amount of variability shows how much the mutated sequence has diverged from the wild type sequence and consequently how much impact it will have on the protein function. In this method the evaluation is based on Pearson correlation, which provides values between 0 and ± 1. Then, if the comparison gives a correlation coefficient (r) near 1.0, there is perfect correlation even after alteration. However, lower and negative values mean that the variant would have a high functional impact. In order to assess the pathogenicity rating of

each target, an r threshold of 0.97 was selected as the cutoff between benign and pathogenic variants. Variants with correlation coefficients lower than this threshold are classified as significant pathogenic mutations. As a result, after manipulating the data, the code performs statistical analysis by computing the Pearson correlation coefficient between the "Prediction Score_x" and "Prediction Score_y" columns. The resulting coefficient is stored in the variable "corr". Finally, the code examines the value of "corr". If it is less than 0.97, the variable "pathogenicity" is assigned the value "P", indicating a pathogenic variant. Conversely, if "corr" is equal to or greater than 0.97, the variable "pathogenicity" is set to "B", indicating a benign variant.

2. UniProt result as input: The second argument of the code is an Excel file from UniProt, providing gene information and their respective sequence lengths. This input is crucial for calculating the location of mutations and determining their distances from the N-terminus. The process of determining the subcellular location and its relationship to the mutations is explained in detail in the previous section. By utilizing the gene information and sequence lengths from this file, the code accurately calculates the mutation locations and their distances, contributing to the subsequent analysis.

3. InterProScan result as input: The code takes an input file from InterProScan, which contains domain information, as its third argument. This file is crucial for associating target variants with their respective domains. The code extracts domain information for the wildtype gene and retrieves relevant information for the mutated form. The code efficiently reads the InterProScan output file, setting the appropriate parameters to handle the file format, and labels the DataFrame columns for ease of data manipulation. During execution, the program iterates through every entry in this dataset, utilizing regular expressions to locate the line that corresponds to the target variant. If a match is found, the code adds the domain information to the wildtype list if it corresponds to the wildtype gene. Similarly, if it corresponds to the mutated form, the domain information is added to the altered list. Similarly, if it corresponds to the mutated form, the domain information is added to the altered list. These domain lists are later attached to the final table, providing a means for the user to compare the domains and identify any changes or loss of important domains.

It's important to note that the information from second and third arguments do not directly impact the classification of the variant as benign or pathogenic. However, they provide an overview of the occurrence place of the variant and conserved domains to assists the user in making a final decision based on these insights. In the end, statistical measurement of fluctuations in protein function, related domain prediction analysis and the targets' location on the sequence are integrated to interpret the results. In this regard, the output table of pathogenicity predictions includes the variant, transcript, gene name, correlation, location, pathogenicity prediction (either "P" or "B"), wildtype domain, and altered domain.

3.3. Performance assessment

3.3.1. Benchmarking with ClinVar dataset

For a comprehensive evaluation and comparison of PMPred with other methods, a benchmark dataset was created. This dataset encompassed a wide range of non-overlapping variations, ensuring a thorough assessment of PMPred's performance. This dataset consisted of 272 variants obtained from ClinVar, encompassing different types of mutations, including in-frame, frameshift, and nonsense mutations. Within this benchmark dataset, there were a total of 138 pathogenic variants, distributed as 79 nonsense mutations, 7 in-frame mutations, and 52 frameshift mutations. These pathogenic variants were known to be associated with disease or deleterious effects. In addition, the benchmark dataset contained 134 benign variants, which served as a control group. Among the benign variants, there were 28 nonsense mutations, 33 in-frame mutations, and 73 frameshift mutations. These variants were considered to have no or minimal impact on protein function and were not associated with disease.

PMPred generates prediction results by employing the statistical assessment method described earlier, which calculates a correlation coefficient (r) for each variant. The target variants were then ranked according to r to evaluate if PMPred could appropriately prioritize a deleterious mutation over a neutral variant. In order to perform a comparison, the pathogenicity rating of the same variants was predicted using the CADD web server and variables were sorted according to scaled PHRED-like scores. In the CADD system, the scaled scores range from 1 to 99, and the recommended cutoff on deleteriousness is between 10 and 20. For our assessment, we chose to use the more conservative threshold of 20. In this regard, scores below the cutoff were allocated to the harmful class, whereas scores above this threshold were defined as indicating neutral variants. VEST-indel is the other program that is participating in this comparison. The provided results are between 0.0 and 1.0 and the defined cutoff is 0.5. Scores above 0.5 represent pathogenic variants and scores below this point are benign. After pathogenicity prediction, for performance measurement, sensitivity, specificity and accuracy were calculated for each model.

Our benchmark dataset encompassed a diverse array of variations in terms of mutation type and pathogenicity, offering a comprehensive representation of the variants commonly encountered in genomic analysis. This ensured that the evaluation and comparison of PMPred with other methods were performed on a robust and diverse set of variants, enhancing the validity and reliability of the findings.

3.3.2. Case study

The case study involved the validation of PMPred against CADD using a specific mutation, namely "c.2397delA" on the CFH gene (NM_000186). The CFH gene, encoding the complement factor H protein, plays a pivotal role in maintaining the delicate balance of the human immune system and is notably associated with Atypical Hemolytic-Uremic Syndrome 1 (Pérez-Caballero et al., 2001). In related disorder, an

overactive immune response can lead to the destruction of red blood cells, platelet activation and kidney injury, often resulting in severe renal failure (Ault, 2000). Additionally, the CFH gene has been implicated in age-related Macular Degeneration (AMD), a leading cause of vision impairment and blindness in older adults. Certain mutations in CFH have been shown to contribute to the development of basal laminar drusen, which are abnormal deposits in the retina associated with AMD (Hageman et al., 2005). Moreover, defects in the CFH gene can lead to Complement Factor H Deficiency, impairing the body's ability to regulate the complement system properly, thereby increasing the susceptibility to various autoimmune diseases. Overall, this multifaceted gene is implicated in several critical physiological processes and alterations within it can lead to a spectrum of damaging health conditions. One such variation is the "c.2397delA" mutation, which can cause harmful effects by introducing an early stop codon. During pathogenicity prediction, PMPred correctly classified this mutation as pathogenic where CADD misclassified it as benign. To further investigate the pathogenicity of this variant, we employed structural analysis.

3.3.2.1. Structure of the wildtype CFH

Initially, we utilized AlphaFold, an advanced AI-based protein structure prediction method, to acquire the structure of the CFH protein. By submitting the CFH protein symbol to the AlphaFold platform, we obtained a precise and comprehensive three-dimensional depiction of the CFH protein's structure, which was saved as a PDB (Protein Data Bank) file. This file contains essential information about the atomic coordinates and other structural attributes, allowing for a detailed analysis and understanding of the CFH protein's architecture. In order to use AlphaFold and obtain the protein structure based on a gene symbol, the following steps were followed:

1. Accessing the AlphaFold platform.
2. Entering the gene symbol or identifier in the provided search box.
3. Initiating the search.
4. Waiting for the prediction process to complete.
5. Retrieving the generated protein structure, usually in a PDB file format.
6. Analyzing and interpreting the structure using bioinformatics tools.

3.3.2.2. Structure of the mutated CFH

In order to generate the structure of the mutated form associated with the "c.2397delA" variant, we employed the SwissModel platform. SwissModel is an extensively utilized homology modeling tool well-known for its ability to predict protein structures by leveraging existing template structures as references. The modeling process using SwissModel commenced by supplying the specific sequence of the mutated form, obtained through the deletion of the nucleotide "A" at cDNA position 2397 in the CFH

gene. The platform then executed a series of intricate computational steps to identify suitable template structures that closely resembled the target protein sequence. These templates serve as structural templates, enabling the prediction of the three-dimensional conformation of the mutated form. During the modeling process, SwissModel employed advanced algorithms to align the target sequence with the selected template structure(s) and generate a refined model that accurately represented the mutated form's structure. The resulting model provided a detailed representation of the spatial arrangement of atoms and residues, providing valuable insights into the potential structural consequences of our target mutation. By utilizing the capabilities of the SwissModel platform, we obtained a reliable and informative model of the mutated form, facilitating further analysis and interpretation of its structural characteristics. The following steps were performed to conduct the analysis:

1. Accessing the SwissModel website or the designated platform where the service is available.

2. Entering the protein sequence of the mutated form, obtained from the "c.2397delA" variant.

3. Selecting appropriate parameters and options, such as template selection and modeling methods.

4. Initiating the modeling process by submitting the sequence and specifying the desired output format.

5. Waiting for the modeling process to complete, which involves sequence alignment, template selection, and model construction.

6. Retrieving the generated structure, usually in a format such as PDB or SWISS-MODEL Repository (SMR).

7. Assessment of the generated structure using various tools and techniques, such as visualization software and structural validation algorithms.

By employing the SwissModel platform, we were able to generate the structure of the mutated form corresponding to the "c.2397delA" variant. This structure provides valuable insights into the potential structural changes and functional implications of the mutation.

3.3.2.3. Structure of the binding ligand

In our study, we conducted an analysis with the aim of exploring the interaction between CFH and its ligand, specifically focusing on identifying the ligand involved in this interaction. In this case, we utilized UniProt, a comprehensive protein database, to determine the binding ligand. The process of detecting the binding ligand involves the following steps:

1. Accessing UniProt: Visiting the UniProt website or utilize UniProt's API (Application Programming Interface) to access the database.

2. Searching for the CFH protein: Entering the CFH protein identifier or gene name in the search box provided by UniProt to retrieve the relevant protein entry.

3. Exploring protein information: Navigating through the protein entry to explore the details and annotations associated with CFH.

4. Identifying the interacting ligand: Looking for information or annotations related to ligand binding or interaction partners. UniProt provides valuable information about protein-ligand interactions, including specific ligands that are known to bind to CFH.

Through the utilization of the resources and annotations provided by UniProt, we can effectively detect and ascertain the ligands that engage in interactions with a given protein. This enables us to acquire significant insights into protein-ligand interactions and their consequential functional implications. In our case, we identified Complement Receptor 3 (CR3/ITGAM) as the interacting ligand for CFH. The ITGAM gene encodes the integrin alpha M protein, which is involved in immune responses and cell adhesion. Mutations in the ITGAM gene have been associated with various autoimmune diseases, such as systemic lupus erythematosus. Indeed, the interaction between CFH and ITGAM is a crucial component of immune regulation and host defense (DiScipio, Daffern, Schraufstätter, & Sriramarao, 1998). CFH, a key regulator of the complement system, binds to CR3, a cell surface receptor found on immune cells like macrophages and neutrophils (Losse, Zipfel, & Józsi, 2010). This interaction plays a pivotal role in immune regulation by modulating the complement cascade, preventing excessive immune responses, and safeguarding healthy tissues from complement-mediated attacks. It also helps balance the inflammatory response, fine-tuning the immune system's ability to eliminate threats while minimizing collateral damage. Additionally, CFH- ITGAM interactions promote phagocytosis, aiding in the clearance of pathogens. Dysregulation of this interaction can lead to immune-related disorders and autoimmune diseases, underscoring its critical importance in immune homeostasis and host defense. This information allows us to gain insights into the molecular interactions between CFH and ITGAM, which can be further explored and analyzed to understand their functional relationship and potential implications in biological processes.

In addition to its interaction with ITGAM, CFH plays a crucial role in binding to C3b, a vital component of the complement cascade. This interaction is instrumental in regulating the complement system, preventing excessive activation, and safeguarding host cells from complement-mediated damage (Noris & Remuzzi, 2013). CFH binding to C3b helps maintain immune tolerance by decoding the complement signal on host cells, thereby averting autoimmune responses and excessive inflammation (Moore, Menon, Cortes, & Ferreira, 2021). Disruptions in CFH's ability to bind to C3b can lead

to dysregulated complement activation and an increased risk of autoimmune and inflammatory diseases. Thus, CFH's dual role in interacting with ITGAM and C3b highlights its significance in orchestrating immune responses and preserving immune homeostasis.

3.3.2.4. Docking

In order to facilitate molecular docking experiments, we took the initiative to acquire and install three essential software tools, namely AutoDock Vina, MGLTools and PyMOL. AutoDock Vina was obtained from its official website, where we securely downloaded the appropriate version compatible with our system. Following the provided installation instructions, we successfully set up AutoDock Vina, which served as the primary docking engine for our simulations. Additionally, we recognized the potential benefits of incorporating MGLTools to complement the functionality of AutoDock Vina. Consequently, we proceeded to obtain MGLTools from its dedicated website and installed it alongside AutoDock Vina. By integrating MGLTools, we gained access to advanced features and utilities for preparation and analysis of protein and ligand structures. This comprehensive toolkit facilitated our visualization of docking results, enabling us to discern and interpret the intricate binding interactions between CFH (both the wildtype and mutated forms) and ITGAM. In order to further enhance our analysis and visualization capabilities, we also utilized PyMOL, a powerful molecular visualization software. With PyMOL, we were able to visualize the docking results in a more detailed and comprehensive manner.

An overview of the key steps involved in the docking process can be found in the appendix A. Through the execution of these steps, the CFH protein is appropriately prepared, the ITGAM ligand is configured, the docking grid is defined, and the AutoDock Vina software conducts the docking simulation. The resultant output file (output.pdbqt) encompasses the predicted binding structures, along with their corresponding binding affinities, offering valuable knowledge on the interaction between CFH and ITGAM. This information aids in the validation of the pathogenicity associated with the "c.2397delA" mutation in the CFH protein. Furthermore, the visualization of the binding using PyMOL provides additional evidence for supporting the significant impact of the "c.2397delA" mutation on the binding mode of CFH. The observed structural changes and alterations in the protein-ligand interactions can highlight the association between this mutation and its potential pathogenicity in the CFH protein.

3.4. Pipeline automation

In order to streamline the analysis of genetic variants and integrate multiple computational tools, a Nextflow pipeline was implemented. Nextflow is a powerful workflow management system that enables the orchestration and execution of complex bioinformatics tasks. The pipeline was designed to automate the annotation, protein domain prediction, and pathogenicity prediction of genetic variants.

The Nextflow pipeline consists of several interconnected processes that perform specific tasks at each stage of the analysis. Each process is executed using relevant tools and scripts, such as ANNOVAR, InterProScan, and UniGOPred, which have been previously described in detail in their respective sections. By leveraging the capabilities of Nextflow, the pipeline ensures efficient parallel execution of the processes, allowing for faster and scalable analysis of large datasets. The pipeline automatically manages the flow of data between the processes, reducing manual intervention and minimizing errors.

3.5. Web interface integration

A user-friendly web interface was implemented using the Flask framework to enhance the accessibility and usability of the analysis pipeline. Flask is a compact web framework in Python that allows for the creation of web applications. The web interface provides an intuitive and interactive platform for users to submit their input files, initiate the analysis pipeline, and view the results. Through the web interface, users can easily upload their input files and configure the analysis parameters. Once the analysis is initiated, users can monitor the progress and receive real-time updates on the status of their analysis.

The Flask code consists of several routes that handle different web pages and actions. The home route ('/') renders the homepage template, while the about route ('/about') renders the about page template. The main functionality is implemented in the 'home' route, which handles the file upload and execution of the analysis pipeline. Upon file submission, the uploaded files are securely saved to the designated upload folder. The analysis pipeline is then triggered by calling Nextflow through subprocess commands, passing the uploaded file as input. The pipeline execution and subsequent data processing are performed in the background. Once the pipeline execution is completed, the results are read from the output file ('PMPred.csv') and displayed in an HTML table format using the Pandas library. The results are rendered on the 'content' template, providing users with a comprehensive view of the variant table.

The Flask application is deployed and hosted using the app.run() method, allowing users to access the web interface through their browser. Overall, the integration of the Flask framework and HTML templates provides an interactive web interface for users to conveniently submit their input files, execute the analysis pipeline, and visualize the results.

3.6. Sending results to the user

In our code, the functionality of sending email relies on the integration of the Mailjet API, a powerful email service provider. This API enables us to effortlessly deliver the 'PMPred.csv' file, which contains the analysis results, directly to the user's email address. Once the analysis pipeline completes and generates the 'PMPred.csv' file, it is imported into a Pandas DataFrame for further processing. The DataFrame holds the

structured data obtained from the analysis, facilitating easy manipulation and extraction of relevant information. In order to prepare the 'PMPred.csv' file for email attachment, we convert the DataFrame into a CSV string using the 'to_csv()' method provided by the Pandas library. This string representation of the data is then encoded using base64 encoding, ensuring compatibility and secure transmission within the email.

The Mailjet API enables us to construct a comprehensive email message. It includes the sender's email address and the recipient's email address, which is dynamically provided by the user. The subject of the email is set as "PMPred Results," indicating the nature of the attached file. The HTML content of the email message is carefully crafted to provide clear instructions and relevant information to the recipient. In our case, we include a simple message stating "Results are provided in csv format", highlighting the format of the attached 'PMPred.csv' file. To attach the 'PMPred.csv' file, we utilize the Mailjet API's attachment functionality. The file is included as an attachment, with the base64-encoded content serving as the attachment's data. Finally, the email is sent using the Mailjet API by making a POST request with the constructed email data. By incorporating this functionality, the system can automatically send the results to the user via email, providing convenient access to the 'PMPred.csv' file. This ensures that users can easily receive and review the analysis results, facilitating further investigation and interpretation of the variant table.

3.7. Availability

PMPred is freely accessible via a Web service at https://pmpred.kansil.org/. The server is available for nonsense, frameshift and in-frame variants, provided in VCF format. Prediction results are provided as a sorted CSV file, from most destructive variants to neutral ones. Our Flask-based Web Server allows users to both visualize and download the outcomes for additional analysis. The PMPred code is available at https://github.com/kansil/PMPred (Figure 8).

The PMPred website serves as a platform for accessing and utilizing the prediction tool. The interface allows users to easily upload their VCF files containing nonsense or indel variants. The website provides a straightforward submission process, guiding users through the necessary steps to initiate the analysis pipeline. Once the analysis is complete, the website presents the results in a sorted CSV file format, arranged from the most deleterious variants to the neutral ones. This organized presentation enables users to quickly identify and assess the pathogenic potential of the variants.

Furthermore, on our GitHub page, users can find the complete source code of PMPred, including the necessary scripts, modules, and dependencies. This open-source approach not only enables users to understand the inner workings of our prediction tool but also encourages them to enhance its functionalities, customize it for their specific needs, and contribute back to the community by sharing their modifications and improvements.

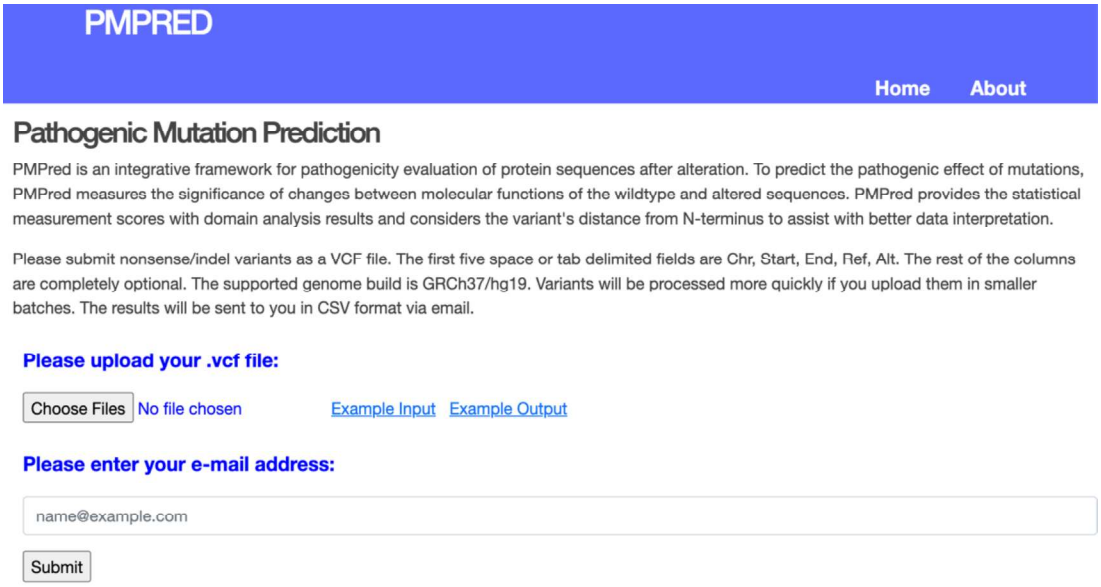

Figure 8: PMPred Web Server and Workflow. The figure showcases the PMPred web interface, facilitating the analysis of nonsense, frameshift, and in-frame variants. The results are presented in a sorted CSV format, allowing users to efficiently evaluate the pathogenic potential of the variants.

CHAPTER 4

RESULTS AND DISCUSSION

We present a comprehensive performance evaluation of PMPred, comparing it with other existing methods for pathogenicity prediction. The evaluation encompasses various methodologies, including ROC curve analysis, performance metrics tables, molecular docking simulations and case study with clinical exome sequencing data.

Each evaluation approach provides unique insights into the predictive capabilities of PMPred and its counterparts. The ROC curve analysis offers a visual representation of the trade-off between sensitivity and specificity, allowing us to assess the overall discriminative power of the methods. Performance metrics tables present detailed quantitative assessments, such as accuracy, precision, recall, and F1-score, enabling a comprehensive comparison of the methods' performance across different categories of variants. Additionally, the docking simulations reveals the structural implications of variants on protein-ligand interactions, providing crucial evidence for the functional impact of the mutations. Also, the case study involves the analysis of clinical exome sequencing data from patients with unknown pathogenic variants related to specific diseases. We identified and confirmed the deleterious variants associated with the related disease cause, employing PMPred. By organizing the results into distinct sections based on the evaluation methodologies, we aim to present a comprehensive and coherent analysis of PMPred's performance and highlight its effectiveness as a pathogenicity prediction tool.

Furthermore, to gain deeper insights into the predictive features and their impact on PMPred's performance, we conducted an analysis using decision tree models. By employing decision tree algorithms, we identified the most influential features contributing to pathogenicity prediction. This analysis allowed us to discern the relative importance of individual features, providing valuable information for further model optimization and enhancement. Moreover, the decision tree analysis enabled us to detect feature interactions and uncover possible collaborations between different characteristics. Understanding how various features interact to influence the prediction outcome can aid in refining the model and tailoring it to specific variant types or disease contexts.

4.1. Performance evaluation

4.1.1. Multi-method benchmark set

We conducted a comprehensive evaluation and comparison of PMPred with other prediction methods using the benchmark dataset we created. This dataset consisted of 272 non-overlapping variants obtained from ClinVar, encompassing various mutation types, such as in-frame, frameshift, and nonsense mutations.

To assess PMPred's performance, we conducted evaluation on the test set in two distinct ways. First, we focused our estimation solely on nonsense mutations, allowing us to specifically evaluate PMPred's ability to predict pathogenicity in this specific mutation type. Next, we conducted the evaluation on the entire dataset, including all in-frame, frameshift, and nonsense mutations. This broader assessment allowed us to gain insights into PMPred's overall performance across different mutation types, providing a comprehensive picture of its predictive capabilities. By measuring the performances of different prediction methods on this diverse test set, we identified the strengths and weaknesses of each approach.

4.1.1.1. Precision and sensitivity

The validation results of the three methods are shown in Table 1. For nonsense variants, PMPred and CADD, with a sensitivity of 1.0 each, are found to be perfectly sensitive, with no false negative nonsense variants. On the other hand, VEST-indel was optimized to reject variants with no disease-causing effect accurately. Therefore, VEST-indel achieved the highest precision/specificity. However, PMPred achieved almost identical precision, and CADD’s precision was moderately lower. From the point of view of F1 scores, all three methods performed well, with PMPred having the highest F1. PMPred and VEST-indel attained nearly equal AUC (Area Under the Curve), and CADD’s AUC was lower. In terms of the MCC (Matthews Correlation Coefficient) results, PMPred was evaluated as having a significantly better prediction performance when compared to CADD and VEST-indel. In addition, PMPred outperformed the other approaches in terms of its misclassification rate (final column in the table). Overall, during the first evaluation phase, PMPred significantly outperformed VEST-indel and CADD in sensitivity and precision.

The second scenario evaluates the methods’ performances in the classification of in-frame and frameshift variants along with nonsense variations. The results indicate that the sensitivity of PMPred suffered slightly, with precision and overall misclassification rate decreasing more significantly. CADD's performance metrics changed similarly, with slightly lowered precision and sensitivity being lowered considerably. For VEST-indel, accuracy remained mostly the same, while sensitivity dropped to 0.68 and precision dropped slightly. In comparison to other methods, PMPred represents a good balance of sensitivity and precision, despite the reduction in performance when in-frame and frameshift variants are added. The findings confirm

that PMPred is a reliable prediction method; it retains high specificity while capturing the majority of functionally relevant mutations.

Table 1: Prediction accuracies of PMPred, CADD, and VEST-indel, in predicting (a) exclusively nonsense mutations and (b) all mutations in the test set.

	Multi-method benchmark					
	Sensitivity	Precision	AUROC	F1 score	MCC	Accuracy
Nonsense, Frameshift, In-frame						
PMPred	0.949	0.598	0.623	**0.734**	0.328	0.639
CADD	**0.964**	0.573	0.612	0.719	0.318	0.618
VEST-indel	0.688	**0.674**	**0.671**	0.681	**0.343**	**0.672**
Nonsense only						
PMPred	**1.000**	0.782	**0.607**	**0.878**	**0.409**	**0.794**
CADD	**1.000**	0.705	0.530	0.827	0.207	0.710
VEST-indel	0.722	**0.792**	0.593	0.755	0.174	0.654

4.1.1.2. ROC curve analysis

To visualize the performance of the prediction methods at different thresholds, we generated ROC curves, as represented in Figure 9. The ROC curves graphically depict the true positive rate (sensitivity) against the false positive rate (1-specificity) at different classification thresholds. These curves provide a comprehensive overview of the trade-off between sensitivity and specificity, allowing for the comparison of prediction methods based on their discriminatory power across a range of threshold values. A well-performing method would exhibit a higher true positive rate and a lower false positive rate, resulting in a ROC curve that closely approaches the upper-left corner of the plot. This indicates that the method achieves a high sensitivity while keeping the false positive rate at a minimum. By analyzing the ROC curves, we determined the optimal threshold as 0.97 where PMPred achieves an optimal balance between sensitivity and specificity. Consequently, the area under the ROC curve (AUC) becomes a valuable metric for gauging overall performance, where a higher AUC signifies better predictive ability.

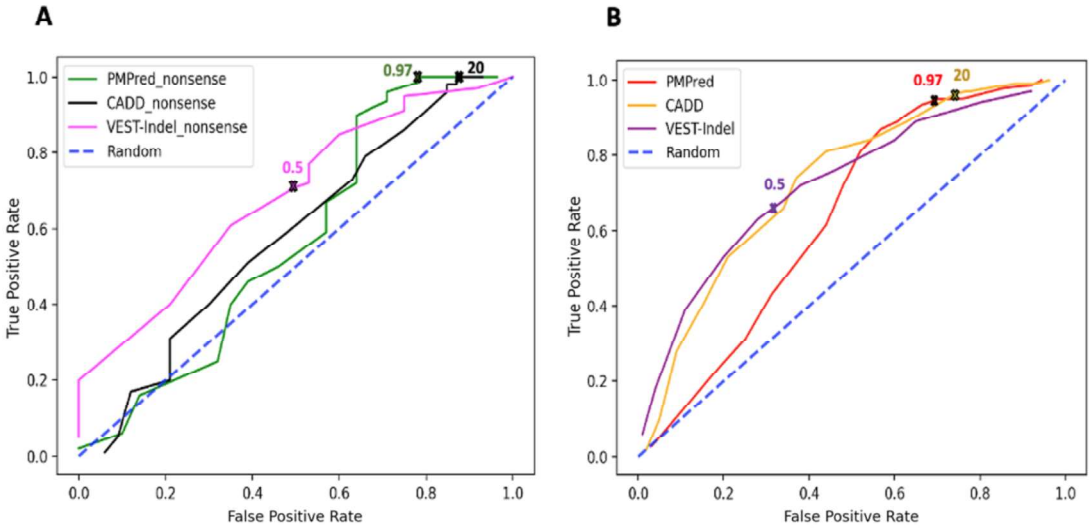

Figure 9: TP vs. FP rate at different classification thresholds. A) Performance evaluation based on predicting nonsense variants. B) Comparison involves all types including nonsense, frameshift and in-frame variants. All methods are marked with the optimal threshold for getting the best prediction results, where sensitivity is the highest and specificity is the lowest.

4.1.2. Docking with CADD's unrecognized pathogenic mutation

In addition to our approach focused on analyzing nonsense and frameshift variants, we conducted in-silico validation to further investigate the pathogenicity of the identified variants. Specifically, we performed molecular docking simulations to evaluate the binding interactions between the wildtype form and mutated form of a target variant with its corresponding binding ligand. The CFH gene was selected for molecular docking simulations due to the difference in its variant classification between CADD (benign) and PMPred (pathogenic). This choice aimed to further investigate and provide insights into the potential pathogenicity of the variant based on its binding interactions.

The differences observed in the docking structures between the wildtype and mutated forms of CFH and their corresponding binding ligands reflect the impact of the mutation on the protein-ligand interactions. These structural variations arise due to changes in the amino acid sequence caused by the mutation, leading to alterations in the three-dimensional conformation and properties of the protein. Furthermore, the changes in the three-dimensional conformation and properties of the protein due to the mutation can potentially disrupt critical functional regions or protein-protein interaction sites, thereby influencing downstream cellular processes (Figure 10).

For the wildtype form, the docking results revealed several modes of binding with varying affinities. The top-ranked mode exhibited an affinity of -4.0 kcal/mol, suggesting a strong binding interaction. The subsequent modes showed slightly lower affinities, with increasing distances from the best mode. These results indicate that the

wildtype form maintains favorable binding characteristics with the binding ligand. In contrast, the docking results for the mutated form demonstrated alterations in the binding affinities and distances from the best mode compared to the wildtype form. The top-ranked mode for the mutated form exhibited an affinity of -3.5 kcal/mol, indicating a slightly weaker binding compared to the wildtype. The subsequent modes also displayed varied affinities and distances, reflecting the impact of the mutation on the binding interactions.

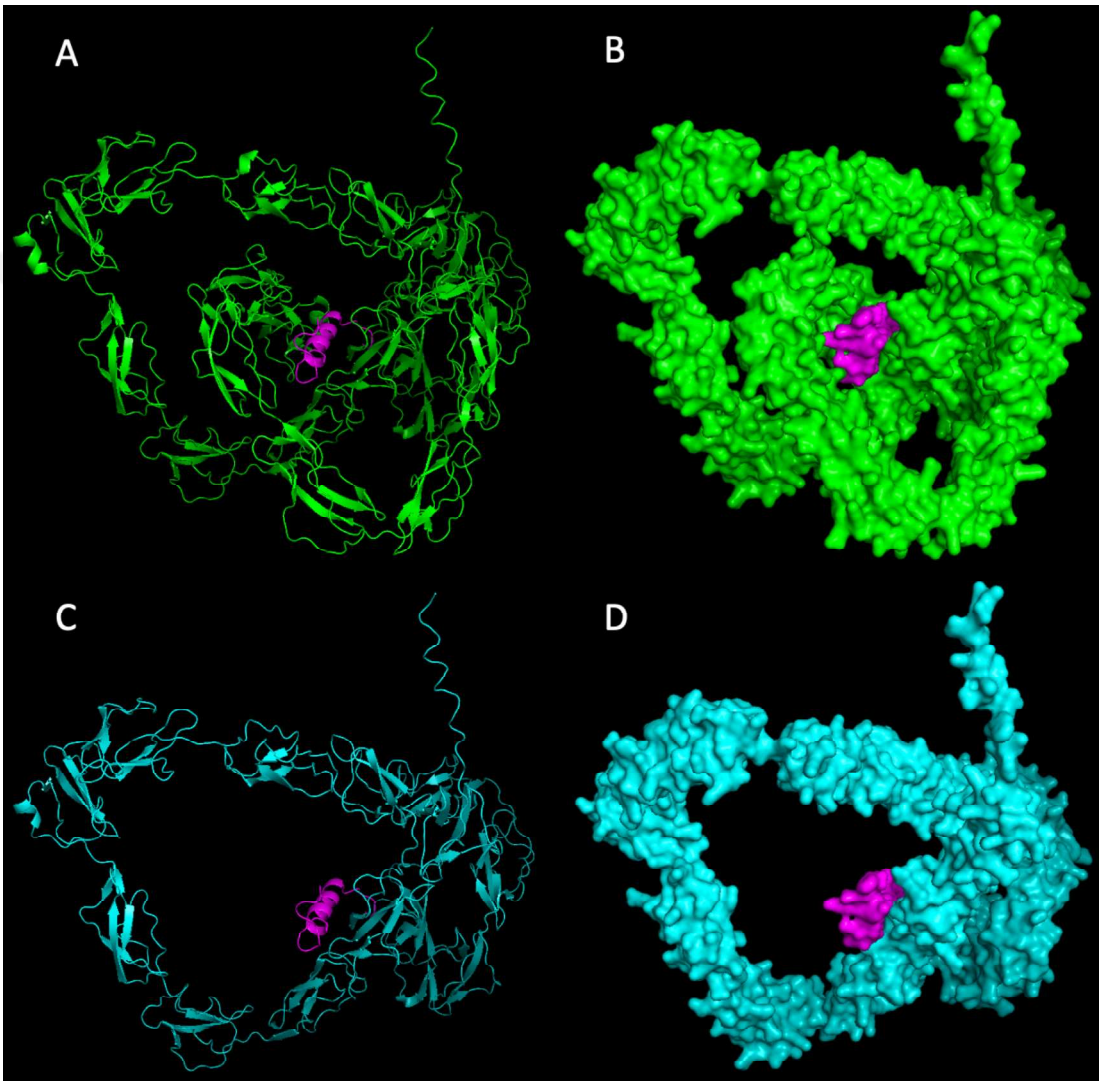

Figure 10: The docking of CFH to its ligand ITGAM. (A), (B): Wild-type CFH (in green) and ligand (in magenta), cartoon and space-filling representations, respectively (C), (D): Mutated CFH (in cyan) and ligand (in magenta), cartoon and space-filling representations, respectively, illustrating the altered molecular surface and binding regions caused by the mutation.

Based on these docking results, it is evident that the mutated form shows a noticeable deviation in the binding characteristics compared to the wildtype form. This disruption in the binding interactions strongly suggests a functional impact of the mutation, potentially leading to an altered molecular function. Therefore, in conjunction with the

findings from our primary approach, which highlighted the functional measurement feature as a reliable discriminator for pathogenic variants, the docking results provide further support for classifying this mutation as pathogenic.

4.1.3. Case study with clinical exome ssequencing data

To investigate the performance of PMPred, data from a de-identified patient diagnosed with type I diabetes was analyzed anonymously. The whole blood genomic DNA of the case had been extracted and sent for clinical exome sequencing (CES). The results showed a total of 11,020 variants called from CES data, of which four likely pathogenic variants remained after ACMG filtering guidelines were applied. Unfortunately, based on the phenotype information from Online Mendelian Inheritance in Man (OMIM), none of those four variants were associated with type I diabetes. Following the failure in finding a causative variant, Human Phenotype Ontology (HPO)(Köhler et al., 2021) filtering criteria were applied to find related genes to the target disease based on literature reviews. According to this phenotype matching algorithm, a stop gained substitution (ENST00000356113.6 c.1030C>T) with autosomal dominant inheritance was found on a diabetes-related gene named LIPC. No clinical data were found in ClinVar for this variant, whereas some prediction methods, such as MutationTaster and CADD, reported it as a deleterious mutation. Consequently, we predicted the pathogenic impact of this variant using PMPred, which classified the variant as damaging with a correlation value of 0.64 and PLAT/LH2 domain loss. Note here that a lower correlation means that the variant is more likely to be deleterious, and variants with correlations below 0.9 are considered as such.

Another de-identified patient had recurrent fractures and joint dislocation history. Like the prior example, mutations that ACMG criteria classify as harmful variations could not explain patient symptoms. For this reason, other discovered variants with unknown clinical significance were examined to find the cause of the disease. As a result of this investigation, the specialists reported a stop-gained mutation on the SCN9A gene as the responsible alteration for the patient's complaints. According to ClinVar data, W1150* (ENST00000409672.1 c.3449G>A) is a variant without proven pathogenic impact. PMPred, this paper's approach, on the other hand, classifies the target variant as pathogenic and thus provides disease-based data interpretation.

4.2. Analysis of the impact of prediction features on PMPred performance using the decision tree approach

We also tested the effect of different features on the prediction performance of the PMPred pipeline. These features are: (1) correlation: the agreement between the (UniGOPred) molecular function annotation sets for the wildtype and mutated sequences; (2) consequence: effect of mutation in a protein coding region (nonsense, frameshift, etc.); (3) distance: distance of a mutation from the protein start codon; and (4) domain: domain-level losses in the polypeptide chain after mutation. Figure 11 shows the decision tree results for the case when all features were involved.

The correlation of molecular function assignments (described in sec 3.1.4) is found to be the decision tree's master node, and contributes the greatest amount of accuracy. In this algorithm, the max-depth parameter is used to determine the ideal number of splits and the optimum accuracy outcomes. According to the optimal classification, the feature selection process using four features for the combined set of variants produced sensitivity and precision values of 0.90 and 0.54, respectively.

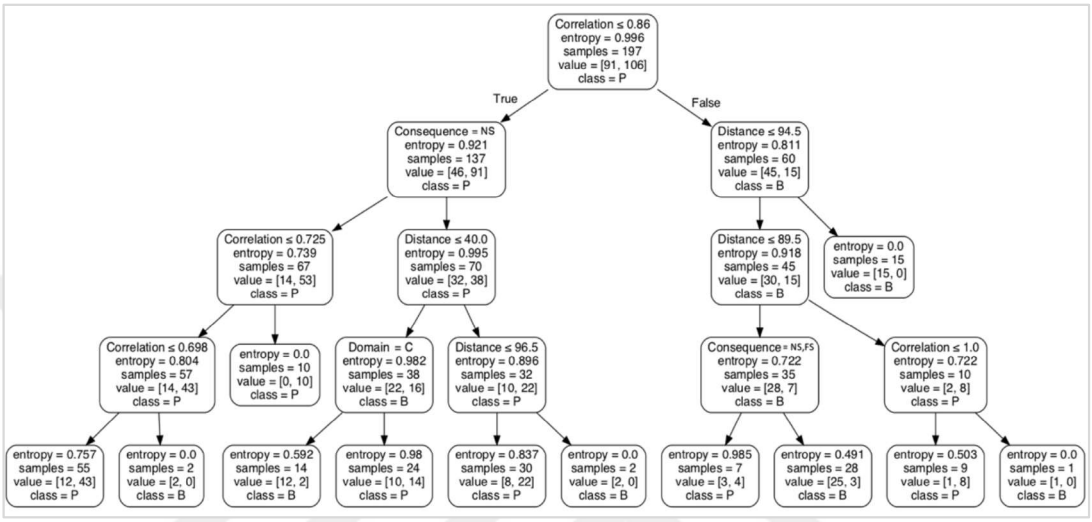

Figure 11: Decision tree: 197 samples of benign (Class = B) and pathogenic (Class = P) variants. NS stands for Non-sense, FS for Frameshift and C for Conserved. The decision tree was built using Scikit-Learn(Pedregosa et al., 2011) using the Entropy Criterion. Starting with 197 variants at the root node, among 4 features, correlation was identified as the most distinguishing feature in classifying the mutations. In the second split, after correlation, the mutation's consequence and its distance from the start codon become important. Lower levels of the tree are mostly homogenous, indicating that the bulk of the classification is done by the first two levels. Clearly, the major factor is correlation; no other factor has as much effect in predicting the pathogenicity of variants. The leaf nodes contain the decision or outcome of the decision tree. The decision tree depth was bounded at 4 as this was found to be the optimal compromise between accuracy and model simplicity, using 25/75 holdout.

4.3. Evaluation of PMPred on clinical exome and gnomAD data and statistical method selection

In the initial stages of our work, we evaluated the efficiency of PMPred by testing it on a set of clinical exome data from patients diagnosed with muscle dystrophy. This dataset provided us with known pathogenic variants and their corresponding disease-causing mechanisms, allowing us to validate the performance of PMPred against well-established clinical evidence. Our analysis encompassed various types of causative variants, including missense and nonsense mutations.

During this evaluation, we observed that PMPred performed well in accurately predicting the pathogenicity of indels, frameshift and in-frame variants, demonstrating its potential as a powerful tool for variants of these types. However, when applied to

missense variants, we encountered some limitations, leading us to shift our focus primarily to frameshift and in-frame mutations in our subsequent analyses.

To determine the most suitable statistical method for PMPred, we conducted comparative tests using both t-test and Pearson correlation. While the t-test failed to reveal a clear pattern for distinguishing pathogenic and benign variants, resulting in inconsistent and random outcomes, the application of Pearson correlation yielded robust and trustworthy results. The calculated r-values enabled us to confidently differentiate between pathogenic and benign variants, thereby confirming the suitability of Pearson correlation as the statistical measurement for PMPred.

In addition to testing the efficiency of PMPred on a set of clinical exome data from patients with muscular dystrophy, we also explored its applicability to a broader dataset from gnomAD (Karczewski et al., 2020). Prior to applying the prediction, we diligently examined the data distribution using normality tests. It became evident that the data was not normally distributed, leading us to choose non-parametric tests like the Wilcoxon rank-sum test. However, despite our efforts, both the Wilcoxon test and t-test analysis did not yield the desired results in effectively distinguishing pathogenic variants from benign ones. Consequently, we sought an alternative approach and found that using Pearson correlation proved to be the most successful method for predicting pathogenicity. The correlation analysis effectively quantified the relationships between variant features and pathogenic outcomes, enabling us to confidently differentiate between pathogenic and benign variants.

It is essential to highlight that we excluded the clinical exome data from our final analysis due to privacy and ethical concerns, as they contained personal patient information. Furthermore, while gnomAD provided a diverse set of variants for evaluation, it lacked specific pathogenicity annotations for the involved variants. Instead, to ensure a more accurate and comprehensive benchmarking of PMPred, we turned to utilize the publicly available ClinVar dataset, which contains well-curated clinical significance information for each variant. This decision allowed us to conduct a rigorous evaluation and validation of our method's performance, providing reliable results for our analysis.

4.4. Precision and reliability of predictions

In this study, we developed the PMPred method for automatically evaluating the disease-causing potential of variants. PMPred demonstrates remarkable precision and reliability in predicting the pathogenicity of variants, particularly owing to its utilization of the function measurement feature. The currently available functional information about gene products is represented in GO terms. Positive annotations link genes to GO terms and show that the gene products generally take part in the functions specified by the respective GO terms. Following a mutation, the decreased value of related GO terms reflects that there is a defect in protein function and the gene product may no longer carry out the function specified by this term. The results suggest that

the function measurement feature by itself makes a precise and reliable prediction for the intended variant set. By quantitatively assessing the functional alterations caused by mutations, PMPred can effectively distinguish between pathogenic and benign variants with high accuracy. Additionally, domain analysis results and the mutation's location in the sequence are reported, in order to further assist the user in drawing their conclusions.

4.5. Importance of functional measurement feature in accurate categorization of nonsense and frameshift variants

Our work focused on nonsense and frameshift variants as they require a different approach for accurate categorization(citation). Such an approach needs features that take the entire region into account rather than just the immediate area around the target site. Measurement of molecular functions in wildtype and mutated sequences, as the main feature of our method, works efficiently by considering perturbation of the whole sequence. The applied decision tree analysis confirms that the most accurate discriminator is indeed this functional measurement feature. Although other examined features have been found to be useful in decreasing FP, none of these improve the sensitivity toward the actual disease-causing variations. The outcome indicates that only the molecular function feature is sufficient to make a sensitive distinction between pathogenic nonsense, frameshift and in-frame indels. With a threshold of 0.97, this characteristic offers the most accurate differentiation of deleterious variants. Therefore, the prediction can be made simply with just this one feature. Nevertheless, the outcomes of additional features are provided to the user, for final decision-making. As disease characteristics can vary from patient to patient, clinicians can take into account any or all results for a final conclusion depending on their own criteria.

4.6. Validation through docking and literature comparison

In addition to our primary approach focused on analyzing nonsense and frameshift variants, we conducted in-silico validation to further investigate the pathogenicity of the identified variants. Specifically, we performed molecular docking simulations to evaluate the binding interactions between the wildtype form and mutated form of the CFH gene as target variant with its corresponding binding ligand. The CFH gene was selected for molecular docking due to the discrepancy in its variant classification between CADD (benign) and PMPred (pathogenic). This choice aimed to further investigate and provide insights into the potential pathogenicity of the variant based on its binding interactions. The wildtype form exhibited a top-ranked mode with an affinity of -4.0 kcal/mol, indicating a strong binding interaction. However, the mutated form showed alterations in binding affinities, with the top-ranked mode exhibiting an affinity of -3.5 kcal/mol, representing a decrease of approximately 12.5% compared to the wildtype. The docking results revealed distinct patterns for the wildtype and mutated forms, indicating differences in their binding characteristics. The results suggest that the mutation has a noticeable impact on the binding interactions. These

findings, combined with our primary approach that highlighted the reliability of the functional measurement feature as a discriminator, provide strong support for the efficacy of our prediction method, PMPred. By applying docking as a validation tool, we were able to assess the impact of the mutation on the binding interactions and strengthen the case for the accuracy of our classification.

To comprehend the pathogenicity of the identified variant, it's crucial to explore the significance of CFH domains, specifically their role in binding to C3b and regulating the complement cascade. CFH is comprised of 20 complement control protein (CCP) modules, with distinct functions. The N-terminal modules (CCP1-4) are pivotal for binding to C3b, acting as cofactors for the CFI-mediated cleavage into the inactive iC3b (Kavanagh, Goodship, & Richards, 2013). Conversely, the C-terminal modules (CCP19 and 20) facilitate binding to C3b and glycosaminoglycans on host cells, thereby mediating cell surface protection. Notably, individuals with atypical Hemolytic-Uremic Syndrome (aHUS) often develop autoantibodies that target the C-terminus of CFH, interfering with the interaction of CFH with C3b (Raina et al., 2022). This disruption results in dysregulation and overactivity of the complement pathway. Importantly, genetic variants associated with aHUS frequently cluster in CCPs 19 and 20 of CFH, though they can manifest throughout the CFH molecule (Jokiranta et al., 2006). In this context, the examined variant on CFH, which locates within domain 14, leads to the loss of these crucial C-terminal domains and several others, severely compromising CFH's ability to bind to C3b and maintain complement regulation. Aligning our findings with existing research in the field emphasizes the pathogenicity of the studied variant due to the loss of essential domains, thereby contributing to the dysregulation of the complement pathway.

The identification of specific molecular GO terms, such as 'GO:0019838' (growth factor binding), 'GO:0005504' (FAD binding), 'GO:0031625' (ubiquitin protein ligase binding), 'GO:0016278' (lysosomal transport), and 'GO:0050839' (cell adhesion molecule binding), as highly fluctuated following the occurrence of 'c.2397delA' mutation on CFH gene, offers critical insights into the underlying molecular mechanisms of CFH-related diseases, notably aHUS. These terms are directly related to vital cellular functions, including cell growth regulation, enzymatic activities, protein degradation, intracellular trafficking, and cell adhesion. Perturbations in these processes have been extensively associated with the dysregulation of the complement system, a key factor in the pathogenesis of aHUS. Particularly, disruptions in lysosomal transport ('GO:0016278') can hinder the clearance of complement activation products, contributing to the overactivity of the complement pathway. Furthermore, the altered binding affinities, such as those related to 'GO:0005504' (FAD binding) and 'GO:0050839' (cell adhesion molecule binding), can affect the interaction between CFH and complement components like C3b, crucial for the regulation of complement activation. The variant's impact on these GO terms indicates a significant molecular disturbance, further reinforcing its pathogenicity and its potential contribution to the development of aHUS. Alignment of these findings with previous research highlights the importance of these functional categories in the context of CFH-related diseases

and provides additional evidence for the deleterious consequences of the 'c.2397delA' mutation.

4.7. Normality of benchmark set

In addition to the decision tree analysis, we performed a normality test on the input data to assess the distribution of the variant features (Figure 12). Understanding the data distribution is crucial for ensuring the appropriate application of statistical methods and other algorithms. By examining the normality of the input data, we were able to determine the most suitable approaches for subsequent analyses and verify the assumptions made during the performance evaluation.

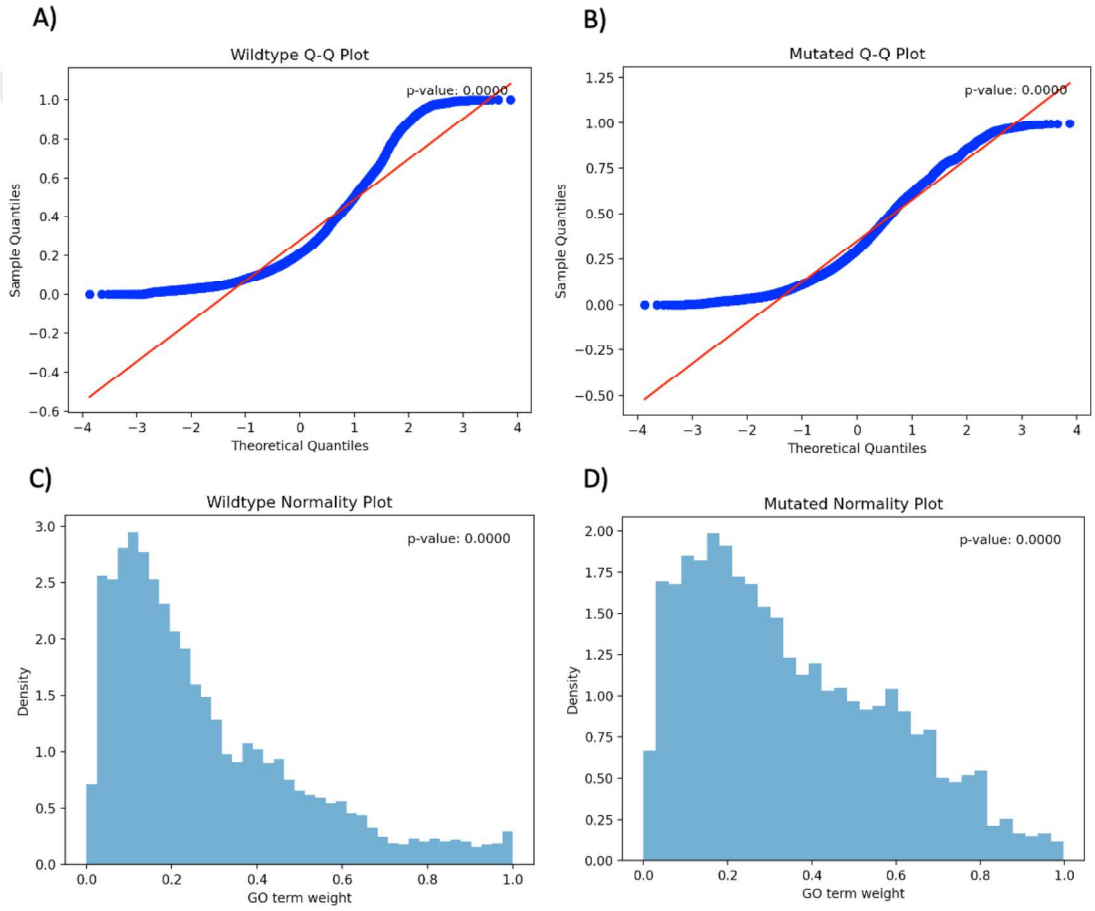

Figure 12: A) Wildtype Q-Q plot: Compares theoretical and observed quantiles for GO term weights in wildtype variants. B) Mutated Q-Q plot: Compares theoretical and observed quantiles for GO term weights in mutated variants. C) Wildtype normality plot: Histogram showing the distribution of GO term weights in wildtype variants. D) Mutated normality plot: Histogram displaying the distribution of GO term weights in mutated variants.

We evaluated the distribution of the variant features in our dataset, which consisted of 272 genes, each with its own CSV file containing 50 rows of GO term weights.

Understanding the data distribution is essential as it ensures the appropriate application of statistical methods and algorithms during subsequent analyses and validation. To assess normality, we employed two important visual tools: the Quantile-Quantile (Q-Q) plot and the normality plot. The Q-Q plot visually compares the quantiles of our data against those of a theoretical normal distribution. The normality plot, on the other hand, provides a histogram of our data overlaid with the fitted normal distribution curve.

These plots provide valuable insights into the distribution of the weights of GO terms for both the wildtype and mutated variants in our dataset. When analyzing the QQ plots for both sets, we observed that the data points deviated from the diagonal line, indicating non-normality. Furthermore, the histograms of both datasets showed a right-skewed distribution, further confirming the non-normality of the data. The results of the Shapiro-Wilk normality test further supported our observations, as it returned extremely low p-values ($p = 0.0$) for both wildtype and mutated data. This indicates that the data does not follow a normal distribution and underscores the importance of applying appropriate statistical methods.

To measure the statistical significance and identify significant changes between wildtype and mutated data, we chose Pearson correlation as our statistical measurement. The rationale behind selecting Pearson correlation for the statistical measurement in the normality test is that it is more influenced by outliers. Being sensitive to outliers, can help identify significant changes in the mutated data that may be associated with pathogenic impact. When comparing the wildtype and mutated weights of GO terms for target variants, the presence of outliers can indicate potentially important variations in the data. By using Pearson correlation, we can detect these extreme data points and assess their potential impact on the overall pattern of the data. If certain GO terms exhibit significant changes in their weights due to mutations, these alterations could be biologically meaningful and relevant to the pathogenicity of the variants.

4.8. Limitations

PMPred currently targets coding regions of the genome due to the availability of functional annotations by ontology providers. The decision to target coding regions is supported by the fact that these regions play a fundamental role in protein synthesis, and variations in this area are more likely to have direct effects on the protein's function. While focusing on coding regions represents a powerful starting point, it is important to acknowledge that genomic variations in other regions, such as promoters and splice sites, can also contribute to disease pathogenesis. As part of our future work, we plan to explore the extension of PMPred to encompass these regions, provided that appropriate functional annotations become available.

Our method encountered a limitation in predicting the pathogenic effect of mutations occurring at the start codon of a sequence. This specific challenge arises from the fact

that the tools we utilized for variant annotation, does not provide a sequence for mutated form to compare with the wild-type form of the target at the start codon. As a result, the lack of this essential sequence information suppresses the accurate assessment of variants located at the beginning of a gene. The start codon is a critical element in the initiation of protein synthesis, and variations in this region can have significant implications for the functionality of the resulting protein. Unfortunately, without the necessary sequence data, our method is unable to evaluate the impact of mutations at this specific site. Consequently, the prediction for variants occurring in this region may be limited or not available through the current version of PMPred, which highlights an area for potential improvement. As part of our future work, we plan to explore alternative approaches and incorporate additional sequence information to overcome this limitation.

The first version of PMPred is intended to prioritize the evaluation of frameshift, in-frame, and nonsense variants, as these types of mutations are well-established contributors to disease pathogenicity. However, we acknowledge that missense mutations represent another crucial class of variants that can also play a significant role in disease development. To broaden the scope of PMPred and enhance its predictive capabilities, we plan to incorporate additional traits to effectively categorize and predict the pathogenicity of missense mutations. Missense mutations involve the substitution of a single amino acid in the protein sequence, making them particularly challenging to assess solely based on functional changes. To address this complexity, we intend to leverage structure-based predictors (Zhang et al., 2012), which utilize three-dimensional structural information of proteins to distinguish between harmful and presumably benign missense mutations. By considering the structural context of these mutations, we can gain valuable insights into how they might affect protein folding, stability, and interactions with other molecules. By integrating structure-based predictors into PMPred, we aim to complement the existing function measurement feature and provide a more comprehensive framework for accurately predicting the pathogenicity of missense mutations. This expansion will further bolster the precision and reliability of PMPred, enabling it to cover a broader range of genetic variants and offer a valuable tool for the assessment of disease-causing potential in different mutation types.

It is worth noting that, in the current version of PMPred, GRCh37 is the reference genome used for data annotation. This choice was made to ensure compatibility with existing datasets and annotations available in the GRCh37 assembly. However, as advancements in genomics research continue to unfold, newer genome releases such as GRCh38, offer improved and updated genomic information. Recognizing the importance of staying current with the latest genomic references, we are committed to updating PMPred to support newer genome releases and enhance its utility for variant annotation. By adapting to newer genome releases, PMPred will effectively align with the most up-to-date genomic information, ensuring that users can confidently apply it to their variant analysis with the latest knowledge.

Moreover, the potential applicability of PMPred is not limited to the human genome alone. While our initial focus has been on human genetic variants, the underlying principles and methodology of PMPred can be extended to explore phenotype and genotype associations in various other organisms as well.

CHAPTER 5

CONCLUSION

5.1. Research objective

Genetic mutations play a vital role in human health and disease. These alterations in the DNA sequence can lead to various changes in gene products, such as proteins, enzymes, and other functional molecules. Depending on their location and type, mutations can have profound effects on the structure and function of these gene products. Some mutations may introduce amino acid changes, resulting in altered protein folding or catalytic activity, while others may disrupt crucial regulatory elements, leading to abnormal gene expression. Identifying and characterizing these mutations are crucial for clinical diagnosis and treatment.

Determining the pathogenicity of genetic variants, especially frameshift indels and nonsense mutations, presents distinct challenges. While missense variants have been extensively studied, the functional impact of frameshift and in-frame variants is less explored, leading to limited understanding of their molecular effects and clinical implications. The impact of nonsense changes and indels is particularly crucial, as they can lead to drastic changes in protein function, potentially causing severe genetic disorders. Reliable assessment of their pathogenicity is essential for accurate diagnosis and personalized treatment plans. However, existing prediction methods often struggle to precisely distinguish pathogenic from benign variants in this context, resulting in potential misclassifications and suboptimal clinical decision-making.

Throughout this research, our main objective was to develop and evaluate PMPred, a novel method designed to automatically assess the disease-causing potential of genetic variants. The motivation behind this work arose from the urgent need for accurate and efficient tools that can aid in the interpretation of variant pathogenicity, especially in the context of indels and nonsense variants. In summary, decoding the consequences of genetic mutations is essential for comprehending disease mechanisms, guiding clinical decisions, and advancing precision medicine. By developing methods such as PMPred for evaluating variant pathogenicity, we try to bridge the gap between genomic data and clinical insights, ultimately improving patient care and health outcomes.

5.2. Contribution

PMPred was considered to address this challenge by employing the valuable functional information encoded in Gene Ontology (GO) terms and translating it into a predictive model. The integration of functional annotations from GO terms was crucial to the success of PMPred. By utilizing the annotations that link genes to specific functions, we were able to infer the functional impact of mutations on gene products. In this regard, for each variant, we first extracted the GO terms associated with the respective gene products and assigned weights to these terms. These weights represented the functional importance of each GO term in the wildtype sequence. Subsequently, we repeated this process for the mutated sequence to obtain the GO term weights in the post-mutation state. The next step involved a statistical analysis of the changes in GO term weights between the wildtype and mutated sequences. By comparing the functional impact before and after the mutation, we were able to discern potential defects in protein function caused by variants, thus contributing to the prediction of pathogenicity.

In addition to utilizing Gene Ontology (GO) terms for pathogenicity prediction, our approach incorporates domain prediction analysis results from InterProScan, which provide crucial insights into the functional impact of genetic variants. By identifying conserved domains affected by mutations, we gain a deeper understanding of how these alterations may disrupt essential protein functions. Furthermore, the precise location of the mutation within the protein sequence is highlighted in the output file, enabling users to interpret the results and correlate specific variants with potential functional consequences. The combination of GO term-based predictions and domain analysis results enhances the comprehensiveness of our pathogenicity prediction method.

The comprehensive evaluation of PMPred's performance, combining ROC analysis, performance metrics, and case studies, consistently demonstrated its effectiveness as a pathogenicity prediction tool. To evaluate the performance of PMPred thoroughly, we employed a diverse benchmark dataset from ClinVar, encompassing various types of mutations, including in-frame, frameshift, and nonsense variants. Our analysis revealed that PMPred exhibited remarkable sensitivity and reliability, particularly in predicting the pathogenicity of nonsense mutations. The application of ROC analysis provided a visual representation of the true positive rate against the false positive rate at different classification thresholds, allowing us to assess the overall discriminative power of PMPred. In addition to ROC analysis, the performance metrics tables offered detailed quantitative assessments, including accuracy, precision, recall, and F1-score. These metrics allowed us to comprehensively compare the performance of PMPred with other methods for different categories of variants. PMPred consistently demonstrated high accuracy, precision, and sensitivity, emphasizing its robustness in predicting pathogenicity. Furthermore, the case studies involving clinical exome sequencing data provided a real-world assessment of PMPred's performance. By accurately predicting the pathogenicity of known variants, PMPred showcased its potential as a reliable tool for assisting in variant interpretation and clinical decision-making.

Overall, the integration of the evaluation methodologies covered our confidence in PMPred's efficacy as a powerful pathogenicity prediction method for indels, frameshift, and nonsense mutations. By providing a reliable and automated tool for pathogenicity prediction, we hope that PMPred will contribute to the realization of personalized medicine, offering valuable insights for clinicians, researchers, and ultimately, improving patient outcomes.

5.3. Future directions

In future research, it would be valuable to conduct an in-depth analysis of the specific GO terms that exhibit significant score fluctuations after genetic variants have been introduced. By conducting such an analysis, it would be possible to identify the precise biological processes and molecular functions that are most profoundly affected by these mutations. This work could serve as a foundational step toward elucidating the critical molecular mechanisms governing variant pathogenicity. The identification and characterization of these influential GO terms could offer valuable insights into the etiology of various genetic disorders, as well as the potential therapeutic targets associated with them. Subsequently, a comprehensive review of the existing scientific literature and experimental data would be essential to validate the roles of these GO terms within the context of disease pathogenesis. Furthermore, it would facilitate the elucidation of the specific cellular pathways and biological functions governed by these GO terms.

In the pursuit of precision medicine, integrating phenotype information, patient clinical data, and disease databases into PMPred represents a valuable future direction. By incorporating phenotypic data, it can offer personalized predictions that are fitted to specific patient populations, enabling more accurate diagnosis and treatment decisions. Moreover, by considering disease-specific databases, PMPred can be customized to address the unique genetic landscape of various disorders, making it a multitalented tool in clinical settings. Another direction to further advance the field involves integrating multi-omics data into PMPred. By incorporating additional layers of information, such as epigenomic and transcriptomic profiles, PMPred can provide a more comprehensive view of the functional impact of variants and reveal intricate interactions between genetic alterations and molecular phenotypes.

Despite the considerable progress made in the development of PMPred, there are some limitations that should be acknowledged. Firstly, our current version of PMPred predominantly focuses on frameshift, in-frame, and nonsense variants. While these variants are known to have significant functional impacts, the predictive accuracy for missense variants remains an area of improvement. Missense variants present unique challenges due to their diverse impact on protein structure and function, requiring further exploration and model refinement to enhance prediction accuracy for this class of mutations. In the future, we propose to extend the capabilities of PMPred to include the prediction of missense mutations, incorporating additional features like structural and evolutionary information. Expanding the predictive power of PMPred by integrating structural analysis methods, such as homology modeling and molecular dynamics simulations, could yield valuable insights into the functional effects of variants at the protein three-dimensional level.

Secondly, it's worth noting that our current version of PMPred relies on the hg19 reference genome for data annotation. To make PMPred even more flexible and applicable to a wider range of genetic analyses, we will incorporate support for the hg38 reference genome. This expansion will enable researchers and clinicians to leverage the enhanced capabilities of PMPred for variant annotation and interpretation in the context of the hg38 genome. By accommodating multiple reference genomes, we aim to provide users with a more comprehensive and flexible tool for pathogenicity prediction, aligning PMPred with the latest genomic resources and ensuring its relevance for diverse genetic studies and clinical applications.

Lastly, experimental validation of PMPred's predictions through functional assays is beneficial to ensuring the reliability and trustworthiness of its results. Although we employed a diverse benchmark dataset from ClinVar, encompassing various types of mutations, the limited size of the training data may pose a challenge. Incorporating functional data obtained from experimental assays can provide essential validation for the computational predictions, supporting the confidence in PMPred's accuracy and efficacy. By combining computational and experimental approaches, PMPred can solidify its position as a robust and reliable method for variant pathogenicity prediction.

REFERENCES

Adzhubei, I., Jordan, D. M., & Sunyaev, S. R. (2013). Predicting functional effect of human missense mutations using PolyPhen-2. *Current Protocols in Human Genetics*, *Chapter 7*, Unit7.20-Unit7.20. https://doi.org/10.1002/0471142905.hg0720s76

Alford, R. F., Leaver-Fay, A., Jeliazkov, J. R., O'Meara, M. J., DiMaio, F. P., Park, H., … Gray, J. J. (2017). The Rosetta All-Atom Energy Function for Macromolecular Modeling and Design. *Journal of Chemical Theory and Computation*, *13*(6), 3031–3048. https://doi.org/10.1021/acs.jctc.7b00125

AlphaFold and beyond. (2023). *Nature Methods*, *20*(2), 163. https://doi.org/10.1038/s41592-023-01790-6

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. *Nucleic Acids Research*, *25*(17), 3389–3402. https://doi.org/10.1093/nar/25.17.3389

Amberger, J. S., Bocchini, C. A., Schiettecatte, F., Scott, A. F., & Hamosh, A. (2015). OMIM.org: Online Mendelian Inheritance in Man (OMIM®), an Online catalog of human genes and genetic disorders. *Nucleic Acids Research*, *43*(D1), D789–D798. https://doi.org/10.1093/nar/gku1205

Ault, B. H. (2000). Factor H and the pathogenesis of renal diseases. *Pediatric Nephrology*, *14*(10), 1045–1053. https://doi.org/10.1007/s004670050069

Auton, A., Abecasis, G. R., Altshuler, D. M., Durbin, R. M., Abecasis, G. R., Bentley, D. R., … National Eye Institute, N. I. H. (2015). A global reference for human genetic variation. *Nature*, *526*(7571), 68–74. https://doi.org/10.1038/nature15393

Awad, M., & Khanna, R. (2015). Support Vector Regression BT - Efficient Learning Machines: Theories, Concepts, and Applications for Engineers and System Designers. In M. Awad & R. Khanna (Eds.) (pp. 67–80). Berkeley, CA: Apress. https://doi.org/10.1007/978-1-4302-5990-9_4

Bosch, A. van den. (2010). Hidden Markov Models BT - Encyclopedia of Machine

Learning. In C. Sammut & G. I. Webb (Eds.) (pp. 493–495). Boston, MA: Springer US. https://doi.org/10.1007/978-0-387-30164-8_362

Breiman, L. (2001). Random Forests. *Machine Learning*, *45*(1), 5–32. https://doi.org/10.1023/A:1010933404324

Bromberg, Y., & Rost, B. (2007). SNAP: predict effect of non-synonymous polymorphisms on function. *Nucleic Acids Research*, *35*(11), 3823–3835. https://doi.org/10.1093/nar/gkm238

Capriotti, E., Altman, R. B., & Bromberg, Y. (2013). Collective judgment predicts disease-associated single nucleotide variants. *BMC Genomics*, *14 Suppl 3*(Suppl 3), S2. https://doi.org/10.1186/1471-2164-14-S3-S2

Capriotti, E., & Fariselli, P. (2017). PhD-SNPg: a webserver and lightweight tool for scoring single nucleotide variants. *Nucleic Acids Research*, *45*(W1), W247–W252. https://doi.org/10.1093/nar/gkx369

Capriotti, E., Fariselli, P., & Casadio, R. (2005). I-Mutant2.0: predicting stability changes upon mutation from the protein sequence or structure. *Nucleic Acids Research*, *33*(Web Server issue), W306-10. https://doi.org/10.1093/nar/gki375

Carter, H., Douville, C., Stenson, P. D., Cooper, D. N., & Karchin, R. (2013). Identifying Mendelian disease genes with the variant effect scoring tool. *BMC Genomics*, *14 Suppl 3*(Suppl 3), S3. https://doi.org/10.1186/1471-2164-14-S3-S3

Cheng, J., Randall, A., & Baldi, P. (2006). Prediction of protein stability changes for single-site mutations using support vector machines. *Proteins*, *62*(4), 1125–1132. https://doi.org/10.1002/prot.20810

Choi, Y., & Chan, A. P. (2015). PROVEAN web server: a tool to predict the functional effect of amino acid substitutions and indels. *Bioinformatics (Oxford, England)*, *31*(16), 2745–2747. https://doi.org/10.1093/bioinformatics/btv195

Church, D. M., Lappalainen, I., Sneddon, T. P., Hinton, J., Maguire, M., Lopez, J., … Flicek, P. (2010, October). Public data archives for genomic structural variation. *Nature Genetics*. United States. https://doi.org/10.1038/ng1010-813

Cingolani, P., Platts, A., Wang, L. L., Coon, M., Nguyen, T., Wang, L., … Ruden, D. M. (2012). A program for annotating and predicting the effects of single nucleotide polymorphisms, SnpEff: SNPs in the genome of Drosophila melanogaster strain w1118; iso-2; iso-3. *Fly*, *6*(2), 80–92. https://doi.org/10.4161/fly.19695

Cooper, G. M., Stone, E. A., Asimenos, G., Green, E. D., Batzoglou, S., & Sidow, A. (2005). Distribution and intensity of constraint in mammalian genomic sequence. *Genome Research*, *15*(7), 901–913. https://doi.org/10.1101/gr.3577405

Danecek, P., Auton, A., Abecasis, G., Albers, C. A., Banks, E., DePristo, M. A., … Durbin, R. (2011). The variant call format and VCFtools. *Bioinformatics (Oxford,*

England), *27*(15), 2156–2158. https://doi.org/10.1093/bioinformatics/btr330

Davydov, E. V, Goode, D. L., Sirota, M., Cooper, G. M., Sidow, A., & Batzoglou, S. (2010). Identifying a high fraction of the human genome to be under selective constraint using GERP++. *PLoS Computational Biology*, *6*(12), e1001025. https://doi.org/10.1371/journal.pcbi.1001025

den Dunnen, J. T., Dalgleish, R., Maglott, D. R., Hart, R. K., Greenblatt, M. S., McGowan-Jordan, J., … Taschner, P. E. M. (2016). HGVS Recommendations for the Description of Sequence Variants: 2016 Update. *Human Mutation*, *37*(6), 564–569. https://doi.org/10.1002/humu.22981

Di Tommaso, P., Chatzou, M., Floden, E. W., Barja, P. P., Palumbo, E., & Notredame, C. (2017). Nextflow enables reproducible computational workflows. *Nature Biotechnology*, *35*(4), 316–319. https://doi.org/10.1038/nbt.3820

DiScipio, R. G., Daffern, P. J., Schraufstätter, I. U., & Sriramarao, P. (1998). Human polymorphonuclear leukocytes adhere to complement factor H through an interaction that involves alphaMbeta2 (CD11b/CD18). *Journal of Immunology (Baltimore, Md. : 1950)*, *160*(8), 4057–4066.

Dong, C., Wei, P., Jian, X., Gibbs, R., Boerwinkle, E., Wang, K., & Liu, X. (2015). Comparison and integration of deleteriousness prediction methods for nonsynonymous SNVs in whole exome sequencing studies. *Human Molecular Genetics*, *24*(8), 2125–2137. https://doi.org/10.1093/hmg/ddu733

Douville, C., Masica, D. L., Stenson, P. D., Cooper, D. N., Gygax, D. M., Kim, R., … Karchin, R. (2016). Assessing the Pathogenicity of Insertion and Deletion Variants with the Variant Effect Scoring Tool (VEST-Indel). *Human Mutation*, *37*(1), 28–35. https://doi.org/10.1002/humu.22911

Finn, R. D., Clements, J., & Eddy, S. R. (2011). HMMER web server: interactive sequence similarity searching. *Nucleic Acids Research*, *39*(Web Server issue), W29-37. https://doi.org/10.1093/nar/gkr367

Folkman, L., Yang, Y., Li, Z., Stantic, B., Sattar, A., Mort, M., … Zhou, Y. (2015). DDIG-in: Detecting disease-causing genetic variations due to frameshifting indels and nonsense mutations employing sequence and structural properties at nucleotide and protein levels. *Bioinformatics*, *31*(10), 1599–1606. https://doi.org/10.1093/bioinformatics/btu862

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, *29*, 1189–1232.

Gorlov, I. P., Kimmel, M., & Amos, C. I. (2006). Strength of the purifying selection against different categories of the point mutations in the coding regions of the human genome. *Human Molecular Genetics*, *15*(7), 1143–1150. https://doi.org/10.1093/hmg/ddl029

Grinberg, M. (2018). *Flask web development: developing web applications with*

python. " O'Reilly Media, Inc."

Hageman, G. S., Anderson, D. H., Johnson, L. V, Hancox, L. S., Taiber, A. J., Hardisty, L. I., … Allikmets, R. (2005). A common haplotype in the complement regulatory gene factor H (HF1/CFH) predisposes individuals to age-related macular degeneration. *Proceedings of the National Academy of Sciences*, *102*(20), 7227–7232. https://doi.org/10.1073/pnas.0501536102

Hicks, S. A., Strümke, I., Thambawita, V., Hammou, M., Riegler, M. A., Halvorsen, P., & Parasa, S. (2022). On evaluation metrics for medical applications of artificial intelligence. *Scientific Reports*, *12*(1), 5979. https://doi.org/10.1038/s41598-022-09954-8

Hu, J., & Ng, P. C. (2013). SIFT Indel: Predictions for the Functional Effects of Amino Acid Insertions/Deletions in Proteins. *PLoS ONE*, *8*(10), 1–8. https://doi.org/10.1371/journal.pone.0077940

Hubbard, T., Barker, D., Birney, E., Cameron, G., Chen, Y., Clark, L., … Clamp, M. (2002). The Ensembl genome database project. *Nucleic Acids Research*, *30*(1), 38–41. https://doi.org/10.1093/nar/30.1.38

Hulo, N., Bairoch, A., Bulliard, V., Cerutti, L., De Castro, E., Langendijk-Genevaux, P. S., … Sigrist, C. J. A. (2006). The PROSITE database. *Nucleic Acids Research*, *34*(Database issue), D227-30. https://doi.org/10.1093/nar/gkj063

Ioannidis, N. M., Rothstein, J. H., Pejaver, V., Middha, S., McDonnell, S. K., Baheti, S., … Sieh, W. (2016). REVEL: An Ensemble Method for Predicting the Pathogenicity of Rare Missense Variants. *American Journal of Human Genetics*, *99*(4), 877–885. https://doi.org/10.1016/j.ajhg.2016.08.016

Jokiranta, T. S., Jaakola, V.-P., Lehtinen, M. J., Pärepalo, M., Meri, S., & Goldman, A. (2006). Structure of complement factor H carboxyl-terminus reveals molecular basis of atypical haemolytic uremic syndrome. *The EMBO Journal*, *25*(8), 1784–1794. https://doi.org/10.1038/sj.emboj.7601052

Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., … Hassabis, D. (2021). Highly accurate protein structure prediction with AlphaFold. *Nature*, *596*(7873), 583–589. https://doi.org/10.1038/s41586-021-03819-2

Käll, L., Krogh, A., & Sonnhammer, E. L. L. (2004). A Combined Transmembrane Topology and Signal Peptide Prediction Method. *Journal of Molecular Biology*, *338*(5), 1027–1036. https://doi.org/https://doi.org/10.1016/j.jmb.2004.03.016

Karczewski, K. J., Francioli, L. C., Tiao, G., Cummings, B. B., Alföldi, J., Wang, Q., … Consortium, G. A. D. (2020). The mutational constraint spectrum quantified from variation in 141,456 humans. *Nature*, *581*(7809), 434–443. https://doi.org/10.1038/s41586-020-2308-7

Karczewski, K. J., Weisburd, B., Thomas, B., Solomonson, M., Ruderfer, D. M., Kavanagh, D., … MacArthur, D. G. (2017). The ExAC browser: displaying

reference data information from over 60 000 exomes. *Nucleic Acids Research*, *45*(D1), D840–D845. https://doi.org/10.1093/nar/gkw971

Katsonis, P., Wilhelm, K., Williams, A., & Lichtarge, O. (2022). Genome interpretation using in silico predictors of variant impact. *Human Genetics*, *141*(10), 1549–1577. https://doi.org/10.1007/s00439-022-02457-6

Kavanagh, D., Goodship, T. H., & Richards, A. (2013). Atypical hemolytic uremic syndrome. *Seminars in Nephrology*, *33*(6), 508–530. https://doi.org/10.1016/j.semnephrol.2013.08.003

Köhler, S., Gargano, M., Matentzoglu, N., Carmody, L. C., Lewis-Smith, D., Vasilevsky, N. A., … Robinson, P. N. (2021). The Human Phenotype Ontology in 2021. *Nucleic Acids Research*, *49*(D1), D1207–D1217. https://doi.org/10.1093/nar/gkaa1043

Kryshtafovych, A., Schwede, T., Topf, M., Fidelis, K., & Moult, J. (2019). Critical assessment of methods of protein structure prediction (CASP)-Round XIII. *Proteins*, *87*(12), 1011–1020. https://doi.org/10.1002/prot.25823

Landrum, M. J., Lee, J. M., Benson, M., Brown, G. R., Chao, C., Chitipiralla, S., … Maglott, D. R. (2018). ClinVar: Improving access to variant interpretations and supporting evidence. *Nucleic Acids Research*, *46*(D1), D1062–D1067. https://doi.org/10.1093/nar/gkx1153

Lee, T. I., & Young, R. A. (2013). Transcriptional regulation and its misregulation in disease. *Cell*, *152*(6), 1237–1251. https://doi.org/10.1016/j.cell.2013.02.014

Lek, M., Karczewski, K. J., Minikel, E. V, Samocha, K. E., Banks, E., Fennell, T., … Consortium, E. A. (2016). Analysis of protein-coding genetic variation in 60,706 humans. *Nature*, *536*(7616), 285–291. https://doi.org/10.1038/nature19057

Liu, Y., Yeung, W. S. B., Chiu, P. C. N., & Cao, D. (2022). Computational approaches for predicting variant impact: An overview from resources, principles to applications. *Frontiers in Genetics*, *13*, 981005. https://doi.org/10.3389/fgene.2022.981005

Loewe, L., & Hill, W. G. (2010, April). The population genetics of mutations: good, bad and indifferent. *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences*. England. https://doi.org/10.1098/rstb.2009.0317

Losse, J., Zipfel, P. F., & Józsi, M. (2010). Factor H and factor H-related protein 1 bind to human neutrophils via complement receptor 3, mediate attachment to Candida albicans, and enhance neutrophil antimicrobial activity. *Journal of Immunology (Baltimore, Md.: 1950)*, *184*(2), 912–921. https://doi.org/10.4049/jimmunol.0901702

Luo, Y., Hitz, B. C., Gabdank, I., Hilton, J. A., Kagda, M. S., Lam, B., … Cherry, J. M. (2020). New developments on the Encyclopedia of DNA Elements (ENCODE) data portal. *Nucleic Acids Research*, *48*(D1), D882–D889.

https://doi.org/10.1093/nar/gkz1062

Magrane, M. (2011). UniProt Knowledgebase: a hub of integrated protein data. *Database : The Journal of Biological Databases and Curation*, *2011*, bar009. https://doi.org/10.1093/database/bar009

Marwaha, S., Knowles, J. W., & Ashley, E. A. (2022). A guide for the diagnosis of rare and undiagnosed disease: beyond the exome. *Genome Medicine*, *14*(1), 23. https://doi.org/10.1186/s13073-022-01026-w

Matthews, B. W. (1975). Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *Biochimica et Biophysica Acta (BBA) - Protein Structure*, *405*(2), 442–451. https://doi.org/https://doi.org/10.1016/0005-2795(75)90109-9

McLaren, W., Gil, L., Hunt, S. E., Riat, H. S., Ritchie, G. R. S., Thormann, A., … Cunningham, F. (2016). The Ensembl Variant Effect Predictor. *Genome Biology*, *17*(1), 122. https://doi.org/10.1186/s13059-016-0974-4

Melo, F. (2013). Area under the ROC Curve BT - Encyclopedia of Systems Biology. In W. Dubitzky, O. Wolkenhauer, K.-H. Cho, & H. Yokota (Eds.) (pp. 38–39). New York, NY: Springer New York. https://doi.org/10.1007/978-1-4419-9863-7_209

Mistry, J., Chuguransky, S., Williams, L., Qureshi, M., Salazar, G. A., Sonnhammer, E. L. L., … Bateman, A. (2021). Pfam: The protein families database in 2021. *Nucleic Acids Research*, *49*(D1), D412–D419. https://doi.org/10.1093/nar/gkaa913

Moore, S. R., Menon, S. S., Cortes, C., & Ferreira, V. P. (2021). Hijacking Factor H for Complement Immune Evasion. *Frontiers in Immunology*, *12*(February), 1–23. https://doi.org/10.3389/fimmu.2021.602277

Mulder, N., & Apweiler, R. (2007). InterPro and InterProScan: tools for protein sequence classification and comparison. *Methods in Molecular Biology (Clifton, N.J.)*, *396*, 59–70. https://doi.org/10.1007/978-1-59745-515-2_5

Noris, M., & Remuzzi, G. (2013). Overview of complement activation and regulation. *Seminars in Nephrology*, *33*(6), 479–492. https://doi.org/10.1016/j.semnephrol.2013.08.001

O'Leary, N. A., Wright, M. W., Brister, J. R., Ciufo, S., Haddad, D., McVeigh, R., … Pruitt, K. D. (2016). Reference sequence (RefSeq) database at NCBI: current status, taxonomic expansion, and functional annotation. *Nucleic Acids Research*, *44*(D1), D733-45. https://doi.org/10.1093/nar/gkv1189

Park, Y. M., Squizzato, S., Buso, N., Gur, T., & Lopez, R. (2017). The EBI search engine: EBI search as a service-making biological data accessible for all. *Nucleic Acids Research*, *45*(W1), W545–W549. https://doi.org/10.1093/nar/gkx359

Paysan-Lafosse, T., Blum, M., Chuguransky, S., Grego, T., Pinto, B. L., Salazar, G. A., … Bateman, A. (2023). InterPro in 2022. *Nucleic Acids Research*, *51*(D1), D418–D427. https://doi.org/10.1093/nar/gkac993

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., … Duchesnay, E. (2011). Scikit-learn: Machine Learning in {P}ython. *Journal of Machine Learning Research*, *12*, 2825–2830. Retrieved from http://www.jmlr.org/papers/volume12/pedregosa11a/pedregosa11a.pdf

Pejaver, V., Urresti, J., Lugo-Martinez, J., Pagel, K. A., Lin, G. N., Nam, H.-J., … Radivojac, P. (2020). Inferring the molecular and phenotypic impact of amino acid variants with MutPred2. *Nature Communications*, *11*(1), 5918. https://doi.org/10.1038/s41467-020-19669-x

Pérez-Caballero, D., González-Rubio, C., Esther Gallardo, M., Vera, M., López-Trascasa, M., Rodríguez de Córdoba, S., & Sánchez-Corral, P. (2001). Clustering of missense mutations in the C-terminal region of factor H in atypical hemolytic uremic syndrome. *American Journal of Human Genetics*, *68*(2), 478–484. https://doi.org/10.1086/318201

Pires, D. E. V, Ascher, D. B., & Blundell, T. L. (2014). mCSM: predicting the effects of mutations in proteins using graph-based signatures. *Bioinformatics (Oxford, England)*, *30*(3), 335–342. https://doi.org/10.1093/bioinformatics/btt691

Quevillon, E., Silventoinen, V., Pillai, S., Harte, N., Mulder, N., Apweiler, R., & Lopez, R. (2005). InterProScan: Protein domains identifier. *Nucleic Acids Research*, *33*(SUPPL. 2), 116–120. https://doi.org/10.1093/nar/gki442

Raina, R., Mangat, G., Hong, G., Shah, R., Nair, N., Abboud, B., … Sethi, S. K. (2022). Anti-factor H antibody and its role in atypical hemolytic uremic syndrome. *Frontiers in Immunology*, *13*(August), 5–10. https://doi.org/10.3389/fimmu.2022.931210

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

Rentzsch, P., Witten, D., Cooper, G. M., Shendure, J., & Kircher, M. (2019). CADD: Predicting the deleteriousness of variants throughout the human genome. *Nucleic Acids Research*, *47*(D1), D886–D894. https://doi.org/10.1093/nar/gky1016

Rice, P., Longden, I., & Bleasby, A. (2000). EMBOSS: the European Molecular Biology Open Software Suite. *Trends in Genetics : TIG*, *16*(6), 276–277. https://doi.org/10.1016/s0168-9525(00)02024-2

Richards, S., Aziz, N., Bale, S., Bick, D., Das, S., Gastier-Foster, J., … Committee, ; on behalf of the ACMG Laboratory Quality Assurance. (2015). Standards and guidelines for the interpretation of sequence variants: a joint consensus recommendation of the American College of Medical Genetics and Genomics

and the Association for Molecular Pathology. *Genetics in Medicine*, *17*(5), 405–423. https://doi.org/10.1038/gim.2015.30

Rifaioglu, A. S., Doğan, T., Saraç, Ö. S., Ersahin, T., Saidi, R., Atalay, M. V., … Cetin-Atalay, R. (2018). Large-scale automated function prediction of protein sequences and an experimental case study validation on PTEN transcript variants. *Proteins: Structure, Function and Bioinformatics*, *86*(2), 135–151. https://doi.org/10.1002/prot.25416

Rogers, M. F., Shihab, H. A., Mort, M., Cooper, D. N., Gaunt, T. R., & Campbell, C. (2018). FATHMM-XF: accurate prediction of pathogenic point mutations via extended features. *Bioinformatics (Oxford, England)*, *34*(3), 511–513. https://doi.org/10.1093/bioinformatics/btx536

Sanner, M. F. (1999). Python: a programming language for software integration and development. *Journal of Molecular Graphics & Modelling*, *17*(1), 57–61.

Saraç, O. S., Atalay, V., & Cetin-Atalay, R. (2010). GOPred: GO molecular function prediction by combined classifiers. *PloS One*, *5*(8), e12382. https://doi.org/10.1371/journal.pone.0012382

Sayers, E. W., Bolton, E. E., Brister, J. R., Canese, K., Chan, J., Comeau, D. C., … Sherry, S. T. (2022). Database resources of the national center for biotechnology information. *Nucleic Acids Research*, *50*(D1), D20–D26. https://doi.org/10.1093/nar/gkab1112

Sayers, E. W., Bolton, E. E., Brister, J. R., Canese, K., Chan, J., Comeau, D. C., … Sherry, S. T. (2023). Database resources of the National Center for Biotechnology Information in 2023. *Nucleic Acids Research*, *51*(D1), D29–D38. https://doi.org/10.1093/nar/gkac1032

Schrödinger, L. L. C. (2010). The PyMOL molecular graphics system, version 1.3 r1. August.

Schultz, J., Copley, R. R., Doerks, T., Ponting, C. P., & Bork, P. (2000). SMART: a web-based tool for the study of genetically mobile domains. *Nucleic Acids Research*, *28*(1), 231–234. https://doi.org/10.1093/nar/28.1.231

Schwarz, J. M., Rödelsperger, C., Schuelke, M., & Seelow, D. (2010, August). MutationTaster evaluates disease-causing potential of sequence alterations. *Nature Methods*. United States. https://doi.org/10.1038/nmeth0810-575

Schwede, T., Kopp, J., Guex, N., & Peitsch, M. C. (2003). SWISS-MODEL: An automated protein homology-modeling server. *Nucleic Acids Research*, *31*(13), 3381–3385. https://doi.org/10.1093/nar/gkg520

Sherry, S. T., Ward, M.-H., Kholodov, M., Baker, J., Phan, L., Smigielski, E. M., & Sirotkin, K. (2001). dbSNP: the NCBI database of genetic variation. *Nucleic Acids Research*, *29*(1), 308–311. https://doi.org/10.1093/nar/29.1.308

Siepel, A., Bejerano, G., Pedersen, J. S., Hinrichs, A. S., Hou, M., Rosenbloom, K., … Haussler, D. (2005). Evolutionarily conserved elements in vertebrate, insect, worm, and yeast genomes. *Genome Research*, *15*(8), 1034–1050. https://doi.org/10.1101/gr.3715005

Sim, N. L., Kumar, P., Hu, J., Henikoff, S., Schneider, G., & Ng, P. C. (2012). SIFT web server: Predicting effects of amino acid substitutions on proteins. *Nucleic Acids Research*, *40*(W1), 452–457. https://doi.org/10.1093/nar/gks539

Stenson, P. D., Mort, M., Ball, E. V., Shaw, K., Phillips, A. D., & Cooper, D. N. (2014). The Human Gene Mutation Database: Building a comprehensive mutation repository for clinical and molecular genetics, diagnostic testing and personalized genomic medicine. *Human Genetics*, *133*(1), 1–9. https://doi.org/10.1007/s00439-013-1358-4

Tan, P.-N. (2009). Receiver Operating Characteristic BT - Encyclopedia of Database Systems. In L. LIU & M. T. ÖZSU (Eds.) (pp. 2349–2352). Boston, MA: Springer US. https://doi.org/10.1007/978-0-387-39940-9_569

Tang, H., & Thomas, P. D. (2016). PANTHER-PSEP: predicting disease-causing genetic variants using position-specific evolutionary preservation. *Bioinformatics (Oxford, England)*, *32*(14), 2230–2232. https://doi.org/10.1093/bioinformatics/btw222

Tate, J. G., Bamford, S., Jubb, H. C., Sondka, Z., Beare, D. M., Bindal, N., … Forbes, S. A. (2019). COSMIC: the Catalogue Of Somatic Mutations In Cancer. *Nucleic Acids Research*, *47*(D1), D941–D947. https://doi.org/10.1093/nar/gky1015

Ting, K. M. (2017). Confusion Matrix BT - Encyclopedia of Machine Learning and Data Mining. In C. Sammut & G. I. Webb (Eds.) (p. 260). Boston, MA: Springer US. https://doi.org/10.1007/978-1-4899-7687-1_50

Trott, O., & Olson, A. J. (2010). AutoDock Vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. *Journal of Computational Chemistry*, *31*(2), 455–461. https://doi.org/10.1002/jcc.21334

Wang, K., Li, M., & Hakonarson, H. (2010). ANNOVAR: Functional annotation of genetic variants from high-throughput sequencing data. *Nucleic Acids Research*, *38*(16), 1–7. https://doi.org/10.1093/nar/gkq603

Worth, C. L., Preissner, R., & Blundell, T. L. (2011). SDM--a server for predicting effects of mutations on protein stability and malfunction. *Nucleic Acids Research*, *39*(Web Server issue), W215-22. https://doi.org/10.1093/nar/gkr363

Wu, C., Macleod, I., & Su, A. I. (2013). BioGPS and MyGene.info: organizing online, gene-centric information. *Nucleic Acids Research*, *41*(Database issue), D561-5. https://doi.org/10.1093/nar/gks1114

Young, J. Y., Westbrook, J. D., Feng, Z., Peisach, E., Persikova, I., Sala, R., … Burley,

S. K. (2018). Worldwide Protein Data Bank biocuration supporting open access to high-quality 3D structural biology data. *Database : The Journal of Biological Databases and Curation*, *2018*. https://doi.org/10.1093/database/bay002

Zhang, T., Faraggi, E., Xue, B., Dunker, A. K., Uversky, V. N., & Zhou, Y. (2012). Spine-d: Accurate prediction of short and long disordered regions by a single neural-network based method. *Journal of Biomolecular Structure and Dynamics*, *29*(4), 799–813. https://doi.org/10.1080/073911012010525022

Zhou, N., Jiang, Y., Bergquist, T. R., Lee, A. J., Kacsoh, B. Z., Crocker, A. W., … Friedberg, I. (2019). The CAFA challenge reports improved protein function prediction and new functional annotations for hundreds of genes through experimental screens. *Genome Biology*, *20*(1), 244. https://doi.org/10.1186/s13059-019-1835-8

APPENDIX

APPENDIX A

DOCKING

The docking process using three tools are provided to elucidate the methodology. These steps encompass the preparation of the protein and ligand structures, defining the docking parameters and search space, performing the docking simulation, and analyzing the resulting docking poses to evaluate the binding interactions between CFH and ITGAM.

1. Open MGLTools and upload the target protein PDB file:
    - This step involves launching MGLTools and loading the three-dimensional structure of the target protein (CFH) in the Protein Data Bank (PDB) format. The target protein serves as the receptor for the docking simulations and must be prepared appropriately to ensure accurate and reliable docking results.
2. Prepare the target protein:
    - Click on "EDIT" and select "delete water":
        - Water molecules are typically not directly involved in protein-ligand interactions and can hinder docking calculations. Removing them simplifies the docking process.
    - Click on "EDIT" and select "hydrogens", then click on "add" and select "only polar":
        - Adding hydrogen atoms to the protein structure is crucial for accurately representing the potential hydrogen bonding interactions between the protein and ligand during the docking process.

        - Adding hydrogen atoms selectively to polar atoms helps reduce the complexity of the system and focuses on relevant interactions.

    - Click on "EDIT" and select "charges" and then select "add Kollman charges":

        - Assigning partial charges to the protein atoms using the Kollman charges method is necessary for calculating the electrostatic interactions between the protein and ligand during docking.

        - Charges play a vital role in determining the strength and stability of the protein-ligand interactions.

3. Save the prepared target protein as a PDBQT file:

    - Click on the Autodock Tools symbol and select "Grid".

    - From the "macromolecule" section, click on "choose" and select the protein file.

    - Click on "select molecule" to load the protein as the receptor.

    - Saving the receptor as a PDBQT (AutoDock's specific format).

        - Saving file in this format ensures that it contains all the necessary information and parameters for successful docking calculations. The PDBQT file format includes information about atomic coordinates, atom types, charges, and other properties relevant to the docking simulations.

4. Prepare the ligand:

    - Upload the ligand molecule and select "ligand" and "input":

        - The ligand represents the protein candidate that will bind to the target protein.

    - Choose the ligand molecule for docking by selecting it in the MGLTools interface.

    - Click "output" and save the ligand as a PDBQT file:

        - Converting the ligand to the PDBQT file format ensures it contains the necessary information required for successful docking, including atom types, charges, and other properties.

5. Define the docking grid:

- Click on "Grid" in the Autodock Tools symbol.
- Select "grid box" and choose the "output grid dimension file" to define the dimensions of the docking grid.
- Saving the grid configuration as a grid.txt file specifies the coordinates and dimensions of the grid box within which the ligand will be docked.
- The grid box defines the search space for the ligand and helps restrict the docking calculations to specific regions of the receptor.

6. Create a config file:
    - Open a text editor and create the config.txt file:
        - Use a text editor to create a new file and save it as "config.txt" in the desired directory.
        - The config.txt file should include relevant information such as the receptor and ligand filenames, search space dimensions, exhaustiveness, and other parameters specific to AutoDock Vina.
    - Define the contents of the config.txt file: The config.txt file typically includes parameters such as:
        - receptor: Specifies the filename of the prepared receptor (CFH.pdbqt) for the docking simulation.
        - ligand: Specifies the filename of the prepared ligand (ITGAM.pdbqt) for the docking simulation.
        - center_x, center_y, center_z: Specifies the coordinates of the center of the docking grid, which determines the region where the ligand will be docked.
        - size_x, size_y, size_z: Specifies the dimensions of the docking grid, defining the search space for the ligand.
        - exhaustiveness: Determines the thoroughness of the docking search algorithm, with higher values indicating more exhaustive search but longer computational time.
        - Other parameters: AutoDock Vina provides additional options to control the docking process, such as energy range, number of output poses, etc.
    - Save the config.txt file:

- After defining the necessary parameters in the config.txt file, save it in the same directory where the docking files are located.

7. Run the docking simulation:
    - Open the terminal or command prompt and navigate to the directory where the docking files are located and execute the AutoDock Vina command, providing the receptor (CFH), ligand (ITGAM) and config.txt file as an argument:

```
"path\to\Vina\vina.exe" --receptor CFH.pdbqt --ligand ITGAM.pdbqt --config config.txt --log log.txt --out output.pdbqt
```

        - This command initiates the docking simulation using AutoDock Vina, using the specified configuration file (config.txt) and generating the log file (log.txt) and pdbqt file as the output.
        - AutoDock Vina utilizes the parameters specified in the config.txt file to perform the docking calculations, exploring the ligand's conformational space within the defined search space and producing multiple docking poses.

8. Log file (log.txt):
    - The log.txt file contains valuable information about the progress and details of the docking simulation, including docking scores, convergence information, and other relevant details.
    - It serves as a record of the docking process, enabling you to assess the success of the docking and analyze the results obtained.
    - The log file can provide insights into the quality of the docking poses, convergence criteria, and any potential issues or warnings encountered during the simulation.

9. Visualization: Load the docking output files and visualize the binding modes:
    - Open PyMOL and load the receptor (CFH) and ligand (ITGAM) PDBQT files for both the wildtype and mutated forms.
    - Select each binding mode and display it in the cartoon representation to visualize the overall structure and interactions.
        - Go to the "Display" menu and choose "Cartoon" to represent the structure in the cartoon style.
        - Adjust the color scheme and style of the cartoon representation as needed to enhance visualization and distinguish different elements (e.g., helices, strands, loops).

- Switch to the surface representation to gain insights into the spatial arrangement and accessibility of the binding site in each binding mode.
    - Go to the "Display" menu and choose "Surface" to represent the structure as a molecular surface.
    - Adjust the parameters of the surface representation, such as transparency, opacity, and color, to optimize the visualization.

CURRICULUM VITAE

PERSONAL INFORMATION

Surname, Name: Mona Shojaei

Nationality: Iran

Marital Status: Married

EDUCATION

Degree	Institution	Year of Graduation
MS	METU	2016
BS	Tabriz University	2012
High School	Nikandish	2007

WORK EXPERIENCE

Year	Place	Enrollment
2022-Present	Nucleus Genetics	Bioinformatics Specialist
2016-2022	Intergen Genetics	Bioinformatics Specialist

FOREIGN LANGUAGES

Native Persian, Advanced Turkish, Advanced English