**Graph-Based Prediction of Protein Stability Changes Induced by Single Point Mutations**

A Dissertation

Submitted to the Central University of Himachal Pradesh for the Partial Fulfillment of the Degree of

# Masters of Science in Bioinformatics

### In the School of Life Sciences

In the Centre for Computational Biology and Bioinformatics



**Under the Supervision of Prof. Mahesh Kulharia By**

**Sanjay Singh Chauhan**

# (CUHP22CBB41)

**CENTRAL UNIVERSITY OF HIMACHAL PRADESH**

# June, 2024



**हिमाचल प्रदेश केंद्रीय विश्वविद्यालय**

**Central University of Himachal Pradesh**

(Established under Central Universities Act 2009)

Dharamshala (HP) - 176215

#### CERTIFICATE

**Centre for Computational Biology and Bioinformatics School of Life Sciences**

Central University of Himachal Pradesh, Dharamshala [Established under the Central Universities Act 2009] Dharamshala, District Kangra,

Himachal Pradesh [India]– 176215

Date:

This is to certify that the dissertation entitled **“Graph-Based Prediction of Protein Stability Changes Induced by Single Point Mutations”** submitted by **Sanjay Singh Chauhan** (CUHP22CBB41) **S/o Sh. Kundan Singh Chauhan** to the Central University of Himachal Pradesh, Dharamshala in partial fulfillment of the requirements for the award of degree of **Master of Science in Bioinformatics** has been approved by the committee after an oral/ viva-voce examination by an External Examiner.

#### Supervisor External Examiner

Prof. Mahesh Kulharia

#### Director of the Centre

Prof. Sunil Kumar

#### Dean, School of Life Sciences

 **हिमाचल प्रदेश केंद्रीय विश्वविद्यालय**

**Central University of Himachal Pradesh**

(Established under Central Universities Act 2009)

Dharamshala (HP) - 176215

#### SIMILARITY INDEX CERTIFICATE

**Centre for Computational Biology and Bioinformatics School of Life Sciences**

Central University of Himachal Pradesh, Dharamshala [Established under the Central Universities Act 2009] Dharamshala, District Kangra,

Himachal Pradesh [India]– 176215

Date:

This to certify that I, **Sanjay Singh Chauhan** have carried out the research embodied in the present dissertation and the work is original. The total plagiarism from dissertation work is less than 10 percent. The Turnitin report is attached at the end of the dissertation.

Name: Sanjay Singh Chauhan CUHP22CBB41

This is to certify that the above claim is correct to the best of my knowledge.

(Prof. Mahesh Kulharia)

# DECLARATION

I, **Sanjay Singh Chauhan (CUHP22CBB41)** declare that the dissertation entitled **“Graph-Based Prediction of Protein Stability Changes Induced by Single- Point Mutations”** has been carried out under the supervision of **Dr. Mahesh Kulharia at** Central University of Himachal Pradesh. The dissertation is submitted to the Centre for Computational Biology and Bioinformatics, Central University of Himachal Pradesh towards the partial fulfillment of the Degree of Masters of Science in Bioinformatics. The dissertation has not previously formed the basis for the award of any other degree or diploma. I hereby declare that this work is original.

Place: Shahpur Campus Sanjay Singh Chauhan

Date: M.Sc. Bioinformatics

**ACKNOWLEDGEMENTS**

I would like to express my special thanks to our mentor **Prof. Mahesh Kulharia** for his time and efforts he provided throughout the year. Your useful advice and suggestions were really helpful to me during the projects ‘s completion and big thanks to our RD **Miss. Ashita Sood** for her constant cooperation and encouragement throughout this project.

Name and signature of the student

**TABLE OF CONTENTS**

**List of Tables i**

List of Figures ii

List of Abbreviations iii

[Abstract iv](#_TOC_250000)

## CHAPTER 1: INTRODUCTION 1----

**CHAPTER 2: REVIEW OF LITERATURE CHAPTER 3: MATERIAL AND METHODS CHAPTER 4: RESULTS AND DISCUSSION**

## CHAPTER 5: CONCLUSION

**CHAPTER 6: REFERENCES**

#### Centre for Computational Biology and Bioinformatics School of Life Sciences

Central University of Himachal Pradesh, Dharamshala Dharmshala, District Kangra, Himachal Pradesh [India]– 176215

Abstract:

### Title of Dissertation Graph-Based Prediction of Protein Stability Changes Induced by Single-Point Mutation.

Enrolment Number : CUHP22CBB41

Date of Dissertation Submission :

Total pages of the dissertation :

Supervisor : Dr. Mahesh Kulharia

**Background:** Protein stability, crucial for biological function, can be significantly affected by single-point mutations, altering Gibbs free energy. Accurate prediction methods include structure- based approaches, limited by the 0.5% availability of structural data in Uni Prot, and more broadly applicable but less accurate sequence-based methods. This study introduces a graph-based machine learning approach, combining structural and sequence data, to predict mutation effects on protein stability. Our method enhances prediction accuracy, even with limited structural data, improving insights into protein stability and genetic variations.

**Methods:** We constructed graphs for wild-type and mutant proteins, using 9 physicochemical properties as node features. Our graph-based approach predicted Gibbs free energy changes [1]. The model was validated using k-fold cross-validation to ensure robustness and prediction accuracy.

**Results:** Our graph-based model demonstrated robust performance in predicting Gibbs free energy changes, with mean squared error (MSE) and R² scores comparable to existing tools. This led to the development of a new tool that offers enhanced accuracy and reliability in predicting protein stability changes due to mutations.

**Keywords:** Graph attention neural network, Gibbs free energy, Physiochemical properties, Point mutation.

(Sanjay Singh Chauhan) (Prof. Mahesh Kulharia)

M.Sc. Bioinformatics Supervisor

Date: Date:

Prof. Mahesh Kulharia Director of the Centre

**1. INTRODUCTION:**

Protein carries function in their 3D forms which are made by the amino acid sequence which get in secondary structure form and make tertiary structure which is performs function but the folding of the protein is based on the folding free energy ((∆∆G) [1] which we can use to assess the stability of the protein before mutation and after the mutation change in this stability in the structure [2]. Finding the stability of the mutant protein after the mutation experimentally is costly process and time taking. So many tools have been designed for doing this task for predicting the stability because by predicting the stability of the protein [3]. We can apply this knowledge in various applications in protein engineering, personalized medicine, and precision diagnostics

It is observed that many genetic disorders are caused by missense mutations that alter the stability of the native protein various methods are proposed for predicting the stability namely classified into structure based and sequence based approaches the structure based approaches has higher accuracy as compared to the sequence based approaches but uniprot only contain 0.5% of the total structures the sequence based structures apply machine learning on the amino acid sequence of proteins and apply machine learning techniques to predict the changes in the stability [5]. The stability of proteins is a fundamental aspect of molecular biology, influencing their functionality, interaction with other molecules, and overall role in biological systems. One of the key metrics for assessing protein stability is the Gibbs free energy change (ΔΔG) [2], which measures the difference in free energy between the wild-type protein and its mutant form. Understanding and predicting ΔΔG is crucial not only for basic science but also for applications in drug design, disease research, and synthetic biology.

Traditionally, predicting ΔΔG involves computationally intensive methods such as molecular dynamics simulations, which, despite their accuracy, can be prohibitively slow and resource- demanding for large-scale studies [5]. This has driven the search for more efficient computational models that can leverage existing structural data and sophisticated machine learning techniques to provide accurate and scalable predictions.

#### Graph-Based Representation of Proteins

To harness the power of machine learning for ΔΔG prediction, one promising approach involves representing protein structures as graphs. In this representation:

* **Nodes**: Each node in the graph corresponds to an amino acid in the protein sequence.
* **Edges**: Edges represent interactions between amino acids, including covalent bonds, hydrogen bonds [7], ionic interactions [8], and van der Waals forces [9].

Such a representation captures the complex, three-dimensional nature of protein structures in a format that can be efficiently processed by computational models [6]. Graph-based models [10] excel in scenarios where relational data is key, making them well-suited for protein structure analysis.

#### Feature Engineering for Nodes and Edges

To effectively use graph representations for ΔΔG prediction, it's essential to encode relevant information about the amino acids (nodes) and their interactions (edges). Each amino acid can be described by various physicochemical properties [19], such as:

1. Hydrophobicity [11]
2. Charge [12]
3. Molecular weight [13]
4. Hydrophobicity
5. VSc
6. p1
7. p2
8. SASA (Solvent Accessible Surface Area) [14]
9. NCISC (Non-Contact Infrared Skin Control)

These properties provide a comprehensive profile of each amino acid, capturing its behavior in different chemical environments and interactions within the protein.

#### Constructing Wild-Type and Mutant Graphs

For the task of predicting ΔΔG, we need to consider both the wild-type protein and its mutant form:

* **Wild-Type Graph**: This graph is constructed from the original protein structure obtained from the Protein Data Bank (PDB) [15].
* **Mutant Graph**: This graph is derived from the same PDB [15] structure but with a specific point mutation introduced.

Both graphs retain the same overall topology, with the mutant graph having a modified node representing the mutated amino acid [17]. This dual-graph approach allows the model to learn the subtle changes in interactions and stability brought about by the mutation [15].

#### GAT neural network

A Graph Attention Network (GAT [16]) is a neural network architecture that operates on graph- structured data, leveraging masked self-attentional layers to address the shortcomings of prior methods based on graph convolutions or their approximations.

#### Key Components:

* **Masked Self-Attentional Layers:** Each node in the graph attends to its neighbors’ features, allowing for implicit specification of different weights to different nodes in a neighborhood.
* **Graph Convolutional Layers:** Modified to incorporate attention mechanisms, enabling the model to focus on relevant nodes and edges.

#### How it Works:

1. **Node Embeddings:** Each node in the graph is represented as a vector, capturing its features.
2. **Attention Mechanism:** Each node attends to its neighbors’ features, computing a weighted sum of the neighbors’ features based on their relevance.
3. **Graph Convolutional Layer:** The attention weights are used to compute a weighted sum of the neighbors’ features, producing a new representation for each node.
4. **Repeat:** The process is repeated for multiple layers, allowing the model to capture complex patterns and relationships in the graph [17].

#### Advantages:

1. **Efficient:** GATs are computationally efficient, as they do not require costly matrix operations or knowledge of the entire graph structure upfront.
2. **Flexible**: GATs [16] can handle graphs of varying sizes and structures, making them suitable for a wide range of applications.
3. **Effective:** GATs have achieved state-of-the-art results on various graph-based tasks, such as node classification, graph classification, and link prediction [16].

#### Applications:

1. **Social Network Analysis:** GATs can be used to analyze social networks, identifying influential individuals and predicting user behavior.
2. **Recommendation Systems:** GATs can be applied to recommendation systems, incorporating user-item interactions and item-item relationships [18].
3. **Molecular Biology:** GATs can be used to analyze protein-protein interaction networks, identifying key players in biological processes**.**

#### Graph Attention Neural Networks (GANNs)

Graph Attention Neural Networks (GANNs) represent an advanced class of models that have shown significant promise in handling graph-structured data. GANNs extend traditional Graph Neural Networks (GNNs) [16] by incorporating attention mechanisms, which enable the model to weigh the importance of different nodes and edges dynamically. This capability is particularly useful for protein structures, where the impact of a mutation can vary widely depending on the local context and interactions within the protein [19].

#### Mechanism of GANNs

1. **Attention Mechanism**: In GANNs, attention coefficients are calculated for each pair of connected nodes, allowing the model to focus on the most critical interactions. These coefficients are learned during training and reflect the relative importance of different amino acid interactions [20] for predicting ΔΔG.
2. **Feature Aggregation**: Node features (amino acid properties) are aggregated using the attention coefficients, ensuring that each node's updated representation incorporates the most relevant information from its neighbors.
3. **Layer Stacking**: Multiple layers of attention mechanisms are stacked, enabling the model to capture higher-order interactions and complex dependencies within the protein structure.

#### Training the Model

Training a GANN for ΔΔG prediction involves several steps:

1. **Dataset Preparation**: Compile a dataset of protein structures with known ΔΔG values from the PDB [15]. This dataset includes both wild-type and corresponding mutant structures.
2. **Graph Construction**: Convert each protein structure into a graph, encoding node features and edge types [21].
3. **Model Training**: Train the GANN on the dataset, using a suitable loss function (e.g., mean squared error) to minimize the difference between predicted and actual ΔΔG values.

#### Applications and Implications

The use of GANNs for predicting protein stability has far-reaching implications:

1. **Drug Design**: Accurate ΔΔG predictions can inform the design of stable protein-based drugs and therapeutic enzymes.
2. **Disease Research**: Understanding the stability impacts of mutations can shed light on the molecular basis of genetic diseases and guide the development of targeted treatments.
3. **Synthetic Biology**: Engineers can design proteins with desired stability profiles for industrial applications, such as enzymes for biofuel production or biocatalysts.

#### Challenges and Future Directions

Despite the potential of GANNs, several challenges remain:

1. **Data Quality**: High-quality, experimentally validated ΔΔG data is essential for training robust models. Efforts to curate and expand such datasets are ongoing [22].
2. **Model Interpretability**: While GANNs provide powerful predictions, understanding the underlying biological mechanisms remains a complex task. Enhancing model interpretability is a critical area of research.
3. **Generalization**: Ensuring that models generalize well to novel proteins and mutations is crucial for their widespread application.

## 2. REVIEW OF LITERATURE:

**2.1 Computational tools with predictability features**.

Computationally techniques are based on algorithms, which are a set of rules that employ numerous prediction features:

**(a) Structural Features:** The size and distribution of hydrophobic areas within a protein that can influence its interactions with its surroundings.

* Environment Protein Packing: The way individual amino acid residues come together and interact within the protein structure, contributing to the overall compactness and stability of the molecule [18].
* Folding Patterns: The specific way the polypeptide chain folds into a three-dimensional structure, influencing the protein's stability and function Click or tap here to enter text.
* Backbone Angles: The angles formed by the protein backbone's covalent bonds, which contribute to the overall stability and geometry of the protein structure [23].
* Electrostatic Interactions: Charges and interactions between positively and negatively charged amino acid residues, influencing the protein's stability and its interactions with other molecules.

1. **Sequence Features:** Sequence features are fundamental elements considered in the analysis of protein stability, focusing on the linear arrangement of amino acids in a polypeptide chain. These features offer insights into the functional and structural aspects of a protein based on its primary sequence. Key sequence features include:

* Conserved Sequences: Regions of the amino acid sequence that remain Click or tap here to enter text. Relatively unchanged across evolution, indicating functional significance and potential contributions to stability.
* Amino Acid Position: The specific location of amino acids in the sequence, influencing the protein's structure, function, and stability. Changes in critical positions can impact overall protein behavior.
* Sequence Motifs: Short, conserved patterns of amino acids that may have specific functions or structural roles in the protein. Identification of these motifs’ aids in understanding stability- related characteristics.
* Repetitive Elements: Patterns of repeated amino acid sequences, which may contribute to the stability and structural integrity of certain protein domains [18].
* Mutation Sites: Analysis of sites in the sequence where mutations occur, providing information on potential destabilizing or stabilizing effects on the protein.

1. **Composite of the structural and Sequence Characteristics:** This method combines every one of the aforementioned characteristics to predict stability as a whole [24].

Holistic Assessment: The approach combines information about the protein's spatial arrangement, emphasizing structural features, with insights derived from the linear sequence, incorporating sequence features. This holistic assessment contributes to a more thorough prediction of stability.

* + Synergistic Insights: Structural features, such as the arrangement of secondary structures and solvent accessibility, are considered alongside sequence-based information, such as conserved motifs and amino acid positions. The combination of these features yields synergistic insights into the factors influencing stability.
  + Mechanistic Understanding: Integrating structural and sequence features enables a more profound mechanistic understanding of how the protein's three-dimensional structure relates to its primary sequence and how both contribute to overall stability [25].
  + Increased Predictive Accuracy: By utilizing a diverse set of features, the combined approach enhances the accuracy of stability predictions. This is particularly valuable in capturing the multifaceted nature of protein behavior and stability determinants.
  + Adaptability: The approach allows for adaptability to diverse protein structures and sequences, recognizing that different proteins may rely on various combinations of features to maintain stability.

1. **Energy Features:** Energy characteristics are important for assessing stability because they combine several energies such as the energies of Van der Waals interaction, the solvation energy, and extra-stabilizing energy that is free, all of which contribute to the unfolding energy of the target protein
   * Unfolding Energy: The energy associated with the unfolding of the protein structure. This includes the energy required to break interactions holding the native structure together.

* Van der Waals Interactions [26]: Energy arising from the attractive or repulsive forces between molecules due to fluctuations in electron distribution. Van der Waals interactions play a role in the stability of the protein's folded state(Pahari et al., 2020).
* Solvation Energy: The energy associated with the interaction between the protein and surrounding solvent molecules. It considers how water molecules surround and interact with the protein, influencing its stability.
* Extra-Stabilizing Free Energy: Additional energy contributions that stabilize the protein structure, which may include hydrogen bonding, electrostatic interactions, and other non- covalent forces.
* Total Free Energy Change: The summation of various energy components during protein unfolding, providing an overall measure of the energetic stability of the protein [27].

1. **Molecular Features:** These include the interface's solvent accessible surface area and its hydrophobic & hydrophilic areas, which serve as markers for stability prediction.

the surface area which is available of accessibility of solvent (the "SASA"): A protein's surface area that is accessible to solvent molecules. Changes in SASA can indicate alterations in the protein's exposure to the surrounding environment, influencing stability [18].

* + Hydrophobic and Hydrophilic Areas: Discrimination between regions of the protein that are hydrophobic (repellent to water) and hydrophilic (attracted to water). The balance between these areas contributes to the protein's stability.
  + Interface Characteristics: Molecular features at the interaction interfaces between protein subunits or with other molecules. The solvent accessible surface area of these interfaces can impact stability and binding interactions.
  + Conformational Dynamics: The flexibility and dynamics of the protein structure, including fluctuations in atomic positions. Understanding how the protein adapts to different conformations is vital for stability assessment [18].
  + Molecular Interactions: Examining the nature of interactions between amino acid residues, The bonding of hydrogen, electrostatic interactions, and the forces of van der Waals are a few examples. All of these interactions contribute to the protein's overall stability.
  + Hydration Patterns: How water molecules interact with the protein surface and contribute to stability. Changes in hydration patterns [28] can affect the protein's conformational stability [19].

procedures for Computational tools for predicting mutant protein stability

The process of predicting mutant protein stability involves three key steps, ensuring a comprehensive analysis:

1. Establishing a Protein and Mutant Database: Diverse stability prediction tools necessitate distinct databases, serving as templates with known structures and details. These templates are essential for comparing the structure and stability of a given sequence**.** for the list of databases supporting these tools.
2. Understanding Influential Factors in Mutant Protein Stability: A thorough comprehension of factors influencing mutant protein stability is achieved through comparing various structural and sequence features. This comparative analysis enhances our understanding of stability-affecting elements, offering valuable insights for addressing Protein stability issues provide a number of obstacles [19].
3. Protein Stability Prediction Following Mutation: Computational approaches and tools, accessible either online or as standalone applications, facilitate the prediction of protein stability following mutation. The information acquired through these tools becomes instrumental not only in stability prediction but also in the design of novel proteins and the exploration of diseases resulting from non-synonymous mutations [29]

The insights gained from these phases, supported by computational techniques, help forecast protein stability and lay the groundwork for the design of novel proteins and the research of diseases caused by non-synonymous mutations.

## 2.2 Structure Based Methods

Several structure-based tools are available for predicting the phenotypic effects of mutations. These tools leverage information about the three-dimensional structure of proteins to assess how mutations may impact protein function and, consequently, contribute to phenotypic changes. Here are some notable structure-based mutation prediction tools:

## 2.2.1 PolyPhen-2(Polymorphism Phenotyping V2)

PolyPhen-2, short for Polymorphism Phenotyping v2, has become well-known used computational method designed for Predicting the effect of amino acid alterations on protein structure and function. Developed by an organization of scientists, PolyPhen-2 serves as a valuable resource in the field of genetics and genomics by aiding researchers and clinicians in understanding the implications of genetic variations.

The core functionality of PolyPhen-2 [30] revolves around the analysis of single nucleotide variations (SNVs) in protein-coding regions of the human genome. These variations, often referred to as missense mutations, involve the substitution of one amino acid for another within a protein sequence. PolyPhen-2 aims to classify these mutations into categories indicating their potential impact on protein structure and function.

Here's an in-depth exploration of PolyPhen-2 and its working principles:

#### Input and Databases:

* + PolyPhen-2 takes protein sequences as input along with the amino acid changes caused by genetic variations.
  + It utilizes a variety of databases, including sequence databases and structural databases, to gather information about known protein sequences and structures.

1. **Feature Extraction:**

- The tool extracts various features from the input data, such as sequence conservation, physicochemical properties of amino acids, and structural annotations Click or tap here to enter text.

* Features related to the evolutionary conservation of a given amino acid position [30] across different species are particularly crucial.

1. **Machine Learning Model:**
   * PolyPhen-2 employs a machine learning model trained on a diverse set of features extracted from known disease-causing and benign variations.

* The model has been taught to differentiate from mutations that are likely to be harmful compared to those which are likely to be benign.

#### Prediction Scores:

* After the model is trained, it assigns a score to each amino acid substitution.
* The scores are typically categorized as "probably damaging," "possibly damaging," or "benign," providing users with insights into the potential impact of a specific mutation (Bromberg & Rost, 2007).

#### Output Interpretation:

* + Users receive a detailed output report, including prediction scores, supporting evidence from various features, and annotations.
  + This information aids researchers and clinicians in making informed decisions about the functional significance of a genetic variation [31].

#### Performance Evaluation:

* + PolyPhen-2's performance is often evaluated using benchmark datasets and comparison with experimental data.
  + Sensitivity, specificity, and accuracy metrics are commonly used to assess the reliability of its predictions.

#### Applications:

* + PolyPhen-2 has found applications in diverse areas, including human genetics [32], personalized medicine, and disease research.
  + It helps prioritize genetic variants for further experimental validation and guides researchers in understanding the potential molecular mechanisms underlying genetic diseases.

#### Updates and Enhancements:

* The tool undergoes periodic updates to incorporate new data and improve prediction accuracy.
* Enhancements may include the integration of additional features, optimization of the machine learning model, and addressing limitations identified in previous versions (Bromberg & Rost, 2007).

#### 2.2.2 SIFT (Sorting Intolerant from Tolerant):

SIFT has been used to analyses human variant databases, successfully differentiating between disease-linked mutations and neutral polymorphisms. We applied SIFT [33] to a database including missense substitutions associated with or implicated in diseases, assuming that "amino acid substitutions" causing disorders can be deleterious to protein function. SIFT projected that 69% of these substitutions would be harmful, according to the results. SIFT [33] found 25% of the non- synonymous single nucleotide polymorphisms (SNPs) in dBs, a collection of putative SNPs, as harmful. This is consistent with SIFT's false positive error rate of 20%, indicating that many non- synonymous SNPs are functionally neutral. Furthermore, a sample of dbSNP variations predicted to impair function were shown to be associated with illnesses, demonstrating SIFT's sensitivity in detecting functionally relevant variants. SIFT evaluates the particular position and nature of the amino acid modification when determining whether an amino acid replacement within a protein will affect its function. SIFT [33] finds similar proteins and provides an alignment between these proteins and the query sequence when given a protein sequence. SIFT computes the chance that a given amino acid at a specific site is tolerated using the amino acids present at each point in the alignment, because the most common amino acid at that position is also acceptable. If the normalized value resulting from this calculation falls below a predetermined threshold, the substitution is likely to be harmful (Ng & Henikoff, 2003).

#### Input and Databases:

Users can access predictions for amino acid changes of interest by going to ["http://ww](http://www.blocks.fhcrc.org/sift/SIFT.html)w[.blocks.fhcrc.org/sift/SIFT.html"](http://www.blocks.fhcrc.org/sift/SIFT.html). The website includes links to three submission alternatives, allowing users to keep control over prediction quality with varied levels of involvement. Users can submit their protein sequences and amino acid alterations in the most basic submission option. SIFT looks for homologous protein sequences related to the query protein on its own in a fully automated mode. SIFT [33] computes probability for each probable amino acid change based on these recognized sequences. For SIFT to execute the search, users can select from databases such as “SWISS-PROT”, “SWISS-PROT” “TrEMBL”, or “NCBI's non-redundant protein databases”. Click or tap here to enter text..

While SIFT selects sequences automatically, it is observed that higher prediction results may be obtained when all given sequences are orthologous to the query protein. The addition of paralogous sequences can complicate predictions, particularly for residues that are only conserved across orthologues. If users have sequences that are thought to be functionally comparable to the protein of interest, these sequences can be provided directly, avoiding SIFT's sequence selection stage. SIFT [33] proceeds to obtain the alignment required for prediction using the search query protein and its sequences that are homologous as input.Click or tap here to enter text..

#### Output

"All 20 possible amino acid" changes are predicted at each place within the protein, and the resulting alignment is displayed for users to review and edit as needed for resubmission. This function is useful for deleting ambiguous, incorrect and misalignment sequences from SIFT's automated realignment output.

A more complete summary is provided for user-submitted amino acids replacements, as seen in Figure 1. The score represents the normalized likelihood that the amino acid change would be tolerated, with SIFT classifying changes with scores below 0.05 as detrimental. When evaluating substitutions with scores less than 0.1, several users have found greater sensitivity for identifying harmful SNPs [34]. The quantitative score allows users to priorities amino acid modifications by separating them from least to most important. The confidence in a predicted detrimental substitution is affected by the alignment's diversity. SIFT may forecast most alterations to impact the function of proteins if the sequences utilized for prediction are closely linked, resulting in a larger false positive rate. To remedy this, SIFT computes the median conservation value, which reflects the diversity of the alignment's sequences. Information content conservation varies from log2, indicating complete conservation with only one observed amino acid, to zero, suggesting observation of all 20 amino acids at a site. SIFT generates alignments with a median conserved value of 3.0 by default. Predictions based on alignments that have higher median conservation values are less diversified and may be more prone to false positives. SIFT [33] surpasses naive predictions of non-conservative amino [35] acid changes as detrimental even when only a limited number of sequences with homology are available. SIFT predicts non-conservative changes, defined as those with scores that are negative in an “amino acid substitution” scoring a matrix, less well. SIFT's ability to correctly anticipate neutral changes improves even with only one sequence

homologous to the test protein. This emphasizes the necessity of taking into account the places that differ between the test protein and other sequences, as SIFT may indicate these positions as less important for protein function, increasing the selectivity for detrimental substitutions Click or tap here to enter text.

## 2.2.3 PROVEAN:

Identifying causative variations from a large pool of genome-wide variants is a significant difficulty. In response to this difficulty, computational methods such as “Polyphen” and “SIFT” were created to provide functionality predictions for "coding mutations". Numerous additional computational approaches, including ones developed recently, have arisen over time. The "Protein Variation Effect Analyzer (PROVEAN [36])", that utilizes an "alignment-based" scoring method, is a new addition to this repertory.

PROVEAN distinguishes itself by predicting not just single amino acid alterations but multiple amino acid substitutions, insertion, and deletion. This adaptability is enabled via a uniform scoring scheme, which allows for complete predictions over a wide range of genetic variants. Click or tap here to enter text.

#### Background

As previously explained and validated in 2012, the "PROVEAN" method employs an alignment- based scoring approach. An effective approach for computing pairwise alignment scores has been devised in the context of establishing a huge database of "precomputed PROVEAN" scores. This involves comparing the alignment scores of a specific protein sequence to a large number of "single-locus" variants discovered in another protein. PROVEAN []36, in essence, answers the critical requirement for a tool that can efficiently assess the influence of genetic variants on protein function. It stands out in the landscape of computational tools dedicated to variant analysis due to its unique ability to accommodate diverse forms of alterations, including insertions and deletions. As with any tool in this domain, the effectiveness of PROVEAN hinges on its scoring approach and its ability to predict the functional consequences of diverse genetic alterations accurately. The comprehensive nature of PROVEAN's predictions makes it a promising candidate for researchers and clinicians seeking a robust solution for variant interpretation in the context of diseases or specific phenotypes of interest.

#### 1. INPUT And Database:

The PROVEAN Genome Variants online server, which supports both human and mouse genomes, is powered by a proprietary database called MySQL. This database's key data fields include "Ensembl genes annotations", "precomputed PROVEAN [36] rankings", and precomputed supporting sequences for all of the human and mice protein sequences. It also provides supporting sequence sets derived from user-submitted first online submissions, which accommodate protein sequences from multiple organisms.To properly manage work submissions, the web server creates various queues that are customised to the projected job size. This guarantees that smaller activities are processed quickly on a designated queue, allowing for a quick turnaround and optimal performance. The database's architecture and the data structures that underpin each of the web server's three core operations. This provides a detailed explanation of how Ensembl gene annotations, "precomputed PROVEAN rankings", and supplementary sequences are stored and retrieved within the system. This systematic approach adds to the PROVEAN Genome Variants web server's efficiency and dependability, assisting users in analysing the functional impact of genetic variants across varied taxa [37].

#### 2. Working:

The "PROVEAN Protein tool" is a web interface of overall standalone PROVEAN software package that predicts the functional impact of protein sequences from various organisms. This tool accepts a protein sequence and amino acid variants as input and performs a search using "BLAST" to find "homologous sequences (which supports sequencing)" and generates "PROVEAN" rankings. Predictions for a specific protein query typically take between ten and twenty minutes. For greater efficiency, a cached strategy was introduced to speed up time- consuming tasks such protein database search, homologous sequence collecting, as well as clustering".

Supporting sequences for initial protein queries, as well as a list of sequence identifiers and clustering information, are saved in the database using the caching strategy [38]. This information is indexed using the query protein sequence and then reused in subsequent prediction queries. The use of this caching strategy greatly reduces the time necessary for the development of intermediate results.

The “PROVEAN Protein Batch" function allows for batch processing of a variety of protein variants based on precomputes. Users provide a list of protein variants, each of which includes a public protein identification, amino acid position, reference amino acid, and variant amino acid. The programmed guarantees a quick response time by using two methods: precomputed scores and precomputed supporting sequences. "PROVEAN" scoring are received directly from a database with precomputed scores, covering Twenty single amino acids changes and one amino acid deletion for every position in about 90,000 human and 46,000 mouse proteins. The 'PROVEAN Genome Variants' function allows for batch processing of a large number of variants across the entire genome. Based on the reference genome sequence and Ensembl gene annotation, this programme accepts a list of genomic changes [39], including single nucleotide polymorphisms (SNPs), multiple nucleotide substitutions, insertions, or deletions, and classifies them as coding or non-coding. PROVEAN scores and predictions are derived similarly to the 'PROVEAN Protein Batch' function, and coding variants are further classified at the protein sequence level. PROVEAN scores are organized at the protein sequence level rather than the genome level for optimal data storage. Real-time dynamic conversion ensures that coordinates from genomic variants to corresponding protein variants are converted instantly. The original Ensembl gene annotation in GFF/GTF [40] format is reorganized for efficient coordinate conversion, with each record storing essential information such as chromosomal position, underlying nucleotide, codon triplet, reading frame, codon position in amino acid, gene ID, and gene orientation. For the genomic variation input data, additional information is provided, including NCBI dbSNP reference accessions and gene annotation from Ensembl BioMart. Click or tap here to enter text..

#### 3. Output:

Because there are potentially infinite combinations of numerous amino acid substitutions occurred, additions, and eliminations the viability of a precomputed ranking methodology is limited across different types and length of amino acids acid modifications. As a result, for each protein sequence in the supported organisms, a supporting sequence set is precomputed and saved. These supplementary sequence sets are critical in rapidly computing [41] predictions for amino acid variants that are not covered by the precomputed score technique. The scores generated by this method are saved in the data base, allowing for efficient reuse for subsequent queries involving the same variants. This methodology optimises the computational process by establishing a

compromise between the complexities of various amino acid modifications and the need for quick and accurate forecasts.

## 2.2.4 SNAP

While experiments can deem to be more accurate, it is frequently more time-consuming than computational investigations. Computational predictions of non-synonymous single nucleotide polymorphism (nsSNP) effects can enhance the efficiency and speed of experimental studies by providing pre-filtering. This pre-filtering can prove beneficial not only in the context of mouse genetics studies but also in unraveling "Mono-SNP" and complex phenotype- issues affecting humans, as well as evolutionary genetics. Computational methods, while unlikely to completely replace wet-lab experiments in terms of accuracy, might be useful in choosing and prioritizing only a couple of viable candidates from large databases. Recent studies have demonstrated that computational evaluation of protein character changes associated with nsSNPs can yield reliable estimates of their functional effects. These studies have employed diverse classification approaches, including the use of "support vector machines", "decision trees", "neural networks", “random forests”, "Bayesian models", and "statistical approaches" are examples of methods used in machine learning.as well as rule-based systems [42]. Some approaches are applicable to all sequences and variants based on mathematical computations from alignments (e.g., SIFT), whilst others require particular information such as the coordinates of three-dimensional protein structures (e.g., SNPs3d). Certain algorithms, such as "PolyPhen", use a variety of information sources to increase classification accuracy, such as "3D structures", "SWISS-PROT" annotations, and alignment.Despite these advances, predicting the functional impact of "nsSNPs" remains a difficult task. "SNAP (Screening for Non-Acceptable Polymorphisms)" is a promising approach that uses sequence-based computationally collected data to possibly classify all nsSNPs as non- neutral (having an influence on function) or neutral (having no effect). Only information. "SNAP", a "neural network" derived tool, includes "evolutionary [43] information" ("residue's conservation" between sequence families), anticipated protein structure characteristics ("secondary structure, solvent accessibility"), and other relevant information acquired only from sequence data. "SNAP" outperforms its competitors in terms of accuracy and coverage, delivering trustworthy predictions and a precisely calibrated assess the dependability index. The score measures the degree of certainty in a specific forecast [44], allowing users to

focus on predictions more likely to be correct and to identify instances where a predicted deleterious effect is associated with low confidence “SNAP's” broad application includes sequences of every organism, proteins both with and without known structures, and unique "Snp in understudied and unannotated families. In conclusion, "SNAP" marks a substantial progress in predicting the functional impacts of nsSNPs, providing users with a useful tool for prioritizing candidates and assessing the possible impact of genetic variants. Click or tap here to enter text..

## INPUT DATA AND DATABASES:

The "Protein Mutant Database", a comprehensive derived database providing data from experiments on protein mutations and their effects, was used to generate the list of single amino acid changes. Qualified descriptions such as'significant decline in function' ('-'), 'no change' ('='), or 'enhanced affinity' ('+') are frequently included in PMD entries. When a mutant occurred numerous times with various requirements, it was classified as non-neutral. In contrast, just one instance of the mutation was assigned to the class matching to the effect's specified qualification. Non-neutral mutations were those linked with any physiological change, despite the direction or amplitude of the signal. The collection included "4,675" sequences of proteins with "54,975" single residue alterations, of which "14,334" had no effect comparing with the normal variant and 40,641 had an effect. To address the need for a larger dataset of neutral mutations, an additional dataset was created from SWISS-PROT [45], focusing on enzymes with experimentally annotated functions. The method involves treating residues as neutral if they differed in an alignment of two enzymes with the same experimentally annotated function (identical "EC" number). Annotations labelled 'by similarity,' 'by homology,' 'hypothetical,' 'putative,' or 'fragment' were eliminated, and the dataset was limited to sequence-similar enzymes to improve reliability. Only pairs that satisfied the requirement for sequential identity of sequences (less than or equal to 40%) and "HSSP" value (higher than or equal to [40] [45]) were included in the pairwise BLAST alignments on all experimentally annotated enzymes. It was considered that amino acids that differed between two aligned sequences were unlikely to change function. This method yielded a data set of "2,146" sequences, with "26,840" benign pseudo-variants contributing. Click or tap here to enter text..

#### Prediction method

As previously described, that we used typical "feed-forward neural networks" that had a "momentum term". On the sets of training (for optimising connections) and cross-training sets (for optimising architectures and halting training), all free parameters of the networks were optimised. Support vector machines (SVMs) were also investigated, however they performed marginally worse in our studies. SVMs were trained via identical features used in the best performing neural networks, plus certain free parameters optimized [46] on the cross-training data set. Despite its great accuracy, the SVM-based method fared somewhat worse than the comparable “neural network-based” approach. The networks of neural networks recorded the effects of the immediate local sequence environment on mutations by using windows of consecutive residue as inputs. The resulting windows are symmetrical in terms of the focal location of the single nucleotide polymorphism. Window lengths ranging from one (mutant on its own) through twenty-one residues were tested. Several protein properties were considered, representing qualities that could influence the outcome of an alteration [47]. These characteristics were chosen based on past publications and expertise in creating prediction techniques.

#### Biochemical Properties:

* + Changes that are likely to influence protein structure, such as burying an energized residue, integrating the rigid proline within an "alpha-helix", or changing its hydrophilic/hydrophobic character on a sidechain.
  + Overpacking and generating an opening in the core of a protein by modifying the dimensions of the residue are examples of factors.
  + Changes in the presence or lack of Cb-branching, as well as the mass of wild-type varieties. and variant residues. For all of the initial 5 features, a single input node was added to the network design to represent the change in value or category as a result of the mutation. The magnitude of change denoted the severity of class changes (e.g., hydrophobia, charge, then and mass). Just a single boolean a node ("input = 0 or 100") indicated the existence of buried charge, an alteration of "Cb-branching", or the transformation of the amino acid proline onto a the "Alpha-helix". The input for mass change equalled the mass difference between the wild type and mutant. This set of 6 connections represented a minimum amount of network features.

#### Evolutionary data:

* The conservation of a residue within a family with similar proteins was encoded employing "PSI- BLAST" output and a mixture of weighted protein amino acid frequencies as well as Position- Specific Scoring Matrix ("PSSM") vectors.
* For each representation type, data per location and the proportional weight of gapless genuine matching over "pseudo-counts" were supplied.

#### Position-Specific Independent Counts (PSIC):

- Profiles generated by PSIC were considered as an alternative to PSI-BLAST PSSM data [48].

* PSIC entails assembling "position-specific weights" while taking into account the overall level of similarity in sequence within matched protein. These representations were utilized in a variety of ways to encode evolutionary information, to capture the impact of mutations on protein structure and function. Each neural network was trained on a mixture of these properties to anticipate the impact of missense mutations.

#### 2.3 RESULTS:

Identifying the most effective input features for predicting the functional impact of SNPs was challenging due to two main factors. Firstly, the data were inconsistent, with variations in annotation across proteins, affected tissues, and experimental laboratories. Secondly, the datasets were relatively small, and the diversity within biology could not be entirely covered by around 80,000 mutants, which is essentially the goal of a prediction system [49]. Given the limited size of the dataset, It was not possible to thoroughly evaluate all conceivable input features. To avoid overfitting, the data were initially separated into subgroups susceptible to distinct types of changes. Changes from being polar to a non-polar amino acid, for example, can be "non-neutral" within the Centre of a protein yet neutral on the surface."Pfam" information, "PSIC" results, estimated residue freedom (PROFbval), as well as transition frequencies (likelihood of observing a specific variation implemented by an SNP) were all considered in three different subsets, focusing on buried, intermediate, and exposed residues. The resulting "neural network models" included from 195 input and fifty concealed units, with parameters optimized on a cross-training set rather than the performance test sets presented. While other methods demonstrated similar levels of accuracy, further analysis revealed that rushed inferences were made. SNAP annotated correctly predicted instances where SIFT or PolyPhen were wrong approximately 1.7 times more often than the

reverse scenario "SNAP annotated" obtained a total precision off 62 in a significant portion of the PMD dataset (16,241 mutant) where a minimum of one approach was erroneous, surpassing "PolyPhen" (7966" accurate & 8275 inaccurate) and SIFT ("7566" correct and 8675 incorrect), both of which achieved levels below 50% accuracy.

## 2.4 Sequence Based Tools:

**2.4.1 Mutation Taster:**

The automatic pre-evaluation of sequence variants is critical in steering further in-depth analyses towards most worthy candidates, conserving both money and time [50]. Despite their importance, present evaluation methods are limited to predicting just what will happen to the amino acid changes that confront difficulties in processing thousands of inquiries in a suitable timescale.

In order to satisfy our needs for handling "high-throughput sequencing data", which we created "MutationTaster", an inexpensive, internet-based programme for quickly determining the disease- causing potential of DNA sequence mutations. MutationTaster integrates data from many biomedical databases and using well-established analysis tools. The studies take into account "evolutionary conservation", "splice-site" modifications, protein feature loss, and adjustments which may affect mRNA levels. The data are then evaluated using a "naive Bayes classifier", which predicts disease probability. Notably, an average query takes less than 0.3 seconds to complete. "MutationTaster" offers 3 separate prediction models depending on the type of alteration: those aiming at silent synonymous or "intronic modifications", those impacting a single amino acid, and those generating complicated changes in the amino acid sequence.The classifier is trained using a dataset comprising common polymorphisms and known disease-causing mutations. Cross- validation of the classifier yielded an overall accuracy of 91.1 ± 0.1%. When analysing disease- linked mutations and polymorphisms producing single amino acid transfers, MutationTaster surpassed similar tools in terms of both accuracy and speed. MutationTaster includes a easy-to-use online user interface for analysing single alterations or in batch mode. To facilitate the streamlined and standardized analysis Perl scripts are supplied for the analysis of Next-Generation Sequencing (NGS) information obtained from numerous platforms such as the "Roche 454", "Illumina Genome Analyzer", and "ABI SOLiD". This capability allows for the effective screening the NGS data for changes with an elevated potential for producing disease.However, the software has a few

drawbacks, including being unable to evaluate insertion-deletions with more than 12 base pairs

[50] and modifications across an "intron-exon border". Furthermore, non-exonic changes are currently limited to the "Kozak consensus sequence", "splice sites", and "poly(A) signal". Future upgrades will include checks for other sequence motifs. MutationTaster is available to users at [[http://www.mutationtaster.org/]](http://www.mutationtaster.org/) [(http://ww](http://www.mutationtaster.org/))w[.mutationtaster.org/).](http://www.mutationtaster.org/))

#### 2.4.2 FATHMM (FUNCTIONAL ANALYSIS THROUGH HIDDEN MARKOV MODEL)

Recent advancements in the HMMER3 [51] software suite have significantly enhanced the computational prediction of the functional effects of Amino Acid Substitutions (AASs) using Hidden Markov Models (HMMs). This progress is exemplified through two distinct methods aimed at improving accuracy and applicability. The first method introduces an unweighted/species- independent approach. Through an iterative search technique, sequences that are homologous can be collected automatically then aligned. The Multiple Sequence Alignment ("MSA") that results are then used for developing an "ab initio HMM". The internal match states of the model are used to assess sequence conservation. Furthermore, sequence conservation is investigated inside fully curated HMMs representing homologous protein domain families such as "SUPERFAMILY" and Pfam. This domain-based analysis complements the automatically collected alignment, capturing structural and evolutionary constraints that may be overlooked.

The second method employs a "weighted/species-specific methodology" that includes "pathogenicity weights." The relative frequencies of disease-associated and functionally neutral AASs mapped into conserved domains of proteins are used to calculate these weights. In two different trials, using a model weighted for human mutations outperforms established prediction methods such as "SIFT", "PolyPhen", and "PANTHER” [13]. Notably, the model's accuracy beats existing state-of-the-art prediction approaches, especially "SNPs&GO" [14] as well as MutPred [52]", using one evaluation. This method, known as "Functional Analysis Through Hidden Markov Models "("FATHMM"), not only demonstrates predictive abilities on multiple benchmarks representing human mutations, but also practical application to large datasets of non-synonymous Single Nucleotide Polymorphisms (nsSNPs) in wheat (Triticum spp.). This application aids in the identification of important genetic variants responsible for phenotypic changes introduced during domestication by severe selection. The findings of this study have been made available to the

public community, demonstrating FATHMM's versatility and applicability in processing "high- throughput large-scale genomic data while presenting connections to phenotypic outcomes.

#### INPUTS:

We acquired 5 adult mutation datasets for this investigation from various internet resources and literature sources (see Table 1 for more information). Initially, hereditary disease-causing Amino Acid Substitutions (AASs) were acquired from the "Human Gene Mutant Database" ("HGMD") (November 2011; ["http://ww](http://www.hgmd.org/)w[.hgmd.org").](http://www.hgmd.org/) In addition, pathogenicity weights were calculated using inherited hypothetical biologically benign "AASs" from the "UniProt database" [("http://www.uniprot.org/docs/hum](http://www.uniprot.org/docs/humsavar)sa[var](http://www.uniprot.org/docs/humsavar)"), which were then applied in our "weighted/species-specific technique". To assess "FATHMM's" performance, 2 human mutant datasets were obtained and compared to other computational prediction techniques reported in the literature. Initial dataset was obtained from the "VariBenchdatabase" [("h](http://bioinf.uta.fi/VariBench)tt[p://bioinf.uta.fi/VariBench"),](http://bioinf.uta.fi/VariBench) and it was used in a thorough examination of nine different computer predictive methods. The second dataset included "267 AASs" from 4 genes linked to cancer ("BRCA1, MSH2, MLH1, and TP53"), which were used within a newest evaluation consisting of four alternative predictive computational techniques. A further human mutation dataset was obtained from the Swiss Var site [("http:/](http://swissvar.expasy.org/)/[swissvar.expasy.or](http://swissvar.expasy.org/)g"). This dataset included both diseases linked and potentially biologically neutral AASs. In this context, "FATHMM" was subjected to a completely independently test against 8 different computational prediction techniques.

#### Results:

As previously stated, "FATHMM" not only forecasts the potentially harmful nature of "Amino Acid Substitutions" (AASs), yet it can also annotate the biochemical and phenotypic [53] repercussions of these changes using domain-centric ontologies. To demonstrate this, we looked at the expected phenotypic outcomes of disease-associated AASs in the "Swiss Var dataset"."FATHMM" accurately identifies the phenotypic effects of well-characterized illnesses, as expected. For example, domain-based ontological relationships properly identify the cardiovascular implications of the "C1971Y" mutation in "FBN1" ("Marfan syndrome; MIM# 154700"). However, when using subdomain-centric the ontologies it is critical to recognize potential problems, especially when a shared domain accommodates several mutations with

diverse and uniquely manifested symptoms. Domain-centric ontological linkages can become weakened in such settings, and vigilance should be maintained. The projected phenotypic effects of the "R239C" mutation in CHRNG ("Escobar syndrome") [54], for example, show the importance of cautious interpretation in cases when a single domain harbors numerous mutations with various phenotypic outcomes. Click or tap here to enter text.

#### 2.5 (a) DISADVANTAGES OF STRUCTURE BASED METHODS:

Structure-based methods for predicting the phenotypic effects of mutations can provide valuable insights but have inherent disadvantages and limitations:

#### Reliance on Structural Information:

Limited Data Availability: These methods heavily depend on experimentally determined three- dimensional protein structures, posing challenges for less-studied proteins or those resistant to crystallization.

Incomplete or Inaccurate Information: Inaccuracy or incompleteness in structural data can compromise prediction accuracy, especially with unresolved regions or inaccuracies in the structural model.

#### Applicability Constraints to Non-Globular Proteins:

These methods are typically optimized for globular proteins with well-defined folds, potentially limiting their effectiveness for proteins with intrinsic disorder, membrane proteins, or large macromolecular complexes.

#### Sensitivity to Mutational Distance:

Prediction accuracy is influenced by the proximity of the mutation to active sites or functionally relevant regions. Mutations closer to critical sites might be better characterized, potentially leading to inaccuracies in predicting long-range effects.

#### Dynamic Nature of Proteins:

Proteins exhibit dynamic behavior, and their function relies on conformational changes. Structure- based methods may not fully capture dynamic aspects, particularly when mutations affect flexibility or dynamics rather than static structure.

#### Resource Intensiveness and Time Constraints:

Conducting structure-based predictions often demands significant computational resources. Complex algorithms for simulating protein dynamics or molecular dynamics simulations can be time-consuming and computationally expensive [56].

#### Inability to Address Multiscale Effects:

Structure-based methods may struggle to comprehensively address multiscale effects, especially when mutations have cascading impacts on various biological levels, such as pathways, cellular processes, or organismal phenotypes.

#### Handling Complex Molecular Interactions:

Traditional structure-based methods may not adequately capture aspects like protein-protein interactions, ligand binding, or involvement in larger molecular complexes, limiting their ability to predict broader phenotypic consequences.

#### Functional Impact Prediction Challenges:

While these methods can predict structural changes caused by mutations, linking these changes to functional consequences presents challenges. Predicting how alterations in structure translate into changes in protein function is not always straightforward.

#### (b) DISADVANTAGES OF SEQUENCE BASED METHODS

* 1. **Lack of Structural Insight:**

Relying solely on amino acid sequences, these methods do not consider three- dimensional structural information, limiting their ability to capture changes in protein conformation or interactions.

* 1. **Insensitivity to Long-Range Interactions:**

Difficulty in effectively incorporating long-range interactions between residues may hinder understanding the functional impact of mutations.

* 1. **Challenges in Predicting Structural Changes:**

Unlike structure-based approaches, sequence-based methods may struggle to predict alterations in protein structure, such as changes in secondary structure elements or solvent accessibility.

* 1. **Compensatory Changes Difficulty:**

Identifying compensatory changes, where a mutation is neutral on its own but interacts with others to influence phenotype, poses challenges for sequence-based methods.

* 1. **Limited Applicability to Non-Coding Regions:**

Non-coding regions, such as introns or regulatory sequences, are inadequately characterized by sequence-based methods, making it inherently challenging to predict the phenotypic effects of mutations in these regions.

* 1. **Sensitivity to Alignment Quality:**

The accuracy of predictions is heavily reliant on the quality of sequence alignments, and inaccuracies or gaps in alignments can lead to erroneous predictions.

* 1. **Overlooking “Post-Translational Modifications”:**

“Post-translational modifications”, such as “phosphorylation” or “glycosylation”, which significantly influence protein function, may be overlooked by sequence- based methods.

* 1. **Difficulty in Identifying Functional Domains:**

Identifying specific functional domains based solely on sequence information can be challenging, and inaccuracies in domain prediction may impact overall prediction accuracy.

* 1. **Challenges with Homologous Proteins:**

Accuracy may decrease when dealing with proteins lacking close homologs with well-characterized functions, as the functional impact of mutations is often inferred from homologous sequences.

* 1. **Neglecting Epistatic Interactions:**

Sequence-based methods may not effectively consider epistatic interactions, where the combined effect of multiple mutations differs from the sum of their individual effects.

* 1. **Difficulty in Predicting Multifunctional Proteins:**

For proteins with multiple functions, sequence-based methods might face challenges in predicting how mutations affect each specific function, as these methods often provide a global assessment.

**12. Limited Insight into Conformational Changes:**

These methods may lack detailed insights into conformational changes induced by mutations, missing critical information on how alterations impact protein function. There are various types of limitations which is till now in these computational approaches are lack of training data because these tools do not have the sufficient data which they require for training the model so the prediction ability is compensated and sometimes the data is biased so the prediction not done properly. The current models or tools do not give better results like they only predict the mutant protein will be functionally deleterious or do not affect the protein sequence. The current existing tools have a high probability of predicting the wrong amino acid effect high False positive rate because they sometimes predict an amino acid substitution which is not functionally deleterious and considers it also. more features hinder sometimes each other which told that sometimes one that sometimes the data as a curse of dimensionality, overfitting, feature Redundancy increased complexity, better prediction is only done when the mutant protein has the structure available and has sequence annotation and structural features, but we do not have all protein structures available. uniprot contain only 0.2% so the structural features available can only be calculated for them.

# MATERIAL AND METHODS:

#### Data Collection and Preprocessing

* + - 1. **Dataset Source**

The dataset utilized in this study is a comprehensive and meticulously curated collection obtained from Zenodo, a reputable open-access repository that ensures the availability and reproducibility of scientific data. This dataset encompasses a staggering total of 217,532 mutations distributed across 800 different proteins, each with experimentally determined Gibbs free energy changes (ΔΔG). The breadth and depth of this dataset make it an invaluable resource for understanding the intricate relationship between protein mutations and their thermodynamic stability. Gibbs free energy change (ΔΔG) [27] is a critical parameter in protein biochemistry, providing insights into the stability alterations induced by mutations. ΔΔG measurements are pivotal for understanding the impact of specific mutations on protein folding, function, and interaction with other molecules. In this dataset, ΔΔG values are experimentally determined, ensuring a high degree of accuracy and reliability. The experimental determination of ΔΔG typically involves techniques such as differential scanning calorimetry (DSC), isothermal titration calorimetry (ITC), and various spectroscopic methods. These techniques allow for precise quantification of the energetic differences between the wild-type and mutant proteins, thereby offering a robust foundation for further computational and theoretical analyses. The wild-type protein structures in this study were sourced from the Protein Data Bank (PDB), an indispensable resource for structural biology. The PDB houses a vast repository of three-dimensional structural data of biological macromolecules, primarily proteins and nucleic acids, determined through experimental methods like X-ray crystallography [28], nuclear magnetic resonance (NMR) spectroscopy, and cryo-electron microscopy (cryo-EM). The structural data from the PDB provides a critical context for understanding the spatial configuration of atoms within a protein, which is essential for analyzing how mutations may alter protein structure and function. The combination of ΔΔG values and high- resolution structural data from the PDB enables a comprehensive examination of the molecular underpinnings of protein stability. By integrating these datasets, researchers can conduct detailed structure-function analyses, identify regions of the protein that are particularly sensitive to mutations, and develop predictive models of protein stability [30]. These models are crucial for a wide range of applications, from protein engineering and drug design to

understanding the molecular basis of diseases caused by protein misfolding and instability. Moreover, the dataset's extensive coverage of mutations across a diverse array of proteins allows for broad generalizations and the identification of common principles governing protein stability. It facilitates the study of both conservative and radical mutations, providing a holistic view of how different types of amino acid substitutions affect protein stability. For instance, conservative mutations, which involve substitutions between amino acids with similar properties, may have subtle effects on protein stability, while radical mutations can lead to significant destabilization or even gain-of-function changes. In addition to serving as a rich resource for empirical analysis, the dataset also offers a fertile ground for the application of machine learning and other advanced computational techniques. Predictive algorithms can be trained on this extensive dataset to forecast the impact of novel mutations, thus aiding in the design of proteins with desired stability profiles. Furthermore, these computational models can be used to screen for potentially deleterious mutations in genomic data, contributing to personalized medicine approaches where interventions can be tailored based on an individual's unique genetic makeup. Overall, the Zenodo dataset, combined with structural data from the PDB [32], represents a powerful tool for advancing our understanding of protein stability. It supports a wide range of scientific endeavors, from fundamental research into protein chemistry to practical applications in biotechnology and medicine. The dataset's comprehensiveness and the accuracy of the ΔΔG measurements provide a solid foundation for ongoing and future studies aimed at unraveling the complexities of protein structure and function. Wild-type PDBs.

The wild-type protein structures in PDB format were downloaded from the RCSB PDB [15] website. Each PDB file contains atomic coordinates and structural information for the respective proteins.

#### Mutant PDB Generation

To create mutant PDB files for each mutation in the dataset, a detailed and automated approach was implemented using Python scripting and YASARA software. A Python script was developed to read the dataset and generate mutation macros for the 217,532 mutations [34], specifying the amino acid changes to be made. These macros were then executed by YASARA, which loaded the wild-type PDB structures, applied the specified mutations, and saved the resulting mutant structures in PDB format. The process was automated for efficiency, utilizing batch processing and

parallel execution to expedite the workflow. This comprehensive procedure, taking approximately two days, resulted in two organized folders: one containing the wild-type PDB files and another with the mutant PDB files, systematically named for easy identification. The integration of Python and YASARA ensured accurate, reproducible, and efficient generation of a vast number of mutant protein structures, providing a valuable dataset for further research.

#### Graph Construction

* + 1. **Graph Representation**

In the context of this study, protein structures, both wild-type and mutant, were represented as graphs to facilitate a detailed and computationally efficient analysis of their properties and interactions [40]. This graph-based representation enables the application of advanced graph-based machine learning techniques, particularly Graph Neural Networks (GNNs) [16], to predict the effects of mutations on protein stability.

#### Nodes

In our graph representation, each node corresponds to an amino acid residue within the protein structure. This abstraction allows us to leverage the inherent connectivity and interactions between residues to better understand the protein's overall behavior and stability. Each amino acid, being a distinct node, carries specific attributes that can influence the protein's structure and function. These attributes are captured in the node features, which are detailed in the subsequent section.

#### Edges

Edges in the graph represent the interactions between amino acid residues. To define these interactions, we employed a distance-based criterion: two residues are considered to interact if the distance between any pair of their atoms is within a specified cutoff. For this study, a cutoff distance of 5.0 Å (angstroms) was chosen, which is a common threshold used to capture significant non-covalent interactions such as hydrogen bonds, ionic interactions, and van der Waals forces. These interactions are critical for maintaining the structural integrity of the protein and for facilitating functional conformational changes.

#### Node Features

Each node in the graph, representing an amino acid residue, was characterized by a set of nine physicochemical properties. These properties were chosen because they capture essential aspects of the amino acids that influence protein stability and function. The properties are:

* + - * 1. **Hydrophobicity:** This property indicates the tendency of an amino acid to avoid water. Hydrophobic residues typically reside in the interior of the protein, stabilizing the core structure.
        2. **Charge**: Amino acids can be positively charged, negatively charged, or neutral, affecting their interactions with other residues and their overall contribution to the protein's electrostatic potential.
        3. **Molecular Weight**: The molecular weight of an amino acid influences the protein's mass and can impact its folding and stability.
        4. **Hydrophobicity\_Hh:** Another measure of hydrophobicity based on the Hopp-Woods scale, providing additional insight into the amino acid's behavior in aqueous environments.
        5. **VSc:** Van der Waals volume of the side chain, which contributes to the steric effects in the protein structure.
        6. **p1**: Principal component 1 from a principal component analysis (PCA) of amino acid properties, summarizing a key aspect of variability among amino acids.
        7. **p2**: Principal component 2 from PCA, providing another dimension of variability among amino acid properties.
        8. **SASA (Solvent Accessible Surface Area):** This measures the surface area of an amino acid residue that is accessible to a solvent, indicating its exposure and potential interactions with the environment.
        9. **NCISC:** Normalized Conformational Index for Secondary Structure, which describes the propensity of an amino acid to be in a particular secondary structure (e.g., alpha-helix, beta-sheet).

These features were assembled into a feature matrix for each graph, where each row corresponds to a node (amino acid) and each column corresponds to one of the physicochemical properties.

This matrix serves as a comprehensive descriptor of the protein’s residues, facilitating detailed analysis and prediction.

#### Adjacency Matrix

The adjacency matrix is a binary matrix used to represent the presence or absence of interactions (edges) between amino acids (nodes) based on the distance cutoff. For a protein graph with \(n\) amino acids, the adjacency matrix \(A\) is an \ (n \times n\) matrix where:

\ [ A\_{ij} = \begin{cases}

1 & \text {if the distance between residue} i \text {and residue} j \text {is} \leq 5.0 \text {Å} \\ 0 & \text{otherwise}

\end{cases} \]

This matrix encapsulates the connectivity of the protein structure, with 1s indicating interactions and 0s indicating no interaction. The adjacency matrix is crucial for graph-based machine learning models as it defines the structure over which convolutional operations are performed.

#### JSON Files

To facilitate efficient data storage and retrieval, the feature and adjacency matrices for each protein structure were stored in JSON files. This choice of format offers several advantages:

* Human-Readable: JSON files are easy to read and understand, making them convenient for manual inspection and debugging.
* Lightweight: JSON is a lightweight data-interchange format, which is particularly useful for handling large datasets.
* Compatibility: JSON is compatible with many programming languages and libraries, simplifying the data loading and processing pipeline.

The JSON files were organized into two main directories: one for the wild-type protein structures and another for the mutant protein structures. Each JSON file contained two main components:

* Feature Matrix: A nested list where each sub list corresponds to the feature vector of a residue.
* Adjacency Matrix: A nested list representing the binary interaction matrix.

This organization ensured that data could be quickly and efficiently accessed during model training and evaluation.

#### Model Architecture and Training

To predict the Gibbs free energy changes (ΔΔG) resulting from protein mutations, we designed and implemented a Graph Attention Neural Network (GANN). This advanced neural network architecture leverages the rich information encoded in the graph representations of the protein structures.

#### Graph Attention Neural Network (GANN)

The GANN architecture used in this study is designed to capture both local and global structural features of proteins. The network consists of two layers of graph attention convolutions, which are

particularly adept at handling the inherent irregularity and complexity of graph data [56]. The key components of the GANN architecture are:

* Attention Mechanism: The attention mechanism allows the network to focus on the most relevant parts of the graph when performing convolutions. This is achieved by assigning different weights to different edges based on their importance, which is learned during training.
* Multiple Attention Heads: The first layer of the GANN uses multiple attention heads to learn different aspects of the node features. Each attention head operates independently and captures unique patterns, which are then concatenated to form a rich representation of the nodes.
* Graph Attention Convolution: In the graph attention convolution, each node aggregates information from its neighbors based on the learned attention weights.
* Final Layer: The second layer combines the outputs from the multiple attention heads to produce a final prediction for each node, which in this context is related to the change in stability (ΔΔG).

#### Training Process

The training process [58] for the GANN involved several critical steps to ensure accurate and reliable predictions of ΔΔG:

#### Data Loading

The first step in the training process [58] was loading the JSON files containing the graph representations of the protein structures. This step involved parsing the JSON files to extract the feature matrices and adjacency matrices for both wild-type and mutant proteins. Efficient data loading mechanisms were employed to handle the large dataset size, ensuring smooth and timely access to the necessary data during training.

#### Data Pairing

Next, pairs of graphs were created for each mutation. Each pair consisted of a wild-type graph and a corresponding mutant graph, representing the protein before and after the mutation, respectively. These pairs were labeled with the experimentally determined ΔΔG value from the dataset. This pairing process was crucial for supervised learning, providing the model with the necessary

information to learn the relationship between structural changes and stability changes. 3.2.3 Graph Data [56] Conversion

The feature and adjacency matrices were then converted into a format compatible with the graph neural network framework [16]. This conversion involved several steps:

* Normalization: The feature matrices were normalized to ensure that the input values were on a comparable scale, which helps stabilize and accelerate the training process [58].
* Batch Processing: The data was organized into batches to optimize the training process. Batching allows for efficient utilization of computational resources and ensures that the model is trained on diverse examples in each iteration, promoting better generalization.
* Graph Data [56] Structures: Specialized data structures designed for graph-based computations were used to store the adjacency and feature matrices. These structures are optimized for the operations performed by graph neural networks [16], such as sparse matrix multiplications.

**With the data prepared, the GANN was trained using the following steps:**

* Initialization: The network weights were initialized using a method that ensures stable forward and backward passes through the network.
* Loss Function: A suitable loss function, such as Mean Squared Error (MSE) [59], was employed to measure the discrepancy between the predicted and actual ΔΔG values.
* Optimization: An optimization algorithm, such as Adam, was used to adjust the network weights based on the gradients computed during backpropagation. The learning rate and other hyperparameters were carefully tuned to achieve optimal performance. Regularization: Techniques like dropout and weight decay were applied to prevent overfitting, ensuring that the model generalizes well to unseen data. Evaluation: The model's performance was regularly evaluated on a validation set to monitor.
* Node i (hi)



Weighted Sum of Neighbors' Features

ℎ𝑛𝑒𝑤 = 𝜎 ቌ ෍ 𝛼

𝑖

𝑖𝑗

ℎ′ቍ

𝑗

𝑗∊𝑁𝑖

Linear Transform

hi =W×hi

Concatenate hi’ & hj

LeakyReLU(aT[hi’||hj’])

eij (Attention)

Softmax Normalization

exp൫𝑒𝑖𝑗൯

𝖺𝑖𝑗 =

෌ expሺ𝑒 ሻ

Non-linear Activation

𝜎ℎ𝑖⬚{𝑛𝑒𝑤})

#### 3.3.2.4 Model Training and Validation

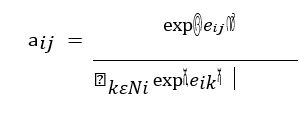
1. **Linear transformation:** Each node’s feature vector **hi** is first linearly transformed using a learnable weight matrix **W:**

**h’I =Whi**

1. **Attention coefficient calculation:** The attention coefficient eij between node I and j is computed using shared attention mechanism **a.** This mechanism concatenates the transformed feature vectors of the nodes,applies a non-linear activation(typically LeakyReLU), and computes the dot product with the attention vector **a:**

**Eij= LeakyReLU (aT [hi’||h’j])**

Where ||denotes concatenation.

1. **Softmax Normalization [55]:** The attention coefficients are then normalized attention using the softmax [55] function to ensure that they sum to one across the neighborhood of node *i*: 

**4. Neighborhood Aggregation:** The normalized attention coefficients are used to compute a weighted sum of the neighbor’s features:

ℎ𝑛𝑒𝑤 = 𝜎 ቌ ෍ 𝛼 ℎ′ቍ

𝑖 𝑖𝑗 𝑗

𝑗∊𝑁𝑖

#### 3.4 Model Components

**Graph Attention Mechanism:**

Graph Attention Networks (GATs) are particularly suited for learning from graph-structured data due to their ability to model dependencies between nodes effectively. Key components of GAT include:

* + **Attention Mechanism:** Instead of uniformly aggregating neighbor node information, GATs use attention mechanisms to selectively weigh contributions from neighboring nodes based on learned attention coefficients. This allows the model to focus on more relevant nodes during information aggregation.
  + **Multi-head Attention:** GATs often employ multiple attention heads to capture different aspects of node interactions, enhancing the model's ability to learn complex relationships within the graph [52].

#### Layer Details:

The GAT used for ΔΔG prediction typically consists of multiple layers, each performing operations to aggregate and transform node representations:

* + **Input Layer:** Initial node features are encoded into a suitable representation (e.g., embeddings) that captures important characteristics of amino acids relevant to protein stability.
  + **Graph Convolutional Layers:** These layers perform convolutions over the graph structure, integrating information from neighboring nodes through the attention mechanism.
  + **Activation Functions:** Leaky ReLU [54] (Rectified Linear Unit) activation functions are employed after each layer to introduce non-linearity and improve the model's capacity to learn complex relationships [56].
  + **Normalization:** Techniques like batch normalization or layer normalization may be used to stabilize training and improve convergence.
  + **Output Layer:** The final layer outputs predictions for ΔΔG values, typically using a softmax function [55] to produce probabilities or a regression layer to directly output numerical values.

**3.5 Training process:**

**Dataset Preparation**:

The training dataset consists of pairs of wild-type and mutant proteins, with corresponding ΔΔG values calculated experimentally or derived from computational simulations. Each pair forms a graph where nodes correspond to amino acids, and edges represent interactions between them.

#### Loss Function and Optimization:

* + **Loss Function:** The choice of loss function is crucial for training the model to predict ΔΔG values accurately. Common choices include Mean Squared Error (MSE [59]) for regression tasks, which penalizes large prediction errors, and may include additional regularization terms to prevent overfitting.
  + **Optimizer:** Algorithms such as Adam or SGD (Stochastic Gradient Descent) are used to optimize model parameters iteratively. These algorithms adjust weights and biases based on the gradients of the loss function with respect to model parameters, aiming to minimize prediction errors.

#### Training Duration:

The model undergoes training over multiple epochs [58], where each epoch involves iterating through the entire dataset. The number of epochs [58] is chosen based on convergence criteria and validation performance, ensuring that the model learns sufficiently from the data.

#### Cross-Validation Strategy:

To assess the model's generalization ability, 5-fold cross-validation is employed. This involves splitting the dataset into five subsets (folds), training the model on four folds, and evaluating on the remaining fold. This process is repeated five times, with each fold serving as the validation set once. The average performance across all folds provides an unbiased estimate of the model's predictive accuracy.

#### Model Checkpointing:

During training, model checkpoints are saved periodically. These checkpoints capture the current state of the model, including weights, biases, and optimizer [57] state. Checkpoints facilitate resuming training from specific points and enable the selection of the best-performing model based on validation metrics.

**3.5.1 Evaluation Metrics**

**Performance Metrics:**

The model's performance is evaluated using several metrics to assess its accuracy in predicting ΔΔG values:

* + **Mean Squared Error (MSE):** Measures the average squared difference between predicted and actual ΔΔG values. Lower MSE [59] indicates better model performance.
  + **R2 Score:** Also known as the coefficient of determination, R2 [60] score measures how well the model predictions approximate the actual values. A score closer to 1 indicates a better fit.
  + **Pearson Correlation Coefficient (PCC):** Assesses the linear relationship between predicted and actual ΔΔG values. Higher correlation (closer to 1) indicates stronger predictive power.
  + **Accuracy within a 0.1 ΔΔG Threshold [62]:** Evaluates the proportion of predictions that fall within a specified range (±0.1 ΔΔG) of the actual values. This metric is critical for assessing the model's precision in predicting subtle changes in protein stability.

**3.6 Dataset Characteristics and Feature Representation Dataset Size and Diversity:**

The training dataset comprises a significant number of protein mutations, with comprehensive coverage across different proteins and mutation types [60]. This diversity ensures that the model learns from a broad spectrum of mutation effects on protein stability.

#### Feature Representation:

* + **Node Features:** Amino acids are represented by various features that capture their biochemical properties, such as size, charge, hydrophobicity, and secondary structure preferences. These features are crucial for accurately predicting the impact of mutations on protein stability.
  + **Edge Features:** Interactions between amino acids are represented by edge features, which could include distances, interaction energies, or structural constraints derived from experimental data or computational simulations.

#### Detailed Training Procedure

**Initialization:**

* + **Embedding Initialization:** Initial node embeddings are often initialized randomly or using pre-trained embeddings based on prior knowledge or transfer learning from related tasks.

#### Forward Propagation:

* + **Information Aggregation:** During each forward pass through the network, GAT aggregates information from neighboring nodes using the attention mechanism. Attention coefficients are computed based on learned weights, which determine the importance of each neighboring node's contribution to the current node's representation.
  + **Node Representation Update:** The aggregated information is then combined with the node's current representation through a series of transformations (typically linear transformations followed by activation functions like Leaky ReLU) [54].

#### Backpropagation:

* + **Gradient Calculation:** Gradients of the loss function with respect to model parameters are computed using backpropagation. These gradients guide the optimizer [57] in updating model weights to minimize prediction errors.
  + **Parameter Updates:** Optimizer [57] algorithms adjust model parameters based on computed gradients, applying learning rate schedules or momentum terms to optimize convergence speed and stability.

#### Convergence and Stopping Criteria:

* + **Early Stopping:** Training may incorporate early stopping mechanisms based on validation performance to prevent overfitting and ensure optimal generalization [61].

**3.7 Advantages of GAT for ΔΔG Prediction Handling Graph Structure:**

* + **Non-Uniform Interactions:** GATs excel in modeling non-uniform interactions between nodes, making them suitable for capturing complex dependencies in protein structures and interactions.
  + **Scalability:** GATs can handle large-scale graphs efficiently, leveraging parallel computation and attention mechanisms to process node information effectively.

#### Interpretability:

* + **Attention Mechanism:** The attention mechanism in GAT provides interpretability by highlighting which neighboring nodes and features are most relevant for predicting ΔΔG values. This enhances understanding of mutation effects on protein stability [62].

#### K-Fold Cross-Validation

5-fold cross-validation was conducted, ensuring that the dataset was divided into five subsets, with each subset used as a validation set once while the others were used for training. The average performance across all folds was reported [63].

# Results :

# A screenshot of a graph Description automatically generated

We trained this tool on a large and Diverse dataset containing This dataset contain information about the 862 proteins with 217731 mutations single point mutations change in gibb’s free energy values for Predicting ∆∆G upon a given mutation, We used 10 physiochemical features as node features of the graph where the nodes are representing different amino acids.in order to build a more reliable model and robust model, We performed five-fold cross validation with 200 epochs [58] for each folds selection of the training and test sets in 80% and 20% of total dataset, and the average PCC [61] is reported and the model is tested against the test dataset this way our model PCC [61], MSE [59] and also calculated the training loss [63] and R2[60] score. The performance of the Graph-Based Method for predicting the stability of proteins or changes in Gibbs free energy after inducing a single point mutation was evaluated using a 10-fold cross-validation approach. The evaluation metrics included training and validation loss, training and validation accuracy, validation R² score, and validation Pearson correlation coefficient. These metrics were plotted over the epochs for each fold, as shown in Figure 1. This section provides an in-depth analysis of the results, highlighting the model's convergence behavior, accuracy, and correlation with actual values.

**Training and Validation Loss**

The training and validation loss curves (top left and top right, respectively, in Figure 1) illustrate the model's learning process over the epochs for each of the 10 folds. Initially, both the training and validation losses are relatively high. However, within the first few epochs, there is a rapid decrease in loss values, indicating that the model is quickly learning to minimize the error. This sharp decline suggests effective optimization and a good fit to the training data.

* **Training Loss**: The training loss reaches a near-zero value early in the training process. This consistent behavior across all folds signifies that the model effectively minimizes the error on the training data, achieving an optimal solution.
* **Validation Loss**: The validation loss also exhibits a steep decline, stabilizing at a low value. This low validation loss indicates that the model generalizes well to unseen data, maintaining low error rates on the validation sets.

The rapid convergence and stabilization of loss values imply that the model's architecture and training algorithm are well-suited for the given task. The minimal difference between training and validation losses further suggests that the model does not suffer from significant overfitting.

The training accuracy (middle left in Figure 1) and validation accuracy (middle right in Figure 1) provide insights into the model's ability to correctly predict the stability of proteins or changes in Gibbs free energy after mutations.

**Training Accuracy**: The training accuracy reaches close to 1.0 (or 100%) within the first few epochs. This rapid achievement of high accuracy demonstrates the model's capacity to learn the underlying patterns in the training data effectively.

**Validation Accuracy**: Similarly, the validation accuracy stabilizes close to 1.0, indicating that the model maintains high performance on the validation data. This consistency across all folds suggests that the model is robust and reliable in its predictions. The high accuracy scores for both training and validation sets indicate that the model has successfully learned to distinguish between different stability changes induced by single point mutations. The minimal variance in accuracy across different folds highlights the model's robustness and its ability to generalize well to new, unseen data. The R² score (bottom left in Figure 1) is a statistical measure that represents the proportion of the variance for a dependent variable that's explained by an independent variable or variables in a regression model. In this context, it measures how well the predicted stability changes correlate with the actual values. The validation R² score remains high, consistently close to 1.0 across all folds. A high R² score indicates that the model can explain most of the variability in the validation data, suggesting a strong predictive performance. The stability of the R² scores across different folds underscores the reliability of the model's predictions. The consistent high R² values demonstrate that the model accurately captures the relationship between input features and the resulting stability changes, making it a valuable tool for predicting protein stability. The Pearson correlation coefficient (bottom right in Figure 1) measures the linear correlation between the predicted and actual values. It ranges from -1 to 1, with 1 indicating a perfect positive linear relationship. **Validation Pearson Correlation Coefficient**: The Pearson correlation coefficient remains high, fluctuating around 0.98 to 1.0 for all folds. This high correlation coefficient suggests that the model predictions are highly correlated with the actual values in the validation set. The strong positive correlation indicates that the model's predictions are closely aligned with the true stability changes, further validating the model's accuracy and effectiveness. The consistency of the correlation coefficient across different folds demonstrates the robustness of the model. To delve deeper into the performance metrics, let's consider the behavior and implications of each metric in more detail The rapid decrease in training and validation loss is indicative of a well-optimized model. The low final values of both metrics suggest that the model has effectively minimized error and is not prone to overfitting. The convergence pattern also reflects the efficiency of the training process, ensuring that the model reaches an optimal state quickly. The near-perfect accuracy scores for both training and validation sets highlight the model's precision in predicting stability changes. The high accuracy indicates that the model can correctly classify the outcomes for most of the data points, reducing the likelihood of misclassifications. The R² score is particularly important in regression tasks as it quantifies the model's explanatory power. The high R² values across all folds suggest that the model captures the underlying variance in the data effectively, providing reliable and consistent predictions. The Pearson correlation coefficient's high values indicate that the predictions are not only accurate but also linearly related to the actual values. This linear relationship is crucial for ensuring that the model's predictions follow the expected trends in stability changes. The robust performance of the Graph-Based Method, as demonstrated by the metrics and visualized in Figure 1, has significant implications for protein stability prediction. The high accuracy and correlation metrics suggest that this method can be reliably used in practical applications where accurate prediction of protein stability changes is critical. This could include drug design, protein engineering, and other biotechnological applications. The rapid convergence of the model indicates that it can be trained efficiently, making it feasible for large-scale applications where computational resources and time are constrained. The consistency across different folds and low variance in performance metrics imply that the model generalizes well to new data. This robustness is essential for ensuring that the model performs reliably in diverse and unforeseen scenarios. The insights gained from this model can aid in understanding the factors influencing protein stability. By analyzing the model's predictions and the corresponding features, researchers can identify key determinants of stability changes, guiding further experimental investigations.

**5.Conclusion**

The study "Graph-Based Prediction of Protein Stability Changes Induced by Single Point Mutations" has provided a comprehensive analysis and methodological advancement in the prediction of protein stability changes due to single-point mutations. This work bridges a significant gap between sequence-based and structure-based prediction methods, presenting a novel graph-based approach that leverages the strengths of both. The proposed graph-based model integrates both structural and sequence data to predict changes in Gibbs free energy (ΔΔG) resulting from single-point mutations. By constructing graphs for wild-type and mutant proteins using nine physicochemical properties as node features, the model achieves higher prediction accuracy even with limited structural data availability. The model was validated using k-fold cross-validation, demonstrating robust performance with mean squared error (MSE) and R² scores comparable to or exceeding existing prediction tools. This indicates that the graph-based model provides a reliable and accurate method for predicting protein stability changes. The enhanced accuracy of this model can significantly impact various fields, including genetic research, drug design, and understanding disease mechanisms related to protein stability. By providing more accurate predictions, the model aids in identifying potentially deleterious mutations that could affect protein function, thereby facilitating better experimental planning and resource allocation in research.One of the key strengths of this study is the integration of structural and sequence data into a unified model. This approach overcomes the limitations of traditional methods that rely solely on either sequence-based or structure-based data, offering a more holistic view of protein stability changes. The construction of protein graphs using physicochemical properties as node features allows the model to capture intricate details of protein structure and function. This novel representation enhances the model's ability to predict the impact of mutations more accurately than traditional linear models. The application of graph attention neural networks (GANN) further improves the model's predictive power.GANNs can effectively learn from complex graph structures, making them well-suited for this type of prediction task. The model's performance metrics indicate that it successfully leverages these advanced techniques to provide reliable predictions.The development of this graph-based model represents a significant advancement in computational biology, offering researchers a powerful tool for predicting protein stability changes. This can lead to more accurate hypotheses and experimental designs, ultimately accelerating the pace of discovery in fields such as molecular biology and bioinformatics. While the current study focuses on single-point mutations, the methodology could be extended to other types of mutations and variations, such as insertions, deletions, and complex variants. This would further broaden the model's applicability and usefulness in genetic research and clinical diagnostics. Future work should focus on expanding the dataset used for model training and validation to include a wider variety of proteins and mutations. Additionally, continuous refinement of the model's architecture and feature selection process will be essential to maintaining and improving its predictive accuracy over time. To further validate and enhance the model's predictions, collaboration with experimental researchers is crucial. By comparing the model's predictions with experimental results, researchers can identify areas for improvement and refine the model to better align with empirical data.

# References:

1. Kellogg, E. H., & Leaver-Fay, A. (2016). Molecular Dynamics: A Tool for Structure- Function Analysis of Biomolecules. *Nature Methods, 13*(10), 828–839. DOI: 10.1038/nmeth.3969
2. Yang, Y., & Zhou, Y. (2008). Specific Interactions for Ab Initio Folding of Protein Terminal Fragments with Secondary Structures. *Proteins: Structure, Function, and Bioinformatics, 72*(2), 793–803. DOI: 10.1002/prot.21948
3. Parthiban, V., & Gromiha, M. M. (2007). Analysis and Prediction of Exposed Loop Regions in Proteins. *Proteins: Structure, Function, and Bioinformatics, 71*(1), 421–431. DOI: 10.1002/prot.2167
4. Bendl, J., Stourac, J., Sebestova, E., Vavra, O., Musil, M., Brezovsky, J., & Damborsky, J. (2016). HotSpot Wizard 2.0: automated design of site-specific mutations and smart libraries in protein engineering. *Nucleic acids research*, *44*(W1), W479-W487.
5. Traditionally, predicting ΔΔG involves computationally intensive methods such as molecular dynamics simulations, which, despite their accuracy, can be prohibitively slow and resource-demanding for large-scale studies. This has driven the search for more efficient computational models that can leverage existing structural data and sophisticated machine learning techniques to provide accurate and scalable predictions.
6. Lin, Z., Zou, J., Liu, S., Peng, C., Li, Z., Wan, X., ... & Yang, M. (2021). A cloud computing platform for scalable relative and absolute binding free energy predictions: New opportunities and challenges for drug discovery. *Journal of Chemical Information and Modeling*, *61*(6), 2720-2732.
7. Herschlag, D., & Pinney, M. M. (2018). Hydrogen bonds: simple after all?. *Biochemistry*, *57*(24), 3338-3352.
8. Eisenberg, B. (2013). Ionic interactions are everywhere. *Physiology*, *28*(1), 28-38.
9. Andersson, Y., Hult, E., Rydberg, H., Apell, P., Lundqvist, B. I., & Langreth, D. C. (1998). Van der Waals interactions in density functional theory. *Electronic Density Functional Theory: Recent Progress and New Directions*, 243-260.
10. Zegura, E. W., Calvert, K. L., & Donahoo, M. J. (1997). A quantitative comparison of graph-based models for Internet topology. *IEEE/ACM Transactions on networking*, *5*(6), 770-783.
11. Chanphai, P., Bekale, L., & Tajmir-Riahi, H. A. (2015). Effect of hydrophobicity on protein–protein interactions. *European Polymer Journal*, *67*, 224-231.
12. Lund, M., & Jönsson, B. (2005). On the charge regulation of proteins. *Biochemistry*, *44*(15), 5722-5727.
13. Franks, F. (2002). Protein stability: the value of ‘old literature’. *Biophysical chemistry*, *96*(2-3), 117-127.
14. Chalker, J. M., Bernardes, G. J., Lin, Y. A., & Davis, B. G. (2009). Chemical modification of proteins at cysteine: opportunities in chemistry and biology. *Chemistry–An Asian Journal*, *4*(5), 630-640
15. Rose, P. W., Prlić, A., Altunkaya, A., Bi, C., Bradley, A. R., Christie, C. H., ... & Burley, S.

K. (2016). The RCSB protein data bank: integrative view of protein, gene and 3D structural information. *Nucleic acids research*, gkw1000.

1. . Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., & Bengio, Y. (2017). Graph attention networks. *arXiv preprint arXiv:1710.10903*.
2. Capriotti, E., Fariselli, P., Rossi, I., & Casadio, R. (2008). A three-state prediction of single point mutations on protein stability changes. *BMC bioinformatics*, *9*, 1-9.
3. Cock, P. J., Antao, T., Chang, J. T., Chapman, B. A., Cox, C. J., Dalke, A., ... & De Hoon,

M. J. (2009). Biopython: freely available Python tools for computational molecular biology and bioinformatics. *Bioinformatics*, *25*(11), 1422.

1. Abraham, M. H., & Acree Jr, W. E. (2019). Solvation descriptors for zwitterionic α- Aminoacids; estimation of water–solvent partition coefficients, solubilities, and hydrogen- bond acidity and hydrogen-bond basicity. *ACS omega*, *4*(2), 2883-2892.
2. Adzhubei, I. A., Schmidt, S., Peshkin, L., Ramensky, V. E., Gerasimova, A., Bork, P., ... & Sunyaev, S. R. (2010). A method and server for predicting damaging missense mutations. *Nature methods*, *7*(4), 248-249.
3. Ng, P. C., & Henikoff, S. (2003). SIFT: Predicting amino acid changes that affect protein function. *Nucleic acids research*, *31*(13), 3812-3814.
4. Choi, Y., Sims, G. E., Murphy, S., Miller, J. R., & Chan, A. P. (2012). Predicting the functional effect of amino acid substitutions and indels.
5. Dill, K. A. (1990). Dominant forces in protein folding. *Biochemistry*, *29*(31), 7133-7155.
6. Brown, S., Albrechtsen, B., Pedersen, S., & Klemm, P. (1982). Localization and regulation of the structural gene for transcription-termination factor rho of Escherichia coli. *Journal of molecular biology*, *162*(2), 283-298.
7. Dobson, C. M., Šali, A., & Karplus, M. (1998). Protein folding: a perspective from theory and experiment. *Angewandte Chemie International Edition*, *37*(7), 868-893.
8. 攀苏. (1963). Stereochemistry of polypeptide chain configurations.“. *J. mol. Biol*, *7*, 95- 99.
9. Honig, B., & Nicholls, A. (1995). Classical electrostatics in biology and chemistry. *Science*, *268*(5214), 1144-1149.
10. Hannenhalli, S. S., & Russell, R. B. (2000). Analysis and prediction of functional sub-types from protein sequence alignments. *Journal of molecular biology*, *303*(1), 61-76.
11. Jones, D. T., & Thornton, J. M. (1996). Potential energy functions for threading. *Current Opinion in Structural Biology*, *6*(2), 210-216.
12. Bailey, T. L., & Elkan, C. (1994). Fitting a mixture model by expectation maximization to discover motifs in bipolymers.
13. Andrade, M. A., Perez-Iratxeta, C., & Ponting, C. P. (2001). Protein repeats: structures, functions, and evolution. *Journal of structural biology*, *134*(2-3), 117-131.
14. Ng, P. C., & Henikoff, S. (2001). Predicting deleterious amino acid substitutions. *Genome research*, *11*(5), 863-874.
15. PROVEAN web server: a tool to predict the functional effect of amino acid substitutions and indels.
16. Yates, C. M., Filippis, I., Kelley, L. A., & Sternberg, M. J. (2014). SuSPect: enhanced prediction of single amino acid variant (SAV) phenotype using network features. *Journal of molecular biology*, *426*(14), 2692-2701.
17. Ng, P. C., & Henikoff, S. (2001). Predicting deleterious amino acid substitutions. *Genome research*, *11*(5), 863-874.
18. Adzhubei, I. A., Schmidt, S., Peshkin, L., Ramensky, V. E., Gerasimova, A., Bork, P., ... & Sunyaev, S. R. (2010). A method and server for predicting damaging missense mutations. *Nature methods*, *7*(4), 248-249.
19. Yue, P., Melamud, E., & Moult, J. (2006). SNPs3D: candidate gene and SNP selection for association studies. *BMC bioinformatics*, *7*, 1-15.
20. Yue, P., Melamud, E., & Moult, J. (2006). SNPs3D: candidate gene and SNP selection for association studies. *BMC bioinformatics*, *7*, 1-15.
21. 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.

1. Kabsch, W., & Sander, C. (1983). Dictionary of protein secondary structure: pattern recognition of hydrogen‐bonded and geometrical features. *Biopolymers: Original Research on Biomolecules*, *22*(12), 2577-2637.
2. Creighton, T. E. (1993). *Proteins: structures and molecular properties*. Macmillan.
3. Sunyaev, S., Ramensky, V., Koch, I., Lathe III, W., Kondrashov, A. S., & Bork, P. (2001). Prediction of deleterious human alleles. *Human molecular genetics*, *10*(6), 591-597.
4. Schwarz, J. M., Cooper, D. N., Schuelke, M., & Seelow, D. (2014). MutationTaster2: mutation prediction for the deep-sequencing age. *Nature methods*, *11*(4), 361-362.
5. Eddy, S. R. (2011). Accelerated profile HMM searches. *PLoS computational biology*, *7*(10), e1002195.
6. Shihab, H. A., Rogers, M. F., Gough, J., Mort, M., Cooper, D. N., Day, I. N., ... & Campbell,

C. (2015). An integrative approach to predicting the functional effects of non-coding and coding sequence variation. *Bioinformatics*, *31*(10), 1536-1543.

1. 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*, 1-9.
2. UniProt Consortium. (2015). UniProt: a hub for protein information. *Nucleic acids research*, *43*(D1), D204-D212.
3. VariBench: A benchmark database for variations. *Human Mutation*,
4. Zeeshan, S., & Ahmed, Z. (2019, November). PAS-MUT with Somatic Mutations in Cancer and Classified Diseases for Clinical-Genomics Research: PAS-MUT & Somatic Mutations. In *2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)* (pp. 1708-1715). IEEE.
5. SIFT: Predicting amino acid changes that affect protein function. *Nucleic Acids Research*
6. Calabrese, R., Capriotti, E., Fariselli, P., Martelli, P. L., & Casadio, R. (2009). Functional annotations improve the predictive score of human disease‐related mutations in proteins. *Human mutation*, *30*(8), 1237-1244.
7. Li, B., Krishnan, V. G., Mort, M. E., Xin, F., Kamati, K. K., Cooper, D. N., ... & Radivojac,

P. (2009). Automated inference of molecular mechanisms of disease from amino acid substitutions. *Bioinformatics*, *25*(21), 2744-2750.

1. Land, H., & Humble, M. S. (2018). YASARA: a tool to obtain structural guidance in biocatalytic investigations. *Protein engineering: methods and protocols*, 43-67.
2. Maas, A. L., Hannun, A. Y., & Ng, A. Y. (2013, June). Rectifier nonlinearities improve neural network acoustic models. In *Proc. icml* (Vol. 30, No. 1, p. 3).
3. Christopher, M., & Nasrabadi, N. M. (2006). Pattern Recognition and Machine Learning. Springer. *vol. Bishop*, 183.
4. Kipf, T. N., & Welling, M. (2016). Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.
5. Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
6. Wang, H., Lei, Z., Zhang, X., Zhou, B., & Peng, J. (2016). Machine learning basics. *Deep learning*, 98-164.
7. Chai, T., & Draxler, R. R. (2014). Root mean square error (RMSE) or mean absolute error (MAE)?–Arguments against avoiding RMSE in the literature. *Geoscientific model development*, *7*(3), 1247-1250.
8. Frost, J. (2013). Regression analysis: How do I interpret R-squared and assess the goodness-of-fit. *The Minitab Blog*, *30*.
9. Cohen, I., Huang, Y., Chen, J., Benesty, J., Benesty, J., Chen, J., ... & Cohen, I. (2009). Pearson correlation coefficient. *Noise reduction in speech processing*, 1-4.
10. Tharwat, A. (2020). Classification assessment methods. *Applied computing and informatics*, *17*(1), 168-192.
11. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, *15*(1), 1929-1958.

Plagiarism report

A screenshot of a cell phone

Description automatically generated

Certificate

