CHALMERS
UNIVERSITY OF TECHNOLOGY

UNIVERSITY OF GOTHENBURG

Machine Learning Prediction of Enzymes' Optimal Catalytic Temperatures

Master's Thesis in Computer Science and Engineering

CAMILLE FINLINSON PORTER

Department of Computer Science and Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
UNIVERSITY OF GOTHENBURG
Gothenburg, Sweden 2022

Master's thesis 2022

Machine Learning Prediction of Enzymes' Optimal Catalytic Temperatures

CAMILLE FINLINSON PORTER

UNIVERSITY OF
GOTHENBURG

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Computer Science and Engineering
Chalmers University of Technology
University of Gothenburg
Gothenburg, Sweden 2022

Machine Learning Prediction of Enzymes' Optimal Catalytic Temperatures

Supervisor: Graham Kemp, Department of Computer Science and Engineering
Supervisor: Martin Engqvist, Department of Biology and Biological Engineering
Examiner: Peter Damaschke, Department of Computer Science and Engineering

Master's Thesis 2022
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Typeset in LaTeX
Gothenburg, Sweden 2022

Machine Learning Prediction of Enzymes' Optimal Catalytic Temperatures
CAMILLE FINLINSON PORTER
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg

Abstract

Enzymes that have been genetically engineered to withstand high temperatures are used by industry to make products with less waste and pollution. Different features of protein structure affect the optimal catalytic temperature ("topt") at which enzymes catalyze reactions most efficiently. We sought to use information from protein structures to predict the topt. To do this, we analyzed the structures and optimal catalytic temperatures of 1379 proteins in 7 different ways. For a set of analyses based on Delaunay atomic interactions, the atoms for each protein were categorized by their Tsai atomic group, Popelier atomic group, or by their amino acid, and the nearest neighbors of each atom were then found by Delaunay triangulation. Next, the neighbors were classified by their atomic group and their frequencies calculated. For a separate analysis of atomic interactions ("threshold residue atomic interactions"), the atoms for each protein were categorized by the beta carbon of their amino acids. Any beta carbons within 8Å were found to be interacting. A third set of analyses based on the frequencies of each category of atom on the protein interior and surface was also performed. Each atom was again categorized by Tsai atomic group, Popelier atomic group, or amino acid residue. All of the frequencies in these seven groups were separately used as the predictor variables in regression to predict the response variable, the optimal catalytic temperature. Four different kinds of regression were tried: elastic net, sparse group lasso, decision tree, and support vector. The predictions had maximum testing R^2 values of 0.4. These results are similar to results in previous work done by Ulfenborg 2020. We found that being very detailed in defining interactions and categories did not give better results.

Keywords: enzyme, protein, amino acid, protein structure, optimal catalytic temperature

Acknowledgements

I wish to thank Dr. Martin Engqvist and Dr. Graham Kemp and for their valuable suggestions and support.

Camille Finlinson Porter, Gothenburg, January 2022

Contents

List of Figures

List of Tables

1

Introduction

The fundamental dogma of biology is that DNA is transcribed to RNA, which is then translated to make proteins. Proteins are the machines of the cell; almost every function that is done by or in the cell is performed by a protein. Proteins are made of a string of amino acids which fold into a complex structure. The shape that the protein folds into determines its function.

An enzyme is a specific type of protein that acts as a biological catalyst to speed up a chemical reaction. Enzymes can function at different temperatures but have an optimal temperature ("topt") at which they catalyze reactions most efficiently. At temperatures higher than the optimal temperature, the catalytic rate decreases because the protein begins to denature or unfold. At lower temperatures catalysis proceeds more slowly because diffusion occurs more gradually as temperature decreases.

Enzymes are used in industrial chemical reactions to make products with less waste and less toxic byproducts than traditional manufacturing methods. For industrial purposes, it is desirable to create enzymes that can withstand high temperatures. Because of the diffusion effect, chemical reactions proceed more rapidly when the reaction temperature is increased. This makes the reaction more efficient. Higher temperatures also help remove unwanted reaction byproducts (Vogt, Woell, and Argos 1997). The enzymes that are used in industry are often the result of genetic engineering because no enzyme from nature has been found to perform the needed task at the desired speed. In order to avoid denaturation, the thermal stability needs to be enhanced when the protein is designed.

The overall goal for this thesis is to find features from an enzyme that influence the optimal catalytic temperature. A protein can be divided into amino acids, which can be divided into atoms. It is the combination of the environment and interactions of each atom that effectuate how the protein folds and functions. We hypothesize that we should therefore be able to find meaningful information about protein's stability by examining them at the atomic level. We will examine the interactions of the protein's atoms with each other to try to predict the temperature at which the enzyme is optimally functioning. Learning more about these features could inform future protein engineering efforts.

In the background section (Chapter Two) we discuss protein structure and the different chemical bonds which are present in proteins. These bonds affect general protein stability. We also present different ways to classify the atoms into atomic groups. In the theory section (Chapter Three) we discuss Delaunay triangulation in order to find the nearest atomic neighbors. This is important because we hypothesized that the environment of each atom was crucial to understanding protein

stability. In Chapter Four we present the statistical methods that were used to analyze the dataset. The results and conclusion are covered in Chapters Five and Six.

2

Background

2.1 Protein Structure

The primary structure of a protein is the order of its amino acids. Each amino acid has the same backbone with a nitrogen bonded to a carbon, which is bonded to another carbon, which in turn is bonded to an oxygen (N-C_α-C-O). The first carbon is the C_α as shown in Figure 2.1. The sidechain, or R group, is attached to the C_α in the backbone and begins with another carbon (C_β). The atoms in the sidechain determine the amino acid identity. There are 20 standard amino acids used by most organisms, so there are 20 standard side chains, each with different atoms and properties. When multiple amino acids connect, the nitrogen connects to an oxygen of the previous amino acid. This forms a chain along the backbone of the amino acids (sometimes called the main chain).

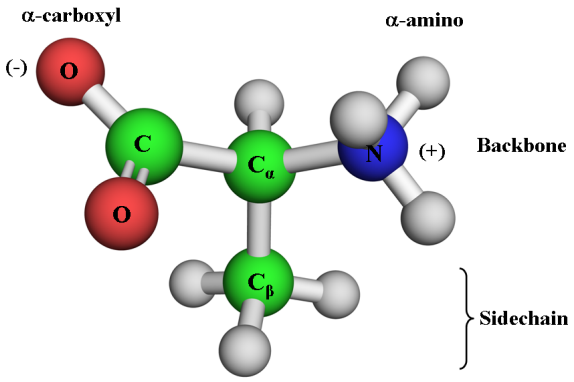

Figure 2.1: The structure of the amino acid alanine in 3D from Kessel, Amit 2021. Green represents carbon, blue represents nitrogen, red represents oxygen, and grey represents hydrogen. All amino acids have a central C_α which is bonded to four things: a hydrogen, a carboxyl group, an amino group, and a fourth group called side chain. The side chain can vary and its composition determines which amino acid it is.

The secondary structure of a protein is made by local folded structures (α-helices and β-sheets) that form due to hydrogen bonds between the main-chain peptide groups, as shown in Figure 2.2. An α-helix is formed by a hydrogen bond between the backbone N-H hydrogen and the backbone C=O of an amino acid four residues earlier in the primary sequence. α-helices are the most numerous local structure in proteins. β-sheets are also generally made by a hydrogen bond between

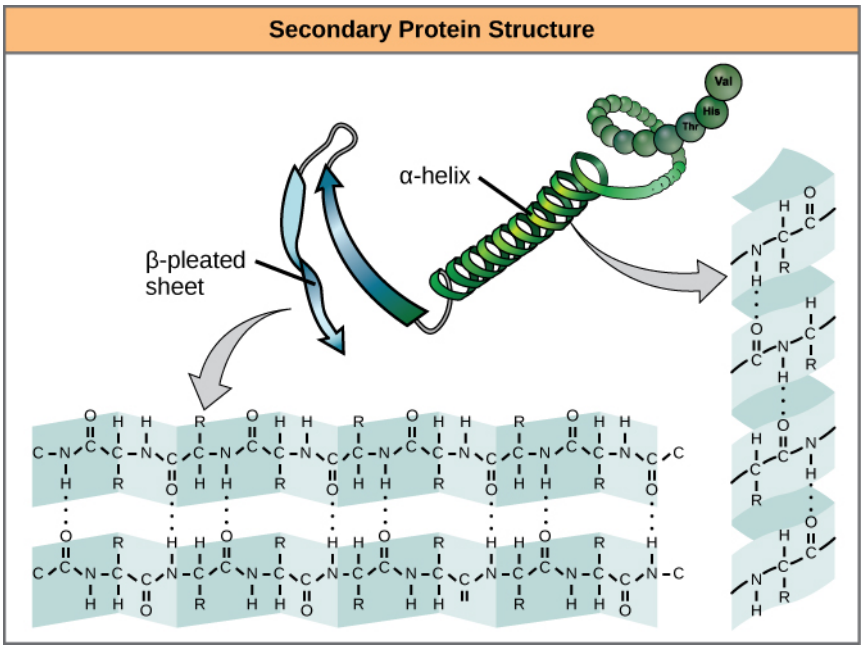

Figure 2.2: α-helices and β-sheets are formed by interactions of the side-chains of amino acids. The R in the amino acid structure represents the side chain. Image from OpenStax 2014.

the backbone N-H hydrogen and the backbone C=O. It is fairly simple to predict the secondary structure based on only knowledge of the primary structure.

Tertiary structure is formed because of the interactions (ionic, hydrogen, and Van der Waals bonds) between the side chains of the amino acids. They can form between amino acids that are quite far from each other in the primary structure and are hard to predict. Quaternary structure comes from multiple amino acid chains interacting together. It is only present when there are multiple amino acid chains, which is not always the case. It is also hard to predict.

An amino acid can be close in space to another amino acid that is far away from it in the primary structure. If amino acids are far apart from each other in the primary structure but close to each other in space after the protein folds, we assume that they are important to protein domain architecture or are important for substrate specificity. These amino acids are not covalently bonded together, so they are not forced to be physically near each other. Instead, they are close because that is the most energetically favorable configuration for the protein to take. The non-covalent bonds that are formed by their association increase the stability of the protein.

In this work, we will consider the environment of each atom in a protein. At the atomic level, the space around a protein is very full of water molecules and other substrates. Each atom is usually influenced only by things that are very close to it; potential interaction from a faraway molecule in the protein will generally be blocked by closer atoms in the surrounding water. There are some long-range electrostatic interactions in proteins, but these are not considered in this.

2.2 Bond Types

Protein stability is an important consideration because stable proteins can withstand higher temperatures without denaturing, therefore, stable enzymes have higher optimal catalytic temperatures. In general, more rigid protein structures increase protein stability (Vihinen 1987). This rigidity is caused by intramolecular interactions. These interactions include covalent bonds, ionic bonds, hydrogen bonds, Van der Waals bonds, and hydrophobic bonds (Vogt, Woell, and Argos 1997; Kopp and Schwede 2004). These bonds are described below.

The strongest chemical bonds in a protein structure are covalent bonds. Covalent bonds are formed as atoms fill their outer shell by sharing electrons with other atoms. Covalent bonds connect the atoms in an amino acid and connect the amino acids to each other. This type of bond is very hard to break. Cysteine an example of an amino acid with a side chain that can form a covalent bond. Cysteine can bond with another cysteine, creating a disulfide bridge (–CH2-S-S-CH2–). Disulfide bridges have been shown to increase protein stability (Boutz, Whitelegge, and Yeates 2007).

Ionic bonds, or salt bridges, are formed when one positively charged atom donates an electron to a negatively charged atom. The atoms then become attracted to each other because they have opposite charges. In a protein, these bonds are not as strong as covalent bonds because they take place in the presence of water, which can break them. Salt bridges have been found to increase stability in many proteins (Petsko 2001). The amino acids that are involved in salt bridges are the three alkaline amino acids (arginine, lysine, and histidine) and the two acidic amino acids (aspartic acid and glutamic acid). The ionic bonds have different strengths which depend on the difference of the amino acids' pK_as (Xie et al. 2015).

Hydrogen bonds come from the electrostatic attraction from a hydrogen that is covalently bonded to an electronegative atom (such as nitrogen or oxygen). This results in a molecule that has a partial positive charge on one side and a partial negative charge on the other. The partial positive charge of the molecule is attracted to a partial negative charge of other molecules. These bonds can form in many different places in a protein as well as with the surrounding water molecules. Each amino acid forms on average two hydrogen bonds when the protein is folded (Gong, Porter, and Rose 2011). A higher number of hydrogen bonds is associated with protein stability (Vogt, Woell, and Argos 1997). The exposed polar surface is able to form hydrogen bonds with water. An increase in polar fractional surface means more hydrogen bonding with water and more energetic stability (Vogt, Woell, and Argos 1997).

Van der Waals bonds are transient, weak bonds formed by attractions to opposite charges. They are made by electrons which are constantly moving and creating brief negative and positive charges. They can form between atoms that are packed tightly together, which happens in the protein interior. Having a more compact interior and therefore more Van der Waals bonds is associated with protein stability (DeDecker et al. 1996).

2.3 Protein Stability

Different methods have been used to study protein thermal stability. One is to compare the same protein from different organisms. Some of the organisms are mesophiles which grow best at temperatures between 20 to 50°C, some are thermophiles which prefer temperatures between 50 and 75°C, and some are psychrophiles which prefer temperatures between 0 and 20°C (Hickey and Singer 2004). Proteins are said to be more thermostable if they can withstand higher temperatures. By comparing similar proteins from organisms with different preferred temperatures, it is possible to see which parts of the protein are correlated with higher or lower temperatures.

Another method used to study protein thermal stability is determining the denaturation temperature, the temperature at which the protein begins to unfold and break its non-covalent bonds. Harrington and Schellman first showed that ribonuclease A becomes denatured at high temperatures and that this is reversible when the temperature is lowered (Harrington and Schellman 1956). The denatured proteins were chains of amino acids bonded together with only covalent bonds. In this state, the proteins resembled strings rather than globular structures. The proteins folded back into their original structures when the harsh conditions were reversed. Thermal denaturation is now commonly used to study a protein's stability. It occurs when all or most of the tertiary structures have been disrupted (Petsko 2001). Having more weak bonds that hold a protein together increases a protein's stability. When the disruption temperature is higher, the protein is more stable because it has more bonds holding it together.

The method that we will use to study protein stability is the optimal catalytic temperature (topt). Only enzymes have a topt as other proteins do not catalyze reactions. Up to a point, higher temperatures cause catalysis to occur more rapidly. Diffusion, or the movement of particles in fluid, occurs more quickly at higher temperatures. When the enzyme becomes too warm, the non-covalent bonds begin to break and the protein begins to denature or unfold. This causes the catalytic rate to decrease. The warmest temperature that an enzyme can endure while holding the required shape will be the temperature when catalysis occurs the most quickly. This can be measured by quantifying how quickly the substrate is converted at different temperatures.

There is a problem with the way the optimal catalytic temperatures are measured. The temperature are not tested with one degree changes, but with gaps of several degrees. Therefore, the optimal temperature histogram of our dataset is not smooth as it should be, but bumpy. There are temperatures that are common to test. Any enzyme with a topt close to that temperature will have that common temperature selected as its topt, though it may be a little inaccurate. The histogram of the data is shown in Figure 2.3. This is not a fatal flaw for analysis, but it will make the results slightly less accurate.

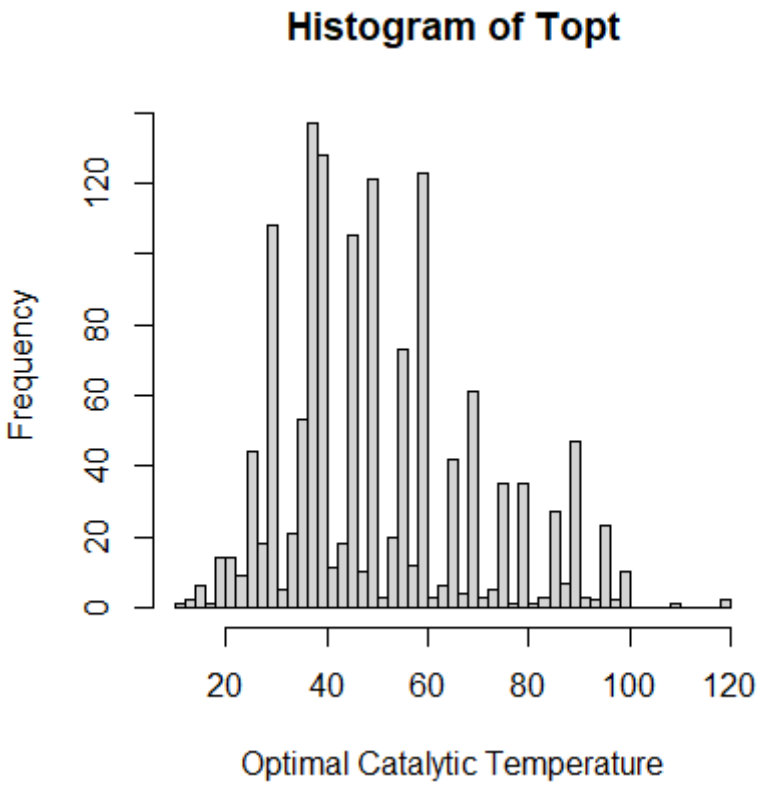

Figure 2.3: A histogram of the optimal catalytic temperatures in our dataset. When the optimal catalytic temperature is tested, it is common to only test a few temperatures, and not every possible temperature. This results in a histogram that is not smooth as it should be, instead it has gaps in the temperatures that are uncommon to test.

2.4 Context

The thermal stability of proteins is not due to a single cause, but instead to a combination of all of the elements mentioned above as well as other factors (Petsko 2001). Many studies have compared a few related proteins and drawn conclusions about what makes those proteins more or less stable. But now that so much work has already been done to sequence and find structures for many proteins, there is a large dataset of proteins available. This large dataset of proteins now allows us to study larger trends and uncover new factors affecting thermostability.

Previous PhD and Master's theses have focused on this problem and laid the groundwork (G. Li et al. 2019; Ulfenborg 2020). This work will expand upon those efforts. The first thesis from this group that predicted enzyme catalytic temperature only used the amino acid frequencies, dipeptide frequencies and other basic protein properties of different amino acids to predict the optimal temperature (G. Li et al. 2019). They used protein sequence data from 7565 archaea, bacteria, fungi and protozoa. The growth temperature of the organism was used as the optimal catalytic temperature. This growth temperature is less accurate than an experimentally measured optimal catalytic temperature because organisms can have different temperatures in different locations. It used because it allowed for a much larger number of sequences to be annotated. Most of these proteins did not have verified structures, and therefore structural data was not used. Although the dataset was imperfect, they were able to create a model with an R^2 value of 0.48. They were able to explain about half of the variation which was present in their dataset with their model which used only information from the primary structure of the proteins.

The next thesis project used a set of proteins that had more accurate optimal catalytic temperatures and known structures (Ulfenborg 2020). This thesis used enzymes that had experimentally determined optimal catalytic temperatures, which are much more accurate. Finding the optimal temperature for each enzyme is time consuming and expensive and therefore there are many fewer enzymes in this dataset compared to the first dataset. The position of each atom in each protein was known for this dataset. Some of the protein structures were experimentally verified with crystallography, which is exceedingly time consuming. Some of the protein structures come from SWISS-MODEL or ModBase, where a computer generates the structure based on homologous (or related) proteins (Kopp and Schwede 2004; Pieper et al. 2004). This can only be done when similar proteins have experimentally verified structures. Only 1903 enzymes had structural information and measured catalytic optima.

Information from the physical structure of each protein was extracted, including pairwise interaction between residues, contact order, radius of gyration, atomic groups on the surface, and residue torsion angles. Each category of features was analyzed separately as well as in different combinations of categories. When analyzed separately, the best category of features was a pairwise distance matrix and the second best was the frequency of surface atoms. The pairwise distance matrix measured how frequently different amino acid residues came into contact within the protein. The surface atoms categorized each atom into different groups and then found the frequency of each group on the protein surface.

The goal of this project is to learn more about the previous features that performed the best—the amino acid residue contacts and the surface atoms. Those features looked promising, and perhaps a more detailed study of them will lead to better results. The dataset used in the current project was very similar to the dataset used by Ulfenborg 2020. The main difference with the current project was the way the atoms were classified and how atomic interactions were counted.

There is a drawback to using the experimentally verified structures. Because it is so time consuming to experimentally verify protein structures, many proteins do not have verified structures. The computer generated structures are based on the proteins with verified structures. This creates a set of homologous (related) proteins that do not reflect the full diversity of possible protein configurations.

The problem of a non-diverse set of proteins led to poor results in another study. Yang et al. 2019a compared 100 pairs of proteins which were almost identical. Each pair consisted of a protein from nature (the wild-type protein) and a mutated version of that protein. The wild-type protein had a higher denaturation temperature than the mutant. They used only 9 different wild-type proteins, but 125 different mutant proteins were paired with them. They tried to predict the difference in the denaturation temperature for each pair. Their results indicated significantly overtrained models with poor performance on the testing data. They concluded that this was either due to their training data or the features they used to build models. Due to only using mutations of 9 base proteins, merely a small part of the possible protein configurations were sampled.

2.5 Atomic Group Classifications

In this project, we classified the atoms in three different ways: by their Tsai atomic group, by their Popelier atomic group, or by their amino acid residue. An explanation of each follows. The atoms were classified in hope that by classifying atoms into groups it would be possible to see which parts of amino acids are most important for protein stability. We hypothesized that we would get more accurate and more understandable results by breaking the proteins down by their atomic groups instead of by their amino acids. Thus, we would be able to see which parts of the amino acids were interacting.

2.5.1 Tsai Atomic Groups

A protein is made of amino acids, each of which is composed of a different combination of only five atoms: hydrogen, carbon, nitrogen, oxygen, and sulfur. The position of the hydrogen atoms is usually not known in the experimental structures. There are 13 ways to combine carbon, nitrogen, oxygen, and sulfate with different numbers of hydrogen. Tsai et al. 1999, used each possible combination as an atomic group and measured the size of each group. The group size includes the covalently bonded hydrogen atoms that are always present but have not been included in the positional data. The groups are therefore named by how many hydrogen atoms are bonded to each atom and by their valency in the outer shell. For example, C3H0 is a carbon with three non-hydrogen covalent bonds. C4H3 is a carbon with three

atomic group	chemical formula	radius (Å)
C3H0	-C<	1.61
C3H1	-CH-	1.76
C4H1	-CH<	1.88
C4H2	$-CH_2-$	1.88
C4H3	$-CH_3$	1.88
N3H0	>N-	1.64
N3H1	>NH	1.64
N3H2	$-NH_2$	1.64
N4H3	$-NH_3^+$	1.64
O1H0	=O or $-O^-$	1.42
O2H1	-OH	1.46
S2H0	-S-	1.77
S2H1	-SH	1.77

Table 2.1: Atomic groups with their sizes from Tsai et al. 1999. Because the sulfurs in cysteine form a covalent bond if and only if they are within 2.5 Å, these sulfurs were examined. When two sulfurs were found to be within the threshold distance, they were re-assigned from S2H1 to S2H0.

covalently bonded hydrogen atoms. Table 2.1 shows the size and composition of each Tsai atomic group.

These groups contain a wide variety of atoms. C3H0, for example, contains the alpha carbon from all amino acids, but it also contains the gamma carbon from asparagine, aspartic acid, histidine, phenylalanine, tryptophan, and tyrosine, the delta carbon from glutamine and glutamic acid, and part of the side chain in tryptophan and tyrosine. These atoms are not bonded to the same things and do not all have the same properties such as polarity or pK_a. These groups were made by Tsai to classify their atoms by their size, nothing more. We therefore decided to try classifying the atoms with an additional atomic group.

2.5.2 Popelier Atomic Groups

Popelier and Aicken 2003 created a different way of categorizing atoms in which each atom is classified according to the atoms it is covalently bonded to. Table 2.2 shows the 23 possible atomic groups. The actual classifications that were used for each atom in each amino acid are included in the Appendix in Table A.1.

Popelier categorized many kinds of atoms. Some are not contained in amino acids, and we therefore do not use all of their categories. There are 23 different Popelier atomic groups which are found in amino acids. We hypothesized that the Popelier groups would be more accurate than the Tsai groups. Each Popelier group should have more properties in common than the Tsai groups because the atoms are classified by the atoms they are bonded to and not by their sizes. There are many more groups than in the Tsai atomic groups because there are many different possible ways for each atom to bond.

The Popelier atomic groups may also be an imperfect system. A few of the

groups seem imperfect for our purposes, but we have used them as they were delineated. One such group is C_{10}, which contains the C_α of glycine, serine, and tyrosine. It makes sense that the C_α from glycine has its own group, because it has no side chain, but putting serine and tyrosine together with it is odd since they have side chains. We think it would have made more sense to categorize the C_α in serine and tyrosine as C_8 with the other C_α.

2.5.3 Amino Acid Residue

This categorization is simple. The atoms were classified only by which amino acid they were a part of. This is less precise than categorizing every atom by its type, so our hypothesis was that this categorization would be less useful and have lower scores than classifying the atoms by their atomic groups.

Atomic Group	Coordination	Description
C2	[H|H|H|C,S]	Methyl bonded to C or S
C3	[H|H|C|C,S]	Methylene bonded to C or S
C4	[H|C|C|C]	Tertiary C
C7	[H|C|C|N]	C_α in Lys
C8	[H|C|C|N]	C_α in 16 amino acids
C9	[H|C|C|O]	C_β in Thr (secondary alcohol)
C10	[H|C,H|C|N]	C_α in Gly, Ser, Tyr and methylene bonded to N
C12	[C|C|C,H]	Olefinic, in conjugated ring
C14	[C|C|C]	Olefinic, in conjugated ring
C15	[H|C|N]	Enaminic C (C_2 in indole) ($C_{4,5}$ in imidazole)
C17	[H|N|N]	C_2 in imidazole
C18	[C|C|N,O]	Bridge C in indole, C_α in phenol
C19	[C|N,O|O]	Amidic/carboxy bonded to C
C21	[N|N|N]	Guanidinic
O2	[C|H]	Phenol O (ArOH) or hydroxy oxygen in Glu and Asp derivatives
O3	[C|H]	Alcohol O bonded to alkyl group
O4	[C]	Keto O in the carboxy group
O5	[C]	Amide O
N1	[H|C,H|C,H]	tricoordinated and (nearly) tetrahedral
N2	[H|C,H|C,H] or [C|H]	Tricoordinated and largely planar or bicoordinated
S3	[C|H]	S in Cys
S6	[C|C]	S in Met
SS	[S]	Disulfide bridge

Table 2.2: Relevant atomic groups from Popelier and Aicken 2003. Vertical bars separate the atoms bonded to carbon atoms and alternatives are separated by a comma. The coordination contains the atoms that the central atom is covalently bonded to. An extra group was created for disulfide bonds. When two sulfur atoms were found to be within 2.5 Å of each other, they were reclassified as SS. This group of sulfurs is covalently bonded together, forming a disulfide bridge.

3

Theory

A protein is made of many atoms bonded together. We hypothesized that by studying the environment of each atom we would be able draw conclusions about the protein as a whole. In order to study the atoms' environment, we measured how each atom interacts with every other atom within the protein by finding the closest atomic neighbors. Interactions with distant neighbors are blocked by atoms within the protein and by the fluid in which the protein exists, therefore, atoms can only interact with their close neighbors. We want to find the nearest atomic neighbor for each atom in every direction. This will give a picture of the environment for each atom.

Finding the neighbors in three-dimensions is difficult; this is not a simple nearest neighbor problem. Most nearest neighbor algorithms such as K-Nearest Neighbors, find one closest neighbor or a fixed number of close neighbors. Other algorithms find all neighbors within a threshold distance. These algorithms are not optimal for our purposes because there could be atoms in between supposed neighbors, blocking their interaction. A Delaunay triangulation can provide the nearest neighbors in every direction, and is not based on finding a certain number of neighbors. An explanation of Delaunay triangulation follows.

3.1 Delaunay Triangulation

A geometrical triangulation is a subdivision of a planar object or a higher-dimensional geometrical object into simplices. A given set P of discrete points in the geometrical object are used as the vertices of the simplices. Points in two dimensions are subdivided into two dimensional simplices, or triangles. Points in three dimensions are subdivided into three dimensional simplices, or tetrahedrons. A Delaunay triangulation is a special kind of triangulation with specific properties. In Delaunay triangulation, the triangles are chosen such that no point is inside the circumcircle of any triangle. Delaunay triangulations maximize the minimum angle of all of the triangles in the triangulation (Lawson 1977). This means that the triangle that is the nearest to an equilateral triangle will be preferred (Figure 3.1). Because of this, Delaunay triangles usually avoid sliver triangles. A sliver triangle has a long, thin shape, and would result in connection to a further-away point. As a result of Delaunay triangulation, every point is connected to every close point. This provides an elegant way to find the nearest neighbors.

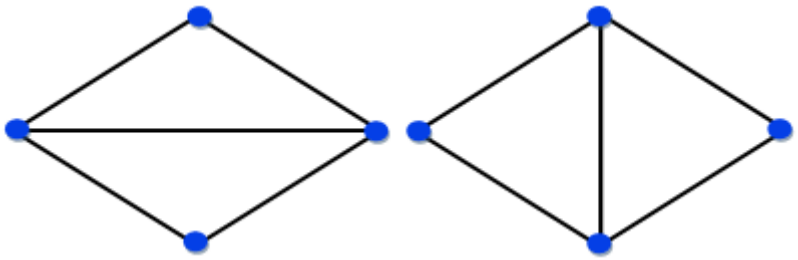

Figure 3.1: If there are multiple possible triangles for a set of points, the triangulation that maximizes the minimum angle will be preferred. This will create triangles more similar to equilateral triangles. In this graphic, the triangulation on the right is preferred as it maximizes the smallest angle.

3.1.1 Delaunay Algorithms

There are many different possible ways to calculate a Delaunay triangulation. First we will discuss a simple method and then a more complex method.

The flip algorithm is an easy-to-understand way of computing a Delaunay triangulation. We begin with the set of points to be triangulated. It begins with the property that a triangulation is Delaunay if and only if every edge in the triangulation is locally Delaunay. A locally Delaunay edge is not always a Delaunay edge, but if every edge in a triangulation is locally Delaunay, then all of the edges are Delaunay edges. The flip algorithm constructs any triangulation, then chooses any edge that is not locally Delaunay and flips it (Lawson 1977). This is repeated until every triangle is Delaunay. Figure 3.1 shows a non-Delaunay triangulation being flipped to become a Delaunay triangulation. It is possible that flipping one edge will change what was formerly a Delaunay triangle into a non-Delaunay triangle. Therefore, this can be a slow method. This algorithm is easy to understand, but it takes $O(n^2)$ time (Hurtado, Noy, and Urrutia 1999) in the worst case. It will be unreasonably slow for large numbers of points.

There is another property of Delaunay triangulation that has not been previously discussed. Given a set of P points in m dimensions, it is possible to project the points onto a paraboloid of m+1 dimensions. This can be done by giving each point p an extra coordinate p^2, which will form a paraboloid shape. This process is known as lifting. The paraboloid shape is a lower convex hull. When P is projected onto a lower convex hull, the result is a curve made of flat Delaunay triangles. This is shown in Figure 3.2. This property was discovered by Brown 1979. The paraboloid method created an efficient algorithm for computing Delaunay Triangulation in $O(n \log(n))$ time.

The algorithm we used is known as Quickhull or QHULL (Barber, Dobkin, and Huhdanpaa 1996), which is a popular program. It finds the convex hull as well as the Delaunay Triangulation in $O(n \log(n))$ time while using less memory than Brown's version of the convex hull. It can also compute the convex hull in two, three, or four dimensions.

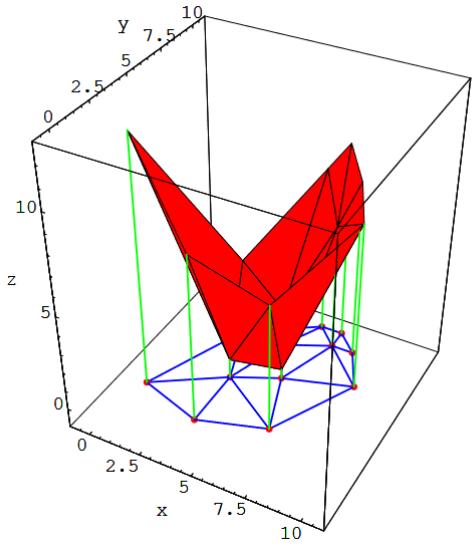

Figure 3.2: When points are projected onto a lower parabola in one higher dimension, the result is a convex hull made of flat Delaunay triangles. This property also holds for higher dimensions, thought it is difficult to visualize. Image from Gallier and Quaintance 2017.

3.2 Distance from Delaunay Neighbor

In order to find the closest atomic neighbors, the atomic locations in three-dimensional space were used as points for our Delaunay triangulations. This means the atoms were simplified and modeled as having the same size. In reality, different atoms (carbon, nitrogen, oxygen, etc) have slightly different sizes.

We define the neighborhood of any point as the set of points that the point is connected to. We began our triangulation by numbering each atom uniquely. The result of a Delaunay triangulation is a series of triangles, or in our case, simplices. Each simplex is made of four numbered atoms. We defined neighbors as any atom that was contained in the same simplex. There are usually multiple simplices that contain any one atom. The neighbors for each atom were saved as a set of all of the atoms which were found in the same simplex.

We will show how this was used with a toy example of 6 atoms. The atoms' position are shown in Figure 3.3. Delaunay neighbors are given by their simplices. The simplices for this toy example are: [1 5 0 4], [1 3 5 4], [1 3 2 4], and [1 3 2 0]. Anything in the same simplex is a Delaunay neighbor. The neighbors for the toy example protein are shown in Table 3.1.

When atoms are Delaunay neighbors, there is no atom in the dataset directly in between them blocking them from interacting. However, there could be atoms that are not a part of the dataset that would be present in reality that could block interactions. Proteins are found in a fluid made mainly of water. It is possible for this fluid to block the interaction of atoms. This would result in atoms which were Delaunay neighbors but were in actuality blocked from interacting with each other.

In order to solve the problem of non-neighbor interactions, the next step was

Atom Number	Neighbors
0	1, 2, 3, 4, 5
1	0, 2, 3, 4, 5
2	0, 1, 3, 4
3	0, 1, 2, 3, 4, 5
4	0, 1, 2, 3, 5
5	0, 1, 3, 4

Table 3.1: Delaunay neighbors for the toy example protein.

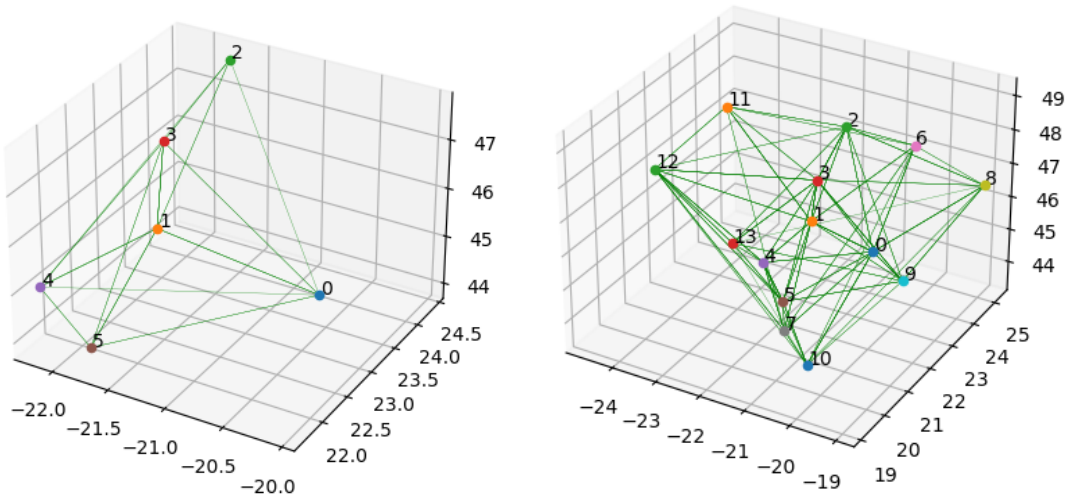

Figure 3.3: Right: The Delaunay triangulation for a toy example consisting of 6 atoms. Left: The Delaunay triangulation for the same 6 atoms with the Triominoes closest water positions added (represented by the points above 5).

to find a method to winnow the Delaunay neighbors that are not representative of atomic interactions. One possibility would be to use a threshold to remove atoms that are Delaunay neighbors but are actually far apart. In this case, any Delaunay neighbor greater than the threshold distance would be removed. We decided against this method in favor of a more precise approach. Instead, a program called Triominoes (Kemp 2019) was used to winnow the neighbors. Triominoes takes in the position of each atom in a protein and calculates its size. Since the positions of the hydrogen atoms are unknown, it uses the sizes found by Tsai et al. 1999 to find the amount of space the atoms and their associated hydrogen atoms are taking up. It then rolls a sphere the size of a water molecule (which has a radius of 1.5 Å) over the surface of the protein, finding each place where the sphere can touch three atoms at once. Each of those locations is the deepest place on the protein surface that a water molecule can be found near the protein. The output of the program is all of the locations where the sphere can touch three atoms at once. The result is a net of points that hover just over the entire protein surface. We used this net of points to winnow the neighbors from Delaunay triangulation and find which atoms were actually interacting.

We used Delaunay triangulation combined with this net of points to in order to winnow the neighbors. The Delaunay triangulation was calculated for the dataset which consisted of the position of the atoms combined with the nearest sphere positions. These positions are the closest positions that water molecules could fit into the protein. Then any simplex that included a Delaunay probe position was removed. This removes atoms that are Delaunay neighbors, but are not actually interacting because a water molecule is blocking their interactions. The close sphere positions follow the surface of the protein, so removing any simplex with a close sphere position lets us break up large gaps between Delaunay neighbors without setting an arbitrary threshold. It allows us to break up any far away contacts that are on the exterior of the protein. Removing the simplices with water molecules allows us to break up exterior connections and keep interior connections.

In the toy example none of the atoms are very far apart—the maximum distance is 3.78 Å between atoms 2 and 4. After including the Triominoes water positions, the Delaunay neighbors are the same except atoms 2 and 4 are no longer neighbors. Atoms 2 and 4 were the furthest apart, and their interaction is blocked by atoms 1 and 3. We can also see there is a small convexity in the [1,2,3,4] simplex that is causing the relationship between 2 and 4 to become broken after interacting with water molecules. This is just a toy example, but we can see from Figure 3.4 that this method does successfully remove the relationships between neighbors that are far apart in a real protein.

Another method was also used to winnow the counts of atomic neighbors. The most common interactions were removed from the counts of atomic group interactions so that their numbers don't overwhelm the other interactions and drown out the signal. First, all atom contacts from the same amino acid were excluded. Second, contacts that are between atoms belonging to the protein backbone (nitrogen, carbon_α, carbon, and oxygen) that are one amino acid before or after the current amino acid in the primary sequence were also excluded. This removed the covalent bonds from the dataset.

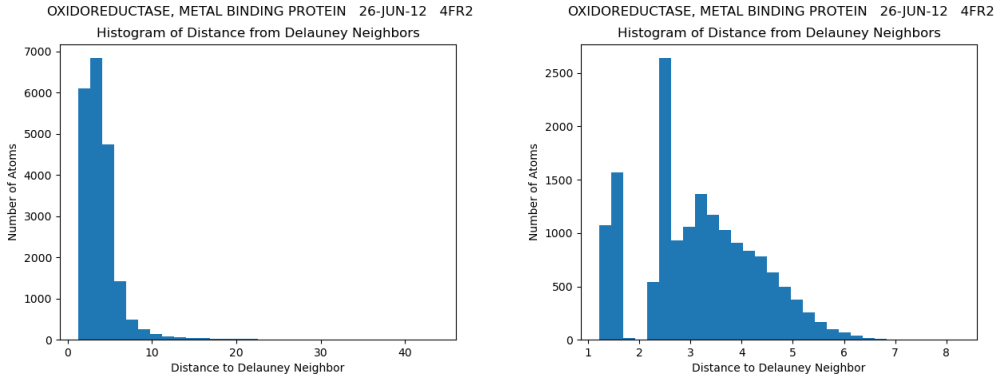

Figure 3.4: Histograms of distances from Delaunay neighbor for the oxidoreductase protein group III alcohol dehydrogenase (Elleuche et al. 2013). The left graph shows all the distances from any Delaunay neighbor. The larger distances do not represent atomic interactions and are not useful. The right graph shows distances when they have been filtered by removing simplices with water positions. This graph shows the distances before covalent bonds have been removed, which is why there are contacts between 1-2A.

Atom Number	Atom Name	Amino Acid	Tsai Atomic Group	Popelier Atomic Group
0	N	HIS	N3H1	N2
1	CA	HIS	C4H1	C8
2	CG	HIS	C3H0	C15
3	CB	HIS	C4H2	C3
4	C	HIS	C3H0	C19
5	O	HIS	O1H0	O5

Table 3.2: The atoms found in the toy example protein come from one amino acid in a real protein. Their atomic groups have been included. When the atoms are found to interact, then the atomic groups they belong to are found to interact.

After finding and winnowing the Delaunay neighbors, the remaining neighbors were counted to find which atomic groups were interacting. We will show how this occurs with the toy example protein. The atoms and positions in the toy example protein were taken from a real protein and can be categorized into atomic groups. Their chemical composition and atomic group numbers are shown in Table 3.2. When atoms 0 and 1 are found to be Delaunay neighbors, this means that N3H1 and C4H1 interact once for the Tsai atomic groups and N2 and C8 interact once for the Popelier atomic groups. When atoms 0 and 2 are found to be Delaunay neighbors, this means that N3H1 and C3H0 interact once for the Tsai atomic groups and N2 and C15 interact once for the Popelier atomic groups. All of the neighbors' interactions are added together to produce Figure 3.5. There are 13 unique relationships in the toy example: 0-1, 0-2, 0-3, 0-4, 0-5, 1-2, 1-3, 1-4, 1-5, 2-3, 2-5, 3-4, and 4-5. If the number of atomic interactions from the upper right triangle of the matrix of Figure 3.5 are added, they sum to 13. The atoms were also classified based on which amino acid residue they belong to.

Since all the toy atoms come from the same histidine, once the atomic interactions from within an amino acid are removed, there are no interactions left and the atomic group interaction matrix is all zeros for every different grouping of atoms that we have. In a normally sized protein, there would be interactions remaining after same amino acid contacts are removed.

3.3 Atom Frequency Classifications

The frequencies of the atoms were examined in several different ways. First, the atoms' interactions were classified four different ways to create four different data sets. They are classified by breaking them into Tsai and Popelier atomic groups as well as by their residues (the residue interactions were calculated two different ways). Second, the surface / inner frequencies were examined for the Tsai and Popelier atomic groups and for the residues. The result is seven unique data sets that classify atomic frequencies based on their composition and location. Each type of interaction is described below.

3.3.1 Atomic Group Interaction

The first way that atoms were counted was by their interactions with nearby atoms. The neighbors from Delaunay triangulation were used to find which atomic groups are interacting. Three different ways of categorizing the atoms have been used with these Delaunay neighbors: by the Tsai atomic groups, by the Popelier atomic groups, and by the amino acid residues.

There are 13 different Tsai atomic groups and if they are combined in every possible way, there are $13 + 12 + 11 + ... + 2 + 1 = 91$ different Tsai atomic group interactions to count. There are 23 Popelier atomic groups with $23+22+...+2+1 = 276$ different possible Popelier atomic group interactions. There are 20 residues which can create $20 + 19 + ... + 2 + 1 = 210$ different possible residue interactions. One matrix for each classification (sized 13x13, 23x23, and 20x20) was created to count the atomic interactions.

	C3H0	C3H1	C4H1	C4H2	C4H3	N3H0	N3H1	N3H2	N4H3	O1H0	O2H1	S2H0	S2H1
C3H0	0	0	2	2	0	0	2	0	0	1	0	0	0
C3H1	0	0	0	0	0	0	0	0	0	0	0	0	0
C4H1	2	0	0	1	0	0	1	0	0	1	0	0	0
C4H2	2	0	1	0	0	0	1	0	0	1	0	0	0
C4H3	0	0	0	0	0	0	0	0	0	0	0	0	0
N3H0	0	0	0	0	0	0	0	0	0	0	0	0	0
N3H1	2	0	1	1	0	0	0	0	0	1	0	0	0
N3H2	0	0	0	0	0	0	0	0	0	0	0	0	0
N4H3	0	0	0	0	0	0	0	0	0	0	0	0	0
O1H0	1	0	1	1	0	0	1	0	0	0	0	0	0
O2H1	0	0	0	0	0	0	0	0	0	0	0	0	0
S2H0	0	0	0	0	0	0	0	0	0	0	0	0	0
S2H1	0	0	0	0	0	0	0	0	0	0	0	0	0

	C2	C3	C4	C7	C8	C9	C10	C12	C14	C15	C17	C18	C19	C21	N1	N2	O1	O2	O3	O4	O5	S3	S6	SS
C2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C3	0	0	0	0	1	0	0	0	0	1	0	0	1	0	0	1	0	0	0	0	1	0	0	0
C4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C8	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	1	0	0	0	0	1	0	0	0
C9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C15	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
C17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C19	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0
C21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
N1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
N2	0	1	0	0	1	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	1	0	0	0
O1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
O2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
O3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
O4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
O5	0	1	0	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0
S3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
S6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SS	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	ALA	ARG	ASN	ASP	CYS	GLN	GLU	GLY	HIS	ILE	LEU	LYS	MET	PHE	PRO	SER	THR	TRP	TYR	VAL
ALA	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ARG	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ASN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ASP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
CYS	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
GLN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
GLU	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
GLY	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HIS	0	0	0	0	0	0	0	0	13	0	0	0	0	0	0	0	0	0	0	0
ILE	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
LEU	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
LYS	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
MET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
PHE	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
PRO	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SER	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
THR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
TRP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
TYR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
VAL	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 3.5: Upper: the Tsai atomic group interaction counts for the toy example protein. Middle: the Popelier atomic group interaction counts for the toy example protein. Lower: the residue interaction counts for the toy example protein. All atoms come from the same amino acid, so they are all in one residue group, His-His. There are 13 unique interactions and therefore the upper right triangle of all of the matrices sums to 13. All of these atoms come from the same amino acid, so when same-amino acid interactions are removed, the counts become zero for each group.

After two atoms are found to be interacting, each atom is classified by which atomic groups are interacting. Then the interaction is added to the total number of interactions. For example, if an C_α from an Alanine is found to interact with a N from an Alanine, the carbon is classified as C4H1 for the Tsai group and C8 for the Popelier group. The nitrogen is classified as N3H1 for the Tsai group and N2 for the Popelier group. Then one is added to each of the counts of C4H1-N3H1, C8-N2, and Ala-Ala. The counts for one protein are found by looking at all close atomic Delaunay neighbors, classifying each into their respective atomic groups, and then counting each interaction one at a time.

In the end, there are three different matrices which have separately counted their respective atomic group interactions. The Tsai atomic group interaction table has 91 different atomic group interaction counts, the Popelier table has 267 atomic group interaction counts, and the residue table has 210 residue interaction counts. When they are displayed in a matrix where each row and column are the 13 different atomic groups, it is symmetrical around the main diagonal–the upper left and lower right values are identical because they represent the same group interactions. An example of all three Delaunay counts for one protein is shown in Figure 3.6.

The atomic group interaction counts are dependent on the size of the protein. A larger protein will have a higher count of every type of interaction. Therefore, to normalize the counts, they were converted to frequencies. The frequencies were created by dividing each atomic group interaction count by the total number of atoms in the protein. The frequency was preferred in order to have a more universal measure of the interactions. The number of atoms was included as a separate variable in each dataset.

The frequencies were calculated by finding the total of the upper right matrix (including the main diagonal), which is the total number of atomic interactions. All three categories for the atoms (Tsai, Popelier, and residue) had the same total count of atoms. Then every cell was divided by that total. When all the frequencies of either the upper right or lower left matrices are added, they sum to one. The atomic interaction frequencies for an example protein are shown in Figure 3.7.

3.3.2 Atomic Threshold Residue Interaction

This method was first implemented in a previous master's thesis by Ulfenborg 2020. It was the best category of features from that thesis. The atomic interactions were created to try to improve upon this feature set. This method was implemented in the current study in order to compare it to the Tsai, Popelier, and Delaunay residue atomic interactions.

The threshold residue-residue interactions determine how often any pair of amino acid residues interact. Instead of considering the position of every atom in an amino acid, only the C_β were considered. Glycine does not have a C_β, so in its place its C_α was used. Delaunay triangulation was not used to find which atoms were interacting. Instead, the distance between all C_β was calculated and the amino acids were considered to interact if the C_β were less than 8Å apart. Adjacent residues were not included in the counts. Because only one kind of atom was used, the only classification of atoms that was considered was the residues. There

	C3H0	C3H1	C4H1	C4H2	C4H3	N3H0	N3H1	N3H2	N4H3	O1H0	O2H1	S2H0	S2H1
C3H0	834	1362	2016	2183	950	146	2904	407	2	3003	273	14	4
C3H1	1362	1066	572	1202	896	16	933	142	0	1136	194	16	4
C4H1	2016	572	389	1546	1227	135	2114	174	4	2976	320	28	4
C4H2	2183	1202	1546	1016	1097	235	1976	461	23	3108	365	80	4
C4H3	950	896	1227	1097	606	25	865	125	5	1348	227	55	14
N3H0	146	16	135	235	25	0	28	10	0	179	14	2	0
N3H1	2904	933	2114	1976	865	28	549	284	1	3430	332	21	4
N3H2	407	142	174	461	125	10	284	65	4	477	59	2	0
N4H3	2	0	4	23	5	0	1	4	0	16	1	0	0
O1H0	3003	1136	2976	3108	1348	179	3430	477	16	1292	417	37	6
O2H1	273	194	320	365	227	14	332	59	1	417	18	4	2
S2H0	14	16	28	80	55	2	21	2	0	37	4	2	0
S2H1	4	4	4	4	14	0	4	0	0	6	2	0	0

	C2	C3	C4	C7	C8	C9	C10	C12	C14	C15	C17	C18	C19	C21	N1	N2	O2	O3	O4	O5	S3	S6	SS
C2	564	966	425	3	686	109	68	811	111	55	14	54	727	71	130	874	53	165	172	1149	14	31	10
C3	966	816	183	32	1104	42	316	925	218	120	25	58	1597	98	405	1921	81	240	709	2040	4	55	23
C4	425	183	4	0	186	7	8	118	9	11	0	5	177	7	21	223	4	17	29	269	2	5	0
C7	3	32	0	0	4	0	2	2	0	2	0	0	22	2	2	33	0	0	5	41	0	0	0
C8	686	1104	186	4	102	87	53	446	120	60	1	23	1506	14	152	1850	41	150	260	2210	2	13	6
C9	109	42	7	0	87	0	1	25	2	4	1	2	89	0	14	117	6	76	14	104	0	0	0
C10	68	316	8	2	53	1	4	97	54	6	3	1	186	4	68	316	4	66	46	357	0	2	4
C12	811	925	118	2	446	25	97	1008	355	40	30	251	551	50	119	750	90	83	136	875	4	14	0
C14	111	218	9	0	120	2	54	355	21	40	8	23	106	6	20	195	0	10	8	155	0	2	0
C15	55	120	11	2	60	4	6	40	40	30	2	14	75	8	18	211	9	8	20	105	0	1	0
C17	14	25	0	0	1	1	3	30	8	2	0	5	6	5	9	63	3	4	10	16	0	1	0
C18	54	58	5	0	23	2	1	251	23	14	5	0	13	4	8	51	38	3	2	26	0	0	0
C19	727	1597	177	22	1506	89	186	551	106	75	6	13	638	14	285	2668	26	189	488	2271	4	7	5
C21	71	98	7	2	14	0	4	50	6	8	5	4	14	4	92	77	0	13	11	43	0	14	0
N1	130	405	21	2	152	14	68	119	20	18	9	8	285	92	69	295	12	48	210	283	0	2	0
N2	874	1921	223	33	1850	117	316	750	195	211	63	51	2668	77	295	577	44	302	422	3187	4	12	11
O2	53	81	4	0	41	6	4	90	0	9	3	38	26	0	12	44	1	4	15	51	0	0	0
O3	165	240	17	0	150	76	66	83	10	8	4	3	189	13	48	302	4	13	81	270	2	4	0
O4	172	709	29	5	260	14	46	136	8	20	10	2	488	11	210	422	15	81	117	336	0	9	4
O5	1149	2040	269	41	2210	104	357	875	155	105	16	26	2271	43	283	3187	51	270	336	839	6	13	11
S3	14	4	2	0	2	0	0	4	0	0	0	0	4	0	0	4	0	2	0	6	0	0	0
S6	31	55	5	0	13	0	2	14	2	1	1	0	7	14	2	12	0	4	9	13	0	0	0
SS	10	23	0	0	6	0	4	0	0	0	0	0	5	0	0	11	0	0	4	11	0	0	2

	ALA	ARG	ASN	ASP	CYS	GLN	GLU	GLY	HIS	ILE	LEU	LYS	MET	PHE	PRO	SER	THR	TRP	TYR	VAL
ALA	1022	266	296	316	32	398	104	229	135	193	170	57	36	290	148	199	250	154	218	255
ARG	266	1322	212	352	16	102	189	163	153	221	194	25	20	346	206	171	205	212	130	211
ASN	296	212	1477	254	48	278	86	306	95	186	119	64	61	280	324	96	266	51	231	225
ASP	316	352	254	1515	22	230	57	211	153	141	189	89	2	390	327	266	333	99	212	182
CYS	32	16	48	22	86	10	0	13	0	24	61	8	0	25	0	0	35	18	17	36
GLN	398	102	278	230	10	1221	53	182	75	140	200	35	52	192	217	136	273	2	264	204
GLU	104	189	86	57	0	53	489	45	27	82	86	29	16	168	70	63	158	71	52	105
GLY	229	163	306	211	13	182	45	635	50	136	257	28	80	373	203	127	166	123	429	187
HIS	135	153	95	153	0	75	27	50	687	122	32	74	19	81	55	129	119	117	225	146
ILE	193	221	186	141	24	140	82	136	122	1098	272	39	31	464	190	122	250	134	238	169
LEU	170	194	119	189	61	200	86	257	32	272	1167	62	111	352	252	222	359	144	213	221
LYS	57	25	64	89	8	35	29	28	74	39	62	264	0	38	19	46	50	45	64	10
MET	36	20	61	2	0	52	16	80	19	31	111	0	246	64	105	32	87	30	46	125
PHE	290	346	280	390	25	192	168	373	81	464	352	38	64	1933	273	223	400	283	416	439
PRO	148	206	324	327	0	217	70	203	55	190	252	19	105	273	1156	135	212	55	163	167
SER	199	171	96	266	0	136	63	127	129	122	222	46	32	223	135	705	116	104	165	111
THR	250	205	266	333	35	273	158	166	119	250	359	50	87	400	212	116	1309	175	240	247
TRP	154	212	51	99	18	2	71	123	117	134	144	45	30	283	55	104	175	665	144	188
TYR	218	130	231	212	17	264	52	429	225	238	213	64	46	416	163	165	240	144	1390	381
VAL	255	211	225	182	36	204	105	187	146	169	221	10	125	439	167	111	247	188	381	1184

Figure 3.6: Delaunay atomic interaction counts for a transferase, the levansucrase from *Gluconacetobacter Diazotrophicus* (uniprot id: Q43998). The top is the Tsai interaction counts, the middle is the Popelier interaction counts, and the bottom is the residue interaction counts. The matrices are symmetric along the main diagonal. The 2 in the S2H0-S2H0 and SS-SS groups show that there are two separate disulfide bonds in this protein. Carbon is the most common element in proteins, while sulfur is rare. This is reflected in the atomic counts.

	C3H0	C3H1	C4H1	C4H2	C4H3	N3H0	N3H1	N3H2	N4H3	O1H0	O2H1	S2H0	S2H1
C3H0	0.01735	0.02833	0.04193	0.04540	0.01976	0.00304	0.06040	0.00846	0.00004	0.06245	0.00568	0.00029	0.00008
C3H1	0.02833	0.02217	0.01190	0.02500	0.01863	0.00033	0.01940	0.00295	0.00000	0.02363	0.00403	0.00033	0.00008
C4H1	0.04193	0.01190	0.00809	0.03215	0.02552	0.00281	0.04397	0.00362	0.00008	0.06189	0.00666	0.00058	0.00008
C4H2	0.04540	0.02500	0.03215	0.02113	0.02281	0.00489	0.04110	0.00959	0.00048	0.06464	0.00759	0.00166	0.00008
C4H3	0.01976	0.01863	0.02552	0.02281	0.01260	0.00052	0.01799	0.00260	0.00010	0.02803	0.00472	0.00114	0.00029
N3H0	0.00304	0.00033	0.00281	0.00489	0.00052	0.00000	0.00058	0.00021	0.00000	0.00372	0.00029	0.00004	0.00000
N3H1	0.06040	0.01940	0.04397	0.04110	0.01799	0.00058	0.01142	0.00591	0.00002	0.07133	0.00690	0.00044	0.00008
N3H2	0.00846	0.00295	0.00362	0.00959	0.00260	0.00021	0.00591	0.00135	0.00008	0.00992	0.00123	0.00004	0.00000
N4H3	0.00004	0.00000	0.00008	0.00048	0.00010	0.00000	0.00002	0.00008	0.00000	0.00033	0.00002	0.00000	0.00000
O1H0	0.06245	0.02363	0.06189	0.06464	0.02803	0.00372	0.07133	0.00992	0.00033	0.02687	0.00867	0.00077	0.00012
O2H1	0.00568	0.00403	0.00666	0.00759	0.00472	0.00029	0.00690	0.00123	0.00002	0.00867	0.00037	0.00008	0.00004
S2H0	0.00029	0.00033	0.00058	0.00166	0.00114	0.00004	0.00044	0.00004	0.00000	0.00077	0.00008	0.00004	0.00000
S2H1	0.00008	0.00008	0.00008	0.00008	0.00029	0.00000	0.00008	0.00000	0.00000	0.00012	0.00004	0.00000	0.00000

	C2	C3	C4	C7	C8	C9	C10	C12	C14	C15	C17	C18	C19	C21	N1	N2	O2	O3	O4	O5	S3	S6	S5
C2	0.0117	0.0201	0.0088	0.0001	0.0143	0.0023	0.0014	0.0169	0.0023	0.0011	0.0003	0.0011	0.0151	0.0015	0.0027	0.0182	0.0011	0.0034	0.0036	0.0239	0.0003	0.0006	0.0002
C3	0.0201	0.0170	0.0038	0.0007	0.0230	0.0009	0.0066	0.0192	0.0045	0.0025	0.0005	0.0012	0.0332	0.0020	0.0084	0.0400	0.0017	0.0050	0.0147	0.0424	0.0001	0.0011	0.0005
C4	0.0088	0.0038	0.0001	0.0000	0.0039	0.0001	0.0002	0.0025	0.0002	0.0002	0.0000	0.0001	0.0037	0.0001	0.0004	0.0046	0.0001	0.0004	0.0006	0.0056	0.0000	0.0001	0.0000
C7	0.0001	0.0007	0.0000	0.0000	0.0001	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0005	0.0000	0.0000	0.0007	0.0000	0.0000	0.0001	0.0009	0.0000	0.0000	0.0000
C8	0.0143	0.0230	0.0039	0.0001	0.0021	0.0018	0.0011	0.0093	0.0025	0.0012	0.0000	0.0005	0.0313	0.0003	0.0032	0.0385	0.0009	0.0031	0.0054	0.0460	0.0000	0.0003	0.0001
C9	0.0023	0.0009	0.0001	0.0000	0.0018	0.0000	0.0000	0.0005	0.0000	0.0001	0.0000	0.0000	0.0019	0.0000	0.0003	0.0024	0.0001	0.0016	0.0003	0.0022	0.0000	0.0000	0.0000
C10	0.0014	0.0066	0.0002	0.0000	0.0011	0.0000	0.0001	0.0020	0.0011	0.0001	0.0001	0.0000	0.0039	0.0001	0.0014	0.0066	0.0001	0.0014	0.0010	0.0074	0.0000	0.0000	0.0001
C12	0.0169	0.0192	0.0025	0.0000	0.0093	0.0005	0.0020	0.0210	0.0074	0.0008	0.0006	0.0052	0.0115	0.0010	0.0025	0.0156	0.0019	0.0017	0.0028	0.0182	0.0001	0.0003	0.0000
C14	0.0023	0.0045	0.0002	0.0000	0.0025	0.0000	0.0011	0.0074	0.0004	0.0008	0.0002	0.0005	0.0022	0.0001	0.0004	0.0041	0.0000	0.0002	0.0002	0.0032	0.0000	0.0000	0.0000
C15	0.0011	0.0025	0.0002	0.0000	0.0012	0.0001	0.0001	0.0008	0.0008	0.0006	0.0000	0.0003	0.0016	0.0002	0.0004	0.0044	0.0002	0.0002	0.0004	0.0022	0.0000	0.0000	0.0000
C17	0.0003	0.0005	0.0000	0.0000	0.0000	0.0000	0.0001	0.0006	0.0002	0.0000	0.0000	0.0001	0.0001	0.0001	0.0002	0.0013	0.0001	0.0001	0.0002	0.0003	0.0000	0.0000	0.0000
C18	0.0011	0.0012	0.0001	0.0000	0.0005	0.0000	0.0000	0.0052	0.0005	0.0003	0.0001	0.0000	0.0003	0.0001	0.0002	0.0011	0.0008	0.0001	0.0000	0.0005	0.0000	0.0000	0.0000
C19	0.0151	0.0332	0.0037	0.0005	0.0313	0.0019	0.0039	0.0115	0.0022	0.0016	0.0001	0.0003	0.0133	0.0003	0.0059	0.0555	0.0005	0.0039	0.0101	0.0472	0.0001	0.0001	0.0001
C21	0.0015	0.0020	0.0001	0.0000	0.0003	0.0000	0.0001	0.0010	0.0001	0.0002	0.0001	0.0001	0.0003	0.0001	0.0019	0.0016	0.0000	0.0003	0.0002	0.0009	0.0000	0.0003	0.0000
N1	0.0027	0.0084	0.0004	0.0000	0.0032	0.0003	0.0014	0.0025	0.0004	0.0004	0.0002	0.0002	0.0059	0.0019	0.0014	0.0061	0.0002	0.0010	0.0044	0.0059	0.0000	0.0000	0.0000
N2	0.0182	0.0400	0.0046	0.0007	0.0385	0.0024	0.0066	0.0156	0.0041	0.0044	0.0013	0.0011	0.0555	0.0016	0.0061	0.0120	0.0009	0.0063	0.0088	0.0663	0.0001	0.0002	0.0002
O2	0.0011	0.0017	0.0001	0.0000	0.0009	0.0001	0.0001	0.0019	0.0000	0.0002	0.0001	0.0008	0.0005	0.0000	0.0002	0.0009	0.0000	0.0001	0.0003	0.0011	0.0000	0.0000	0.0000
O3	0.0034	0.0050	0.0004	0.0000	0.0031	0.0016	0.0014	0.0017	0.0002	0.0002	0.0001	0.0001	0.0039	0.0003	0.0010	0.0063	0.0001	0.0003	0.0017	0.0056	0.0000	0.0001	0.0000
O4	0.0036	0.0147	0.0006	0.0001	0.0054	0.0003	0.0010	0.0028	0.0002	0.0004	0.0002	0.0000	0.0101	0.0002	0.0044	0.0088	0.0003	0.0017	0.0024	0.0070	0.0000	0.0002	0.0001
O5	0.0239	0.0424	0.0056	0.0009	0.0460	0.0022	0.0074	0.0182	0.0032	0.0022	0.0003	0.0005	0.0472	0.0009	0.0059	0.0663	0.0011	0.0056	0.0070	0.0174	0.0001	0.0003	0.0002
S3	0.0003	0.0001	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001	0.0000	0.0000	0.0000	0.0000	0.0001	0.0000	0.0000	0.0001	0.0000	0.0000	0.0000	0.0001	0.0000	0.0000	0.0000
S6	0.0006	0.0011	0.0001	0.0000	0.0003	0.0000	0.0000	0.0003	0.0000	0.0000	0.0000	0.0000	0.0001	0.0003	0.0000	0.0002	0.0000	0.0001	0.0002	0.0003	0.0000	0.0000	0.0000
S5	0.0002	0.0005	0.0000	0.0000	0.0001	0.0000	0.0001	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001	0.0000	0.0000	0.0002	0.0000	0.0000	0.0001	0.0002	0.0000	0.0000	0.0000

	ALA	ARG	ASN	ASP	CYS	GLN	GLU	GLY	HIS	ILE	LEU	LYS	MET	PHE	PRO	SER	THR	TRP	TYR	VAL
ALA	0.0213	0.0055	0.0062	0.0066	0.0007	0.0083	0.0022	0.0048	0.0028	0.0040	0.0035	0.0012	0.0007	0.0060	0.0031	0.0041	0.0052	0.0032	0.0045	0.0053
ARG	0.0055	0.0275	0.0044	0.0073	0.0003	0.0021	0.0039	0.0034	0.0032	0.0046	0.0040	0.0005	0.0004	0.0072	0.0043	0.0036	0.0043	0.0044	0.0027	0.0044
ASN	0.0062	0.0044	0.0307	0.0053	0.0010	0.0058	0.0018	0.0064	0.0020	0.0039	0.0025	0.0013	0.0013	0.0058	0.0067	0.0020	0.0055	0.0011	0.0048	0.0047
ASP	0.0066	0.0073	0.0053	0.0315	0.0005	0.0048	0.0012	0.0044	0.0032	0.0029	0.0039	0.0019	0.0000	0.0081	0.0068	0.0055	0.0069	0.0021	0.0044	0.0038
CYS	0.0007	0.0003	0.0010	0.0005	0.0018	0.0002	0.0000	0.0003	0.0000	0.0005	0.0013	0.0002	0.0000	0.0005	0.0000	0.0000	0.0007	0.0004	0.0004	0.0007
GLN	0.0083	0.0021	0.0058	0.0048	0.0002	0.0254	0.0011	0.0038	0.0016	0.0029	0.0042	0.0007	0.0011	0.0040	0.0045	0.0028	0.0057	0.0000	0.0055	0.0042
GLU	0.0022	0.0039	0.0018	0.0012	0.0000	0.0011	0.0102	0.0009	0.0006	0.0017	0.0018	0.0006	0.0003	0.0035	0.0015	0.0013	0.0033	0.0015	0.0011	0.0022
GLY	0.0048	0.0034	0.0064	0.0044	0.0003	0.0038	0.0009	0.0132	0.0010	0.0028	0.0053	0.0006	0.0017	0.0078	0.0042	0.0026	0.0035	0.0026	0.0089	0.0039
HIS	0.0028	0.0032	0.0020	0.0032	0.0000	0.0016	0.0006	0.0010	0.0143	0.0025	0.0007	0.0015	0.0004	0.0017	0.0011	0.0027	0.0025	0.0024	0.0047	0.0030
ILE	0.0040	0.0046	0.0039	0.0029	0.0005	0.0029	0.0017	0.0028	0.0025	0.0228	0.0057	0.0008	0.0006	0.0096	0.0040	0.0025	0.0052	0.0028	0.0049	0.0035
LEU	0.0035	0.0040	0.0025	0.0039	0.0013	0.0042	0.0018	0.0053	0.0007	0.0057	0.0243	0.0013	0.0023	0.0073	0.0052	0.0046	0.0075	0.0030	0.0044	0.0046
LYS	0.0012	0.0005	0.0013	0.0019	0.0002	0.0007	0.0006	0.0006	0.0015	0.0008	0.0013	0.0055	0.0000	0.0008	0.0004	0.0010	0.0010	0.0009	0.0013	0.0002
MET	0.0007	0.0004	0.0013	0.0000	0.0000	0.0011	0.0003	0.0017	0.0004	0.0006	0.0023	0.0000	0.0051	0.0013	0.0022	0.0007	0.0018	0.0006	0.0010	0.0026
PHE	0.0060	0.0072	0.0058	0.0081	0.0005	0.0040	0.0035	0.0078	0.0017	0.0096	0.0073	0.0008	0.0013	0.0402	0.0057	0.0046	0.0083	0.0059	0.0087	0.0091
PRO	0.0031	0.0043	0.0067	0.0068	0.0000	0.0045	0.0015	0.0042	0.0011	0.0040	0.0052	0.0004	0.0022	0.0057	0.0240	0.0028	0.0044	0.0011	0.0034	0.0035
SER	0.0041	0.0036	0.0020	0.0055	0.0000	0.0028	0.0013	0.0026	0.0027	0.0025	0.0046	0.0010	0.0007	0.0046	0.0028	0.0147	0.0024	0.0022	0.0034	0.0023
THR	0.0052	0.0043	0.0055	0.0069	0.0007	0.0057	0.0033	0.0035	0.0025	0.0052	0.0075	0.0010	0.0018	0.0083	0.0044	0.0024	0.0272	0.0036	0.0050	0.0051
TRP	0.0032	0.0044	0.0011	0.0021	0.0004	0.0000	0.0015	0.0026	0.0024	0.0028	0.0030	0.0009	0.0006	0.0059	0.0011	0.0022	0.0036	0.0138	0.0030	0.0039
TYR	0.0045	0.0027	0.0048	0.0044	0.0004	0.0055	0.0011	0.0089	0.0047	0.0049	0.0044	0.0013	0.0010	0.0087	0.0034	0.0034	0.0050	0.0030	0.0289	0.0079
VAL	0.0053	0.0044	0.0047	0.0038	0.0007	0.0042	0.0022	0.0039	0.0030	0.0035	0.0046	0.0002	0.0026	0.0091	0.0035	0.0023	0.0051	0.0039	0.0079	0.0246

Figure 3.7: The three different kinds of atomic interaction frequencies for the same transferase protein as above. The top is the Tsai interaction frequencies, the middle is the Popelier interaction frequencies, and the bottom is the residue interaction frequencies. The matrices are symmetric along the main diagonal, and both the upper and lower halves sum to one.

	ALA	ARG	ASN	ASP	CYS	GLN	GLU	GLY	HIS	ILE	LEU	LYS	MET	PHE	PRO	SER	THR	TRP	TYR	VAL
ALA	86	46	46	48	11	66	12	65	16	33	44	7	6	46	39	46	56	24	32	60
ARG	46	8	22	41	2	9	20	58	10	24	21	2	2	34	15	26	31	12	10	31
ASN	46	22	48	42	2	36	8	47	7	14	13	10	9	32	44	19	43	8	22	27
ASP	48	41	42	26	6	28	12	49	31	19	19	10	0	40	30	40	34	14	29	23
CYS	11	2	2	6	4	2	0	2	0	4	9	2	0	4	0	0	4	2	4	6
GLN	66	9	36	28	2	18	5	41	15	19	14	7	6	27	25	23	27	0	24	31
GLU	12	20	8	12	0	5	8	15	6	12	18	2	2	10	18	10	18	0	7	6
GLY	65	58	47	49	2	41	15	80	14	35	54	6	17	57	68	34	61	17	53	51
HIS	16	10	7	31	0	15	6	14	4	10	3	4	2	17	12	10	15	5	12	16
ILE	33	24	14	19	4	19	12	35	10	24	50	2	7	42	27	22	44	15	28	40
LEU	44	21	13	19	9	14	18	54	3	50	40	12	16	40	46	40	36	12	23	39
LYS	7	2	10	10	2	7	2	6	4	2	12	4	0	4	0	4	10	0	6	4
MET	6	2	9	0	0	6	2	17	2	7	16	0	4	9	6	10	11	2	5	20
PHE	46	34	32	40	4	27	10	57	17	42	40	4	9	46	41	30	42	14	34	47
PRO	39	15	44	30	0	25	18	68	12	27	46	0	6	41	26	14	33	4	18	28
SER	46	26	19	40	0	23	10	34	10	22	40	4	10	30	14	28	42	12	22	23
THR	56	31	43	34	4	27	18	61	15	44	36	10	11	42	33	42	36	14	21	36
TRP	24	12	8	14	2	0	0	17	5	15	12	0	2	14	4	12	14	4	16	8
TYR	32	10	22	29	4	24	7	53	12	28	23	6	5	34	18	22	21	16	12	40
VAL	60	31	27	23	6	31	6	51	16	40	39	4	20	47	28	23	36	8	40	60

	ALA	ARG	ASN	ASP	CYS	GLN	GLU	GLY	HIS	ILE	LEU	LYS	MET	PHE	PRO	SER	THR	TRP	TYR	VAL
ALA	0.0188	0.0101	0.0101	0.0105	0.0024	0.0144	0.0026	0.0142	0.0035	0.0072	0.0096	0.0015	0.0013	0.0101	0.0085	0.0101	0.0122	0.0052	0.0070	0.0131
ARG	0.0101	0.0017	0.0048	0.0090	0.0004	0.0020	0.0044	0.0127	0.0022	0.0052	0.0046	0.0004	0.0004	0.0074	0.0033	0.0057	0.0068	0.0026	0.0022	0.0068
ASN	0.0101	0.0048	0.0105	0.0092	0.0004	0.0079	0.0017	0.0103	0.0015	0.0031	0.0028	0.0022	0.0020	0.0070	0.0096	0.0042	0.0094	0.0017	0.0048	0.0059
ASP	0.0105	0.0090	0.0092	0.0057	0.0013	0.0061	0.0026	0.0107	0.0068	0.0042	0.0042	0.0022	0.0000	0.0087	0.0066	0.0087	0.0074	0.0031	0.0063	0.0050
CYS	0.0024	0.0004	0.0004	0.0013	0.0009	0.0004	0.0000	0.0004	0.0000	0.0009	0.0020	0.0004	0.0000	0.0009	0.0000	0.0000	0.0009	0.0004	0.0009	0.0013
GLN	0.0144	0.0020	0.0079	0.0061	0.0004	0.0039	0.0011	0.0090	0.0033	0.0042	0.0031	0.0015	0.0013	0.0059	0.0055	0.0050	0.0059	0.0000	0.0052	0.0068
GLU	0.0026	0.0044	0.0017	0.0026	0.0000	0.0011	0.0017	0.0033	0.0013	0.0026	0.0039	0.0004	0.0004	0.0022	0.0039	0.0022	0.0039	0.0000	0.0015	0.0013
GLY	0.0142	0.0127	0.0103	0.0107	0.0004	0.0090	0.0033	0.0175	0.0031	0.0076	0.0118	0.0013	0.0037	0.0125	0.0149	0.0074	0.0133	0.0037	0.0116	0.0111
HIS	0.0035	0.0022	0.0015	0.0068	0.0000	0.0033	0.0013	0.0031	0.0009	0.0022	0.0007	0.0009	0.0004	0.0037	0.0026	0.0022	0.0033	0.0011	0.0026	0.0035
ILE	0.0072	0.0052	0.0031	0.0042	0.0009	0.0042	0.0026	0.0076	0.0022	0.0052	0.0109	0.0004	0.0015	0.0092	0.0059	0.0048	0.0096	0.0033	0.0061	0.0087
LEU	0.0096	0.0046	0.0028	0.0042	0.0020	0.0031	0.0039	0.0118	0.0007	0.0109	0.0087	0.0026	0.0035	0.0087	0.0101	0.0087	0.0079	0.0026	0.0050	0.0085
LYS	0.0015	0.0004	0.0022	0.0022	0.0004	0.0015	0.0004	0.0013	0.0009	0.0004	0.0026	0.0009	0.0000	0.0009	0.0000	0.0009	0.0022	0.0000	0.0013	0.0009
MET	0.0013	0.0004	0.0020	0.0000	0.0000	0.0013	0.0004	0.0037	0.0004	0.0015	0.0035	0.0000	0.0009	0.0020	0.0013	0.0022	0.0024	0.0004	0.0011	0.0044
PHE	0.0101	0.0074	0.0070	0.0087	0.0009	0.0059	0.0022	0.0125	0.0037	0.0092	0.0087	0.0009	0.0020	0.0101	0.0090	0.0066	0.0092	0.0031	0.0074	0.0103
PRO	0.0085	0.0033	0.0096	0.0066	0.0000	0.0055	0.0039	0.0149	0.0026	0.0059	0.0101	0.0000	0.0013	0.0090	0.0057	0.0031	0.0072	0.0009	0.0039	0.0061
SER	0.0101	0.0057	0.0042	0.0087	0.0000	0.0050	0.0022	0.0074	0.0022	0.0048	0.0087	0.0009	0.0022	0.0066	0.0031	0.0061	0.0092	0.0026	0.0048	0.0050
THR	0.0122	0.0068	0.0094	0.0074	0.0009	0.0059	0.0039	0.0133	0.0033	0.0096	0.0079	0.0022	0.0024	0.0092	0.0072	0.0092	0.0079	0.0031	0.0046	0.0079
TRP	0.0052	0.0026	0.0017	0.0031	0.0004	0.0000	0.0000	0.0037	0.0011	0.0033	0.0026	0.0000	0.0004	0.0031	0.0009	0.0026	0.0031	0.0009	0.0035	0.0017
TYR	0.0070	0.0022	0.0048	0.0063	0.0009	0.0052	0.0015	0.0116	0.0026	0.0061	0.0050	0.0013	0.0011	0.0074	0.0039	0.0048	0.0046	0.0035	0.0026	0.0087
VAL	0.0131	0.0068	0.0059	0.0050	0.0013	0.0068	0.0013	0.0111	0.0035	0.0087	0.0085	0.0009	0.0044	0.0103	0.0061	0.0050	0.0079	0.0017	0.0087	0.0131

Figure 3.8: The threshold residue interactions counts (above) and frequencies (below) of the transferase protein shown previously. Only the residues with C_βs that are within an 8 Å threshold are included. Because only one atom in each residue is counted, the counts are lower than for the Delaunay method. Once the counts are converted to frequencies, the methods become very similar again.

are a few different C_β categorizations for the Tsai and Popelier atomic groups but using these groups would not lead to interesting knowledge of how the atoms are interacting, so these classifications were not used.

There are 20 different amino acid residues, so a matrix of size 20x20 was created to count the interactions. When two amino acids were found to be interacting, one was added to the count of those amino acid interactions. This matrix is also symmetrical around the main diagonal. Figure 3.8 shows the threshold residue counts and frequencies for one protein. In the end, there are $20+19+...+2+1 = 210$ different possible amino acid interactions.

3.3.3 Surface Inner Atomic Frequencies

In addition to counting the interactions of the atoms, the frequencies on the atom surface and interior were also considered. This began as a feature similar to a feature from Ulfenborg 2020. Their second-best category of features was finding the counts of the atomic groups on the surface of the protein and converting them to frequencies. That paper used only the Tsai atomic groups to categorize the atoms. The current

Tsai Atomic Group	Surface Counts	Inner Counts	Surface Freq	Inner Freq
C3H0	441	1034	0.0578	0.1354
C3H1	240	374	0.0314	0.0490
C4H1	422	697	0.0553	0.0913
C4H2	685	427	0.0897	0.0559
C4H3	248	253	0.0325	0.0331
N3H0	18	48	0.0024	0.0063
N3H1	381	660	0.0499	0.0864
N3H2	177	39	0.0232	0.0051
N4H3	14	0	0.0018	0
O1H0	712	585	0.0933	0.0766
O2H1	86	76	0.0113	0.0100
S2H0	2	14	0.0003	0.0018
S2H1	0	2	0	0.0003

Table 3.3: Counts and frequencies of atoms classified by their Tsai atomic group on the surface and interior of the transferase protein used previously (uniprot id: Q43998). Carbon atoms are the most common. The atoms that participate in a disulfide bond were reclassified as S2H0.

work used the Tsai groups as well as the Popelier groups and the residues.

The work of calculating which atoms were on the surface and which were on the interior of the protein was done by the Triominoes program (Kemp 2019; Lee and Richards 1971). The program, as previously described, rolls a sphere over the protein. Any atom that is touched by the probe is on the surface and any molecule that is not touched by the probe is on the interior.

The frequencies of all three categories of groups on the inside of the protein were also counted. The Tsai classification has 13 different groups, so 13 counts were performed for the interior of the atom and 13 counts were done for the exterior. The count for each group was divided by total number of atoms in the protein and therefore the inner and outer frequencies of the Tsai groups sum to one. This was done for the Tsai atomic groups (creating 13 + 13 = 26 different categories), the Popelier atomic groups (creating 23 + 23 = 46 different categories), and the residues (creating 20 + 20 = 40 different categories). The counts and frequencies for an example protein are included in Tables 3.3, 3.4, and 3.5.

Popelier Atomic Group	Surface Counts	Inner Counts	Surface Freq	Inner Freq
C2	246	243	0.0322	0.0318
C3	576	390	0.0754	0.0511
C4	43	117	0.0056	0.0153
C7	6	8	0.0008	0.0010
C8	371	488	0.0486	0.0639
C9	42	30	0.0055	0.0039
C10	69	91	0.0090	0.0120
C12	189	351	0.0247	0.0460
C14	18	120	0.0024	0.0157
C15	40	34	0.0052	0.0045
C17	24	4	0.0031	0.0005
C18	16	44	0.0021	0.0058
C19	369	832	0.0483	0.1090
C21	27	33	0.0035	0.0043
O2	191	39	0.0250	0.0051
O3	399	708	0.0523	0.0927
O4	30	12	0.0039	0.0016
O5	56	64	0.0073	0.0084
N1	237	75	0.0310	0.0098
N2	475	510	0.0622	0.0668
S3	0	2	0	0.0003
S6	2	10	0.0003	0.0013
SS	0	4	0	0.0005

Table 3.4: The counts and frequencies of atoms classified by their Popelier atomic groups on the surface and interior of the transferase protein used previously (uniprot id: Q43998). Atoms in a disulfide bridge are reclassified as SS.

Residue	Surface Counts	Inner Counts	Surface Freq	Inner Freq
ALA	274	211	0.0359	0.0276
ARG	296	232	0.0388	0.0304
ASN	300	244	0.0393	0.0320
ASP	329	263	0.0431	0.0344
CYS	4	32	0.0005	0.0042
GLN	250	218	0.0327	0.0286
GLU	95	103	0.0124	0.0135
GLY	223	169	0.0292	0.0221
HIS	164	116	0.0215	0.0152
ILE	114	270	0.0149	0.0354
LEU	125	291	0.0164	0.0381
LYS	85	41	0.0111	0.0054
MET	20	76	0.0026	0.0100
PHE	205	455	0.0269	0.0596
PRO	258	204	0.0338	0.0267
SER	104	184	0.0136	0.0241
THR	207	297	0.0271	0.0389
TRP	64	188	0.0084	0.0246
TYR	171	333	0.0224	0.0436
VAL	138	282	0.0181	0.0369

Table 3.5: The counts and frequencies of the residues on the surface and interior of the transferase protein used previously (uniprot id: Q43998).

4

Methods

This project's focus was to learn which features from enzyme structures can be used to predict their optimal operational temperature and thereby learn which features are important for protein thermostability. First, the data was cleaned and then categorized in seven different ways: by the Delaunay Tsai atomic interactions (Tsai AI), by the Delaunay Popelier atomic interactions (Pop AI), by the Delaunay residue atomic interactions (Del Res AI), by the threshold atomic interactions (Thr Res SI), and by the frequency of atoms on the surface and interior of the protein for the Tsai (Tsai SI), Popelier (Pop SI), and residue groups (Res SI). Next, the correlations of each dataset were inspected to determine if there were correlations that would change the way that the datasets need to be analyzed. Finally, each dataset was analyzed using four different regression methods: SVR, random forest regression, elastic net regression, and group lasso regression. These methods are described in more detail below.

4.1 Pre-processing

The protein structures were taken from three different repositories: SWISS-MODEL, Protein Data Bank (PDB), and Modbase. Modbase is a database with protein structure models which were simulated by MODPIPE (Pieper et al. 2004). SWISS-MODEL is similar, it's a database with protein structure models which were simlulated by SWISS-MODEL (Kopp and Schwede 2004). PDB is a database of proteins each of which has been experimentally determined though much painstaking work and is therefore the most accurate (Berman et al. 2000). We are using proteins from sources in addition to PDB to increase the size of our dataset. All enzymes from those databases with experimentally verified optimal enzyme temperatures (topt) were used in this project. The script to download the proteins was provided by Martin Engqvist.

The data was pre-processed with a script written by Martin. Some of the proteins have multiple chains or models. Only the first was used. There were 10 proteins that included the positions for the hydrogen atoms. Our methods assume that we do not know the position of these atoms, so the hydrogen atoms were deleted. Then the test / train / val split was done by Martin with a script. He used CD-HIT (W. Li and Godzik 2006) to cluster the proteins based on their sequences. All proteins which clustered together were assigned to the same data set to ensure that the testing, training, and validation data sets were separate and did not contain extremely similar proteins. The purpose of separating related proteins is to prevent

data leakage. If related proteins are present both in the test and training sets, this will result in scores that are better than they should be.

There are 218 proteins that have extremely similar homologs or multiple structures in our dataset. Many of the multiple versions were computationally simulated proteins from SWISS-MODEL. These proteins were judged by the GMQE score if it was available, or the QMEAN score otherwise. Only the highest quality protein of the homolog group was kept in the dataset. When the homologs did not have a verified physical structure, no quality score was available, so a Qmean score was estimated using the SWISS-MODEL website and the protein with the highest score was kept. Where there were experimental and simulated proteins available, the best experimental proteins were kept. 316 homologous structures were removed. One of the training proteins, P19515, had many atoms with unknown positions, leaving a structure that was almost completely made of carbon atoms. This protein was removed from the dataset. After pre-processing was completed, there were 1122 training proteins, 126 test proteins, and 131 validation proteins.

4.2 Collinearity

Collinearity occurs when two or more predictor variables have a linear relationship. When different predictors are collinear, it means they share much of the same information. The degree of collinearity can be measured with different scores. Collinearity can cause problems with a regression analysis, particularly with the interpretation of the covariates. When there is collinearity present in the data, the estimates of the parameters are unstable; they can have large fluctuations with only small changes in the sample. This means that the model is unstable and it is impossible to say which variables are most important (Dormann et al. 2013). This is problematic for our analysis because the goal is to interpret the final model.

When covariates are correlated in a dataset, the methods needed to analyze the dataset are different. Therefore, the correlations of the covariates within each dataset were examined. Each dataset was analyzed separately to test its effectiveness. Therefore, the correlation for each category was examined individually as well. This allows us to compare this work to previous work. This was also done because the correlation graph with all covariates is too large to view properly.

Correlation is generally thought to be problematic if there are covariates that have Pearson correlation coefficients more extreme than 0.5 and -0.5 or alternately 0.7 and -0.7 (Dormann et al. 2013). Unfortunately, there are many correlations with coefficients above that in our datasets. The correlation plots are shown in Figures 4.1 - 4.7.

There is a lower amount of correlation in the residue data categories (Figures 4.3, 4.4, and 4.7) than the graphs with the Popelier and Tsai atomic groups (Figures 4.1, 4.2, 4.5, 4.6). The amount of extreme correlation (greater than 0.7 or less than -0.7) is much lower for the residue groups. The lower levels of correlation in the residue groups could be due to the fact that the categories are less well defined. Categorizing atoms by their residues groups together a wider variety of atoms than categorizing based on atomic group. Perhaps this leads to a "smearing" effect which lessens the amount of correlation.

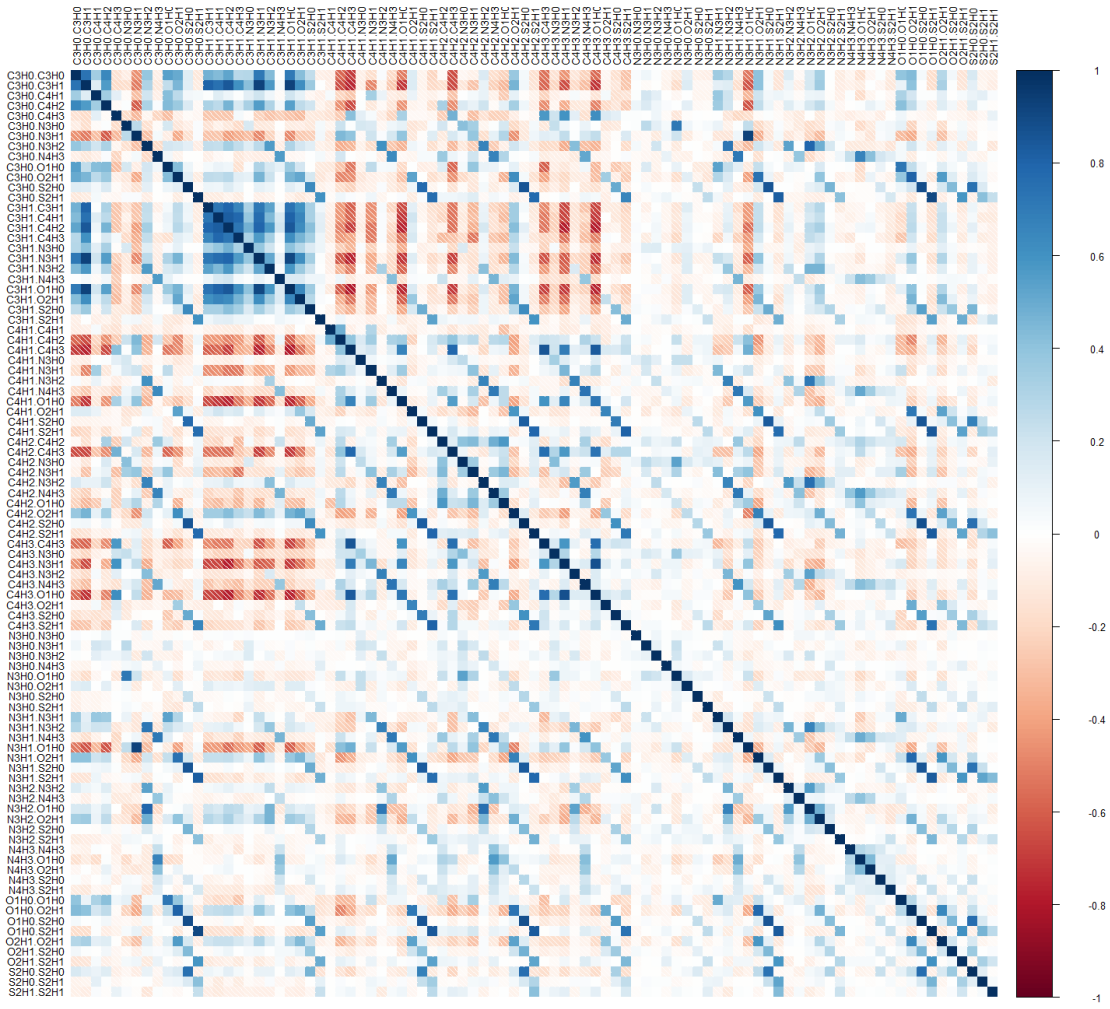

Figure 4.1: The correlation plot of each Tsai atomic interaction with every other Tsai atomic interaction. Out of 4186 total correlations, there are 236 with coefficients stronger than +/-0.5 and 85 with coefficients stronger than +/-0.7. The diagonal correlation trends are created when an atomic group is correlated to itself. For example, O1H0 interacting with O1H0, O2H1, S2H0, or S2H1 are all positively correlated to each other. When more O1H0 is found, it is interacting more with several groups.

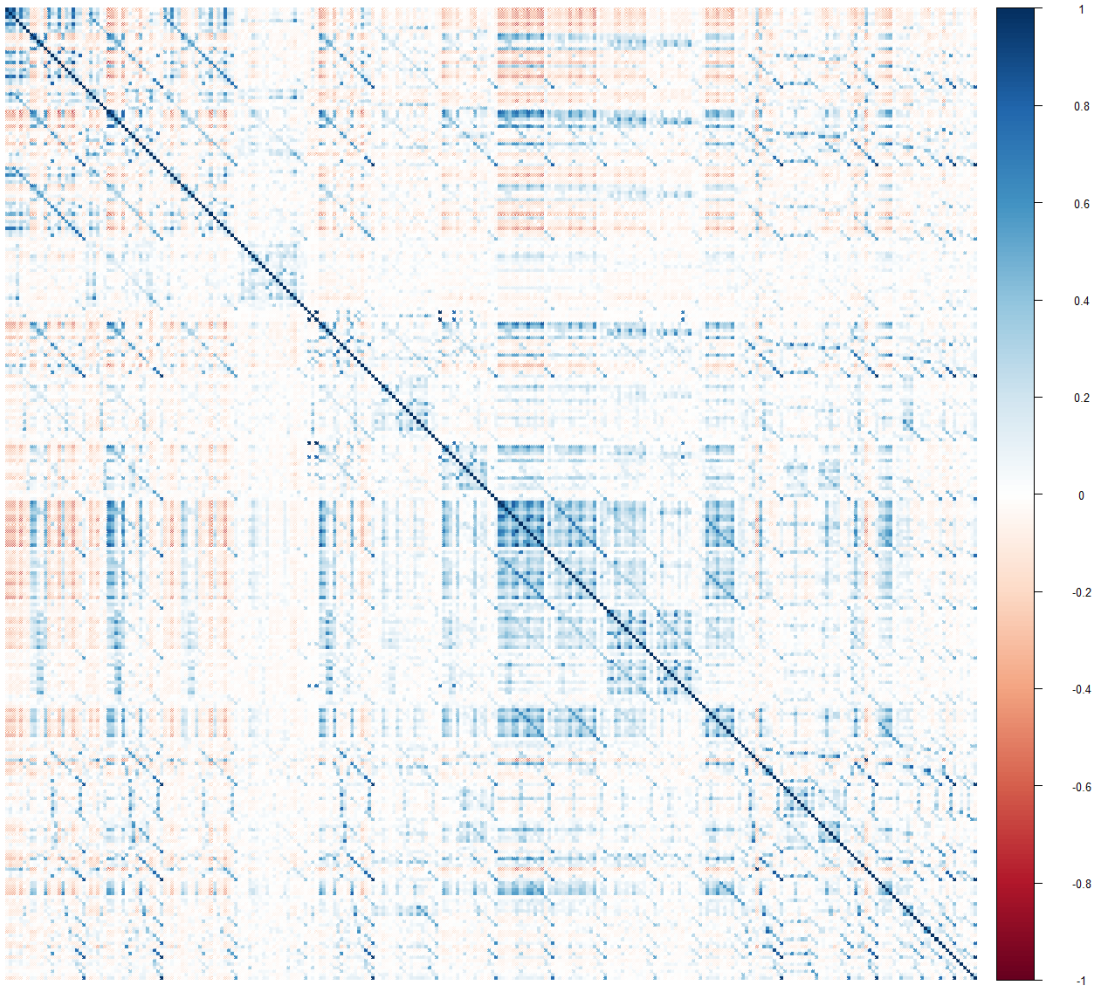

Figure 4.2: The correlation plot of each Popelier atomic group with every other Popelier atomic group. There are too many variables to list their names readably. The same diagonal trends from the previous plot are found in this plot. Out of 38,226 total correlations, there are 856 with coefficients stronger than +/-0.5 and 164 with coefficients stronger than +/-0.7.

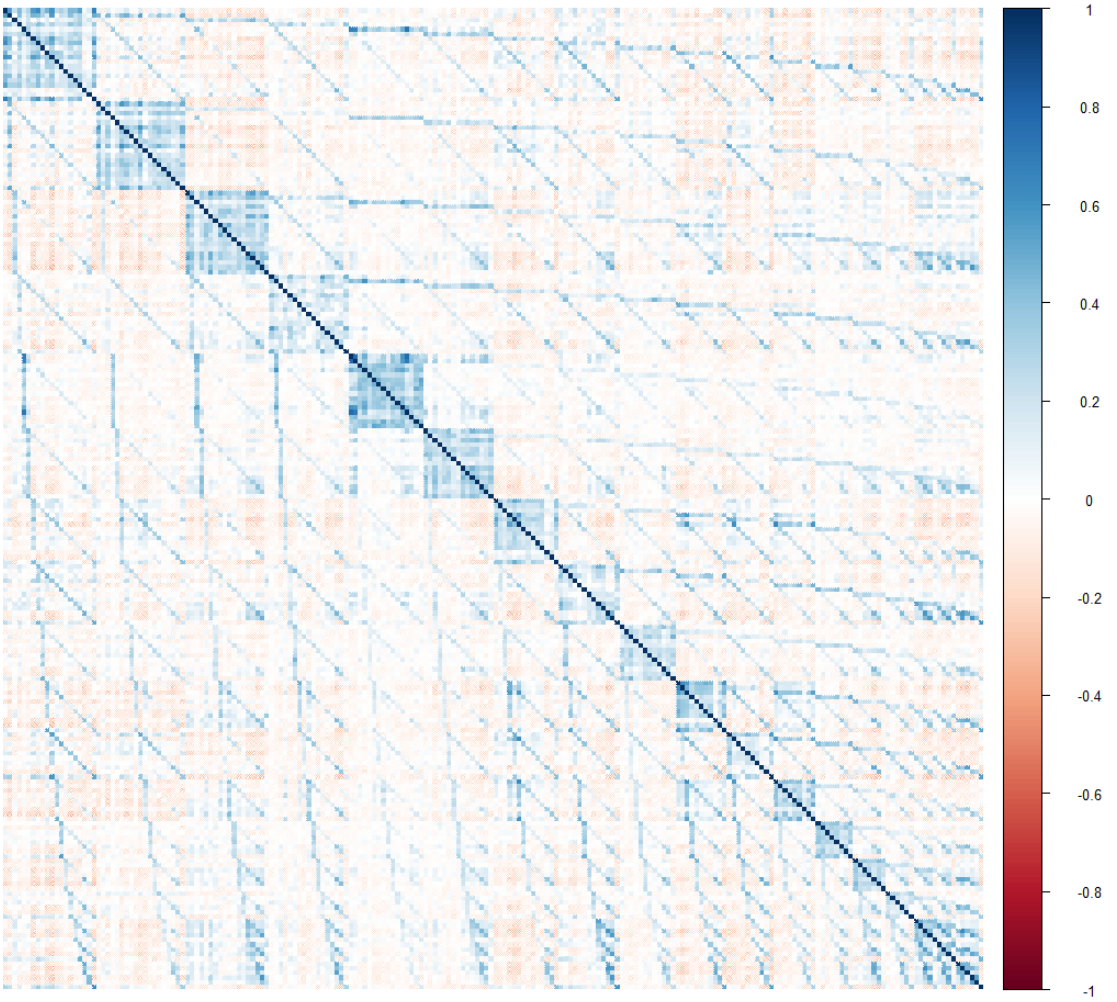

Figure 4.3: The correlation plot of each Delaunay residue atomic group with every other Delaunay residue atomic group. The upper left blue box has all the proteins that begin with alanine, the next blue box has all the proteins that begin with arginine, and so on. It appears that the residue frequencies that contain the same amino acid residue are correlated with each other. Out of 22,155 total correlations, there are 87 with coefficients stronger than +/-0.5 and 4 with coefficients stronger than +/-0.7. Although the residues are correlated with each other, this correlation lower than the correlations found in the Popelier and Tsai atomic group interaction datasets, especially at the more extreme values. The residues have very strong diagonal trends–they are very correlated with themselves, though the strength of the correlation is low.

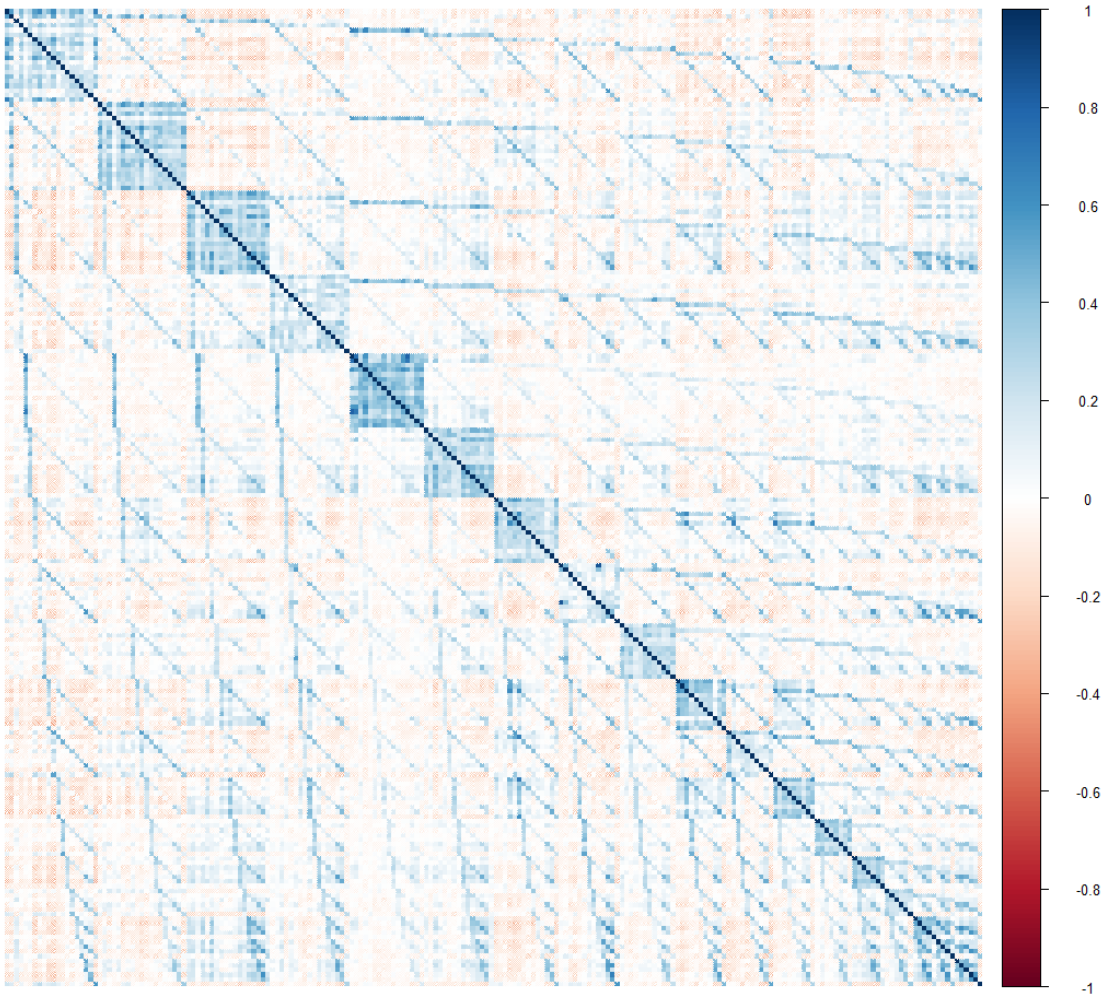

Figure 4.4: The correlation plot of each threshold residue atomic group with every other threshold residue atomic group. This correlation plot is nearly identical to the previous plot, indicating that the Delaunay and threshold residue atomic interactions are very similar. Out of 22,155 total correlations, there are 147 with coefficients stronger than +/-0.5 and 5 with coefficients stronger than +/-0.7.

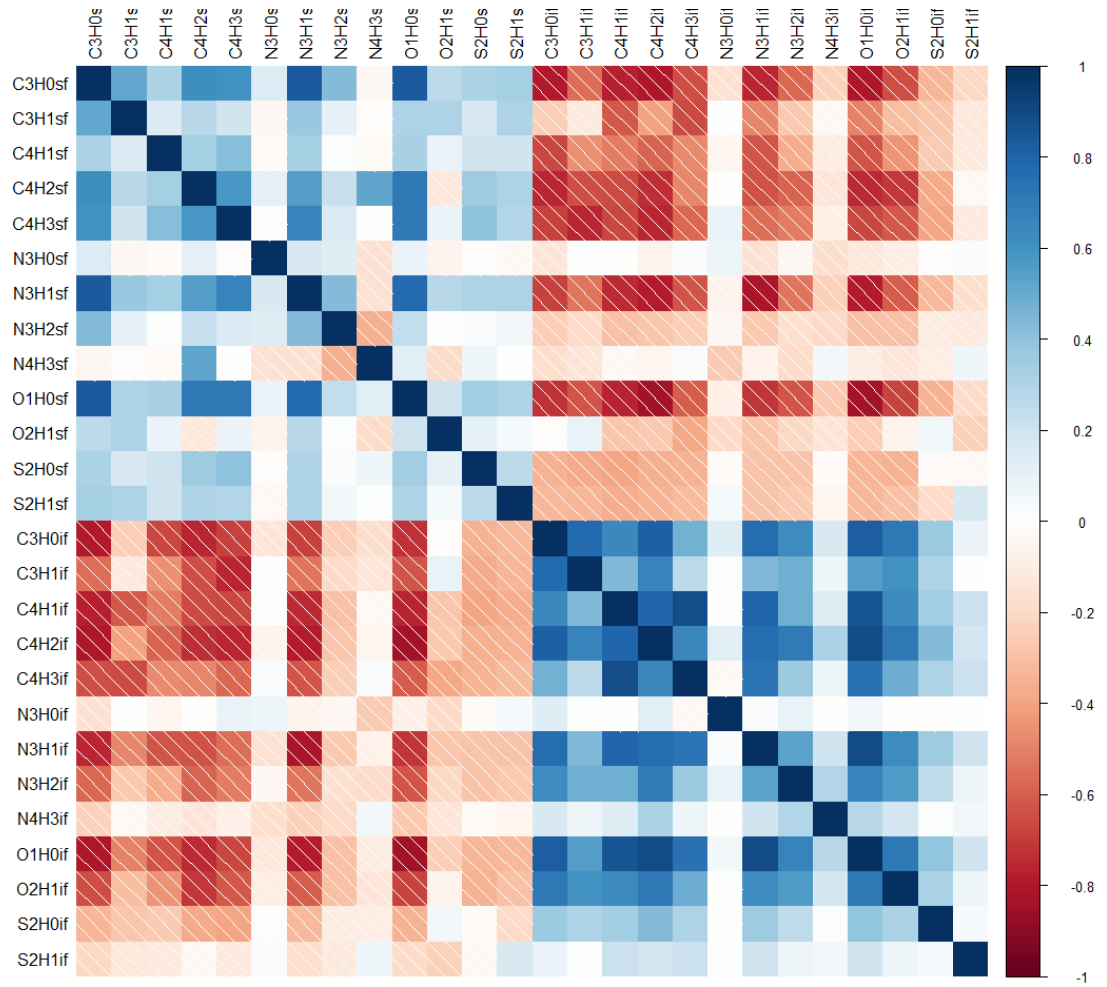

Figure 4.5: The correlation plot of each Tsai atomic group frequency on the protein surface and inside the protein. Most if the internal frequencies are correlated with each other and most of the surface frequencies are correlated with each other. Out of 351 total correlations, there are 91 with coefficients stronger than +/-0.5 and 42 with coefficients stronger than +/-0.7.

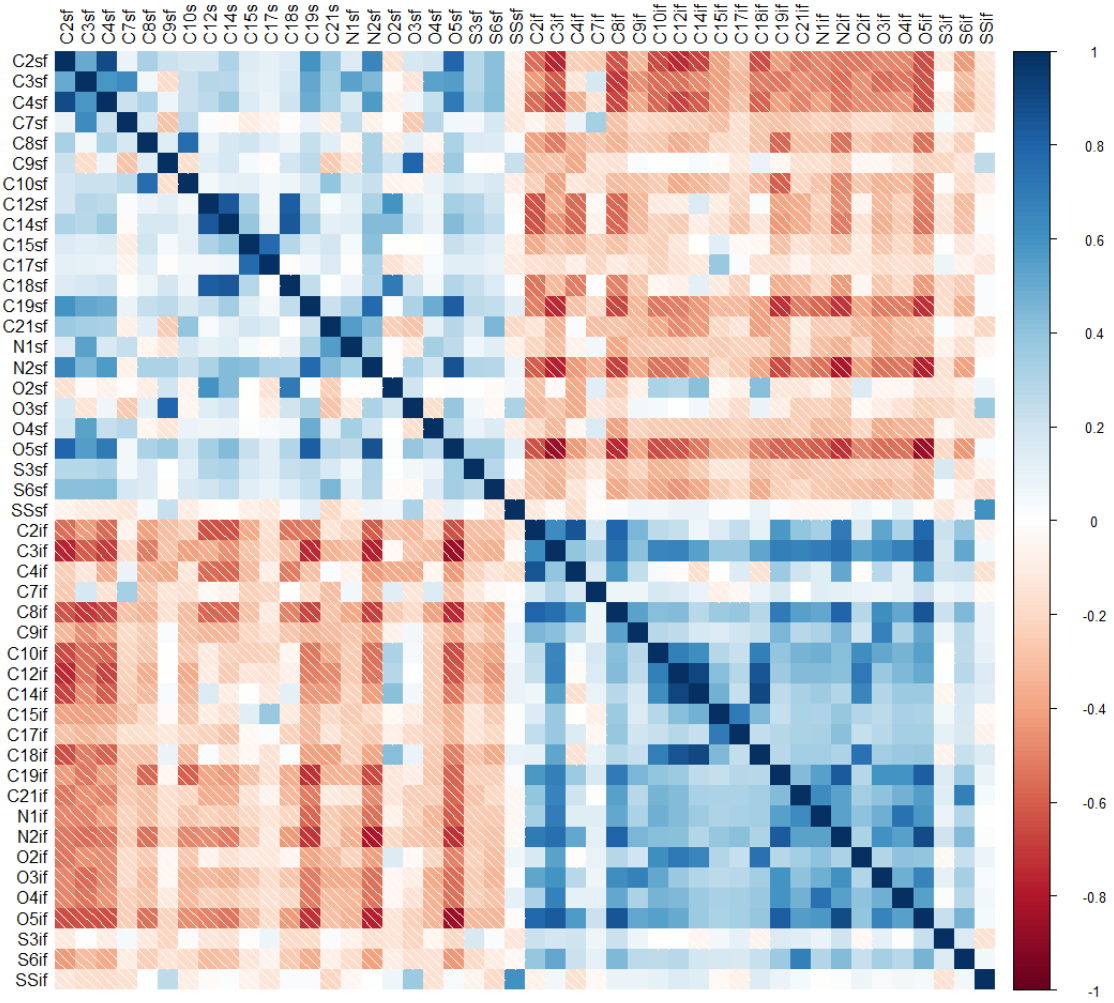

Figure 4.6: The correlation plot of each Popelier atomic group frequency on the protein surface and inside the protein. Most if the internal frequencies are correlated with each other, and most of the surface frequencies are correlated with each other. Out of 1081 total correlations, there are 173 with coefficients stronger than +/-0.5 and 47 with coefficients stronger than +/-0.7.

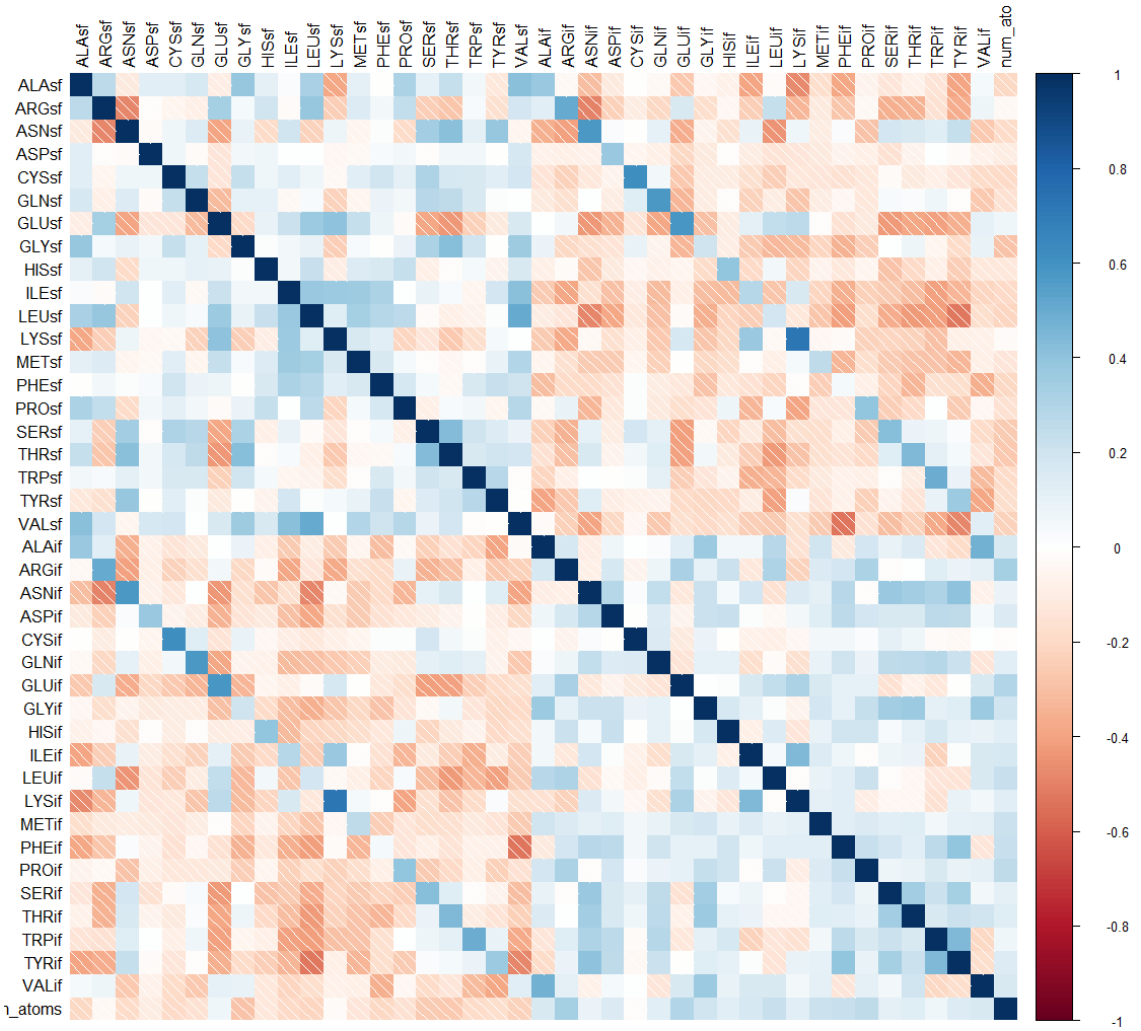

Figure 4.7: The correlation plot of each residue frequency on the protein surface and inside the protein. Most if the internal frequencies are correlated with each other, and most of the surface frequencies are correlated with each other. Out of 861 total correlations, there are 9 with coefficients stronger than +/-0.5 and 1 with a coefficient stronger than +/-0.7.

There is a high degree of multicollinearity in these datasets. These datasets are compositional–made of things that sum to 1. Compositional datasets are collinear because when there is more of something, there must necessarily be less of another Dormann et al. 2013. There may be an amount of intrinsic collinearity as well. Perhaps some atoms are often found near each other because of the chemical bonds in proteins. We did lessen this by excluding any atoms in the same amino acid and any atoms in the protein backbone of neighboring amino acids, but a lot of correlation persists.

4.3 Regression Methods

Regression is the best method to use for our analysis because we want to know which covariates are the most useful in predicting the optimal temperature. This could reveal which parts of the protein structures are most essential for maintaining structural integrity. Regression is appealing because the covariates are interpretable– it is possible to analyze the covariates and find the strength of their effect, as well as if they positively or negatively affect the optimal temperature.

Traditional ordinary least squares (OLS) regression is not able to accurately analyze a dataset with correlated covariates. Therefore, different regression methods were used which were purportedly better. These methods were: random forest regression, support vector regression (SVR), elastic net regression, and group lasso regression. Elastic-net and group lasso are forms of linear regression that are useful for this dataset because they can perform model selection and treat correlated covariates in the same way. They are forms of regularized regression. Regularized regression adds a penalty to the OLS regression. This means feature selection (deciding which covariates will be included in the model) becomes part of the model estimation process. Choosing a model with a smaller number of variables is desirable when there are datasets with many features (also known as covariates) because it will create a more manageable model which is easier for humans to understand. Elastic net is a combination of two common kinds of regularized regression: lasso regression and ridge regression. Support vector regression and random forest regression are quite different from (OLS) regression. All of these methods are explained below.

4.3.1 Ordinary Least Squares Regression

Ordinary least squares (OLS) regression assumes that the relationship between the dependent variable y and the independent variables X is linear. There is also an assumed error ε which adds random noise to the relationship. The model takes the form:

$$y = \beta_0 + \beta_1 x_{i1} + ... + \beta_p x_{ip} + \varepsilon_i = x_i\beta + \varepsilon_i, \quad i = 1, ..., n$$

This can be be summarized by writing it in matrix notation

$$y = X\beta + \varepsilon$$

In the matrix notation, y is a vector of values y_i $(i = 1...n)$ called the dependent variables. X is an $n \times p$ matrix of row-vectors x_i called the independent variables. The βs form a p-dimensional vector also known as the regression coefficient. They are the numbers by which the data values x are multiplied. They are chosen so as to minimize the distance between the actual and predicted data. The variable ε is also a vector of values ε_i called the error or noise. These variables are all included in the different regression variants, and they have the same meanings. Often when the formulas are discussed, the error ε is not included, but it is always assumed to exist.

4.3.2 Ridge Regression

Ridge regression was introduced by Hoerl and Kennard 1970. It begins with the OLS equation with a penalty added to regularize the data.

$$\hat{\beta}_{ridge}(\lambda) = arg\min ||y - X\beta||_2^2 + \lambda||\beta||_2^2$$

where λ is a hyper-parameter between 0 and 1 that determines the strength of the penalty, the lower 2 on the penalty term indicates the use of Euclidean norm, and the upper 2 means it is squared. The penalty term $\lambda||\beta||_2^2$ constrains the β parameters. When λ is high, the β values are penalized when they grow large. A λ value close to zero means that the βs are not penalized very much. Although the β values are smaller, ridge regression does not set any β coefficients equal to zero.

Ridge regression uses the L2 norm, or Euclidean distance. Because the ridge penalty is convex, it exhibits a grouping effect where highly correlated variables have similar coefficients (Zou and Hastie 2005). Ridge regression is good at treating correlated variables in the same way—their β coefficients will be similar. This is a beneficial outcome of ridge regression. However, it also has the negative property of being non-parsimonious because all covariates are included in the model. Using ridge regression by itself for this dataset would result in a very large model (with all variables included) that is hard to interpret. Therefore, another penalty, the lasso penalty, was added for model selection.

4.3.3 Lasso Regression

Lasso regression begins with the same OLS equation as linear regression but adds a different penalty than ridge regression.

$$\hat{\beta}_{lasso}(\lambda) = arg\min ||y - X\beta||_2^2 + \lambda||\beta||_1$$

The λ value again indicates the strength of the penalty to the β values. When it is large (close to 1), the β values are smaller. The subscript 1 on the penalty term indicates the use of the L1 norm. Lasso regression uses the L1 instead of the L2 norm; it uses Manhattan distance instead of Euclidean distance. Because of this, many coefficients are forced to become zero and a sparse solution is produced. If any of the variables are not very useful, their β coefficients will become zero instead of just being a small number. As a result, are fewer variables in the final model, which

makes the model more simple and more useful for interpretation. This property means that lasso regression is useful for model selection.

A lasso has some potential problems. Each variable is considered separately, and therefore correlated variables are not necessarily treated in the same way. When a cluster of variables has high pairwise correlation, the lasso arbitrarily selects only one variable from the cluster (Zou and Hastie 2005). If there are many non-relevant variables that are correlated with relevant variables, this can lead to the selection of a non-optimal model.

4.3.4 Elastic Net

Elastic Net regression was introduced by Zou and Hastie 2005 and combines the lasso regression penalty with the ridge regression penalty.

$$arg\min_{\beta}\frac{1}{2}\left|\left|y - X\beta\right|\right|_2^2 + \lambda\Big(\frac{1-\alpha}{2}||\beta||_2^2 + \alpha||\beta||_1\Big)$$

This is a hybrid of the L1 and L2 norms. The ridge penalty ensures that all correlated variables have similar coefficients. The lasso penalty picks just a few of the terms to have a coefficient and and makes the rest equal 0. This combination of both penalties means elastic-net regression groups correlated covariates together and either keeps or eliminates the group as a whole.

There are two hyper-parameters to optimize when using elastic net. The first is α, which is a measure of the mix of the lasso (L1) and ridge (L2) penalties. When $\alpha = 0$ then ridge regression is performed and when $\alpha = 1$, lasso regression is performed. Any number in between will result in an elastic net regression with a mix of the ridge and lasso penalties. When α is closer to 0, the penalty is most similar to ridge regression, with only a little lasso regression used. The second is λ, which is a measurement of how much the data is regularized. When λ is high, the coefficients are forced to have a normal distribution with a mean of 0. The model will be more simple. A low λ (close to 0) will result in β coefficients that have a flatter distribution. When λ is 0, the penalty terms have no effect and coefficients are not regularized–it is the same as performing ordinary least squares. Their values will be more similar to each other and few of them will become 0, resulting in a more complex model. λ has no upper limit, but a value of 5 would be quite high.

4.3.5 Sparse Group Lasso Regression

Group lasso regression is similar to elastic net regression in that it is a version of penalized linear regression. The difference is that there are user-defined groups that are input into the equation. These groups are forced to have similar coefficients. This can be useful because when groups form in data, the whole group should be treated similarly. Similar variables should either be in the model together or be forced to have a coefficient of 0. The elastic net can find correlated clusters, but only does so for highly correlated variables.

Group lasso from Yuan and Lin 2006 solves:

$$arg\min_{\beta}\frac{1}{2}\left|\left|y - X\beta\right|\right|_2^2 + \lambda\sum_{k=1}^{K}\left|\left|B_k\right|\right|_2$$

where B_k is a vector of coefficients β_i for the k-th group and λ is a measure of how much the data is regularized. By regularizing, group lasso encourages the clusters of variables to have a β coefficient of zero or non-zero with similar coefficients.

Sparse group lasso from Simon et al. 2013 gives us sparsity of groups and within each group. β can be zero for a whole group but for the groups that are non-zero, some of the β coefficients are still allowed to be set to zero. The equation for sparse group lasso is

$$arg\min_{\beta}\frac{1}{2n}\left|\left|y - \sum_{l=1}^{m}X^{(l)}\beta^{(l)}\right|\right|_2^2 + (1-\alpha)\lambda\sum_{l=1}^{m}\sqrt{p_l}\left|\left|\beta^{(l)}\right|\right|_2 + \alpha\lambda||\beta||_1$$

where $X^{(l)}$ is the submatrix of X with columns corresponding to the predictors in group l, $\beta^{(l)}$ is the coefficient vector of that group, p_l is the number of covariates in group l, and λ is the regularization coefficient. $\alpha \in [0, 1]$ is a convex combination of the lasso and group lasso penalties (where $\alpha = 0$ has a group lasso fit and $\alpha = 1$ has a lasso fit). It is different from elastic net because the L2 penalty is undifferentiable at zero, which means the β values for some groups are set to zero and completely removed from the model. This will result in a parsimonious model.

4.3.6 Random Forest Regression

Random forest regression is quite different from OLS regression. A random forest is made by combining many decision trees. An example of one decision tree is given in Figure 4.8. This tree examines different features of real trees and predicts an age in years. A random forest is made of many trees, each of which includes different features. Random forest regression calculates the average of all of the decision trees' predictions to create an estimate. The result is not a straight line predicting the outcome, as would be returned from linear regression, but a line with many steps and jumps, as seen in Figure 4.9. This creates a model that can fit the training data very well, but there is also a danger of overfitting the data, especially when there is correlation in the data. Correlation in the data can cause the correlated covariates to be over-weighted in a random forest model (Tolosi and Lengauer 2011). The resulting model is less accurate.

We are mainly interested in performing regression so that we may interpret the model. Interpreting the results of a random forest can be difficult because it is an amalgamation of many decision trees. One decision tree alone with a depth of 10 can contain thousands of decision boundaries. Combining many together in a random forest is more difficult. It is possible to partially interpret them by following the paths in the forest that lead to particular y-values. We can see which explanatory variables lead to the decision with the software package treeinterpreter (Saabas, Ando 2015). This does not lead to a perfect understanding of the model but can reveal a glimpse of what it is doing.

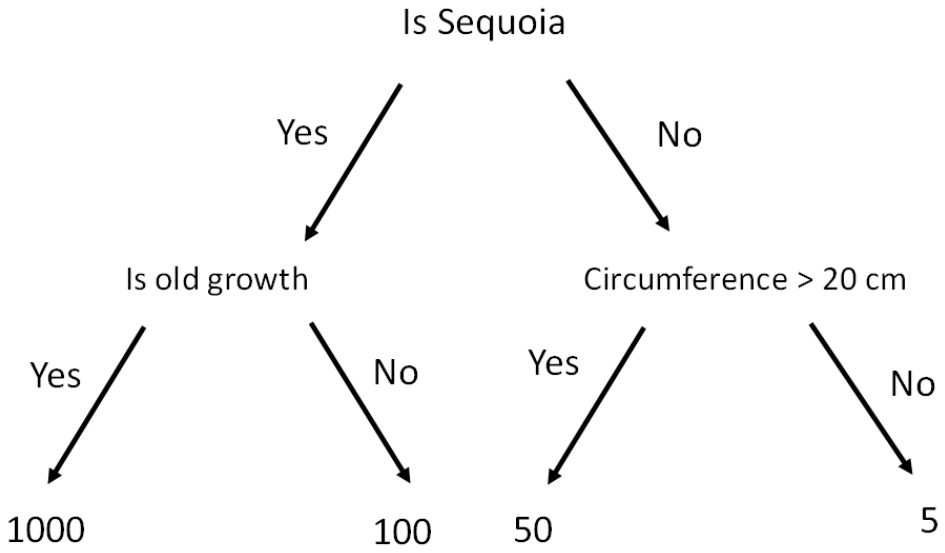

Figure 4.8: An example decision tree. This tree predicts the age of a tree in years. A decision tree begins at the root node. Branches examine one variable at a time and the leaf nodes contain the final outcomes.

4.3.7 Support Vector Regression

Support vector regression (SVR) performs linear regression in higher dimensional space. Support vector machines uses the kernel-trick to project the data to a higher-dimensional space (Awad and Khanna 2015). It then fits the data to a hyperplane (or a multidimensional line). The hyperplane $y = \beta X + b + \epsilon$ is created where ϵ is the distance from the hyperplane. When ϵ is added, a tube is formed around the function. Any errors that are within the tube are ignored, and errors outside the tube are penalized. Changing the value of ϵ changes the radius of the tube. Support vector regression uses all of the training data to choose the optimal hyper-parameters. After the hyper-parameters are chosen, SVR uses a subset of the training data, called support vectors, for future prediction (Awad and Khanna 2015). Support vector regression is often better than simple linear regression at making predictions because it can easily capture non-linearity.

4.4 Scoring

Different scoring methods were used to find the best models and to compare the models to each other.

The Mean Squared Error, or MSE, is the sum of the squared residuals.

$$MSE = \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n-2}$$

The residuals are the difference between what the model predicts and what the response variable actually is. RMSE is the square root of the MSE. It was used

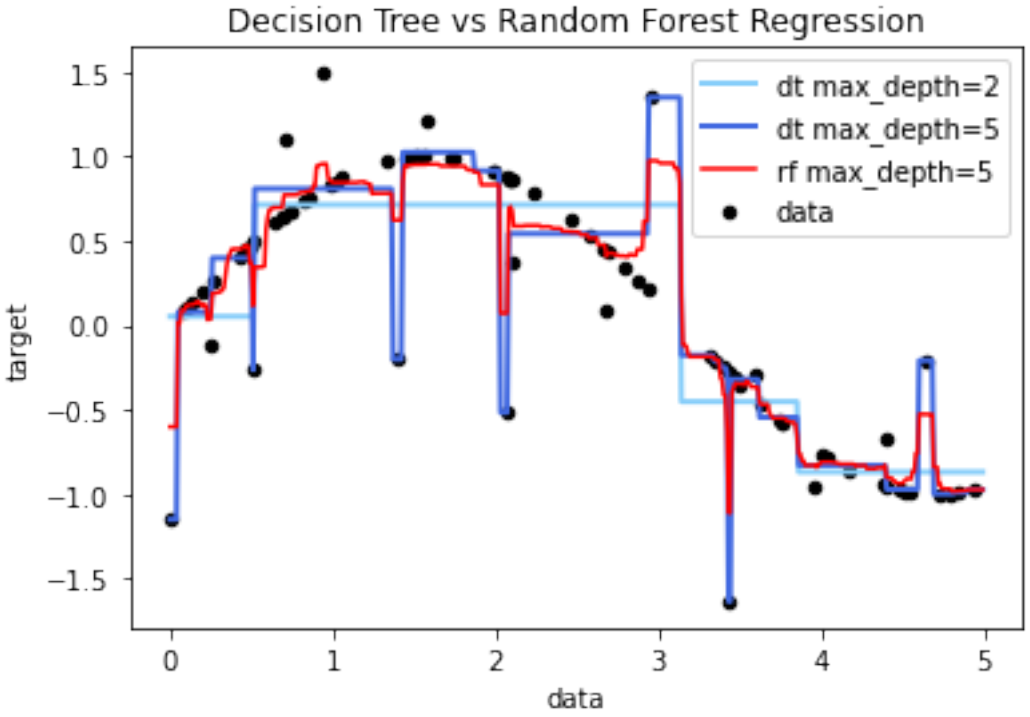

Figure 4.9: This is an example of the boundary line for a decision tree regressor and a random forest regressor. The resolution of a decision tree regressor depends on the max depth the tree is allowed to have. We have a decision tree with a max depth of 2, which is under-fitting the data, and a decision tree with a max depth of 5, which is over-fitting that data. The under-fit tree (light blue) has a boundary that doesn't capture all of the points because it is not flexible enough. The over-fit tree (dark blue) is too flexible and moves to fit every data point, which will not generalize well. The random forest tree (red) is a better fit for the data because it averages many decision trees.

during the model selection process.

R^2 is used to evaluate the results after the final models are selected.

$$R^2 = 1 - \frac{\text{Unexplained Variation}}{\text{Total Variation}}$$

The highest possible value is 1, and getting that score would mean that the model explains every change in the response variable. If the score is lower than 1, there are things that are not accounted for in the model that cause the response variable to have a different value than expected.

5

Results

The protein structures have been analyzed in seven different ways: by their Delaunay Tsai atomic interactions (Del Tsai AI), by their Delaunay Popelier atomic interactions (Del Pop AI), by their Delaunay residue atomic interactions (Del Res AI), by their threshold residue atomic interactions (Thr Res AI), by their Tsai surface inner frequencies (Tsai SI), by their Popelier surface inner frequencies (Pop SI), and by their residue surface inner frequencies (Res SI). The number of atoms were added to every dataset except for the threshold residue atomic interaction, because the threshold method didn't use all the atoms. The seven datasets were analyzed in four different ways–with Support Vector Regression (SVR), Random Forest Regression, Elastic Net Regression, and group lasso regression.

Three of the methods that were used come from SciKit-learn (Pedregosa et al. 2011) (SVR, Random Forest, and Elastic Net). For these methods, grid search with 5-fold cross-validation was used to find the optimal hyper-parameters. Group lasso was installed separately as its own package (yngvem 2019). Grid search was implemented manually for this method. The root mean squared error (RMSE) was used as the criterion for goodness of fit during training for all methods. The models with the lowest RMSE for each dataset were selected as the best models, and then the testing and training R^2 values were calculated.

5.1 Elastic Net

A grid search was performed using 5-fold cross validation to find the best hyper-parameters with 15 evenly spaced α values from 0.01 to 1 (values less than 0.01 are unstable for SkiKit Learn) and 15 evenly spaced lambda values from 0.01 to 1.5.

The best hyper-parameters for each data category were:

1. All data combined
   λ: 0.54, l1 ratio / α: 1
   R^2 train: 0.48, R^2 test: 0.37
2. Tsai AI λ: 0.33, l1 ratio / α: 0.43
   R^2 train: 0.21, R^2 test: 0.21
3. Popelier AI λ: 0.65, l1 ratio / α: 0.86
   R^2 train: 0.27, R^2 test: 0.21
4. Residue Del AI λ: 0.54, l1 ratio / α: 0.51
   R^2 train: 0.38, R^2 test: 0.28
5. Residue Thr AI λ: 0.44, l1 ratio / α: 0.79
   R^2 train: 0.38, R^2 test: 0.28

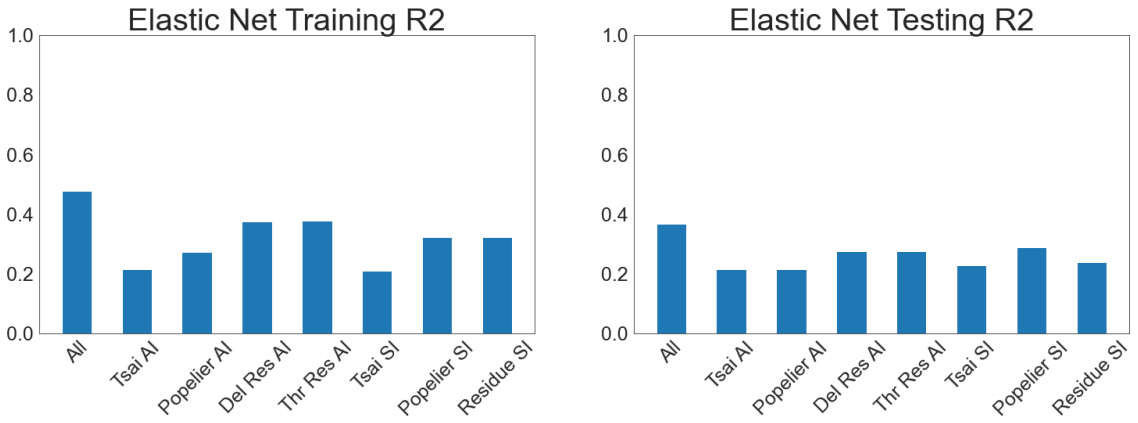

Figure 5.1: The R^2 values for elastic net regression for different categories of data. There is not over-fitting from this method as the training R^2 scores are not very much higher than the testing R^2 scores. The best performing category is the whole dataset. The best sub-categories are: first, the Popelier Surface Inner Frequencies, and second, the residue interactions (both the Delaunay and Popelier score almost exactly the same.)

6. Tsai Surface Inner Freq λ: 0.01, l1 ratio / α: 1.0
   R^2 train: 0.21, R^2 test: 0.23
7. Popelier Surface Inner Freq λ: 0.01, l1 ratio / α: 1.0
   R^2 train: 0.32, R^2 test: 0.29
8. Residue Surface Inner Freq λ: 0.22, l1 ratio / α: 0.93
   R^2 train: 0.32, R^2 test: 0.24

The R^2 results are shown in Figure 5.1. The whole dataset combined scores higher than any one piece. This method does not over-fit very much on the training dataset and is easy to interpret, so the best models will be examined despite their low scores. The models that have the highest R^2 testing values are the Popelier Surface Inner Frequency at 0.29, and the Residue Atomic Interactions (both the threshold and Delaunay versions), which both have a testing R^2 value of 0.28. The coefficients with the most extreme values (positive or negative) have the most influence on the model. Negative values mean an inverse correlation with the explanatory variable, temperature.

First, the Popelier Surface Inner Frequency will be examined. The whole model is shown in Figure 5.1 The most extreme value for the Pop SI is for O3. O3 comes from an alcohol O bonded to an alkyl group. It is present only in serine and threonine, both of which are polar amino acids. It is interesting that the presence of an O3 is most important when it is on the exterior of the protein, but it is also important when it is on the interior. It has negative values both times, therefore it is associated with protein instability. Threonine on the protein surface was found to be important by the random forest residue surface inner frequency model, but it had a positive value then. The next most extreme value is for C10, which is located on the C_α in glycine, serine, and tyrosine. It has a strong positive value when found on the interior and surface.

The second category of data that will be examined is the residue atomic in-

Atom Type	β coef value	Atom Type	β coef value
O3 sf	-13.411	C8 if	0.694
O3 if	-7.577	number atoms	0.865
C21 sf	-6.076	SS sf	0.956
C2 sf	-5.418	C12 sf	1.072
C19 sf	-5.181	C18 sf	1.239
C17 sf	-4.472	C14 if	1.391
C3 if	-4.339	S6 sf	1.501
S3 if	-3.237	C15 if	1.538
C19 if	-3.039	C14 sf	1.568
C3 sf	-2.96	C7 if	1.694
C21 if	-2.489	SS if	1.773
C12 if	-2.449	C2 if	2.102
N1 if	-1.916	O4 if	2.113
O2 sf	-1.736	S6 if	2.358
C17 if	-1.511	N2 sf	2.511
S3 sf	-1.277	O5 sf	3.363
O4 sf	-1.067	C9 if	3.565
N1 sf	-0.429	O5 if	3.637
O2 if	-0.132	C4 if	4.842
C8 sf	0	C4 sf	6.275
C15 sf	0	C9 sf	7.024
C18 if	0	C10 sf	7.832
N2 if	0	C10 if	8.14
C7 sf	0.095		

Table 5.1: The Popelier Surface Inner Frequency had the best training R^2 scores out of all of the elastic net models. The β coefficients are listed in order from smallest to greatest. Sf stands for surface frequency and if stands for inner frequency. There are few enough covariates in this dataset that all coefficients have been shown. The amount of O3 and C10 on the surface and the interior have the biggest influence in this model.

teractions. Cys-Cys (1.84) has the highest positive score, which confirms that the model is working well, as the presence of disulfide bridges is known to be stabilizing. The next highest score is for the number of atoms (1.31), meaning that a larger protein is more stable. The next most positive interactions are Glu-Val (1.25), Ile-Tyr (1.23), and Glu-Tyr (1.17). The interactions that most negatively affect the model are Gln-Lys (-1.55), Cys-Ile (-1.36), Pro-Ser (-1.36), Cys-Leu(-1.11) and Arg-Gln (-1.09).

Table 5.2: Coefficients with non-zero values from residue-residue Delaunay atomic interactions. There are 211 coefficients in the full model, but only 117 in the final model. The 10 most extreme values are darker blue and the 10 next extreme are lighter blue.

interacting amino acids	β coef value	interacting amino acids	β coef value
ALA-ARG	-0.05	GLN-MET	-0.018
ALA-ASP	-0.785	GLN-PHE	-0.541
ALA-CYS	-0.985	GLN-PRO	0.551
ALA-GLU	-0.609	GLN-SER	-0.07
ALA-GLY	-0.195	GLN-THR	-0.232
ALA-ILE	1.056	GLU-GLU	0.355
ALA-LYS	-0.006	GLU-GLY	0.24
ALA-PHE	0.161	GLU-HIS	0.247
ALA-PRO	-0.149	GLU-ILE	0.548
ALA-SER	-0.091	GLU-LEU	0.493
ALA-TYR	0.605	GLU-LYS	0.177
ALA-VAL	0.238	GLU-PRO	0.906
ARG-ARG	-0.021	GLU-SER	-0.496
ARG-ASN	-0.048	GLU-TRP	0.409
ARG-CYS	-0.331	GLU-TYR	1.172
ARG-GLN	-1.094	GLU-VAL	1.254
ARG-GLU	0.871	GLY-HIS	-0.316
ARG-GLY	0.194	GLY-PHE	0.021
ARG-HIS	-0.292	GLY-TRP	0.328
ARG-MET	-0.211	HIS-LEU	-0.247
ARG-SER	-0.935	HIS-LYS	-0.695
ARG-THR	-0.813	HIS-PRO	-0.053
ARG-TRP	-0.54	HIS-SER	-0.277
ARG-VAL	0.529	HIS-TYR	-0.217
ASN-CYS	0.142	HIS-VAL	-0.583
ASN-GLN	-0.642	ILE-ILE	0.655
ASN-GLY	0.029	ILE-MET	-0.269
ASN-HIS	-0.766	ILE-PHE	-0.391
ASN-ILE	-0.072	ILE-PRO	0.282
ASN-LEU	-0.399	ILE-SER	-0.121
ASN-LYS	-0.075	ILE-TYR	1.228
ASN-PHE	-0.814	LEU-LYS	0.717
ASN-TRP	0.217	LEU-MET	-0.167

ASN-VAL	0.104	LEU-SER	-0.492
ASP-CYS	0.503	LEU-THR	-0.585
ASP-GLN	-0.247	LEU-TYR	-0.345
ASP-GLU	-0.097	LEU-VAL	0.209
ASP-LEU	-0.611	LYS-MET	0.429
ASP-LYS	-0.429	LYS-VAL	0.591
ASP-MET	0.027	MET-PRO	0.175
ASP-PRO	0.062	MET-SER	-0.668
ASP-TYR	0.244	MET-THR	0.399
ASP-VAL	0.087	MET-TRP	-0.062
CYS-CYS	1.836	PHE-PHE	0.015
CYS-GLN	0.474	PHE-TYR	0.653
CYS-GLU	-0.826	PRO-PRO	0.168
CYS-GLY	-0.441	PRO-SER	-1.359
CYS-ILE	-1.36	PRO-THR	-0.797
CYS-LEU	-1.112	PRO-TYR	0.375
CYS-LYS	-0.095	PRO-VAL	0.157
CYS-MET	-0.103	SER-THR	-0.417
CYS-PHE	-0.695	SER-TYR	-0.252
CYS-VAL	-0.486	SER-VAL	-0.429
GLN-GLU	-0.313	TRP-TYR	0.082
GLN-GLY	0.691	TYR-TYR	-0.202
GLN-HIS	-0.342	TYR-VAL	0.657
GLN-ILE	-0.691	VAL-VAL	0.799
GLN-LEU	-0.958	num atoms	1.309
GLN-LYS	-1.55		

The third category of data that will be examined is the threshold residue atomic interactions. The full model is shown in Figure 5.3. The model is a bit different from the Delaunay interaction model even though the R^2 scores are so similar. The highest score is for Arg-Glu (1.87) which has a negative value in the Delaunay model. Ile-Tyr (1.72), Cys-Cys (1.70), Glu-Val (1.25), and Ala-Ile (1.35) have strong values in both models. Pro-Ser (-1.92) and Cys-Ile(-1.78) have about the same values as in the Delaunay model. Cys-Glu (-1.52), Gln-Phe (-1.30), and Arg-Ser(-1.224) have a much stronger values in the threshold than Delaunay model. If these models were really accurate, they should have similar scores for the same amino acid interactions. Some of the scores are the same, and these are probably the more trustworthy results. Alanine, isoleucine, and cysteine are α-helix promoters (Adams et al. 2002), which may explain the Ile-Tyr and Ala-Ile relationships. Proline and glycine can be helix breakers, which may explain the Pro-Ser relationship. Glutamic Acid and Arginine have a strong ionic bond. It is interesting that Aspartic Acid and Arginine are not present in the model at all, though they have the strongest ionic bond. The way that we measured the amino acid interactions does not ensure that there was a bond, only that one atom from the amino acid was about 8A or less from an atom in the other amino acid.

Table 5.3: Covariates with non-zero values from threshold residue-residue atomic interactions. There are 211 coefficients in the full model, but only 91 are in the final model. The 10 most extreme values are darker blue and the 10 next extreme are lighter blue.

interacting amino acids	β coef value	interacting amino acids	β coef value
ALA-ARG	-0.193	GLN-LYS	-1.091
ALA-ASP	-0.793	GLN-PHE	-1.297
ALA-CYS	-0.725	GLN-SER	-0.071
ALA-GLN	-0.257	GLN-TRP	0.222
ALA-GLU	-0.731	GLN-VAL	-0.067
ALA-ILE	1.345	GLU-GLU	0.016
ALA-LYS	-0.198	GLU-HIS	0.19
ALA-THR	-0.118	GLU-ILE	0.529
ALA-TYR	0.356	GLU-LEU	0.339
ALA-VAL	0.684	GLU-LYS	0.664
ARG-ARG	-0.357	GLU-PRO	1.258
ARG-CYS	-0.163	GLU-THR	-0.176
ARG-GLN	-0.165	GLU-TRP	0.424
ARG-GLU	1.87	GLU-TYR	0.49
ARG-LYS	0.077	GLU-VAL	1.481
ARG-SER	-1.224	GLY-PHE	-0.016
ARG-THR	-1.163	GLY-TRP	0.531
ARG-TRP	-0.429	HIS-ILE	-0.113
ARG-TYR	0.417	HIS-LEU	-0.316
ASN-ASP	0.21	HIS-LYS	-0.653
ASN-GLN	-0.385	HIS-MET	-0.216
ASN-GLU	-0.266	HIS-SER	-0.001
ASN-HIS	-0.997	HIS-TYR	-1.124
ASN-LEU	-0.397	HIS-VAL	-0.779
ASN-LYS	-0.434	ILE-ILE	0.423
ASP-CYS	0.854	ILE-TYR	1.722
ASP-GLN	-0.047	LEU-LYS	0.904
ASP-ILE	-0.348	LEU-MET	-0.245
ASP-LEU	-0.327	LEU-SER	-0.316
ASP-LYS	-0.315	LEU-THR	-0.719
ASP-PHE	0.125	LEU-VAL	0.031
ASP-TRP	0.592	LYS-TYR	0.085
ASP-VAL	-0.386	MET-SER	-0.937
CYS-CYS	1.696	MET-THR	0.615
CYS-GLN	0.197	PHE-TYR	1.025
CYS-GLU	-1.524	PRO-SER	-1.917
CYS-ILE	-1.798	PRO-THR	-0.048
CYS-LEU	-0.916	PRO-TYR	1.267
CYS-LYS	-0.088	SER-THR	-0.586
CYS-PHE	-0.64	SER-TYR	-0.579

CYS-VAL	-0.205	THR-TYR	-0.191
GLN-GLU	-0.572	THR-VAL	0.266
GLN-GLY	0.627	TRP-TRP	0.068
GLN-HIS	-0.628	TYR-VAL	0.824
GLN-ILE	-0.29	VAL-VAL	1.211
GLN-LEU	-0.797		

5.2 Group Lasso

Group lasso requires that the groups of covariates be predefined before running. In order to find which groups of covariates were correlated, hierarchical clustering with ward linkage was done on the covariate correlation matrix. Covariates with similar correlations were clustered together. Different numbers of clusters (between 2 and 25) were scored based on their silhouette value, which is calculated by using the mean intra-cluster distance (i) and the mean nearest-cluster distance (n). The silhouette value balances the cohesion of an object (how similar it is to the cluster it is in) compared to the separation (how similar it is to other clusters). The mean silhouette value for all samples is taken $(i-n)/max(i,n)$. Higher values are better and indicate clusters with more separation. The plot of the silhouette scores in Figure 5.2 shows the optimal number of clusters for each dataset. Basing the covariate clustering on their correlation scores meant that correlated clusters of covariates were forced to have similar coefficients.

Those clusters that are shown in Figures 5.3, 5.4, 5.5, 5.6, 5.7, 5.8, and 5.9 were used as the group labels for group lasso. Then these scores were compared to the scores where all clusters were forced to be together. Then a manual grid search of 20*15*2*3 different parameters was performed. These are the parameter options searched: group reg [20 evenly spaced numbers between 0 and 1.5], l1 reg [15 evenly spaced numbers between 0 and 1], frobenius [True, False], and scale reg [group size, none, inverse group size]. The validation $RMSE$ was used to find the best possible combination of hyperparameters.

The best hyper-parameters for each category of data were:

1. All
   Group Reg: 0.2, L1 Reg: 0, frobenius: True, scale reg: group size, Number variables: 900, Number of chosen variables: 900
   Train R^2:0.32 Test R^2: 0.28
2. Tsai Atomic Interaction
   Group Reg: 1, L1 Reg: 0, frobenius: True, scale reg: inverse group size, Number variables: 92, Number of chosen variables: 88
   Train R^2:0.21 Test R^2: 0.20
3. Popelier Atomic Interaction
   Group Reg: 0.4, L1 Reg: 0.07, frobenius: True, scale reg: inverse group size, Number variables: 277, Number of chosen variables: 205
   Train R^2:0.37 Test R^2: 0.16
4. Residue Delaunay Atomic Interaction
   Group Reg: 0, L1 Reg: 0.57, frobenius: False, scale reg: group size, Number

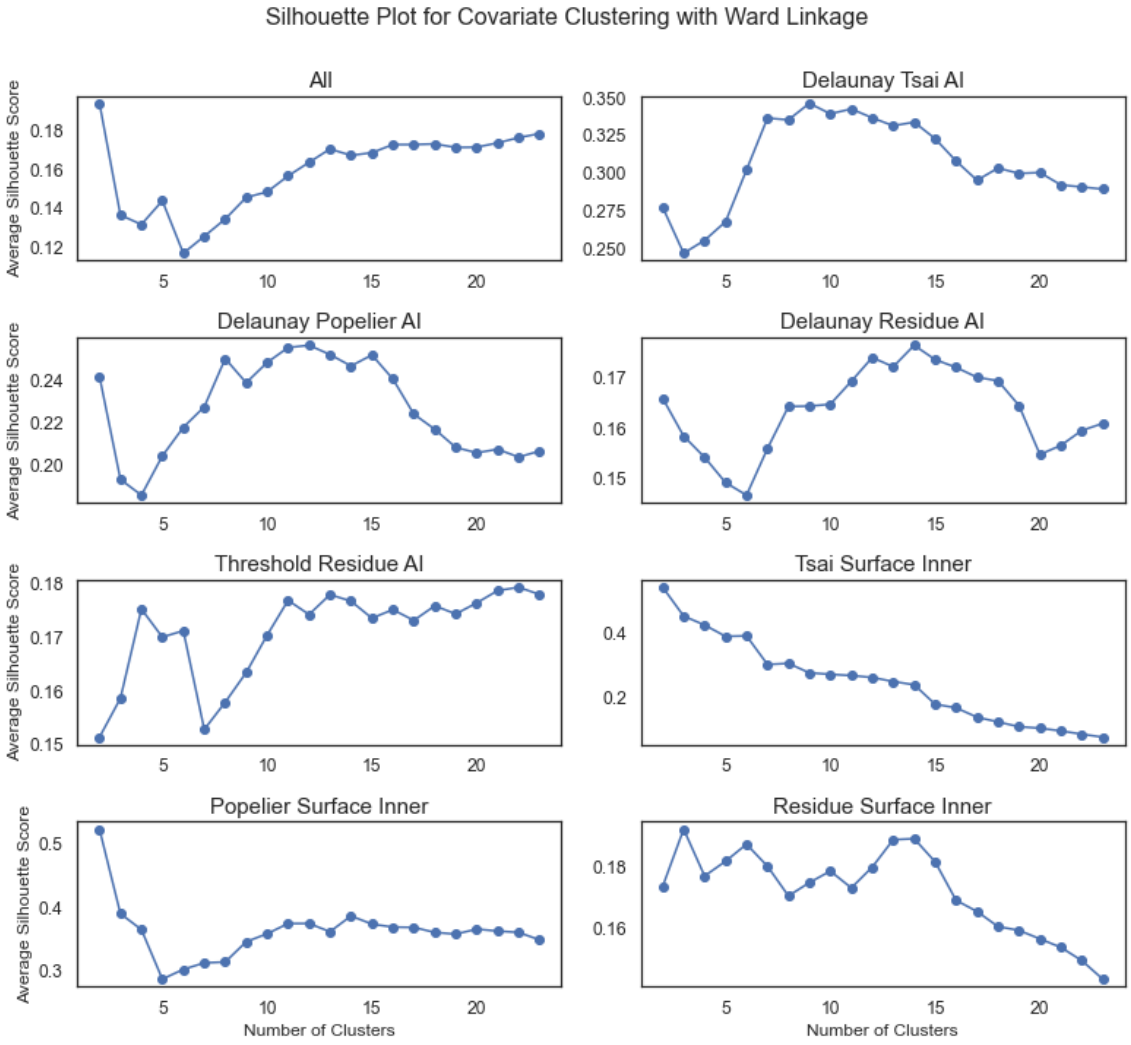

Figure 5.2: The silhouette scores for different numbers of clusters. Higher scores are best. The optimal number of clusters for the whole dataset was 2, for Delaunay Tsai AI was 9, for Delaunay Popelier AI was 12, 14 for Delaunay Residue AI, 22 for Threshold Residue AI, 2 for Tsai Surface Inner, 2 for Popelier Surface Inner, and for Residue Surface Inner was 3.

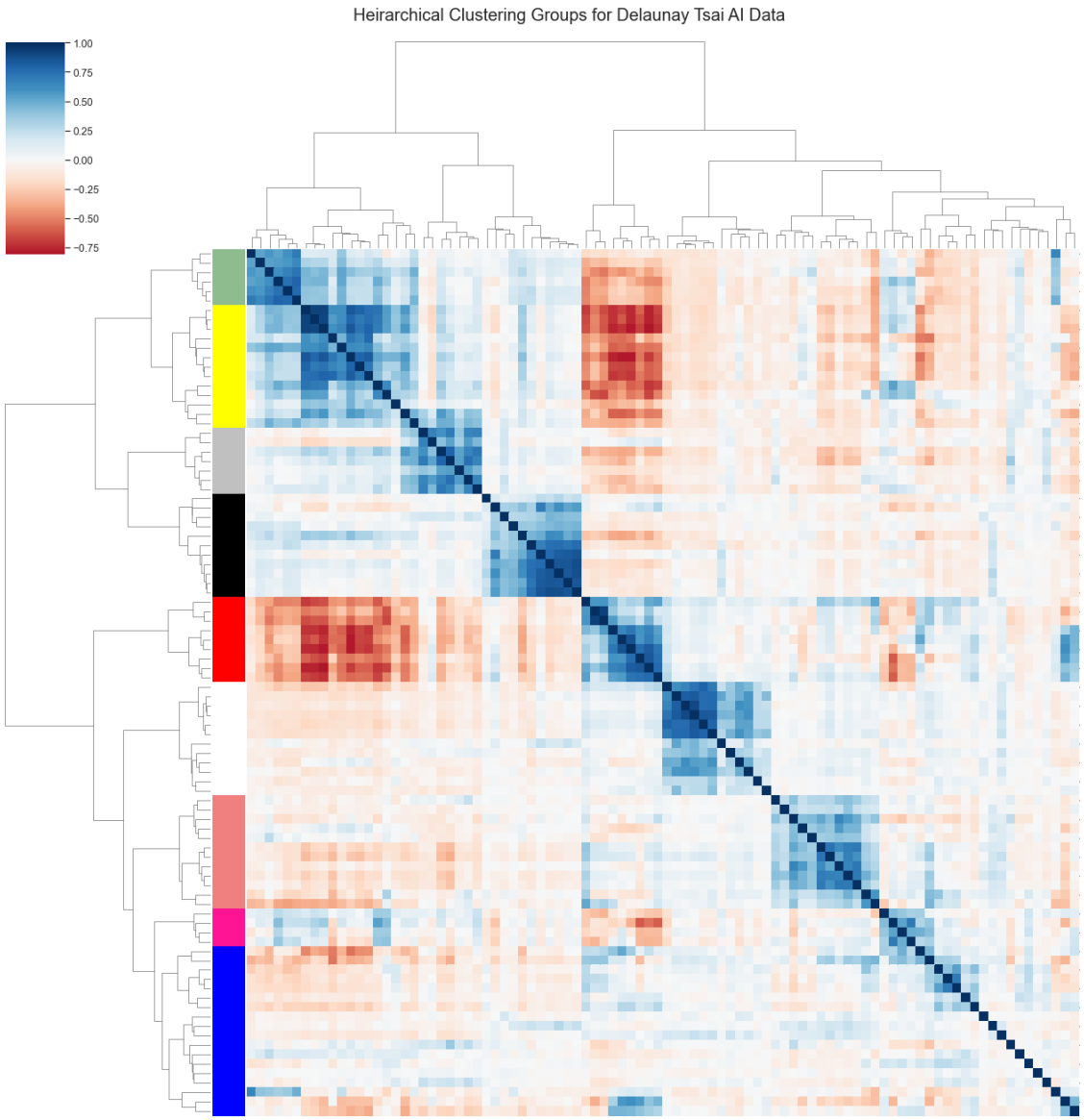

Figure 5.3: The results of hierarchical clustering on the Tsai atomic group interaction correlation matrix. Dividing the data into nine clusters had the highest silhouette score. The green cluster are all bonded to O2H1: O2H1, O1H0, N3H1, C4H2, C4H1, and C3H0. The yellow cluster is N3H2-O2H1, C3H0-C3H0, and every C3H1 interaction except S2H0. The gray cluster is bonded to N3H2: C3H0, C4H1, C4H2, C4H3, N3H1, N3H2, and O1H0. The black color is every S2H0 group except N4H3. The red cluster is N3H1-O1H0, C3H0-N3H1, C4H1-O1H0, C4H1-C4H2, and C4H3 bonded to O1H0, N3H1, C4H3, C4H2, and C4H1. The white cluster is every S2H1 interaction except N4H3. The peachy pink cluster is C4H2-O1H0, C4H2-C4H2, and every N4H3 cluster except S2H1, and S2H0. The hot pink cluster is O1H0-O1H0, N3H1-N3H1, C3H0-O1H0 and C3H0-C4H1. The blue cluster is num atoms, N4H3-S2H1, N4H3-S2H0, C4H3-O2H1, C4H2-N3H1, C3H0-C4H3, C4H1-N3H1, C4H1-C4H1, and every N3H0 group except S2H0. There is a lot of correlation of same Tsai group with itself.

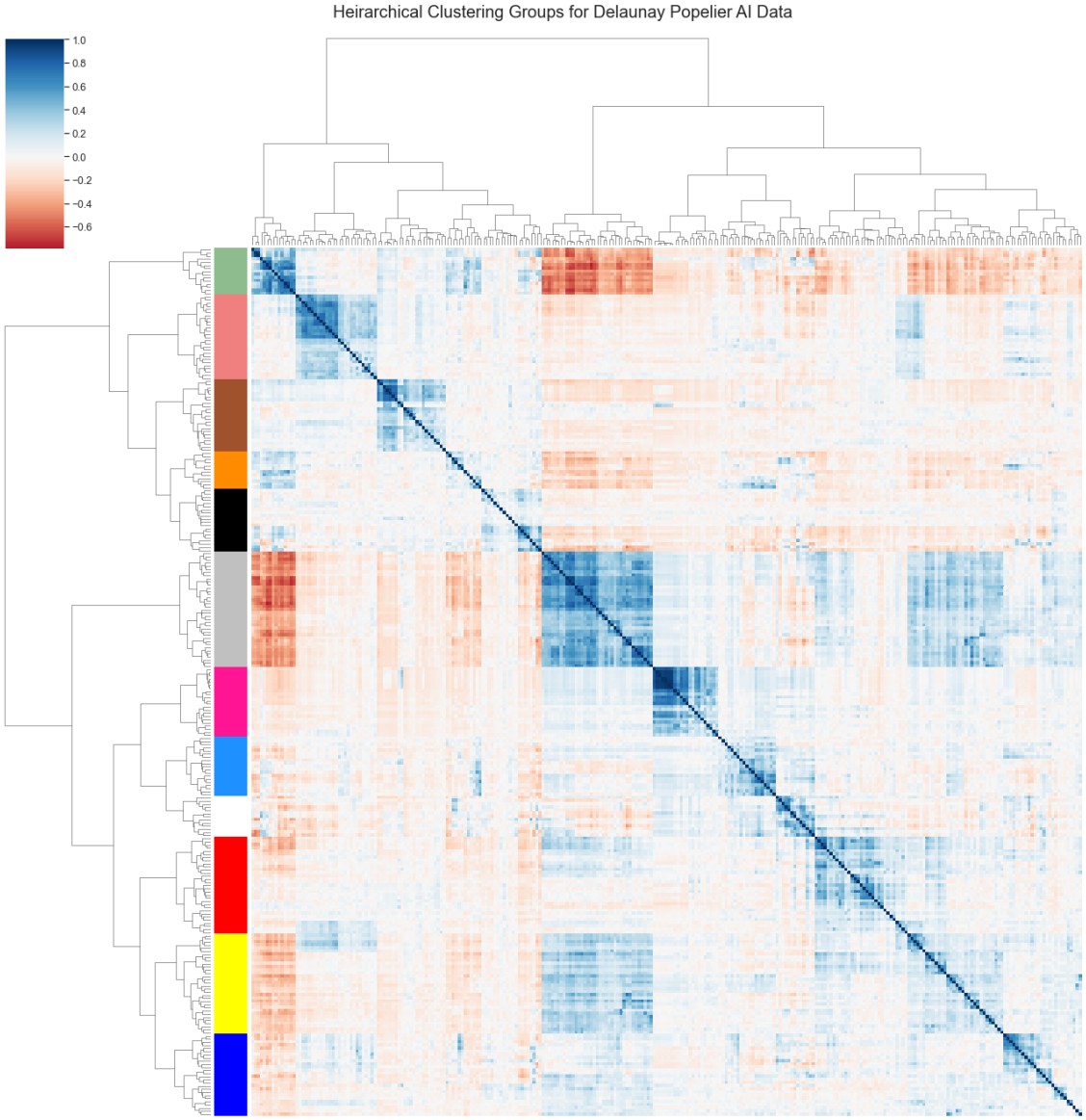

Figure 5.4: The results of hierarchical clustering on the Popelier atomic group interaction correlation matrix. Dividing the data into twelve clusters had the highest silhouette score. Green is C2 interacting with 7 groups, N2-O5, C8-O5, C4-O5, C4-N2, C4-C8, C3-C8, C3-C4, and C19-N2. The salmon cluster is S6 interacting with 12 groups and C21 interacting with 14 groups. The brown cluster is every S3 group. The orange cluster is interactions for C4 (7 groups) and C2 (5 groups). Black is interactions for C7 (16 groups) and C3 (4 groups). Gray is C12 (13 groups), C14 (8 groups), O2 (8 groups), and C18 (8 groups). The hot pink cluster is every interaction involving SS. The lighter blue cluster is C9-O5, O3 (10 groups) and C9 (8 groups). The white cluster is O5-O5, N2-N2, C8-N2, C8-C8, O4 with 5 groups, and C19 (4 groups). The red cluster is a mix of C15 and C17 interactions. They interact with the same 15 things and each other. The yellow cluster is O2 (8 groups), C18 (8 groups), C14 (9 groups), and C12 (7 groups). The dark blue cluster is num atoms, O2-S6, C9-C14, C3-O4, C18-S6, C18-C18, C14-O3, C7 (4 groups), C10 (8 groups) and N1 (7 groups).

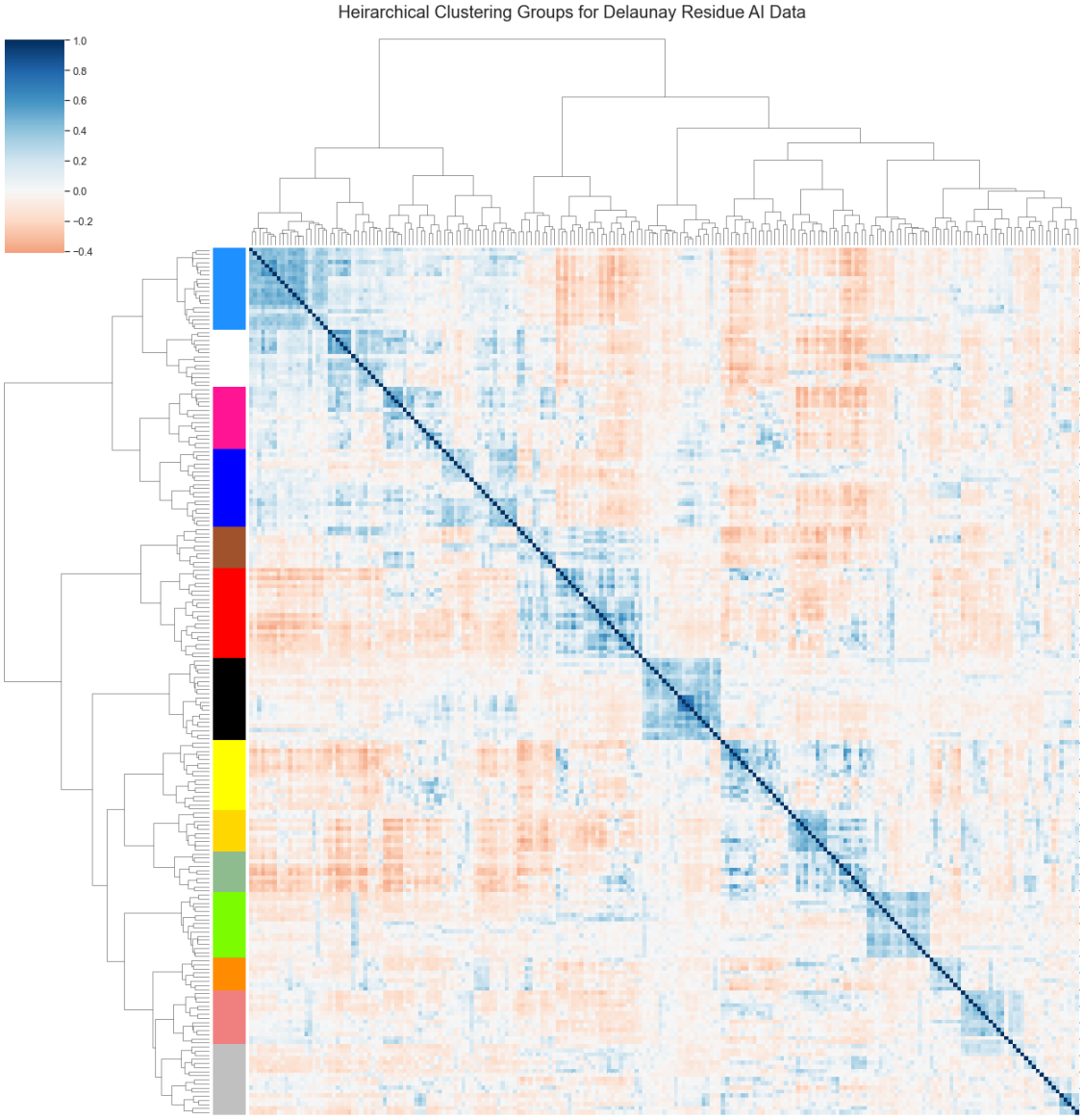

Figure 5.5: The results of hierarchical clustering on the residue Delaunay atomic group interaction data. The optimal number of clusters is 19 according to the silhouette scores. There are too many residue groups to list them all.

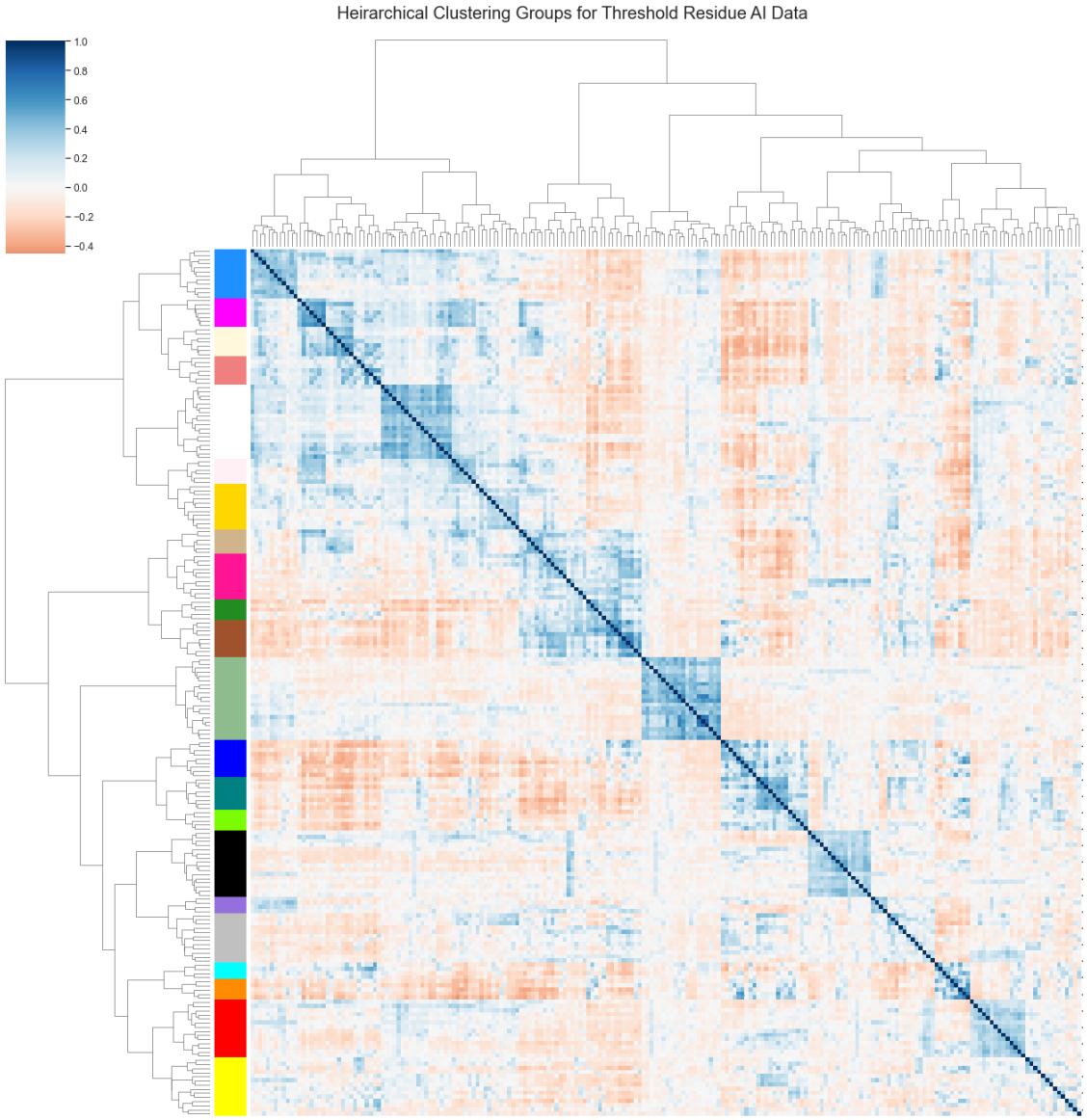

Figure 5.6: The results of hierarchical clustering on the threshold residue atomic group interaction data. The optimal number of clusters is 19 according to the silhouette scores. There are too many residue groups to list them all.

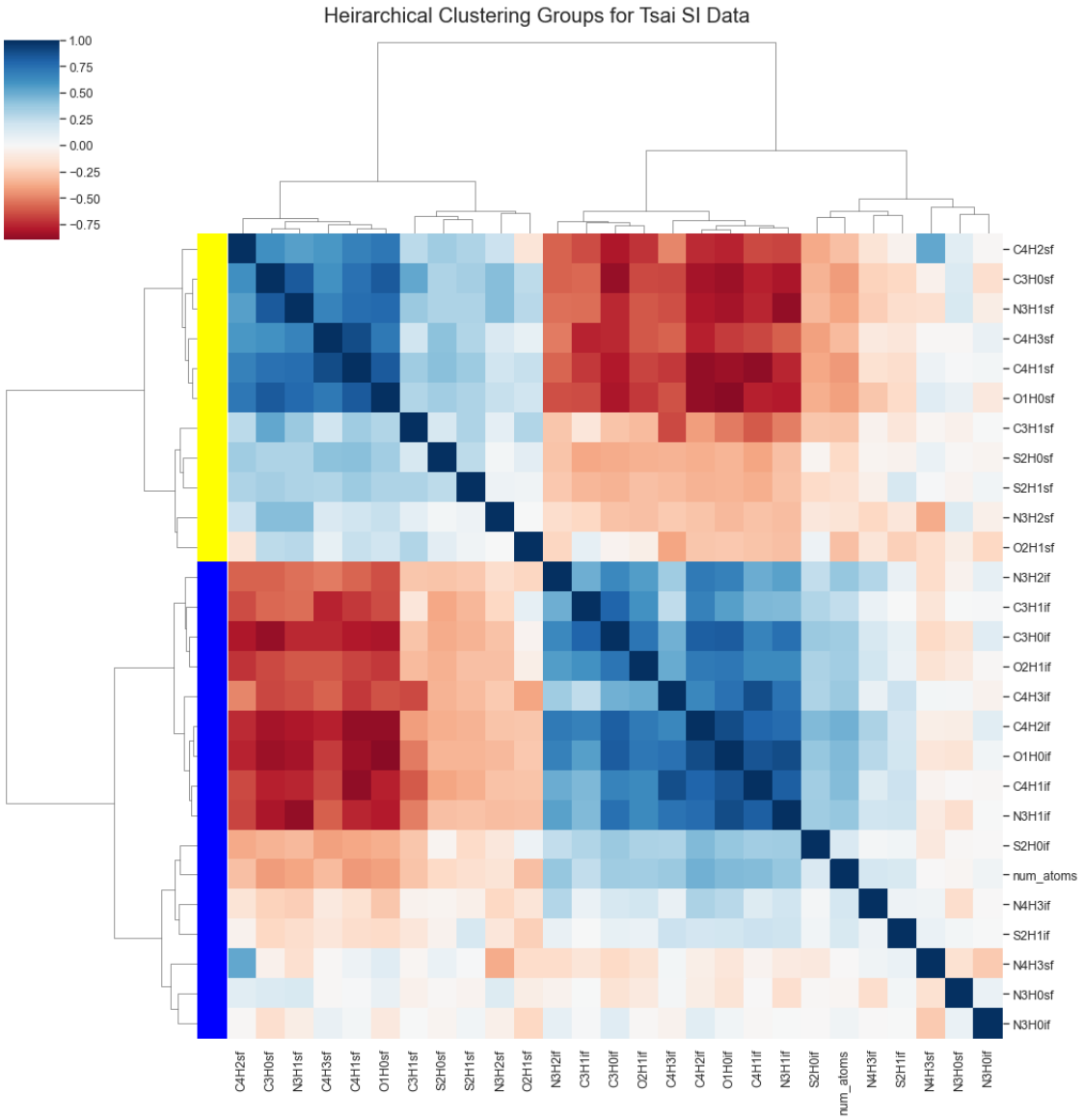

Figure 5.7: The results of hierarchical clustering on the Tsai surface / inner frequency correlations. There are two clusters, one containing the surface atoms and the other containing the inner atoms. The number of atoms is correlated with the inner atoms because the number of atoms on inside the protein increases the most as protein grows larger.

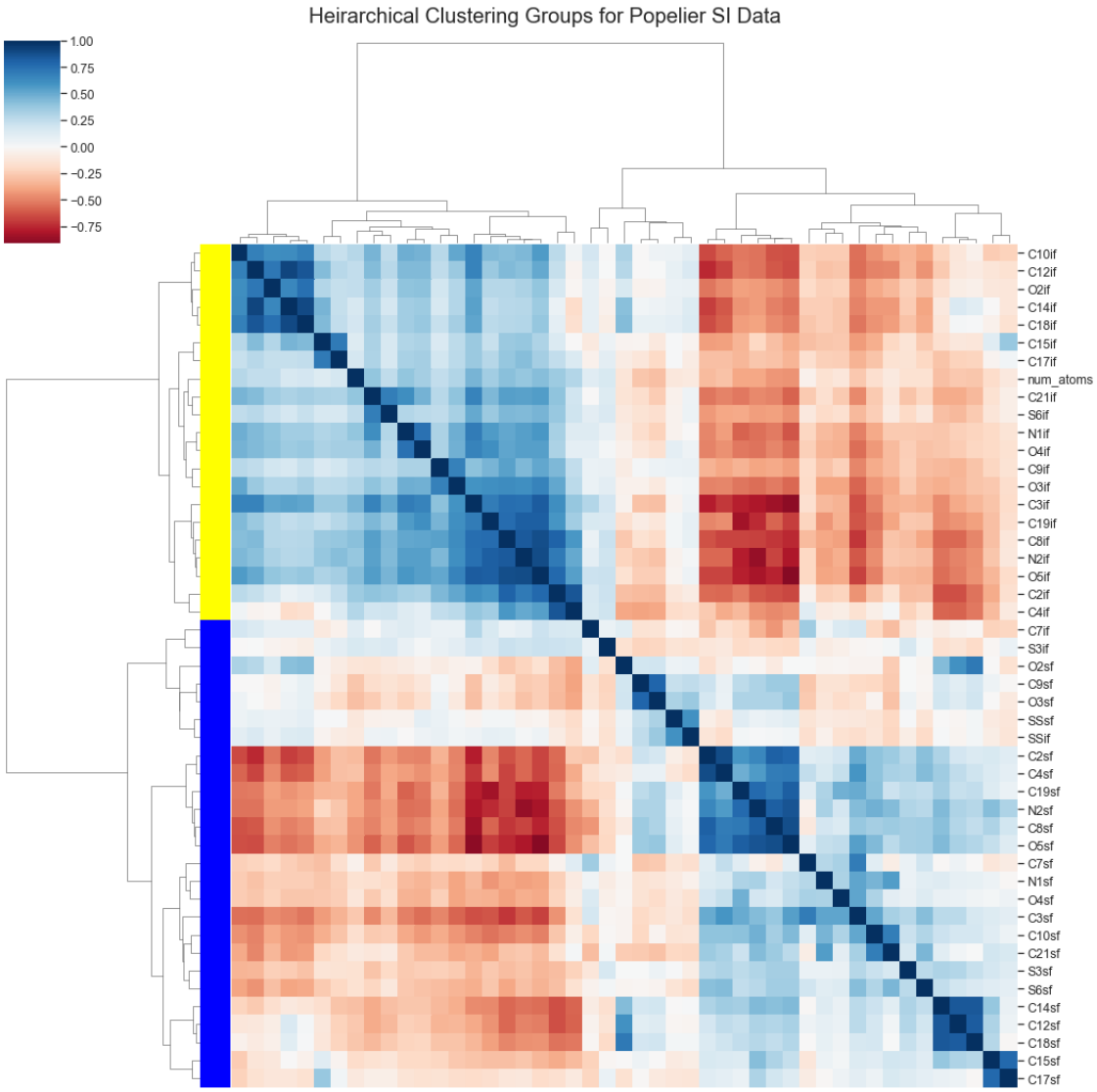

Figure 5.8: The results of hierarchical clustering on the Popelier surface / inner frequency correlations. The optimal number of clusters is two according to the silhouette scores. The yellow cluster contains inner atomic groups. The blue cluster contains a few inner atomic groups and all of the surface atomic groups.

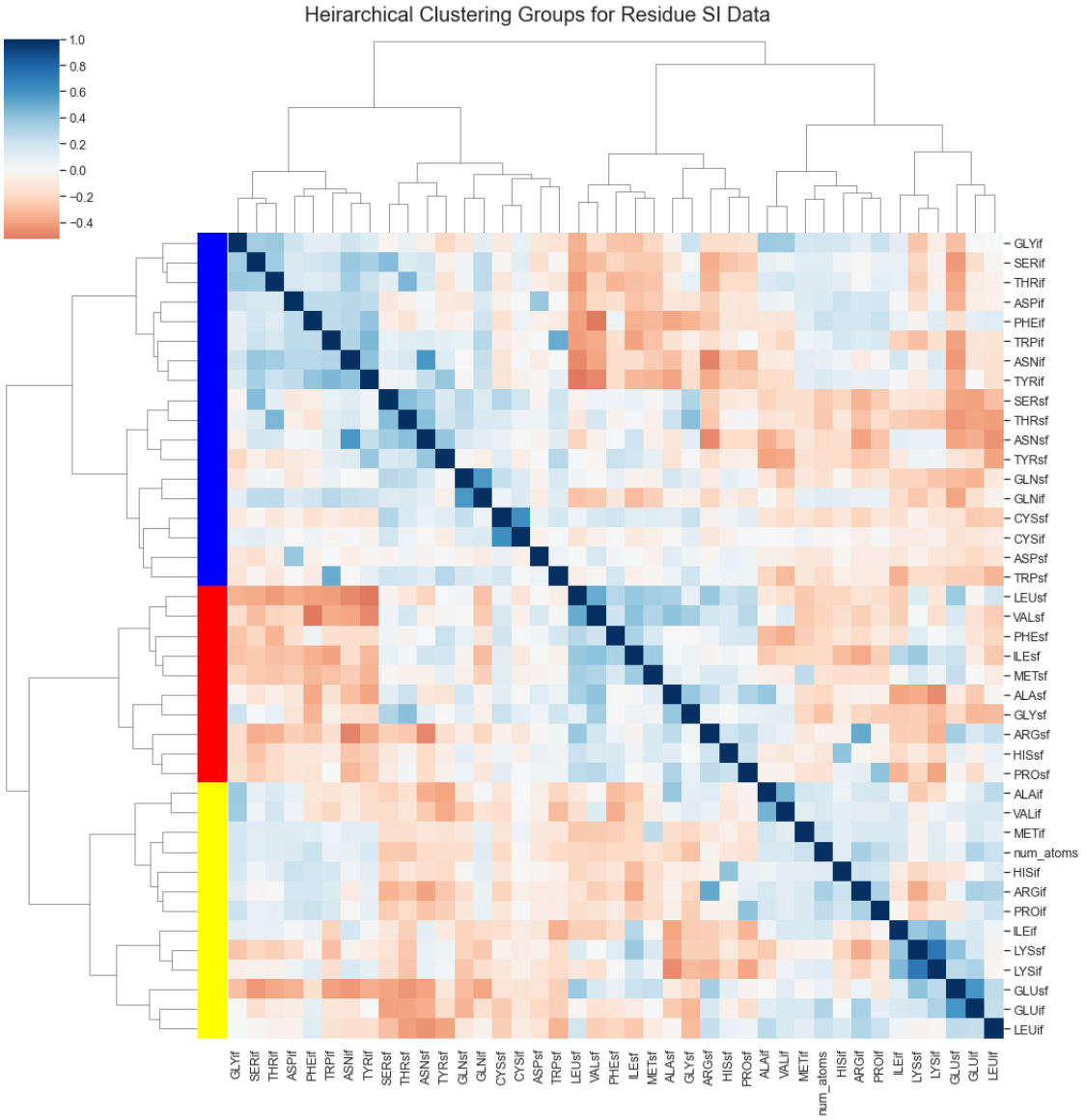

Figure 5.9: The results of hierarchical clustering on the residue surface / inner frequency correlations. Dividing the data into three clusters gave the highest silhouette score.

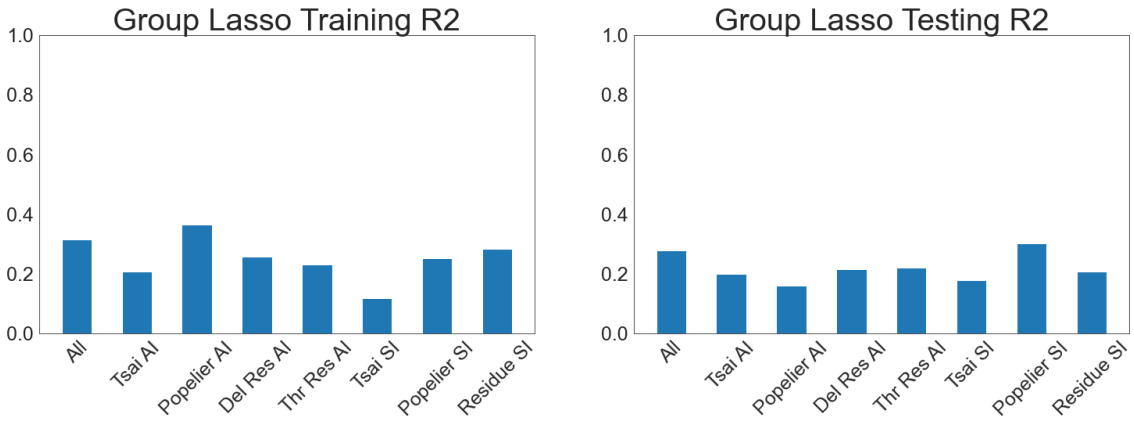

Figure 5.10: The testing and training R^2 values for the group lasso models. They perform the worst out of all the models tested.

variables: 211, Number of chosen variables: 77
Train R^2:0.26 Test R^2: 0.21

5. Residue Threshold Atomic Interaction
   Group Reg: 0, L1 Reg: 0.36, frobenius: False, scale reg: none, Number variables: 210, Number of chosen variables: 118
   Train R^2:0.30 Test R^2: 0.22

6. Tsai Surface Inner Frequency
   Group Reg: 1, L1 Reg: 0, frobenius: True, scale reg: none, Number variables: 27, Number of chosen variables: 27
   Train R^2:0.12 Test R^2: 0.18

7. Popelier Surface Inner Frequency
   Group Reg: 0.75, L1 Reg: 0, frobenius: True, scale reg: none, Number variables: 47, Number of chosen variables: 47
   Train R^2:0.25 Test R^2: 0.30

8. Residue Surface Inner Frequency
   Group Reg: 2, L1 Reg: 0.14, frobenius: True, scale reg: inverse group size, Number variables: 41, Number of chosen variables: 30
   Train R^2:0.28 Test R^2: 0.21

The L1 regularizer is the α variable from the equation that was previously discussed. When $\alpha = 0$, there is a group lasso fit and $\alpha = 1$, there is a lasso fit. Most of the L1 regularizers have *alpha* values closer to 0, so group lasso fits have the highest scores. λ measures how much the data is regularized, and the optimal values range from 0 (no regularization) to 2 (a high level of regularization) for these categories of data.

The R^2 values are shown in Figure 5.10. The training models generalize well– the testing scores are about the same as the training scores. Unfortunately, the group lasso models have the lowest testing R^2 scores out of all the models, so these models will not be examined.

5.3 Random Forest

A grid search with 5-fold cross validation was performed to find the best hyper-parameters for each dataset from the following choices: max depth: [5, 10], max features: [sqrt, log2], number estimators: [50, 100, 150], min impurity decrease: [0.3, 0.4, 0.5], ccp alpha: [0.1, 0.2, 0.3]

The best parameters for each category were:

1. All
   ccp alpha: 0.1, max depth: 10, max features: sqrt, min impurity decrease: 0.4, num estimators: 150
   R^2 train: 0.82, R^2 test 0.37
2. Tsai Atomic Interactions ccp alpha: 0.3, max depth: 10, max features: log2, min impurity decrease: 0.5, num estimators: 50
   R^2 train: 0.70, R^2 test 0.27
3. Popelier AI ccp alpha: 0.2, max depth: 10, max features: sqrt, min impurity decrease: 0.5, num estimators: 50
   R^2 train: 0.75, R^2 test 0.20
4. Residue Del AI ccp alpha: 0.2, max depth: 10, max features: sqrt, min impurity decrease: 0.4, num estimators: 150
   R^2 train: 0.80, R^2 test 0.31
5. Residue Threshold AI ccp alpha: 0.1, max depth: 10, max features: sqrt, min impurity decrease: 0.3, num estimators: 150
   R^2 train: 0.81, R^2 test 0.33
6. Tsai Surface Inner Freq ccp alpha: 0.1, max depth: 10, max features: sqrt, min impurity decrease: 0.3, num estimators: 50
   R^2 train: 0.75, R^2 test 0.20
7. Popelier Surface Inner Freq ccp alpha: 0.1, max depth: 10, max features: sqrt, min impurity decrease: 0.5, num estimators: 150
   R^2 train: 0.75, R^2 test 0.24
8. Residue Surface Inner Freq ccp alpha: 0.2, max depth: 10, max features: sqrt, min impurity decrease: 0.5, num estimators: 100
   R^2 train: 0.77, R^2 test 0.38

The test and training R^2 values in Figure 5.11 show that all of the random forest models have over-fit the data. The training scores are lower than the testing scores, but they are still some of our highest scores. The datasets with the residues score the most highly. Our hypothesis is that this is due to the smaller amounts of correlation in these datasets.

In order to interpret the Random Forest models, the features that were used by the testing data were traced with the treeinterpreter program Saabas, Ando 2015. Each protein in the test set gives a weight for how much each covariate was used. The weights for the test set were then averaged. This method for finding weights is not as good as the weights for linear regression, but it is the best possible explanation for random forest regression. The weights for the best model, the residue surface inner frequencies, are included in Table 5.4. These weights are more like a rough approximation for the model than the exact model that is used. Glutamic acid on either the surface or the interior was found to be impactful. It is a polar amino acid

GLU if	0.3	TRP sf	-0.01
THR sf	0.24	ARG sf	-0.01
GLU sf	0.21	ALA if	-0.01
CYS if	0.08	LYS if	-0.02
ILE if	0.07	MET if	-0.02
LEU sf	0.04	MET sf	-0.02
ASP if	0.03	ASP sf	-0.03
PRO sf	0.03	SER if	-0.03
HIS sf	0.03	PHE sf	-0.04
PHE if	0.02	GLY if	-0.04
VAL if	0.02	CYS sf	-0.04
TRP if	0.02	TYR sf	-0.04
ALA sf	0.01	TYR if	-0.05
HIS if	0.0	SER sf	-0.05
THR if	0.0	GLN if	-0.05
ILE sf	-0.0	LYS sf	-0.05
LEU if	-0.01	num atoms	-0.08
ASN if	-0.01	GLN sf	-0.08
VAL sf	-0.01	PRO if	-0.08
ARG if	-0.01	ASN sf	-0.09
GLY sf	-0.01		

Table 5.4: Random forest weights for residue surface inner frequencies. Sf stands for surface frequency and in stands for inner frequency. The weights are an average of the weights from the test data. The most impactful variables are the frequency of glutamic acid on the interior (GLU if), the frequency of threonine on the surface (THR sf), and the frequency of glutamic acid on the surface (GLU sf).

that can be involved in ionic bonds. Threonine is another polar amino acid and was influential when on the protein surface.

5.4 SVR

A grid search using 5-fold cross validation was performed to find the best hyper-parameters from the following possibilities: epsilon: [1, 5, 10, 15], kernel: [linear, rbf, poly, sigmoid], C: [1, 10, 20]. The results are found in Figure 5.12. All of the datasets experienced problems with over-training, with the training set having R^2 scores twice as high as the testing set. The best performing category of data was the threshold residue atomic interactions (Res Thr AI), which was the method used in an earlier thesis by Ulfenborg 2020. It is interesting that this method scored higher than the Delaunay residue atomic interaction (Del Res AI), which measures the same interactions, only more carefully. The residue surface interior frequencies (Res SI) also scored really well.

The best hyper-parameters and scores for each dataset were:

1. All
   C: 20, epsilon: 1, kernel: rbf

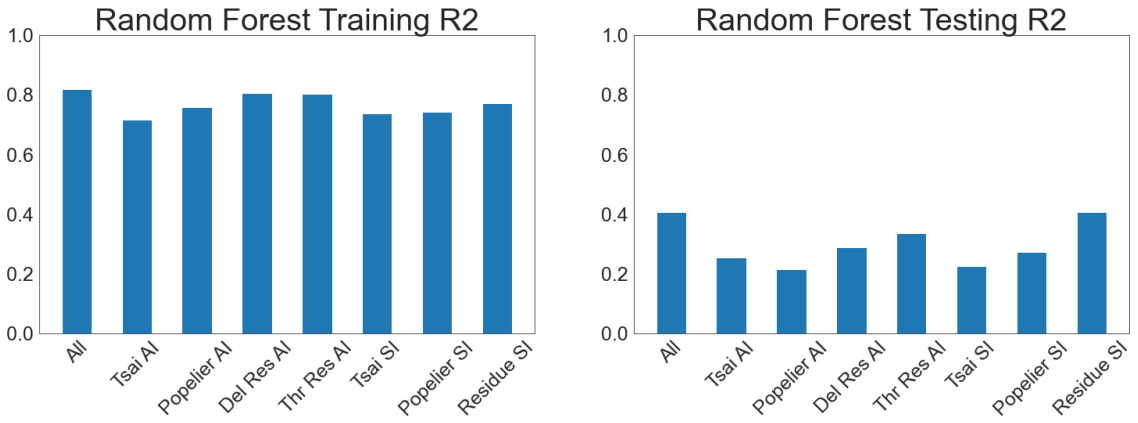

Figure 5.11: The R^2 values for Random Forest Regression for different categories of data. The Residue SI is one of the best performing models in this work and is examined in detail.

R^2 train: 0.83, R^2 test 0.41

2. Tsai Delaunay AI
   C: 10, epsilon: 10, kernel: rbf
   R^2 train: 0.46, R^2 test 0.24
3. Popelier Delaunay AI
   C:10, epsilon: 10, kernel: rbf
   R^2 train: 0.51, R^2 test: 0.23
4. Residue Delaunay AI
   C: 10, epsilon: 5, kernel: rbf
   R^2 train: 0.62, R^2 test: 0.34
5. Residue Threshold AI
   C:20, epsilon:1, kernel: rbf
   R^2 train: 0.78, R^2 test: 0.40
6. Tsai Surface Inner Freq
   C: 10, epsilon: 5, kernel: rbf
   R^2 train: 0.44, R^2 train: 0.27
7. Popelier Surface Inner Freq
   C: 10, epsilon: 5, kernel: rbf
   R^2 train: 0.53, R^2 test: 0.34
8. Residue Surface Inner Freq
   C: 10, epsilon: 1, kernel: rbf
   R^2 train: 0.64, R^2 test 0.41

All of the categories of data appear to be over-fitting; the training R^2 values are twice as high as the testing R^2. These are the models with the highest training R^2 scores. The RBF kernel was selected as the best kernel for all of our models. Unfortunately, Support Vector Regression with an RBF kernel is not human-interpretable and therefore our goal of understanding which covariates are the most important for predicting the optimal temperature is impossible for this model. The QQ-plots of the residuals for all of the SVR models were examined and found to be normally distributed.

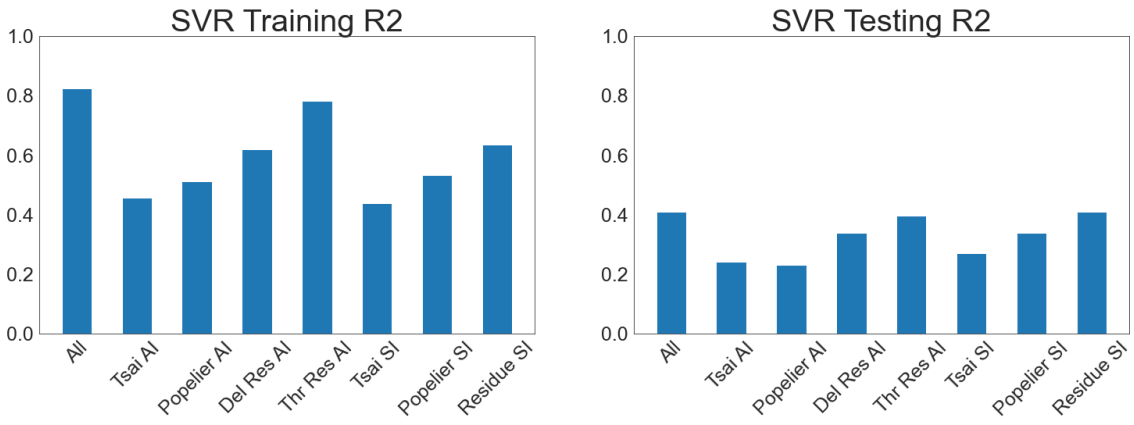

Figure 5.12: The R^2 values for SVR for different categories of data. The best models involve the residues: the residue surface inner frequencies, threshold residue atomic interactions, and Delaunay residue atomic interactions. The models are over-trained, but are still perform better than models from other statistical methods.

6
Discussion

The aim of this thesis was to learn more about the effect of protein structure on thermostability. Our hypothesis was that breaking the protein structures down into fine detail and classifying each atom by its atomic group would give very good results. When atoms are classified into their atomic groups, we thought it would be easier to see ionic or covalent bonds that should have a positive effect on the protein structure, and therefore also on the optimal temperature. We have not seen the results that we were looking for; the predictive power was lower than we thought it would be. There are two possibilities that could be responsible for the lower scores: either the frequency of these interactions does not explain that much of the temperature variation, or the amount of collinearity impedes proper analysis.

We examined the collinearity of the variables because it was potentially problematic. However, it does not appear to be the main hindrance to the models' performance. Table 6.1 shows the amount of correlation in each data collection and what percent of the covariates are highly correlated. The Tsai surface inner frequencies (Tsai SI freq) have the highest amount of correlation. If the amount of correlation were the only factor in the scores, the Tsai SI freq should be the worst-performing group. Instead, the Tsai atomic interactions (Tsai AI) are often the worst-performing group, and its level of correlation is five times lower than the Tsai SI freq. If correlation were the main impediment to the models' performance, the Tsai SI freq should have the lowest score for every statistical method. In the SVR analysis, the groups with the least amount of correlation do perform better, though it doesn't match the magnitude of the degree of correlation. This trend does not hold for the other statistical methods. Therefore, the amount of collinearity is not the most significant factor influencing the models' scores.

Having ruled out collinearity as being the major contributor to the low scores, we are left with the possibility that the frequency of atoms on the surface/interior and the frequency of the atomic interactions do not adequately explain the response variable (topt) because it is not possible for them to do so. The atomic interactions can conceivably measure ionic and covalent bonds (namely disulfide bridges), but it is difficult to see how other types of bonds would be captured by the interaction frequencies. Hydrogen bonds could perhaps be captured by nitrogen and oxygen interactions. But other features that are known to affect protein stability are protein packing (the density of atoms within the protein), hydrogen bonds with the surrounding fluid, and hydropathy (Yang et al. 2019b). These features are not able to be measured by our atomic interaction models.

Besides not including some factors known to be important for protein stability, we are also miscounting the surface interactions by examining only the within-

Data Collection	Num Covariate Interactions	Num Corr >0.5	Num Corr >0.7	Percent >0.5	Percent >0.7
Tsai AI	4186	236	85	0.0564	0.0203
Popelier AI	38226	856	164	0.0224	0.0043
Del Res AI	22155	87	4	0.0039	0.0002
Thr Res AI	22155	147	5	0.0066	0.0002
Tsai SI	351	91	42	0.2593	0.1197
Pop SI	1081	173	47	0.1600	0.0435
Residue SI	861	9	1	0.0105	0.0012

Table 6.1: This table captures the number of correlated covariates in each dataset. For each collection of data, the number of covariate interactions is shown, along with the number of covariate interactions with a Pearson correlation score greater than 0.5 and the number of covariate interactions with a Pearson correlation score greater than 0.7. The amount of covariate correlation is also shown as a percentage of the total number of covariate interactions. The Tsai surface inner frequencies have the highest amount of correlation, with 26% of the covariates having a high amount of correlation and 12% of the covariates having an extremely high amount of correlation.

protein interactions. This is a problem because the atoms are forming bonds that we are not counting. When the surface atoms' neighbors are reduced to remove interactions with non-protein atoms, the number of interactions that each surface atom has is reduced. Therefore, they will have lower counts of neighbors than atoms that are within the protein. The surface atoms' frequencies are therefore artificially lower than they should be, and this could interfere with the models' performance. In addition to having artificially low counts, the total number of stabilizing hydrogen bonds is not being accurately calculated. With our nearest neighbor interaction method, no atom can have a nearest neighbor that is not part of the protein. Therefore, hydrogen bonds with the surrounding fluid are not measured. Having potentially polar atoms (namely nitrogen and oxygen) on the surface has been shown to increase stability (Vogt, Woell, and Argos 1997). The surface inner models could possibly measure hydrogen bonds with the surrounding fluid by counting the atoms on the surface, and this could be the reason they perform better.

One defining feature of this study was how many different ways the same data was examined. We had a hypothesis that breaking down the proteins finely would give the best results. We tested this in several different ways. The Popelier, Tsai, and residue groups all had different levels of granularity and comparing their results tests our assumption about granularity. We hypothesized that the Popelier atomic groups would be better than the Tsai atomic groups because they broke the data down more finely. The results are inconclusive as to which category is better. The Tsai atomic group interactions scored slightly higher than the Popelier atomic group interactions for all methods. However, the Popelier surface interior scores were all higher than the Tsai surface interior scores. Neither category is decisively better.

Another way we tested the granularity hypothesis was by comparing the Delaunay and threshold residue atomic interactions. It is surprising that the threshold

residue atomic interactions (Thr Res AI), which only examine the position of one atom in the protein, gives very similar results to breaking down the results by their Delaunay residue atomic interactions (Del Res AI). There are several differences in how the datasets were created. First, for the threshold interactions, all of the atoms were used instead of only the carbon beta. Second, instead of using a threshold of 8Å, we painstakingly found the closest Delaunay neighbors. Third, the atoms were classified by their atomic group instead of by which amino acid residue they are in. If there is useful information to be found from these interactions, then being more careful about defining which things are interacting should be better. However, in the group lasso and elastic net results, the Delaunay and threshold residue interactions score almost exactly the same. In the SVR and random forest models, the threshold interactions score higher. Being more careful with the defining of which atoms were interacting did not increase the predictive power.

The ultimate test of granularity comes from finding which models performed the best. The highest scoring models were those with the lowest levels of granularity. Our best models were the SVR residue surface inner frequencies (Res SI $R^2 = 0.409$), random forest residue surface inner frequencies (Res SI $R^2 = 0.406$), and SVR threshold residue atomic interactions (Thr Res AI $R^2 = 0.395$). All of these models use the residues instead of atomic groups. None of the models give insight into atomic neighbor interactions. Surprisingly, categorizing the data finely was not effective.

One possible limitation of our model is related to the dataset and not to the way the data was analyzed. Previous studies have had difficulties with dataset limitations leading to poor performance (Yang et al. 2019a). That dataset was quite limited, only studying variations of 9 proteins. Our dataset contained many more proteins, with 1122 training proteins, 126 test proteins, and 131 validation proteins. The proteins in the analysis are not extremely similar, because the similar proteins were removed. Though we have more than 9 proteins, the proteins we used do not cover the full range of possible protein configurations. Only 295 protein structures were experimentally verified, and the rest are based on verified proteins. With the introduction of AlphaFold, there should soon be an even larger number of protein structures available to study. AlphaFold is a recently released computer program that can predict protein structures very accurately and quickly (Jumper et al. 2021). It does not depend so heavily on previous structures. Perhaps adding more AlphaFold structures to the training data could increase the utility of our models.

There are some overall conclusions to be drawn from this study. First, breaking down the protein into the most detailed categories did not give better results than broader categories. Second, examining a protein's interactions with itself did not give the best results. The surrounding environment of the protein is also important and should also be considered when investigating protein stability. Doing so could help us to learn more about protein structure, thermostability, and how to build better enzymes.

6.1 Future Directions

Our best scores are similar to the best scores from Ulfenborg 2020, the original study that inspired this work. It seems that there are no higher scores to be gained from these data categories and methods. It is possible that a different statistical method that can better handle collinearity could produce higher scores. Collinearity-weighed regression, octagonal shrinkage clustering and regression (OSCAR), latent root regression, principal component regression, and partial least squares are all methods that were made to analyze collinear data (Dormann et al. 2013). They are uncommon methods, but it is possible they would work better. Other features that are known to influence protein stability, such as protein packing, could also be included.

When the structures were broken down by their atomic types, perhaps some information was lost. Other studies show that protein stability is due to a combination of many factors. A method that could examine many factors at the same time would probably work better because it could mirror the complexity inherent in the dataset. Doing these things could improve the predictions in a future analysis.

Bibliography

Adams, H et al. (2002). "The presence of a helix breaker in the hydrophobic core of signal sequences of secretory proteins prevents recognition by the signal-recognition particle in Escherichia coli". In: *Eur J Biochem* 269.22, pp. 5564–71. DOI: 10.1046/j.1432-1033.2002.03262.x.

Awad, Mariette and Rahul Khanna (2015). *Efficient Learning Machines*. ApressOpen.

Barber, C. Bradford, David P. Dobkin, and Hannu Huhdanpaa (1996). "The Quickhull algorithm for convex hulls". In: *ACM TRANSACTIONS ON MATHEMATICAL SOFTWARE* 22.4, pp. 469–483.

Berman, H.M. et al. (2000). "The Protein Data Bank". In: *Nucleic Acids Research* 28, pp. 235–242. DOI: doi:10.1093/nar/28.1.235.

Boutz D. R.and Cascio, D., L. J. Whitelegge J.and Perry, and T.O. Yeates (2007). "Discovery of a thermophilic protein complex stabilized by topologically interlinked chains". In: *Journal of Molecular Biology* 368.5, pp. 1332–1344.

Brown, Kevin Q. (1979). "Voronoi diagrams from convex hulls". In: *Information Processing Letters* 9 (5), pp. 223–228. DOI: 10.1016/0020-0190(79)90074-7.

DeDecker, Brian S. et al. (1996). "The crystal structure of a hyperthermophilic archaeal TATA-box binding protein". In: *Journal of Molecular Biology* 264.5, pp. 1072–1084. ISSN: 00222836. DOI: 10.1006/jmbi.1996.0697.

Dormann, Carsten F. et al. (2013). "Collinearity: A review of methods to deal with it and a simulation study evaluating their performance". In: *Ecography* 36.1, pp. 27–46. ISSN: 16000587. DOI: 10.1111/j.1600-0587.2012.07348.x.

Elleuche, S. et al. (2013). "Structural and biochemical characterisation of a NAD+-dependent alcohol dehydrogenase from Oenococcus oeni as a new model molecule for industrial biotechnology applications". In: *Applied Microbiology Biotechnology* 97, pp. 8963–8975. DOI: 10.1007/s00253-013-4725-0.

Gallier, Jean and Jocelyn Quaintance (2017). *Aspects of Convex Geometry Polyhedra, Linear Programming, Shellings, Voronoi Diagrams, Delaunay Triangulations*, p. 334.

Gong, Haipeng, Lauren L. Porter, and George D. Rose (2011). "Counting peptide-water hydrogen bonds in unfolded proteins". In: *Protein Science* 20.2, pp. 417–427. ISSN: 09618368. DOI: 10.1002/pro.574.

Harrington, WF and John A Schellman (1956). "Evidence for the instability of hydrogen-bonded peptide structures in water, based on studies of ribonuclease and oxidized ribonuclease." In: *C R Trav Lab Carlsberg Chim* 30 (3), pp. 21–43.

Hickey, Donal A. and Gregory A.C. Singer (2004). "Genomic and proteomic adaptations to growth at high temperature". In: *Genome Biology* 5.10, pp. 1–7. ISSN: 14656906. DOI: 10.1186/gb-2004-5-10-117.

Hoerl, Arthur E. and Robert W. Kennard (1970). "Ridge Regression: Biased Estimation for Nonorthogonal Problems". In: *Technometrics* 12.1, pp. 55–67. ISSN: 15372723. DOI: 10.1080/00401706.1970.10488634.

Hurtado, F., M. Noy, and J. Urrutia (1999). "Flipping Edges in Triangulations". In: *Discrete Computational Geometry* 3 (22), pp. 333–346. DOI: 10.1007/PL00009464.

Jumper, J. et al. (2021). "Highly accurate protein structure prediction with AlphaFold". In: *Nature* (596), pp. 583–589. DOI: 10.1038/s41586-021-03819-2.

Kemp, Graham (2019). *Triominoes 0.1.3*. [Online; accessed April 23, 2021]. URL: http://www.cse.chalmers.se/~kemp/triominoes/manual_page.txt.

Kessel, Amit (2021). *Amino Acid Structure*. [Online; accessed April 23, 2021]. URL: https://amit1b.files.wordpress.com/2008/03/amino-acid-structure2.png.

Kopp, Jürgen and Torsten Schwede (2004). "The SWISS-MODEL repository of annotated three-dimensional protein structure homology models". In: *Nucleic Acids Research* 32.DATABASE ISS. Pp. 230–234. ISSN: 03051048. DOI: 10.1093/nar/gkh008.

Lawson, C. L. (1977). "Software for C1 Surface Interpolation". In: *Jet Propulsion Laboratory*.

Lee, B. and F.M. Richards (1971). "The interpretation of protein structures: Estimation of static accessibility". In: *Journal of Molecular Biology* 55.3, pp. 379–400.

Li, Gang et al. (2019). "Machine Learning Applied to Predicting Microorganism Growth Temperatures and Enzyme Catalytic Optima". In: *ACS Synthetic Biology* 8.6, pp. 1411–1420. ISSN: 21615063. DOI: 10.1021/acssynbio.9b00099.

Li, W and A Godzik (2006). "Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences". In: *Bioinformatics* 22.13, pp. 1658–9. DOI: 10.1093/bioinformatics/btl158.

OpenStax (2014). [Online; accessed 2 Nov 2021]. URL: https://openstax.org/books/biology/pages/3-4-proteins.

Pedregosa, F. et al. (2011). "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12, pp. 2825–2830.

Petsko, G. A. (2001). "Structural basis of thermostability in hyperthermophilic proteins, or "there's more than one way to skin a cat"". In: *Methods in Enzymology* 334.i, pp. 469–478. ISSN: 00766879. DOI: 10.1016/S0076-6879(01)34486-5.

Pieper, Ursula et al. (2004). "MODBASE, a database of annotated comparative protein structure models, and associated resources". In: *Nucleic Acids Research* 32.DATABASE ISS. ISSN: 03051048. DOI: 10.1093/nar/gkh095.

Popelier, Paul L.A. and Fiona M. Aicken (2003). "Atomic properties of amino acids: Computed atom types as a guide for future force-field design". In: *ChemPhysChem* 4.8, pp. 824–829. ISSN: 14394235. DOI: 10.1002/cphc.200300737.

Saabas, Ando (2015). *TreeInterpreter: Random forest interpretation with scikit-learn*. [Online; accessed 9 Sept, 2021]. URL: https://github.com/andosa/treeinterpreter.

Simon, Noah et al. (2013). "A sparse-group lasso". In: *Journal of Computational and Graphical Statistics* 22.2, pp. 231–245. ISSN: 10618600. DOI: 10.1080/10618600.2012.681250.

Tolosi, L and T. Lengauer (2011). "Classification with correlated features: unreliability of feature ranking and solutions". In: *Bioinformatics* 14.27, pp. 1986–94. DOI: 10.1093/bioinformatics/btr300..

Tsai, Jerry et al. (1999). "The packing density in proteins: Standard radii and volumes". In: *Journal of Molecular Biology* 290.1, pp. 253–266. DOI: 10.1006/jmbi.1999.2829.

Ulfenborg, Josephine (2020). "Machine learning to predict enzymes' optimal catalytic temperature".

Vihinen, M (1987). "Relationship of protein flexibility to thermostability". In: *Protein engineering* 6.1, pp. 477–480. DOI: https://doi.org/10.1093/protein/1.6.477.

Vogt, Gerhard, Stefanie Woell, and Patrick Argos (1997). "Protein thermal stability, hydrogen bonds, and ion pairs". In: *Journal of Molecular Biology* 269.4, pp. 631–643. ISSN: 00222836. DOI: 10.1006/jmbi.1997.1042.

Xie, Neng Zhong et al. (2015). "Exploring strong interactions in proteins with quantum chemistry and examples of their applications in drug design". In: *PLoS ONE* 10.9, pp. 1–19. ISSN: 19326203. DOI: 10.1371/journal.pone.0137113.

Yang, Yang et al. (2019a). "ProTstab - Predictor for cellular protein stability". In: *BMC Genomics* 20.1, pp. 1–9. ISSN: 14712164. DOI: 10.1186/s12864-019-6138-7.

— (2019b). "ProTstab - Predictor for cellular protein stability". In: *BMC Genomics* 20.1, pp. 1–9. ISSN: 14712164. DOI: 10.1186/s12864-019-6138-7.

yngvem (2019). "Efficient Group Lasso in Python". In: URL: GitHub%20repository:%20https://github.com/yngvem/group-lasso.

Yuan, M. and Y Lin (2006). "Model selection and estimation in regression with grouped variables". In: 68, pp. 49–67. URL: http://pages.stat.wisc.edu/~myuan/papers/glasso.final.pdf.

Zou, H. and T. Hastie (2005). "Regularization and Variable Selection via the Elastic Net". In: *Journal of the Royal Statistical Society: Series B* 67, pp. 301–320. DOI: 10.1111/j.1467-9868.2005.00503.x.

A

Appendix 1

The Popelier atomic groups that were used to classify each atom are listed in Table A.1.

Amino Acid	Atom	Popelier Group	Amino Acid	Atom	Popelier Group
ALA	C	C19	LEU	N	N2
ALA	CA	C8	LEU	O	O5
ALA	CB	C2	LEU	OXT	O4
ALA	N	N2	LYS	C	C19
ALA	O	O5	LYS	CA	C7
ALA	OXT	O4	LYS	CB	C3
ARG	C	C19	LYS	CD	C3
ARG	CA	C8	LYS	CE	C3
ARG	CB	C3	LYS	CG	C3
ARG	CD	C10	LYS	N	N2
ARG	CG	C3	LYS	NZ	N1
ARG	CZ	C21	LYS	O	O5
ARG	N	N2	LYS	OXT	O4
ARG	NE	N2	MET	C	C19
ARG	NH1	N1	MET	CA	C8
ARG	NH2	N1	MET	CB	C3
ARG	O	O5	MET	CE	C21
ARG	OXT	O4	MET	CG	C3
ASN	C	C19	MET	N	N2
ASN	CA	C8	MET	O	O5
ASN	CB	C3	MET	SD	S6
ASN	CG	C19	MET	OXT	O4
ASN	N	N2	PHE	C	C19
ASN	ND2	N1	PHE	CA	C8
ASN	O	O5	PHE	CB	C3
ASN	OD1	O4	PHE	CD1	C12
ASN	OXT	O4	PHE	CD2	C12
ASP	C	C19	PHE	CE1	C12
ASP	CA	C8	PHE	CE2	C12
ASP	CB	C3	PHE	CG	C14
ASP	CG	C19	PHE	CZ	C12
ASP	N	N2	PHE	N	N2
ASP	O	O5	PHE	O	O5

ASP	OD1	O4	PHE	OXT	O4
ASP	OD2	O4	PRO	C	C19
ASP	OXT	O4	PRO	CA	C8
CYS	C	C19	PRO	CB	C3
CYS	CA	C8	PRO	CD	C3
CYS	CB	C3	PRO	CG	C3
CYS	N	N2	PRO	N	N2
CYS	O	O5	PRO	O	O5
CYS	SG	S3	PRO	OXT	O4
CYS	OXT	O4	SER	C	C19
GLN	C	C19	SER	CA	C10
GLN	CA	C8	SER	CB	C3
GLN	CB	C3	SER	N	N2
GLN	CD	C19	SER	O	O5
GLN	CG	C3	SER	OG	O3
GLN	N	N2	SER	OXT	O4
GLN	NE2	N1	THR	C	C19
GLN	O	O5	THR	CA	C8
GLN	OE1	O4	THR	CB	C9
GLN	OXT	O4	THR	CG2	C2
GLU	C	C19	THR	N	N2
GLU	CA	C10	THR	O	O5
GLU	CB	C3	THR	OG1	O3
GLU	CD	C19	THR	OXT	O4
GLU	CG	C3	TRP	C	C19
GLU	N	N2	TRP	CA	C8
GLU	O	O5	TRP	CB	C3
GLU	OE1	O4	TRP	CD1	C15
GLU	OE2	O4	TRP	CD2	C14
GLU	OXT	O4	TRP	CE2	C18
GLY	C	C19	TRP	CE3	C12
GLY	CA	C8	TRP	CG	C14
GLY	N	N2	TRP	CH2	C12
GLY	O	O5	TRP	CZ2	C12
GLY	OXT	O4	TRP	CZ3	C12
HIS	C	C19	TRP	N	N2
HIS	CA	C8	TRP	NE1	N2
HIS	CB	C3	TRP	O	O5
HIS	CD2	C15	TRP	OXT	O4
HIS	CE1	C17	TYR	C	C19
HIS	CG	C15	TYR	CA	C10
HIS	N	N2	TYR	CB	C3
HIS	ND1	N2	TYR	CD1	C12
HIS	NE2	N2	TYR	CD2	C12
HIS	O	O5	TYR	CE1	C12
HIS	OXT	O4	TYR	CE2	C12

ILE	C	C19	TYR	CG	C14
ILE	CA	C8	TYR	CZ	C18
ILE	CB	C4	TYR	N	N2
ILE	CD1	C2	TYR	O	O5
ILE	CG1	C3	TYR	OH	O2
ILE	CG2	C2	TYR	OXT	O4
ILE	N	N2	VAL	C	C19
ILE	O	O5	VAL	CA	C8
ILE	OXT	O4	VAL	CB	C4
LEU	C	C19	VAL	CG1	C2
LEU	CA	C8	VAL	CG2	C2
LEU	CB	C3	VAL	N	N2
LEU	CD1	C2	VAL	O	O5
LEU	CD2	C2	VAL	OXT	O4
LEU	CG	C4			

Table A.1: A table of each atom in every amino acid classified according to Popelier's categories.