INTEGRATION OF PROTEIN THREE-DIMENSIONAL STRUCTURE INTO THE WORKFLOW OF INTERPRETATION OF GENETIC VARIANTS

Dissertation
zur Erlangung des Grades
des Doktors der Naturwissenschaften
der Fakultät für Mathematik und Informatik
der Universität des Saarlandes

vorgelegt von
Alexander Gress

Saarbrücken
2020

Kolloqium gehalten am 24. Juli 2020

Dekan der Fakultät: Prof. Dr. Thomas Schuster

Kommitee:
Vorsitzende: Prof. Dr. Isabel Valera
Gutachterin: Prof. Dr. Olga Kalinina
Gutachter: Prof. Dr. Volkhard Helms
Beisitzerin: Dr. Christina Backes

ABSTRACT

Life stores information in large biopolymer molecules, which can be represented as a sequence of letters. Computers stores information in sequences of zeros and ones. This predestines computers for automated processing of biological data and with a great success. Computational biology has produced many methods and tools based on biological sequences. However, reducing life to just sequences radically reduces the whole picture. The functionality of biomolecules, especially proteins, is performed in the three-dimensional (3D) space. Thus, limiting methods in computational biology to sequences will never yield sufficient insights in the ways molecular biology operates.

In this thesis I present my work on the integration of protein 3D structure information into the methodological workflow of computational biology. We developed an algorithmic pipeline that is able to map protein sequences to protein structures, providing an additional source of information.

We used this pipeline in order to analyze the effects of genetic variants from the perspective of protein 3D structures. We analyzed genetic variants associated with diseases and compared their structural arrangements to that of neutral variants. Additionally, we discussed how structural information can improve methods that aim to predict the consequences of genetic variants.

ZUSAMMENFASSUNG

Das Leben speichert Informationen mit der Hilfe von langen Biopolymermolekülketten. Man kann solche Ketten durch Buchstabensequenzen beschreiben. Computer speichern Informationen in Sequenzen von Nullen und Einsen. Dies prädestiniert Computer zur Verabeitung biologischer Daten und tatsächlich hat die Bioinformatik, mit großem Erfolg, Methoden und Werkzeuge entwickelt, die auf der Verarbeitung solcher Sequenzen basieren.
Allerdings, spielt sich die Funktionalität von Biomolekülen, insbesonders die von Proteinen, im drei-dimensionalen (3D) Raum ab. Und deshalb werden bioinformatische Methoden, die sich auf Sequenzdaten beschränken niemals in der Lage sein, mikrobiologische Vorgänge funktionell zu beschreiben.
Diese Thesis widmet sich der Integration von Protein 3D Strukturinformationen in die Abläufe bioinformatischer Methodiken. Wir haben eine algorithmische Pipeline entwickelt, die es ermöglicht Proteinsequenzen auf Proteinstrukturen abzubilden um so eine zusätzliche Informationsquelle beizusteuern.
Wir benutzten diese Methodik um die Effekte von genetischen Variationen aus der Sichtweise von Proteinstrukturen zu analysieren. Wir haben die Tendenzen der räumlichen Verteilung von genetischen Varianten, die man mit Krankheiten in Verbidung gebracht hat, analysiert und sie mit denen von neutralen Varianten verglichen. Desweiteren, haben wir geprüft in wie weit das Einbeziehen strukureller Daten die Vorhersage von Konsequenzen genetischer Varianten verbessert.

ACKNOWLEDGMENTS

In the first place, I would like to thank Prof. Dr. Olga Kalinina. It was her influence as a scientific mentor that quickened my interest in science and therefore now I plan to stay in academia longer as I had thought five years ago.

I also want to thank Prof. Dr. Volkhard Helms for taking the time to review my thesis.

I want to thank Prof. Dr. Andreas Keller for his support in the development of StructMAn, which really pushed the first part of my doctoral studies.

I want to thank Prof. Dr. Dr. Thomas Lengauer for giving me the opportunity to do my doctoral studies in the first place. The time in his group at the Max Planck Institute was amazing due to the huge amount of amazing people working there, whom I also want to thank for the awesome time.

Some of those awesome people migrated together with me to start Olga's new group at the Helmholtz Institute, and those also contributed a lot to my scientific work, thank you for all of that Sebastian and Sanjay. Later, we grew and Fawaz joined our young group, also thanks to you.

I want to thank former colleagues, who shared a lot of the fun along the way: Alex, Reini, Christoph, Kerstin, Jan, and Thorsten. Some of those were part of the 'mensa gang', which to this date helps me to survive the daily grind, so thank you, Michael, Nick, and Andreas.

Thank you Thorsten, Sebastian, Sanjay, and Fawaz for proofreading parts of the thesis. And again thank you, Thorsten, for providing me with the latex template for this thesis.

Needless to say I want to thank my family for their undaunted support throughout my studies.

Ganz zum Schluss, möchte ich meiner Verlobten danken. Kathrin hatte es sicher oft nicht leicht, wenn Ich meiner Doktorarbeit mehr Zeit einräumte als Ihr. Es tut mir leid, wenn du in dieser Zeit zu wenig Aufmerksamkeit von mir bekommen hast, denoch ändert es nichts daran:
Ich liebe Dich.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

Since the introduction of the central dogma of molecular biology in 1970 [1] it is universally accepted in the scientific community that the mechanisms of life are based on the flow of genetic information from the genes over transcripts to the proteins. Nowadays, each step in that succession spans a whole research area named after the corresponding biomolecule with the suffix -omics, e.g. genomics, transcriptomics or proteomics. Later on, new discoveries expanded this classical central dogma and lead to the creation of more specialized fields, like epigenomics, interactomics, metabolomics, metagenomics and many more leading to the omics-age of molecular biology. Each omics field on its own developed experimental methods creating massive amounts of data, which are impossible to be processed in a manual fashion. This need for automatization introduced computational biology to all omics fields. Computational biology also plays an important role when it comes to the connection between individual fields. The tasks range from the transformation of different types of data, over mapping dataset from one type of biomolecules to another type of biomolecules, to the simulation and prediction of whole biological systems from a dataset that is upstream on the flow of biological information. The accomplishment of the last task is particularly desirable. Since in the central dogma the flows of information between all biomolecules are connected, it should be possible to calculate, simulate or predict the states of one type of biomolecule by the data of another connected field, preferably a field for which the data can be obtained more easily. One especially important example of such a task is the prediction of the phenotype from the genome and/or epigenome. Performing that feat perfectly would require a total understanding of molecular biology, which sounds very utopic given our current understanding. The solution to another similar problem seems feasible: given a defined reference system, predict how the system changes upon perturbation. In particular, in our setting, this means the prediction of the effects of genetic variation on the phenotype.

The research presented in this thesis is related to several omics fields including genomics, transcriptomics, proteomics, and interactomics. From the genome, one can deduce the nucleic acid composition of the corresponding expressed gene products. For us, the most important products are the messenger RNAs (mRNAs) that are translated into proteins. Through this process, variations in the genome called mutations are also transferred to the mRNAs as well as to their corresponding proteins where sometimes they cause amino acid variations. When they reach this stage, mutations can influence the function of proteins and the interactions proteins participate in. The transition from variants in the mRNA to their effects on proteins is the key research subject of this thesis.

In the majority of cases, the input information is derived from genome-level data produced by next-generation sequencing (NGS) techniques. To understand the effects of genetic variations on the proteome and interactome, they have

to be mapped to the level of the proteome. Individual sequence positions corresponding to mutations have to be related to individual amino acids in the context of three-dimensional protein structures, and their role in the interactome has to be inferred by the analysis of their participation in biochemical interactions.

We developed an algorithmic solution that maps protein sequences, which include genetic variants into the spatial world of protein three-dimensional (3D) structures, performing structural analyses of the mapped structures and producing features, which finally are used to predict the functional impact of the corresponding genetic variant.

In general, we call any form of assignment of protein sequences to protein structures a *structural annotation*. Such an annotation can be performed on a sequence level or on a residue-wise level, where individual positions in the sequence, are mapped to individual residues in a protein 3D structure. For the analysis of mutations in protein structures, a residue-wise structure annotation is unavoidable.

A simple form of structural annotation is only considering experimentally resolved structures where the amino acid sequence is identical to the target sequence. We call such a structure a corresponding structure. This task is basically solved, since the major protein sequence repositories, for example, Uniprot [2], include this kind of information. The residue-level annotation of a sequence to a corresponding structure is a very simple task since the sequences are identical. More interesting is the annotation of structures with weaker restrictions. Here the goal is to assign appropriate structures of proteins, which are not identical to the target protein. What constitutes an appropriate structure and why certain techniques can reveal interesting functional insights for the protein are key topics discussed in this thesis. One famous example of a structural annotation is the template structure search as a preparation step for homology-based protein structure modeling.

Structural annotation methods often are combined with some form of structural analysis. There are many forms of structural analyses, which are more or less complex. Simpler, but also faster analyses calculate and assign structural properties. For example, DSSP [3] calculates the solvent-accessible surface area, secondary structure elements, and other geometrical properties. Also quite simple, but computationally more expensive is the calculation of a pairwise distance matrix of all residues of a protein structure. A similar approach is the calculation of the surface area of the pairwise interaction interface between all residues, an example method doing that would be Probe [4]. Other structural analysis methods estimate the stability of a protein structure by measuring the free energy of the protein folding. Some of these methods also can estimate the difference in folding energy upon mutation, for example, Cupsat [5], MCSM [6] and FoldX [7]. Even more computationally expensive methods that fall into the category of structural analysis are docking methods. Such methods estimate the best spatial composition or docking poses of two interaction partners, examples are FlexX [8] for protein-ligand docking and HADDOCK [9] for protein-protein docking. The arguably most computationally expensive structural analyses are molecular dynamics simulations. In such simulations, the ambitious task is to

compute as precise as possible all individual forces on each atom, followed by the movement of the atoms according to the calculated forces in a minuscule time step and repeating the process multiple times. An example method for conducting molecular dynamics simulations is GROMACS [10].
The structural annotation method that we developed in the frame of this thesis is named Structural Mutation ANnotation (StructMAn). While developing this tool, we aimed to achieve several goals that would ensure the optimal usability of the tool. The tool should be not constrained to a given input: proteins and mutations in any number from any species should be processable. Further, an important aspect of StructMAn is the idea to include as much structural information as possible with the goal to produce better structural analyses and to increase the number of the cases, for which the annotation is applicable. For this, we used not only the corresponding experimentally resolved protein structures, but also the experimentally resolved protein structures of homologs. Through the realization of these goals, the method left the realm of structural mutation annotation and is now able to structurally annotate any form of amino acid sequences in a position-specific manner.
The estimation of the differences between a reference and a slightly modified system introduced by a genetic alteration has many names: variant impact prediction [6, 11, 12], variant effect prediction [13, 14] or variant prioritization [15, 16] are the most common names. Since StructMAn is able to produce a lot of structural features, also for variants that could not be structurally annotated before, we developed our own variant effect prediction method. It combines established evolutionary features with a complex array of structural features from StructMAn in a random forest classifier approach.
The overall ambition of the work described in this thesis is structural annotation and prediction of the impact of genetic variants on the largest possible scale, which would enable an unprecedented level of connection between genome and proteome. This ambitious goal comes with an array of challenges. The amount of data that has to be processed renders the general usage of computationally expensive methods like protein structure prediction, docking experiments, and molecular dynamics simulations impossible. Efficient solutions are required that balance the amount and accuracy of generated information with time constraints and computational resources.
This thesis contains four projects, each more specifically explained in their own chapters. The first project (Chapter 4) describes the development of a specific structural analysis method for estimating the relative solvent accessible area of individual residues. The second project (Chapter 5) is about the automatization of the genome to proteome mapping and the structural analysis process and presents the development of StructMAn. The third project (Chapter 6) demonstrates the application of the methodology developed in the second project in a comprehensive study focusing on variants associated with genetic diseases. The fourth project (Chapter 7) also builds on the methods from the second project and uses the results produced by the structural analyses as features in a machine learning tool for predicting functional and pathogenic consequences of mutations.

1.1 FIRST AUTHOR PUBLICATIONS RELATED TO DOCTORAL STUDIES

Gress, A., Ramensky, V., Büch, J., Keller, A. and Kalinina, O.V., **"StructMAn: annotation of single-nucleotide polymorphisms in the structural context"**, *Nucleic Acids Research*, vol. 44, Jul. 2016.

Abstract: The next generation sequencing technologies produce unprecedented amounts of data on the genetic sequence of individual organisms. These sequences carry a substantial amount of variation that may or may be not related to a phenotype. Phenotypically important part of this variation often comes in form of protein-sequence altering (non-synonymous) single nucleotide variants (nsSNVs). Here we present StructMAn, a Web-based tool for annotation of human and non-human nsSNVs in the structural context. StructMAn analyzes the spatial location of the amino acid residue corresponding to nsSNVs in the three-dimensional protein structure relative to other proteins, nucleic acids and low molecular-weight ligands. We make use of all experimentally available three-dimensional structures of query proteins, and also, unlike other tools in the field, of structures of proteins with detectable sequence identity to them. This allows us to provide a structural context for around 20% of all nsSNVs in a typical human sequencing sample, for up to 60% of nsSNVs in genes related to human diseases, and for around 35% of nsSNVs in a typical bacterial sample. Each nsSNV can be visualized and inspected by the user in the corresponding three-dimensional structure of a protein or protein complex. The StructMAn server is available at `http://structman.mpi-inf.mpg.de`.

Gress, A., Ramensky, V., and Kalinina, O.V., **"Spatial distribution of disease-associated variants in three-dimensional structures of protein complexes"**, *Oncogenesis*, vol. 6, Sep. 2017.

Abstract: Next-generation sequencing enables simultaneous analysis of hundreds of human genomes associated with a particular phenotype, for example, a disease. These genomes naturally contain a lot of sequence variation that ranges from single-nucleotide variants (SNVs) to large-scale structural rearrangements. In order to establish a functional connection between genotype and disease-associated phenotypes, one needs to distinguish disease drivers from neutral passenger variants. Functional annotation based on experimental assays is feasible only for a limited number of candidate mutations. Thus alternative computational tools are needed. A possible approach to annotating mutations functionally is to consider their spatial location relative to functionally relevant sites in three-dimensional (3D) structures of the harboring proteins. This is impeded by the lack of available protein 3D structures. Complementing experimentally resolved structures with reliable computational models is an attractive alternative. We developed a structure-based approach to characterizing comprehensive sets of non-synonymous single-nucleotide variants (nsSNVs): associated with cancer, non-cancer diseases and putatively functionally neutral. We searched experimentally resolved protein 3D structures for potential homology-modeling templates for proteins harboring corresponding mutations.

We found such templates for all proteins with disease-associated nsSNVs, and 51 and 66% of proteins carrying common polymorphisms and annotated benign variants. Many mutations caused by nsSNVs can be found in protein-protein, protein-nucleic acid or protein-ligand complexes. Correction for the number of available templates per protein reveals that protein-protein interaction interfaces are not enriched in either cancer nsSNVs, or nsSNVs associated with non-cancer diseases. Whereas cancer-associated mutations are enriched in DNA-binding proteins, they are rarely located directly in DNA-interacting interfaces. In contrast, mutations associated with non-cancer diseases are in general rare in DNA-binding proteins, but enriched in DNA-interacting interfaces in these proteins. All disease-associated nsSNVs are overrepresented in ligand-binding pockets, and nsSNVs associated with non-cancer diseases are additionally enriched in protein core, where they probably affect overall protein stability.

Gress, A. and Kalinina, O.V., **"SphereCon - A method for precise estimation of residue relative solvent accessible area from limited structural information."**, *Bioinformatics*, **Under review**
Abstract: Motivation: In proteins, solvent accessibility of individual residues is a factor contributing to their importance for protein function and stability. Hence one might wish to calculate solvent accessibility in order to predict the impact of mutations, their pathogenicity, and for other biomedical applications. A direct computation of solvent accessibility is only possible if all atoms of a protein three-dimensional structure are reliably resolved. Results: We present SphereCon, a new precise measure that can estimate residue relative solvent accessibility (RSA) from limited data. The measure is based on calculating the volume of intersection of a sphere with a cone cut out in the direction opposite of the residue with surrounding atoms. We propose a method for estimating the position and volume of residue atoms in cases when they are not known from the structure, or when the structural data are unreliable or missing. We show that in cases of reliable input structures, SphereCon correlates almost perfectly with the directly computed RSA, and outperforms other previously suggested indirect methods. Moreover, SphereCon is the only measure that yield accurate results when the identities of amino acids are unknown. A significant novel feature of SphereCon is that it can estimate RSA from inter-residue distance and contact matrices, without any information about the actual atom coordinates. Availability: https://github.com/kalininalab/spherecon

1.2 COAUTHOR PUBLICATIONS DURING DOCTORAL STUDIES

Mueller, S.C., Backes C., Gress, A., Baumgarten N., Kalinina O.V., Moll A., Kohlbacher O., Meese E., Keller A., **"BALL-SNPgp-from genetic variants toward computational diagnostics."**, *Bioinformatics*, vol. 32, no. 12, Jun. 2016.
Abstract: In medical research, it is crucial to understand the functional consequences of genetic alterations, for example, non-synonymous single nucleotide variants (nsSNVs). NsSNVs are known to be causative for several human dis-

eases. However, the genetic basis of complex disorders such as diabetes or cancer comprises multiple factors. Methods to analyze putative synergetic effects of multiple such factors, however, are limited. Here, we concentrate on nsSNVs and present BALL-SNPgp, a tool for structural and functional characterization of nsSNVs, which is aimed to improve pathogenicity assessment in computational diagnostics. Based on annotated SNV data, BALL-SNPgp creates a three-dimensional visualization of the encoded protein, collects available information from different resources concerning disease relevance and other functional annotations, performs cluster analysis, predicts putative binding pockets and provides data on known interaction sites.
My contribution: I implemented the structural annotation performed in the algorithmic pipeline of BALL-SNPgp.

2 BIOLOGICAL BACKGROUND

2.1 PROTEIN BIOSYNTHESIS

The central dogma of molecular biology [1] states the direction of transfer and the usage of information stored in the genome and in the biological processes leading to the biosynthesis of proteins (Figure 2.1).
The composition and 3D structure of biological molecules, most importantly proteins, as well as their interplay ultimately result in a well-behaving biological system called life. While this system as a whole has multiple levels of complexity, there is a common denominator in the form of protein biosynthesis [17]. The process involves two major steps, transcription and translation. It starts with the genome, which is a large molecule consisting of two sequential concatenations of nucleotides forming DNA strands. Nucleotides are the monomeric units of all nucleic acids, and consist of a sugar, a phosphate and a nucleobase. There are four nucleobases: Adenine (A), Thymine (T), Guanine (G) and Cytosine (C) (Figure 2.2). Using hydrogen bonds (H-bonds) they can form so-called base pairs with their complementary base, A is complementary to T and G is complementary to C. This way, two DNA strands with complementary sequences form together double-stranded DNA, which forms the famous double-helix structure [18]. The genetic information is stored in all living organisms (expect some classes of viruses, although it is still debated whether viruses can be regarded as living organisms) in double-stranded DNA.
In the transcription process, short segments of the genome that correspond to individual genes are inversely copied, by concatenating complementary nucleotides forming a messenger RNA molecule (mRNA). The nucleotides forming RNA differ from nucleotides forming DNA strands by their sugar, which is ribose instead of deoxyribose. Further, RNA strands use the base Uracil (U) instead of T.
The translation process happens at the ribosomes, huge complexes of multiple proteins and structured RNA. Here, mRNA molecules are sequentially processed, whereby every three bases form a triplet or codon, which are decoded to a corresponding amino acid. Collinear to the mRNA sequence codons are processed and their corresponding amino acids are bound together with a peptide bond resulting in an amino acid sequence.

2.2 PROTEINS AS SEQUENCES OF AMINO ACIDS

There are 20 standard different naturally occurring amino acids, which differ by chemical properties of their sidechains. This sidechain is also called a residue. The backbone is identical for 19 of the 20 amino acids and can be connected by the peptide bond [19] depicted in Figure 2.3. The chemical composition

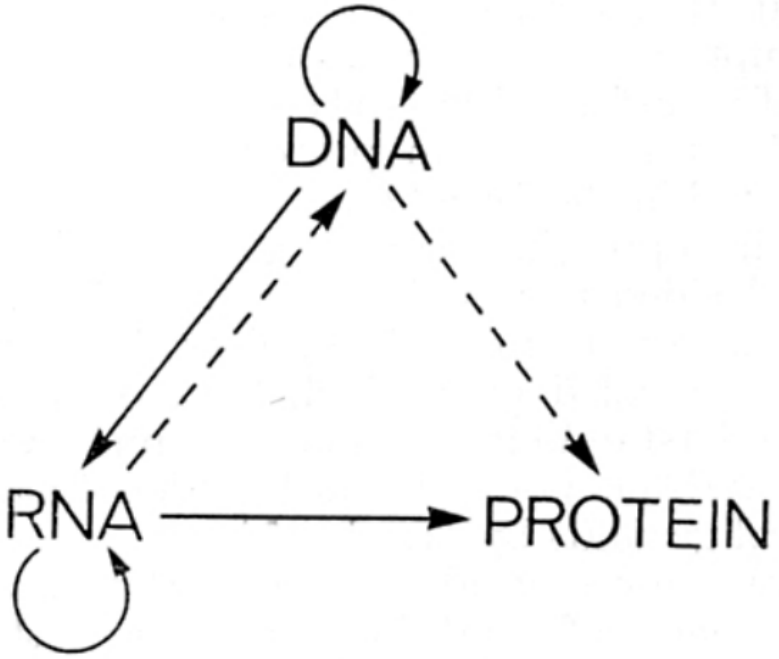


Figure 2.1: *(Source: (Crick, 1970)[1]) The central dogma of molecular biology showing the flow of information in living systems.*

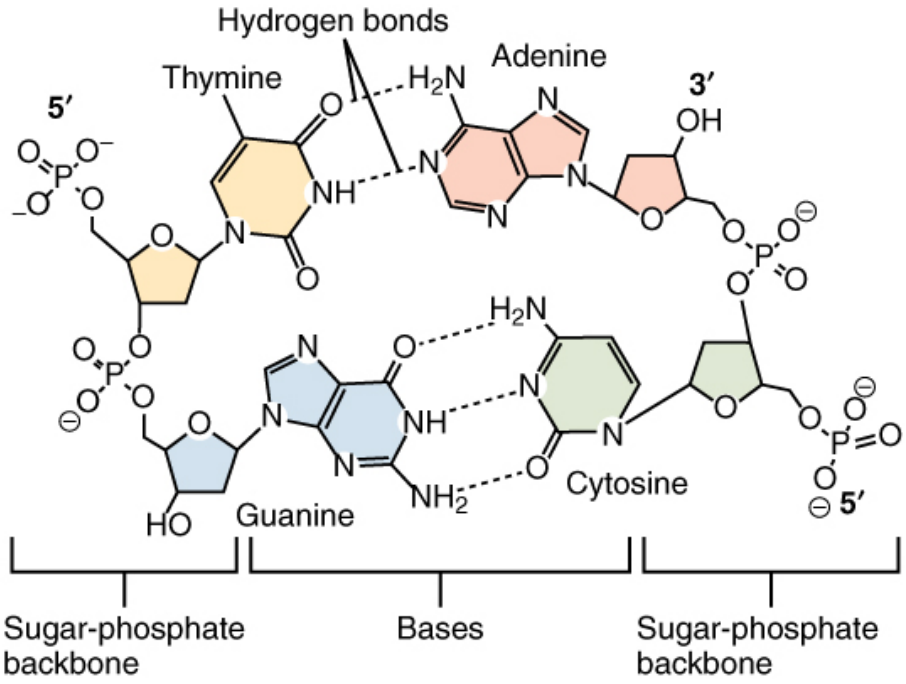


Figure 2.2: *(Source: https://upload.wikimedia.org/wikipedia/commons/d/d3/0322_DNA_Nucleotides.jpg) The four nucleotides form two base-pairs: thymine and adenine (connected by double hydrogen bonds) and guanine and cytosine (connected by triple hydrogen bonds). The individual nucleotide monomers are chain-joined at their sugar and phosphate molecules.*

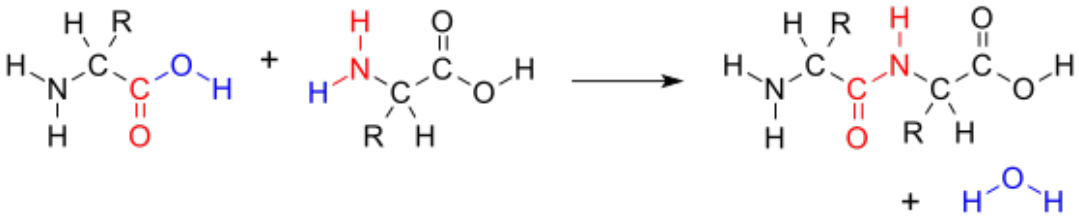


Figure 2.3: *(Source: https://en.wikipedia.org/wiki/Peptide_bond) The peptide bond (red atoms) connects two amino acids under the expulsion of water to a dipeptide.*

of a sidechain gives the amino acid its name and its special physicochemical attributes. The exception is proline, whose sidechain is connected twice with its backbone. It can still form peptide bonds.
This allows amino acids to form polymeric chain molecules, called polypeptides. Longer peptides, which fulfill biological functions, are called proteins. Proteins can be built of a near-infinite amount of possible amino acid sequences.
The bonds between the alpha carbon and the carboxyl carbon (phi-bond) and between the alpha carbon and the nitrogen (psi-bond) are rotatable freely. This leads to a plentitude of possibilities for the three-dimensional orientation of a protein's amino acids. Luckily, the majority of proteins have one or at least a few natural ways to pack their amino acids, determined by the physicochemical interactions between them and characterized by the lowest free energy [20]. Such a mutual placement of amino acids is called protein fold or the three-dimensional structure of a protein [21].

2.3 PROTEIN STRUCTURES

Proteins can be distinguished into the ones that fold autonomously (or with the help of chaperones) into a particular shape called protein structure and those proteins, that do not fold into a stable structure. The latter ones are called non-structured or disordered proteins. There are also partially structured proteins, which combine segments of both kinds.
Structured proteins can be described in terms of four levels of structural organization [22–25]. In this hierarchy (Figure 2.4), the amino acid sequence is described as the primary structure of a protein. Hydrogen bonds between the amino acid backbones build regular reappearing structural patterns, known as secondary structure elements. While there exist finer distinctions, the simplest and most well-known classification of secondary structure elements are alpha-helices, beta-sheets, and loops or coiled regions. The alpha-helices and beta-sheets can be viewed as less flexible building blocks, that can be arranged in any formation in space, connected by highly flexible loop stretches. This arrangement of secondary structure elements forms the tertiary structure. The next level in the hierarchy leaves the scope of a single protein sequence. As the quaternary structure, one understands a structural composition of multiple protein chains to form a functional biological entity.

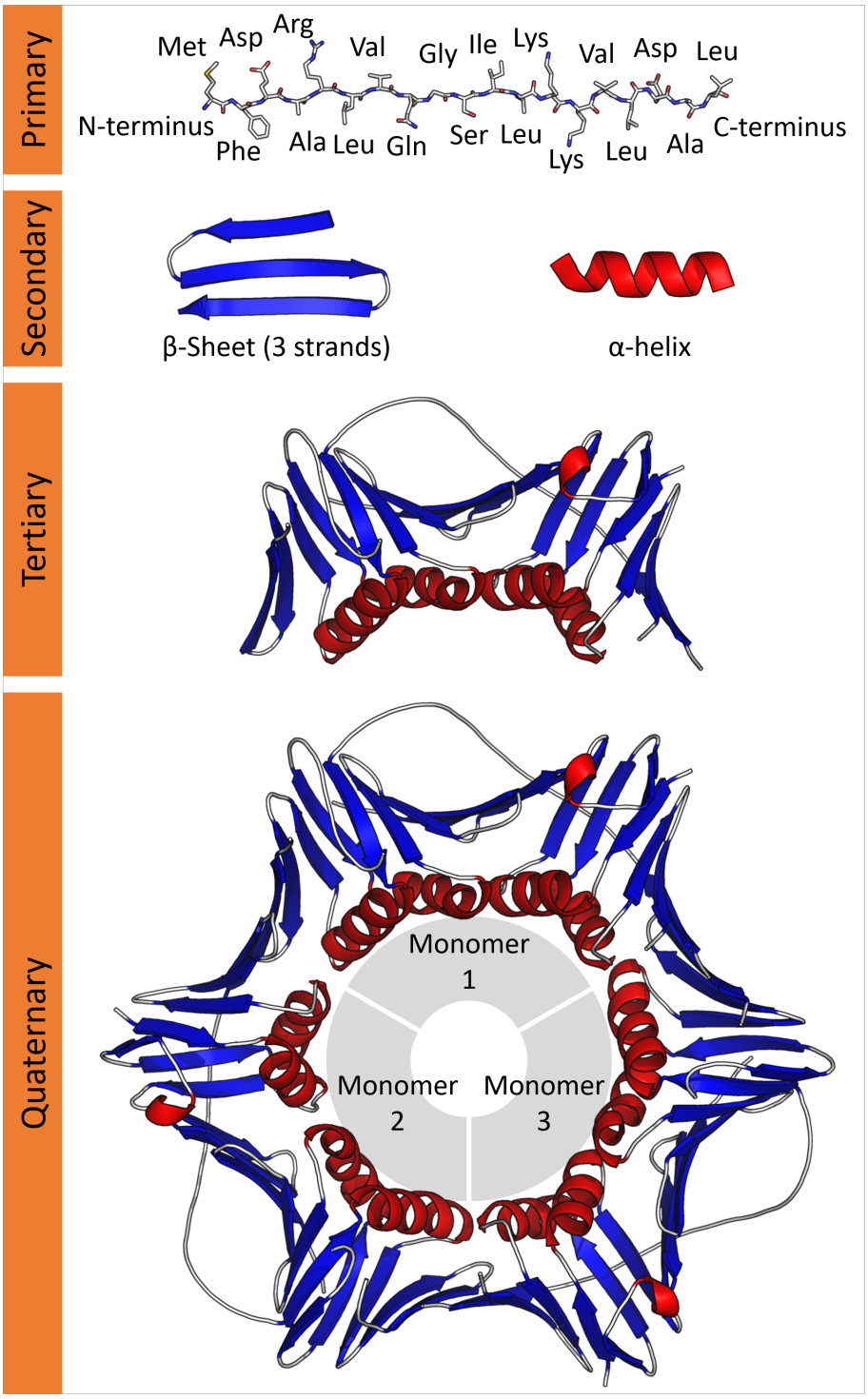


Figure 2.4: *(Source: https://en.wikipedia.org/wiki/File:Protein_structure_(full).png) The four levels of protein structure hierarchy on the example protein PCNA (PDB: 1AXC)*

Segments of proteins, whose fold is evolutionary conserved and that form independent functional units, are called protein domains [26, 27]. Often domains can be observed to specifically bind to other domains or low molecular weight ligands. While smaller proteins consist entirely of one domain, the combination of domains enables the construction of larger proteins with more possibilities regarding interaction partners. Smaller recurring structural elements are called structural motifs and are linked to specific roles in protein structure formation [28].

2.3.1 *Protein Function Through Interactions*

Most biological processes and functions are mediated by the physical and chemical properties of protein structures and their ability to form very specific complexes with other proteins, nucleic acids, lipids, low molecular weight ligands. Further, we call such tight and specific binding to other molecules interactions. While there are proteins, which function and form interactions in a disordered state [29], a very common way of proteins engaging in interactions is by the formation of more or less defined 3D structures. Protein structures are usually not completely rigid. The ability to slightly change their 3D conformation to adapt to the specific structure of a corresponding interaction partner is called induced fit and enables the interaction to a variety of interaction partners [30].
This concept can be expanded to partially structured proteins. Their disordered parts can form more or less structured interaction interfaces during the formation of a complex. Overall, the function of a protein is determined by the various dynamic interactions it facilitates and the complexes it participates in. Since most complex formations are determined by the specific folds of protein structures, there is a strong connection between protein structure and protein function [31].
Thus, when proteins perform their function, they inevitably engage in interactions with other biologically relevant molecules. This includes a wide variety of possibilities, ranging from binding smaller molecules in order to catalyze chemical reactions up to forming massive complexes building whole cellular compartments [32]. The only other type of biological macromolecules, which can fulfill similar tasks, are structured RNAs. This can also be done in cooperation with proteins, a prime example of such cooperation being the ribosomes. However, the vast majority of cellular processes are mediated by proteins.
As previously stated, protein function is determined by its structure. More precisely, proteins fulfill their functions through interactions with other biomolecules. Enzymes are interacting with their corresponding substrates, transcription factors interact with nucleic acid chains, membrane proteins interact with different cell membranes, signaling proteins interact with other proteins and signaling molecules and so on. All these interactions have to be very specific, such that only with the correct interaction partners a complex can be formed. This specificity is guaranteed by the particular structure of a protein, hence protein structure still determines protein function, but knowing the structure of

a protein without knowledge about the corresponding interaction partners, deductions about the function cannot be made, at least as long as one cannot precisely predict interaction interfaces and partners based on the structures.
For that reason, the investigation of protein structures without the context of their interactions is not sufficient and only through research of interactions a protein participates in, protein function can be deduced. Thus protein structure modeling does not suffice to fill the gap between known sequence and known structures when the protein function is of interest. Most protein structure determination experiments are designed in a way that they aim to resolve the protein structures together with their interactions partner, especially in the case of X-ray crystallography experiments. The possibility of multiple interaction partners makes the investigation of multiple experimental results for the same protein structure very valuable. One protein can interact with many different proteins using distinct interaction interfaces or the same interface and/or by switching its conformational state [33]. Often the binding of multiple putative interaction partners even excludes each other. Different structure resolution experiments can uncover different complexes formed by the same protein. This principle applies not only to proteins as interaction partners, but also to low molecular weight molecules and the combination of both types. When only the data from one experimentally resolved structure is used when performing a structural analysis, one can overlook important mechanisms and functions of a protein.

2.3.2 *The Relation Between Sequence and Structure Similarity*

The structural fold of a protein is determined by its sequence. It has been observed that similar sequences will fold into similar structures [34]. However, since protein function is determined by its structure and not by its sequence, the theory is that evolutionary pressure affects the structure stronger than the sequence of a protein, leading to stronger conservation of structure in comparison to sequence conservation. Empirically, this has been shown by Illergård er al. [35]. While structures are more complex to analyze than sequences, due to the additional dimension, the diversity of structures observed in nature is less than the diversity of sequences [28]. This observation indicates that the possibilities for protein structure folds, which properly fulfill their function, are limited. In other words, accumulating evolutionary changes over time can lead to highly diverse sequences folding into similar structures. As a conclusion, similarity in structure does not strictly imply sequence similarity (for example Figure 2.5), but empirically a high sequence identity is a very high indicator for a high structural similarity [34, 36].
This fact is relevant in practice because a similar structure is a strong indicator for a protein to have a similar function. Structures of proteins with a similar sequence to the protein of interest can be studied, when for the protein of interest no structure is available. Conclusions regarding protein function drawn from related structures can be transferred back to the original protein. Since even not directly related proteins may share similar structural domains, which

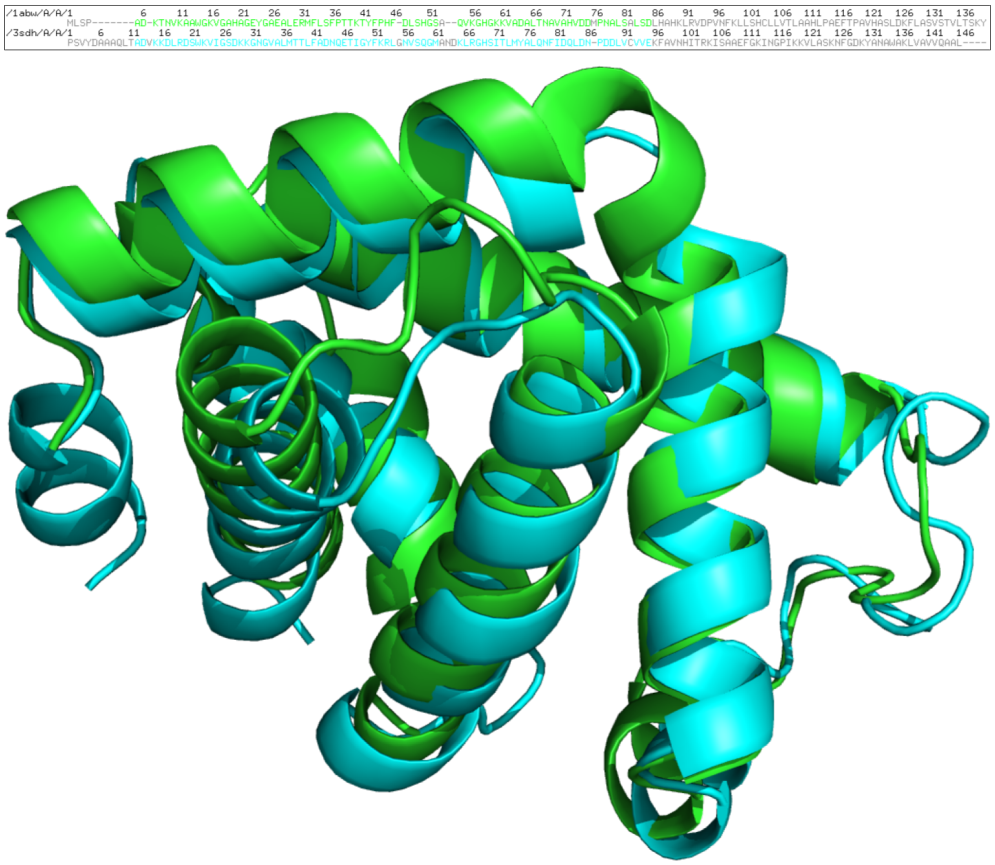


Figure 2.5: *Superimposition of structures of Hemoglobin subunit alpha HBA1 in human (in green, PDB id 1ABW, chain A) and Hemoglobin subunit 1 in Anadara inaequivalvis (in magenta, PDB id 3SDH, chain A); sequence identity of the homologous sequences:* 19%. *Above the structures: sequence alignment induced from the structural superimposition.*

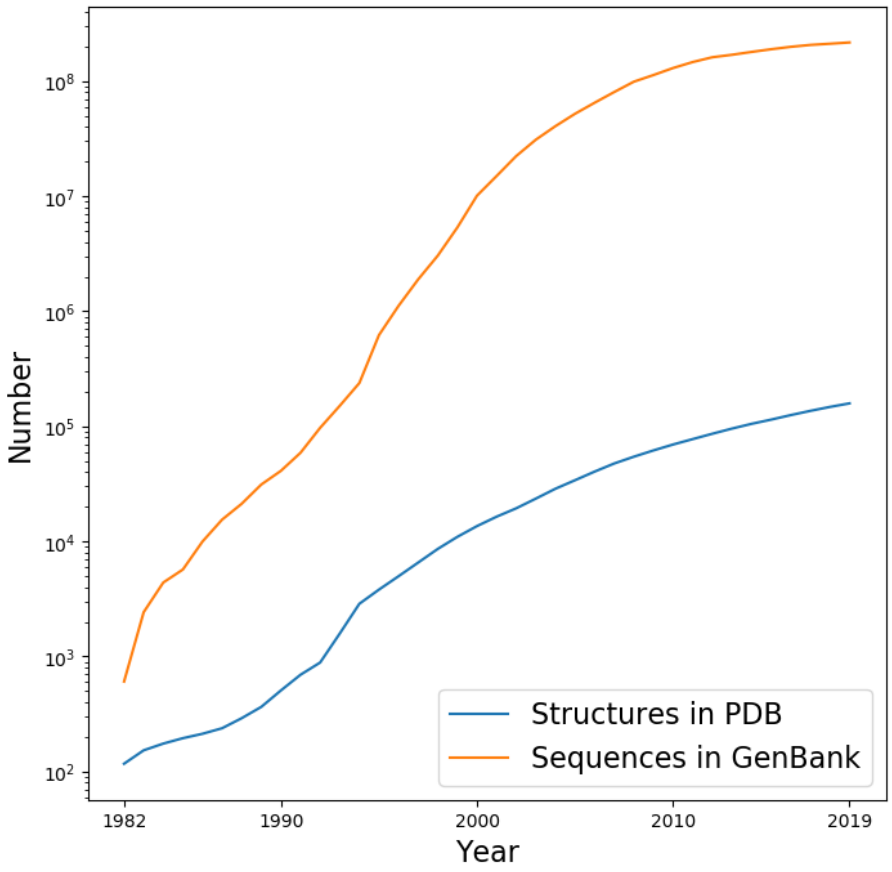


Figure 2.6: *Total number of available data over time: for the number of gene sequence records in the NCBI GenBank [39] and the number of structures in the PDB.*

can be linked to specific functions [37], the analysis of structures from proteins, which show sequence similarity only for a part of the sequence, can be very fruitful, too.
Since the structure of most proteins is not known, the previously explained relationship between similar structures needs to be used for filling this knowledge gap. For that purpose, there is a need to find similar structures, which is achieved by searching for similar sequences of proteins with known structures, which is explained in more detail in Chapter 5.
Currently, only for 22% of all human proteins there is an experimentally resolved structure in the Protein Data Bank (PDB) [38], the major repository for results of protein structure resolution experiments. Since more and more structures are experimentally determined over time, the question arises, whether the need for studying similar structures is just a temporary situation until all structures are available. While the state of universally available protein structures would be desirable, it seems a) very far away and b) may never be reached since the number of newly sequenced genes grows more rapidly than the number of newly resolved structures (Figure 2.6).

2.4 GENETIC VARIANTS

When comparing a specific genome to the reference genome of the corresponding species, their differences called genetic variation. The variant that is observed in the reference is called wildtype and all other variants we call mutant. There are many forms of genetic variants known. They can be described in terms of their general structure, such as their size or the biological event that caused the variation, and in terms of the region of the genome, in which they occur. One can differentiate three major classes of genetic variants: large-scale or structural variants, medium-sized insertions and deletions which are called indels, and point mutations, which are also called single nucleotide polymorphisms (SNPs) or single nucleotide variants (SNVs). The term SNV refers to all point mutations, whereas SNPs are only SNVs that occur in less than 1% of the population [40]. Structural variation can be specified further [41], but since they a comparably rare form of genetic variation in the human genome [42, 43] we ignore them in the scope of this thesis. Indels are more frequent than structural variations, but still about 25 times rarer than SNVs [42]. SNVs that change a codon and lead to an amino acid substitution in the resulting protein are called non-synonymous SNVs (nsSNVs), and the focus of this thesis is exclusively on nsSNVs.
In 2001 it was estimated that there are at least 10 million SNPs in the population [44], in 2019 there are over 69 million SNPs listed in dbSNP [45], which are reported in less than 1% of the population. Each individual can carry about 3 million SNVs [46] and from these 99% have a population frequency $< 1\%$ [47].

2.4.1 *Influence of nsSNVs on Protein Function*

The result of an nsSNV is a substitution of one amino acid in the protein sequence. Since a protein's function is fulfilled via its interactions, the impact of a mutation on protein function can be interpreted in terms of its impact on the interactions of a protein. We distinguish three basic modes of how a mutation can impact an interaction [48–50]. The most obvious mode is to directly alter an interaction, either by replacing an amino acid, which formed a non-covalent bond in the wild type with an interaction partner, destroying this bond, or by introducing a new bond to an interaction partner. The second mode is by destabilizing the protein structure, which may lead to formation of incorrect conformations and/or partial or complete misfolding of the protein. Allosteric effects define the third mode, where comparatively subtle structural changes impact an interaction although the site that was altered is distant from the interaction site. Such effects can become rather complicated, mitigated by a chain of minor molecular events and are difficult to assess. All these effects differ in their magnitude. Some mutations can completely disable all functions of the protein, some disable just one function and others only result in smaller effects [51]. For example, a mutation can impede the formation of a particular complex without having any effect on interactions with other partners. Other mutations may only change the binding affinity of an interaction instead of completely inhibiting the binding, resulting in even more subtle changes or even in an

increased binding affinity. Another example of a possible effect is the change in the turnover rate of an enzyme, leading to a change of the product molecule concentration in the cell, which could have further ramifications. Overall, we can hypothesize that variants in the protein core have an increased chance of inducing the complete loss of function, variants on interaction interfaces tend to have more subtle effects and variants lying on the protein surface that are not part of an interaction interface lead to rather neutral effects.

2.5 GENETIC DISEASES

Genetic diseases, or genetic disorders, are diseases that result from genetic alterations and can be observed as specific phenotypes. These alterations most commonly occur randomly and can be inherited from the ancestors. The term phenotype goes back to 1909 to the work of Wilhelm Johannsen as described by [52]. While the term can be used in various meanings, we focus on the distinction between neutral or healthy phenotypes and malicious disease phenotypes, especially in humans. On a population scale over a long time, the process of evolution through random mutation and recombination of genetic information is the driving force behind the diversity of life [17]. Zooming into the perspective of individual fates, the collateral damage of evolution in the form of genetic diseases seems very cruel. These diseases often do not directly lead to death, but cause abnormalities and/or decrease quality of life of the affected individuals. This makes fighting genetic diseases one of the main motivations behind plenty of scientific research. Cancer is usually not considered to be a genetic disorder, although it is caused by genetic alterations.

The set of mutations causing a genetic disease can range from individual point mutations up to very complex multi-mutation pattern. One distinguishes between germline mutations, which occur in germ cells either inherited from the parents or appeared before the differentiation of the germ cells, and somatic mutations, which were introduced into the genetic code during the lifespan of the individual [53]. Since all cells in the human body are descendants from germline cells, germline mutations are omnipresent in all cells and tissues in an individual, while somatic mutations spread from the time and place of their first origination through the replication cycle of living cells only to a limited number of cells in the body. Germline mutations are often the cause of the inheritance of a genetic disease. The earlier somatic mutations happen in the differentiation process in the life cycle of a multicellular eukaryote, the more it is spread to different tissues. Somatic mutations can cause diseases when occurring in a specific tissue while having a neutral effect in another tissue.

When going down the road from genotype to phenotype, one should never underestimate the complexity of the greater picture. The genotype is not alone responsible for the phenotype. Epigenetics and environmental factors also influence the phenotype. Cases, in which the presence of disease-associated mutations do not result in their expected phenotype are called variants with reduced penetrance [54] and can be very puzzling. Some can be explained by heterozygosity, meaning the variant is only carried by the recessive allele, some

are due to epigenetic effects and others remain unexplained.
The Online Mendelian Inheritance in Man (OMIM) database [55] collects genetic disorders and their corresponding associated genes. Disease phenotypes, which can be associated with malfunction or inhibition of a single gene or its corresponding gene product exclusively is called a monogenic disorder [56]. They need to be distinguished from diseases caused by a combination of perturbations, which are called multifactorial disorders or complex disorders. In OMIM (January 28th, 2020), the 5520 monogenic disorders outnumber the complex disorders that only have 694 entries. Surprisingly the same holds true for genes associated with monogenic disorders (3832) and genes associated with complex disorders (501).

2.5.1 *Monogenic Disorder*

As mentioned before, diseases resulting from mutations located in one specific gene are called monogenic disorders [56]. All patients that have the disease also share a perturbation in the gene. This could be, for example, a mutation leading to an amino acid change in its corresponding protein, which then disrupts the protein's function. If the causative mutation is known, the corresponding disease can be diagnosed easily via sequencing the corresponding genome segment. If therapy is available, this is especially helpful for diseases, whose symptoms are developing over time. However, due to the cases of reduced penetrance explained previously, this concept may lead to false-positive diagnoses. For understanding, fighting and curing monogenic diseases, the knowledge of the statistical connection between the specific mutation and the corresponding monogenic disorder is not sufficient. The impacts and implications of the mutation on the underlying biological system have to be evaluated thoroughly to design a potent cure.

2.5.2 *Multifactorial Disorder*

As the name suggests, multifactorial disorders have multiple mutations amounting together to a disease phenotype. Examples of such diseases are diabetes, heart disease and schizophrenia [56]. In comparison to monogenic disorders, they affect a much larger fraction of the world's population [57]. The interactions between the individual genetic variants in a multifactorial disorder can become extremely complicated, and multifactorial disorders are also known as complex disorders. The most infamous complex diseases are cancers, described in more detail in the next section. In the majority of cases, the malignant mutations are combinations of germline and somatic mutations. Since it is possible that completely different combinations of mutations result in the same pathogenic phenotype, a diagnosis based on specific mutations gets a lot more challenging compared to monogenic disorders. But it is not impossible: some mutations are more frequently observed in pathogenic phenotypes than others, and based on that observation statistical associations can be deduced. Thus, some mutations have a greater pathogenic potential than others. Resulting

from this observation, a whole field developed, which creates models that are predictive of the pathogenic potential of individual mutations. It will be in more detail introduced in Chapter 7.

2.5.3 *Cancer*

There are many definitions of cancer. In general, cancer is a group of cells with abnormal proliferation and many other abnormal cellular characteristics resulted from genetic alterations [58, 59]. Cancer is distinct from benign tumors, which are also characterized by abnormal proliferation but do not pose a direct threat to its neighboring tissue and its host. Cancer is a malicious tumor and thus able to invade other types of tissue causing metastasis [60]. Cancer can happen in all types of tissues, and different forms of cancer are generally broadly classified by their tissue of origin. Each cancer is different and often rapidly changes over time, which renders it the most complex existing disease. Some cancers are caused by very few mutations, but most often the accumulation of unfavorable somatic mutations in combination with malignant germline mutations is observerd during tumor progression [58]. A benign tumor can also develop into malicious cancer this way.
A collection of the traits of cancer that differentiate them from healthy cells is known as the hallmarks of cancer [61, 62]. It includes ten hallmarks, which can be seen as changes in cellular phenotype typical for cancer:

1. **Resisting cell death.** Cancer can disrupt the natural signaling pathways, which initiate apoptosis of damaged cells. The well-known tumor suppressor protein TP53, which is able to sensor cell damage and to start a signaling cascade that leads to apoptosis, is perturbed in many cancers, for example.

2. **Deregulating cellular energetics.** The uncontrolled proliferation of cancer cells requires more energy in comparison to healthy cells of the same tissue type. To fulfill that increased demand for energy, cancer cells are known to upregulate their glucose metabolism.

3. **Sustaining proliferative signaling.** The growth and eventual division of healthy cells are strongly regulated by signaling pathways. Cancer cells are upregulating cell-growth promoting signals to reach a state of uncontrolled proliferation.

4. **Evading growth suppressors.** The effect of the previous hallmark is also supported by another mechanism, namely the deregulation of growth-suppressing signaling pathways.

5. **Avoiding immune destruction.** To evade the destruction through the immune system of the host, cancer cells can produce immuno-suppressive agents.

6. **Enabling replicative immortality.** While healthy cells are limited regarding the number of cell divisions, cancer cells can reach a state, where their

replication is unlimited. Together with the resisting cell death hallmark, this leads to a seemingly immortal tumor, which when left untreated dies only with the death of the host due to the loss of nutrition supply. This ability is useful in molecular biology experimental routine since in vitro immortalized cancer linages can be stored forever and multiplied at will to produce research samples of constant quality.

7. **Tumor-promoting inflammation.** An inflammation, triggered by the tumor itself or by external causes, can change the metabolic environment of the tumor in a way supporting the development of the other hallmarks.

8. **Activating invasion and metastasis.** A cancerous tumor originates from one type of tissue. A further developed tumor, however, is able to initiate cancerogenesis in neighboring tissues of a different type. It is also possible for a cancer tumor to separate single cells, which can be transported through the lymphatic and blood system of the host enable metastatic colonies throughout the host body. This hallmark has the most negative consequences for the host.

9. **Inducing angiogenesis.** Angiogenesis is the biological process in which blood vessels create new branches. Cancer cells affect their surrounding tissue by initiating angiogenesis to ensure a sufficient supply of nutrients from the host.

10. **Genome instability and mutation.** The acquisition and development of all hallmarks are driven by (epi-)genetic perturbations. Cancer increases the rate of genetic variation and accumulates a larger number of mutations, which accelerates the development of all hallmarks.

While the hallmarks are numbered here, they do not have any particular order and their joint presence is not necessary to classify a tumor as cancer.

2.5.4 *Phenotypic Effects of nsSNVs*

Since mutations can affect the functions of proteins, they also can impact the resulting phenotype. Sometimes specific mutations co-occurring with genetic disorders can be statistically linked. Such mutations are said to be in association with the disease and are collected in databases, for example, ClinVar [63].
Predicting the effects of nsSNVs on a phenotype is equivalent to the investigation of the impact of the altered function of a protein on an entire biological system. Even when the alteration of the function is exactly understood, it is an immensely complex task. At first glance, this does not seem too complicated for a monogenic disorder, but the process in which the affected protein is involved may not be completely understood [57]. The issue becomes even more complicated when the association between the gene and the disorder is not known. Such cases are not rare, which can be seen by the investigation of the entry statistics of OMIM (`https://omim.org/statistics/entry`).
The prediction is particularly difficult for complex disorders. Since different mutations are causing the phenotype in combination here, the effect of a single

nsSNV on a phenotype can be relatively subtle. The phenotypic effect of nsSNVs can be approximated by their impact on protein function. The underlying assumption is that strong effects on protein function correlate with a strong contribution to the change in phenotype [49]. In this model, all the complex networks of biological pathways are ignored.

Since this assumption is a significant simplification of the underlying biological system, the prediction of phenotypic effects of individual mutations in this manner will never be an ideal method.

Overall, proteins can be more or less important and are more or less susceptible to mutations. The tendency is that the more interactions a protein participates in, the more important they are to maintain the phenotype and the biological system is affected more by mutations in that protein. This tendency has a secondary repercussion. Some proteins are more researched because of their known important roles in diseases like cancer, resulting ultimately in the existence of more data on them, including data on their interactions. At first glance, this sounds desirable, but when working in that field one should always keep this in mind since it introduces some heavy biases. In this thesis, we explore these and other biases that can obscure the computational predictions in this field in Chapter 7.

3 EXPERIMENTAL AND COMPUTATIONAL TECHNIQUES

3.1 EXPERIMENTAL METHODS FOR DATA ACQUISITION

3.1.1 *Genome Sequencing*

Sequencing techniques are experiments with the goal to determine the correct sequential order of nucleotides in a given DNA or RNA molecule (protein sequencing technologies exist, but are not commonly used and not considered here). Since the discovery of the structure of DNA [18] and the following formulation of the central dogma [1], it became clear that the mystery of life is coded in the nucleotide sequence of the DNA and the quest to crack the code was born. After some time, experimental methods were developed in order to sequence fragments of DNA (Sanger sequencing) [64] and through further improvements on that technique [65, 66] the foundation for the sequencing of the human genome was laid [67]. The famous first sequencing of the human genome by the Human Genome Project [68–70] which resulted in a total cost of $3 billion. The result defined the very first human reference genome, which simplified the task for future human genome sequencing efforts.
The most current reference genome at the moment, hg38, resulted from the 1000 Genomes Project [46]. The main focus of the project was to map all genetic variations in the human genome. As the name suggests, the idea was to sequence 1000 genomes and with the following sequence alignment, one could identify the varying locations in the genome.
In comparison to the Sanger sequencing, modern sequencing techniques are characterized by their higher throughput of sequenced nucleotide per time and cost. They are generally described as Next-Generation Sequencing (NGS) techniques and are discussed in the following.
All sequencing techniques involve the fragmentation of the given sequence since only fragments of a certain length range can be determined. The experimentally determined sequence of such a fragment is called a read. Different sequencing techniques are mainly distinguished by the corresponding typical read lengths, which splits the field into short-read assays and long-read assays. Other attributes are the rate of errors per nucleotide and how much time and money are required for the generation of the reads. To reassemble the whole sequence of a given sample, reads have to be processed by computational methods. Different read lengths and coverage rates require appropriate computational methods. Here it should be noted that the existence of a corresponding reference genome simplifies the problem significantly.
As the name suggests, sequencing by ligation (SBL) techniques exploit the biological process of DNA ligation. In contrast to the biological process, in SBL the DNA ligase does not join two DNA double-strands, but target sequences with labeled oligonucleotides, for example in the SOLiD platform [72]. Sequenc-

Method	Read length	Error rate	Cost per Gb
Sanger sequencing	short	0.1%	$2,400,000
Sequencing by ligation	short	0.1%	$70 – 130
Sequencing by synthesis	short	0.1%	$7 – 150
Single-molecule real-time sequencing	long	3 – 13%	$7 – 40
Synthetic-long read sequencing	long	0.1%	$30 – 150

Table 3.1: *Different DNA sequencing technologies; Gb: a million bases; this table is based on information from* https://en.wikipedia.org/wiki/DNA_sequencing *and from Goodwin et al. (2016)[71].*

ing by synthesis (SBS) uses another biological DNA-related protein, the DNA polymerase (e.g. Taq polymerase) in a similar fashion as in the polymerase chain reaction (PCR). In SBS assays, the target sequence fragments are fixed in place, fluorophore-labeled single nucleotides are added step by step, one type of nucleotide per step. The DNA polymerase adds the nucleotide to the complement of the target fragments, enabling the detection of the fluorophore, which is cleaved from the complex in the process, corresponding to the spatial fixation. Example platforms using SBS are Illumina and Qiagen [71]. In contrast to SBS in single-molecule real-time (SMRT) sequence, the target fragments are not fixed. Instead, one DNA polymerase molecule is fixed to a specific location hence 'single-molecule'. The target fragments are complemented by labeled nucleotides. In SMRT all four types of nucleotides are always present, enabling the polymerase to work as fast as *in vivo*, hence 'real-time'. Each nucleotide concatenation is accompanied by the cleavage of a fluorophore, which enables the detection based on the specific location of each polymerase. This allows for the sequencing of larger fragments in one go. Synthetic long-read sequencing techniques are, in principle, extended short read sequencing techniques. Here, the target DNA is first fragmented into longer pieces and then separated into sets of a several thousand fragments, which are then further fragmented to short read size and tagged. All short fragments are finally sequenced and with the help of the tags long reads can be rediscovered.

There are also further specialized DNA sequencing techniques. With single-cell sequencing, for example, it is possible to identify genomic differences in individual cells and to reveal allele frequencies of individual samples.

In medical practice, the cost of sequencing is of utmost importance. To reach the goal of personalized medicine, i.e. for each patient provide a treatment tailored to their individual characteristics, we need to be able to obtain as much genomic information as possible for individual patients for the lowest possible price. With the advent of next-generation sequencing techniques this became a realistic opportunity. The resulting data amounts call for the development of computational methods harnessing the potentials of the ongoing advancements in sequencing technology [73].

Sequencing technology is not limited to DNA. RNA sequencing is able to identify all transcribed segments of the genome and is especially important when studying protein expression profiles since RNA sequencing can tell which genes are transcribed and what are the quantities of the resulting gene products.

While the world of sequencing technology is far more versatile and fascinating than described here, for this thesis we only focus on the processed data produced by sequencing technology. Further, we also treat computational methods for the processing of raw sequencing data as a black box, assuming that the task of gene annotation and variant calling are solved.

3.1.2 *Protein Structure Determination*

The spatial image of proteins was a mystery for a long time in molecular biology. In 1958, the first protein 3D structure was experimentally resolved, which was the structure of myoglobin [74]. This feat was possible with the technique of X-ray crystallography, which is still the most prevalent protein structure determination technique and consists of two steps. In the preparation phase, the molecules, for which the structure should be resolved are chemically driven into a crystal state, which means a regular symmetric repeating arrangement of so-called asymmetric units. In the main experiment itself, the crystal is penetrated by X-rays, which are diverted by the repeating organization of the crystal lattice in a way specific to the substructure of the asymmetric units forming the lattice [17]. The diverted X-rays are detected and the structure of the target molecule can be deduced from the specific geometry of the diversion. The complicated preparation of the crystals is comparatively labor-intensive. A suitable crystallization process of each protein needs to be determined and for some, no crystals can be derived despite the best effort. Transmembrane proteins are especially complicated to crystallize because of their hydrophobic surfaces facing the membrane. The advantages of X-ray crystallography are the possibility to achieve very sharp resolutions and the ability to resolve large molecules, including protein complexes. A disadvantage is that it is just a snapshot ignoring all possible dynamics of the resolved structures.
Nuclear magnetic resonance (NMR) spectroscopy is an experimental protein structure determination technique that is capable of capturing protein structure dynamics to a certain extent. In an NMR experiment, the molecule structure of which is being resolved is put into the influence of a strong magnetic field [17]. This construction is then probed by radio waves of differing intensity. Different intensities put different hydrogen protons (and other atoms with an uneven number of protons with a less strong signal) into a state of resonance, which can be measured. The result is a spectrum curve showing peaks for different energy intensities. Which peak corresponds to which atoms are determined by their chemical environment. Thus one can deduce the distance constraints between atoms in the molecule of interest. In NMR it is also possible to detect fluctuations on an atomic level in the structure, which means protein structure dynamics can be detected. The greatest disadvantage of NMR is that the larger the structure the more the peaks in the spectrum start to overlap. Then their structural determination becomes unfeasible at some point. Thus, NMR is only applicable to small proteins. Recent developments are expanding the applicability of this technique to larger structures.
Recently, a third major protein structure determination method starts to es-

tablish itself, the cryo-electron microscopy (cryo-EM), which is a specialized version of electron microscopy (EM) setup at very low temperatures [75, 76]. Cryo-EM opens EM for resolutions sharp enough for the determination of structures at the level of individual atoms, but currently still cannot compete with X-ray and NMR structure determination their productivity. The greatest contributions offered by Cryo-EM are the structures of large protein complexes, since there are no limitations on the size of the molecules of interest or their type and no need to laboriously produce crystals, enabling the resolution of very large complexes including membrane protein complexes and other molecules that represent a challenge for X-ray crystallography and NMR.
The raw experimental data is transformed into atomic coordinates (with or without hydrogen atoms) using special software specific to each experimental technique. The results from every protein structure determination experiment are submitted to the PDB, independent of the experiment type (see Figure 3.1). Each structure is stored in files with a special file format, which contains all the information from the experiment, most importantly including the atomic coordinates. Such a file can contain one or multiple models, which are made up of one or multiple protein chains. Each chain consists of a list of residues, which are basically lists of the residues's individual atoms.

3.1.3 *Clinical and Experimental Annotation of Impacts of Genetic Variants*

One key aspect of this thesis is the analysis and prediction of the impact of genetic variants. To train models and evaluate the quality of analyses and predictions, the usage of variants with known impact is inevitable. There are two major ways to create such annotated variants: either their impact could be clinically observed *in vivo*, or they can be experimentally analyzed *in vitro*. Clinically observed variants are usually reported with respect to their association with a pathogenic phenotype. Here, multiple independent observations are crucial to exclude random cause-effect deductions. Experimental annotation of genetic variants is often very specific. If only the impact on one particular function of the protein is tested, for example, one loses much of the surrounding perspective.
A frequently used database collecting clinically annotated variants is ClinVar [63], which has an integrated system rating the quality of each variant. Entries from singular submissions are rated with 1 star. Entries with multiple submissions, which do not contradict each other, are assigned 2 stars. Entries submitted by expert panels receive 3 stars and entries, which are mentioned in practical guidelines are granted with 4 stars. Currently, ClinVar holds slightly over 1 million variants, whereby nearly $100,000$ entries have 2 stars or more. While ClinVar includes all types of genetic variations, there are databases that specialize in that regard. For example, dbSNP [45] contains only SNPs or dbVar [77] contains only structural variations. Other databases are specialized for the collection of variants that can be associated with specific phenotypes, like COSMIC [78], which concentrates on somatic genetic variants associated with human cancer, or HGMD [79] that contains variants associated with all kinds

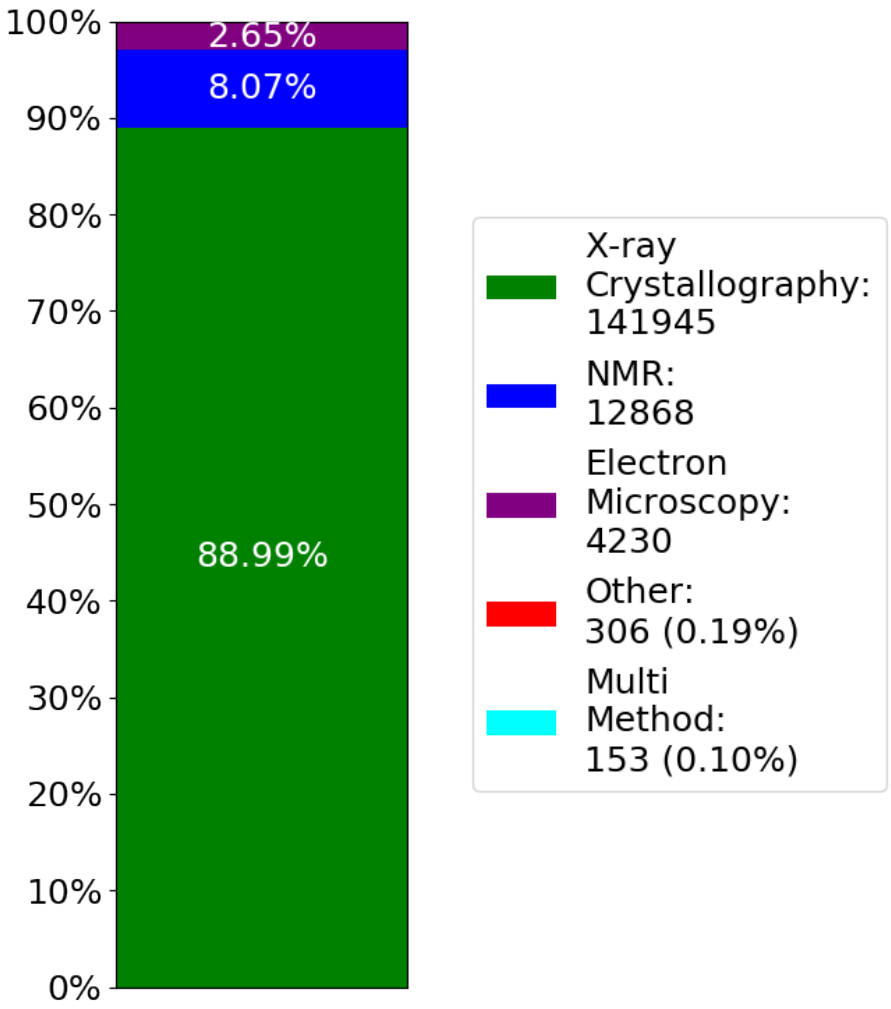


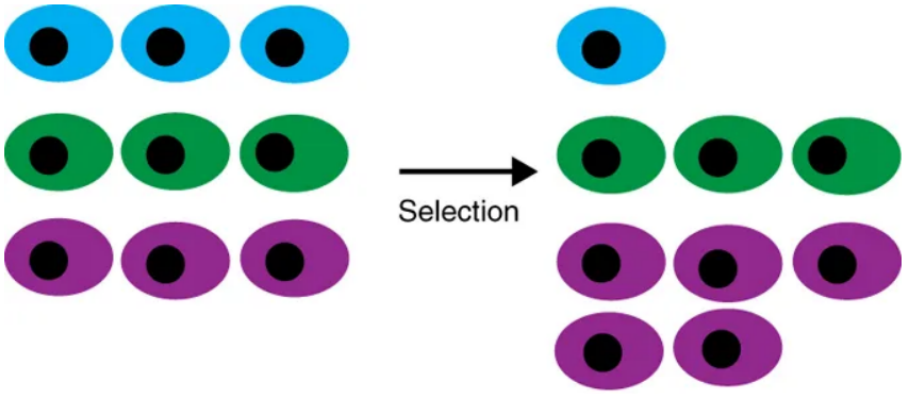
Figure 3.1: *Number of structures in the PDB (January 2020: https://www.rcsb.org/stats/summary) by experimental technique.*

of human Mendelian diseases.
An important group of experimental methods analyzing the impact of genetic variants on protein-protein interactions are protein-protein binding assays, where the strength of the interaction between two proteins is measured. To estimate the impact of a mutation on that interaction, the assay has to be performed on the mutant protein as well as on the wildtype protein. The most comprehensive database collecting the results of such experiments is IntAct [80]. The price for the large size of the database is paid in a loss of precision since the database combines the outcomes from many different experimental techniques into a rough classification scheme. A similar, but smaller database containing more detailed entries is SKEMPI [81], which usually contains exact values for the change in binding affinity for its entries.
A type of experimental assays for the estimation of the impact of genetic variants that is gaining popularity recently are deep mutational scanning (DMS) assays [82], which benefit from modern high-throughput experimental techniques, especially NGS methods (Figure 3.2). The goal of a DMS experiment is to assess the impact of as much as possible genetic variants on the function of a single protein. The first step is the creation of a variant library, usually in the form of a cell-based assay, whose cells contain the protein of interest artificially introduced using some vector. The trick is to obtain a roughly uniform distribution of mutants throughout the assay. Fruther, in the assay there is a possibility to select the cells based on the function of the protein of interest. The evolved cell population is sequenced afterwards and from the allele frequencies of each position of the protein, one can deduce the individual impact of each variant. The idea behind that deduction is based on the selection that reduces the number of cells carrying variants, which disrupts the target protein function, leading to a reduced allele frequency of the variant. The resulting functional impact estimations for all possible mutations for individual proteins are highly valuable in understanding the contribution of individual residues to the overall protein function.

3.2 COMPUTATIONAL METHODS

3.2.1 *Sequence Similarity Search*

Sequence similarity search techniques are one of the most fundamental methods in computational biology. Given an input query sequence and a database of sequences, the goal is to find a subset of the database with sequences similar to the query sequence, often called hits. Over the years, a variety of methods have been developed, whereby the focus was primarily to improve two characteristics, runtime efficiency and sensitivity. In sequence similarity search tools, the sensitivity is the ability of the method to find more distantly related sequences. The most well-known search tool is BLAST [83]. It divides the query sequence into k-mers (subsequences of length k), which are then matched against all sequences in the database. This matching is efficiently performed due to a preliminary indexing of the database. All matched k-mers are elongated in both directions, resulting in a local alignment. In this process, newly added matches



Variant	Mutation	Counts (input)	Counts (selected)	Functional score
Blue	A60P	3	1	0.33
Green	WT	3	3	1
Purple	S36T	3	5	1.67

Figure 3.2: *Taken with permission from Fowler and Fields (2014) [82]: Deep mutational scanning draws on high-throughput DNA sequencing to assess the functional capacity of a large number of variants of a protein simultaneously. First, a library of protein variants is created and introduced into a system where the genotype of each variant is linked to a selectable phenotype. Second, a selection for the function of the protein is imposed. Variants with high activity increase in frequency, whereas variants with low activity decrease in frequency. High-throughput DNA sequencing is used to measure the frequency of each variant before and after selection. These frequency data are analyzed to generate functional score for each of the protein variants.*

are scored by a substitution matrix and it is possible that elongated matches can fuse with other matches. The elongation is prolonged as long the scores of the added matches are above a certain threshold. Each alignment with a sufficient score is returned as a hit.
Another class of sequence similarity search methods are hidden Markov Models (HMM)-based methods. The latter type of method calculates a profile HMM, which is based on the sequences in the search database [84]. The target sequence is matched against the profile HMM by calculating the path corresponding to the target sequence through the profile HMM that has the maximum probability. A set of models from the data bank, for which this probability is high enough, correspond to a set of related sequences that form a list of hits. A well known HMM-based method is HMMER [85]. Modern methods are able to be balanced between efficiency and sensitivity by the user.
The sequence similarity search is a key step in StructMAn (Chapter 5), where many sequences have to be searched in a static search database with high sensitivity. To fulfill these demands, the method of choice was MMseqs2 [86], which is able to handle multiple searches in parallel.

3.2.2 *Pairwise Sequence Alignment*

The diversity of life is driven by evolution, which takes place by the introduction of random mutations into biological sequences. From this principle follows that similar biological sequences originate from a common ancestor sequence. More similar sequences indicate a closer evolutionary relatedness. Sequence similarity can be deduced by a list of computational methods, where sequence alignment is the most prominent. The goal of a pairwise sequence alignment is, in general, given two sequences of characters to match as many as possible identical or similar characters in both sequences onto each other by shifting the sequences with respect to one another and introducing gap characters into the sequences. Since all important biological macromolecules are representable as sequences of characters of a defined alphabet, sequence alignment methods are fundamental in computational biology. Sequence identity is the proportion of identically matched characters in a sequence alignment. While any mapping of two sequences with any amount of added gaps is technically an alignment, the most desired alignments maximize sequence identity or some other similarity measure.
For the alignment of protein sequences, in particular, a more complex similarity measure is taken. Compared to nucleic acid sequences, the alphabet of possible characters is larger and since amino acids can often be replaced by another amino acid with similar chemical properties, alignments of protein sequences may have a comparably low sequence identity and the proteins can still be considered as evolutionary related and perform identical or highly similar functions. For that reason, instead of taking just identically matching positions into account, usually a substitution matrix is used in which entries contain scores for all possible matched amino acid pairs, where identical or chemically similar amino acids receive a positive score, while dissimilar amino acid pairs

get a penalty in form of a negative score. There are two well-known sets of substitution matrices, Point Accepted Mutation (PAM) matrices [87] and BLOck SUBstitution Matrices (BLOSUM) [88]. Instead of maximizing the sequence identity, all pairwise scores are summed and the total penalty is minimized. Furthermore, there are additional penalties for opening and for elongating a gap to prevent alignment algorithms from introducing a large number of gaps to match a single character and ripping the sequences apart in the process. Two types of pairwise alignments are considered in computational biology, to solve different problems: local and global alignments. For example, aligning the sequence of a single gene with a much larger sequence of a chromosome requires to match the gene completely to one place in the chromosome without spreading the gene sequence all over the larger sequence. This scenario requires local alignment, since only a small part of one of the sequences is covered. Another application is the alignment of two protein domains, which should match approximately from end to end without many gaps, assuming some evolutionary relation. This scenario requires global alignment, where the majority of both sequences are matched.

3.2.3 *Computational Prediction of Protein Three-dimensional Structure*

An attractive option to span the bridge over the gap between the amount of protein sequence data and the amount of protein structure data is the *in silico* protein structure prediction. In most general terms, it is the prediction of the three-dimensional structure of a protein given just its sequence. In a perfect world, this method can replace the need for protein structure determination experiments. However, even after huge amounts of research and development in this area, protein structure prediction methods are still struggling with certain challenges. The sheer amount of possible arrangements of the amino acid sequence in three-dimensional space makes the base complexity of the problem nearly unsolvable. At a first glance, one could think that since the folding process of the protein is determined by physical interactions between the atoms of the protein molecule, it should be possible to simulate this process from the first principles, hence delivering the protein structure. However, the correct physical behavior in such a complex system is not fully understood yet. Still one could argue that the current physical models would suffice to simulate crudely enough to reach an acceptable solution. Currently, the computational complexity of such a simulation starting from the unfolded amino acid sequence by far exceeds the capabilities of the available hardware systems.
Hence, other methods abridging the real folding process are created. The practice has shown that protein structure prediction can be performed successfully if the structure of a closely evolutionarily related protein is known. The latter is obviously not always the case. Thus, protein structure prediction methods can be categorized by the amount and detail of data from other proteins required. Homology-based protein structure prediction is able to produce the most accurate predictions and requires the most specific data. Also known as template-based protein structure prediction, these methods use an experimen-

tally resolved protein structure of a protein homologous to the target protein as a template and model the amino acid sequence of the target protein around the template. Naturally, the more similar the template protein and the target protein are and the higher the quality of the structure determination experiment of the template is, the more accurate will the predicted model be.
The two most well-known homology-based protein structure prediction methods are Modeller [89] and SwissModel [90]. While having differences in details, the core principle is the same and is explained in the following. The first step is the alignment of the target and the template protein sequences, which sometimes can be corrected by using the 3D structure of the template. Matching amino acids of the target protein in the alignment can be used initially to reconstruct the backbone of the model by copying the coordinates of matched amino acids backbone atoms from the template. What is left are the gaps in the alignment, which are reconstructed as backbone loops bridging the previously placed fragments of the backbone. Obviously, this step is harder the larger the number of and the longer the gaps are. For that reason, it is harder to model structures based on a template with low sequence similarity. If the template is too far related, the predicted structure will be probably wrong. So the coordinates of the backbone are reconstructed. In the next step, the empty space in the model is filled with the sidechain atoms which are placed in as correctly orientated as possible by using a rotamer library. Finally different local optimizations are performed. Usually, this process is repeated several times, resulting in a list of possible predicted structures. Structure quality assessment methods then order the resulting structures to present the most promising solutions.
A more difficult problem is the protein structure prediction without any structure from an evolutionarily related protein, called template-free modeling. Methods in this field are usually based on conformational sampling [91], which uses conformation libraries containing experimentally resolved structure fragments for short sequences. Based on sequence similarity, such fragments are assembled against the sequence of the target protein, which results in a set of possible combinations of fragments representing individual conformations. Then, Monte Carlo simulations are used to find conformations with low energy, which are further refined similar to the optimization process done in template-based modeling. Examples of methods of this category are I-TASSER [92] and Rosetta [93].
Recently, template-free methods are improving at a fast rate due to the inclusion of deep learning methods. Using multiple sequence alignments of evolutionarily related sequences as the input, deep neural networks are able to predict residue-residue contacts and distances based on evolutionary coupled mutations [94]. When one residue of a pair of residues, which are interacting with each other in the structure, is mutated and loses the interaction, the other residue can be mutated simultaneously or shortly after that to compensate for the loss and thus reinitiate the interaction. As a result, such mutation pairs tend to cooccur in related sequences, even if they are not close in the sequence. This enables a more accurate prediction of long-range contacts and, together with the prediction of secondary structure elements and relative solvent accessible

area, the conformation space can be filtered much more precisely.

3.2.4 *Supervised Machine Learning Methods*

Described very broadly, the goal of machine learning methods is to create predictive models that are able to assign labels to samples [95]. Samples can basically mean anything that can be grouped in any way, for example, images, texts, patients or any kind of data. The samples have to be cast into a mathematical framework describing properties of each sample. This is commonly a so-called feature vector of numerical values, called features. For the example of images, the features could be the color values for each pixel. The goal is to find and recognize a certain logic or mathematical patterns in the features for a given set of samples. This set is called a training set. Based on the patterns learned, the samples are either scored numerically in regression problems or labeled in classification problems. When the corresponding ground truth scores or labels are unknown for the training set, we speak of an unsupervised learning problem. This class of problems is not further discussed in this thesis. In the opposed setting, supervised machine learning requires a training dataset of labeled samples to infer the relationship between the features of a sample and the corresponding label. This is done in the training process of the model. The more features are present, the more accurate the model can become. However, also more samples are needed to correctly understand the relationships between the features and the labels. In good practice, the training of a model is followed by a testing phase, in which the performance of the model is tested by predicting the labels for samples, for which the labels are known but which were not present in the training. It is of utmost importance that the samples used in the training and the samples used in the testing are non-overlapping and do not share too many common features. A perfect model is trained on an infinite amount of error-free independent training samples and uses as few features as possible to predict all labels correctly.
In practice, multiple problems and challenges arise. First, the more complex the underlying system, the more complex the model has to be. Thus, more features are required. In this thesis, biological systems that are very complex are considered. Second, if many features are present, many labeled samples are required. Labeled samples in biological problems are the results of expensive experiments, consequently they cannot be produced in endless numbers. This often leads to situations where one has a lot of features for a limited amount of annotated training samples. In these scenarios, machine learning tools tend to produce too complex models, which are able to perfectly explain the training samples. This is achieved by misinterpreting little details in the feature space and ultimately leads to misusing these details on unseen data, which results in false predictions. This behavior is called overtraining and is an indicator that the produced model is too complex for the underlying problem. Overtraining can be prevented by either increasing the amount of training data, which increases the complexity of the underlying problem and thus matches a more complex model, or by simplifying the model, for which there are appropriate techniques

for almost all machine learning methods.
Following Occam's razor, the complexity of the model should be appropriate to the problem. Often simpler machine learning methods suffice to explain the problem exhaustively. A large class of machine learning methods is based on the assumption that the labels are computable by a linear combination of the feature values. The difference in the true labels and the labels predicted by such a function is called error. Linear machine learning methods are the result of a linear combination of the feature space, which is minimizing the error. For many of the problems there is no analytical nor optimal solution to the minimization problem. There are many different linear methods and the mathematical background is very rich, so it cannot be discussed in detail in this thesis, but we want to mention that studying linear methods is a good entry point for getting involved with machine learning methods in general.
The only machine learning method used in this thesis is the random forest, which is a supervised method. It is based on decision trees (Figure 3.3), which are supervised learning methods themselves. In the training phase, decision trees separate the training samples by connecting a series of decisions. A single decision divides a set of samples based on the values of each sample for a single feature. This division is optimized to separate the corresponding labels in the resulting subsets according to some measure (e.g. the Gini index). The decisions are organized in a tree structure, non-leaf nodes represent an individual decision that is splitting a given set of samples into two subsets, which are then given to its two child nodes individually. If all samples of a resulting node share the label, the division process is stopped and the node is declared as a leaf node with the corresponding label.
When using a trained decision tree for predicting the label for a sample, one traverses the tree from the root to a leaf node. The path is determined based on the decisions, which are made based on the corresponding feature values. The leaf node that is reached returns the label, which was assigned to it in the training process, as the prediction.
If in the training phase, the the fact that all labels in the sample set for a node is the only stopping criterion, then the resulting tree is called full-grown. Hence, a full-grown decision tree on a sample that is part of the training set will always predict the correct label, meaning that full-grown decision trees have no training error. Obviously, in testing and application this is not useful, because it is likely that samples will not coincide with the samples from the training set, and the full-grown decision tree turns out to be hopelessly overtrained.
The counter method for the overtraining of decision trees is pruning, i.e. the introduction of other stopping criteria. Pruning leads to smaller trees and by that, the leaves may not have identically labeled sample sets anymore, which is leading to an increase in training error, but hopefully in a decrease in testing error. However, too much pruning will again result in an increase in the testing error, making the amount of pruning a central hyperparameter (i.e. parameter that is not optimized in the training phase) for constructing decision trees.
The difficulty with decision trees is the choice of the decisions and the order to organize them. Using a decision that seems suboptimal at first glance, could lead to a better predictor overall.

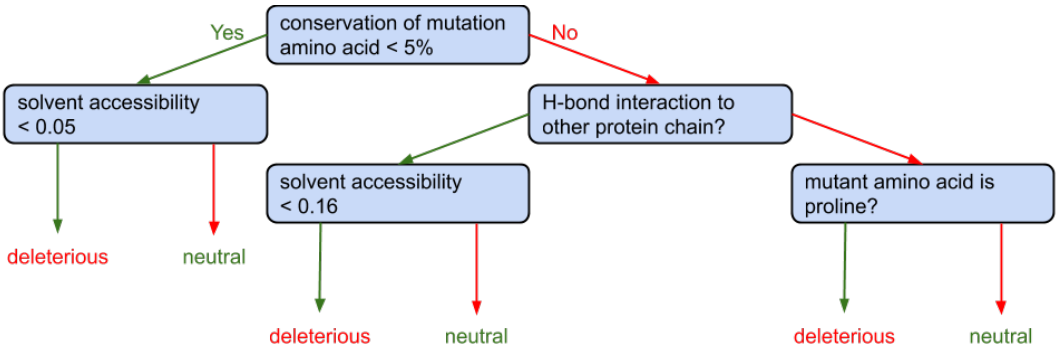


Figure 3.3: *Example decision tree, for the task of predicting if a mutation has a deleterious effect or a neutral effect.*

Random forests are a method to overcome this problem. Instead of regular decision trees, random decision trees are used. The growth of random decision trees is perturbed by random events, usually limiting the choice of features used for the single decisions down to a random subset of all features. Random forests grow multiple random decision trees, which due to the random events now have varying topology from each other. Each of the trees is handled as a weak predictor and combined together they form a strong predictor, the random forest. The important hyperparameters are then the pruning of the single trees and the magnitude of the perturbations through the random events. The advantage of a random forest compared to other machine learning methods is its ability to incorporate dissimilar sets of features, for example, the combination of categorical and floating-point values is possible. Further, its moderate computational costs allow the inclusion of large amounts of features for a great number of samples. Whether more features make a better random forest predictor is to be debated at another place.

4

ASSESSING THE SOLVENT ACCESSIBLE AREA FOR PROTEIN STRUCTURES WITH LIMITED QUALITY

In the development of StructMAn, we implemented many types of structural analyses, one of which one was the differentiation between residues lying in the protein core and residues located on the surface of a protein. This is done by estimating how much a residue has access to the solvent surrounding the protein. When working with a protein 3D structure with atomic resolution, the most exact measure for the solvent accessible area is based on the area one obtains when rolling a small probe sphere over the van-der-Waals surface of a protein [96]. This area is called the solvent accessible area (SA) of a protein and can be broken down into the contributions of each residue. In order to address the different volumes of different amino acids, amino acid type-specific maximum SA values were calculated [97] and the SA value of a residue divided by the corresponding maximum SA results in the relative solvent accessible area (RSA), which can be used to compare the solvent access of different types of amino acids.
We tried several methods for the RSA calculation and all of them computed first the whole SA of the protein and then separated the SA into the contributions from individual residues. One major motivation of StructMAn is the annotations of nsSNVs, hence single amino acid positions were in the focus and thus the structural analysis of individual residues. This lead us to ask ourselves if we could come up with a measure for solvent accessibility, applicable individually for each residue, which would then be more efficient than the established methods.
After continuing the development of StructMAn at some point, it became clear that the RSA calculation never will be a bottleneck. But the ideas we had in this area remained interesting and we were able to develop a new measure for solvent accessibility, that can be applied more universally, not only in the scope of structural annotation on nsSNVs. We submitted a publication on the subject to *Bioinformatics*, which is currently under review (minor revision).

4.1 INTRODUCTION

Residues located on the surface of a protein and residues buried inside the core of a protein have different functions. Thus, the distinction between residues on the surface of a protein and in the core of a protein has a lot of implications and applications [98] and mutations tend to have a different mode of how they impact protein function based on the location of the mutated residue in the structure. On the one hand, mutated residues buried in the protein core are known to be more disruptive for the function of the protein [99–102], on the other hand, mutated residues located on the surface of a protein can affect an

interaction interface and by that change the stability of the resulting complex or even inhibit the complex formation completely [51]. Mutations of a residue located on the surface that does not coincide with an interaction interface rarely impact the function of the protein [103]. To differentiate between surface and buried residues is non-problematic for protein structures with atomic resolution [96]. However, as mentioned in Chapter 2.3.2 in the majority of proteins, there is no experimentally resolved 3D structure and for that scenario, an array of sequence-based RSA prediction methods has been developed [104–107]. This still leaves the niche for cases, where there is limited (coarser than atomic resolution level) structural information available. In order to address these cases, measures are developed, which can be calculated from limited structural information and should correlate with RSA as much as possible. In this project, we developed a measure SphereCon applicable in four different scenarios and compared it to the actual calculation of RSA via probe rolling and two established measures Coordination Number (CN) [108] and Half-Sphere Exposure (HSE) [109].
The four scenarios are:

1. The full atomic information, in order to have a direct comparison to RSA
2. The backbone coordinates with sequence information, which showcases the full potential of SphereCon and represents structures from experimental structure resolution experiments with coarser than atomic resolution.
3. The backbone coordinates without sequence information, typical intermediate structures in threading procedures. This scenario is also the main scenario for CN and HSE.
4. A predicted distance or contact matrix of the protein. This scenario allows SphereCon to expand its applications, with the help of distance and/or contact matrix prediction methods, to cases where only sequence information is avaialble.

4.1.1 *Related Work*

Measures for solvent accessible area are invented for protein structures with limited information. They tend to be constructed in a rather simple fashion in order to limit the number of cases, where they cannot be applied. The presumably first measure is also the simplest, the coordination number (CN) [108]. The CN is based on a sphere centered at the Cα atom of the residue, the measure is applied to (Figure 4.1). The Cα atoms of other residues located inside the sphere are counted. This number is then called the coordination number. The radius of the sphere is variable and each possible radius creates its own measure, but the most commonly used radius is 13Å.
Why this measure anti-correlates with the RSA is obvious: a residue on the surface of a protein should be surrounded by fewer other residues compared to a residue in the core of the protein. But counter-examples can easily be constructed (Figure 4.2).
The advantages of CN are obvious: due to its simplicity, it requires only the

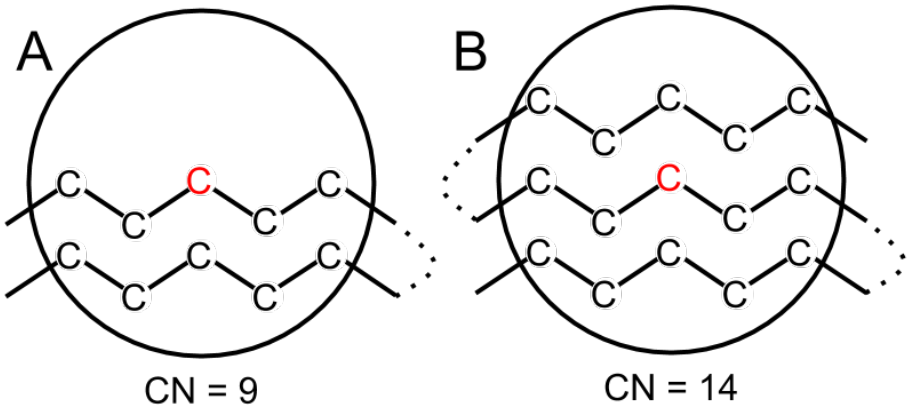


Figure 4.1: *A two-dimensional schematic representation for two example CNs. All residues are represented by their Cα atom depicted as 'C' in the figure. The residue for which the measure is applied to is the red 'C' in the circle center.; A: The upper half of the circle lacks any residues and thus is filled with solvent, this means that the target residue has access to the solvent, also described as located on the protein surface. In this example, this results in a CN equal to* 9*. B: The target residue has no access to the solvent since the upper part of the circle is filled with other residues. This results also in a higher CN of* 14*.*

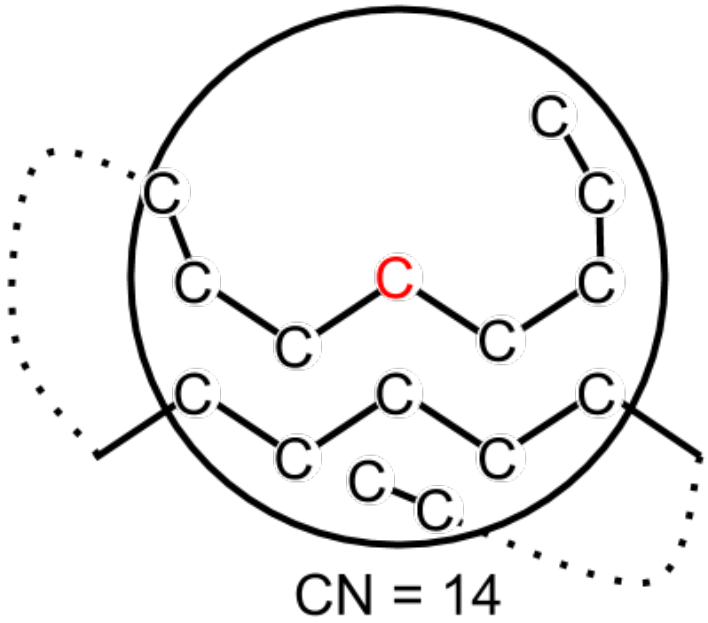


Figure 4.2: *The scheme of this figure is identical to the previous figure. The target residue lies on the protein surface, but due to the changed overall structure, the CN is the same as the CN of the buried residue in the previous figure.*

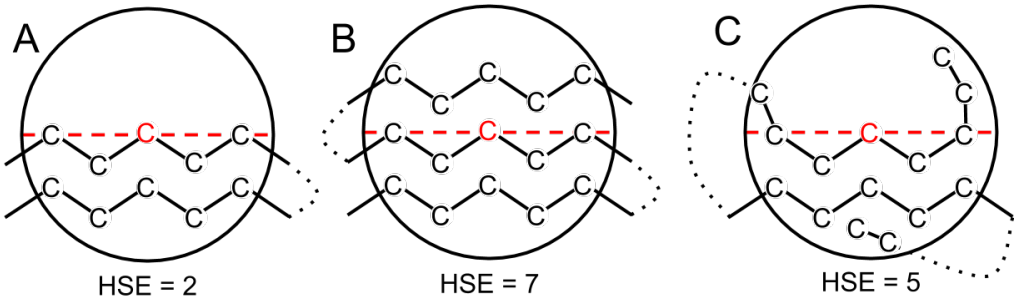


Figure 4.3: *A two-dimensional schematics of HSE examples in the same fashion as the CN examples. The red dotted line depicts the dividing plane. The content of the upper half circles is used for the HSE calculation.; A: The surface residue example results in a low HSE value equal to 2; B: The buried residue example results in a higher HSE value equal to 7; C: The CN counter-example results in an HSE value equal to* 5 *and shows the advantage of HSE over CN.*

Cα-coordinates of the protein. An improved version of CN is HSE [109] (Figure 4.3). The idea behind HSE is that amino acid atoms can be split into backbone and sidechain, giving each amino acid two sides. Geometrically, the two sides are separated by a plane going through the Cα atom and spanned by the normal vector CαCβ. The authors of HSE have shown that the space around the sidechain side is much more important for the solvent accessibility of an amino acid than the space around the backbone side. The plane also divides the CN-sphere around the target residue into two half spheres. The HSE is similar to the CN in that it simply counts the amount of other Cα atoms in the sidechain half sphere.
Still, HSE might have one disadvantage compared to CN: the requirement of Cβ coordinates in addition to Cα coordinates. The authors of HSE came up with a solution by constructing the CαCβ vector based on the coordinates of the Cα atoms of the residues neighboring in sequence. This construction is also applied for all glycines since they do not have Cβ atoms.
Any measure calculation is not possible for proteins, for which there is no resolved structure available. To address those cases, over time a separate field has emerged aiming to predict the RSA of individual residues from the sequence of a protein. It is closely related to the prediction of secondary structure from sequence. Most of such sequence-based RSA prediction methods are end-to-end machine learning methods and hence the state-of-the-art is dominated by neural network-based tools: SPIDER3 [105], NetSurfP-2.0 [106], JPred4 [104] and RaptorX-Property [107] to name just the more recent ones. Breaking away from the end-to-end paradigm SPOT-1D [110] uses predicted contact maps as an intermediate step in order to predict several structural properties on residue level, which can be directly compared to SphereCon's distance matrix mode.

4.2 METHODS

4.2.1 *General Approach*

Just as for CN and HSE, the idea behind SphereCon is to estimate how much of the space surrounding the target residue is occupied by other residues. This is achieved by defining a search space, which is generally based on a sphere, and the calculation of sphere-sphere intersections between so-called intersecting spheress and the search space. The intersecting spheress are either van-der-Waals spheres of spatially close atoms or spheres representing individual residues, depending on the corresponding scenario. For each scenario, the search space is constructed slightly different, since different levels of information allow more or less precise design setups. The less information is available, the more has to be inferred with heuristics and predictions.
The general formula for SphereCon is:

$$\text{SphereCon}(R) = \frac{\text{Volume}(S(R)) - I(S(R))}{\text{Volume}(S(R))}, \tag{4.1}$$

where $S(R)$ is the search space of residue R, and $I(S(R))$ is the sum of the volume of the intersections between all intersecting spheress and the search space.
The maximum value of SphereCon is 1, for residues that are completely exposed. In that case, the sum of interactions is zero. The lower the SphereCon value is, the more buried is the residue and by that it is designed to positively correlate with RSA. SphereCon can have negative values, since the intersecting spheress can intersect with each other and this is not taken into account in the calculation of SphereCon, so the sum of intersection volumes can become larger than the total volume of the search space, resulting in negative SphereCon values.
Inspired by the idea behind HSE, we wanted SphereCon to have a focus on the space in front of the sidechain of the residue. While HSE cuts away half of the sphere, we used a cone, whose apex is identical to the search sphere center and its axis coincides with the line through the search sphere center and the Cα atom of the residue (Figure 4.4). The exact construction of the search sphere and intersecting spheress is different for each of the scenarios and is described in the following.

4.2.2 *Scenario 1 (SC-S1) - Structures with Complete Atom Coordinate Information*

When the coordinates of all atoms of a protein structure are known, one does not have to rely on measures, which are designed for limited structural information, in order to estimate the solvent accessibility of its individual residues. However, in this scenario, SphereCon is able to use the additional information and we can exhaust the measure to its maximal potential. Comparing SphereCon in this scenario directly with RSA, we can investigate whether the concept

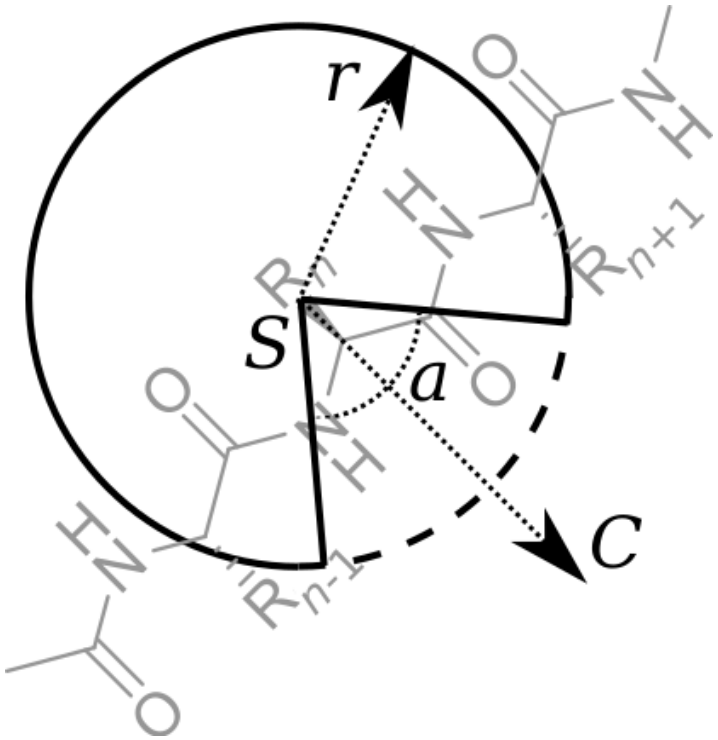


Figure 4.4: *Schematic representation of the design of the search sphere. In grey, a small segment of a protein,* R_n *denotes the residue for which SphereCon value is being computed.* S *is the center of the sphere, which is identical to the apex of the cone,* r *is the search sphere radius,* α *is the apex angle of the cone and* C *is the cone axis.*

behind SphereCon functions as intended.
The search sphere center is placed at the centroid, calculated from the coordinates of all sidechain atoms of the residue. The cone axis is placed at the line through the Cα atom and the sidechain centroid of the residue. For glycine we used the construction method from HSE, which simulates a CαCβ vector by placing it on the plane spanned by the N, Cα and the carboxyl carbon and is orientating it at a 120° angle with respect to the NCα vector facing away from the carboxyl carbon. The search sphace parameters (the radius of the search sphere r and the apex angle of the cut-out cone) can be optimized specifically for each residue type. The intersecting spheres are the van-der-Waals sphere of the atoms of all other residues.

4.2.3 *Scenario 2 (SC-S2) - Only Cα Coordinates*

It is sometimes difficult to resolve protein structures to atomic resolution (for example, for very large protein complexes), leading to experimental results with coarser resolution, from which only the coordinates of the Cα atoms could be deduced. In the absence of the full atomic coordinates, we cannot calculate the sidechain centroid anymore. Based on a gold standard set of high-quality protein structures (described in 4.2.6), we calculated mean centroids for each residue type, which we use in order to predict the location of all centroids of residues in such low-resolution structures. The remaining part of the search sphere design is identical to SC-S1, just with a predicted centroid

instead of a calculated one. The intersecting spheress are spheres centered at the predicted centroids of all other residues with radii specific to the volume of the corresponding residue type (Supplementary Table 9.1).

4.2.4 *Scenario 3 (SC-S3) - Only Cα Coordinates, Unknown Residue Types*

To be in a fair competition with CN and HSE, we remove the information about the individual residue types, leaving a spatial trace of Cα coordinates without any further information. This scenario represents the minimal information, for which CN and HSE are applicable, and represents for example intermediate structures produced as an internal step in some threading algorithms. The possibility to apply a measure in these cases was one major motivation in the creation of CN and HSE.
In this scenario, the centroid prediction, the search sphere parameters, and the intersecting spheress can no longer be residue type-specific. Here we used an overall mean centroid, optimized just one search sphere radius and one apex angle jointly for all residues and used identical sized intersecting spheress with radius 3.23Å, which corresponds to a sphere with mean volume over all amino acids in the gold standard dataset.

4.2.5 *Scenario 4 (SC-S4) - Distance Matrix Mode*

Since the calculation of a sphere-sphere intersection volume requires only the radii of both spheres and the distance of the sphere centers, it is possible to calculate SphereCon using only residue-residue distances. This expands the application possibilities of SphereCon to the estimating the solvent accessibility based on predicted distance matrices. At first glance, the fourth scenario seems to be the scenario with the strictest limitations, but here we include the residue type information again, which are necessary to predict distance matrices.
Since no coordinates are known, no centroids can be predicted and no cutting cones can be applied. Still, we can calculate the sphere-sphere intersection based on the given distances between the target residue and other residues. The search sphere radii can again be optimized specifically for each residue type and the intersecting spheres radii are identical to the ones in SC-S2. Since SC-S4 can be used for predicted distance matrices, which are not completely filled, SphereCon calculates the sparsity of the given matrix and uses different sets of optimized parameters for different sparsity values.

4.2.6 *Gold Standard Dataset, Parameter Optimization, and Performance Evaluation*

For evaluating the performance of all variant of considered measures, we constructed a gold standard set (Supplementary Table 9.2), by choosing one high-quality structure for each SCOP [27] family (only SCOP classes a, b, c, and d), for which we all heavy atoms are resolved. To pick one structure per family we took the one with the best resolution. We calculated the RSA for each residue in the gold standard dataset using DSSP [3]. The Pearson's correlation coefficient of a measure to RSA over the whole datasets is used for evaluation

of a measure.
For different scenarios, the parameters of the search sphere design have to be optimized. Here we used a simple grid search approach, maximizing the correlation value of different parameter combinations, ranging the r in the segment $[4Å, 20Å]$ in $0.25Å$ steps and ranging $\cos(\alpha)$ in the segment $[0, 1]$ in 0.05 steps. The optimal parameters are called r^* and $\cos(\alpha)^*$.
We used a cross-validation setup, using for each validation round all structure belonging to one SCOP class as the test dataset, and the remaining structures as the training dataset for the parameter optimization.
We performed the parameter optimization for SC-S4, using different distance matrices, which are all based on the true distance matrix calculated from the structure. First, we used the complete matrix. Second, we removed all distances between residues, which are neighbors or the neighbors of neighbors in sequence, and third, in addition to the distances removed in the second version, we randomly removed a certain portion of all distances.
As an ultimate test for SC-S4, we took 13 structures (Table 4.4) and their corresponding sequences from the latest round of CASP [111], used the distance matrix prediction webserver from RaptorX [107] to predict pairwise distances from these sequences, calculated SC-S4 on the predicted matrices and compared the results to true RSA values obtained from the structures.

4.3 RESULTS

4.3.1 *Parameter Optimization*

For SC-S1 and SC-S2 we expected larger optimized search sphere radii for larger amino acid types (all optimized parameters are presented in Supplementary Table 9.3). This expectation was met, for example, comparing the smallest amino acid glycine (SC-S1: $r^* = 6.75Å$, SC-S2: $r^* = 6.75Å$) to the largest tryptophan (SC-S1: $r^* = 8.0Å$, SC-S2: $r^* = 9.0Å$). The optimal cutting cone angle defined by $\cos(\alpha)^*$ varies between 0.8 and 1.0 in SC-S1 (0.85 and 1.0 for SC-S2) and thus from search spaces in SphereCon a smaller volume is cut out, compared to the bisection of the search sphere in HSE.
For SC-S3 we optimized only two parameters, and with $r^* = 8.0Å$ and $\cos(\alpha)^* = 0.8$ they happen to be roughly in the same range as the optimal parameters of SC-S1 and SC-S2.
For SC-S4 we determined r^* for each amino acid type and for different sparse distance matrices. Compared to the other scenarios the r^* vary less between the different amino acid types, which might be due to the relatively more limited information in SC-S4 restraining the method from drawing more yield from the additional information of knowing the amino acid type.

4.3.2 *SCOP-based Cross-validation*

Pearson's correlation values of SphereCon to RSA in the SCOP-based cross-validation (Table 4.1) are basically unchanged between the different test-rounds. As expected, the correlation of SphereCon to RSA is decreasing from SC-S1 to SC-S4, since with each scenario we decrease the amount of incorporated

	SCOP class A	SCOP class B	SCOP class C	SCOP class D
SC-S1	0.951 (0.950)	0.949 (0.951)	0.950 (0.950)	0.950 (0.950)
SC-S2	0.921 (0.921)	0.921 (0.921)	0.921 (0.921)	0.919 (0.922)
SC-S3	0.893 (0.892)	0.893 (0.892)	0.890 (0.894)	0.895 (0.892)
SC-S4	0.873 (0.872)	0.874 (0.870)	0.871 (0.877)	0.879 (0.871)

Table 4.1: *Pearson's correlations of SphereCon to RSA, using the parameters optimized in the cross-validation setup (performance of SphereCon evaluated directly on the training set in parenthesis). The distance matrices for SC-S4 are directly calculated from the structures in the gold standard dataset.*

	RSA	CN	HSE	SC-S4	SC-S3	SC-S2	SC-S1
RSA	1.0	−0.770	−0.823	0.878	0.893	0.921	0.950
CN		1.0	0.817	−0.795	−0.866	−0.830	−0.778
HSE			1.0	−0.819	−0.882	−0.860	−0.835
SC-S4				1.0	0.899	0.939	0.903
SC-S3					1.0	0.949	0.907
SC-S2						1.0	0.944
SC-S1							1.0

Table 4.2: *Pearson's correlation between all tested measures on the full gold standard dataset.*

information. Surprisingly, these drops revealed to be smaller than expected and with using only a distance matrix as input, we are able to reach a correlation to RSA greater than 0.87. We also notice that the training error is as good as equals to the test error, meaning that the parameter optimization showed no overtraining effects.

4.3.3 *Comparison of SphereCon to RSA, CN, and HSE*

We calculated SphereCon values in all four scenarios as well as RSA, CN, and HSE values for each residue in the gold standard dataset and calculated the Pearson's correlation between all of these seven measures (Table 4.2).
For the correlations between CN to RSA and HSE to RSA, our results are in agreement with the evaluation performed by the authors of HSE [109]. The correlations of SphereCon to RSA are all greater than of HSE to RSA and, as expected, are increasing as the information in the specific scenarios increases. HSE and SC-S2 are designed to be applied for the same application scenarios and using mostly the same information. The comparison of their performance is most important for the evaluation of SphereCon. Here, SC-S2 (correlation to RSA 0.921) clearly outperforms HSE (correlation to RSA −0.823). Even if we remove the sequence information (since HSE does not use this information), SC-S3 (correlation to RSA 0.893) is still superior.

4.3.4 *Optimization for Sparse Distance Matrices*

For SC-S4 different sets of parameters were optimized based on artificially perturbed distance matrices in order to simulate distance matrices coming from prediction methods. Again, we used the SCOP-based cross-validation (Table

Sparsity	Test set (training set consists of the other three SCOP classes)			
(% of distances removed)	SCOP class A	SCOP class B	SCOP class C	SCOP class D
None	0.873 (0.872)	0.874 (0.870)	0.871 (0.877)	0.879 (0.871)
0%	0.881 (0.878)	0.877 (0.868)	0.877 (0.880)	0.882 (0.878)
10%	0.831 (0.851)	0.831 (0.850)	0.831 (0.841)	0.834 (0.845)
20%	0.821 (0.826)	0.820 (0.817)	0.815 (0.826)	0.827 (0.828)
30%	0.801 (0.813)	0.802 (0.816)	0.792 (0.807)	0.806 (0.808)
40%	0.753 (0.793)	0.758 (0.788)	0.741 (0.797)	0.763 (0.767)
50%	0.768 (0.752)	0.769 (0.777)	0.758 (0.784)	0.776 (0.779)
60%	0.728 (0.755)	0.735 (0.754)	0.718 (0.757)	0.741 (0.730)
70%	0.746 (0.732)	0.747 (0.744)	0.734 (0.733)	0.750 (0.736)
80%	0.725 (0.733)	0.728 (0.725)	0.712 (0.731)	0.732 (0.728)
90%	0.713 (0.719)	0.713 (0.715)	0.703 (0.709)	0.722 (0.712)

Table 4.3: *Pearsons's correlation of SC-S4 and RSA with parameter optimization and evaluation performed on sparse distance matrices. Correlation of SC-S4 and RSA for the training set is given in parentheses. None sparsity: the full distance matrix. 0% sparsity: distances of residues separated by less than three amino acids in sequence are removed. Sparsity > 0%: additionally randomly selected entries from the distance matrix are removed.*

4.3) to demonstrate that the parameter optimization does not introduce any overtraining effects into SC-S4.
In comparison to the cross-validation of the other three scenarios, the obtained correlation values are less stable. This was to be expected due to the fact that we randomly remove distances from the matrix. The correlation values are continuously decreasing the higher the sparsity of the distance matrices get.

4.3.5 *SC-S4 on Predicted Distance Matrices for CASP Targets*

For the CASP targets that we used for a real-life evaluation of SphereCon's SC-S4, the sparsity of the predicted distance matrices fluctuates strongly (Table 4.4). The reason for this is that RaptorX reports only a top-quantile (first $L/5$ top-scoring pairs where L is the length of the input sequence) of all predicted distances. The size of this quantile is linear with respect to the length of the input sequence, the size of a distance matrix grows quadratically, thus for longer sequences, we get less complete matrices.

4.4 DISCUSSION

With SphereCon, we developed a new measure for estimating relative solvent accessible area, which is applicable for protein structures with limited information. Its simple geometrical design is inspired by its predecessors [108, 109]. SphereCon aims to include all available information to improve the performance of the measure. SC-S1 shows a very high correlation to RSA and acts as a proof of concept. Since its simpler design and applicability to single residues, SC-S1 can even be used in scenarios, where the computation time is more important than the precision of the measure. However, we are not aware of such a scenario in real life.

Structure	#Amino acids	Sparsity	Pearson's correlation to RSA
6EK4 chain A	342	92.3%	0.600
6F45 chain A	68	70.5%	0.484
5W9F chain A	72	58.8%	0.460
6CP8 chain A	157	78.8%	0.740
6BTC chain A	84	67.1%	0.771
6CP9 chain A	116	73.2%	0.742
6CP9 chain B	114	75.1%	0.706
6CCI chain A	354	93.2%	0.631
6G57 chain A	97	66.0%	0.404
6GNX chain A	98	72.3%	0.729
6D7Y chain A	89	69.2%	0.620
6Q64 chain A	319	90.6%	0.552
6MSP chain A	80	59.7%	0.719

Table 4.4: *Pearson's correlation of SC-S4 to RSA on* 13 *structures with predicted distance matrices.*

SphereCon is directly comparable to CN and HSE in SC-S2 and SC-S3, where it worked exactly as intended and demonstrate a great increase in performance. Here SphereCon has the potential to improve any method, which relies on CN or HSE as a measure.
SC-S4 addresses a fundamentally different problem of prediction of solvent accessibility based on the sequence alone. The evaluation of artificially perturbed distance matrices proved that in theory, it is possible to use SC-S4 on sparse distance matrices, achieving a comparable performance to CN with full information even after removing half of the matrix along with the distances to residues close in sequence. However, the evaluation of SC-S4 on real predicted distance matrices showed that the performance does not necessarily depend on the completeness of the matrix. After all, the sparsity of the predicted matrix is no measure for the quality of individual predicted distances, and the performance of SC-S4 is directly dependent on the quality of the predicted matrix. Still, the varying performance of SC-S4 for the 13 CASP structures is not very convincing. In any case, the completely different concept of SC-S4 that uses simple geometric calculations suggests that it can be worthwhile and interesting to include it in a possible consensus method.

5 EFFICIENT ANNOTATION OF PROTEIN SEQUENCES WITH STRUCTURAL INFORMATION

This chapter describes the development of StructMAn. In my master's thesis, I developed an automated computational pipeline capable of structurally annotating non-synonymous single nucleotide variants (nsSNVs) and selecting an appropriate structure to be used as a template in a follow-up homology-based protein structure modeling. To assess the consequences of an nsSNVs for the protein function, it is crucial to analyze interactions, in which the protein participates. So, after the homology modeling, further expensive computational methods, like docking experiments or molecular dynamics simulations, are required. Especially for larger proteins, this can become very time-consuming. Stacking more and more complex computational methods on top of each other also accumulates errors. We are convinced that instead of accumulating errors in this way, it is more appropriate to directly analyze more available experimentally resolved structures, transferring the implications gained from structures of homologous proteins. Thus, we moved away from the automated modeling process and focused the pipeline on the combination of information gained from the analysis of multiple structures.

5.1 INTRODUCTION

The rise of next-generation sequencing techniques resulted in huge amounts of available sequence data. Utilizing this flood of data is frequently stated as one of the major motivations behind the development of methods in computational biology. The information one can obtain analyzing protein structures is plentiful (see Chapter 2.3.1), hence connecting the protein sequence world to the protein structure world is very promising. We call such computational methods that map protein sequences to protein structures structural annotation methods [112, 113].
At first glance, structural annotation can be done manually and does not require the development of a specialized method. For example, for any protein in Uniprot [2], all PDB entries corresponding to it are listed in the Uniprot entry. But that way one risks losing a lot of useful structural information, which can be found in structures, whose sequence is not identical to the protein sequence listed in the Uniprot entry. As already mentioned in Section 2.3.2, methods producing experimentally resolved protein structures cannot keep up with the sheer amounts of newly sequenced genes and transcripts. Thus the number of protein sequences without corresponding experimentally resolved structures is increasing. Since the function of a protein and the interactions, it participates in, are conserved among homologs even with low sequence identity [114], one can transfer implications from a structural analysis of homologous proteins. This widely used technique [115, 116] increases the number of protein sequences for

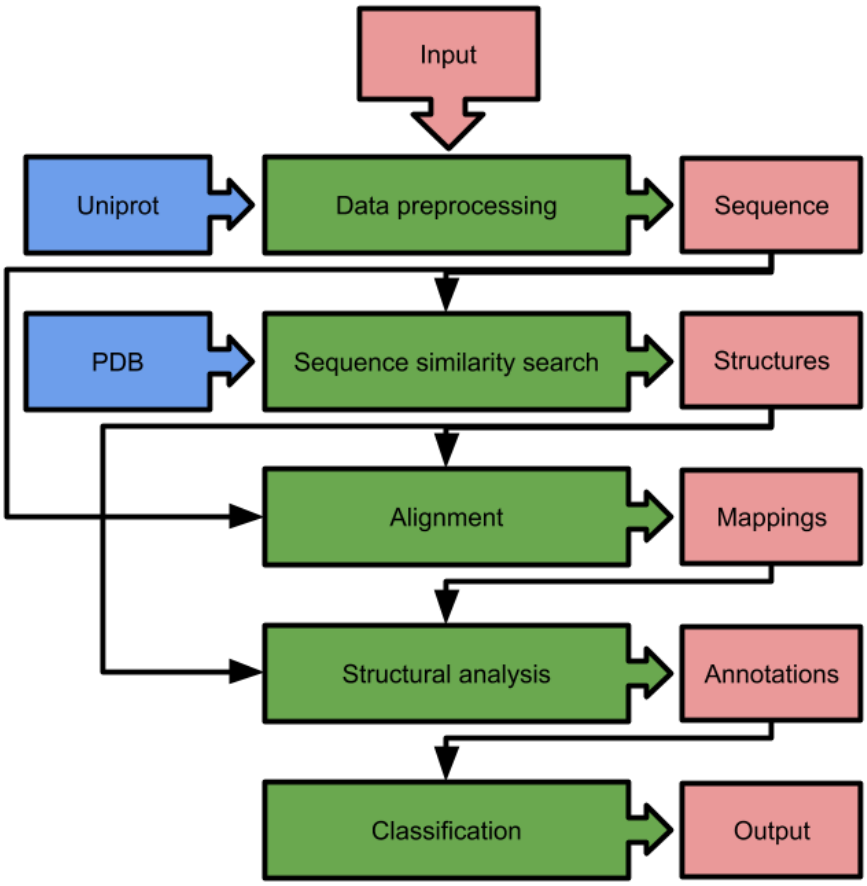


Figure 5.1: *Schematic representation of the algorithmic pipeline behind StructMAn. Blue rectangles are external data sources, green rectangles are the intermediate steps of StructMAn and red rectangles are the types of data of different inputs and outputs.*

which we can find protein structures that can be used for performing structural analyses.

The goal of this project is the creation of a sequence-to-structure annotation method, which is as computationally efficient as possible in order to process all possible input sizes. However, this efficiency must not result from compromising on the annotation quality. The method should be able to process protein sequences at the resolution level of each individual position. It should collect all available structural information, process this information and supply further downstream methods with it in a concise form.

With the implementation of StructMAn, we were able to fulfill all these requirements. This fully automated algorithmic pipeline takes individual amino acids from a given protein sequence and maps them directly to multiple experimentally resolved protein structures. All residues are analyzed individually, the results are combined and ultimately lead to a structural classification at a residue level.

Briefly, the algorithmic pipeline has five major steps (Figure 5.1). In the data preprocessing step, a list of given protein sequence IDs are translated to Uniprot accession numbers (`https://www.uniprot.org/help/accession_numbers`), and

their corresponding sequences are acquired. Since being protein isoform-specific, Uniprot accessions function as the primary type of protein IDs in StructMAn. In the sequence similarity search step, for each protein sequence the PDB [38] is searched for corresponding experimentally resolved protein structures. Next, the sequences of the found structures are aligned to their corresponding target protein sequence. This alignment then defines the mappings of individual amino acid positions in the target sequence to specific residues in the protein structures. The mapped residues are structurally analyzed in the fourth step, producing residue-specific annotations. The final step is to combine the annotations from all structures associated with a target position in order to produce position-specific structural annotations, which are the foundation for the structural classification of the position.

5.1.1 *Related Work*

The history of structural annotations methods is tightly connected to the methods for mapping of genetic variants into protein three-dimensional (3D) structure. Thus most of the related methods (Table 5.1) enable querying for specific mutations. In addition to the structural annotation, these methods also usually provide either a structural analysis or connections to database entries corresponding to the given variant.
When developing a structural annotation method, one has to balance the comprehensiveness of the analysis of individual structures and the total number of variants or proteins the method can process. Construction of a pre-computed database allows for more complex analysis, but one loses the ability to process arbitrary variants provided by the user. Based on that, we can categorize structural annotation methods into high-focus, high-throughput and database methods, but rarely a method falls into just one category. High-focus methods use a comprehensive analysis of the annotated structures, which usually makes them computationally more expensive and as a consequence, they are restricted in the number of variants, which can be processed. High-throughput methods try to maximize the amount of data, which can be processed. This is achieved by using computationally less expensive methods, automatization techniques and the inclusion of structures from evolutionarily related proteins. Database methods are characterized by the use of pre-computed data. Many high-throughput methods are supported by integrated databases to yield the advantages of a database method without the limitations typic for pure databases.
Databases of annotated mutations usually promise the most comprehensive results for individual mutations. Also, runtimes are no issue here, and hence databases seem to be the most desirable type of structural annotation methods. The obvious drawback of using databases is that they are limited to already annotated mutations, which makes the analysis of novel mutations impossible. The solution is to combine structural annotation pipelines with an integrated database.
To the best of my knowledge, StructMAn is the first high-throughput structural annotation pipeline efficient enough to annotate all human proteins. However, many methods can map genetic variants to protein structures on a smaller scale.

Method	Category	Whole sequence annotation	Custom nsSNVs possible	Species	Including structures of homologs	Considering protein isoforms
Mechismo [115]	High-throughput	Yes	Yes	8	Yes	No
dSysMap [117]	High-throughput	Yes	Yes	Human	No	No
VarQ [118]	High-focus	No	Yes	Any	No	Yes
mutfunc [116]	Database	No	No	3	Yes	No
MuPit [119]	High-throughput	No	Yes	Human	No	No
MutDB [112]	Database	Yes	No	Human	No	Yes
SAAPdap [120]	High-focus	No	Yes	Any	No	Yes
LS-SNP/PDB [121]	Database	Yes	No	Human	No	?
SNPs3D [122]	Database	Yes	No	Human	No	No
G23D [123]	High-focus	Yes	Yes	Any	Yes	Yes
PROSAT+ [113]	High-focus	Yes	Yes	Any	Yes	No*
MSV3d [124]	Database	Yes	No	Human	Yes	?
Cancer3D [125]	Database	Yes	No	Human	Yes	No
SNP2Structure [126]	High-focus	Yes	Yes	Human	No	No
AMASS [127]	Database	Yes	No	Any	Yes	No
Aquaria [128]	Database	Yes	No	Any	Yes	No

Table 5.1: *Characteristics of different structural annotation methods; *possible through custom sequence input.*

Most of them also perform a structural analysis of the mapped residues, for example, the method MuPIT [119], which is limited to human variations. The pipeline returns a list of structures and leaves the choice of structure to be used for structural analysis to the user. The input of MuPIT is not limited to single positions but accepts lists of positions of arbitrary length and the processing time is with a few seconds quite fast. These aspects suggest that MuPIT can be categorized as a high-throughput method. The analysis performed for the mapped residue is based on annotated databases, which limits the structural analysis of MuPIT.
dSysMap [117] is capable of processing single positions as well as whole proteins and is considered as a high-throughput method. dSysMap concentrates on Protein-Protein Interactions (PPI), creating not only a PPI network for the given proteins but also placing the given individual mutations into the network, based on their spatial location in the protein complex. Individual mutations are structurally classified and information from multiple external databases is gathered: Pfam [129], 3did [130], BIND [131], BioGRID [132], DIP [133], HPRD [134], InnateDB [135] and IntAct [80]. The weaknesses of dSysMap include the lack of analysis of interactions with small molecules, and the fact that alternative protein isoforms are not considered by this method.
Another high-throughput method is Mechismo [115], whose focus is to predict if a residue participates in an interaction. It takes protein sequences from Uniprot entries as input without considering individual isoforms separately. For each of these Uniprot entries, the experimentally resolved structures in the PDB are listed. In order to find experimentally resolved structures of pro-

teins evolutionarily related to the given protein, a sequence similarity search against the other proteins in Uniprot is conducted. The listed structures of the entries resulting from this search are also considered for the structural analysis. Interactions are identified by distance: molecules closer than 5Å are considered to be in interaction. In order to increase the prediction of interactions, Mechismo also includes the experimental information from the interaction databases such as BIND [131], BioGRID [132], IntAct [80] and MINT [136]. For the whole input sequence, Pfam [129] domains are identified and disordered regions are predicted by IUPred [137]. The predicted interactions are divided by the type of interaction partner into protein-protein, protein-chemical, and protein-DNA/RNA interactions. The protein-chemical interactions also are subdivided into organic, inorganic and organometallic.
The Mechismo webserver has low processing times, since it always produces outputs for all variants with precomputed annotations for the given protein, even if individual variants are requested. The reason for its speed is an immense amount of precomputations. Each of the similarity search results is stored, and for all residues in all structures it is precomputed in what kind of interaction they are involved in. This limits the space of allowed inputs to only eight species (*H.sapiens, M.musculus, S.cerevisiae, C.elegans, D.melanogaster, E.coli, B.subtilis* and *M.pneumoniae*) and gives Mechismo a database-like character. But since it stores the intermediate steps and not just the end results, it can be easily expanded by adding precomputations for other sequences and about 60,000 sequences of 8 organisms are currently available (January 2020).
A recently developed method more focused on individual positions, especially nsSNPs, is VarQ [118]. The goal of VarQ is to assess the clinical relevance of a given nsSNP. For that purpose, it combines databases comprising data on clinical effects of mutations (dbSNP [45], BioMuta [138], humsavar [139], and ClinVar [63]) with a custom structural annotation pipeline, for which only corresponding experimentally resolved structures are considered. The structural analysis is performed on a single structure, but when different low molecular weight ligands are bound in different structures, all such structures are analyzed. For the structural analysis, VarQ selects the structure that covers the largest part of the given protein and uses the resolution of a structure as a tie-breaker, when multiple structures have the same coverage. The performed structural analysis is the most comprehensive in the field. The participation of the wildtype residue in a protein-protein interaction is assessed with 3did [130]. The involvement of the residue in an active site is computed with fpocket [140]. The change in the protein structure stability introduced by the amino acid substitution is calculated with FoldX [141]. The relative solvent accessibility (RSA) of the residue is calculated in order to determine if the residue lies on the surface of a protein or is buried in the protein core. Tango [142] is used to estimate the tendency of the mutation to cause aggregation. Conservation of the wildtype and mutant amino acids is assessed using the allele frequency in the alignment of the corresponding Pfam family. The results of all performed analyses are reported individually. Since VarQ uses many computationally expensive methods, it is very slow and can only be used in case studies of preselected mutations. Further, the fact that 3D structures of homologs are not

considered leads to a strong reduction of cases where it is applicable.
These examples show that there is no perfect structural annotation tool. All of the methods have their strengths and weaknesses and this also applies to our method. We have developed a high-throughput method applicable to input sizes that were not approachable in the field so far. Our method also produces comprehensive structural analyses more extensive than most high-focus methods.

5.2 METHODS

5.2.1 *Data Preprocessing*

StructMAn supports two types of file formats for input files, a simple tab-separated text file format, which we called the simple mutation list format (smlf), and the variant call format (vcf) (https://en.wikipedia.org/wiki/Variant_Call_Format). Files in the smlf format include a list of protein IDs. Uniprot IDs (https://www.uniprot.org/help/entry_name), Uniprot accession IDs (https://www.uniprot.org/help/accession_numbers) and Refseq sequence IDs [143] are supported. All types of protein IDs are mapped to Uniprot Accession numbers, which functions internally as the main type of protein IDs. For all given proteins, the corresponding amino acid sequence is retrieved. Instead of a protein ID, one can also use a PDB structure ID (https://www.rcsb.org/pages/help/advancedsearch/pdbIDs) together with a chain identifier. In that case, the amino acid sequence of the given chain is parsed from the corresponding PDB file. In the smlf file format, one can also specify individual positions, individual nsSNVs, and add position-specific tags. Such tags can later be used to create tag-specific statistics.
Files in the vcf format contain a list of genetic variants specified by their genomic coordinates specific for a reference genome and thus lacking protein IDs. For that reason, it is necessary to provide the UCSC ID [144] for the corresponding reference genome. For files in the vcf format, corresponding to a reference genome that is not in the UCSC genome database, one has to provide the corresponding reference genome. For input files in the vcf format, StructMAN uses ANNOVAR [145] to map the variants to the protein IDs, which are used in the corresponding reference genome.

5.2.2 *Structure Search and Quality Assessment*

The structure search step has the goal to find as many experimentally resolved structures as possible that are either a corresponding structure (structure of the corresponding protein) or a structure of a homologous protein. The first step is a sequence similarity search against a sequence database containing all experimentally resolved structures from the PDB [38]. Here instead of taking the content of the SEQRES records, we scanned through the ATOM records in order to represent the structures as precise as possible. Since PDB entries usually contain several structure files, we retained the first biological assembly file and if not available the first asymmetric unit file. The latter case mainly concerns NMR structures, for which we took the structure indicated as the first

model in the file.
The sequence similarity search performed on the resulting sequence database is set to produce as many hits as possible. The version of StructMAn we published in 2016 used BLAST [83] with an e-value cutoff of 0.1. The current implementation uses MMseqs2 [86] removing the default limit for the maximum number of produced hits and limiting minimum hit length to 50. Input proteins with sequences shorter than 100 amino acids are processed separately without the limitation on the length of the hits. Thus, a configuration of the search method with high sensitivity was chosen, which means a lower runtime efficiency. In the greater scheme of constructing a high-performance algorithmic pipeline, this seems counter-intuitive, but practice has shown that the structure search is not the bottleneck of the pipeline by far. When multiple chains of the same structure are found to be similar, all hits are retained and this information is used to infer the oligomeric composition of the complex.
Then we filter for high-quality structures. In the first round of structure filtering, we reject structures, for which resolution is too poor (> 4.5Å). The remaining hits go into the alignment step, which is explained in the next section. After the alignment more precise values for sequence identity and alignment length are available. These are used for the second round of filtering, where all hits below 35% sequence identity and, for proteins with more than 100 amino acids, all hits with an alignment length less than 50 are filtered out. All hits are scored using a structure quality score:

$$Q(S, C, R) = M1(S) + 0.5 \cdot C + 0.25 \cdot M2(R) \tag{5.1}$$

$$M1(x) = \frac{1}{1 + e^{10 \cdot (0.4 - x)}} \tag{5.2}$$

$$M2(x) = \frac{1}{1 + e^{1.5 \cdot x - 4}} \tag{5.3}$$

The quality score combines three measures. Sequence identity (S) and coverage (C) (the number of positions not matched with gaps divided by the length of the target sequence) are based on the pairwise alignment between the target protein sequence and the sequence of the mapped protein structure. The structure's experimental resolution (R) is taken from the header corresponding PDB file reporting of the structure resolution experiment. S and C are a measure for the quality of the match between a target protein and protein structure, R is a measure for the overall quality of the structure. Each measure is scaled to $[0; 1]$ (see equations 5.2 and 5.3), where 1 corresponds to structures of the highest quality. They are then combined in a weighted sum, S is the most important measure, C is weighted half as strong as S, and R is weighted half as strong as C (see equation 5.1).
As seen in the plot (Figure 5.2), M1(5.2) is mapped in a way that the function grows for values between 0.35 and 0.7 linearly, while the growth for values

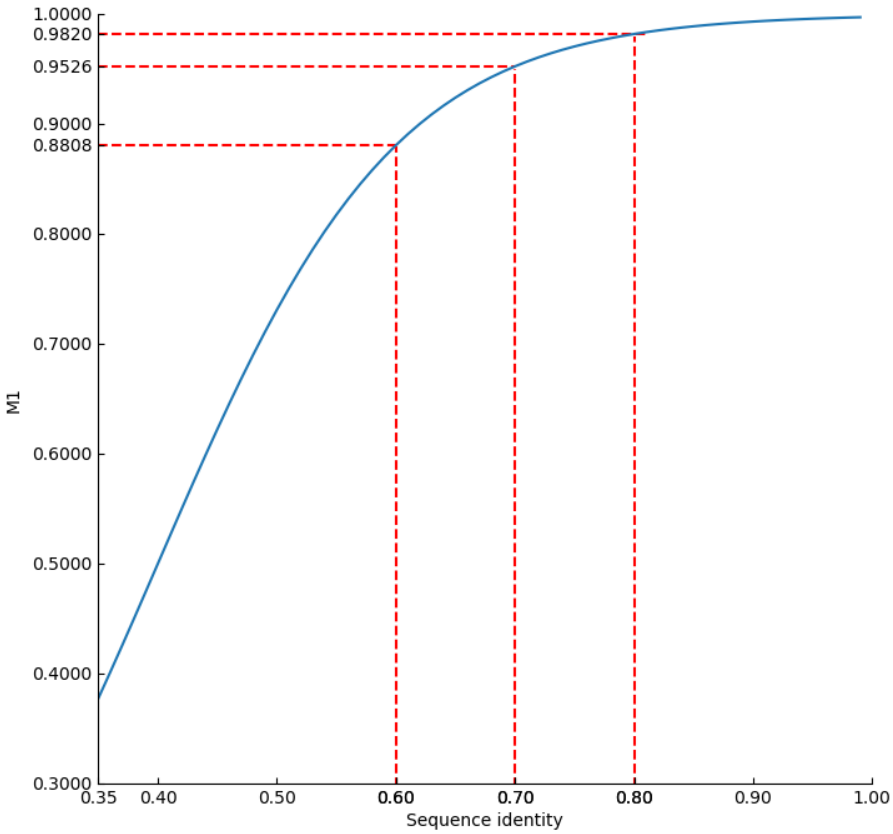


Figure 5.2: *The mapping function* M1 *(5.2) plotted for a sequence identity interval of* 0.35 *to* 1.0

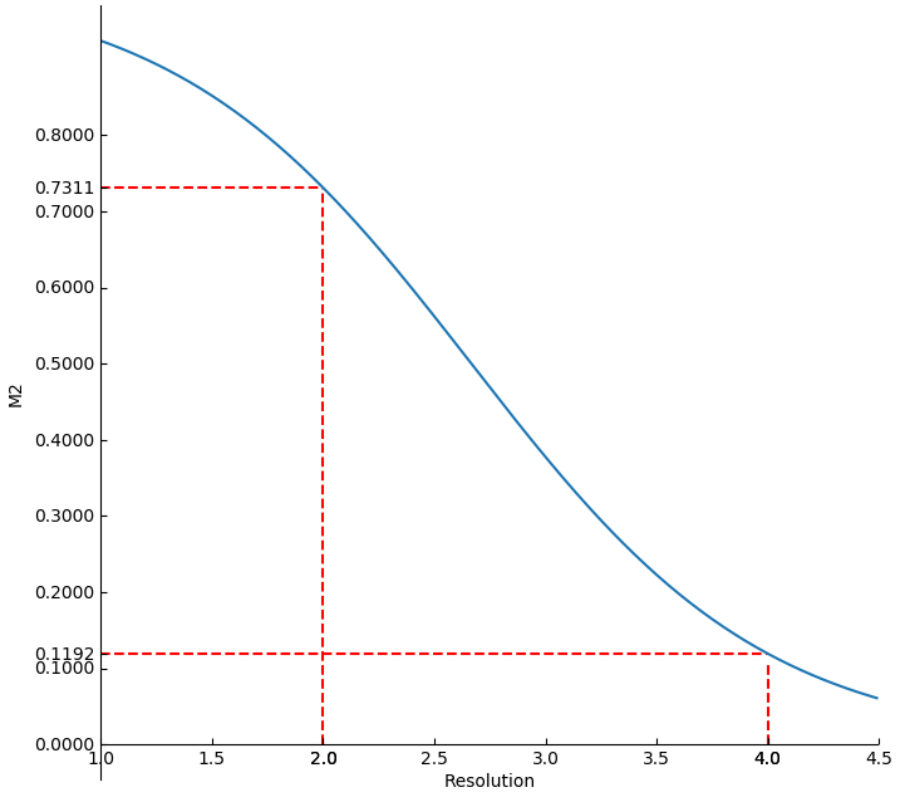


Figure 5.3: *The mapping function* M2 *plotted for a resolution interval of* 1Å *to* 4.5Å

above 0.7 is limited. Translated in the biological terms, this means that the higher the sequence identity of the sequence-to-structure mapping, the more valuable is the information gained from the mapping, but the difference between a structure with 70% sequence identity and a structure with 40% identity is higher than the difference between a structure with 100% sequence identity and a structure with 70% sequence identity. We did not use a mapping function for the coverage since it is already a value in $[0;1]$ and the difference of coverages between two mappings has no biological background, but most often results from experimental limitations. The resolution of the structure is mapped according to the function 5.3, which is plotted in Figure 5.3. The idea of the mapping function is to have high values of the function for resolution below 2Å, for resolution between 2Å and 3.5Å it should drop approximately linearly approaching 0, and for resolution above 4Å the values of the function should be very low.

The quality score is used at later stages in the pipeline when it comes to the combination of different annotations. In theory, the quality scores can range from 0 to 1.75, we could have divided it by 1.75 in order to map it into the range between 0 and 1, but this would only have cosmetic effects. Since the score is of only internal use for weighting the information gained from different structures, we refrained from doing so.

5.2.3 *Prediction of Disordered Regions*

For disordered regions of proteins, a structural analysis does make much sense, since they by definition do not assume a stable 3D structure. But knowing where they are is still a piece of useful information and that can be used for assessing the effect of mutations in them, e.g. with respect to location of possible functional linear motifs that tend to reside in these regions [29]. For all input protein sequences, we predicted the ordered and disordered regions by first searching in the MobiDB3.0 [146] database, which combines the experimental knowledge about the intrinsic disorder with methods for prediction of disordered regions. For proteins not covered in MobiDB3.0, we used IUPred2A [137] to predict the disordered regions.

5.2.4 *Sequence Alignment*

In order to map individual amino acids to individual residues in a structure, we compute the alignments between the target sequence and all sequences of the corresponding proteins in the structures returned from the sequence similarity step. As the alignment algorithm, we choose the Needleman-Wunsch pairwise global alignment method [147] and use the implementation from biopython [148]. As the substitution matrix, we use the BLOSUM62 [88]. The gap open penalty is 10 and the gap extension penalty is 0.5. One challenge that we often face are proteins that are only partially resolved in the structure, which results in a low alignment coverage, while the alignment has a decent sequence identity. In order to address these cases, we chose to not penalize terminal gaps.
The sequence identity and sequence coverage of the resulting alignment are important measures for the quality estimation of the corresponding sequence-to-structure mapping. The alignment itself defines the mapping of individual positions of the target sequence to individual residues in the mapped structure. We call such a mapping a position-specific structural annotation.

5.2.5 *Solvent Accessible Area*

Residues lying on the protein surface are less frequently involved in protein function than residues lying in the core [98], hence calculation of solvent accessible area is an important part of the structural interpretation of mutations. We used DSSP [3] in its default settings and take the total surface area (SA) of each residue. Further, from DSSP we also extract the secondary structure assignment as well as the calculated torsion angles. We divide the SA values by the amino acid type-specific maximum values according to Rost and Sander [97] in order to obtain the relative surface area (RSA). We also calculate the SphereCon values, which are described in detail in Chapter 4. For most cases, when DSSP is not applicable, due to limited structural information, SphereCon measures can still be calculated and are used instead of RSA values.

5.2.6 *Distance Calculations*

For each pair of residues, incuding those from other chains than the one that corresponds to the target sequence, distances are calculated at an atomic

level, i.e. the distance between two residues A and B is the shortest distance between any atom from residue A and any atom of residue B. Analogously, distances between each residue and each co-resolved small molecule (stored in the heteroatoms records in the PDB files) are calculated. A small set of small molecules corresponding to typical buffer components (Supplementary List 9.1.1) were excluded from the analysis. The minimal distances between the residue of interest (if the list of residues of interest is specified in the input, otherwise for each residue) and all potential interaction partners co-resolved in the structure are calculated and all distances below 5Å are stored.
To do this efficiently we compute a so-called fuzzy distance matrix. The fuzzy distance matrix of a structure contains distances for pairs of residues from all chains from a structure. It guarantees correct distances for residue pairs close in space, i.e. contacting residues from the same chain and in interaction interfaces. For residue pairs that are not close in space the distances may be approximated. The calculation of a residue-residue distance matrix completely filled with correct distance values takes time quadratic with respect to the total amount of atoms in a structure, but since most of them are above 5Å, only the correct values for a small subsection of pairs is required. For the calculation of the fuzzy distance matrix, we first calculate minimal Euclidean axis-aligned bounding boxes around each chain, or simpler explained, the minimal and maximal x,y and z values for each chain. Then we construct the smallest possible sphere around the boxes. Now if two such spheres have a distance above 5Å, no residues in their two corresponding chains can have a distance below 5Å and they are not considered for the following distance calculation steps. For all remaining residues, the distance between their first listed atom is calculated, this distance is recorded as approximate distance in the fuzzy distance matrix. For each residue, we calculate the precise (on atomic level) distance only to the twenty residues with lowest approximate distance.
For small structures, a fuzzy distance matrix is basically identical to a precise distance matrix and there are also no improvements in terms of running time. But, for large structures, for many residue pairs distances are approximated or not calculated at all, thus the fuzzy distance matrix becomes sparse, effectively reducing the complexity of the algorithm to be linear with respect to the number of residues. Since the calculation of a precise distance matrix takes quadratic time in relation to the number of atoms in the structure, they can create menacing bottlenecks for huge structures. These bottlenecks were eliminated by the introduction of the fuzzy distance matrix.

5.2.7 *Residue Interaction Networks*

Structures can be represented as residue interaction networks (RINs). RINs are graph structures, where the nodes correspond to amino acids and the edges denote interactions between amino acids. The distance matrices described in the previous section already provide a way of constructing a RIN by taking the distance matrix, removing all distances below a certain threshold and using the resulting matrix as matrix representation of a graph.
In this section, we describe the usage of more precise RINs, generated by RINer-

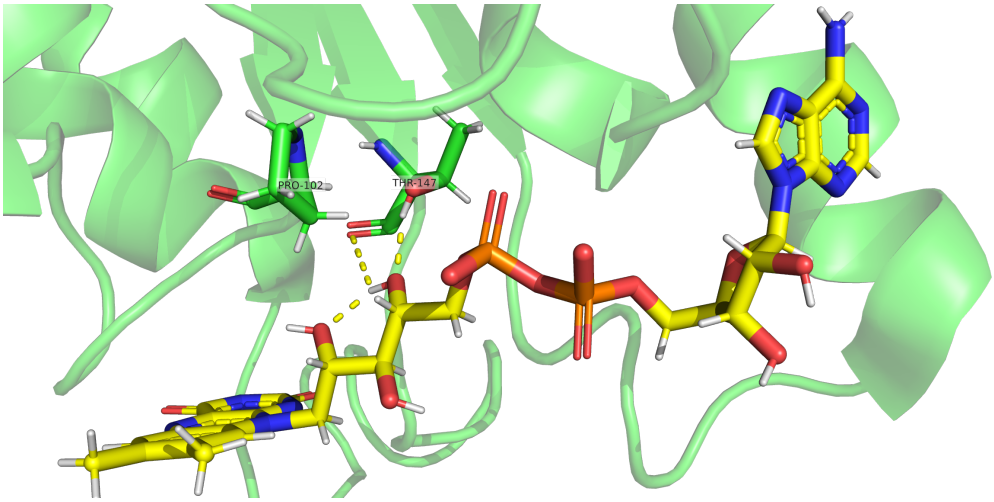


Figure 5.4: *Threonine* 147 *(*T147*) and proline* 102 *(*P102*) of the quinone reductase* 2 *(green cartoon) and flavin-adenine dinucleotide (FAD) (yellow sticks) taken from PDB structure 4FGL; H-bonds shown in yellow dashed lines. The shortest distance between* T147 *and FAD:* 2.68Å*; the shortest distance between* P102 *and FAD:* 2.89Å*; probe score of the interface between* T147 *and FAD:* 2.38*; probe score of the interface between* P102 *and FAD:* 0.04

ator [149]. Instead of calculating distances, RINerator uses Probe [4], which rolls probes spheres across the residue van der Waals surfaces, the resulting clashes of the rolling sphere with other residues surfaces are registered, and a so-called probe score is calculated for each residue-residue interaction. These interactions are further classified into van-der-Waals interactions and hydrogen bonds. In addition, the distinction between sidechain-sidechain, sidechain-backbone, and backbone-backbone interactions is made. Here we had to extend the functionality of RINerator to include directed edges for sidechain-backbone interactions. For our purposes, we store for each residue the node degree and the sum of probe scores for all its edges and distinguish between edges to different types of interaction partners.

This type of analysis is computationally expensive, which is in apparent contradiction to the stated efficiency of the pipeline. For that reason, we constructed a RIN database containing all RINs for all structures in the PDB. The downside of this approach is that external installations cannot use RIN-based analyses without constructing the RIN database locally or downloading a copy of it.

An example of the advantage of detecting an interaction with probe scores rather than computing Euclidean distances is shown in Figure 5.4. The two residues have an almost equal distance to the ligand molecule, however, there is a large difference in their corresponding probe scores, which represent the biological relevance much more faithfully. In fact, the sidechain of the threonine forms two H-bonds with the ligand molecule and the proline just happens to be in near vicinity not directly interacting with the ligand other than via weak van-der-Waals interactions.

5.2.8 *Structural Classification*

Since single positions in a given protein sequence can be mapped to individual residues in up to thousands of 3D structures, the produced results are not interpretable by humans anymore. After calculating all these per-residue values for the different mapped structures, the next step is to combine these into a single structural classification for each residue.
For each type of interaction partner, the minimal distance values from each annotated residue are taken in order to calculate so-called weighted distance values (equation 5.4). This is done for each type of the interacting molecule (proteins, DNA, RNA, metals, non-metal ions and low molecular weight ligands) separately. For this calculation, only distance values lower than 8Å are considered. Each distance is multiplied by the quality score of the corresponding protein structure mapping, the products are summed and finally are divided by the sum of quality scores.

$$W(D) = \frac{\sum q_i \cdot d_i}{\sum q_i}, (d_i, q_i) \in D, \tag{5.4}$$

where D is the set of distances (d_i) and quality scores (q_i, see equation 5.1) for all mapped structures of a target position.
Note that the weighted distance can be undefined for cases, where no distance is below 8Å. For these cases and the cases, when the weighted distance is between 5Å and 8Å, the position is considered to be not in contact with the particular type of interaction partner. If the weighted distance for a particular type of interaction partner is below 5Å, the position is considered to be in an interaction of the particular type, for example, protein-protein interaction. A position can be considered to be in interaction with multiple types of interaction partners. When only one type of interaction is defined, this interaction then defines the classification. When multiple interaction types are registered, they are combined as double, triple, quadruple (and so on) interactions. In most cases, these multiple interactions result from different structures, for example, a position is mapped to residue A in structure X and residue B in structure Y. For residue A a protein-protein interaction is detected and for residue B a ligand interaction is detected. Then the resulting classification of the given position is a double interaction, protein and ligand, although, for no single structure, these interactions are observed simultaneously. In order to distinguish classification with multiple types of interactions, we collect the classifications for all individual structures and provide the list together with the combined classification (for each detected type of interaction we provide one structure ID, which corresponds to the structure with the highest quality score, for which this type of interaction is observed).
For cases, when for a position no interactions are detected, the classification scheme shifts its focus to the question if the mapped residues lie on the surface of the protein or in the core of the protein. Instead of the interaction type classifications, we assign a location type classification here. We decided not to calculate weighted RSA values in the same fashion as the weighted distance values. Structures with a low sequence coverage of a protein can produce very

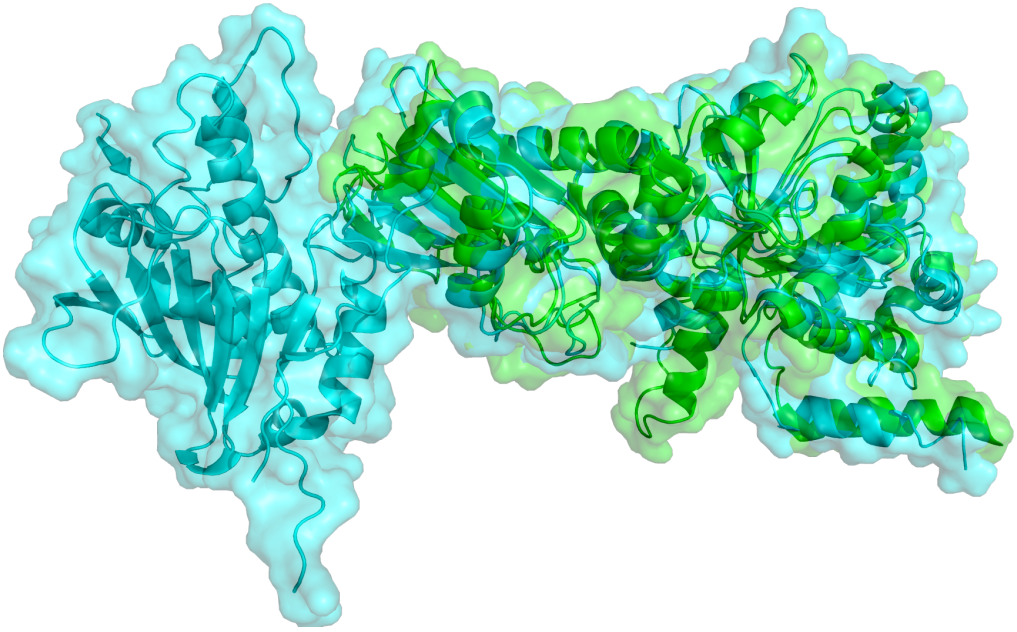


Figure 5.5: *ATP-dependent molecular chaperone HSP82 resolved in two different PDB structures; green: PDB ID 2CGE, chain A; cyan: PDB ID 2CG9, chain B.*

misleading RSA values as shown in Figure 5.5. Here the green structure misses a large part of the protein, which can be seen in the cyan structure that extends the green structure. Thus, some residues calculated to be located on the surface are in reality located in the core. Vice versa this is not the case. Hence, we can conclude that sequence coverage is important, and individual annotated residues located in the core region of a structure are more important than individual annotated residues located on the protein surface.
In an early version of StructMAn, we tried to solve this problem, assigning a position to core, when at least one annotated residue was located in the core. The drawback was that there were a lot of cases, where the structures of distantly related proteins got too much influence on the classification. Now, we determine the combined location type classification (surface or core) by a weighted majority vote. First for each annotated residue the individual location is determined by using the established RSA threshold of 0.16 [97] (surface if RSA $>$ 0.16, core for RSA $\leqslant$ 0.16). In the calculation of the location type classification each annotated residue votes for its location with a weight:

$$WL(D) = \sum_{(q_i,c_i)\in C} 2 \cdot q_i \cdot c_i^5 - \sum_{(q_i,c_i)\in S} q_i \cdot c_i^5, \tag{5.5}$$

where WL is the weighted location, C is the set of coverages (c_i) and quality scores (q_i, see equation 5.1) for all mapped residues buried in the protein core, S is the set of coverages (c_i) and quality scores (q_i, see equation 5.1) for all mapped residues located on the surface of the protein, and $D = S \cup C$.
Positive WL values lead to core classification and negative WL values to a surface classification. At first glance, the constants for the calculation of this weighted majority vote seem arbitrary. They were selected heuristically by

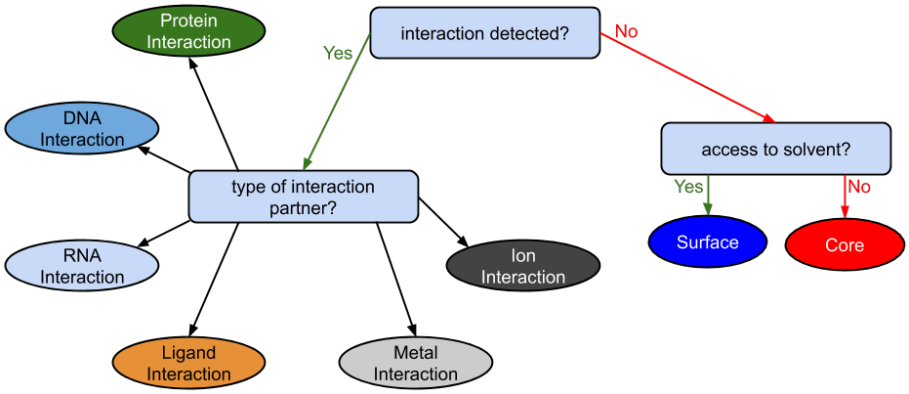


Figure 5.6: *Decision tree for the simple classification scheme.*

searching for examples, where our initial location type classification scheme produced the wrong classifications due to previously explained drawbacks. We choose the parameters of the weighted majority vote such that all the chosen examples are correctly classified while preserving the classification for all cases, where the old scheme was already correct. The resulting weighted majority for the combined location is then taken as a classification for cases when no interactions are detected.

Due to the numerous combinations of interaction types, we also created a simplified classification scheme, where all multiple interactions are simplified by taking only one interaction type based on the following priority order: metal (Supplementary List 9.1.2), ligand (all ligands extracted from the HETATM records except metals and ions), DNA, RNA, protein, ion (Supplementary List 9.1.3).

Thus, for the simple classification, there are eight different classes: the six interaction types and the two location types (Figure 5.6). For both schemes, we also implemented the classification to be based on the RIN-based probe scores instead of the distance calculations.

5.2.9 *Database and Lite Mode*

Efficiency is a matter of the scale of the given input. There are different challenges in terms of runtime efficiency when it comes to the annotation of a single position in a single protein compared to the annotation of all positions for a large set of proteins. For example, when annotating a single position, only single mapped residues are analyzed. It would not make sense to calculate the distance matrix for the whole structure. On the other extreme, when annotating large datasets, it is important not to repeat calculations, for example, when two different positions from different proteins are mapped to the same residue in a structure, the structural analysis should not be repeated. For that purpose, we implemented a relational database in the background of the pipeline. The filling of the database and the checks if results should be calculated or retrieved from the database are producing an overhead in the beginning, but at some point, the cases, when calculations are not necessary and the data can be retrieved from the database, amortize for that. Another advantage of maintaining a database

is the possibility to store of the results from many intermediate steps in the pipeline, which can be used for the development of new downstream analyses without reprocessing the original data.
The default mode of the pipeline is developed in a way to maximize the usage of the database. For example, all residues of all mapped structures are analyzed and the results are stored, independent of whether a particular residue got mapped to an input position. For external installations of the pipeline, the requirements can differ. When no precomputed results are available in the database, the annotation of small inputs can result in a huge overhead, which will never be useful for a particular user. For this use case, we developed the lite mode. It forgoes the usage of the database, saving the overhead coming with it. Thus, the lite mode is much faster for smaller inputs, which are not supported by a well-filled background database, for example, nsSNVs in novel proteins.

5.2.10 *Implementation*

Currently, there are three different implementations of StructMAn. The first one comes in the form of the online webserver and was published in the webserver issue of Nucleic Acid Research in 2016 [150]. The frontend of the webserver was written in PHP, and the backend was written in python 2.5. For the communication between backend and frontend, we used the python flask library. The database runs on in MySQL 5.5.60 for all three implementations. The webserver backend used the python MySQLdb (`http://mysql-python.sourceforge.net/MySQLdb.html`) library for database communication. Since its initial implementation, the webserver never had a major update and lacks many functionalities, described in this thesis.
The in-house version of StructMAn is a locally installed command-line tool, which contains all the described features and is written in python 3.7. We replaced the database communication with the python pymysql library (`https://github.com/PyMySQL/PyMySQL`). The third implementation is a Docker (`https://www.docker.com/`) container (manuscript in preparation). The container will enable a local installation of StructMAn on any system and has the same source files as the in-house version. It includes the lite version and the full version supporting the database that is automatically created and maintained without any required actions from the user.

5.3 RESULTS

5.3.1 *Annotation of the Human Proteome*

The largest dataset we annotated so far was the entire human proteome, which includes the amino acid sequences of all human proteins including separate sequences for different isoforms. The dataset was downloaded from `https://www.uniprot.org/uniprot/?query=*&fil=organism%3A%22Homo+sapiens+%28Human%29+%5B9606%5D%22+AND+reviewed%3Ayes`, whereby the option for canonical+isoforms sequences was chosen. The dataset contains in total 37,702,205 individual positions in 95,117 different isoforms. The first time we performed the complete

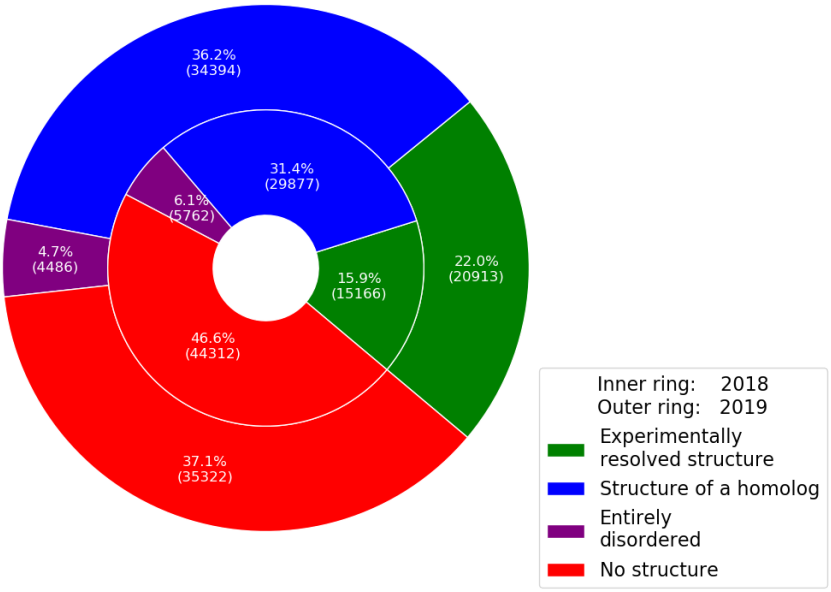


Figure 5.7: *Fractions (total numbers) of proteins from the human proteome, for which StructMAn was able to map at least one structure.*

annotation was done in 2018 and the result was presented in the German Conference for Bioinformatics (GCB) (http://gcb2018.de/abstracts) in the same year. Since then, this dataset functions as a basic database for StructMAn and was reannotated multiple times after each major update. The annotation of this dataset from scratch, which means starting with an empty database, takes around 10 days on a 48-core server (4x Intel Xeon E7-8857 v2 96x 16GB RDIMM, 1.600MHz RAM 1,5TB).
We were able to map over 58% of the proteins to structural data (Figure 5.7) and for the portion of proteins, which could be mapped, the majority was only mapped to the structure of a homolog. These statistics clearly show the benefits of using the structures of homologs. When comparing the 2018 and the 2019 annotation, the number of mapped proteins increased drastically. This has four reasons: the increased numbers of experimentally resolved structures in the PDB, the switch from BLAST [83] to MMseqs2 [86] for performing the sequence similarity search, and overall improved implementation. However the main reason for more mapped proteins and that explains the surprising increase in proteins mapped to corresponding structures from 15.9% to 22.0% is that we changed the way how we calculated the sequence identity of the alignments. In 2018, we divided the number of matched positions by the length of the target sequence. Today, we divide the number of matched positions by the number of positions not matched with a gap. This resulted in higher sequence identity values in general. Around 5% of all proteins, while they could not be mapped to any 3D structure are predicted to be entirely disordered by IUPred2A [137]. That's why we distinguish them from the unmapped proteins.
Figure 5.8 shows similar statistics but on the amino acid level. For the 2018

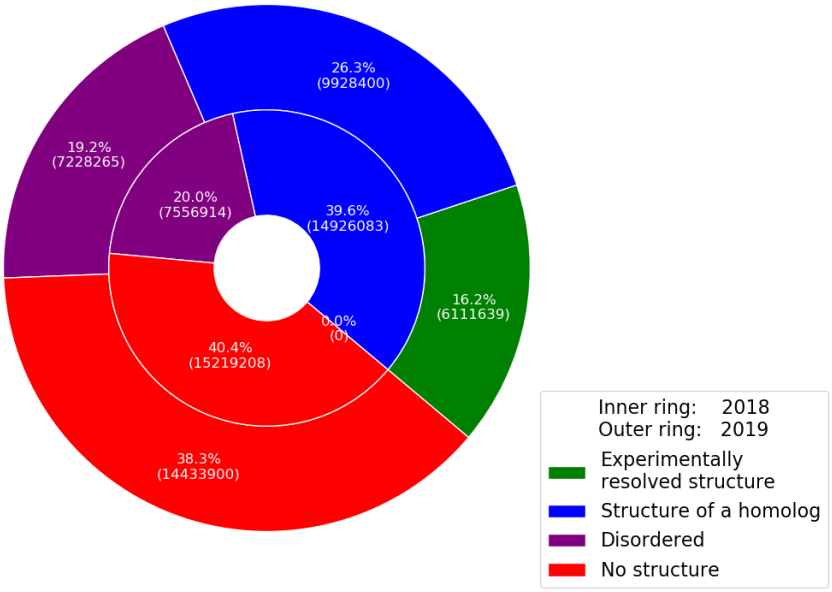


Figure 5.8: *Fractions (total numbers) of individual positions from the human proteome with respect to the availability of StructMAn annotations.*

annotation, we have no data on the distinction between mappings into experimentally resolved structures of the same protein and structures from a homolog. The amount of newly mapped positions comparing 2018 and 2019 is much less than the increase of mapped structures, since the change in the sequence identity calculation has a smaller effect here.

In Figure 5.9 we show the distribution of the classifications for all mapped positions. We present the classifications based on the distance calculations and the classifications based on the RINs separately. Both classifications schemes assign around 31% of the positions as lying in disordered regions because here distance-based classifications and RIN-based classifications do not differ. Around 47% are not directly involved in any interactions, where 20% of the positions are mapped to residues lying in the protein core and nearly 27% of the positions are mapped to residues lying on protein surfaces. The remaining 22% of the positions are assigned to an interaction type classification.

For these positions, the assigned type of interaction classes differs for the two classification schemes. The distributions of interaction types are shown in Figure 5.10. Here we can see how the two ways to assign contacts differently influence structural classification. The RIN-based scheme classifies more positions to be involved in protein and DNA interactions and fewer as ligand and metal interactions. The numbers of ion and RNA interactions are roughly the same for the two schemes. Since the RIN-based classification prefers a larger surface of the interaction interface between the mapped residue and the interaction partner to the minimal distance between both interacting partners, we observe a propensity for this scheme to favor larger interaction partners. Since the distance-based classification, as its name suggests, favors just the

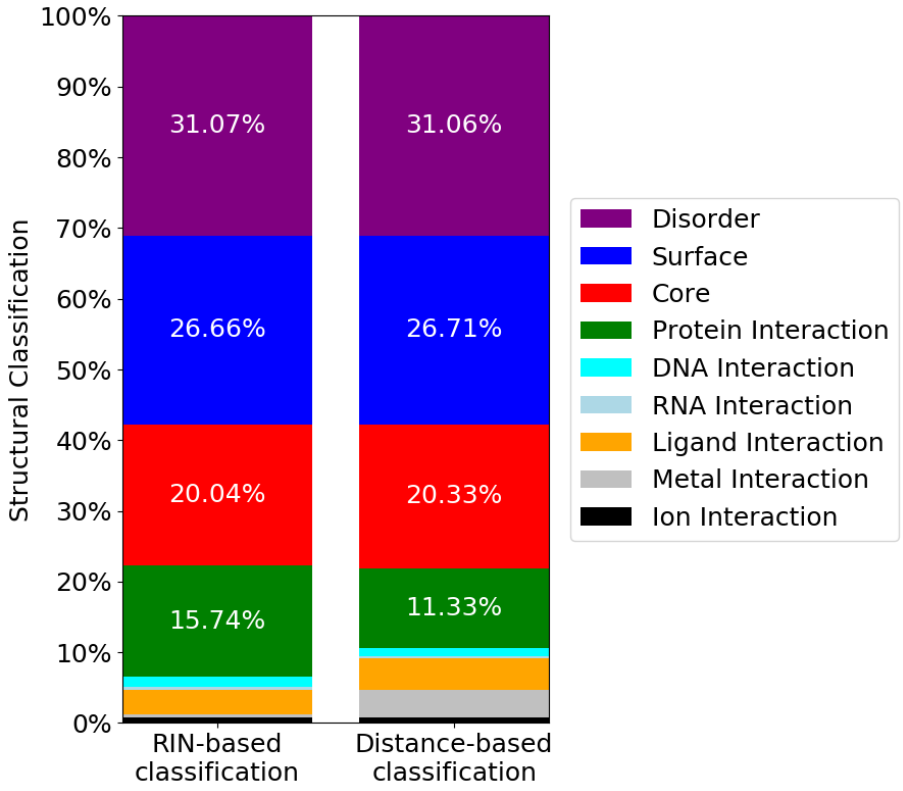


Figure 5.9: *Distribution of RIN-based and distance-based structural classification for the annotation of the human proteome; interaction classes can be seen in detail in Figure 5.10.*

shortest distance between any two atoms from both interaction partners, there is no such trend to be expected.

5.3.2 *Annotation of All nsSNPs of the Genome from an Individual Human Being*

In the first place, StructMAn was developed to map nsSNVs into the context of protein 3D structures. A good example of a large nsSNV-related dataset is a collection of all the individual nsSNPs of one human being. In this example, we took a randomly chosen individual from the 1000 genomes project [151] and obtained all genetic variations in this particular genome in the form of a variant call format (vcf) file. The resulting dataset contains 8750 nsSNPs in 4321 different proteins. In order to provide a protein-specific background, we also annotated all positions in these proteins, which in total contained $3,294,328$ positions.
Comparing the class distribution of this dataset to the class distribution of the human proteome (Figure 5.11), they look very similar with only one difference: for the individual nsSNVs there are more positions classified as surface in favor of positions classified as core. This comparison may not be completely fair since the annotation of the human proteome includes all protein isoform sequences and ANNOVAR mapped the variants only to the major isoforms. When comparing the individual nsSNVs to all positions from the same proteins, we can see an even stronger depletion for positions classified as core, but this

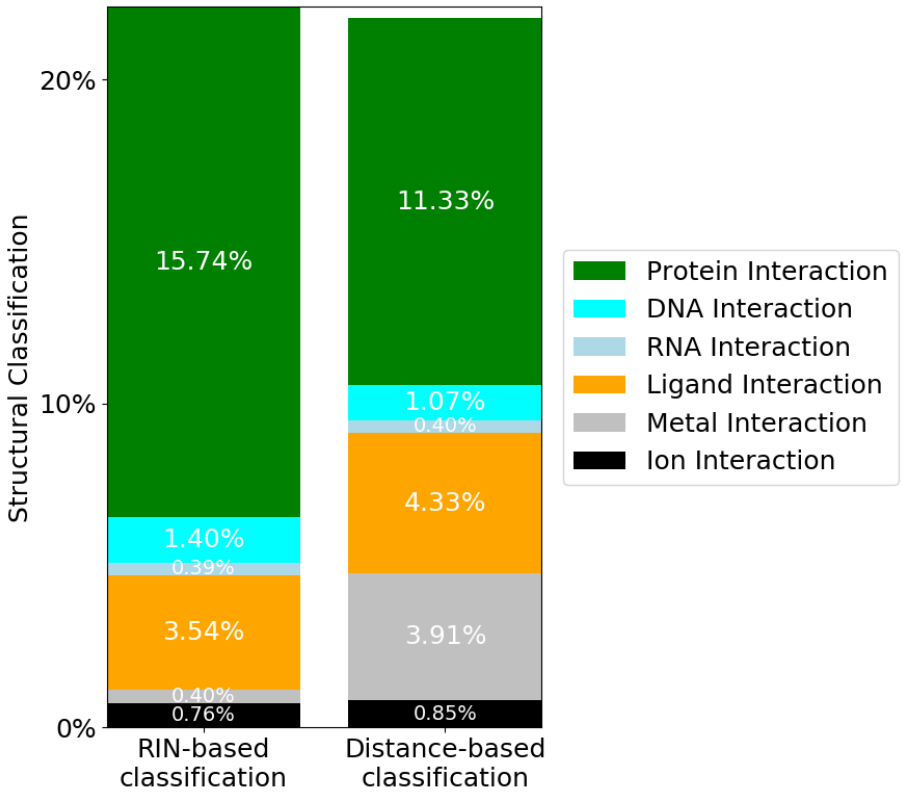


Figure 5.10: *Distribution of RIN-based and distance-based structural interaction classifications over the annotation of the human proteome.*

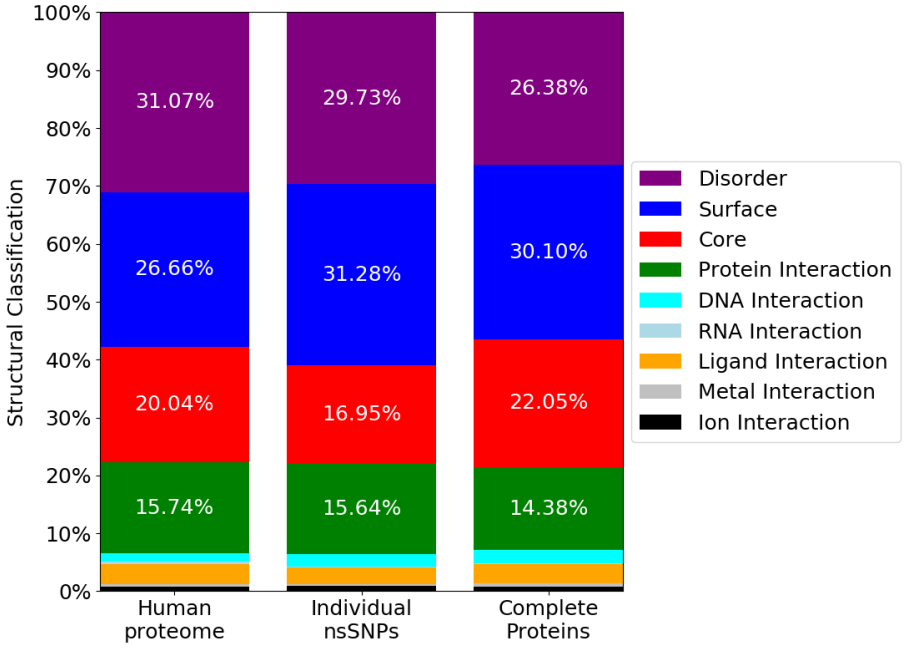


Figure 5.11: *Distribution of RIN-based classifications for the annotation of the human proteome, for all nsSNPs from an individual human being, and for all positions for the same set of proteins; interaction classes can be seen in detail in Figure 5.12.*

time more in favor of positions predicted to lie in disordered regions. This might imply that the distribution of mutations in humans is not completely independent from the structural composition of the proteins they affect.

5.3.3 *Performance Comparison*

We tested the runtimes for seven different types of input for five different installations:

1. The in-house version on a 48-core server (4x Intel Xeon E7-8857 v2 96x 16GB RDIMM, 1.600MHz RAM 1,5TB), full installation with locally installed versions of the PDB and Uniprot, starting with an empty database (from scratch).

2. The in-house version on a 48-core server, full installation with locally installed versions of the PDB and Uniprot, using the lite mode.

3. The in-house version on a 48-core server, full installation with locally installed versions of the PDB and Uniprot, using the annotation of the human proteome as the database.

4. The containerized version run on the same servers, but with access limited to 4 cores. Additionally, the containerized version had no access to locally installed databases, namely Uniprot, PDB and the RIN version of PDB, and for it the RIN-based classifications were not computed.

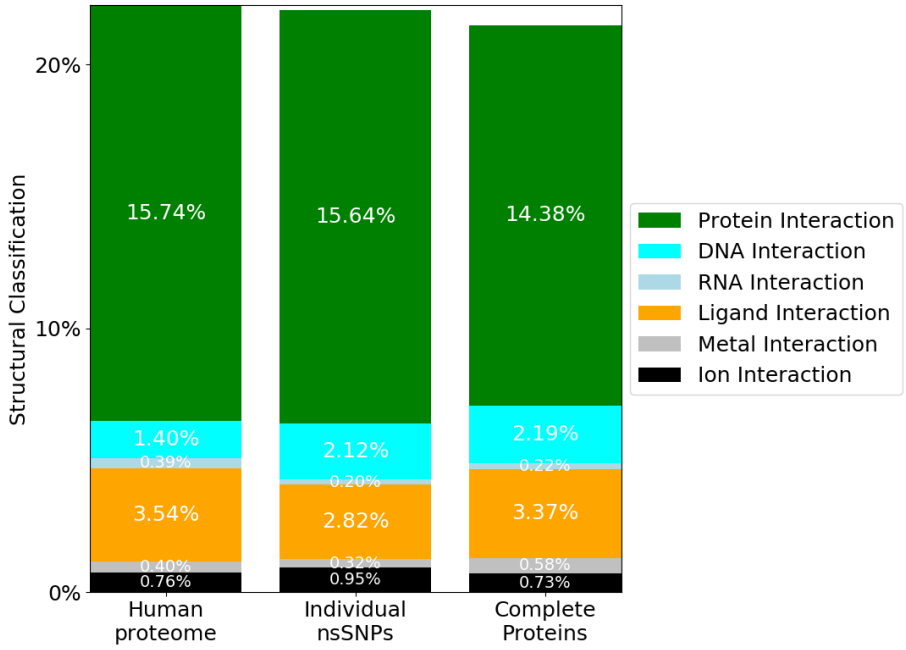


Figure 5.12: *Distribution of RIN-based and distance-based structural interaction classifications over the annotation of the human proteome, for all nsSNPs from an individual human being, and for all positions for the same set of proteins.*

5. The containerized version, limited to 4 cores, without access to the local databases, in the lite mode.

The input scenarios are:

1. A single position from the major isoform of the human protein p53, as an example of a protein that has a lot of experimentally resolved structures in the PDB
2. All positions from p53
3. A single position from the major isoform of the human protein Sart3, as an example of a protein that has only a few experimentally resolved structures in the PDB
4. All positions from Sart3
5. 86 individual positions from 86 different human proteins (taken from the cancer germline dataset, described in Chapter 6.2.1, since we can annotate all of these proteins to a large amount of 3D structures)
6. All positions from the same 86 different human proteins
7. All nsSNVs from one individual (Section 5.3.2)

All results are presented in Table 5.2. The overhead resulting from building the database can be seen when the 'from scratch' column with the 'lite mode' column are compared. In contrast, the advantage of using a pre-built database can be seen when comparing the 'lite mode' column with the 'full database' column.

Scenario	in-house from scratch	in-house lite mode	in-house full database	container from scratch	container lite mode
p53 1 SNV	88	14	8	294	129
p53 full	97	26	8	301	233
Sart3 1 SNV	21	6	7	30	15
Sart3 full	28	20	7	37	26
86 SNVs	1783	463	13	18216	8420
86 full	2697	2623	19	18344	15014
all SNVs	28289	7571	530	not done	not done

Table 5.2: *Runtimes for the different scenarios and different Installations of StructMAn in seconds.*

The database construction overhead is larger for proteins with more experimentally resolved structures (p53 vs. Sart3) and for inputs with one individual position, since during the database construction always full sequences are annotated. For the analysis of a few mutations, the construction of a database seems unnecessary, but when handling larger datasets, the database-related overhead amortizes quickly. The 48-core version is about ten times faster than a local 4-core version, especially in combination with the lite mode. Such a local installation can handle small to medium-sized inputs.

5.4 DISCUSSION

Over the last four years, we developed StructMAn, which acts as the computational centerpiece of this thesis. We wanted to learn from the shortcomings of other structural annotation methods and to fill all the unsupported niches the field demands. Consequently, we have created a structural annotation tool, which can process the largest inputs without any kind of limitations, be it species, isoforms or specific positions. The comprehensiveness of the performed structural analysis is also nearly unmatched, especially when considering the amount of input data it can be applied to. However, we failed so far to deliver this powerful method to the community. The webserver implementation lacks a lot of important features. The containerized version is finished and working, but still not published.
A common problem for all structural annotation methods are cases where no experimentally structures are available. The annotation of the human proteome shows that there are still 38% of all positions left out. The amount of structures in the PDB increases constantly and rapidly and recent developments in cryo-EM and hybrid structure resolution methods keep the hopes alive to further reduce the amount of unannotated positions in the future.
One outstanding question is if protein structure modeling techniques can improve the coverage of structural annotation, particularly given the recent developments in that field. The integration of deep learning methods for the prediction of residue-residue contacts in protein structure prediction pipelines improved especially the quality of the models for proteins, which were considered to be very hard to model [111]. But since in our opinion, the most

important thing about protein structural annotation are the interactions it participates in, we doubt that protein structure modeling alone can solve this issue. Still, it will be wise not to completely rule anything out.

For example, such model can be used for structural comparison with other resolved structures, including those of complexes, and in this way can assist finding remote homologs that are missed by sequence similarity searches.The combination of information gained from structural analysis of multiple different annotated structures and hence the following structural classification are also complex tasks. The usage of structure quality score in order to weigh different structural analysis results is a simple solution to this problem. However, the calculation of the quality scores is based on ad-hoc heuristic parameters, which are not evaluated or optimized, which leaves always room for potential criticism.

There is no gold-standard set, for which we can optimize our parameters in order to produce the perfect structural classification. While most structural annotation methods perform such a classification, the structural classes differ from method to method. The distinction between residues on the protein surface and residues buried in the core is the only classification common to all methods that conduct a structural analysis. We tried to come up with a classification scheme as logical and biologically meaningful as possible. In that regard, a unique feature of the structural classification StructMAn is the way it combines information from different protein structures with varying quality.

After investing four years into StructMAn and working on structural annotation methods, I am convinced that there will be always room for improvement and as long as there is a demand to map genomic data to protein structures, all the time invested into the improvement of structural annotation methods is well-invested time.

6

STRUCTURAL ARRANGEMENTS OF GENETIC VARIANTS ASSOCIATED WITH GENETIC DISEASES

In this chapter, I present an application of the methodology introduced in Chapter 5 to biologically important data. We show the benefits of using a high-performance annotation pipeline on datasets of phenotypically annotated mutations. The basic idea is to compare the differences between collections of mutations associated with genetic diseases and mutations that are known to have no deleterious effect. This study is not focused on the analysis of single mutations for their individual effects but was designed to show that the structural analysis of mutations is able to discover meaningful trends.
This study was published in Oncogenesis in 2017 [152]. This chapter majorly contains the same information as the publication but is extended in that it is explored if the methodological updates of StructMAn since the time of publication had any influence on the results of the study.

6.1 INTRODUCTION

The fight against genetic diseases can only be won by understanding their mechanisms. The research of the basic mechanisms behind genetic diseases is one of the largest fields in biology and medical research, but still, there remains a lot to explore. Next-generation sequencing technology enabled genome-wide association studies (GWAS), in which many genomes, for which its associated phenotype is known, are sequenced. Therefore, it is possible to compare the genomes of healthy individuals with the ones of those carrying a genetic disease. Some genetic variants appear significantly more often in genomes, which are associated with the same phenotype. Thus we can statistically link certain non-synonymous single nucleotide variations (nsSNV) to specific genetic disease phenotypes. We call them disease-associated nsSNV. This type of annotated nsSNVs are used in this project.
We differentiate the disease-associated nsSNVs further into three subsets: cancer-associated germline variants, cancer-associated somatic variants, and non-cancer disease-associated variants. They are compared with mutations annotated as benign and variants common in the population. For all datasets, we applied our structural classification pipeline. From the resulting classifications, we calculated the distributions, called spatial distributions, which we then compared between the different datasets.
Previously (Section 2.4.1) we described possible ways mutations affect the function of the corresponding protein and that the magnitude of the effect is in correlation with the effect on the phenotype. The assumption is that mutations located on any kind of interaction interface and the mutations located in the protein core have a higher potential to be functionally important compared to mutations located on the protein surface and are not involved in interactions.

According to this reasoning, we expected an increased propensity of surface mutations in the spatial distribution for the datasets of benign and common mutations and the opposite behavior for the other classes.
The distributions resulting from this comparative analysis setup confirmed these assumptions. However, at that point, we were left with one major concern: possible existence of biases in the analyzed data. After closer inspection, this concern turned out to be founded. The root of the problem is that some proteins are more frequently associated with diseases than others. These proteins are in general more in the focus of research and consequently are more frequently the target of protein structure determination experiments. This results in the availability of more structures for proteins stronger associated with diseases. Ultimately, the amount of known protein-protein interactions for these proteins increases. Of course, mutations associated with disease, also more frequently happen in proteins, which are associated with disease. This is the other way around for common variants and mutations annotated as benign. Hence, the proportion of mutations classified as located on protein-protein interaction interfaces is artificially decreased among benign mutations. One can of course pose the same question for other type of interactions, such as ligand and nucleic acid interactions.
In order to correct for this bias, we moved away from the direct comparison of spatial distributions. We developed multiple techniques for the construction of biological control distributions, including random in-silico mutagenesis, complete protein annotations, and random sampling. The first two are supposed to identify dataset-specific biases. The random sampling had the goal to construct subsets of the common and benign mutation datasets in a way, that their proteins are covered by structural data equally well as their disease-associated counterparts. These techniques helped to compare the results from different datasets by correcting for biases, which are a consequence of inequality of research quantity for different proteins.
The bias corrections had significant effects on the results of this study. We showed that mutations associated with diseases are not enriched in protein-protein interaction interfaces contrary to previous work [153–155]. At the same time, we support the findings of some other studies [156] in this controversial field. Interestingly, using these bias-correction techniques, we were able to confirm the enrichment of disease-associated mutations in ligand-binding pockets and DNA-interaction interfaces.

6.1.1 *Related Work*

There exists ample research on mapping disease associated-variants into the context of three-dimensional structures. For example, the idea behind Cancer3D [125] is the construction of a database filled with genetic variants associated with human cancer, that can be structurally annotated. In Cancer3D, an own structural annotation pipeline is built, which interestingly works the other way around than StructMAn. Its starting point was a database of genetic variants associated with cancer and a list of proteins harboring them. The list of proteins was extended by including all known alternative isoforms. Pfam [129] was used

in order to assign all domains of all proteins, and the amino acid sequences of the domainswere used to construct a search database. Then all sequences from the PDB were compared to the constructed database using a BLAST [83] search. In order to map then individual mutations into protein 3D structure, the alignments constructed by BLAST were used. In a follow-up study, Cancer3D was used to structurally analyze interaction interfaces of cancer-associated proteins [155], and it was found that cancer-associated variants are strongly enriched for interaction interfaces of cancer driver proteins.
Another study [153] also reached the same conclusion. Here protein sequences were directly mapped on their corresponding protein structure using the Uniprot annotation that provide PDB identifiers for each protein if its structure has been structurally resolved. Mutations in tumor suppressor genes, oncogenes, and all other genes were structurally compared. The mutations were structurally classified by their solvent accessibility and by their participation in protein-protein interactions (PPI). The same structural annotation protocol was used by a third study [154], which extended the structural analysis to also identify interaction interfaces to low molecular weight molecules and nucleic acid chains. Again, the results showed enrichment for genetic variants in all types of interaction interfaces of proteins associated with cancer.
In contrast to these studies, another study [156], first identified the bias of well-studied proteins to have more known interaction partners and hence an increased chance of any variant being on an interactive interface. The focus of their study was the analysis of PPI networks, which were mainly constructed without using any structural information. Thus the proteins with no known association to cancer were samples to be used as frequently as baits in TAP-MS experiments (an experimental technique for large-scale identification on PPIs) as the cancer-associated proteins. After this correction, in fact, it was found that there is no enrichment of the number of interactions for cancer-associated proteins. This raised the question if our study that uses structural information suffers from a similar bias. In particular, since cancer-associated proteins have on average more corresponding 3D structures, if we consider only proteins that carry benign variants and select equally well-studied ones, will we observe a similar number of variants in PPI interfaces?

6.2 METHODS

6.2.1 *Disease-associated Variant Databases*

We constructed five datasets of nsSNVs, that are associated with different phenotypes:

1. cancer-associated germline variants (CG)
2. cancer-associated somatic variants (CS)
3. variants associated with other genetic diseases (NCD)
4. non-common variants considered to have no effect called benign (BeV)
5. variants appearing commonly in the population (CoV)

Dataset	Positions	Proteins	Mapped positions	Mapped into a corresponding structure
Common	26868	10341	31.65%	23.53%
Benign	4669	944	32.47%	11.24%
Cancer germline	349	86	99.71%	72.21%
Cancer somatic	3291	372	99.42%	71.29%
Non-cancer diseases	12315	1588	99.79%	63.56%

Table 6.1: *The five main datasets, including for each the total amount of positions, the total number of proteins, the fraction of position, that could be mapped to at least one structure and at least one corresponding structure (i.e. structure of the exactly same protein).*

The origin of our datasets of phenotype-associated variants are annotated variants from the databases ClinVar [63], COSMIC [78] and Uniprot [2]. The variants from these three databases form the raw data pool used to compose the datasets used in this study. All databases contain variants, which could be observed to be in association with specific resulting phenotypes. The magnitude of the impact of the individual variants on the phenotype is not known and we assume all variants in the same dataset to be equally important. We normalized the labels of the variants in order to ensure sensible collective usage. We filtered variants from each dataset specifically.
Variants from ClinVar were only included if there rating was 2 stars or more (stars system of ClinVar is explained in Section 3.1.3). Variants annotated as 'pathogenic' or 'likely pathogenic' are divided into cancer-associated and non-cancer associated variants based on the NCBI MedGen disease classification, while the cancer-associated variants were further divided into germline variants and somatic variants, based on the 'origin' annotation from ClinVar. The variants that are annotated as 'benign' or 'likely benign' are used to construct the dataset of benign variants.
Variants originating from COSMIC are all somatic cancer variants. Here we only included variants associated with at least two different cancer samples in order to select only reproducibly observed cancer variants and exclude random variance.
Finally, we retrieved all variants from Uniprot that are associated with human disease using the 'humsavar.txt' file. Based on specific keywords, we filtered cancer-associated variants from non-cancer associated variants. The subdivision into pathogenic or benign and somatic or germline is done similarly to variants from ClinVar.
We used ExAc [47] to construct the fifth set containing variants, which are commonly carried in the population, for which we assume to have no deleterious effects. For this we retained variants with a population frequency $\geqslant 5\%$. The compositions of the resulting datasets are shown in Table 6.1.

6.2.2 *Control Datasets*

Since the datasets all have different protein composition, there is a risk of potential protein-specific biases. In order to compare the disease-associated, benign and common variant datasets we provided control datasets for them.

6.2.2.1 *Random Control Datasets*

For each dataset, we computed a randomized dataset. For each protein, we introduced n random nucleotide substitutions, where n is the number of variants of the protein in the corresponding base dataset. We repeated this process ten times. This resulted in randomized datasets, which have the same amount of variants per protein and the same composition of proteins as their original datasets. All presented results for the randomized datasets report the mean over the ten repetitions.

6.2.2.2 *Sampling Control Datasets*

Additionally, we designed a sampling technique that had the goal to enable better comparison between datasets that contain variants from proteins, which are studied to a different extent. As a proxy measure for how well a protein is studied, we took the number of experimental structures StructMAn was able to find matching this protein. In this technique, the BeV and CoV dataset was used as two pools of variants. We selected variants from each pool such that the distribution of numbers of mapped structures for all proteins is as similar as possible to the corresponding disease-associated datasets (note that this is only possible since we have many more variants in the BeV and CoV datasets than in the datasets with disease-associated variants). This method allows us to directly compare datasets with variants associated with different phenotypes, while significantly reducing the bias introduced by a different protein composition.

6.2.2.3 *All Positions Control Datasets*

At the time we conducted this study, the StructMAn pipeline was not powerful enough to annotate complete proteomes. Since we are now able to do so, for each dataset, we added the spatial analysis for all positions of the same proteins and also the comparison to the annotation of the whole human proteome (reported in Chapter 5). This control datasets can capture biases introduced by the specific composition of proteins of a dataset. However, they remove the information about the distribution of variants per protein.

6.2.3 *Spatial Distribution of Genetic Variants*

For all variants of all datasets, the structural classifications by StructMAn were calculated. Since in this study the focus was not on individual variants, we performed a simplified analysis that takes into account only the simple position classification (Section 5.2.8), which should be capable to show trends for spatial location of mutations for whole datasets. We calculated the distribution of classes for the variants belonging to the same set. We call such distributions

spatial distributions. The intention of spatial distributions is to show statistical trends in different datasets of variants in order to explore enrichments in specific classes, which then may enable conclusions about the overall mechanistic trends behind different phenotypes.

6.3 RESULTS

6.3.1 *Spatial Distribution of Disease-associated and Benign Genetic Variants from Gress et al. [152]*

The spatial distribution in the different datasets (Figure 6.1) reveals that the inclusion of the analysis of structures of homologs not only increases the number of total data points but also the proportion of positions classified as being in an interaction universally for all datasets. Other than that, the inclusion of these structures does not change the overall trends in the distribution, particularly when it comes to the comparison of the distributions from different datasets.
The central task of this study was the structural comparison between disease-associated variants with variants annotated as benign and common variants. We can see an enrichment of variants classified as ligand interaction in the three disease-associated datasets (CG, CS, NCD) compared to the BeV and CoV datasets. For the NCD dataset, there is also a strong enrichment of variants classified as core. Variants in CG and CS datasets are enriched in all types of interaction interfaces, protein-protein, protein-low molecular weight molecules, and protein-DNA. Variants in CG are especially often classified as ligand interaction. We observe very few variants classified as lying on RNA-interaction interfaces and exclude them from the further analysis.
Comparing the datasets of variants not associated with disease (BeV, CoV) to their corresponding randomized versions, one notices a depletion in variants classified as core in favor of variants classified as surface, whereas the proportion of variants classified as lying on any of the interaction interfaces are rather unchanged. Doing the same type of comparison for the NCD dataset reveals that there is an enrichment of variants classified as core and variants involved in interactions with small molecules in the actual datasets compared to their randomized versions, confirming the previous observations drawn from the direct comparisons. Putting the cancer-associated datasets (CG, CS) against their randomized counterparts, we can again confirm the enrichment of variants classified as ligand and DNA interactions, however, there is no significant difference in the variants classified as interacting with other protein chains. We can even see a slight depletion of variants classified as core, similarly to the BeV and CoV datasets.
Similar conclusions can be drawn from Figure 6.2: the distances to the next DNA chain are significantly shorter when disease-associated datasets to their randomized versions are compared. This difference is especially noticeable for variants from the NCD dataset. The BeV and CoV datasets show an opposing trend. In a direct comparison to all other datasets, variants from the CoV dataset seem also to have shorter distances to DNA molecules, but they are even shorter for variants from the randomized counterpart. Regarding the

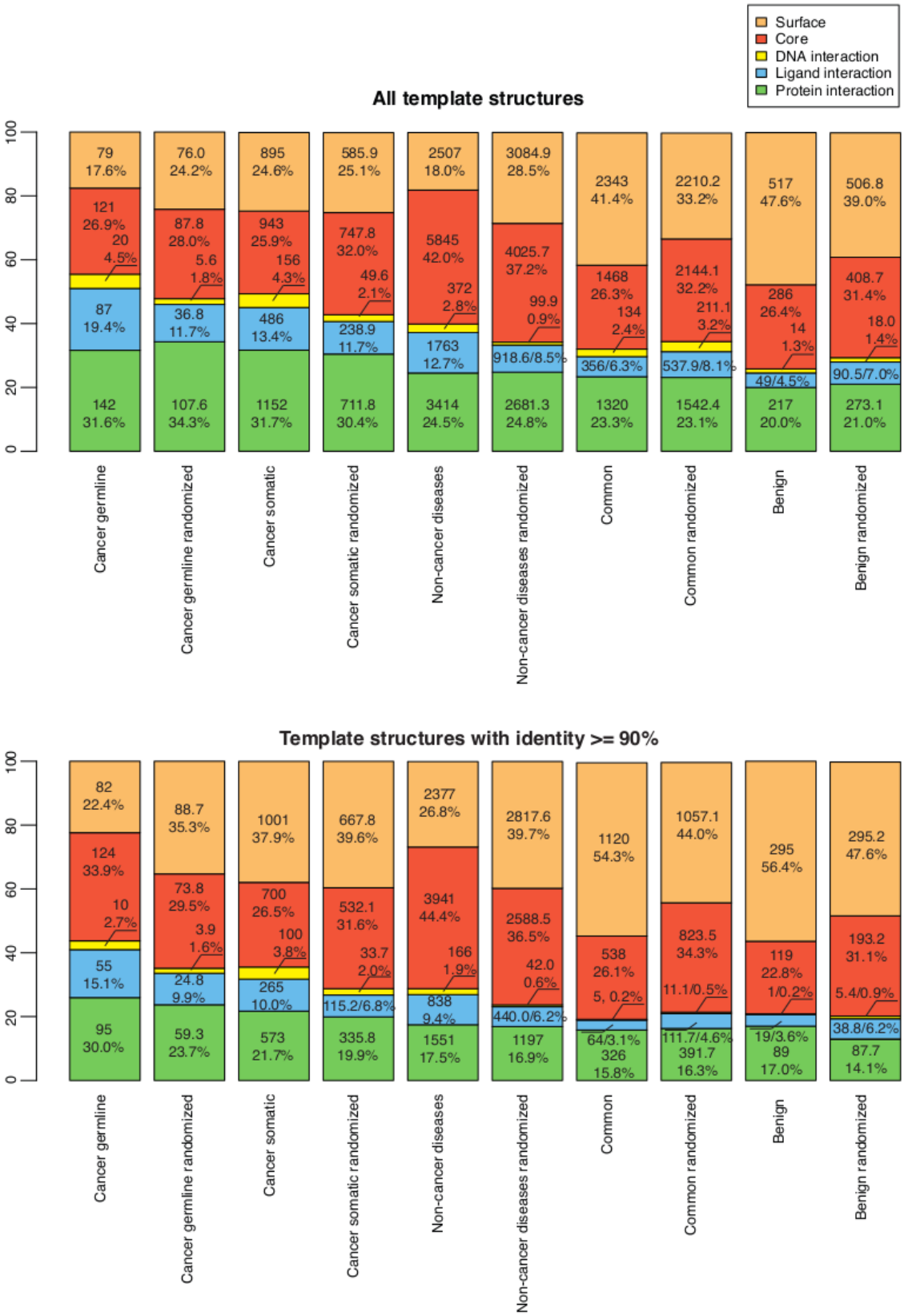


Figure 6.1: *(from Gress et al., 2017 [152]): Spatial distribution of nsSNVs in the analyzed data sets. For randomized data sets, mean values over* 10 *replicas are used. (top) For templates with* $\geqslant$ *35% sequence identity. (bottom) For templates with* $\geqslant$ *90% sequence identity; RNA interaction class was excluded, due to the very low number of detections of variants in that class.*

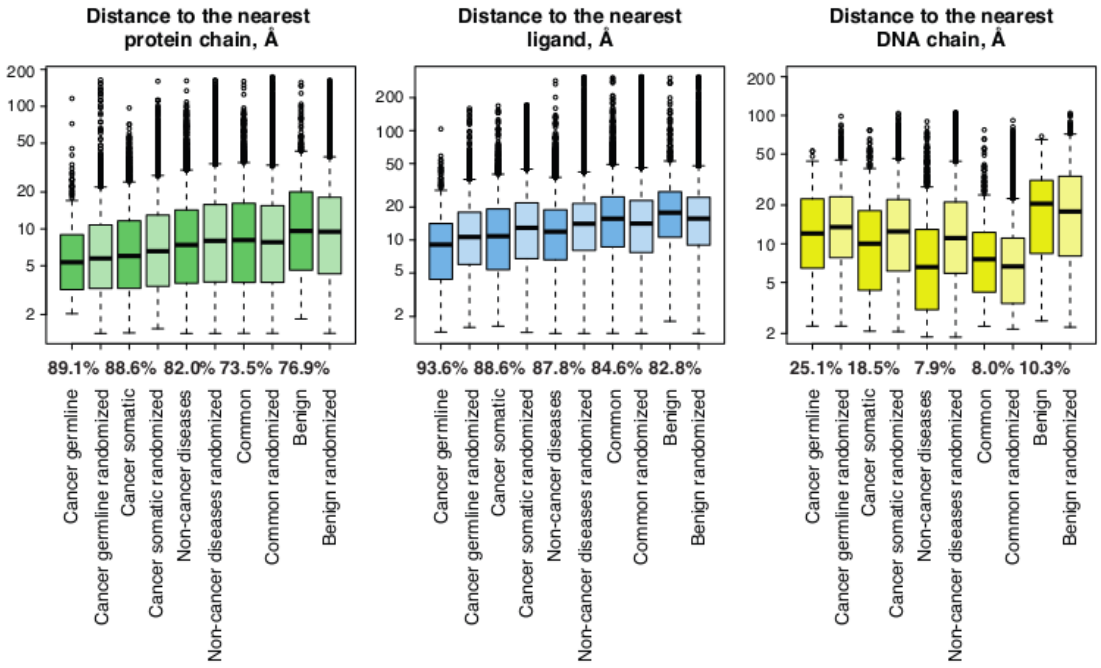


Figure 6.2: *(from Gress et al., 2017 [152]): Distance between residues corresponding to nsSNVs and the nearest interaction partner (log scale). Biological data sets are shown in a darker shade. The fraction of mapped nsSNVs, for which a template with a co-resolved corresponding interaction partner is provided below boxes representing distribution of distances to protein, ligand and DNA interaction partners for each biological data set. For randomized data sets, all* 10 *replicas are used to create the plots. (left plot, green) Distances to the nearest protein chain. (middle plot, blue) Distances to the nearest ligand. (right plot, yellow) Distances to the nearest DNA chain.*

	DNA contacts (%)	Ligand contacts (%)	Protein contacts (%)	Core (%)	Surface (%)
cancer germline	4.5	19.4	31.6	26.9	17.6
100 random samples from cancer germline (mean ± s.d.)	1.2 ± 0.4	5.8 ± 0.7	32.8 ± 1.5	30.6 ± 1.5	29.7 ± 1.9
cancer somatic	4.3	13.3	31.5	25.9	24.9
100 random samples from cancer somatic (mean ± s.d.)	1.4 ± 0.2	8.6 ± 0.3	29.5 ± 0.5	27.4 ± 0.6	33.2 ± 0.6
non-cancer diseases	2.8	12.7	24.5	42	18
100 random samples from non-cancer diseases (mean ± s.d.)	1.6 ± 0.1	6.0 ± 0.1	24.1 ± 0.3	27.5 ± 0.3	40.8 ± 0.3

Table 6.2: *(from Gress et al., 2017 [152]) Distribution of structural classes in disease-associated datasets compared to 100 randomly sampled equally sized sets from the sets of benign variants with the same distribution of identified template structures as in the corresponding disease-associated datasets. Insignificant differences (within four standard deviations) are marked in red.*

distances to the closest low molecular weight ligand, as long as we compare the datasets to their randomized versions, we can draw the same conclusion as drawn for the distances to DNA molecules: the expected shift towards shorter distances for disease-associated variants (CG, CS, NCD) can be observed. For the distances to the nearest protein chain, none of the datasets show any significant differences to their randomized versions. However, there is the expected trend of shorter distances to the nearest protein chain in the direct comparison of disease-associated datasets to CoV and BeV datasets, similar to the ligand distances.

6.3.1.1 *Random Sampling to Close the Gap in the Amount of Structural Information*

For disease-associated variants, we can find more structural information than for variants in the BeV and CoV datasets (see Table 6.1). This could bias the spatial distributions of the corresponding datasets. We designed a random sampling technique to match the amount of structural information between two datasets and performed it to compare datasets of the disease-associated variants (CS, CG, NCD) with the BeV dataset (Table 6.2) and with the CoV dataset (Supplementary Table 9.4).
In Table 6.2 the spatial distributions for the disease-associated datasets (same as in Figure 6.1) are compared to their corresponding sampled control sets constructed using the variants from the BeV dataset. Interestingly, the increase in variants classified as protein interaction for the disease-associated variant datasets cannot be observed anymore. Surprisingly, this test does not show any enrichment of variants classified as core for the CG and CS datasets, but still, we can observe this enrichment for variants annotated as core in the NCD dataset. This may indicate that mutations in cancer more often change protein function instead of completely disrupting it. In unison to all here presented

analyses, for all three datasets with disease-associated variants, we observe an enrichment of variants that are involved in interactions with small molecules and DNA chains. The results from the sampling control test argue against the theory of enrichment in PPI interfaces for variants associated with cancer, supporting the study by Schaefer et al. [156].

6.3.2 *Spatial Distributions for Disease-associated and Benign Variants Calculated with the Latest Version of StructMAn*

Since the publication of this study in [152], StructMAn has experienced significant improvements. The classification of disordered regions, more available protein structures, an improved classification process and the ability to annotate all positions of a protein are newly developed features, that can have an impact on the results of this study.
In Figure 6.3 we compare the spatial distributions of the five datasets used in [152] with the spatial distribution of the complete human proteome. The spatial distribution for the CoV and BeV datasets is very close to the spatial distribution of all positions in the proteome, and it has a strong resemblance to the spatial distribution of the nsSNVs from an individual from the previous chapter (Figure 5.11). This is another hint, that variants common in the population are slightly depleted from protein core due to their deleterious nature.
The other three datasets comprise variants, which are associated with diseases. Through the inclusion of the class for disordered regions, their spatial distributions are strikingly different from the spatial distributions of the proteome ,the CoV, and the BeV datasets. Considering only this plot, a premature deduction would be that variants that associated with disease are differently distributed in the protein structure space. But since all the datasets have a different protein composition, the question arises, if the different distributions are directly comparable.
In the same fashion as in the original study, we need to construct dataset-specific control sets in order to identify the dataset-specific biases. This time we have constructed two types of such control sets: datasets constructed from all positions from the same set of proteins and randomized datasets. We calculated the structural annotations and obtained the spatial distributions for all of them (Figure 6.4).
The first thing to note is that the spatial distributions of both types of control datasets look more similar throughout all five datasets than the five original distributions (Figure 6.4 A-E). The distributions from the datasets covering all positions from the same proteins directly showcase the biases introduced by the individual protein compositions of the datasets. The distributions of the random datasets also are able to reflect the protein composition bias, but still, there are slight differences. For example, the spatial distribution of the randomized versions of the BeV dataset gets very close to the spatial distribution of the proteome. The reason for the slight differences for both types of control datasets is that the random control datasets are sensitive to the mutation-per-protein distributions of the base datasets and so we believe, that the random datasets yield spatial distributions better suited for unmasking dataset-specific biases.

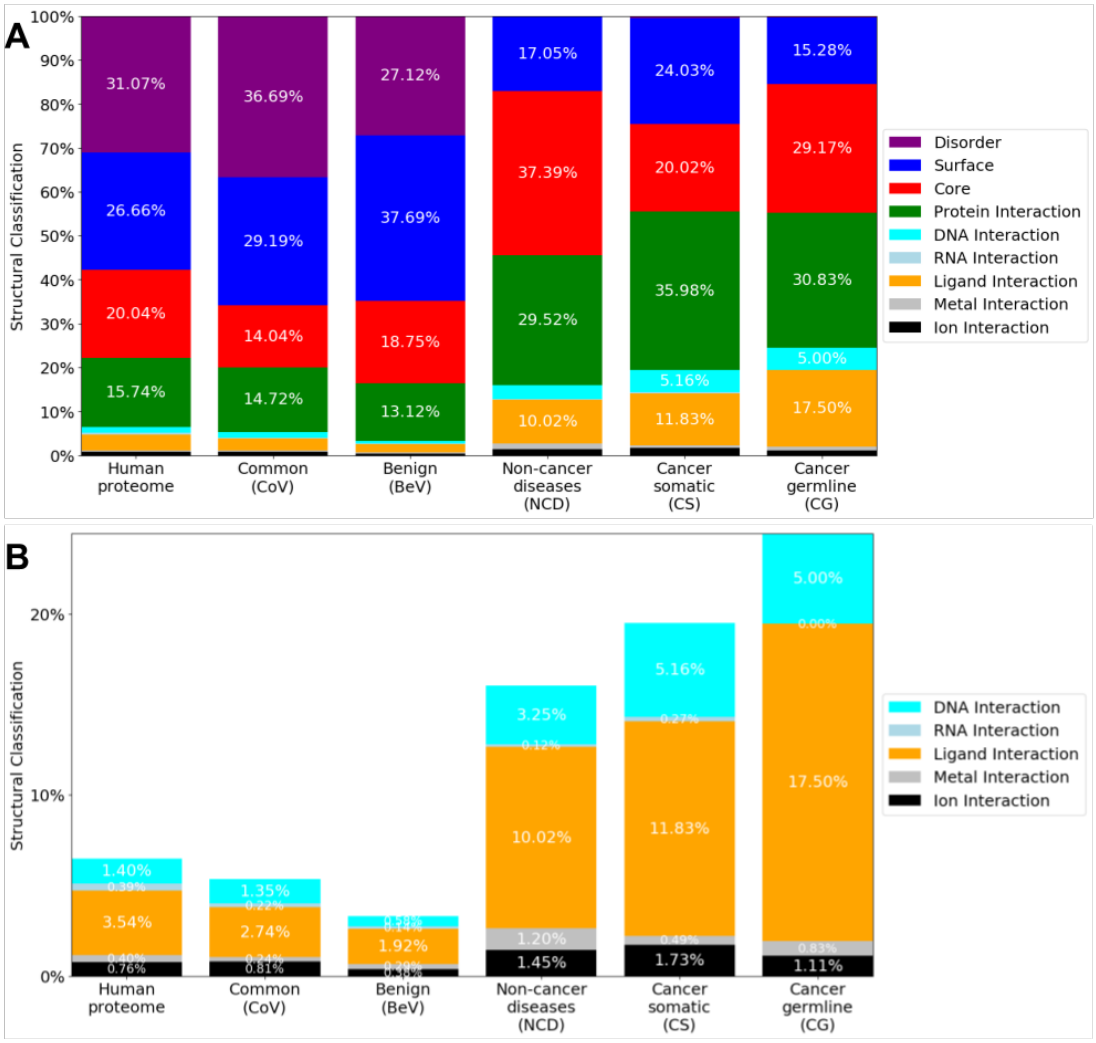


Figure 6.3: *A: Spatial distributions or the five main datasets in comparison to the annotation to the human proteome; B: Spatial distribution of the interaction classes (without protein interactions).*

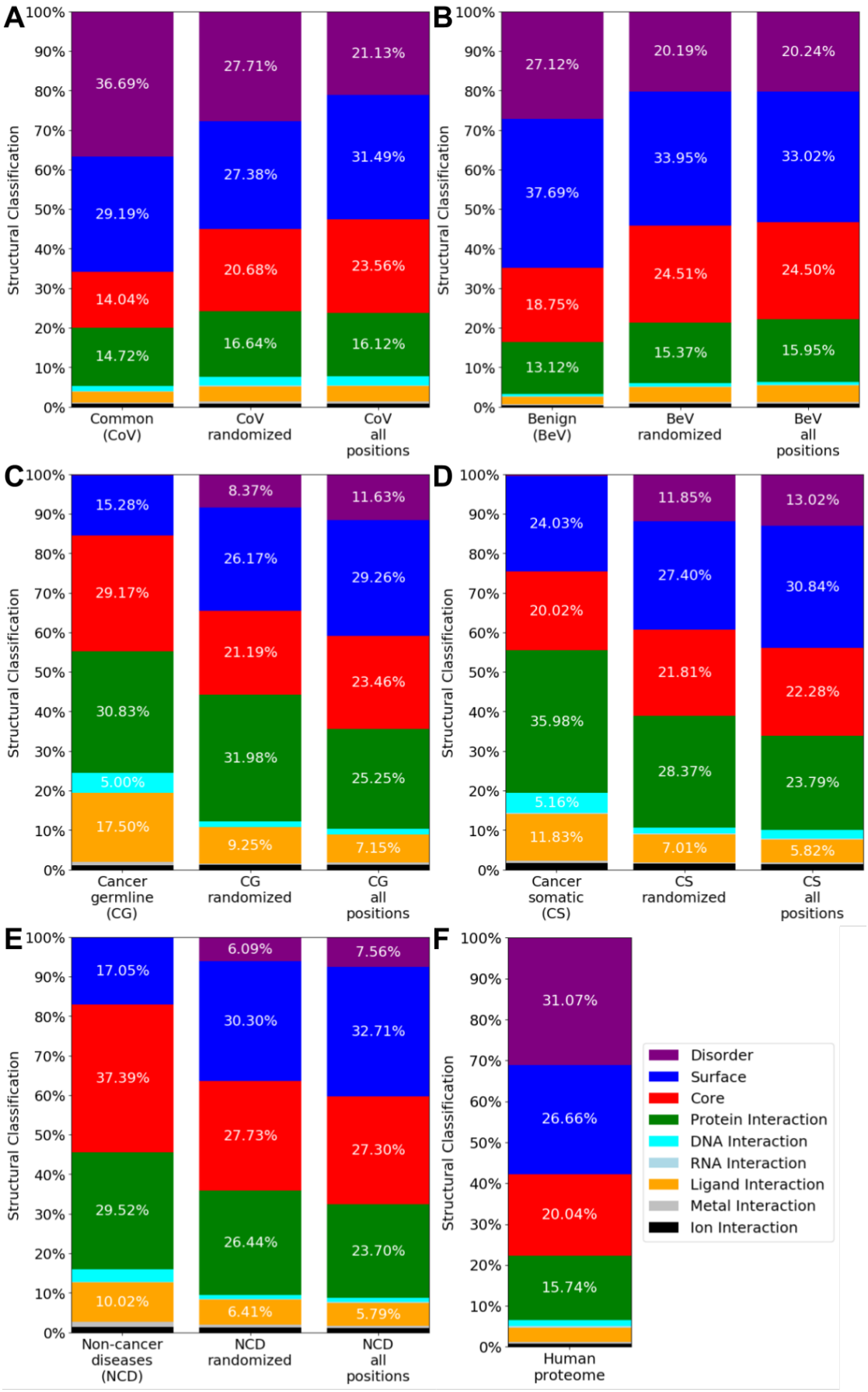


Figure 6.4: *A-E: Spatial distributions of the five main datasets in comparison to control datasets; F: Spatial distribution of the human proteome (same plot but focused on interaction classes in Supplementary Figure 9.1).*

Disease-associated datasets show in general the same traits when compared to their randomized versions. Variants are depleted from the less functionally important classes (surface and disorder) and are enriched in the core and interaction classes. This trend is switched for the CoV and BeV datasets, showing even less functionally important classifications than its randomized control dataset.

6.4 DISCUSSION

The study [152] was to its date the most comprehensive structural analysis of genetic variants associated with cancer and genetic diseases ever done. Besides the number of analyzed variants, we also claim that we used the most precise structural analyses. In particular, the consideration of dataset biases, by the construction of randomized control distributions make this study unique in its class. In that regard, the newer version of StructMAn enabled an even deeper investigation. Especially, the class for detection of residues in disordered regions made the analysis with the newer version of StructMAn a promising undertaking.

The conclusion from the original publications mainly confirmed the findings from the other studies mentioned in the related work section [153–155] with the exception of the enrichment of cancer-associated variants in protein-protein interaction interfaces, where our results agree with Schaefer et al.[156] and are suggesting that there is no such enrichment. This agreement is especially worthwhile since the underlying type of data differs between the two studies, but point to a similar type of bias.

Clearly, the direct comparison of disease-associated variants to common variants shows enrichments in all interaction interfaces as was the outcome of multiple previous studies, but the inclusion of dataset-specific biases question the reliability of such comparisons. Still, even after the correction for biases, one can see the enrichments in interaction interfaces of disease-associated variants, at least if compared to common variants.

In the new analysis, the inclusion of the disorder class radically changed the appearance of the spatial distributions. For proteins with more available experimentally resolved structures, the proportion of positions classified as disorder is decreased by definition: when disordered regions are part of a structure, the classification based on real structural classes will have the precedence, and most such position will be classified as surface. As a result, for datasets, for which we can map the majority of positions into 3D structures we see nearly no positions classified as disorder. However, the distributions will be essentially the same when we combine surface and disorder classifications, which both represent expected neutral effects.

Disease-associated variants are enriched in interaction interfaces with small molecules and DNA chains. This holds true for all performed comparisons: disease-associated variants against variants from the BeV dataset, against variants from the CoV dataset, and against their corresponding control dataset.

7

PREDICTION OF EFFECTS OF GENETIC VARIANTS ON PROTEIN FUNCTION AND CLINICAL EFFECT

In this chapter, I present unpublished data on how high-performance sequence-to-structure annotation can be used for generation of features that are further used for training a machine learning method for prediction of effects of nsNSVs on the phenotype. We discuss the general process of generation of features related to protein 3D structure and the usefulness of such features in different prediction scenarios. This project is ongoing and presents ample room for further development as well as open questions.

7.1 INTRODUCTION

Predicting the various effects of genetic variants is one of the very competitive fields in modern computational biology. There are methods for all thinkable prediction scenarios, be it different types of variations and mutations or different types of effects. They range from very specific scenarios like predicting the change of the binding affinity between two proteins to very broad approaches like predicting the overall clinical effect. Regarding the type of genetic variant, most of the work is focused to single nucleotide variations (SNVs) that are one of the most prevalent types of genetic variants (for details see Chapter 2.4), but other types of variation, e.g. short insertions and deletions are also sometimes considered. Of all SNV, those that happen in coding regions and lead to a change of amino acid sequence (non-synonymous, nsSNVs) are particularly in the focus of research. Machine learning-based prediction methods are prevalent in this field and are characterized by two major factors: first, which machine learning method is used, and second, which features are generated for a given genetic variant.

The most commonly used and most successful features throughout the whole field are the so-called evolutionary features, which roughly correspond to the conservation of the wildtype and mutant amino acids. In order to generate this type of features, a collection of evolutionary related sequences is needed. The rationale behind evolutionary features is that unfavorable genetic variants are sorted out by the evolutionary pressure over time. As a result, functionally important parts of the genome stay conserved through time and thus can be discovered when comparing evolutionary related sequences. Multiple lines of evidence suggesting correlation between amino acid conservation and damaging effects of genetic variants [157, 158], which explains the success of evolutionary features to predict pathogenicity of mutations, however, they do not offer any clue for the mechanisms behind the effect.

Other features are more related to assessing the direct biochemical effect of the change of the amino acid sequence of the protein onto its function. There are multiple mechanisms that can mediate this as described in Section 2.4.1. The

common characteristic of features falling into this category is that they are in some way based on the information gained from the analysis of protein three-dimensional (3D) structures of proteins and complexes in which they may be involved, thus we call them structural features. In comparison to evolutionary features, which can be generated for each variant in the genome, for which we can find evolutionary related sequences, structural features can only be calculated for proteins with known 3D structure. At that point, our structure annotation methods can be very useful. The high-performance sequence-to-structure mapping using structural information from homologs greatly increases the proportion of mutations, where structural features are applicable. Additionally, for variants, for which a corresponding structure is available, we can enrich the structural features with information gained from structures of homologous proteins [35, 114].

It is important to differentiate between methods that aim to predict the impact of mutations directly onto protein functions (e.g. activity of an enzyme or affinity towards an interaction partner) and those that assess the overall effect onto some phenotype (e.g. mutation's pathogenicity). The general benefits of application of structural features in scenarios, where direct effects of mutations on protein function are predicted, are easy to justify. Thus in these scenarios, the usage of structural features is widespread. More critically discussed are the benefits of using structural features when it comes to scenarios predicting effects on the entire phenotypes.

In addition to the structural features, we also implemented the computation of evolutionary features, which are irreplaceable in the field of variant effect prediction. We implemented our own prediction method based on the structural annotation and classification done by StructMAn. Many tools in the field use different types of machine learning methods, e.g. Naive Bayes [11], logistic regression [16] or deep neural nets [15]. As the machine learning approach, we choose the random forest technique, primarily because of the ability of random forests to handle highly diverse feature sets. Moreover, there exists external evidence that random forests perform better compared to other machine learning methods in this particular problem setting [159]. Another advantage of random forests compared to most other machine learning methods, especially neural networks, is their interpretability. This helps in this setting to comprehend the underlying mechanisms of the effect of a variant.

For the prediction of clinical effects, we are particularly interested in predicting effects of mutations that cannot be inferred from the protein as a whole. Such mutations are for example variants from novel proteins (i.e. proteins for which we do not know the outcome of any mutation) and variants that lead to a different outcome than that of other known variants of the same proteins (e.g. benign mutations in disease-associated proteins and damaging mutations in proteins that are not typically associated with disease). We call such variants *difficult*, as one cannot predict their outcome by relying on whether the protein, in which they occur, is associated with disease or not. Our goal is to create a model that understands the mechanisms behind the effect of every variant and uses that understanding for the prediction. Such a model should be able to correctly predict the outcome of difficult as well as easy variants.

7.1.1 *Related Work*

Methods for prediction of the impact of genetic variants can be categorized by the type of effect they aim to predict. There are the clinical effect prediction methods that are usually binary classification methods dividing variants into damaging and non-damaging (or neutral and non-neutral, or benign and pathogenic). One of the first and one of the most well-known methods is SIFT [157], which uses a relatively simple measure based on evolutionary relationships for its prediction. SIFT first determines the most frequent amino acid for the position of the target variant, then it predicts the outcome based on whether the mutant amino acid is chemically dissimilar to the most frequent amino acid. Besides SIFT, most clinical effect prediction methods are supervised machine learning methods. Polyphen-2 [11] uses a combination of evolutionary-based features and simple protein structure-based features in order to train a naive Bayes classifier. Other methods introduce prior knowledge from annotated databases into the models, e.g. SNAP [13] combines evolutionary features, simple structural features and features derived from database lookups in a neural net. Other machine learning techniques were also used: FATHMM [12] uses hidden Markov models, CADD [16] uses logistic regression, DANN [15] uses a deep neural net and M-CAP [160] uses a random forest.
In most of these methods, protein 3D structure is not considered. When it is used, features based on protein structure are comparatively simple and/or are predicted from a sequence. The lack of experimentally resolved structures for most of the human proteins is a major obstacle for the integration of structural features, and thus the inclusion of more complex structural features can lead to the inapplicability of the method for a wide array of input scenarios. However, it has been shown [159] that the more complex structural features can increase the performance of prediction of clinical effect for cases, where they are available.
An important paper of the field (Grimm et al. 2015 [167]) evaluated multiple effect prediction methods using five benchmark datasets: HumVar is one of the training datasets of Polyphen-2 [11]. ExoVar was created to evaluate the performance of different methods for variant effect prediction for data obtained by exome sequencing [168]. VariBench [169] was also created to evaluate prediction methods, it was designed not to contain any variants present in any of training sets of the evaluated methods [170]. PredictSNP is a consensus method and the dataset with the same name was created to evaluate its performance [171]. PredictSNP comprises of variants from three databases: SNPs&GO [172], MutPred [173], and PON-P [174]. SwissVar is a collection of annotated variants retrieved from two databases: UniMed [175] and ModSNP [176].
The conclusion of the paper is that the field is hindered by two types of biases, which could be observed in many commonly used training datasets. Type 1 circularity describes artificially increased model evaluation performances due to variants that are present in the training set and in the test set. It is obvious that such cases lead to inflated test performances and should be avoided at all costs. Type 2 circularity describes the misinterpretations introduced to the model due to protein-specific biases in the training set. They are typically the result of the presence of many proteins in the training set. Here the model learns to predict

Method name	Predicted effect	Model type	Feature types	(Learning) Method
Sift [157]	Clinical effect	Classification	Evolutionary	Conservation-based
Polyphen-2 [11]	Clinical effect	Classification	Evolutionary, structural	Naive Bayes
SNAP [13]	Clinical effect	Classification	Evolutionary, predicted structural, database lookups	Neural net
FATHMM-XF [12]	Clinical effect	Classification	Evolutionary, database lookups	Hidden Markov Model
CADD [16]	Clinical effect	Regression	Database lookups, consensus method	Logistic regression
DANN [15]	Clinical effect	Classification	Database lookups, consensus method	Deep neural net
Dehiya et al. [159]	Clinical effect	Classification	Evolutionary, structural	Random forest
mCSM family [161–164]	Protein stability	Regression	Structural	Supervised machine learning method*
Envision [14]	Protein function	Regression	Evolutionary, structural	Stochastic gradient boosting
CUPSAT [5]	Protein stability	Regression	Structural	Boltzmann' energy calculation
M-CAP [160]	Clinical effect	Classification	Evolutionary, consensus method	Gradient boosting tree
MutationAssesor [165]	Clinical effect	Classification	Evolutionary	Conservation-based
MutationTaster2 [166]	Clinical effect	Classification	Evolutionary	Naive Bayes

Table 7.1: *A collection of methods for prediction of effects of genetic variants. This table is not exhaustive since there are too many methods in the field. in green: machine learning method, in red: method not related to machine learning; *not further specified*

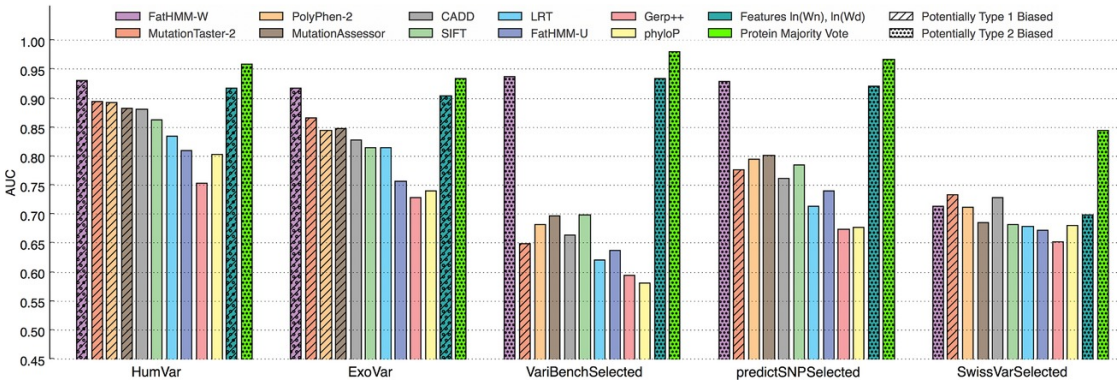


Figure 7.1: *Evaluation of the 10 different pathogenicity prediction tools (by AUC) over five datasets. The hatched bars indicate potentially biased results, due to the overlap (or possible overlap) between the evaluation data and the data used (by tool developers) for training the prediction tool. The dotted bars indicate that the tool is biased due to type 2 circularity. (taken with permission from (Grimm et al., 2015) [167])*

the same label for all variants from the same protein, hence the model learns to directly associate proteins with labels. This leads to misclassifications for variants of pure proteins with a diverging label. Additionally, models trained in that way do not perform well on proteins that are not part of the training dataset. Type 2 circularity typically leads to inflated prediction performances for test sets with variants that include variants in the protein that are also present in the training set and that carry the same label as all other variants in those proteins. While higher accuracy values sound very tempting, the trained model will fail on other datasets, which contains proteins with variants with mixed labels and variants from proteins, which are not covered in the training set (difficult variants).

In addition to the prediction of clinical effects, another scenario, the prediction of the impact of a genetic variant on the function of a protein can also be addressed by computational tools. Whereas a clinical effect can be caused by the combined effect of many genetic variants and other causes (e.g. cell-type-specific protein expression levels), the change of the function of a protein is most commonly mediated by a singule nsSNV. Thus the relationship between mutation and effect is much closer when predicting the impact on the function of a protein.

The mCSM-family [161–164] contains a set of prediction methods sharing the same framework that are specialized for specific scenarios, e.g. mCSM-lig [162] for ligand interactions, or mCSM-PPI2 [164] for protein-protein interactions. Their framework calculates so-called graph-signatures from structural data, which represent chemical properties of the environment around the residue and the change in chemical properties between wildtype and mutant residue. These signatures are then used as features in a machine learning scenario.

Besides machine learning methods, physics-based methods that aim to calculate or simulate the effect directly are widespread. Very often such tools are tailored for specific effects, for example, a lot of methods predict the change in protein stability introduced by a mutation: CUPSAT [5], PoPMuSiC [177], STRUM [178], and INPS [179]. A similar example is the prediction of the impact of a mutation on the protein-protein binding affinity: BeAtMuSiC [180], MutaBind [181], BindProfX [182], and ELASPIC [183]. One could also use a method, which calculates the binding energy of any structure (e.g. FoldX [141]) to process the wildtype structure and the mutated structure and calculate the difference in folding energy. In any case, one needs structural information to use these methods and they are in general computationally expensive, which makes it hard to process larger datasets.

Experimental high-throughput assays that are able to measure the effects on specific protein functions for massive amounts of (or even all) mutations from a particular protein are called deep mutational scans (DMS) [82]. DMS experiments, which provide a measure of the impact of nearly every possible mutations in nearly every position of a protein on its function (see Section 3.1.3), offer a new source of training data, which can be used to build models for the prediction of impact of variants on protein function that are not limited to a specific scenario or type of function. The method, that was developed in order to harness the results from DMS experiments is called Envision [14] and is able

to predict the functional effects of individual variants well for some proteins but not so well for other proteins (Figure 7.2). This leaves an open question if one can improve the model or one simply needs more training data in order to further improve the performance of the methods in this field.
DMS data has the highest density of mutations with measured effect per protein. Analyzing this type of data provides an opportunity to understand the contributions of each individual amino acid to the specific function of the protein. A large amount of annotated mutations for a very limited amount of proteins presents the major challenge for creating models that are supposed to be transferable to other proteins.
Another dangerous pitfall are the biases introduced by having multiple amino acid substitutions for the same position. On the one hand, some residues are very important for the measured function and a substitution to any amino acid is harmful. On the other hand, some residues are exchangeable for all other amino acid types without having any impact on the measured function. As a consequence, there are many samples with a lot of similar features, namely the features that depend on the amino acid position, but not on the type of the mutant amino acid. The features for these samples differ only for features that are dependent on the mutant amino acid (see Section 7.2.3). These samples can have different labels, especially for functionally important residues. Samples with different labels and similar features have the potential to confuse any machine learning method.

7.2 METHODS

7.2.1 *Deep Mutational Scans*

The DMS dataset used in this study was taken from Gray et al. [184]. In their study, they collected DMS data for 14 different proteins (Figure 7.3) and introduced statistical normalization methods to measure the corresponding effects on protein function, such that the results from the different experiments can be compared. All values are transformed in a way, such that neutral mutations are assigned an effect value equal to one. The lower the assigned effect value is, the more harmful the mutation is with respect to the measured function. One should note that zero is not the minimal value, the effect values can be (slightly) negative as well, which is an artifact of the normalization process . Further, values above one are possible, denoting mutations that increase the measured the function (e.g. measured reaction turnover rate).

7.2.2 *Genetic Variant Databases with Association to Pathogenic Phenotypes*

For the prediction of the clinical effect of an nsSNV, we use multiple databases that contain variants with binary labels for their clinical effect: benign or deleterious (Table 7.2). As our main resource, we used ClinVar (Landrum et al., 2016). Additionally, following Grimm et al. [167], we evaluated our method on the very same datasets as used in the paper (Figure 7.1). An important property of a dataset of variants with annotated clinical effect is the proportion

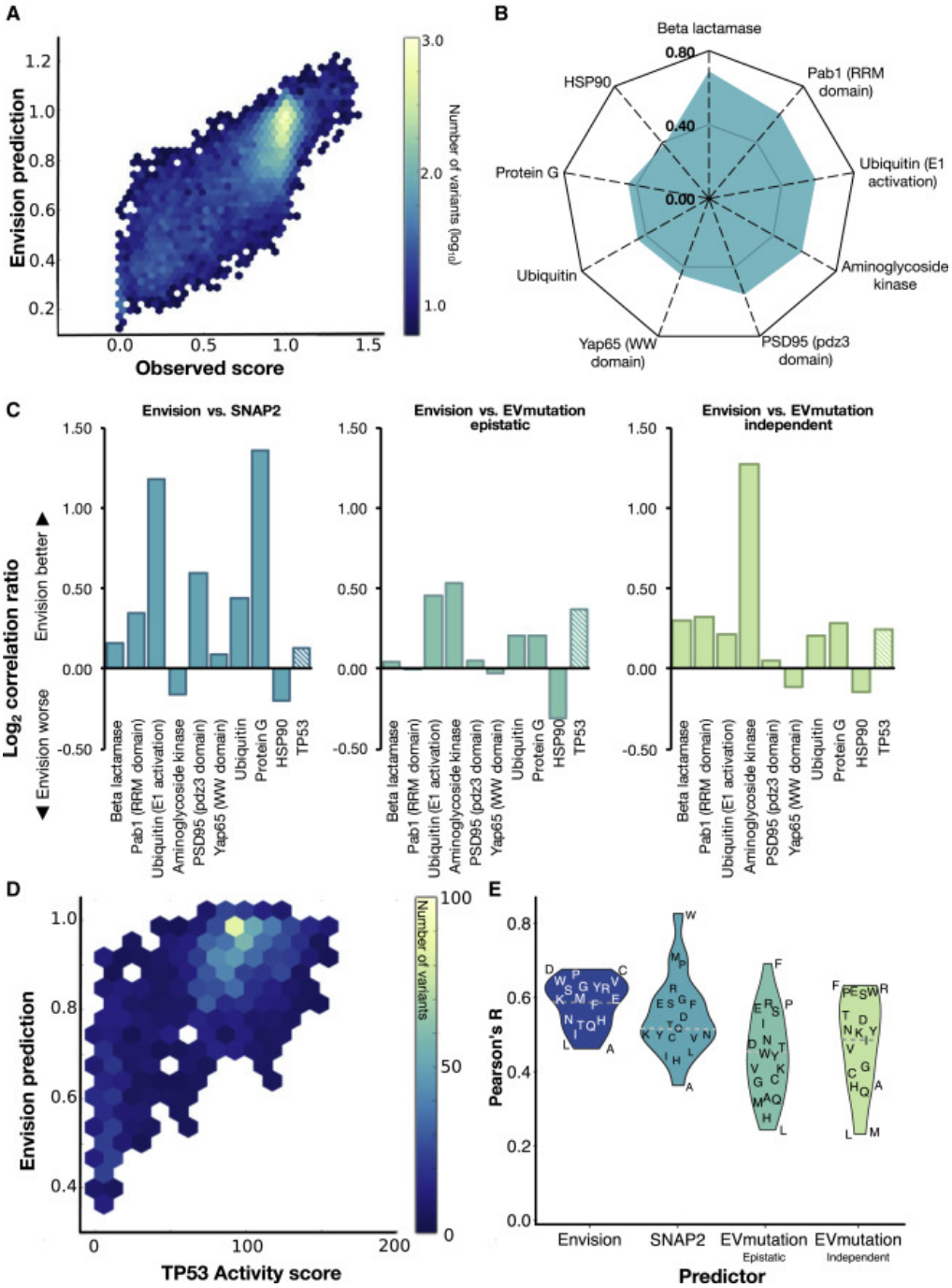


Figure 7.2: *Envision Outperforms Other Quantitative Variant Effect Predictors. (A) A hexagonal bin plot shows the correlation between predicted and observed variant effect scores for all the large-scale mutagenesis data used to train Envision (Pearson's R = 0.79). (B) To evaluate performance on data not used in training, we retrained models excluding each one of the nine proteins. A radar plot shows the correlation (Pearson's R) between predicted and observed variant effect scores when the indicated protein was left out. (C) We also compared the leave-one-protein-out (LOPO) models with SNAP2 (left panel), EVmutation-epistatic (middle panel), and EVmutation-independent (right panel). The log2 ratio of each LOPO model's Pearson's R to another predictor Pearson's R on the left-out data is shown. Hashed bars on the right indicate relative performance on a set of 2,312 TP53 transactivation activity scores measured in a low-throughput assay and not used in training. (D) A hexagonal bin plot shows the correlation between Envision predictions and TP53 activity scores (Pearson's R = 0.58). (E) A violin plot illustrates the distribution of Pearson's correlation coefficients for variant effect scores and Envision, SNAP2, and EVmutation predictions for different mutant amino acids. The dashed horizontal line indicates the median Pearson's correlation coefficients for each predictor. (taken with permission from (Gray et al., 2018) [14])*

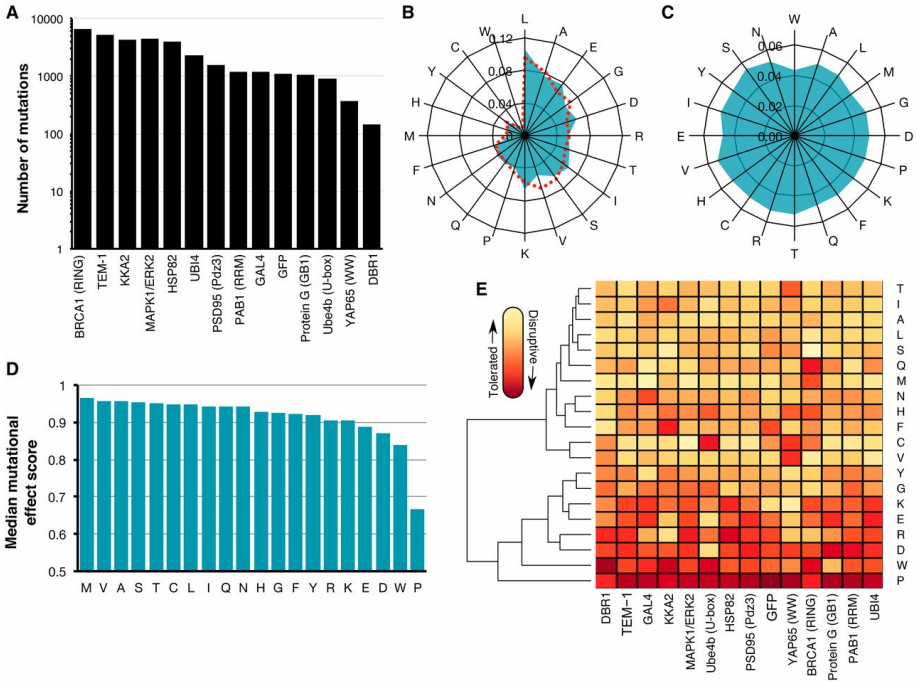


Figure 7.3: *Large-scale mutagenesis data from* 14 *proteins. (A) The number of single amino acid mutations with effect scores in each of the* 14 *proteins is shown. (B) A radar plot shows the relative frequency of occurrence of each amino acid in the wild-type sequences of the* 14 *proteins (blue) or in* 554,515 *proteins in the UniProt Knowledgebase (Magrane and UniProt Consortium 2011) (dashed red). (C) A radar plot shows the relative frequency of each of the* 20 *amino acid substitutions in the large-scale mutagenesis data sets for all* 14 *proteins. (D) The median mutational effect score of each amino acid substitution is shown for* 34,373 *mutations at* 2236 *positions in all* 14 *proteins. (E) A heat map shows the median mutational effect score of each amino acid substitution for each protein separately. Yellow indicates tolerated substitutions while orange indicates disruptive substitutions. Amino acids and proteins were ordered according to similarity using hierarchical clustering with the hclust function from the heatmap2 package in R. (taken with permission from (Gray et al., 2017) [184])*

Dataset	#Proteins	#Variants	#Benign variants	#Deleterious variants	Proportion of pure proteins
ClinVar [63]	6085	84991	69844	15147	55.6%
HumVar [11]	9219	33110	17914	15096	91.4%
ExoVar [168]	3589	6876	3437	3439	95.7%
VariBench [169]	4193	9443	5718	3725	98.5%
predictSNP [171]	391	737	690	47	95.3%
SwissVar [185]	4476	9897	6423	3474	90.3%
Union	14368	141209	103744	37465	78.3%

Table 7.2: *Composition of the six datasets.*

of so-called pure proteins. All variants of a pure protein have the same label in the dataset, either all benign or all deleterious. Non-pure or mixed proteins have variants of both types of labels in the dataset.

In our work, we investigated the possibility to avoid type 2 circularity that can artificially inflate methods' performance, as discussed in Section 7.1.1. An easy technique to avoid type 2 circularity is retaining only a part of the training set, such that each protein contains multiple variants and variants of the same proteins have mixed labels. However, this technique can decrease the total number of data points quite drastically (discussed in more detail in Section 7.2.4.2).
There are features, which have a stronger effect on the introduction of type 2 circularity than others and we can distinguish between five types of features with the potential to introduce type 2 circularity:

1. Protein-specific features (i.e. features that have the same value for all variants in the same protein) that cannot be computed for proteins not present in the training set. Such features are the results from calculations based on all samples in the training set, for example, the mean target value of all variants from the same protein in the training set.

2. Protein-specific features that can be calculated in any case, for example, if the protein is listed in the OMIM [55] database or the number of amino acids.

3. Features that can only be computed for some proteins. Into that category fall all structural features.

4. Features that are identical for parts of a proteins, for example, domain annotation.

5. Features that correlate for variants of the same proteins, for example, the position of the mutation in the protein sequence.

The listed categories are ordered by the descrease of their potential to introduce type 2 circularity. Currently, we exclude features from the first and second categories from our model. The full list of features we are using is given in the following section.

7.2.3 *Feature Generation*

For each nsSNV, we generated three different types of features.

1. Evolutionary features
2. Amino acid property features
3. Structural features

7.2.3.1 *Evolutionary Features*

Evolutionary features are based on the fact that different positions in a protein experience different evolutionary pressure. Residues with important and specific functional roles are more conserved through the evolutionary history than residues that could be easily replaced [49–51]. This results in the fact that proteins from different species that at one point had a common ancestor, have some residues that were replaced and some that were kept identical. By the alignment of many evolutionarily related protein sequences, one can calculate different statistical measures that estimate the evolutionary pressure for each position. We call such statistical measures evolutionary features. Since these are widely used in the field of variant impact prediction, we reproduced the features that other tools had already established. For example, we estimated the frequency of every amino acid in the position in a multiple sequence alignment (MSA), which is constructed for a list of evolutionary related protein sequences (the exact procedure for construction of the MSA is described below). Another measure for evolutionary pressure is the position-specific independent counts (PSIC) measure [158]. In comparison to the frequency of an amino acid, PSIC weights the importance of each sequence in the MSA and uses these weights to estimate how likely it is for the amino acid to be present at the position by chance. dPSIC is the PSIC value of the wildtype amino acid subtracted by the PSIC value of the mutant amino acid.
There is an important difference in the generation of our evolutionary features compared to other tools in the field: instead of constructing a real MSA of the selected evolutionary related sequences, as it is the usual practice, we calculated a list of pairwise alignments between the sequence of the target protein and each related sequence and stacked them on top of each other throwing parts away that are not aligned to the target protein (Figure 7.4). We call the resulting stacking a pseudo multiple sequence alignment (PMSA). The list of related proteins was compiled by using MMseqs2 [86] for a sequence similarity search against the UniRef90 [186] sequence database. In the resulting PMSA, we calculated different types of evolutionary features (see Table 7.3).

7.2.3.2 *Amino Acid Property Features*

Amino acid property features are based on the physico-chemical properties of the wild type amino acid, the mutant amino acid, and the differences of these properties. We used a list of amino acid classes (Table 7.4) based on the physicochemical classification of amino acids by Livingstone and Barton [187] (Figure 7.5). We derived one binary feature for each class, whether the mutant

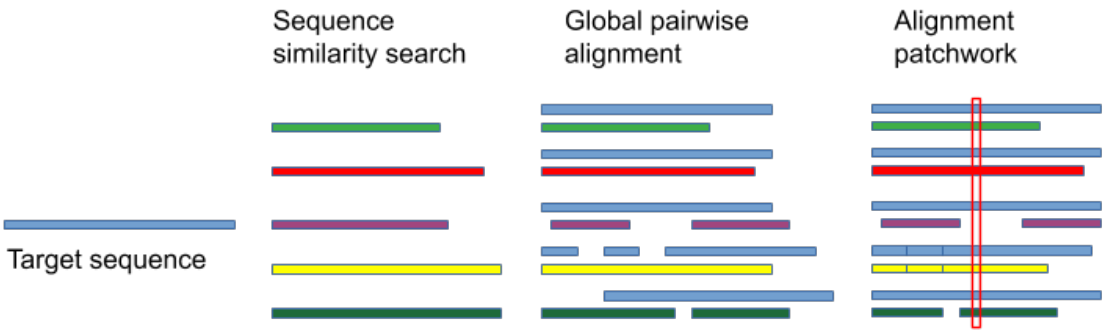


Figure 7.4: *The three steps of the simplified multiple sequence alignment technique; each colored bar represents a sequence, the blue sequence is the target sequence; in the first step a sequence similarity search returns a list of similar sequences (hits); in the second step all hits are pairwise aligned to the target sequence; in the third step the segments of the hits, which got not aligned to the target sequence are thrown away, all pairwise alignments are stacked similar to a multiple sequence alignment using the target sequence as reference.*

Name	Description
Wildtype AA conservation	Frequency of the wildtype amino acid in the PMSA
Mutant AA conservation	Frequency of the mutant amino acid in the PMSA
Other AA conservation	Frequency of amino acids in the PMSA that is not the wildtype or mutant amino acid
Wildtype AA conservation gapless	Frequency of the wildtype amino acid in the PMSA where sequences that match a gap to the position are ignored
Mutant AA conservation gapless	Frequency of the mutant amino acid in the PMSA, where sequences that match a gap to the position are ignored
Other AA conservation gapless	Frequency of amino acids in the PMSA that is not the wildtype or mutant amino acid, where sequences that match a gap to the position are ignored
MSA frequency	Frequency of non-gap letters in the PMSA
PSIC wildtype AA	PSIC value of the wildtype AA in the PMSA
PSIC mutant AA	PSIC value of the mutant AA in the PMSA
dPSIC	PSIC wildtype AA minus PSIC mutant AA

Table 7.3: *List of evolutionary features.*

Class name	List of amino acids in the class
Tiny	G, A, C, S
Small	G, A, C, S, V, T, D, N, P
Aliphatic	I, L, V
Aromatic	F, Y, W, H
Hydrophobic	G, A, C, T, I, L, V, M, F, Y, W, H, K
Positive charged	H, K, R
Negative charged	D, E
Charged	H, K, R, D, E
Polar	C, S, N, T, Q, D, E, K, H, R, Y, W

Table 7.4: *Amino acid classes as described in Livingstone and Barton (1993) [187]*

amino acid is in the same class as the wildtype amino acid or not. Other features are numerical values, e.g. the difference in the volume of the wildtype and mutant amino acid. All amino acid property features are either table lookups or simple calculations done with values from table lookups and are listed them with a short description in Table 7.5.

7.2.3.3 *Structural Features*

Since one of the main goals of this study is to evaluate if protein structure-based features produced by StructMAn can improve variant impact prediction methods, the most interesting features in this project are the structural features (Table 7.6). The generation of the structural features is done by StructMAn in the structural analysis and classification steps and is in detail explained in Chapter 5.

7.2.4 *Parameters of Training and Evaluation*

7.2.4.1 *Deep Mutational Scan Dataset*

The DMS dataset is characterized by its high variant-to-protein ratio. In order to ensure that the model is able to predict the functional impacts of variants on proteins unknown to the model, one has to evaluate its performance in exactly these cases. As it is suggested in the original Envision paper [184], which was created for predicting functional impact of variants from the DMS data, we used the Leave-One-Protein-Out (LOPO) cross-validation, in which in each round all variants belonging to one protein are separated from the dataset in order to create the test set. The remaining samples are used to fit the model.
The hyperparameter optimization of the tree pruning is solely done for the first round of LOPO, which uses the variants from the aminoglycoside kinase as the test set. The pruning parameters obtained from that round are then used in all other rounds without any further optimization.

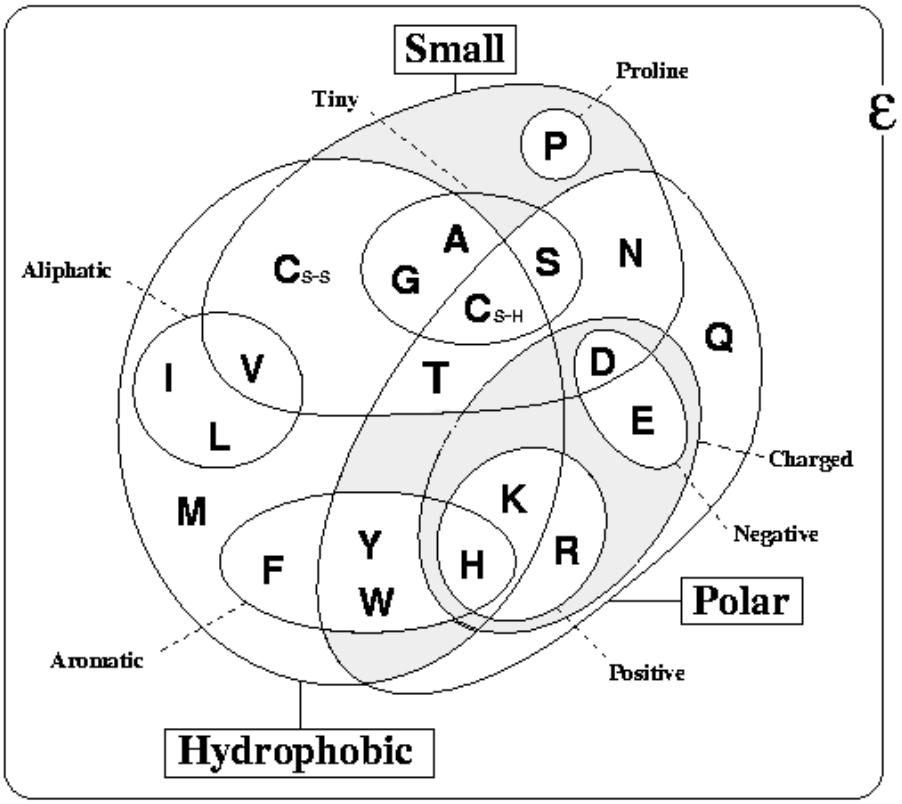


Figure 7.5: *Physico-chemical classes of the amino acids. (taken with permission from (Livingstone and Barton, 1993) [187])*

Name	Description
KD mean	Mean Kyte-Doolittle hydrophobicity score of mutant and wildtype amino acid
Volume mean	Mean total van-der-Waals volume of mutant and wildtype amino acid
Chemical distance	Euclidean distance of chemical descriptor vector [188] of mutant and wildtype amino acid
Blosum62	Value for the amino acid substitution in the Blosum62 [88] substitution matrix
Aliphatic change	0, if both amino acids are in the same same aliphatic class, 1 otherwise
Hydrophobic change	0, if both amino acids are in the same hydrophobic class, 1 otherwise
Aromatic change	0, if both amino acids are in the same aromatic class, 1 otherwise
Positive charged change	0, if both amino acids are in the same positive charged class, 1 otherwise
Negative charged change	0, if both amino acids are in the same negative charged class, 1 otherwise
Polar change	0, if both amino acids are in the same polar class, 1 otherwise
Charged change	0, if both amino acids are in the same charged class, 1 otherwise
Small change	0, if both amino acids are in the same small class, 1 otherwise
Tiny change	0, if both amino acids are in the same tiny class, 1 otherwise
Total change	Sum of all amino acid class change features
IUPred value	Score [137] predicted by IUPred for the residue
Region structure type	IUpred prediction for the residue to be in a globular or disordered region
Wildtype AA (21 features)	Wildtype amino acid type, including 'unknown type', one feature for each type
Mutant AA (21 features)	Mutant amino acid type, including 'unknown type', one feature for each type
AA change (420 features)	The amino acid substitution, one feature for each amino acid combination

Table 7.5: *List of amino acid property features.*

Name	Description
Distance-based classification	Distance-based structural classification by StructMAn
Distance-based simple classification	Distance-based structural simple classification by StructMAn
RIN-based classification	RIN-based structural classification by StructMAn
RIN-based simple classification	RIN-based structural simple classification by StructMAn
Structure location	Surface/Core annotation by StructMAn
Secondary structure assignment	Type of secondary structure the residue belongs to, weighted majority vote over all mapped structures
Modres score	The negative sum of quality scores of mapped structures, where the mapped residue is not posttranslationally modified, plus the sum of quality scores of mapped structures, where the mapped residue is a posttranslationally modified
Modres probability	Weighted frequency of the mapped residue being a modified residue
B Factor	Weighted B factor values over all mapped structures
Centrality	Weighted size-normalized centrality scores of mapped residue vertices in all RINs
[Amino acid part] [interaction type] score (20 features)	Weighted combined probe scores of mapped residue vertices in all RINs, one feature for each combination of amino acid part and interaction type
[Amino acid part] [interaction type] degree (20 features)	Weighted vertex degree of mapped residue vertices in all RINs, one feature for each combination of amino acid part and interaction type
[Amino acid part] [interaction type] H-bond score (20 features)	Weighted H-bond probe scores of mapped residue vertices in all RINs, one feature for each combination of amino acid part and interaction type
Interactions	1, if a particular interaction was annotated in at least one mapped structure, 0, otherwise one feature per type of interaction

Table 7.6: *List of structural features. Amino acid part: side chain or main chain. Interaction type: neighboring, short, and long intrachain interactions and protein, DNA, RNA, metal, and ion interchain interactions.*

7.2.4.2 *ClinVar Dataset*

The evaluations performed on the ClinVar dataset are all ten-fold cross-validations. Similar to the evaluation of the model on the DMS dataset, the first round of cross-validation is used for the hyperparameter optimization.
We do not use any protein-specific feature *per se*, nevertheless, we want to analyze the influence of type 2 circularity on the model introduced by the training procedure. Thus, we introduced an artificial feature, which is associated with type 2 circularity by design. This is done in a similar fashion to Grimm et al. [167] by calculating the mean labels for every protein in the training set and use them as an additional feature. When the model predicts the outcome of a sample and this mean value is not known (this happens, when the protein of the sample is not present in the training set), 0.5 is used. We call this feature protein bias, and while it can only be computed from the information the training set offers, which means it does not offend the basic rules of evaluation of prediction models, we do not suggest to use such a feature in any real applications.
In order to construct the ten cross-validation training and test set pairs, we used two different randomization schemes (Figure 7.6). In the variant-based randomization, we randomly dissect the full dataset into ten approximately equally sized parts and use one of the parts as the test set, while the other nine are used as the training set. Thus, the variant-based randomization represents the classical way of constructing a cross-validation setup.
Additionally, we introduce a different training scenario, the protein-based randomization, where we also construct ten approximately equally sized dataset parts, but ensuring that there is no protein that appears in more than one part. This is achieved by randomly selecting proteins from the dataset and putting them into ten bins, such that the amount of variants in each bin is roughly equal. This way of randomization enables the evaluation of our model in a scenario, in which we aim to predict the effect of variants from proteins previously not seen by the model. A model performing well in such a scenario should also perform well in the prediction of variants of pure proteins with non-identical labels within the same protein. Thus, protein-based randomization evaluates the model for its ability to handle difficult variants (variants in novel proteins and variants with different effect than those of known variants from the same protein). Ideally, we want our model to perform equally well in both randomization scenarios.
Since the focus of this project is on the utility of structural features, we tested all setups also for a subset of the dataset, where we filtered out all samples, for which we could not generate the structural features, due to the lack of structural data.
To estimate the influence of type 2 circularity from differently designed training datasets, we used a set of filtering techniques that are designed to reduce protein-specific biases:

1. Leaving out of proteins with only one variant (singleton filtering)
2. Leaving out of proteins, for which all variants share the label (pure proteins filtering)

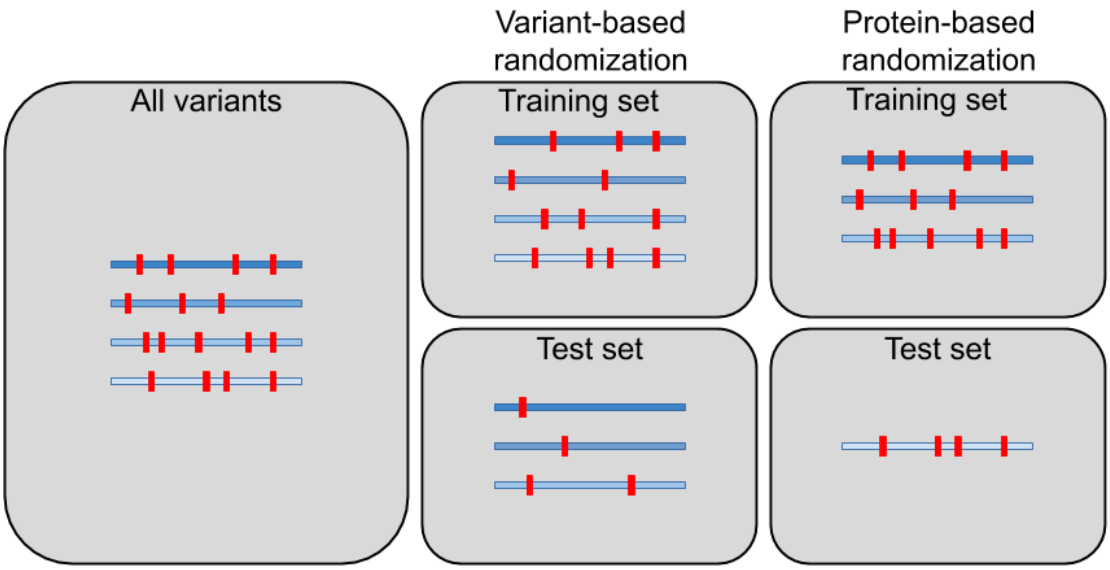


Figure 7.6: *The two randomization schemes. Protein sequences in the datasets are depicted as blue rectangles of different shades, variants in the datasets are depicted as red marks in each protein. In the Variant-based randomization the dataset is split by selecting variants, in the protein-based randomization the dataset is split by selecting proteins.*

3. Leaving out of variants, such that for all proteins the number of variants labeled as benign is equal to the number of variants labeled as deleterious (balanced filtering)

7.2.4.3 *Benchmark Datasets*

The general setup for the training and evaluation of our model for the five benchmark datasets from Grimm et al. [167] is a Leave-One-Dataset-Out (LODO) cross-validation. Since the datasets share some samples, in all setups and in each validation round we filtered out all variants from the respective training set, which are present in the test set. This filtering removes type 1 circularity from all setups. This filtering does not include different variants from the same protein. In order to identify the influence of type 2 circularity, we repeated all setups with filtering all proteins from the training set, if they are present in the test set.
Similar to the cross-validation on the ClinVar dataset, we also tried setups, with and without the introduction of the protein bias feature as well as with and without the filtering of samples with available structural data. In order to analyze the effect of having an increased amount of training samples, we repeated the previously described setups, additionally including the ClinVar dataset into the training set of each validation round.

7.2.5 *Random Forest Classifier and Regressor*

The random forests used in this thesis comprises 100 random decision trees. The amount of pruning is based on a hyperparameter optimization step, by calculating the testing error on unseen samples for different magnitudes of

Uniprot Access	Protein Name	MSE	Correlation
P00552	Aminoglycoside kinase NEO	0.1400	0.5448
P46937	Transcriptional coactivator YAP1	0.0634	0.5963
Q9UK59	Lariat debranching enzyme DBR1	0.1486	0.4161
P38398	Breast cancer type 1 susceptibility protein BRCA1	0.802	0.3404
P02829	ATP-dependent molecular chaperone HSP82	0.0767	0.4140
Q9ES00	Ubiquitin conjugation factor E4 B	0.0575	0.4049
P42212	Green fluorescent protein GFP	0.0887	0.2751
P0CG63	Polyubiquitin UBI4	0.0382	0.5617
P62593	β-lactamase TEM1	0.0633	0.6583
P31016	Disks large homolog 4 DLG4	0.0450	0.4738
P04386	Regulatory protein GAL4	0.1443	0.3522
P04147	Polyadenylate-binding protein PAB1	0.0481	0.5638
P0CG48	Polyubiquitin-C UBC	0.0554	0.6389
P06654	Immunoglobulin G-binding protein G	0.0863	0.3280

Table 7.7: *Results of the LOPO-cross-validation. For each protein the mean squared error (MSE) and the Spearman's correlation values are listed.*

pruning.
For the prediction of the impact of mutations on protein function, we used a normalized DMS dataset, in which the effect is reported as a numeric value. Hence, we used a random forest regressor, which in the model training minimizes the Mean Squared Error (MSE) of the training set. As model performance measures, we use the MSE of the test set as well as Pearson's correlation coefficient between the predicted values of the test set and the true values.
The other scenario, the prediction of the clinical effect of nsSNVs, is a binary classification problem. Here, the random forest maximizes the prediction accuracy of the training set and we report the area under the Receiver-Operator-Characteristics curve (auROC) of the predictions for the test set as the performance measure of the model.

7.3 RESULTS

7.3.1 *Prediction of Functional Impact of Genetic Variants for the DMS Dataset*

The results of the LOPO cross-validation are listed in Table 7.7 and show that the prediction performances for different proteins vary a lot, similar to the evaluation of Envision reported in Gray et al. (2018) [14] (Figure 7.2). The best results were achieved for UBI4, which can easily be explained by the presence of the results from a DMS on UBC, a paralog to UBI4, in the training set. Vice versa, this also led to good results in the UBC round of the LOPO cross-validation. However, the varying results for the other proteins lack such an easy explanation.

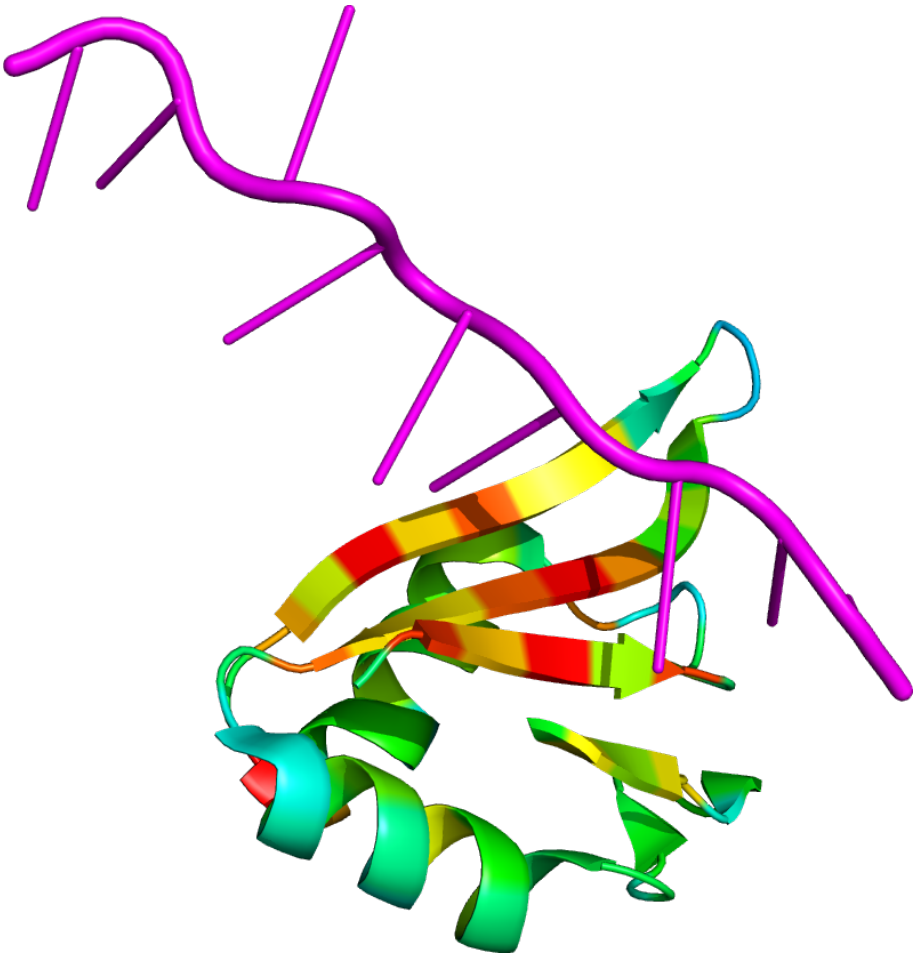


Figure 7.7: *Maximum error projection of the predictions for PAB1 (P04147) (PDB id 4F02, chain A). Maximum error values represented as rainbow color-scale, red:* 1.0 *maximum error, blue:* 0.0 *maximum error. The RNA chain is shown in magenta.*

7.3.1.1 *Maximum Error Projections*

In order to get a more in-depth view of the prediction for individual residues for each protein, we mapped the prediction performances measured as maximum prediction error for any variant at each protein position on the corresponding protein structures. StructMAn provides for each given position a structure recommendation (structure with maximum quality score for all structures, where the mapped residue has the same structural classification). Since these recommendations are position-specific, we get multiple recommended structures for one protein. In order to visualize all variants in a single structure for this analysis, we chose the structure, which was recommended for the largest proportion of positions of a protein. We colored the protein residues in the rainbow color scale, where red corresponds to positions with high maximum error values and blue corresponds to positions with low maximum error values. We did not visualize the segments of the structures, for which the DMS datasets lacked values.
One of the highest correlation of predictions with experimental values was achieved for PAB1, for which we mapped the maximum error values into the

structure with PDB id 4F02, where the protein is in complex with an RNA strand (Figure 7.7). Strikingly, positions with high error values are almost exclusively on the RNA interaction interface. Thus, in this example, the prediction errors coincide with the residues one expects to be functionally important based on the observation of the structure. While this is expected since these residues are the ones with the greatest variance in experimental values, this also means that the model was not able to predict the correct values for the different mutant amino acids of these residues.
An example with a slightly worse performance is GFP, which we mapped to the structure with PDB id 6JGJ. Here one can see an interesting pattern of an alternation of low and high errors on the single β-strands around the β-barrel structure (Figure 7.8). If one zooms into the most striking spot of this pattern (Figure 7.9), one can see that the sidechains of the residues with low error values point outwards the barrel, and the sidechains of the residues with high error values point to the center of the barrel and interact with the chromophore.

A third example, BRCA1 as seen in Figure 7.10, shows again that low and high error values can again be interpreted in terms of their structural location. This time, it seems that the low error values are concentrated on the α-helices of the structures, while the high error values coincide with the loop regions. Investigating a bit deeper into the structure reveals that the loops with the highest error values are the more flexible ones.
In Figure 7.11 we compare the mapping of the error values for the two paralogs UBC and UBI4 on the same structure. The paralogy of UBC and UBI4 means in the respective training rounds for the two proteins there is a very similar protein in the training set. Thus, a very good performance of the model for these two examples was expected. There are more data points for UBI4 in the dataset. Thus there are positions, for which there are no corresponding samples from UBC in the dataset and hence the effects of mutations in these position in UBI4 are more difficult for the model to predict. Consequently, in the segment, where there are no values from UBC, the error values for UBI4 are much higher. On the other hand, the positions of UBC are completely covered by data from UBI4 and thus there are basically no high error positions.
In the last example, we took a look at a protein for which the prediction was rather successful. For the β-lactamase TEM1 (Figure 7.12), the majority of the residues show low error values, with the residues with higher errors are clearly clustering in the core of the protein. Again, these are the positions with higher variance in experimental values. Mutations introduced to the core residues of TEM1 presumably decrease the stability of the protein, whereas the other residues can be substituted without such harmful effect.
Overall the investigation of the individual maximum error projections reveals that positions, where the prediction was less successful were the functional important positions. Interestingly these can be identified in a structural investigation and this leads to the question of why the model was not able to correctly predict the effects for these positions, while it has access to many structural features. Does the random forest rely on protein structure-based features in its prediction in any way?

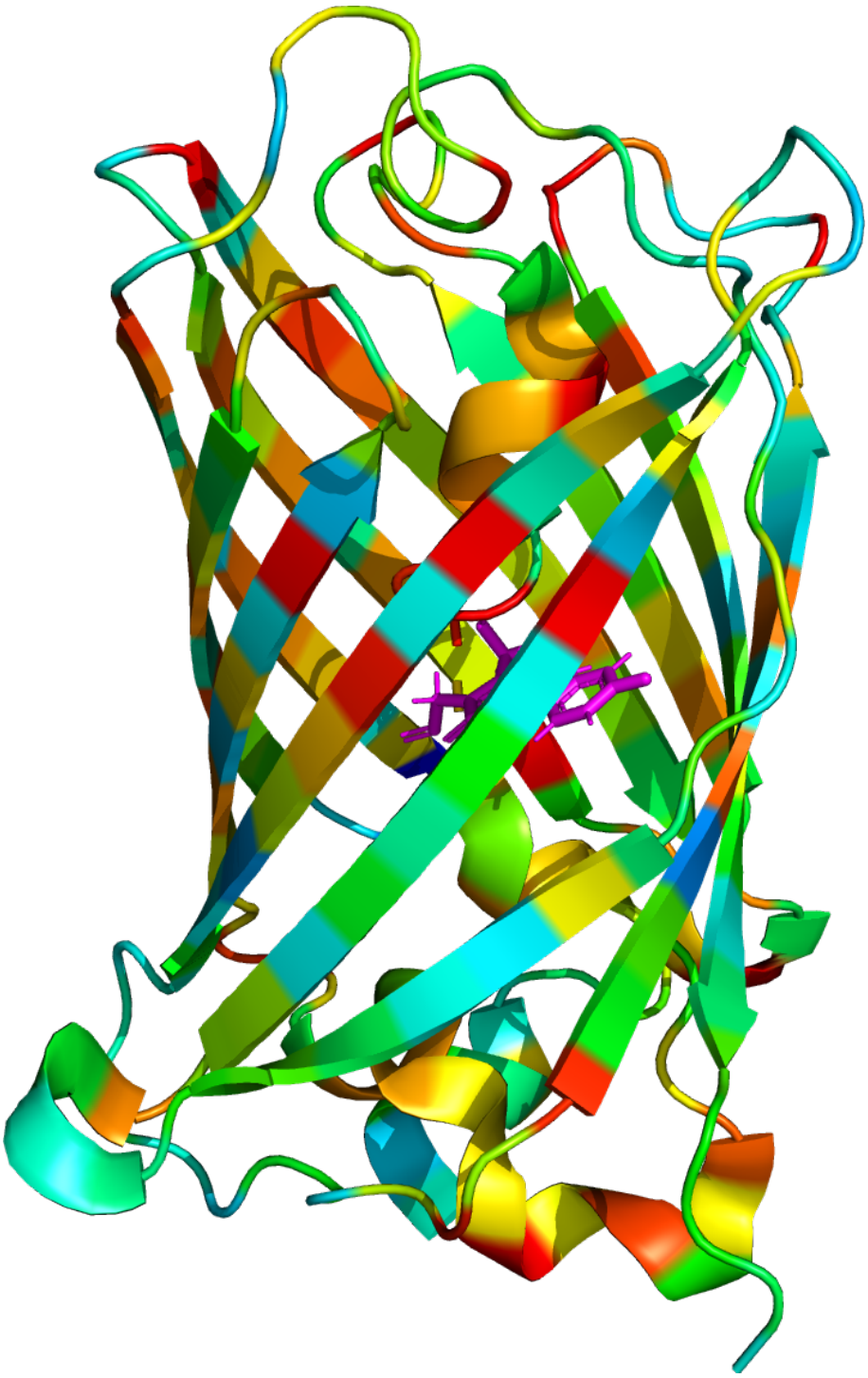


Figure 7.8: *Maximum error projection of the predictions for GFP (P42212) (PDB id 6JGJ, chain A.). The chromophore is shown in magenta.*

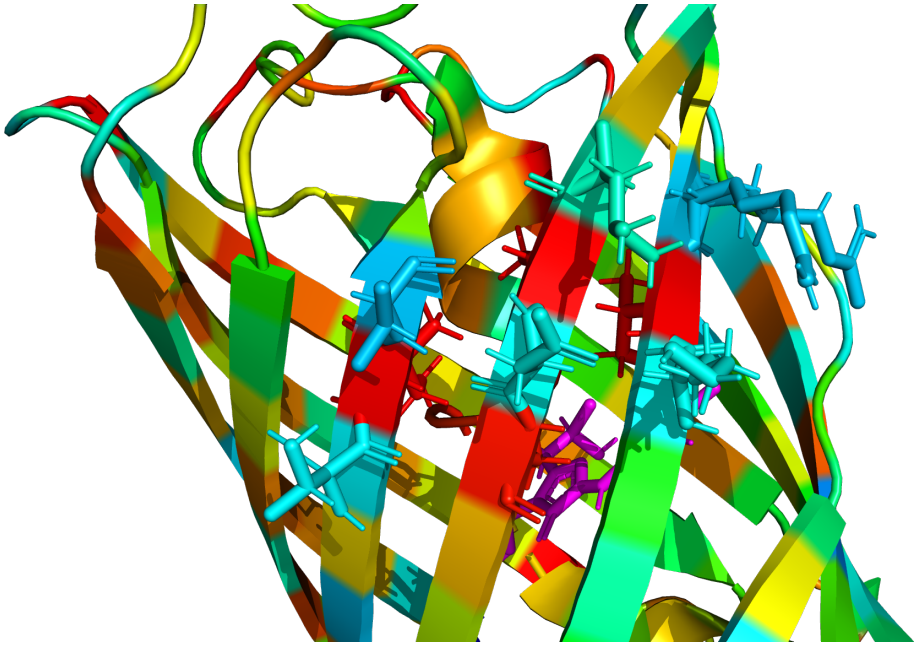


Figure 7.9: *Maximum error projection of the prediction for GFP (P42212) (PDB id 6JGJ, chain A); key residues of the alternation pattern are depicted in stick representation. The chromophore is shown in magenta.*

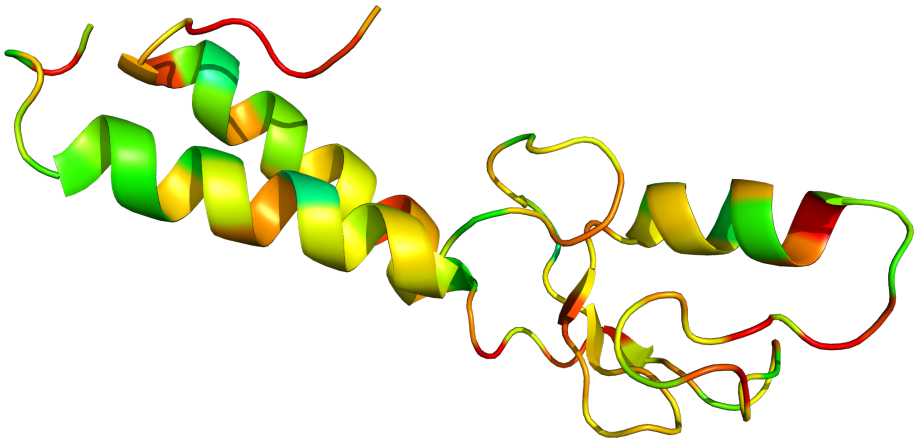


Figure 7.10: *Maximum error projection of the prediction for BRCA1 (P38398) (PDB id 1JM7, chain A.)*

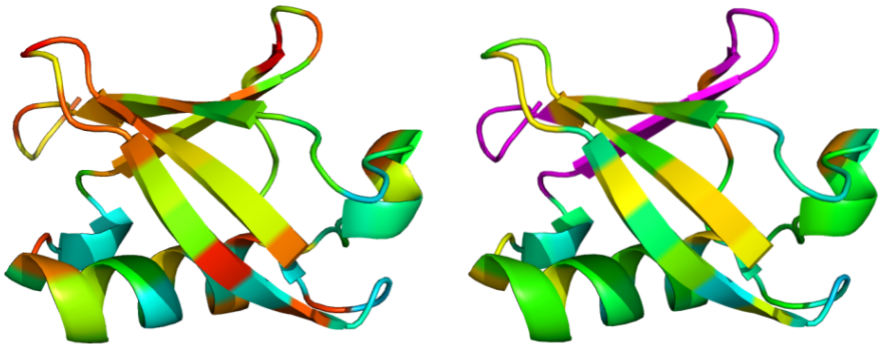


Figure 7.11: *Left structure: Maximum error projection of the prediction for UBI4 (PoCG48) (PDB id 5VIX, chain A); Right structure: Maximum error projection of the prediction for UBC (PoCG63) (PDB id 5VIX); the magenta colored segment is not part of the UBC dataset.*

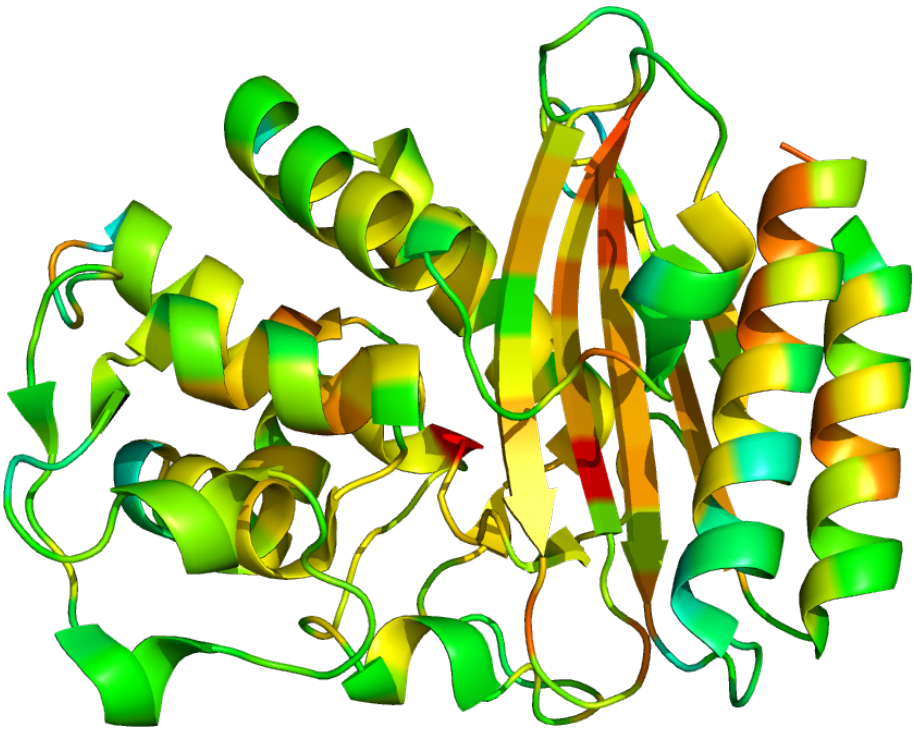


Figure 7.12: *Maximum error projection of the prediction for TEM1 (P62593) (PDB id 1M40, chain A).*

Feature name	Feature type	Feature Importance
dPSIC	Evolutionary	0.198
Frequency of mutant AA	Evolutionary	0.073
Structure location	Structural	0.054
B-factor	Structural	0.045
WT AA conservation	Evolutionary	0.041
Modres score	Structural	0.040
Chemical distance	AA property	0.030
Unknown mutant AA	AA property	0.026
Blosum62 score	AA property	0.022
WT AA match conservation	Evolutionary	0.021
Sidechain long score	Structural	0.020
PSIC mutant AA	Evolutionary	0.018
RIN centrality score	Structural	0.017
Other AA conservation	Evolutionary	0.016
Mutant AA match conservation	Evolutionary	0.016
PSIC WT AA	Evolutionary	0.014
Volume mean	AA property	0.014
KD mean	AA property	0.013
Other AA match conservation	Evolutionary	0.013
Mutant AA proline	AA property	0.012
Sidechain short score	Structural	0.011
Mutant AA conservation	Evolutionary	0.011
Sidechain neighbor score	Structural	0.011
Sidechain long degree	Structural	0.010
Mainchain long score	Structural	0.010

Table 7.8: *Top list of features used by the random forest with feature importance values* $\geqslant 0.01$.

7.3.1.2 *Feature Importance Analysis*

An advantage of using random forests as the machine learning method is that they allow to calculate the individual contribution of each feature. We list the top contributing features in Table 7.8, where one can notice a moderate balance between all three major feature types with a slight advantage in favor of evolutionary features. As expected, it seems to be very predictive if a mutated residue lies on the surface of a protein or in the protein core. From some experimentally resolved structure, we can predict for some residues, if they are posttranslationally modified and the posttranslational modification status plays an important role in the model. The B-factor listed for each residue in X-ray structures is associated with the flexibility of a residue in the structure, and residues with a high B-factor are rarely important for the stability or function of the protein, helping the model in its prediction. On the other hand, residues, which are important for stability tend to have high RIN centrality scores. The model also incorporates multiple interaction scores, which reflect different types of interactions the residue participates in. As expected, to know if a residue is involved in an interaction is important for the model. One can notice that a lot of top-contributing features are probably highly correlated

		Variant-based randomization		Protein-based randomization	
		No PB	PB	No PB	PB
All variants	Without SF	0.744 ± 0.008	0.810 ± 0.011	0.723 ± 0.040	0.642 ± 0.060
	With SF	0.778 ± 0.009	0.813 ± 0.005	0.736 ± 0.039	0.654 ± 0.041
Only variants mapped to 3D structure	Without SF	0.697 ± 0.008	0.786 ± 0.010	0.677 ± 0.033	0.654 ± 0.036
	With SF	0.734 ± 0.010	0.789 ± 0.008	0.689 ± 0.026	0.663 ± 0.024

Table 7.9: *Mean auROC values (± standard deviation) for ten-fold cross-validations on ClinVar for different training setups. SF: structural features, PB: setup includes protein bias feature*

(e.g. B-factors are known to be higher on protein surface, and RIN centrality is probably higher for position buried in the protein core). Analysis of correlations between features and feature selection is a topic of our ongoing research.

7.3.2 *Assessment of the Pathogenic Potential of Genetic Variants*

7.3.2.1 *Evaluation of the Model for ClinVar*

We compared different cross-validation setups using ClinVar in order to estimate the influence of protein-specific biases and how much structural features can improve the quality of the model. In Table 7.9 we list the auROC values for 16 different training setups. When the protein bias is introduced as an additional feature, it dominates the construction of the model (as can be proven by the feature importance analysis, data not shown), thus the forest does not really change when we add or remove structural features. The inclusion of structural features leads to increased auROC values, no matter if we filter only for proteins with available experimentally resolved structures or not, in both cross-validation setups . However, this increase is stronger in the variant-based cross-validation than in the protein-based cross-validation holds true. As expected, the artificial protein bias inflates the performance of the model in the variant-based randomization setup and strongly reduces the auROC of the model in the protein-based randomization setup.
Since the inclusion of structural features improves the performance of the model more for variant-based randomization, we can assume that they introduce type 2 circularity in some way, despite not using any protein-specific information explicitely. This effect is weaker for the setups, where we filtered the samples for which we couldn't find structural data. One explanation can be: when we filter out samples, for which we do not find any structural data, the auROC values always are decreasing and the reason for that is the change in the proportion of benign variants and deleterious variants (and hence auROC is not the ideal measure to compare these particular cases). Indeed, as noted before (Chapter 6), proteins that carry a lot of known pathogenic mutations tend to be better investigated and have more three-dimensional structures resolved. Hence, in the full ClinVar dataset we have the proportion of pathogenic:benign variants equal to 1:4.93, and for the variants that can be mapped into a 3D structure this proportion is 1:3.05

Feature name	Feature type	Feature Importance
dPSIC	Evolutionary	0.267
Modres score	Structural	0.094
Wildtype AA conservation gapless	Evolutionary	0.059
B-factor	Structural	0.035
Wildtype AA conservation	Evolutionary	0.025
PSIC wildtype AA	Evolutionary	0.021
RIN centrality score	Structural	0.021
Frequency of mutant AA	Evolutionary	0.019
Mainchain neighbor score	Structural	0.018
PSIC mutant AA	Evolutionary	0.018
Sidechain short score	Structural	0.018
Mainchain short score	Structural	0.018
Sidechain neighbor score	Structural	0.017
Sidechain long score	Structural	0.017
Mainchain long score	Structural	0.016
Mainchain short H-bond score	Structural	0.015
IUPred score	AA property	0.015
Wildtype AA Cysteine	AA property	0.014
Mutant AA conservation gapless	Evolutionary	0.013
Chemical distance	AA property	0.013
Sidechain protein score	Structural	0.012
Sidechain long H-bond score	Structural	0.012
Volume mean	AA property	0.010
Other AA conservation gapless	Evolutionary	0.010

Table 7.10: *Features importances ($\geqslant$ 0.01) of the random forest, when trained on ClinVar.*

7.3.2.2 *Feature Importance Analysis*

Although, for the ClinVar dataset, we provide structural features only for a fraction of variants, the random forest includes structural features in a normal fashion (Table 7.10). Overall, the feature importances look similar to that of the prediction of functional impact scenario (Table 7.8), which provides evidence that both scenarios are related. In comparison, the evolutionary features are slightly stronger involved for the pathogenicity prediction, especially dPSIC. The amino acid property features are the type of features that are least present among the top-contributing ones. While structural features provide many equally important features, one feature, the modres score, is by far more important than the rest. The information about posttranslational modifications should only help the model for a small fraction of variants. This makes this feature suspicious. The exact contribution of this features may be more clear from the following analysis of the influence of different filtering strategies on the model performance.

	No filtering	Filter out pure proteins		Balanced labels	
		Keep singletons	Filter singletons	Keep singletons	Filter singletons
Reduction of training set	0%	10.7%	11.4%	72.7%	74.3%
Feature importance of protein bias	0.458 (rank 1)	0.453 (rank 1)	0.460 (rank 1)	0.100 (rank 3)	0.0 (last rank)
Feature importance of modres score	0.076 (rank 3)	0.087 (rank 3)	0.089 (rank 3)	0.019 (rank 12)	0.020 (rank 10)
Feature importance of RIN centrality	0.019 (rank 10)	0.018 (rank 10)	0.019 (rank 10)	0.019 (rank 11)	0.019 (rank 12)
mean auROC (VR)	0.778	0.774	0.775	0.659	0.658
mean auROC (PR)	0.736	0.727	0.735	0.653	0.649
mean auROC (VR, PB)	0.813	0.789	0.787	0.660	0.659
mean auROC (PR, PB)	0.654	0.647	0.657	0.646	0.650

Table 7.11: *Ten-fold cross-validation on ClinVar using different filtering techniques revealing bias-introducing features through a change in feature importance; feature importance values for non-protein bias features measured for setup without the protein bias feature; PR: protein-based randomization, VR: variant-based randomization, PB: protein bias features added*

7.3.2.3 *Filtering Identifies Bias-introducing Features*

We tried different filtering techniques (described in Section 7.2.4.2) on the training dataset with the goal to identify features associated with type 2 circularity and to estimate the effect of filtering on the prediction performance of the model (Table 7.11). On the one hand, filtering techniques can reduce type 2 circularity intrinsic to a dataset since this bias is mainly introduced by pure proteins. On the other hand, the reduction the number of the data points in the training set impedes the learning of a model. Thus, a model that does not rely on features that are associated with type 2 circularity does not profit from filtering. However one still can identify features associated with type 2 circularity using filtering techniques.

For ClinVar, we can see that there are only very few singletons (1%, most proteins contain more than one annotated variant), thus the effect of filtering of singletons is subtle. Filtering out of pure protein takes away about a tenth of the training samples, and the balanced filtering is more severe reducing the size of the training sets by over 70%. Unfortunately, this reduction of the number of data points in the training set directly translates into reduced mean auROC values.

Filtering out pure proteins is not sufficient to reduce the feature importance of the protein bias feature, and while the balanced labels filtering (keeping only proteins with roughly equal proportion of deleterious and benign variants)allows to reduce its importance significantly, it can only completely removed by combining balanced labels and singleton filtering. When we look into the feature importances for the modres score feature, we discover a decrease of importance after balanced labels filtering. This suggests that this feature is actually associated with type 2 circularity. After investigating the distribution of values for that particular feature across the samples, we found that its values have a strong negative correlation with the number of 3D structures where

Dataset	No bias added				Protein bias introduced			
	All samples		Only 3D		All samples		Only 3D	
	Without SF	With SF	Without SF	With SF	Without SF	With SF	Without SF	SF
SwissVar	**0.677 (0.694)**	**0.667 (0.674)**	**0.675 (0.687)**	**0.668 (0.669)**	**0.649 (0.653)**	**0.648 (0.649)**	**0.673 (0.682)**	**0.671 (0.681)**
predictSNP	**0.641 (0.653)**	**0.628 (0.635)**	0.661 (0.658)	0.656 (0.647)	**0.583 (0.606)**	**0.593 (0.601)**	**0.643 (0.675)**	**0.640 (0.670)**
ExoVar	0.802 (0.770)	0.837 (0.793)	0.744 (0.714)	0.774 (0.701)	0.882 (0.791)	0.888 (0.794)	0.841 (0.712)	0.848 (0.740)
HumVar	0.803 (0.778)	0.813 (0.792)	0.750 (0.709)	0.748 (0.711)	**0.745 (0.777)**	**0.762 (0.779)**	0.694 (0.690)	0.695 (0.690)
VariBench	**0.663 (0.671)**	0.713 (0.701)	**0.620 (0.696)**	0.637 (0.629)	0.828 (0.644)	0.832 (0.660)	0.776 (0.610)	0.784 (0.646)
Mean	0.761 (0.747)	0.777 (0.760)	0.718 (0.691)	0.722 (0.691)	0.757 (0.738)	0.768 (0.741)	0.721 (0.680)	0.724 (0.689)

Table 7.12: *auROC values for all different LODO setups using the benchmark datasets for the evaluation of the model; in parenthesis: filtered out proteins from training set that are present in the test set; in bold: protein-filtering increased the auROC value. SF:structural features, Only 3D: Only variants that could be mapped to 3D structure*

the variant can be mapped into. However, the feature importance drops not to zero for the combined balanced labels and singleton filtering, thus the feature contains valuable information and can still be used in specifically filtered training sets or in a version of the feature that is normalized by the number of mapped structures. The structural features in general can be suspected to introduce protein-specific biases since they are missing for variants that could not be mapped into 3D structures. However, the filtering techniques did not find any other structural features, whose feature importance drops.
Overall, filtering techniques were not able to improve the prediction performances, but just for this particular cross-validation setup for the ClinVar dataset, which is a consequence of our decision to not include protein-specific features.

7.3.2.4 *Evaluation of the Model on the Benchmark Datasets*

The results for the evaluation of our model on the datasets from Grimm et al. [167] are done in a LODO cross-validation (Table 7.12), and a specialized form of LODO where we added the samples from ClinVar to the training set in each of the five evaluation rounds (Table 7.13). Each column represents two full rounds of LODO cross-validation runs. In parentheses, one can see the auROC values for the LODO runs, where only those proteins from the training set, which were not present in the test set, were retained. We call this technique protein filtering in the following, since this technique is similar to the protein-based randomization from Section 7.3.2.1. Looking at the mean auROC values of a whole LODO run, we can see that protein filtering decreases the prediction performance in all setups, which is to be expected considering that we remove samples from the training dataset. However, there are individual benchmark datasets and setups, for which the protein filtering increases the auROC values (bold numbers in the table). The dataset, for which the protein filtering decreases the auROC in all setups is ExoVar.
Adding the protein bias feature does not have such a strong effect on the mean auROC values as for the ClinVar dataset (Table 7.9), and to a different extent for the individual datasets. For VariBench, the effect of the protein bias feature is especially strong, but so is also the negative effect for the setup where we performed the protein filtering. When we look into the influence of the protein bias

Dataset	No bias added				Protein bias introduced			
	All samples		Only 3D		All samples		Only 3D	
	Without SF	With SF	Without SF	With SF	Without SF	With SF	Without SF	SF
SwissVar	**0.696 (0.708)**	**0.695 (0.699)**	0.688 (0.686)	0.691 (0.689)	0.686 (0.671)	0.684 (0.673)	0.692 (0.678)	0.690 (0.680)
predictSNP	**0.677 (0.683)**	0.734 (0.709)	**0.706 (0.719)**	0.729 (0.703)	**0.588 (0.615)**	**0.590 (0.610)**	**0.624 (0.669)**	**0.626 (0.664)**
ExoVar	0.803 (0.777)	0.833 (0.787)	0.728 (0.689)	0.753 (0.688)	0.895 (0.809)	0.898 (0.814)	0.852 (0.744)	0.856 (0.769)
HumVar	0.786 (0.771)	0.803 (0.782)	0.716 (0.694)	0.722 (0.699)	**0.799 (0.822)**	**0.803 (0.824)**	0.732 (0.760)	**0.741 (0.781)**
VariBench	**0.676 (0.681)**	0.725 (0.724)	**0.605 (0.612)**	**0.635 (0.700)**	0.788 (0.639)	0.810 (0.648)	0.733 (0.598)	0.741 (0.627)
Mean	0.754 (0.746)	0.775 (0.759)	0.694 (0.679)	0.706 (0.687)	0.787 (0.764)	0.792 (0.768)	0.738 (0.717)	0.744 (0.736)

Table 7.13: *auROC values for all different LODO setups using the benchmark datasets for the evaluation of the model and including ClinVar to the training datasets; in parentheses: filtered out proteins from training set that are present in the test set; in red: the auROC values one would report as the performance of the model since its the model in the most difficult evaluation setup excluding any possible type 2 circularity; in bold: protein-filtering increased the auROC value. SF:structural features, Only 3D: Only variants that could be mapped to 3D structure*

feature on individual datasets, we can see for SwissVar, predictSNPSeleceted, and HumVar decreased auROC values, while for ExoVar and variBench adding the protein bias feature results in largely increased auROC values. This increase is, of course, not present when we apply protein filtering. This shows that different evaluation sets are more or less sensitive towards type 2 circularity effects. An increased auROC through the protein bias feature is always present for datasets, which contain many pure proteins that are present in the training set. For the same reason such an increase can no longer be observed when we apply protein filtering. Similar to the evaluation of the model on ClinVar, structural features have a similar, but weaker, effect as the protein bias feature, which means, we still can associate them with type 2 circularity. However, they improve the model quality, no matter if we apply the protein filtering, which means that despite introducing this potential bias they are valuable for the model.

Surprisingly, adding more samples to the training datasets does not increase the model performance in all setups (Table 7.13). It greatly inflates the auROC values for setups with added protein bias and without protein filtering and without structural features. And the same time, this does not hold true for the setup where the structural features were added. This can be observed for auROC values for VariBench that are heavily inflated for the setups that include the protein bias feature.

Interestingly, protein filtering can increase the auROC values for setups thae include the protein bias feature. The explanation for that is the presence of proteins, whose variants are labeled differently in different datasets that were used for construction of the training and test sets. In these cases, the type 2 circularity effect introduced through the protein bias feature has a negative influence on the performance, and removing such proteins from the training set improves the performance again. Overall, in all these setups, structural features improve the model quality and show a decent robustness against type 2 circularity effects.

If one needs to choose the auROC values to be reported in a comparative study,

they would come from the following setup: using all samples, performing protein filtering (this is very restricting since it forcefully removes any performance inflations introduced by type 2 circularity), including structural features but without the protein bias feature. Using these performance values (marked in red in Table 7.13) and comparing them to the evaluation done in Grimm et al. [167] (Figure 7.1), we can see that our model is very robust. It performs as well as or better than the other methods for the datasets, which are generally more difficult (VariBench, predictSNP and SwissVar), while performing worse (but not too much worse) for datasets, for which the other models had less difficulties (HumVar and ExoVar).

7.4 DISCUSSION

In this chapter, we estimated the potential benefits of using structural features for the prediction of functional and phenotypic effects of nsSNVs. This analysis is similar to the study by Dehiya et al.[159], where the authors analyzed how well different machine learning methods are able to incorporate structural features in the prediction of clinical effects. In comparison to that study, we used only one machine learning method, the random forest, which was the one performed best in their study [159]. On the other hand, we vastly expanded the analysis by adding a second scenario, in which the impact of variants on the function of a protein was assessed using the DMS datasets, and by deeply investigating potential training biases.

7.4.1 *Prediction on Functional Impact*

The DMS datasets pose a tough challenge for the incorporation of structural features, since almost all functionally important residues can be replaced by other similar amino acids without having an effect on the function of the protein, while other more dissimilar amino acid substitutions result in a function loss. Since all of our current structural features are identical for mutations at the same position, this behavior is impossible to predict using structural features. The only features that are divergent for these cases are some evolutionary features and the amino acid property features. We have to find a way to allow the model more efficiently use the amino acid property features in order to differentiate between mutations in the same position that have different functional effects.
The maximum error projections clearly showcase that it has to be possible that these cases can be identified using structural features. But at some point our model fails.: either it cannot identify functionally important residues, or it is not able to estimate correctly the differences between different mutations for functional important residues. On the other hand, the deficiencies of the model could have another reason: the low number of proteins in the training dataset. What one needs to be able to train good model for estimating the impact of mutations on biochemical protein function is more training samples, and a model that is able to generally detect functionally important residues and understand, which amino acid substitutions exactly lead to an impact on the function.

7.4.2 *Prediction of Clinical Effects*

After the evaluation of the model using ten-fold cross-validation on ClinVar, one could have regarded the project as a success, due to a better performance when including structural features both in the variant-based randomization setup and in the protein-based randomization. But the increase was smaller for the protein-based randomization and what we actually wanted to achieve was a model that performs equally well in both setups. In reality, the question of why this happens goes deeper than being a failure or a success. Proteins that are associated with diseases have more structural data associated with them (we already observed that in Chapter 7), and variants in proteins associated with diseases have a higher chance to have a clinical effect, which is then the cause for type 2 circularity. This means that if the model has the information, whether a protein has associated structural data, then it can infer a protein-specific bias and hence type 2 circularity.
The problem of type 2 circularity is that models that are influenced by it will mispredict the effects of difficult variants. For variants whose effects are different from those of other variants in the same protein that are present in the training set, a model that is influenced by type 2 circularity will predict the effect of the variants it has seen in the dataset, which is then a misclassifcation. For variants in proteins that are not present in the training set, a model that is influenced by type 2 circularity will predict the effect by chance.
From the results of the different training setups on the five benchmark datasets, we can conclude that there are three factors, which determine the influence of type 2 circularity on the performance measures of a model:

1. Features associated with type 2 circularity (explained in Section 7.2.2)
2. Biased training datasets, which, for example, contain a lot of pure proteins.
3. The composition of the test dataset: in order to identify type 2 circularity, one has to design the test set such that it contains no variants from proteins present in the training dataset, and if possible single proteins should have variants with mixed labels.

The open question is how to avoid the inclusion of type 2 circularity into the model? We showed through the application of different filtering techniques how to identify individual features with the potential to introduce protein-specific bias, and we can already formulate some guidelines. (1) Never use protein-specific features that cannot be generated for proteins that are not present in the training set. (2) When using protein-specific features and other features, which can be associated with type 2 circularity, design the training set in a way to balance out the labels for each protein. (3) If using as many training samples as possible, then avoid including any features which can be associated with type 2 circularity (see Section 7.2.2).
Unfortunately, we cannot fully exclude the association of structural features with type 2 circularity. Luckily, the results from the LODO evaluation including the ClinVar dataset, show that the benefits of including structural features already outweigh the remaining subtle type 2 circularity effects. Although, the

model still performs slightly better in variant-based randomization setups than in protein-based randomization setups, we proved that our model is able to predict the clinical effects of difficult variants.

7.4.3 *The Current State*

After all the exciting results this project delivered so far, one must not forget that it is still ongoing. Most probably, we are using too many features in all performed scenarios and setups. We still have to pin-point how exactly type 2 circularity is always introduced even when do not seem to use any protein-specific features. Alternatively, we could introduce more features that can be associated with type 2 circularity together an appropriate filtering technique that will not remove so many data points from the training set.
Our results so far show that the DMS dataset is particularly challenging, and we have to design structural features that can be better interpreted by the model. The comparison of the results for predicting the functional effect for the DMS data and the pathogenocity for the ClinVar and other related datasets showed clearly that the problem is extremely complicated and both scenarios have their own challenges but are still related. We already showed the potential of structural features, but we also see that there is a lot of room for improvement.

8 PERSPECTIVE

8.1 CONCLUSIONS

The work presented in this thesis produced algorithmic solutions for structural annotation and structural analysis of proteins on the greatest possible scale. This methodology was used to investigate the biological mechanisms of disease-associated nsSNVs on an unprecedentedly large scale, adding new arguments to an ongoing debate on the role of protein-protein interaction interfaces in pathgenic processes. We put the high-throughput structure annotation pipeline, StructMAn, into use in order to generate a wide array of structural features and incorporate them into a machine-learning tool, thus contributing to the vast field of variant effect prediction.

As a step towards developing these features, we developed a new measure for estimating relative solvent accessible area of individual residues: SphereCon (Chapter 4). Its simple geometrical design enable its application for protein structures with limited information. The correlation of SphereCon to the true relative solvent accessible area is larger than those of any other measure. Additionally, a unique feature of SphereCon is the possibility to perform predictions without structural information *per se*, from predicted pairwise contact or distance matrices. This feature is not useful in our structural annotation studies, but expands the applicability of SphereCon beyond the structural bioinformatics field. SphereCon has the potential to improve any method, which relies on a measure for relative solvent accessible area.

The most time and work was invested into the development of StructMAn (Chapter 5). Only the combination of the high-performance processing of huge amounts of inputs with the comprehensive structural analyses, implemented in StructMAn, enabled to conduct the large studies described in Chapter 6 and 7. The structural annotation of every amino acid in the human proteome is uncomparable and is the proof for the very good performance of StructMAn.

The comprehensive study presented in Chapter 6 pertains the structural analysis of genetic variants associated with cancer and genetic diseases. The study confirmed the previously suggested enrichment of disease-associated variants in interaction interfaces with small molecules and DNA chains, while we could not provide any evidence for an enrichment of such variants in protein-protein interaction interfaces. This results were only possible through the correction of dataset-specific biases with the help of control datasets.

In Chapter 7, we used structural features for the prediction of functional and phenotypic effects of nsSNVs and investigate their potential benefits regarding the prediction performance. Another central point in this study was the investigation of the influence of protein-specific biases on such prediction models, in particular their connection to the structural features, which we could not

refute completely. However, we could show that structural features in general are beneficial for prediction performance.

8.2 OUTLOOK

8.2.1 *Further Improving StructMAn*

The expansion of StructMAn is ever ongoing. One major planned addition to the functionality of StructMAn is the ability to process more types of genetic variants, not limited to nsSNVs anymore. The first step in this direction will be the inclusion of short genomic insertion and deletions (indels), which do not result in a frameshift, and thus introduce small insertions and deletions in the corresponding protein. Another step is to account for alternative splicing events that may result in a wide array of protein-level events, from protein truncations, insertion and retentions of whole domains to short indels. We currently work on implementation of these methods in the framework of a BMBF-funded project Sys_CARE that addresses alternative splicing events in heart and renal diseases.
More technical tasks will address the distribution of the method in the scientific community. For this we need to make StructMAn more easily available for potential users by finishing the containerized version that can function as a simple command-line tool. After that, we will implement a new webserver version of StructMAn that will include all features, which got developed since the last release.
Another possible expansion would be the structural annotation of RNA three-dimensional structures since they can, just as proteins, fold into complex shapes that perform a particular function and harbor mutations that can possibly alter their function as well.
In order to further develop the functional impact prediction tools, we continue implementing new types of structural analyses and ways to produce structural features further improving the accuracy of prediction. It is also conceivable to include structural analyses, which are more computationally expensive if performed on a subset of structures. For example, in an ongoing study, we include assessment of the change of protein and/or complex stability, estimated by FoldX [141], for all clinically relevant variants from ClinVar [63] and all population variants from GnomAD [189].

8.2.2 *New Large-scale Studies can Reveal New Insights*

New functionalities of StructMAn also enables new possible studies. For example, we can expand the study of disease-associated variants to insertions and deletions. Another planed study is the structural analysis of the effects introduced by alternative splicing events, whose effects on the protein structure are similar to those of insertions and deletions.
To follow up on the conclusions from the study of disease-associated variants, we can also design a study around the structural analysis of variants with different population-wide allele frequency, possibly revealing the true structural

background distributions of nsSNVs.

8.2.3 *The Future of Structural Features in Variant Effect Prediction*

The most room for improvement is offered by the variant effect prediction method. We already started a separate project, performed by a Masters student Max Jacob, with the aim to reduce the number of features via feature selection and to introduce new structural features, which are better suited for the different prediction scenarios. Another challenge is the construction of meaningful training sets with the goal to minimize any types of training biases.
We also plan to expand the variant impact prediction methods to more specific scenarios. Here we also started a project, performed by another Masters student Sami Laradji, investigating the possibility to create a specialized version of our model into the prediction of impact of nsSNVs on protein-protein interactions. Other specific phenotypes that can be addressed are, for example, antibiotics resistance or specific human diseases.
A completely different big question is if we can combine the both types of training data, using the knowledge learned from the DMS data to improve the prediction quality for clinical effects.
Finally, we plan investigate the impact of combinations of mutations on protein function. It is well-known that some pairs of mutations can have a compensating effect on each other [190], yet most variant impact prediction tools treat variants independently. We will address this shortcoming, in particular for variants occurring in the same gene, by developing new structural features for combinations of amino acid substitutions. This is particularly important in bacteria and viruses, but also in higher organisms, where linkage disequilibrium does not allow handling combinations of mutations independent from each other.

9 APPENDIX

9.1 SUPPLEMENTARY LISTS

9.1.1 *Boring Ligands*

List of PDB Ligand IDs of molecules filtered from the structural analysis of StructMAn:
'MYR', 'ARF', 'BEZ', 'SIA', 'UNK', 'DTD', 'SQD', 'SPD', '2PO', 'DEP', 'DPN', 'CLA', 'PTR', 'CSO', 'NTB', 'EDO', 'B3P', 'MPT', 'CL1', 'NAG', 'PEG', 'P6G', 'MYS', '5UA', 'P4C', 'PEE', 'PHS', 'TAR', 'MES', 'DMS', 'PYR', 'LLP', 'LBT', '2HP', 'TTP', 'HTG', 'PGW', 'PE5', 'DGL', 'PE4', 'ACY', 'WO6', 'D10', 'PGE', 'MRD', 'CAC', 'MAN', '3GR', '7PE', 'PO3', 'PG0', 'DCY', 'EPE', 'LDA', 'CHL', 'U5P', 'AE3', 'PT5', 'OMT', 'DPF', 'TYS', 'PEU', 'SRT', 'DTU', 'ETX', 'DPR', 'BCN', 'MAG', 'NCO', 'ETE', 'PE8', 'C10', 'GAL', 'PI ', 'HEX', 'MC3', 'CSW', 'CZ2', 'MSE', 'FTT', 'I42', 'YBT', 'ACT', 'BCR', 'PTL', 'MPD', 'EGC', 'CSS', 'PEF', 'SOH', 'UNL', '1PG', 'MGE', 'TPQ', 'TRS', 'VO4', 'MOE', 'UND', 'C8E', 'TBU', 'CGU', 'FME', 'CXE', 'SF4', '5GP', 'ORN', 'TOE', 'CO3', 'CEA', 'PSC', '2PE', 'OC9', 'DAL', 'CHD', 'DGN', 'PHF', 'D9G', 'LNK', 'CSD', 'HOH', 'SAC', 'P33', 'SBY', 'DKA', 'PLM', 'M2M', 'GOL', 'PG4', 'CME', 'NBU', 'P3G', 'PIO', 'MPO', 'MXE', 'ANL', 'CBS', 'HF3', 'ACD', 'DIO', 'DTT', 'BME', 'BOG', 'TLA', '6JZ', 'DHI', 'TPO', 'FMT', 'DSN', 'CXM', 'DOD', 'CXS', '12P', 'F3S', '15P', 'BTB', 'PAM', 'PGV', 'DAR', 'PLX', 'C5P', 'SOL', 'NDG', 'DTY', 'SAR', 'HP6', 'CDL', 'DIL', 'PCA', 'DIV', 'TAM', 'STE', 'WO4', 'CE9', 'WO3', 'SGN', 'D12', 'POP', 'GDL', 'HEM', 'HYP', 'OCT', '5HP', 'PPI', 'PG6', '1PE', 'GLC', 'DSG', 'CSX', 'PO4', '5AX', 'KMB', 'DD9', 'I3P', 'LI1', 'ABA', 'NLE', 'DTR', 'HEZ', 'SO4', 'D1D', 'BU1', 'BNG', 'PGO', 'SPM', 'STY', 'ZRC', 'SEP', 'MG8', 'ACE', 'DLE', 'DLY', 'HTO', '6PL', 'A2G', 'AIB', 'LAP', 'OCS', 'LMG', 'PE3', 'IUM', 'DVA', 'MLE', 'WAT', 'DTH', 'FUC', 'LHG', 'NGA', 'SO3', 'DTV', 'LMT', 'KCX', 'MVA', 'BMT', 'PSE', 'PGR', 'TRD', 'LMU', 'DAO', 'IPA', '2HA', 'DSP', 'PGM', 'UPL', 'PG5', 'BMA'

9.1.2 *Metals*

List of PDB Ligand IDs of metals considered by the metal interaction classification of StructMAn:
'0BE', '3CO', '3NI', '4MO', '4PU', '4TI', '6MO', 'AG', 'AL', 'AM', 'AU', 'AU3', 'BA', 'BS3', 'CA', 'CD', 'CE', 'CF', 'CO', 'CR', 'CS', 'CU', 'CU1', 'CU3', 'DY', 'ER3', 'EU', 'EU3', 'FE', 'FE2', 'GA', 'GD', 'GD3', 'HG', 'HO', 'HO3', 'IN', 'IR', 'IR3', 'K', 'LA', 'LI', 'LU', 'MG', 'MN', 'MN3', 'MO', 'NA', 'NI', 'OS', 'OS4', 'PB', 'PD', 'PR', 'PT', 'PT4', 'RB', 'RE', 'RH', 'RH3', 'RU', 'SM', 'SR', 'TA0', 'TB', 'TH', 'TL', 'U1', 'V', 'W', 'Y1', 'YB', 'YB2', 'YT3', 'ZCM', 'ZN', 'ZN2', 'ZR'

9.1.3 *Ions*

List of PDB Ligand IDs of ions considered by the ion interaction classification of StructMAn:
'BR', 'BRO', 'CL', 'CLO', 'F', 'FLO', 'IDO', 'IOD', 'SB'

9.2 SUPPLEMENTARY FIGURES

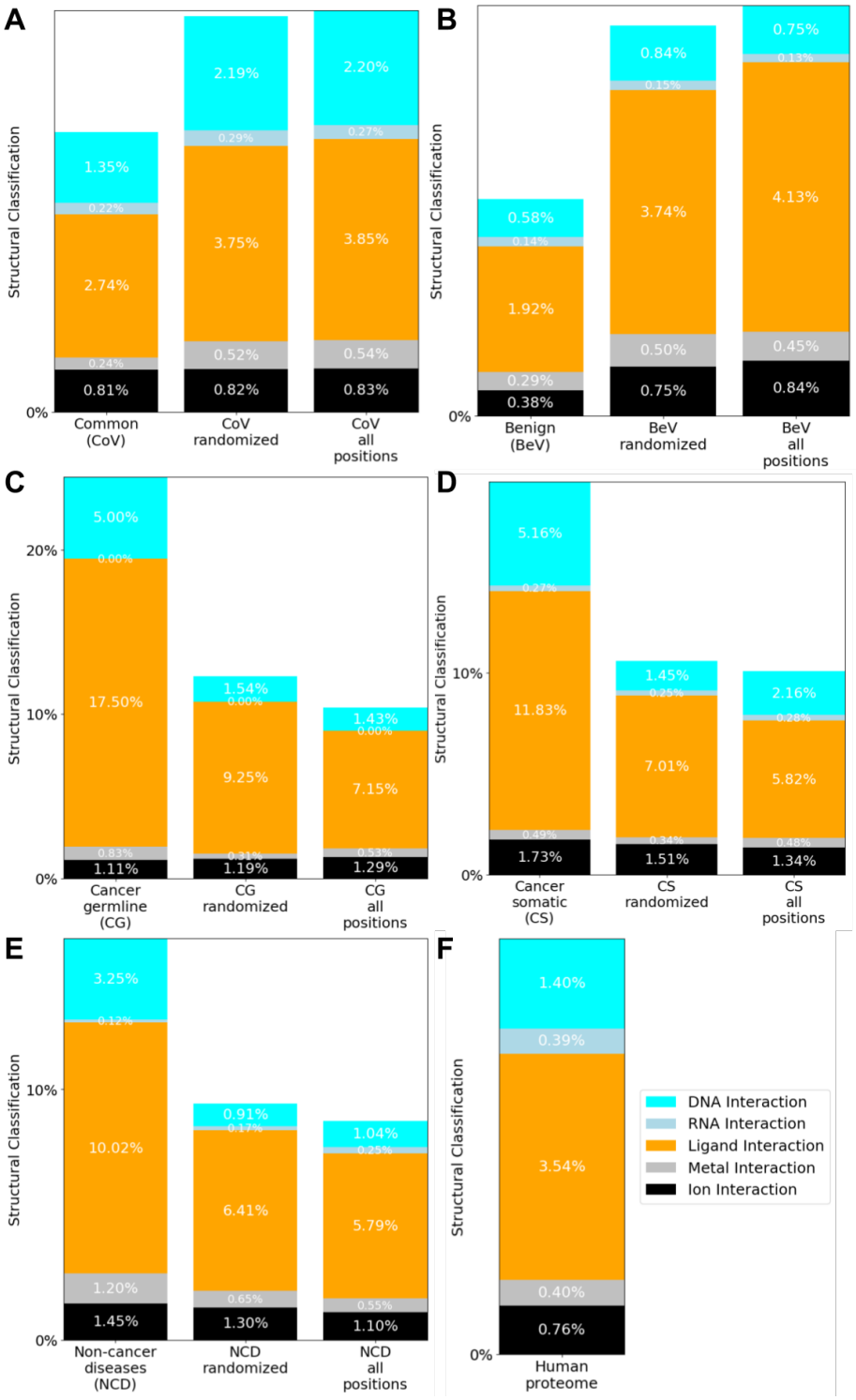


Figure 9.1: *A-E: Spatial distributions of the interaction classes of the five main datasets in comparison to control datasets; F: Spatial distribution of the human proteome.*

9.3 SUPPLEMENTARY TABLES

3-letter code	intersection sphere radius
CYS	3.03
ILE	3.32
SER	2.94
GLN	3.39
LYS	3.43
TRP	4.02
PRO	3.17
THR	3.12
PHE	3.72
ALA	2.80
GLY	2.57
HIS	3.53
ASN	3.23
LEU	3.32
ARG	3.63
ASP	3.24
VAL	3.17
GLU	3.39
TYR	3.80
MET	3.35

Table 9.1: *Radii of intersecting spheres for individual amino acids (Å).*

PDB ID	SCOP ID	SCOP class	PDB ID	SCOP ID	SCOP class	PDB ID	SCOP ID	SCOP class	PDB ID	SCOP ID	SCOP class	PDB ID	SCOP ID	SCOP class	PDB ID	SCOP ID	SCOP class	PDB ID	SCOP ID	SCOP class
1DS7	d.90.1.1	d	1O5W	c.3.1.2	c	1WFQ	b.40.4.5	b	2RD5	c.73.1.0	c	1S70	d.211.1.1	d	2FQ3	a.4.1.18	a	2P5X	c.51.4.0	c
1TQY	c.95.1.1	c	1ZLP	c.1.12.0	c	1CKM	b.40.4.6	b	2CDQ	c.73.1.3	c	2A1I	c.52.1.20	c	1S4I	b.1.8.0	b	4BKM	c.108.1.0	c
1MZJ	c.95.1.2	c	1SGJ	c.1.12.5	c	1GPC	b.40.4.7	b	1JNP	b.63.1.1	b	1HSK	d.145.1.2	d	1CVR	b.1.18.12	b	1AQ6	c.108.1.1	c
1RI1	c.66.1.34	c	1MUM	c.1.12.7	c	2O73	a.288.1.1	a	3CTO	d.306.1.0	d	1REG	d.58.27.1	d	1LSU	c.2.1.9	c	1NEI	d.253.1.1	d
4Q25	a.7.12.0	a	4I5Q	c.47.1.9	c	1A7T	d.157.1.1	d	1R1M	d.79.7.1	d	4IWG	c.135.1.0	c	2ROZ	b.55.1.2	b	1U79	d.26.1.1	d
1QSD	a.7.5.1	a	1Y7Q	a.28.3.2	a	2BJO	d.227.1.0	d	2OQY	c.1.11.0	c	2FQM	d.378.1.1	d	1Mai	b.55.1.1	b	3PR9	d.26.1.0	d
1OIP	a.5.3.1	a	2JS1	a.23.7.1	a	1JYO	d.184.1.2	d	1S3L	d.159.1.7	d	3WVJ	b.29.1.2	b	1M4J	d.109.1.2	d	1WOU	c.47.1.16	c
2MZY	d.279.1.0	d	4P0J	b.42.1.2	b	1B5O	c.67.1.1	c	1V0E	b.68.1.2	b	2VG9	b.29.1.0	b	2NAC	c.2.1.4	c	1GEF	c.52.1.18	c
1JCQ	a.118.6.1	a	1MNT	a.43.1.1	a	2Z1Z	c.67.1.0	c	1G5B	d.159.1.3	d	2H7A	d.350.1.1	d	1NPY	c.2.1.7	c	2RR8	a.40.1.0	a
1J3P	b.82.1.7	b	4J8E	a.118.8.0	a	1IMJ	c.69.1.23	c	1JHJ	b.18.1.9	b	1QU0	b.29.1.4	b	3TU5	d.109.1.1	d	1BHD	a.40.1.1	a
4BTJ	d.144.1.0	d	1RI7	a.4.5.32	a	3I8Z	b.34.13.2	b	4OMC	b.18.1.0	b	2BQ4	a.138.1.0	a	3NGL	c.2.1.0	c	4L0P	b.6.1.5	b
1XQZ	d.144.1.7	d	2YVX	a.118.26.1	a	3K86	b.45.1.0	b	1I9G	c.66.1.13	c	3MD9	c.92.2.0	c	1TLT	c.2.1.3	c	1AK0	a.124.1.2	a
1ZYL	d.144.1.6	d	1ECS	d.32.1.2	d	1F2L	d.9.1.1	d	2HJJ	d.50.3.3	d	2L8D	b.34.9.1	b	1LW3	b.55.1.8	b	2AAN	b.6.1.0	b
2PAM	b.82.1.1	b	1VF6	a.194.1.1	a	2L4N	d.9.1.0	d	4CGE	d.2.1.0	d	2M16	b.34.9.2	b	2GZB	b.42.4.0	b	1S1G	d.42.1.2	d
1ZPS	b.168.1.1	b	1YK3	d.108.1.1	d	1DWY	d.6.1.1	d	1KPI	c.66.1.18	c	4JGX	d.106.1.0	d	1NLT	b.4.1.1	b	1S68	d.142.2.4	d
1IW0	a.132.1.1	a	1TD6	a.234.1.1	a	1SU0	d.224.1.2	d	1LTU	d.178.1.1	d	1V74	a.24.20.1	a	4K8U	b.8.1.0	b	2LV3	c.47.1.0	c
2IC1	b.82.1.19	b	1TFP	b.3.4.1	b	2HTH	b.55.1.12	b	1T4O	d.50.1.1	d	1PVW	d.115.1.2	d	3HU1	b.52.2.0	b	1OPM	b.121.1.2	b
1BXY	d.59.1.1	d	2J9U	a.24.28.1	a	2E1F	a.60.8.1	a	1EJJ	c.76.1.3	c	1SV6	d.177.1.1	d	3MEZ	b.78.1.0	b	4DHX	a.301.1.1	a
2OAU	b.38.1.3	b	1CAU	b.82.1.2	b	5CFO	d.387.1.0	d	1H2I	d.50.1.3	d	1A1U	a.53.1.1	a	2BIW	b.69.15.1	b	1N0G	b.129.1.2	b
1FVA	d.58.28.1	d	3JXF	b.74.1.0	b	2HBJ	a.60.8.4	a	2A4V	c.47.1.10	c	1IZJ	b.1.18.2	b	1JWO	d.93.1.1	d	3LPN	c.61.1.0	c
2ETN	a.2.1.1	a	1WY5	c.26.2.5	c	1ICH	a.77.1.2	a	1EZG	b.80.2.1	b	2DJL	c.1.4.1	c	2W9P	d.17.1.0	d	1E2B	c.44.2.1	c
2P4V	a.2.1.0	a	1BKC	d.92.1.10	d	2LHJ	a.21.1.0	a	1R5P	c.47.1.15	c	1NEP	b.1.18.7	b	1K12	b.18.1.15	b	3GLS	c.31.1.0	c
1QWJ	c.68.1.13	c	1HV5	d.92.1.11	d	1A1W	a.77.1.4	a	1COM	d.79.1.2	d	1OUV	a.118.18.1	a	1AHH	c.2.1.2	c	1DZF	c.52.3.1	c
1MGT	a.4.2.1	a	1OZ9	d.92.1.15	d	1QND	d.106.1.1	d	1NQ3	d.79.1.1	d	4AKM	b.180.1.1	b	1NAE	b.18.1.10	b	2A2N	b.62.1.1	b
1JAJ	d.218.1.2	d	1IHM	b.121.4.3	b	2VO9	d.65.1.5	d	4H2D	c.23.5.0	c	1R1T	a.4.5.5	a	1U8V	a.29.3.1	a	4C3Z	c.37.1.0	c
2JMF	b.72.1.1	b	1G10	d.137.1.1	d	1R28	d.42.1.1	d	3AMR	b.68.3.1	b	1BM8	d.34.1.1	d	1WER	a.116.1.2	a	1S4Q	c.37.1.1	c
1KKE	b.21.1.2	b	2IG3	a.1.1.0	a	5BXD	d.42.1.0	d	3ANS	c.69.1.11	c	2ZME	a.4.5.0	a	1EKG	d.82.2.1	d	1SHS	b.15.1.1	b
4PJ1	b.35.1.0	b	1DLY	a.1.1.1	a	1MWW	d.80.1.4	d	1PBV	a.118.3.1	a	1UXC	a.35.1.5	a	2AQX	d.143.1.3	d	2Z6O	d.20.1.4	d
1KOL	b.35.1.2	b	3DGP	d.295.1.0	d	1OTG	d.80.1.2	d	1V4P	d.67.1.2	d	1P27	d.58.7.1	d	3E3U	d.167.1.0	d	1EGA	c.37.1.8	c
1N69	a.64.1.3	a	2G19	b.82.2.15	b	1SZH	a.226.1.1	a	1OZN	c.10.2.7	c	2LCV	a.35.1.0	a	1HKV	b.49.2.3	b	3WG6	c.1.7.0	c
1NP7	a.99.1.1	a	1HQB	a.28.1.3	a	1IR6	c.107.1.2	c	1SQU	d.252.1.1	d	1FZA	d.171.1.1	d	1BD0	b.49.2.2	b	1G99	c.55.1.2	c
2FZ6	b.138.1.1	b	2KWL	a.28.1.0	a	1LN1	d.129.3.2	d	1EYV	a.79.1.1	a	1BHG	b.1.4.1	b	3HMB	d.118.1.0	d	1HUX	c.55.1.5	c
2A71	b.29.1.13	b	1KLP	a.28.1.1	a	2WQL	d.129.3.1	d	1D0B	c.10.2.1	c	2HJ3	a.24.15.0	a	1Z6I	d.118.1.1	d	1YNS	c.108.1.22	c
3LX2	d.131.1.0	d	3WI7	c.69.1.0	c	3TVQ	d.129.3.6	d	2J5A	d.58.14.1	d	1JR8	a.24.15.1	a	2AVD	c.66.1.1	c	2JGA	c.108.1.21	c
1J2G	d.96.1.4	d	1DQY	c.69.1.3	c	3DJU	d.370.1.1	d	1OGQ	c.10.2.8	c	2N8G	a.4.1.0	a	1I1N	c.66.1.7	c	1EIX	c.1.2.3	c
1B9L	d.96.1.3	d	1JJF	c.69.1.2	c	2DK4	a.140.6.1	a	1K9I	d.169.1.1	d	2IW5	a.4.1.3	a	4A1M	d.60.1.4	d	1JCQ	a.102.4.3	a
3QN0	d.96.1.0	d	1S0X	a.123.1.1	a	2F15	b.1.18.21	b	1Q08	a.6.1.3	a	1IUF	a.4.1.7	a	2F4M	a.189.1.1	a	2A8C	c.53.2.1	c
1FB1	d.96.1.1	d	1LVO	b.47.1.4	b	2ZL1	d.110.7.1	d	3CT7	c.1.2.0	c	3N2N	c.62.1.1	c	1JYH	d.60.1.3	d	3TEN	c.53.2.0	c
1L5J	a.118.15.1	a	1MBM	b.47.1.3	b	1JRM	d.206.1.1	d	2LMI	d.58.7.0	d	1SGM	a.4.1.9	a	1AVD	b.61.1.1	b	2OHW	d.79.8.1	d
1N5Z	b.34.2.1	b	3DFL	b.47.1.0	b	2A1J	a.60.2.5	a	1JX6	c.93.1.1	c	2ARW	b.1.2.0	b	2LOY	b.88.1.0	b	1T90	c.82.1.0	c
3O5Z	b.34.2.0	b	1ARB	b.47.1.1	b	1BJ8	b.1.2.1	b	1YXY	c.1.2.5	c	1I4W	c.66.1.24	c	1Z9H	a.45.1.1	a	1YH2	d.20.1.1	d
2AE0	b.52.1.4	b	1NJS	c.65.1.1	c	1S5U	d.38.1.1	d	1OMO	c.2.1.13	c	1KHI	b.34.5.2	b	1Y4Y	d.94.1.1	d	3M9W	c.93.1.0	c
3JWQ	a.211.1.2	a	1CDZ	c.15.1.1	c	1WU5	a.102.1.0	a	1B0P	c.36.1.8	c	2ZVS	d.58.1.1	d	1WTJ	c.122.1.0	c	1EJF	b.15.1.2	b
2O8H	a.211.1.0	a	1BBY	a.4.5.15	a	1CEM	a.102.1.2	a	2BWB	a.5.2.1	a	1M98	a.175.1.1	a	1FL9	c.37.1.18	c	2JW8	d.254.1.2	d
1UEK	d.14.1.5	d	1KZQ	b.6.2.1	b	1BYQ	d.122.1.1	d	2NN4	a.272.1.1	a	1G8E	a.145.1.1	a	1UVH	a.25.1.1	a	1T43	c.66.1.30	c
2RQG	a.144.1.1	a	1ZE1	b.122.1.1	b	1O0W	a.149.1.1	a	1DAQ	a.139.1.1	a	1B7V	a.3.1.1	a	1AEP	a.63.1.1	a	1NW3	c.66.1.31	c
1B1G	a.39.1.1	a	3EOP	b.122.1.0	b	4GKF	a.70.2.0	a	1S66	d.110.3.2	d	4IZB	c.14.1.0	c	1S05	a.24.3.2	a	1VE6	b.69.7.2	b
1IRJ	a.39.1.2	a	2FYH	d.61.1.0	d	2OEB	a.70.2.1	a	2Z6C	d.110.3.0	d	4U1A	c.14.1.3	c	1YQE	c.56.7.1	c	1KKT	a.102.2.1	a
2QFB	b.88.2.1	b	1JH6	d.61.1.1	d	1GGQ	a.24.12.1	a	1PPR	a.131.1.1	a	2HEY	b.22.1.1	b	1WA8	a.25.3.1	a	1UV6	b.96.1.1	b
1AIR	b.80.1.1	b	1U2P	c.44.1.0	c	1U9L	a.60.4.2	a	1TF7	c.37.1.11	c	4QPY	b.22.1.0	b	2KX3	d.15.1.1	d	1TLJ	d.282.1.1	d
1Q2V	a.129.1.2	a	3JUS	a.104.1.0	a	1CLX	c.1.8.3	c	3LBS	c.94.1.1	c	1EJ2	c.26.1.3	c	4WIP	d.15.1.0	d	2Q7F	a.118.8.1	a
1P5H	c.123.1.1	c	1IZO	a.104.1.1	a	2JIE	c.1.8.0	c	4YAH	c.94.1.0	c	1G71	d.264.1.1	d	3VTV	d.15.1.3	d	1YNH	d.126.1.7	d
2LGW	a.2.3.0	a	2CC3	d.17.4.26	d	2VFW	c.101.1.0	c	1L2M	d.89.1.4	d	2WZL	d.293.1.0	d	2HV7	a.268.1.1	a	1G61	d.126.1.1	d
2X9A	b.37.1.0	b	3BAL	b.82.1.21	b	1DC7	c.23.1.1	c	1S2M	c.37.1.19	c	1RI6	b.69.11.1	b	2ZE5	c.37.1.26	c	4BOF	d.126.1.0	d
1LVF	a.47.2.1	a	3VNA	c.44.3.0	c	2QV0	c.23.1.0	c	1PMI	b.82.1.3	b	2KM4	a.118.9.0	a	1GFF	b.121.5.1	b	1HKS	a.4.5.22	a
2YYN	a.29.2.0	a	4LHN	b.179.1.2	b	1M2E	c.23.1.5	c	1IJY	a.141.1.1	a	1J1J	a.118.16.1	a	1OW5	a.60.1.2	a	2DSO	b.68.6.1	b
1GYM	c.1.18.2	c	1YAC	c.33.1.3	c	3N90	b.80.8.0	b	1Q5Z	a.196.1.1	a	1C2Y	c.16.1.1	c	3BQ7	a.60.1.0	a	1P4X	a.4.5.28	a
1TL2	b.67.1.1	b	2VL6	b.40.4.0	b	2G0Y	b.80.8.1	b	1A6J	d.112.1.1	d	1IB2	a.118.1.8	a	1SV4	a.60.1.1	a	1E2T	d.3.1.5	d
1JFA	a.128.1.5	a	1DY2	d.169.1.5	d	1Z8L	a.48.2.1	a	1FLG	b.70.1.1	b	2OWI	a.91.1.0	a	3C90	a.204.1.4	a	3MH8	b.125.1.0	b
1RQP	b.141.1.1	b	1O7I	b.40.4.3	b	1B79	a.81.1.1	a	2BTD	a.208.1.0	a	2JM5	a.91.1.1	a	1P15	c.45.1.2	c	2GBZ	c.55.3.0	c
1K40	a.24.14.1	a	2OX8	d.169.1.0	d	1E2A	a.7.2.1	a	1PD3	a.30.3.1	a	1BK5	a.118.1.1	a	1FPZ	c.45.1.1	c	1HJR	c.55.3.6	c
1I1I	d.92.1.5	d	2OMD	d.41.5.0	d	4FD9	b.11.1.0	b	4TUM	d.211.1.0	d	3BWT	a.118.1.0	a	1YZ4	c.45.1.0	c	1HI9	c.99.1.1	c
1Q1R	c.3.1.5	c	3B5H	b.1.1.4	b	1ZOX	b.1.1.0	b	1L6Z	b.1.1.1	b	1HDM	b.1.1.2	b	2KZF	d.52.7.0	d	1TV7	c.1.28.3	c
2HKQ	b.34.10.1	b	3DKU	d.113.1.0	d	1EYY	c.82.1.1	c	1K32	b.36.1.3	b	1F05	c.1.10.1	c	1YAV	d.37.1.1	d	1J5S	c.1.9.8	c
1F5M	d.110.2.1	d	2D0O	c.8.6.1	c	4JN6	a.5.7.0	a	3TSV	b.36.1.0	b	1LFW	c.56.5.4	c	3FV6	d.37.1.0	d	1J2T	c.125.1.1	c
1TJN	c.92.1.3	c	1RYA	d.113.1.5	d	1RLM	c.108.1.10	c	1WH1	b.36.1.1	b	1OBR	c.56.5.2	c	4P5U	c.1.9.12	c	1WLG	b.152.1.1	b
3ES6	b.1.18.23	b	1HVX	b.71.1.1	b	1FIQ	d.41.1.1	d	2A18	d.58.56.1	d	3DH1	c.97.1.0	c	1K5D	c.10.1.2	c	1KRM	c.1.9.1	c
1B49	d.117.1.1	d	1MLA	c.19.1.1	c	2HHL	c.108.1.16	c	4OX8	d.58.56.0	d	3QKG	b.60.1.0	b	3O15	c.1.3.1	c	1J6Q	b.40.9.1	b
4QT4	c.56.3.0	c	1W91	b.71.1.2	b	3ESQ	c.108.1.19	c	1PWE	c.79.1.1	c	1S7O	a.4.13.3	a	1ISF	c.23.14.3	c	2IWM	d.153.1.3	d
2AKJ	d.58.36.1	d	1AO6	a.126.1.1	a	1FJR	b.102.1.1	b	1JYO	d.198.1.1	d	1J26	d.50.4.1	d	1XFK	c.42.1.1	c	1ABV	a.70.1.1	a
2FZP	d.345.1.1	d	3KT9	b.26.1.0	b	2J49	d.379.1.1	d	1A0B	a.24.10.1	a	2EBF	a.296.1.1	a	1S57	d.58.6.1	d	1AHS	b.19.1.1	b
1ATJ	a.93.1.1	a	1ECM	a.130.1.1	a	1BG8	a.57.1.1	a	1HX3	d.113.1.2	d	2DUK	d.113.1.1	d						

Table 9.2: *Gold standard dataset of selected structures, one for each SCOP class.*

3-letter code	SC-S1		SC-S2	
	r	a	r	a
CYS	7.0	0.85	7.25	0.85
ILE	7.75	0.9	7.75	0.85
SER	6.5	0.9	7.0	0.85
GLN	7.25	0.95	7.75	1.0
LYS	7.5	0.9	8.0	1.0
TRP	8.0	0.95	9.0	0.6
PRO	7.0	0.85	7.5	1.0
THR	6.75	0.85	7.0	1.0
PHE	7.75	0.85	8.25	0.9
ALA	7.0	0.85	7.0	0.85
GLY	6.75	1.0	6.75	1.0
HIS	7.25	0.8	8.0	0.95
ASN	6.75	0.95	7.25	1.0
LEU	7.75	0.9	7.75	0.85
ARG	7.75	0.95	8.5	1.0
ASP	6.75	0.95	7.25	0.9
VAL	7.5	0.85	7.5	0.9
GLU	7.0	0.95	7.5	1.0
TYR	7.75	0.85	8.75	0.85
MET	7.5	1.0	8.0	1.0

Table 9.3: *Optimal search space paramaters for SphereCon SC-S1 and SC-S2.*

	DNA contacts (%)	Ligand contacts (%)	Protein contacts (%)	Core (%)	Surface (%)
cancer germline	4.5	19.4	31.6	26.9	17.6
100 random samples from cancer germline (mean ± s.d.) (mean ± s.d.)	1.2 ± 0.4	5.8 ± 0.7	32.8 ± 1.5	30.6 ± 1.5	29.7 ± 1.9
cancer somatic	4.3	13.3	31.5	25.9	24.9
100 random samples from cancer somatic (mean ± s.d.) (mean ± s.d.)	1.4 ± 0.2	8.6 ± 0.3	29.5 ± 0.5	27.4 ± 0.6	33.2 ± 0.6
non-cancer diseases	2.8	12.7	24.5	42.0	18.0
100 random samples from non-cancer diseases (mean ± s.d.) (mean ± s.d.)	1.6 ± 0.1	6.0 ± 0.1	24.1 ± 0.3	27.5 ± 0.3	40.8 ± 0.3

Table 9.4: *(from Gress et al., 2017 [152]) Distribution of structural classes in disease-associated datasets compared to 100 randomly sampled equally sized sets from the sets of common variants with the same distribution of identified template structures as in the corresponding disease-associated datasets. Insignificant differences (within four standard deviations) are marked in red.*

BIBLIOGRAPHY

[1] Francis Crick. "Central Dogma of Molecular Biology." en. In: *Nature* 227.5258 (Aug. 1970), pp. 561–563. ISSN: 1476-4687. DOI: 10.1038/227561a0. URL: https://www.nature.com/articles/227561a0 (visited on 11/06/2019).

[2] "UniProt: a hub for protein information." In: *Nucleic Acids Research* 43.Database issue (Jan. 2015), pp. D204–D212. ISSN: 0305-1048. DOI: 10.1093/nar/gku989. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4384041/ (visited on 11/08/2019).

[3] Wolfgang Kabsch and Christian Sander. "Dictionary of protein secondary structure: Pattern recognition of hydrogen-bonded and geometrical features." en. In: *Biopolymers* 22.12 (1983), pp. 2577–2637. ISSN: 1097-0282. DOI: 10.1002/bip.360221211. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/bip.360221211 (visited on 06/19/2019).

[4] J. Michael Word et al. "Visualizing and quantifying molecular goodness-of-fit: small-probe contact dots with explicit hydrogen atoms11Edited by J. Thornton." en. In: *Journal of Molecular Biology* 285.4 (Jan. 1999), pp. 1711–1733. ISSN: 0022-2836. DOI: 10.1006/jmbi.1998.2400. URL: http://www.sciencedirect.com/science/article/pii/S0022283698924007 (visited on 11/28/2019).

[5] Vijaya Parthiban, M. Michael Gromiha, and Dietmar Schomburg. "CUPSAT: prediction of protein stability upon point mutations." In: *Nucleic Acids Research* 34.Web Server issue (July 2006), W239–W242. ISSN: 0305-1048. DOI: 10.1093/nar/gkl190. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1538884/ (visited on 02/20/2019).

[6] Douglas E. V. Pires, David B. Ascher, and Tom L. Blundell. "mCSM: predicting the effects of mutations in proteins using graph-based signatures." In: *Bioinformatics* 30.3 (Feb. 2014), pp. 335–342. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btt691. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3904523/ (visited on 02/20/2019).

[7] Joost Schymkowitz et al. "The FoldX web server: an online force field." In: *Nucleic Acids Research* 33.Web Server issue (July 2005), W382–W388. ISSN: 0305-1048. DOI: 10.1093/nar/gki387. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1160148/ (visited on 11/08/2019).

[8] Matthias Rarey et al. "A Fast Flexible Docking Method using an Incremental Construction Algorithm." en. In: *Journal of Molecular Biology* 261.3 (Aug. 1996), pp. 470–489. ISSN: 0022-2836. DOI: 10.1006/jmbi.1996.0477. URL: http://www.sciencedirect.com/science/article/pii/S0022283696904775 (visited on 12/03/2019).

[9] G.C.P. van Zundert et al. "The HADDOCK2.2 Web Server: User-Friendly Integrative Modeling of Biomolecular Complexes." In: *Journal of Molecular Biology* 428.4 (2016). Computation Resources for Molecular Biology, pp. 720 –725. ISSN: 0022-2836. DOI: https://doi.org/10.1016/j.jmb.2015.09.014. URL: http://www.sciencedirect.com/science/article/pii/S0022283615005379.

[10] Mark James Abraham et al. "GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers." In: *SoftwareX* 1-2 (2015), pp. 19 –25. ISSN: 2352-7110. DOI: https://doi.org/10.1016/j.softx.2015.06.001. URL: http://www.sciencedirect.com/science/article/pii/S2352711015000059.

[11] Ivan A. Adzhubei et al. "A method and server for predicting damaging missense mutations." In: *Nature methods* 7.4 (Apr. 2010), pp. 248–249. ISSN: 1548-7091. DOI: 10.1038/nmeth0410-248. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2855889/ (visited on 02/22/2019).

[12] Mark F Rogers et al. "FATHMM-XF: accurate prediction of pathogenic point mutations via extended features." In: *Bioinformatics* 34.3 (Feb. 2018), pp. 511–513. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btx536. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5860356/ (visited on 11/28/2019).

[13] Yana Bromberg and Burkhard Rost. "SNAP: predict effect of non-synonymous polymorphisms on function." In: *Nucleic Acids Research* 35.11 (June 2007), pp. 3823–3835. ISSN: 0305-1048. DOI: 10.1093/nar/gkm238. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1920242/ (visited on 11/08/2019).

[14] Vanessa E. Gray et al. "Quantitative Missense Variant Effect Prediction Using Large-Scale Mutagenesis Data." In: *Cell Systems* 6.1 (Jan. 2018), 116–124.e3. ISSN: 2405-4712. DOI: 10.1016/j.cels.2017.11.003. URL: http://www.sciencedirect.com/science/article/pii/S2405471217304921 (visited on 02/19/2019).

[15] Daniel Quang, Yifei Chen, and Xiaohui Xie. "DANN: a deep learning approach for annotating the pathogenicity of genetic variants." en. In: *Bioinformatics* 31.5 (Mar. 2015), pp. 761–763. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btu703. URL: https://academic.oup.com/bioinformatics/article/31/5/761/2748191 (visited on 02/19/2019).

[16] Philipp Rentzsch et al. "CADD: predicting the deleteriousness of variants throughout the human genome." en. In: *Nucleic Acids Research* 47.D1 (Jan. 2019), pp. D886–D894. ISSN: 0305-1048. DOI: 10.1093/nar/gky1016. URL: https://academic.oup.com/nar/article/47/D1/D886/5146191 (visited on 02/19/2019).

[17] Bruce Alberts et al. *Molecular Biology of the Cell*. 4th. Garland Science, 2002. ISBN: 978-0-8153-3218-3 978-0-8153-4072-0.

[18] J. D. Watson and F. H. C. Crick. "The Structure of Dna." en. In: *Cold Spring Harbor Symposia on Quantitative Biology* 18 (Jan. 1953), pp. 123–131. ISSN: 0091-7451, 1943-4456. DOI: 10.1101/SQB.1953.018.01.020. URL: http://symposium.cshlp.org/content/18/123 (visited on 01/21/2020).

[19] Nelson David L et al. *Lehninger Principles of Biochemistry*. en. Google-Books-ID: 7chANoUYoLYC. W. H. Freeman, 2005. ISBN: 978-0-7167-4339-2.

[20] Christian B. Anfinsen. "Principles that Govern the Folding of Protein Chains." en. In: *Science* 181.4096 (July 1973), pp. 223–230. ISSN: 0036-8075, 1095-9203. DOI: 10.1126/science.181.4096.223. URL: https://science.sciencemag.org/content/181/4096/223 (visited on 11/06/2019).

[21] Ken A. Dill and Justin L. MacCallum. "The Protein-Folding Problem, 50 Years On." en. In: *Science* 338.6110 (Nov. 2012), pp. 1042–1046. ISSN: 0036-8075, 1095-9203. DOI: 10.1126/science.1219021. URL: https://science.sciencemag.org/content/338/6110/1042 (visited on 11/06/2019).

[22] Robert L. Baldwin and George D. Rose. "Is protein folding hierarchic? I. Local structure and peptide folding." en. In: *Trends in Biochemical Sciences* 24.1 (Jan. 1999), pp. 26–33. ISSN: 0968-0004. DOI: 10.1016/S0968-0004(98)01346-2. URL: http://www.sciencedirect.com/science/article/pii/S0968000498013462 (visited on 11/07/2019).

[23] Gordon M. Crippen. "The tree structural organization of proteins." en. In: *Journal of Molecular Biology* 126.3 (Dec. 1978), pp. 315–332. ISSN: 0022-2836. DOI: 10.1016/0022-2836(78)90043-8. URL: http://www.sciencedirect.com/science/article/pii/0022283678900438 (visited on 11/07/2019).

[24] Barry Honig. "Protein folding: from the levinthal paradox to structure prediction." en. In: *Journal of Molecular Biology* 293.2 (Oct. 1999), pp. 283–293. ISSN: 0022-2836. DOI: 10.1006/jmbi.1999.3006. URL: http://www.sciencedirect.com/science/article/pii/S0022283699930061 (visited on 11/07/2019).

[25] George D. Rose. "Hierarchic organization of domains in globular proteins." en. In: *Journal of Molecular Biology* 134.3 (Nov. 1979), pp. 447–470. ISSN: 0022-2836. DOI: 10.1016/0022-2836(79)90363-2. URL: http://www.sciencedirect.com/science/article/pii/0022283679903632 (visited on 11/07/2019).

[26] T J Hubbard et al. "SCOP: a Structural Classification of Proteins database." In: *Nucleic Acids Research* 27.1 (Jan. 1999), pp. 254–256. ISSN: 0305-1048. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC148149/ (visited on 11/07/2019).

[27] A. G. Murzin et al. "SCOP: a structural classification of proteins database for the investigation of sequences and structures." eng. In: *Journal of Molecular Biology* 247.4 (Apr. 1995), pp. 536–540. ISSN: 0022-2836. DOI: 10.1006/jmbi.1995.0159.

[28] Konstantinos Sousounis et al. "Conservation of the three-dimensional structure in non-homologous or unrelated proteins." In: *Human Genomics* 6.1 (Aug. 2012), p. 10. ISSN: 1479-7364. DOI: 10.1186/1479-7364-6-10. URL: https://doi.org/10.1186/1479-7364-6-10 (visited on 11/07/2019).

[29] Vladimir N. Uversky and A. Keith Dunker. "Understanding Protein Non-Folding." In: *Biochimica et biophysica acta* 1804.6 (June 2010), pp. 1231–1264. ISSN: 0006-3002. DOI: 10.1016/j.bbapap.2010.01.017. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2882790/ (visited on 11/06/2019).

[30] D. E. Koshland. "Application of a Theory of Enzyme Specificity to Protein Synthesis." en. In: *Proceedings of the National Academy of Sciences* 44.2 (Feb. 1958), pp. 98–104. ISSN: 0027-8424, 1091-6490. DOI: 10.1073/pnas.44.2.98. URL: https://www.pnas.org/content/44/2/98 (visited on 11/06/2019).

[31] Peter E Wright and H. Jane Dyson. "Intrinsically unstructured proteins: re-assessing the protein structure-function paradigm." en. In: *Journal of Molecular Biology* 293.2 (Oct. 1999), pp. 321–331. ISSN: 0022-2836. DOI: 10.1006/jmbi.1999.3110. URL: http://www.sciencedirect.com/science/article/pii/S0022283699931108 (visited on 11/06/2019).

[32] Christian von Mering et al. "Comparative assessment of large-scale data sets of protein–protein interactions." en. In: *Nature* 417.6887 (May 2002), pp. 399–403. ISSN: 1476-4687. DOI: 10.1038/nature750. URL: https://www.nature.com/articles/nature750 (visited on 01/29/2020).

[33] Sumeet Agarwal et al. "Revisiting Date and Party Hubs: Novel Approaches to Role Assignment in Protein Interaction Networks." In: *PLOS Computational Biology* 6.6 (June 2010), pp. 1–12. DOI: 10.1371/journal.pcbi.1000817. URL: https://doi.org/10.1371/journal.pcbi.1000817.

[34] C Chothia and A M Lesk. "The relation between the divergence of sequence and structure in proteins." In: *The EMBO Journal* 5.4 (Apr. 1986), pp. 823–826. ISSN: 0261-4189. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1166865/ (visited on 11/07/2019).

[35] Kristoffer Illergård, David H. Ardell, and Arne Elofsson. "Structure is three to ten times more conserved than sequence–a study of structural response in protein cores." eng. In: *Proteins* 77.3 (Nov. 2009), pp. 499–508. ISSN: 1097-0134. DOI: 10.1002/prot.22458.

[36] Hin Hark Gan et al. "Analysis of Protein Sequence/Structure Similarity Relationships." en. In: *Biophysical Journal* 83.5 (Nov. 2002), pp. 2781–2791. ISSN: 0006-3495. DOI: 10.1016/S0006-3495(02)75287-9. URL: http://www.sciencedirect.com/science/article/pii/S0006349502752879 (visited on 01/17/2020).

[37] Cyrus Chothia and Julian Gough. "Genomic and structural aspects of protein evolution." en. In: *Biochemical Journal* 419.1 (Apr. 2009), pp. 15–28. ISSN: 0264-6021. DOI: 10.1042/BJ20090122. URL: /biochemj/article/419/1/15/44959/Genomic-and-structural-aspects-of-protein (visited on 11/07/2019).

[38] Helen M. Berman et al. "The Protein Data Bank." In: *Nucleic Acids Research* 28.1 (Jan. 2000), pp. 235–242. ISSN: 0305-1048. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC102472/ (visited on 11/08/2019).

[39] Dennis A. Benson et al. "GenBank." eng. In: *Nucleic Acids Research* 41.Database issue (Jan. 2013), pp. D36–42. ISSN: 1362-4962. DOI: 10.1093/nar/gks1195.

[40] Nicholas J Schork, Daniele Fallin, and Jerry S Lanchbury. "Single nucleotide polymorphisms and the future of genetic epidemiology." In: *Clinical Genetics* 58.4 (Oct. 2000), pp. 250–264. ISSN: 0009-9163. DOI: 10.1034/j.1399-0004.2000.580402.x. URL: https://onlinelibrary.wiley.com/doi/full/10.1034/j.1399-0004.2000.580402.x (visited on 11/07/2019).

[41] Lars Feuk, Andrew R. Carson, and Stephen W. Scherer. "Structural variation in the human genome." en. In: *Nature Reviews Genetics* 7.2 (Feb. 2006), pp. 85–97. ISSN: 1471-0064. DOI: 10.1038/nrg1767. URL: https://www.nature.com/articles/nrg1767 (visited on 11/08/2019).

[42] "A global reference for human genetic variation." In: *Nature* 526.7571 (Oct. 2015), pp. 68–74. ISSN: 0028-0836. DOI: 10.1038/nature15393. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4750478/ (visited on 11/08/2019).

[43] "A map of human genome variation from population scale sequencing." In: *Nature* 467.7319 (Oct. 2010), pp. 1061–1073. ISSN: 0028-0836. DOI: 10.1038/nature09534. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3042601/ (visited on 11/08/2019).

[44] Leonid Kruglyak and Deborah A. Nickerson. "Variation is the spice of life." en. In: *Nature Genetics* 27.3 (Mar. 2001), pp. 234–236. ISSN: 1546-1718. DOI: 10.1038/85776. URL: https://www.nature.com/articles/ng0301_234 (visited on 11/08/2019).

[45] S. T. Sherry et al. "dbSNP: the NCBI database of genetic variation." In: *Nucleic Acids Research* 29.1 (Jan. 2001), pp. 308–311. ISSN: 0305-1048. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC29783/ (visited on 02/20/2019).

[46] "A map of human genome variation from population scale sequencing." In: *Nature* 467.7319 (Oct. 2010), pp. 1061–1073. ISSN: 0028-0836. DOI: 10.1038/nature09534. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3042601/ (visited on 11/07/2019).

[47] Monkol Lek et al. "Analysis of protein-coding genetic variation in 60,706 humans." In: *Nature* 536.7616 (Aug. 2016), pp. 285–291. ISSN: 0028-0836. DOI: 10.1038/nature19057. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5018207/ (visited on 11/11/2019).

[48] Tjaart A. P. de Beer et al. "Amino Acid Changes in Disease-Associated Variants Differ Radically from Variants Observed in the 1000 Genomes Project Dataset." In: *PLoS Computational Biology* 9.12 (Dec. 2013). ISSN: 1553-734X. DOI: 10.1371/journal.pcbi.1003382. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3861039/ (visited on 11/08/2019).

[49] Tugba G Kucukkal et al. "Structural and physico-chemical effects of disease and non-disease nsSNPs on proteins." In: *Current Opinion in Structural Biology*. New constructs and expression of proteins / Sequences and topology 32 (June 2015), pp. 18–24. ISSN: 0959-440X. DOI: 10.1016/j.sbi.2015.01.003. URL: http://www.sciencedirect.com/science/article/pii/S0959440X15000044 (visited on 09/02/2019).

[50] Marharyta Petukh, Tugba G Kucukkal, and Emil Alexov. "On human disease-causing amino acid variants: statistical study of sequence and structural patterns." In: *Human mutation* 36.5 (May 2015), pp. 524–534. ISSN: 1059-7794. DOI: 10.1002/humu.22770. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4409542/ (visited on 11/08/2019).

[51] Nidhi Sahni et al. "Widespread Macromolecular Interaction Perturbations in Human Genetic Disorders." In: *Cell* 161.3 (Apr. 2015), pp. 647–660. ISSN: 0092-8674. DOI: 10.1016/j.cell.2015.04.013. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4441215/ (visited on 11/08/2019).

[52] Johan Henrik Wanscher. "An analysis of Wilhelm Johannsen's genetical genotype "term" 1909–26." en. In: *Hereditas* 79.1 (1975), pp. 1–4. ISSN: 1601-5223. DOI: 10.1111/j.1601-5223.1975.tb01456.x. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1601-5223.1975.tb01456.x (visited on 11/08/2019).

[53] Luca Grumolato and Stuart A. Aaronson. "2 - Oncogenes and Signal Transduction." In: *The Molecular Basis of Cancer (Fourth Edition)*. Ed. by John Mendelsohn et al. Fourth Edition. Philadelphia: Content Repository Only! 2015, 19 –34.e3. ISBN: 978-1-4557-4066-6. DOI: https://doi.org/10.1016/B978-1-4557-4066-6.00002-0. URL: http://www.sciencedirect.com/science/article/pii/B9781455740666000020.

[54] David N. Cooper et al. "Where genotype is not predictive of phenotype: towards an understanding of the molecular basis of reduced penetrance in human inherited disease." en. In: *Human Genetics* 132.10 (Oct. 2013), pp. 1077–1130. ISSN: 1432-1203. DOI: 10.1007/s00439-013-1331-2. URL: https://doi.org/10.1007/s00439-013-1331-2 (visited on 11/08/2019).

[55] Joanna S. Amberger et al. "OMIM.org: Online Mendelian Inheritance in Man (OMIM®), an online catalog of human genes and genetic disorders." In: *Nucleic Acids Research* 43.Database issue (Jan. 2015), pp. D789–D798. ISSN: 0305-1048. DOI: 10.1093/nar/gku1205. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4383985/ (visited on 11/08/2019).

[56] Maria Jackson et al. "The genetic basis of disease." In: *Essays in Biochemistry* 62.5 (Dec. 2018), pp. 643–723. ISSN: 0071-1365. DOI: 10.1042/EBC20170053. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6279436/ (visited on 01/17/2020).

[57] Stylianos E. Antonarakis and Jacques S. Beckmann. "Mendelian disorders deserve more attention." en. In: *Nature Reviews Genetics* 7.4 (Apr. 2006), pp. 277–282. ISSN: 1471-0064. DOI: 10.1038/nrg1826. URL: https://www.nature.com/articles/nrg1826 (visited on 01/14/2020).

[58] William C. Hahn and Robert A. Weinberg. "1 - Cancer: A Genetic Disorder." In: *The Molecular Basis of Cancer (Fourth Edition)*. Ed. by John Mendelsohn et al. Fourth Edition. Philadelphia: Content Repository Only! 2015, 3 –18.e1. ISBN: 978-1-4557-4066-6. DOI: https://doi.org/10.1016/B978-1-4557-4066-6.00001-9. URL: http://www.sciencedirect.com/science/article/pii/B9781455740666000019.

[59] Hasan Korkaya, April Davis, and Max S. Wicha. "10 - Cancer Stem Cells and the Microenvironment." In: *The Molecular Basis of Cancer (Fourth Edition)*. Ed. by John Mendelsohn et al. Fourth Edition. Philadelphia: Content Repository Only! 2015, 157 –164.e3. ISBN: 978-1-4557-4066-6. DOI: https://doi.org/10.1016/B978-1-4557-4066-6.00010-X. URL: http://www.sciencedirect.com/science/article/pii/B978145574066600010X.

[60] Swarnali Acharyya et al. "18 - Invasion and Metastasis." In: *The Molecular Basis of Cancer (Fourth Edition)*. Ed. by John Mendelsohn et al. Fourth Edition. Philadelphia: Content Repository Only! 2015, 269 –284.e2. ISBN: 978-1-4557-4066-6. DOI: https://doi.org/10.1016/B978-1-4557-4066-6.00018-4. URL: http://www.sciencedirect.com/science/article/pii/B9781455740666000184.

[61] Douglas Hanahan and Robert A Weinberg. "The Hallmarks of Cancer." en. In: *Cell* 100.1 (Jan. 2000), pp. 57–70. ISSN: 0092-8674. DOI: 10.1016/S0092-8674(00)81683-9. URL: http://www.sciencedirect.com/science/article/pii/S0092867400816839 (visited on 11/08/2019).

[62] Douglas Hanahan and Robert A. Weinberg. "Hallmarks of Cancer: The Next Generation." en. In: *Cell* 144.5 (Mar. 2011), pp. 646–674. ISSN: 0092-8674. DOI: 10.1016/j.cell.2011.02.013. URL: http://www.sciencedirect.com/science/article/pii/S0092867411001279 (visited on 11/08/2019).

[63] Melissa J. Landrum et al. "ClinVar: public archive of interpretations of clinically relevant variants." In: *Nucleic Acids Research* 44.Database issue (Jan. 2016), pp. D862–D868. ISSN: 0305-1048. DOI: 10.1093/nar/gkv1222. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4702865/ (visited on 02/20/2019).

[64] F. Sanger, S. Nicklen, and A. R. Coulson. "DNA sequencing with chain-terminating inhibitors." en. In: *Proceedings of the National Academy of Sciences* 74.12 (Dec. 1977), pp. 5463–5467. ISSN: 0027-8424, 1091-6490. DOI:

10.1073/pnas.74.12.5463. URL: https://www.pnas.org/content/74/12/5463 (visited on 01/21/2020).

[65] R J Milner and J G Sutcliffe. "Gene expression in rat brain." In: *Nucleic Acids Research* 11.16 (Aug. 1983), pp. 5497–5520. ISSN: 0305-1048. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC326294/ (visited on 01/09/2020).

[66] S. D. Putney, W. C. Herlihy, and P. Schimmel. "A new troponin T and cDNA clones for 13 different muscle proteins, found by shotgun sequencing." eng. In: *Nature* 302.5910 (Apr. 1983), pp. 718–721. ISSN: 0028-0836. DOI: 10.1038/302718a0.

[67] Elaine R. Mardis. "Next-Generation Sequencing Platforms." In: *Annual Review of Analytical Chemistry* 6.1 (June 2013), pp. 287–303. ISSN: 1936-1327. DOI: 10.1146/annurev-anchem-062012-092628. URL: https://www.annualreviews.org/doi/10.1146/annurev-anchem-062012-092628 (visited on 01/21/2020).

[68] "A physical map of the human genome." en. In: *Nature* 409.6822 (Feb. 2001), pp. 934–941. ISSN: 1476-4687. DOI: 10.1038/35057157. URL: https://www.nature.com/articles/35057157 (visited on 01/09/2020).

[69] "Initial sequencing and analysis of the human genome." en. In: *Nature* 409.6822 (Feb. 2001), pp. 860–921. ISSN: 1476-4687. DOI: 10.1038/35057062. URL: https://www.nature.com/articles/35057062 (visited on 01/09/2020).

[70] International Human Genome Sequencing Consortium. "Finishing the euchromatic sequence of the human genome." en. In: *Nature* 431.7011 (Oct. 2004), pp. 931–945. ISSN: 1476-4687. DOI: 10.1038/nature03001. URL: https://www.nature.com/articles/nature03001 (visited on 12/06/2019).

[71] Sara Goodwin, John D. McPherson, and W. Richard McCombie. "Coming of age: ten years of next-generation sequencing technologies." en. In: *Nature Reviews Genetics* 17.6 (June 2016), pp. 333–351. ISSN: 1471-0064. DOI: 10.1038/nrg.2016.49. URL: https://www.nature.com/articles/nrg.2016.49 (visited on 01/21/2020).

[72] Anton Valouev et al. "A high-resolution, nucleosome position map of C. elegans reveals a lack of universal sequence-dictated positioning." en. In: *Genome Research* 18.7 (July 2008), pp. 1051–1063. ISSN: 1088-9051, 1549-5469. DOI: 10.1101/gr.076463.108. URL: http://genome.cshlp.org/content/18/7/1051 (visited on 01/21/2020).

[73] Shawn E. Levy and Richard M. Myers. "Advancements in Next-Generation Sequencing." In: *Annual Review of Genomics and Human Genetics* 17.1 (Aug. 2016), pp. 95–115. ISSN: 1527-8204. DOI: 10.1146/annurev-genom-083115-022413. URL: https://www.annualreviews.org/doi/10.1146/annurev-genom-083115-022413 (visited on 01/09/2020).

[74] J. C. Kendrew et al. "A Three-Dimensional Model of the Myoglobin Molecule Obtained by X-Ray Analysis." en. In: *Nature* 181.4610 (Mar. 1958), pp. 662–666. ISSN: 1476-4687. DOI: 10.1038/181662a0. URL: https://www.nature.com/articles/181662a0 (visited on 01/14/2020).

[75] Xiao-chen Bai, Greg McMullan, and Sjors H. W Scheres. "How cryo-EM is revolutionizing structural biology." en. In: *Trends in Biochemical Sciences* 40.1 (Jan. 2015), pp. 49–57. ISSN: 0968-0004. DOI: 10.1016/j.tibs.2014.10.005. URL: http://www.sciencedirect.com/science/article/pii/S096800041400187X (visited on 01/17/2020).

[76] Dominika Elmlund and Hans Elmlund. "Cryogenic Electron Microscopy and Single-Particle Analysis." In: *Annual Review of Biochemistry* 84.1 (2015), pp. 499–517. DOI: 10.1146/annurev-biochem-060614-034226. URL: https://doi.org/10.1146/annurev-biochem-060614-034226 (visited on 01/17/2020).

[77] Ilkka Lappalainen et al. "dbVar and DGVa: public archives for genomic structural variation." In: *Nucleic Acids Research* 41.Database issue (Jan. 2013), pp. D936–D941. ISSN: 0305-1048. DOI: 10.1093/nar/gks1213. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3531204/ (visited on 02/20/2019).

[78] Simon A. Forbes et al. "COSMIC: somatic cancer genetics at high-resolution." In: *Nucleic Acids Research* 45.Database issue (Jan. 2017), pp. D777–D783. ISSN: 0305-1048. DOI: 10.1093/nar/gkw1121. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5210583/ (visited on 02/20/2019).

[79] Peter D. Stenson et al. "The Human Gene Mutation Database: towards a comprehensive repository of inherited mutation data for medical research, genetic diagnosis and next-generation sequencing studies." In: *Human Genetics* 136.6 (2017), pp. 665–677. ISSN: 0340-6717. DOI: 10.1007/s00439-017-1779-6. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5429360/ (visited on 02/20/2019).

[80] N. del Toro et al. "Capturing variation impact on molecular interactions in the IMEx Consortium mutations data set." en. In: *Nature Communications* 10.1 (Jan. 2019), p. 10. ISSN: 2041-1723. DOI: 10.1038/s41467-018-07709-6. URL: https://www.nature.com/articles/s41467-018-07709-6 (visited on 02/27/2019).

[81] Justina Jankauskaitė et al. "SKEMPI 2.0: an updated benchmark of changes in protein–protein binding energy, kinetics and thermodynamics upon mutation." en. In: *Bioinformatics* 35.3 (Feb. 2019), pp. 462–469. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/bty635. URL: https://academic.oup.com/bioinformatics/article/35/3/462/5055583 (visited on 02/19/2019).

[82] Douglas M. Fowler and Stanley Fields. "Deep mutational scanning: a new style of protein science." In: *Nature methods* 11.8 (Aug. 2014), pp. 801–807. ISSN: 1548-7091. DOI: 10.1038/nmeth.3027. URL: https:

//www.ncbi.nlm.nih.gov/pmc/articles/PMC4410700/ (visited on 01/10/2020).

[83] Stephen F. Altschul et al. "Basic local alignment search tool." en. In: *Journal of Molecular Biology* 215.3 (Oct. 1990), pp. 403–410. ISSN: 0022-2836. DOI: 10.1016/S0022-2836(05)80360-2. URL: http://www.sciencedirect.com/science/article/pii/S0022283605803602 (visited on 11/08/2019).

[84] S. R. Eddy. "Profile hidden Markov models." en. In: *Bioinformatics* 14.9 (Jan. 1998), pp. 755–763. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/14.9.755. URL: https://academic.oup.com/bioinformatics/article/14/9/755/259550 (visited on 01/09/2020).

[85] Robert D. Finn, Jody Clements, and Sean R. Eddy. "HMMER web server: interactive sequence similarity searching." en. In: *Nucleic Acids Research* 39.suppl_2 (July 2011), W29–W37. ISSN: 0305-1048. DOI: 10.1093/nar/gkr367. URL: https://academic.oup.com/nar/article/39/suppl_2/W29/2506513 (visited on 01/09/2020).

[86] Martin Steinegger and Johannes Söding. "MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets." en. In: *Nature Biotechnology* 35 (Oct. 2017), pp. 1026–1028. ISSN: 1546-1696. DOI: 10.1038/nbt.3988. URL: https://www.nature.com/articles/nbt.3988 (visited on 05/15/2019).

[87] M. O. Dayhoff and R. M. Schwartz. "Chapter 22: A model of evolutionary change in proteins." In: *in Atlas of Protein Sequence and Structure*. 1978.

[88] S Henikoff and J G Henikoff. "Amino acid substitution matrices from protein blocks." In: *Proceedings of the National Academy of Sciences of the United States of America* 89.22 (Nov. 1992), pp. 10915–10919. ISSN: 0027-8424. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC50453/ (visited on 11/11/2019).

[89] Benjamin Webb and Andrej Sali. "Comparative Protein Structure Modeling Using MODELLER." en. In: *Current Protocols in Protein Science* 86.1 (2016), pp. 2.9.1–2.9.37. ISSN: 1934-3663. DOI: 10.1002/cpps.20. URL: https://currentprotocols.onlinelibrary.wiley.com/doi/abs/10.1002/cpps.20 (visited on 01/09/2020).

[90] Andrew Waterhouse et al. "SWISS-MODEL: homology modelling of protein structures and complexes." In: *Nucleic Acids Research* 46.Web Server issue (July 2018), W296–W303. ISSN: 0305-1048. DOI: 10.1093/nar/gky427. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6030848/ (visited on 01/09/2020).

[91] Brian Kuhlman and Philip Bradley. "Advances in protein structure prediction and design." en. In: *Nature Reviews Molecular Cell Biology* 20.11 (Nov. 2019), pp. 681–697. ISSN: 1471-0080. DOI: 10.1038/s41580-019-0163-x. URL: https://www.nature.com/articles/s41580-019-0163-x (visited on 01/21/2020).

[92] Jianyi Yang and Yang Zhang. "I-TASSER server: new development for protein structure and function predictions." eng. In: *Nucleic acids research* 43.W1 (July 2015), W174–W181. ISSN: 1362-4962. DOI: 10.1093/nar/gkv342. URL: https://pubmed.ncbi.nlm.nih.gov/25883148.

[93] Sergey Ovchinnikov et al. "Large-scale determination of previously unsolved protein structures using evolutionary information." In: *eLife* 4 (). ISSN: 2050-084X. DOI: 10.7554/eLife.09248. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4602095/ (visited on 06/25/2019).

[94] Jinbo Xu and Sheng Wang. "Analysis of distance-based protein structure prediction by deep learning in CASP13." eng. In: *Proteins* (Aug. 2019). ISSN: 1097-0134. DOI: 10.1002/prot.25810.

[95] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The elements of statistical learning: data mining, inference and prediction*. 2nd ed. Springer, 2009. URL: http://www-stat.stanford.edu/~tibs/ElemStatLearn/.

[96] Michel F. Sanner, Arthur J. Olson, and Jean-Claude Spehner. "Reduced surface: An efficient way to compute molecular surfaces." en. In: *Biopolymers* 38.3 (1996), pp. 305–320. ISSN: 1097-0282. DOI: 10.1002/(SICI)1097-0282(199603)38:3<305::AID-BIP4>3.0.CO;2-Y. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/%28SICI%291097-0282%28199603%2938%3A3%3C305%3A%3AAID-BIP4%3E3.0.CO%3B2-Y (visited on 06/19/2019).

[97] Burkhard Rost and Chris Sander. "Conservation and prediction of solvent accessibility in protein families." en. In: *Proteins: Structure, Function, and Bioinformatics* 20.3 (1994), pp. 216–226. ISSN: 1097-0134. DOI: 10.1002/prot.340200303. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/prot.340200303 (visited on 06/19/2019).

[98] Patrick Aloy et al. "Automated structure-based prediction of functional sites in proteins: applications to assessing the validity of inheriting protein function from homology in genome annotation and to protein docking11Edited by G. von Heijne." In: *Journal of Molecular Biology* 311.2 (Aug. 2001), pp. 395–408. ISSN: 0022-2836. DOI: 10.1006/jmbi.2001.4870. URL: http://www.sciencedirect.com/science/article/pii/S0022283601948703 (visited on 09/02/2019).

[99] David J. Brockwell et al. "The Effect of Core Destabilization on the Mechanical Resistance of I27." In: *Biophysical Journal* 83.1 (July 2002), pp. 458–472. ISSN: 0006-3495. DOI: 10.1016/S0006-3495(02)75182-5. URL: http://www.sciencedirect.com/science/article/pii/S0006349502751825 (visited on 09/02/2019).

[100] Wendell A. Lim, Dawn C. Farruggio, and Robert T. Sauer. "Structural and energetic consequences of disruptive mutations in a protein core." en. In: *Biochemistry* 31.17 (May 1992), pp. 4324–4333. ISSN: 0006-2960, 1520-4995. DOI: 10.1021/bi00132a025. URL: https://pubs.acs.org/doi/abs/10.1021/bi00132a025 (visited on 09/02/2019).

[101] Sophie E. Jackson et al. "Effect of cavity-creating mutations in the hydrophobic core of chymotrypsin inhibitor 2." en. In: *Biochemistry* 32.42 (Oct. 1993), pp. 11259–11269. ISSN: 0006-2960, 1520-4995. DOI: 10.1021/bi00093a001. URL: https://pubs.acs.org/doi/abs/10.1021/bi00093a001 (visited on 09/02/2019).

[102] Alex N. Bullock et al. "Thermodynamic stability of wild-type and mutant p53 core domain." en. In: *Proceedings of the National Academy of Sciences* 94.26 (Dec. 1997), pp. 14338–14342. ISSN: 0027-8424, 1091-6490. DOI: 10.1073/pnas.94.26.14338. URL: https://www.pnas.org/content/94/26/14338 (visited on 09/02/2019).

[103] Song Yi et al. "Functional variomics and network perturbation: connecting genotype to phenotype in cancer." In: *Nature reviews. Genetics* 18.7 (July 2017), pp. 395–410. ISSN: 1471-0056. DOI: 10.1038/nrg.2017.8. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6020840/ (visited on 06/19/2019).

[104] Alexey Drozdetskiy et al. "JPred4: a protein secondary structure prediction server." In: *Nucleic Acids Research* 43.Web Server issue (July 2015), W389–W394. ISSN: 0305-1048. DOI: 10.1093/nar/gkv332. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4489285/ (visited on 01/14/2020).

[105] Rhys Heffernan et al. "Single-sequence-based prediction of protein secondary structures and solvent accessibility by deep whole-sequence learning." en. In: *Journal of Computational Chemistry* 39.26 (2018), pp. 2210–2216. ISSN: 1096-987X. DOI: 10.1002/jcc.25534. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/jcc.25534 (visited on 01/14/2020).

[106] Michael Schantz Klausen et al. "NetSurfP-2.0: Improved prediction of protein structural features by integrated deep learning." en. In: *Proteins: Structure, Function, and Bioinformatics* 87.6 (2019), pp. 520–527. ISSN: 1097-0134. DOI: 10.1002/prot.25674. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/prot.25674 (visited on 01/14/2020).

[107] Sheng Wang et al. "RaptorX-Property: a web server for protein structure property prediction." In: *Nucleic Acids Research* 44.Web Server issue (July 2016), W430–W435. ISSN: 0305-1048. DOI: 10.1093/nar/gkw306. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4987890/ (visited on 01/14/2020).

[108] Kim T. Simons et al. "Assembly of protein tertiary structures from fragments with similar local sequences using simulated annealing and bayesian scoring functions11Edited by F. E. Cohen." In: *Journal of Molecular Biology* 268.1 (Apr. 1997), pp. 209–225. ISSN: 0022-2836. DOI: 10.1006/jmbi.1997.0959. URL: http://www.sciencedirect.com/science/article/pii/S0022283697909591 (visited on 06/19/2019).

[109] Thomas Hamelryck. "An amino acid has two sides: A new 2D measure provides a different view of solvent exposure." en. In: *Proteins: Structure, Function, and Bioinformatics* 59.1 (2005), pp. 38–48. ISSN: 1097-0134. DOI:

10.1002/prot.20379. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/prot.20379 (visited on 06/19/2019).

[110] Jack Hanson et al. "Improving prediction of protein secondary structure, backbone angles, solvent accessibility and contact numbers by using predicted contact maps and an ensemble of recurrent and residual convolutional neural networks." en. In: *Bioinformatics* 35.14 (July 2019), pp. 2403–2410. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/bty1006. URL: https://academic.oup.com/bioinformatics/article/35/14/2403/5232996 (visited on 10/09/2019).

[111] Andriy Kryshtafovych et al. "Critical assessment of methods of protein structure prediction (CASP)—Round XIII." en. In: *Proteins: Structure, Function, and Bioinformatics* 87.12 (2019), pp. 1011–1020. ISSN: 1097-0134. DOI: 10.1002/prot.25823. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/prot.25823 (visited on 01/22/2020).

[112] Arti Singh et al. "MutDB: update on development of tools for the biochemical analysis of genetic variation." In: *Nucleic Acids Research* 36.Database issue (Jan. 2008), pp. D815–D819. ISSN: 0305-1048. DOI: 10.1093/nar/gkm659. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2238958/ (visited on 11/08/2019).

[113] Antonia Stank, Stefan Richter, and Rebecca C. Wade. "ProSAT+: visualizing sequence annotations on 3D structure." en. In: *Protein Engineering, Design and Selection* 29.8 (Aug. 2016), pp. 281–284. ISSN: 1741-0126. DOI: 10.1093/protein/gzw021. URL: https://academic.oup.com/peds/article/29/8/281/2223264 (visited on 02/03/2020).

[114] Patrick Aloy et al. "The Relationship Between Sequence and Interaction Divergence in Proteins." en. In: *Journal of Molecular Biology* 332.5 (Oct. 2003), pp. 989–998. ISSN: 0022-2836. DOI: 10.1016/j.jmb.2003.07.006. URL: http://www.sciencedirect.com/science/article/pii/S0022283603009999 (visited on 11/08/2019).

[115] Matthew J. Betts et al. "Mechismo: predicting the mechanistic impact of mutations and modifications on molecular interactions." In: *Nucleic Acids Research* 43.2 (Jan. 2015), e10. ISSN: 0305-1048. DOI: 10.1093/nar/gku1094. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4333368/ (visited on 02/21/2019).

[116] Omar Wagih et al. "A resource of variant effect predictions of single nucleotide variants in model organisms." en. In: *Molecular Systems Biology* 14.12 (Dec. 2018), e8430. ISSN: 1744-4292, 1744-4292. DOI: 10.15252/msb.20188430. URL: http://msb.embopress.org/content/14/12/e8430 (visited on 02/19/2019).

[117] Roberto Mosca et al. "dSysMap: exploring the edgetic role of disease mutations." en. In: *Nature Methods* 12.3 (Mar. 2015), pp. 167–168. ISSN: 1548-7105. DOI: 10.1038/nmeth.3289. URL: https://www.nature.com/articles/nmeth.3289 (visited on 02/21/2019).

[118] Leandro Radusky et al. "VarQ: A Tool for the Structural and Functional Analysis of Human Protein Variants." In: *Frontiers in Genetics* 9 (Dec. 2018). ISSN: 1664-8021. DOI: 10.3389/fgene.2018.00620. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6291447/ (visited on 02/19/2019).

[119] Noushin Niknafs et al. "MuPIT Interactive: Webserver for mapping variant positions to annotated, interactive 3D structures." In: *Human genetics* 132.11 (Nov. 2013), pp. 1235–1243. ISSN: 0340-6717. DOI: 10.1007/s00439-013-1325-0. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3797853/ (visited on 11/08/2019).

[120] Jacob M. Hurst et al. "The SAAPdb web resource: A large-scale structural analysis of mutant proteins." en. In: *Human Mutation* 30.4 (2009), pp. 616–624. ISSN: 1098-1004. DOI: 10.1002/humu.20898. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/humu.20898 (visited on 11/08/2019).

[121] Michael Ryan et al. "LS-SNP/PDB: annotated non-synonymous SNPs mapped to Protein Data Bank structures." In: *Bioinformatics* 25.11 (June 2009), pp. 1431–1432. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btp242. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6276889/ (visited on 01/31/2020).

[122] Peng Yue, Eugene Melamud, and John Moult. "SNPs3D: Candidate gene and SNP selection for association studies." In: *BMC Bioinformatics* 7 (Mar. 2006), p. 166. ISSN: 1471-2105. DOI: 10.1186/1471-2105-7-166. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1435944/ (visited on 11/08/2019).

[123] Oz Solomon et al. "G23D: Online tool for mapping and visualization of genomic variants on 3D protein structures." In: *BMC Genomics* 17.1 (Aug. 2016). ISSN: 1471-2164. DOI: 10.1186/s12864-016-3028-0. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5002099/ (visited on 01/31/2020).

[124] Tien-Dao Luu et al. "MSV3d: database of human MisSense variants mapped to 3D protein structure." In: *Database: The Journal of Biological Databases and Curation* 2012 (Apr. 2012). ISSN: 1758-0463. DOI: 10.1093/database/bas018. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3317913/ (visited on 02/03/2020).

[125] Eduard Porta-Pardo, Thomas Hrabe, and Adam Godzik. "Cancer3D: understanding cancer mutations through protein structures." In: *Nucleic Acids Research* 43.Database issue (Jan. 2015), pp. D968–D973. ISSN: 0305-1048. DOI: 10.1093/nar/gku1140. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4383948/ (visited on 02/03/2020).

[126] Difei Wang et al. "SNP2Structure: A Public and Versatile Resource for Mapping and Three-Dimensional Modeling of Missense SNPs on Human Protein Structures." In: *Computational and Structural Biotechnology Journal* 13 (Sept. 2015), pp. 514–519. ISSN: 2001-0370. DOI: 10.1016/j.

csbj.2015.09.002. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4759123/ (visited on 02/03/2020).

[127] Clinton J. Mielke, Lawrence J. Mandarino, and Valentin Dinu. "AMASS: a database for investigating protein structures." In: *Bioinformatics* 30.11 (June 2014), pp. 1595–1600. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btu073. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4029036/ (visited on 02/03/2020).

[128] Seán I. O'Donoghue et al. "Aquaria: simplifying discovery and insight from protein structures." en. In: *Nature Methods* 12.2 (Feb. 2015), pp. 98–99. ISSN: 1548-7105. DOI: 10.1038/nmeth.3258. URL: https://www.nature.com/articles/nmeth.3258 (visited on 02/03/2020).

[129] Alex Bateman et al. "The Pfam protein families database." In: *Nucleic Acids Research* 32.Database issue (Jan. 2004), pp. D138–D141. ISSN: 0305-1048. DOI: 10.1093/nar/gkh121. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC308855/ (visited on 02/21/2019).

[130] Roberto Mosca et al. "3did: a catalog of domain-based interactions of known three-dimensional structure." In: *Nucleic Acids Research* 42.Database issue (Jan. 2014), pp. D374–D379. ISSN: 0305-1048. DOI: 10.1093/nar/gkt887. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3965002/ (visited on 02/21/2019).

[131] Gary D. Bader, Doron Betel, and Christopher W. V. Hogue. "BIND: the Biomolecular Interaction Network Database." In: *Nucleic Acids Research* 31.1 (Jan. 2003), pp. 248–250. ISSN: 0305-1048. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC165503/ (visited on 01/15/2020).

[132] Rose Oughtred et al. "The BioGRID interaction database: 2019 update." In: *Nucleic Acids Research* 47.Database issue (Jan. 2019), pp. D529–D541. ISSN: 0305-1048. DOI: 10.1093/nar/gky1079. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6324058/ (visited on 01/15/2020).

[133] Ioannis Xenarios et al. "DIP: the Database of Interacting Proteins." In: *Nucleic Acids Research* 28.1 (Jan. 2000), pp. 289–291. ISSN: 0305-1048. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC102387/ (visited on 02/03/2020).

[134] Suraj Peri et al. "Human protein reference database as a discovery resource for proteomics." In: *Nucleic Acids Research* 32.Database issue (Jan. 2004), pp. D497–D501. ISSN: 0305-1048. DOI: 10.1093/nar/gkh070. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC308804/ (visited on 02/03/2020).

[135] David J Lynn et al. "InnateDB: facilitating systems-level analyses of the mammalian innate immune response." In: *Molecular Systems Biology* 4 (Sept. 2008), p. 218. ISSN: 1744-4292. DOI: 10.1038/msb.2008.55. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2564732/ (visited on 02/03/2020).

[136] Andrew Chatr-aryamontri et al. "MINT: the Molecular INTeraction database." In: *Nucleic Acids Research* 35.Database issue (Jan. 2007), pp. D572–D574. ISSN: 0305-1048. DOI: 10.1093/nar/gkl950. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1751541/ (visited on 01/15/2020).

[137] Bálint Mészáros, Gábor Erdős, and Zsuzsanna Dosztányi. "IUPred2A: context-dependent prediction of protein disorder as a function of redox state and protein binding." In: *Nucleic Acids Research* 46.Web Server issue (July 2018), W329–W337. ISSN: 0305-1048. DOI: 10.1093/nar/gky384. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6030935/ (visited on 02/28/2019).

[138] Hayley M Dingerdissen et al. "BioMuta and BioXpress: mutation and expression knowledgebases for cancer biomarker discovery." In: *Nucleic Acids Research* 46.Database issue (Jan. 2018), pp. D1128–D1136. ISSN: 0305-1048. DOI: 10.1093/nar/gkx907. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5753215/ (visited on 01/31/2020).

[139] Maria Livia Famiglietti et al. "Genetic Variations and Diseases in UniProtKB/Swiss-Prot: The Ins and Outs of Expert Manual Curation." In: *Human Mutation* 35.8 (Aug. 2014), pp. 927–935. ISSN: 1059-7794. DOI: 10.1002/humu.22594. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4107114/ (visited on 01/31/2020).

[140] Peter Schmidtke et al. "fpocket: online tools for protein ensemble pocket detection and tracking." eng. In: *Nucleic Acids Research* 38.Web Server issue (July 2010), W582–589. ISSN: 1362-4962. DOI: 10.1093/nar/gkq383.

[141] Joost Schymkowitz et al. "The FoldX web server: an online force field." In: *Nucleic Acids Research* 33.Web Server issue (July 2005), W382–W388. ISSN: 0305-1048. DOI: 10.1093/nar/gki387. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1160148/ (visited on 02/21/2019).

[142] Ana-Maria Fernandez-Escamilla et al. "Prediction of sequence-dependent and mutational effects on the aggregation of peptides and proteins." en. In: *Nature Biotechnology* 22.10 (Oct. 2004), pp. 1302–1306. ISSN: 1546-1696. DOI: 10.1038/nbt1012. URL: https://www.nature.com/articles/nbt1012 (visited on 02/21/2019).

[143] Kim Pruitt et al. *The Reference Sequence (RefSeq) Database*. en. National Center for Biotechnology Information (US), Apr. 2012. URL: https://www.ncbi.nlm.nih.gov/books/NBK21091/ (visited on 02/03/2020).

[144] W. James Kent et al. "The Human Genome Browser at UCSC." en. In: *Genome Research* 12.6 (June 2002), pp. 996–1006. ISSN: 1088-9051, 1549-5469. DOI: 10.1101/gr.229102. URL: http://genome.cshlp.org/content/12/6/996 (visited on 02/03/2020).

[145] Kai Wang, Mingyao Li, and Hakon Hakonarson. "ANNOVAR: functional annotation of genetic variants from high-throughput sequencing data." In: *Nucleic Acids Research* 38.16 (Sept. 2010), e164. ISSN: 0305-1048. DOI: 10.1093/nar/gkq603. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2938201/ (visited on 11/11/2019).

[146] Damiano Piovesan et al. "MobiDB 3.0: more annotations for intrinsic disorder, conformational diversity and interactions in proteins." In: *Nucleic Acids Research* 46.Database issue (Jan. 2018), pp. D471–D476. ISSN: 0305-1048. DOI: 10.1093/nar/gkx1071. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5753340/ (visited on 11/26/2019).

[147] Saul B. Needleman and Christian D. Wunsch. "A general method applicable to the search for similarities in the amino acid sequence of two proteins." en. In: *Journal of Molecular Biology* 48.3 (Mar. 1970), pp. 443–453. ISSN: 0022-2836. DOI: 10.1016/0022-2836(70)90057-4. URL: http://www.sciencedirect.com/science/article/pii/0022283670900574 (visited on 01/15/2020).

[148] Peter J. A. Cock et al. "Biopython: freely available Python tools for computational molecular biology and bioinformatics." en. In: *Bioinformatics* 25.11 (June 2009), pp. 1422–1423. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btp163. URL: https://academic.oup.com/bioinformatics/article/25/11/1422/330687 (visited on 01/15/2020).

[149] Nadezhda T. Doncheva et al. "Analyzing and visualizing residue networks of protein structures." en. In: *Trends in Biochemical Sciences* 36.4 (Apr. 2011), pp. 179–182. ISSN: 0968-0004. DOI: 10.1016/j.tibs.2011.01.002. URL: http://www.sciencedirect.com/science/article/pii/S0968000411000132 (visited on 01/15/2020).

[150] Alexander Gress et al. "StructMAn: annotation of single-nucleotide polymorphisms in the structural context." eng. In: *Nucleic Acids Research* 44.W1 (July 2016), W463–468. ISSN: 1362-4962. DOI: 10.1093/nar/gkw364.

[151] Adam Auton et al. "A global reference for human genetic variation." en. In: *Nature* 526.7571 (Oct. 2015), pp. 68–74. ISSN: 1476-4687. DOI: 10.1038/nature15393. URL: https://www.nature.com/articles/nature15393 (visited on 11/29/2019).

[152] A. Gress, V. Ramensky, and O. V. Kalinina. "Spatial distribution of disease-associated variants in three-dimensional structures of protein complexes." eng. In: *Oncogenesis* 6.9 (Sept. 2017), e380. ISSN: 2157-9024. DOI: 10.1038/oncsis.2017.79.

[153] H. Billur Engin, Jason F. Kreisberg, and Hannah Carter. "Structure-Based Analysis Reveals Cancer Missense Mutations Target Protein Interaction Interfaces." In: *PLoS ONE* 11.4 (Apr. 2016). ISSN: 1932-6203. DOI: 10.1371/journal.pone.0152929. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4820104/ (visited on 11/11/2019).

[154] Atanas Kamburov et al. "Comprehensive assessment of cancer missense mutation clustering in protein structures." In: *Proceedings of the National Academy of Sciences of the United States of America* 112.40 (Oct. 2015), E5486–E5495. ISSN: 0027-8424. DOI: 10.1073/pnas.1516373112. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4603469/ (visited on 11/11/2019).

[155] Eduard Porta-Pardo et al. "A Pan-Cancer Catalogue of Cancer Driver Protein Interaction Interfaces." In: *PLoS Computational Biology* 11.10 (Oct. 2015). ISSN: 1553-734X. DOI: 10.1371/journal.pcbi.1004518. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4616621/ (visited on 01/10/2020).

[156] Martin H. Schaefer, Luis Serrano, and Miguel A. Andrade-Navarro. "Correcting for the study bias associated with protein–protein interaction measurements reveals differences between protein degree distributions from different cancer types." In: *Frontiers in Genetics* 6 (Aug. 2015). ISSN: 1664-8021. DOI: 10.3389/fgene.2015.00260. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4523822/ (visited on 11/11/2019).

[157] Pauline C. Ng and Steven Henikoff. "SIFT: predicting amino acid changes that affect protein function." In: *Nucleic Acids Research* 31.13 (July 2003), pp. 3812–3814. ISSN: 0305-1048. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC168916/ (visited on 04/11/2019).

[158] S. R. Sunyaev et al. "PSIC: profile extraction from sequence alignments with position-specific counts of independent observations." eng. In: *Protein Engineering* 12.5 (May 1999), pp. 387–394. ISSN: 0269-2139. DOI: 10.1093/protein/12.5.387.

[159] Vasundhara Dehiya, Jaya Thomas, and Lee Sael. "Impact of structural prior knowledge in SNV prediction: Towards causal variant finding in rare disease." In: *PLoS ONE* 13.9 (Sept. 2018). ISSN: 1932-6203. DOI: 10.1371/journal.pone.0204101. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6161878/ (visited on 02/19/2019).

[160] Karthik A. Jagadeesh et al. "M-CAP eliminates a majority of variants of uncertain significance in clinical exomes at high sensitivity." en. In: *Nature Genetics* 48.12 (Dec. 2016), pp. 1581–1586. ISSN: 1546-1718. DOI: 10.1038/ng.3703. URL: https://www.nature.com/articles/ng.3703 (visited on 11/28/2019).

[161] Douglas E. V. Pires, David B. Ascher, and Tom L. Blundell. "mCSM: predicting the effects of mutations in proteins using graph-based signatures." In: *Bioinformatics* 30.3 (Feb. 2014), pp. 335–342. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btt691. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3904523/ (visited on 01/20/2020).

[162] Douglas E. V. Pires, Tom L. Blundell, and David B. Ascher. "mCSM-lig: quantifying the effects of mutations on protein-small molecule affinity in genetic disease and emergence of drug resistance." In: *Scientific Reports* 6 (July 2016). ISSN: 2045-2322. DOI: 10.1038/srep29575. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4935856/ (visited on 02/20/2019).

[163] Douglas E.V. Pires and David B. Ascher. "mCSM-AB: a web server for predicting antibody–antigen affinity changes upon mutation with graph-based signatures." In: *Nucleic Acids Research* 44.Web Server issue (July 2016), W469–W473. ISSN: 0305-1048. DOI: 10.1093/nar/gkw458. URL:

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4987957/ (visited on 02/20/2019).

[164] Carlos H. M. Rodrigues et al. "mCSM-PPI2: predicting the effects of mutations on protein–protein interactions." en. In: *Nucleic Acids Research* 47.W1 (July 2019), W338–W344. ISSN: 0305-1048. DOI: 10.1093/nar/gkz383. URL: https://academic.oup.com/nar/article/47/W1/W338/5494729 (visited on 10/14/2019).

[165] Boris Reva, Yevgeniy Antipin, and Chris Sander. "Predicting the functional impact of protein mutations: application to cancer genomics." In: *Nucleic Acids Research* 39.17 (Sept. 2011), e118. ISSN: 0305-1048. DOI: 10.1093/nar/gkr407. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3177186/ (visited on 02/20/2019).

[166] Jana Marie Schwarz et al. "MutationTaster2: mutation prediction for the deep-sequencing age." en. In: *Nature Methods* 11.4 (Apr. 2014), pp. 361–362. ISSN: 1548-7105. DOI: 10.1038/nmeth.2890. URL: https://www.nature.com/articles/nmeth.2890 (visited on 11/11/2019).

[167] Dominik G. Grimm et al. "The Evaluation of Tools Used to Predict the Impact of Missense Variants Is Hindered by Two Types of Circularity." In: *Human Mutation* 36.5 (2015), pp. 513–523. ISSN: 1098-1004. DOI: 10.1002/humu.22768. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/humu.22768 (visited on 02/25/2019).

[168] Miao-Xin Li et al. "Predicting Mendelian Disease-Causing Non-Synonymous Single Nucleotide Variants in Exome Sequencing Studies." In: *PLoS Genetics* 9.1 (Jan. 2013). ISSN: 1553-7390. DOI: 10.1371/journal.pgen.1003143. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3547823/ (visited on 02/12/2020).

[169] Preethy Sasidharan Nair and Mauno Vihinen. "VariBench: A Benchmark Database for Variations." en. In: *Human Mutation* 34.1 (2013), pp. 42–49. ISSN: 1098-1004. DOI: 10.1002/humu.22204. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/humu.22204 (visited on 02/12/2020).

[170] Janita Thusberg, Ayodeji Olatubosun, and Mauno Vihinen. "Performance of mutation pathogenicity prediction methods on missense variants." en. In: *Human Mutation* 32.4 (2011), pp. 358–368. ISSN: 1098-1004. DOI: 10.1002/humu.21445. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/humu.21445 (visited on 11/11/2019).

[171] Jaroslav Bendl et al. "PredictSNP: Robust and Accurate Consensus Classifier for Prediction of Disease-Related Mutations." In: *PLoS Computational Biology* 10.1 (Jan. 2014). ISSN: 1553-734X. DOI: 10.1371/journal.pcbi.1003440. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3894168/ (visited on 02/12/2020).

[172] Emidio Capriotti et al. "WS-SNPs&GO: a web server for predicting the deleterious effect of human protein variants using functional annotation." In: *BMC Genomics* 14.Suppl 3 (May 2013), S6. ISSN: 1471-2164. DOI: 10.1186/1471-2164-14-S3-S6. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3665478/ (visited on 11/08/2019).

[173] Biao Li et al. "Automated inference of molecular mechanisms of disease from amino acid substitutions." In: *Bioinformatics* 25.21 (Nov. 2009), pp. 2744–2750. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btp528. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3140805/ (visited on 02/20/2020).

[174] Ayodeji Olatubosun et al. "PON-P: Integrated predictor for pathogenicity of missense variants." In: *Human Mutation* 33.8 (2012), pp. 1166–1174. DOI: 10.1002/humu.22102. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/humu.22102. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/humu.22102.

[175] Anaïs Mottaz et al. "Mapping proteins to disease terminologies: from UniProt to MeSH." In: *BMC Bioinformatics* 9.Suppl 5 (Apr. 2008), S3. ISSN: 1471-2105. DOI: 10.1186/1471-2105-9-S5-S3. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2367626/ (visited on 02/12/2020).

[176] Yum L. Yip et al. "The Swiss-Prot variant page and the ModSNP database: A resource for sequence and structure information on human protein variants." en. In: *Human Mutation* 23.5 (2004), pp. 464–470. ISSN: 1098-1004. DOI: 10.1002/humu.20021. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/humu.20021 (visited on 02/12/2020).

[177] Yves Dehouck et al. "PoPMuSiC 2.1: a web server for the estimation of protein stability changes upon mutation and sequence optimality." In: *BMC Bioinformatics* 12 (May 2011), p. 151. ISSN: 1471-2105. DOI: 10.1186/1471-2105-12-151. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3113940/ (visited on 01/23/2020).

[178] Lijun Quan, Qiang Lv, and Yang Zhang. "STRUM: structure-based prediction of protein stability changes upon single-point mutation." In: *Bioinformatics* 32.19 (Oct. 2016), pp. 2936–2946. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btw361. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5039926/ (visited on 02/20/2019).

[179] Castrense Savojardo et al. "INPS-MD: a web server to predict stability of protein variants from sequence and structure." en. In: *Bioinformatics* 32.16 (Aug. 2016), pp. 2542–2544. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btw192. URL: https://academic.oup.com/bioinformatics/article/32/16/2542/1743481 (visited on 02/20/2019).

[180] Yves Dehouck et al. "BeAtMuSiC: prediction of changes in protein–protein binding affinity on mutations." In: *Nucleic Acids Research* 41.Web Server issue (July 2013), W333–W339. ISSN: 0305-1048. DOI: 10.1093/nar/gkt450. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3692068/ (visited on 01/23/2020).

[181] Minghui Li et al. "MutaBind estimates and interprets the effects of sequence variants on protein–protein interactions." In: *Nucleic Acids Research* 44.Web Server issue (July 2016), W494–W501. ISSN: 0305-1048. DOI: 10.1093/nar/gkw374. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4987923/ (visited on 02/20/2019).

[182] Peng Xiong et al. "BindProfX: Assessing Mutation-Induced Binding Affinity Change by Protein Interface Profiles with Pseudo-Counts." In: *Journal of molecular biology* 429.3 (Feb. 2017), pp. 426–434. ISSN: 0022-2836. DOI: 10.1016/j.jmb.2016.11.022. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5963940/ (visited on 02/20/2019).

[183] Daniel K. Witvliet et al. "ELASPIC web-server: proteome-wide structure-based prediction of mutation effects on protein stability and binding affinity." en. In: *Bioinformatics* 32.10 (May 2016), pp. 1589–1591. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btw031. URL: https://academic.oup.com/bioinformatics/article/32/10/1589/1743335 (visited on 02/20/2019).

[184] Vanessa E. Gray, Ronald J. Hause, and Douglas M. Fowler. "Analysis of Large-Scale Mutagenesis Data To Assess the Impact of Single Amino Acid Substitutions." In: *Genetics* 207.1 (Sept. 2017), pp. 53–61. ISSN: 0016-6731. DOI: 10.1534/genetics.117.300064. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5586385/ (visited on 04/11/2019).

[185] Anaïs Mottaz et al. "Easy retrieval of single amino-acid polymorphisms and phenotype information using SwissVar." In: *Bioinformatics* 26.6 (Mar. 2010), pp. 851–852. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btq028. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2832822/ (visited on 02/12/2020).

[186] Baris E. Suzek et al. "UniRef clusters: a comprehensive and scalable alternative for improving sequence similarity searches." In: *Bioinformatics* 31.6 (Nov. 2014), pp. 926–932. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btu739. eprint: https://academic.oup.com/bioinformatics/article-pdf/31/6/926/569379/btu739.pdf. URL: https://doi.org/10.1093/bioinformatics/btu739.

[187] Craig D. Livingstone and Geoffrey J. Barton. "Protein sequence alignments: a strategy for the hierarchical analysis of residue conservation." en. In: *Bioinformatics* 9.6 (Dec. 1993), pp. 745–756. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/9.6.745. URL: https://academic.oup.com/bioinformatics/article/9/6/745/256310 (visited on 01/27/2020).

[188] Mathura S. Venkatarajan and Werner Braun. "New quantitative descriptors of amino acids based on multidimensional scaling of a large number of physical–chemical properties." en. In: *Molecular modeling annual* 7.12 (Dec. 2001), pp. 445–453. ISSN: 0948-5023. DOI: 10.1007/s00894-001-0058-5. URL: https://doi.org/10.1007/s00894-001-0058-5 (visited on 01/15/2020).

[189] Konrad J. Karczewski et al. "Variation across 141,456 human exomes and genomes reveals the spectrum of loss-of-function intolerance across human protein-coding genes." In: *bioRxiv* (2019). Ed. by Carlos A Aguilar Salinas et al. DOI: 10.1101/531210. URL: https://www.biorxiv.org/content/early/2019/08/13/531210.

[190] Monique Nijhuis et al. "Increased fitness of drug resistant HIV-1 protease as a result of acquisition of compensatory mutations during suboptimal therapy." In: *AIDS (London, England)* 13 (Jan. 2000), pp. 2349–59. DOI: 10.1097/00002030-199912030-00006.