ELUCIDATING DYNAMIC MECHANISMS FOR EXTENDED SPECTRUM ANTIBIOTIC RESISTANCE IN CLASS A BETA-LACTAMASE THROUGH MACHINE LEARNING ON MOLECULAR DYNAMICS SIMULATIONS

by

Chris Avery

A thesis submitted to the faculty of
The University of North Carolina at Charlotte
in partial fulfillment of the requirements
for the degree of Master of Science in
Applied Physics

Charlotte

2019

Approved by:

Dr. Donald Jacobs

Dr. Yuri Nesmelov

Dr. Susan Trammell

ABSTRACT

CHRIS AVERY. Elucidating Dynamic Mechanisms for Extended Spectrum Antibiotic Resistance in Class A Beta-Lactamase through Machine Learning on Molecular Dynamics Simulations. (Under the direction of DR. DONALD JACOBS)

Beta-lactamase is a protein which is produced in bacteria and is a primary cause of antibiotic resistance to Beta Lactam antibiotics. Beta Lactams are among the most common types of antibiotics (including penicillin) and thus understanding the resistance conferred to bacteria by beta-lactamase enzymes is a critical step in developing effective drugs. There are many mutants of the beta-lactamase enzyme, each with the same function but different substrate affinities to the various types of antibiotics. We are interested in identifying intrinsic dynamics of apo structures, if any, to elucidate differences in substrate specificity, specifically the motions which can explain how Extended Spectrum beta-lactamases (ESBLs) are able to bind to such a wide variety of substrates. To characterize dynamical differences between mutant proteins, we are employing Essential Dynamics, which is the most commonly employed method to analyze the internal dynamics of proteins, as well as several other supervised machine learning methods. We show that Quadratic Discriminant Analysis, when used with a filtering method to circumvent the multicollinearity problem, can be used to classify structures from the wild type (TEM-1) and ESBL (TEM-52) with greater accuracy than obtained from unsupervised learning. Finally, an in-house supervised learning method is used to identify differences in molecular motion to elucidate mechanisms likely responsible for, at least in part, the experimentally determined differences in binding affinity between the ESBL TEM-52 with respect to the wild type TEM-1.

ACKNOWLEDGEMENTS

In the writing of this thesis I have been given a lot of guidance and support. I would like to first thank my advisor, Dr. Donald Jacobs, for helping to develop this project and helping me at all the steps. I would also like to thank my other committee members, Dr Yuri Nesmelov and Dr Susan Trammell for their time and assistance in and outside of the classroom. I would like to thank in particular Dr. Jenny Farmer, John Patterson, Dr. Charles David, Lonnie Baker, Tyler Grear, and Dion Chapel for their help and discussions in either computational or conceptual understanding of various components of this project. I would like to thank the other members of the BioMolecular Physics group with whom I had many meetings where they provided insight into my analysis. I would finally like to thank the UNCC and University Reserach Computing team for allowing the use of the copperhead cluster for simulations.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

Antibiotic resistance is one of the greatest problems facing the pharmaceutical industry and the future of healthcare. Currently many forms of antibiotics are available on the market, but the most commonly prescribed are Beta Lactam antibiotics, which includes penicillin.[1][2] There are three major mechanisms of bacterial resistance to Beta Lactam antibiotics.[3] The first prevents the drug from reaching its target and the second is to change the physical properties of the binding site so that the Beta Lactams cannot bind. The third and most common method is that bacteria can secrete enzymes which bind to antibiotics and render them useless. There are many types of these molecules but the most common is an enzyme called beta-lactamase.[3] Beta-lactamase enzymes are produced in a myriad of bacteria such as E. Coli and Staph Aureus.[4] The beta-lactamase family is extremely large and is traditionally broken into 4 large classes based on sequence and structure.[5] Within each class, there are sub families of enzymes which share structural motifs such as secondary structures. For this project we have chosen to work with Class A, TEM-family beta-lactamase because it is most common and this family contains many enzymes with diverse functional characteristics.

When considering the breakdown of different beta-lactamase it is also common to categorize them by their functional characteristics. One of the ways this is commonly done is by classifying them by the substrates to which the enzyme can bind, that is the types of Beta Lactam antibiotics to which the beta-lactamase can confer resistance.[6] These classifications can be defined into three major groups: Cephalosporin resistance, Extended Spectrum and Inhibitor based resistance, and Metallo-beta-lactamase (Class B) resistance.[2] There are many beta-lactamase which are naturally produced by bacteria as a defense against lethal antibiotics. When penicillin was first discovered in the early part of the twentieth century, it was not long before the first beta-lactamases were discovered. They were originally termed Penicillinase or Penicillin Binding Proteins.[7] When antibiotics were first released into the health environment it was noticed that more sequences of these penicillin binding proteins were being discovered, and in 1965 the first TEM family beta-lactamase (TEM-1, also known as the wild type beta-lactamase) was isolated in a patient in Greece.[2] The ancient forms of beta-lactamase were only able to function against penicillins, and TEM-1 could confer resistance to penicillin and first generation Cephalosporins.

The time period around TEM-1's discovery is known as the golden age of drug discovery as many new antibiotics were developed in order to counter the effect of beta-lactamase. Modifications to existing Beta Lactam antibiotics as well as entirely new classes of molecules were designed and the release of these new drugs had the effect of creating a large selective pressure on the existing beta-lactamase sequences. The result of this was an explosion of new beta-lactamases which evolved to increase their functional capacities against the new drugs being pushed out into the environment. Within Class A TEM-family beta-lactamase alone, the number of isolated beta-lactamases has been estimated to be over 200, most of them appearing between 1965 and 2019. [7] [8] These new enzymes have been able to confer the same resistance as TEM-1 as well as second and third generation Cephalosporins, Monobactams, and more recently Carbapenems. Molecules with this extended resistance profile are called to as Extended Spectrum beta-lactamase (ESBL). The resistance conferred to bacteria by these ESBLs is currently a crisis which is facing pharmaceutical companies as well as healthcare professionals because current methods in drug development cannot keep up with the rate of evolution of beta-lactamase. Thus, a better understanding of how extended spectrum resistance can arise in a beta-lactamase due to the point mutations of its sequence is pivotal in order for researchers to design new at a faster pace to heal people with bacterial infections.

In order to study the emergence of these resistance profiles several methods can be employed, one common method being Molecular Dynamics (MD) simulation studies. MD is a method which begins with a 3-dimensional structure of a molecule and simulates the motions that the molecule would undergo by iteratively solving Newton's equations of motion for each atom in the simulation. These motions allow researchers to imitate the physics of biological macromolecules, as well as collect large amounts of data which can be used with statistical mechanics to understand the properties and dynamics of molecules. The major limitation of MD studies is the accuracy to which atomic interactions can be calculated, however there are many forcefields which have been well verified to approximate reality and are routinely used in the field of biophysics to study protein dynamics. The dynamics observed can help elucidate functional mechanisms in proteins.

In order to study the large output of data that results from MD and be able to make physical and biological inference, an array of statistical and machine learning methods have been developed to analyze the data. One of the most common methods used in protein dynamics is called Principal Component Analysis (PCA). PCA was introduced as a type of regression by Pearson in 1901 [9] and developed by Hotelling in 1936 [10]. The purpose is to reduce the dimensionality of a dataset (protein structures in this case) to its most variant components. The data can be reconstructed by using a subset of components that exhibit the highest variance to create a denoised version of the data which show the dominant and presumably important features.[11] PCA has been applied to many fields such as signal processing, image processing, and data reduction or compression. PCA for protein analysis was introduced under the name essential dynamics by Amadei, Linssen, and Brendensen in 1993[12], and continues to be widely used today.

For this project we have run MD simulations on wild type TEM-1 beta-lactamase and ESBL enzymes and we would like to be able to explain the dynamics associated with the ESBL resistance profile. Experimental studies have shown that different mutations within the same family of beta-lactamase have different functions.[13] In biology the idea that the structure and function of proteins are related is well accepted, and although more difficult to quantify and measure, protein dynamics is a critical link between structure and function. Examples of several studies which make use of these ideas to compare different proteins via internal dynamics are reviewed by Micheletti in in [14]. Our aim is to use analysis methods such as PCA, and some supervised machine learning methods on MD simulation data to isolate the dynamics which can discriminate differences between the mutants in order to elucidate how ESBLs arise. This mechanistic information opens the possibility to rationally design new antibiotic drugs which are not just effective against one strain of beta-lactamase but will have a much more general effectiveness against bacterial infection.

CHAPTER 2: BIOCHEMISTRY BACKGROUND

TEM family beta-lactamase are the most commonly found, and most members of the family have a highly conserved sequence and structure. Based on the structure/function paradigm, one might expect Class A beta-lactamases share the same mechanism of action. This mechanism involves a Serine residue at position 70 which is 100 percent conserved across the family. This Serine is known to hydrolyze the Beta Lactam ring structure on the antibiotic ligand. All of the chosen mutants used for this project have the same number of residues, 263, and the mutant sequences differ only by point mutations. The ESBL property is hypersensitive to point mutants. A direct comparison of the intrinsic dynamics across different point mutations will be made.

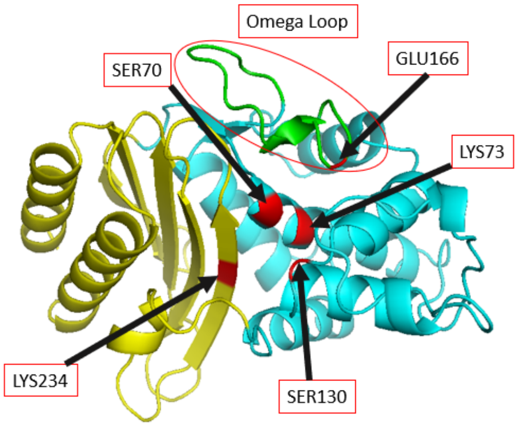


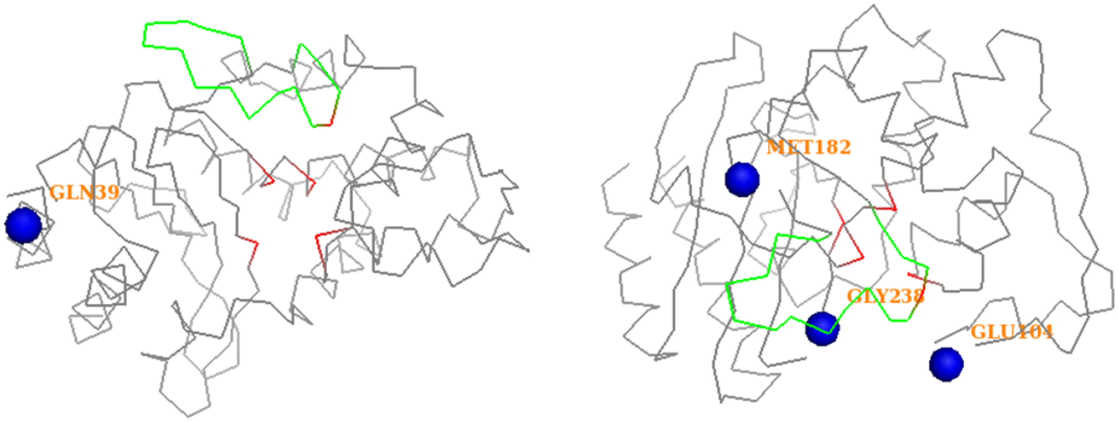
Figure 2.1: Colored image of 3-D structure of TEM-1 beta-lacatamase (PDB Code: 1XPB). Yellow and cyan represent the beta sheet and alpha helix domains, green represents the omega loop, and red highlights the residues most involved with the hydrolysis of beta lactams. The key residues in the active site have also been labeled.

The 3-dimensional structure of TEM-1 beta-lactamase (shown in Figure 2.1) is well studied and has many crystal structure entries in the Protein Database (PDB).

The global structure consists of a single alpha beta domain which can be broken into two main regions where the binding pocket for ligands is in the area connecting them.[15][16] The first region consists of 5 anti-parallel beta sheets which are protected from solvent on one side by the N and C termini, and border the binding pocket on the other side.[16] The second region contains 6 major alpha helices which contain many of the residues important for beta-lactamases function.[15] The binding pocket consists of the cavity between the wall of beta sheets, the alpha helices of the second domain, and a third interesting region called the omega loop which sits on the surface of the enzyme (in green in Figure 2.1). The omega loop consists of the residues 163-178 and has been shown in many studies to play an important role in substrate recognition.[17]

The active site (shown in red) is conserved across the mutations we are considering and consists of the residues 70, 73, 130, 131, 136, 164, 166, 179, 233, and 234 [17] which have been shown to be important in hydrolysis of Beta Lactam by experiment. As stated, the Serine at residue 70 in the most important of these active site residues, as this is what forms the complex with the Beta Lactam. The active site contains residues which directly act in the hydrolyzation process of the Beta Lactam have been described as the primary active residues. Some studies also include other active site residues that are secondary. We have labeled the primary active residues, SER70, LYS73, SER130, GLU166, LYS234, the Mechanistic Site.[18][17] Two of the residues, GLU166 and LYS234, are still debated in the literature, however we include them in order to capture all possible relevant motions.

For this project we began with MD Simulation data which had been generated within our lab for another project.[19] From this data we chose eight different crystal structures of TEM family beta-lactamase. Each crystal structure was to the TEM-1 (wild type), TEM-52 (extended-spectrum) and TEM-2 (functionally similar to TEM-1) sequences using the mutagenesis tool in Edu Pymol 1.7.4.5 if the desired sequence



(a)

(b)

Figure 2.2: Two views of beta-lactamase's structure which highlight the mutation sites of (a) TEM-2 and (b) TEM-52. The omega loop is show in green and the mechanistic residues are in red for reference.

was not already available. The series of mutations which change TEM-1 into TEM-52 is E104K, M182T, and G238S[20], TEM-1 to TEM-2 is Q39K [21], and TEM-30 to TEM-1 mutation is S244R. It should also be noted that the 1JWP crystal structure is a mutation of the TEM-1 sequence, M182T. This mutation does not occur on its own in clinical isolates, however through point mutation studies it has been shown to increase the stability of beta-lactamase proteins.[22] The crystal structures and mutations are summarized in the Table 2.1.

Each mutation to the TEM-1 sequence has been studied extensively to determine their effect on substrate specificity and kinetic activity. Together the three point mutations from TEM-1 to TEM-52 confer extended spectrum resistance to beta-lactamase. This expression of extended spectrum resistance and TEM 52's comparably high activity to cephalosporins and other beta lactams[23] make the protein well suited for this study, comparing the dynamics of the wild type to its ESBL mutant. As stated above, the M182T mutation increases stability but does not in general affect either substrate specificity or kinetic function.[24] It is not found in nature by itself but often expressed with other mutations which do increase kinetic activity but decrease overall stability. It is a common mutation for ESBL's to express along with other mutations because of this. G238S confers resistance to third generation cephalosporins which is the hallmark of ESBLs.[8] E104K on its own does not increase activity, however when combined with other mutations like G238S in the TEM-52 sequence the mutation increases resistance to third generation cephalosporins and aztreonam as well as being proposed to also act to stabilize the effects of the G238S mutation.[8] Both the E104K and G238S mutations act to deform or rearrange residues in the active site and thus directly affect the binding of antibiotics to the substrate.[22] It can be seen in Figure 2.2(b) that the two functional site mutations are close to or part of the active site. The stabilizing mutation, M182T, is found distant from the active site which could help explain why it is not change functional behavior of the enzyme.[25]

The mutation of Q39K (TEM-1 to TEM-2) is notable because the TEM-2 represents the beginning of a different mutation pathway of TEM beta-lactamase [26]. The mutation has been found on its own not to affect the kinetic function of its derivatives, but it does induce more mobility around the omega loop [25]. In Figure 2.2(a) the mutation for this enzyme is shown to be on one of the terminal alpha helices, separated from the active site by a wall of beta sheets. This is another example of a mutation that does not change functional properties and is far from the active site like M182T in TEM-52.[25] This protein was chosen to be included in this work as a control group because of its functional similarity to TEM-1. Based on experimental evidence, then, we expect to find differences between TEM-1 and TEM-52 as well as differences between TEM-2 and TEM-52, but similarity between TEM-1 and TEM-2.

Table 2.1: Summary of crystal structures used and mutations. (*point mutation of TEM-1 sequence)

Crystal Structure (PDB code)	Wild Type	Mutations to TEM-1	Mutations to TEM-52	Mutations to TEM-2
1ERM	TEM-1	0	3	1
1ERO	TEM-1	0	3	1
1ERQ	TEM-1	0	3	1
1HTZ	TEM-52	3	0	4
1JWP	TEM-1*	1	2	2
1LHY	TEM-30	1	4	2
1XPB	TEM-1	0	3	1
3JYI	TEM-1	0	3	1

CHAPTER 3: METHODS

In this chapter we review the molecular dynamics simulations and the two major algorithms used for analyzing the dynamical differences between wild type and ESBL beta-lactamases are described: Principal Component Analysis (PCA) and Quadratic Discriminant Analysis.

3.1 Molecular Dynamics Simulation

For each protein we analyzed a 500 nanosecond MD trajectory consisting of 10000 frames. In the original dataset, complete 500 nanosecond trajectories only existed for the 1ERM and 1HTZ crystal structures (both mutations). For the other 6 structures we followed the exact same protocol as earlier.[19] Specifically we used the MD software GROMACS to generate trajectories.[27] MD simulations were run on GROMACS 5.1.2 or GROMACS 2018 using the force field AMBER99SB-ILDN protein, nucleic AMBER94 [28] and the TIP3P water model. [29] Disulfide bonds in the protein were also preserved. The protein was placed at the center of a cubic box with at least 1 nm of space between it and the walls. Solvation was done using the TIP3P water model and then the whole box was neutralized electronically by replacing some solvent molecules with negative CL ions or positive NA ions. The number of ions added changed depending on the sequence, TEM-1 required 5 NA and TEM-52 required 7 NA. The energy of the coordinates were then minimized using steepest descents and sent to an equilibration stage. The temperature was equilibrated to 300K over 1 ns using a Berendson Thermostat [30] and then equilibrated to 1 Barr of pressure for an additional 100 ps using a Parinello-Rahman barostat. [31] The production run was run for 500 ns, saving configurations every 50 ps giving us 10000 frames per trajectory using a Verlet cutoff scheme. In the analysis we considered the trajectory as a whole and also as broken in to five equal sections of 2000 frames.

Figure 3.1 shows the root mean square deviation of the 24 simulations over all frames. Root mean squared deviation (RMSD) is defined as the average deviation of a certain conformation from a specified reference structure.

$$RMSD = \sqrt{\frac{1}{3N}(\vec{C(t)} - \vec{C_{ref}}) \cdot (\vec{C(t)} - \vec{C_{ref}})} \tag{3.1}$$

RMSD, along with other physical properties, can be used to do evaluate the quality of an MD simulation in terms of whether or not the protein was sufficiently close to equilibrium which is sometimes referred to as the convergence of a simulation. Properly converged simulations will have a constant average RMSD over the whole simulation, however the RMSD will fluctuate around this mean from conformation to conformation.[32] Convergence is important when analyzing dynamics from MD simulation in order to know you are looking at natural motions that the protein may undergo instead of un-physical motions driven by some outside force. The majority of the simulations converged within a local energy basis but there are a few notable exceptions, 1ERM TEM-52 and 1LHY TEM-52 make a major jump in RMSD during their runs. The jump in 1LHY TEM-52 spikes and comes back to its original RMSD but the 1ERM remains in its higher RMSD state indicating a large conformation change. 1LHY TEM-2 did not see such a large spike but it did increase with a small spike at the end. All the 1ERO proteins provide examples of simulations with RMSD that did not undergo a dramatic change indicating that these conformations are representative of a single energy basin. For the purpose of subsequent statistical analysis, these jumps appear to be outliers. In reality, jumps between basins are expected on long-time scales that is not on an attainable time scale for the computing resources available. We therefore view convergence in the context of exploring a single local energy basin, which is standard practice in the field.

For Cartesian PCA (cPCA) we used a software called Java Essential Dynamics

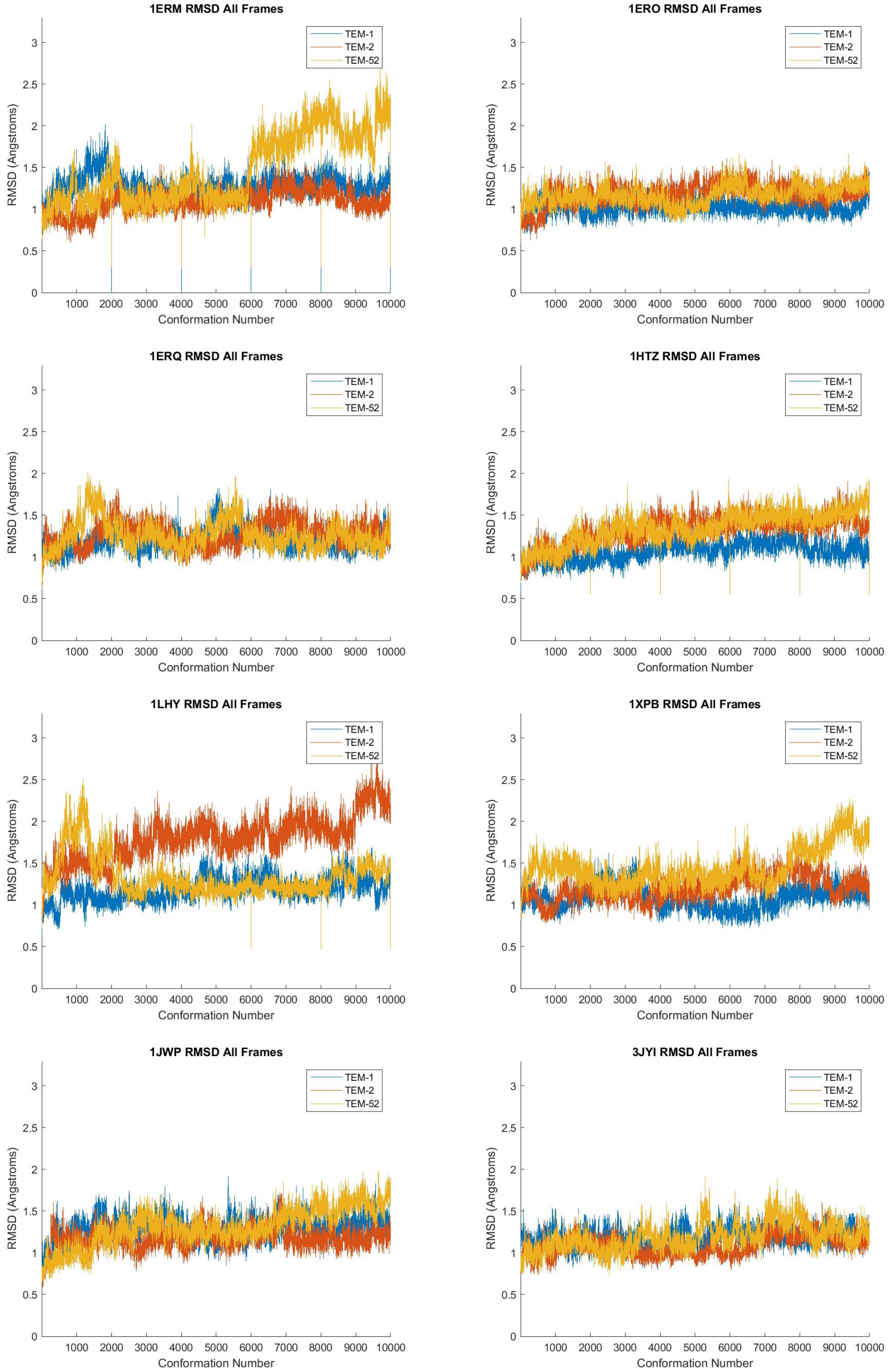


Figure 3.1: RMSDs for all 24 simulations.

(JED).[33] JED performs PCA in two steps: the pre-processing run and the analysis run. The pre-processing run takes as input the set of PDBs which constitute the trajectory and formats them into a data matrix, as well as aligning all the frames of the trajectory to a specified reference structure. The alignment is done by a quaternion rotation of each structure. We chose the reference structure to be the first of the trajectory, however one could also choose the mean structure or the centroid structure. This run will also calculate the RMSF and RMSD across the trajectory. The analysis run performs PCA as well as any other driver chosen. JED has capability of doing cPCA and dpPCA (distance pair) or both on either or any combination of the covariance, correlation, or partial correlation matrix. JED includes several other useful features which include a free energy surface (FES) calculator which calculates a 2-D projection of the FES using two PCA modes as the order parameters, a visualization driver which takes the eigenvectors and makes movies of each mode determined by PCA as well as movies of combinations of modes, and a subspace analysis driver which calculates a variety of statistical metrics for comparing essential subspaces of proteins. Another important feature of JED is its pooling function in which it pools multiple trajectories into the same data matrix. This feature is essential for allowing comparison across proteins. All other analysis was performed using MATLAB 2018a. A new method of PCA, displacement PCA, was developed and implemented in this platform. Linear/Quadratic discriminant analysis and SPLOC was also performed using MATLAB.

3.2 PCA and Essential Dynamics

Essential dynamics is one of the most popular methods used to characterize the internal dynamics of proteins. The goal of both of these methods is to reduce the overall motion that a protein undergoes to a smaller subset of "essential motions" that are broken down into orthogonal component "modes". The particular subset identified is dependent on the algorithm used, for example a few methods commonly used to describe the essential dynamics of a protein, most commonly PCA or Elastic Network Models (ENM) is used.[14]

In general PCA takes a high dimensional data set, data in which the important features depend on complex and difficult to visualize relationships of the degrees of freedom, and reduces it down to a lower dimensional representation in what are considered to be collective variables. This low dimensional representation is meant to simplify the data set while preserving as much of the original variance as possible.[11] This method can be applied to a variety of data sets to provide a way to classify or cluster the data, and is a very popular method for performing dimensionality reduction. In this section the general PCA algorithm will be presented briefly, followed by a more detailed discussion of how to perform PCA on molecular dynamics simulation data.

PCA can be broken into four major steps: form the covariance matrix, perform the eigenvalue decomposition, sort order the eigenvectors and select the essential subspace, and project the original data into its lower dimensional representation.[34] The input required for the algorithm is a series of observations of some data which depends on many variables. In order to get the best results from the algorithm there should be much more independent samples than degrees of freedom in the dataset otherwise the statistics in the dataset may not be enough to approximate the covariance. It is also cautioned that PCA is an algorithm which is based on variance, and therefore the degrees of freedom must be covariant with each other or else the algorithm will return nothing significant. Seeing how PCA is an unsupervised learning method no additional information about the data is required.

To better understand what PCA is doing, a simple example is shown in Figure 3.2. We start with a random Gaussian plotted in x-y coordinates. PCA looks for the directions of the data that have the maximum variance and creates a transformation to represent the original data in these directions. In this case it is clear that the direction of maximum variance is neither X1 or X2, but a linear combination of the two right through the Gaussian. The algorithm is able to identify this and after transforming the data, this collective direction is now the new "X1" variable. The next most variant direction is the new "X2". It should be noted that this example is very simplistic. With only two degrees of freedom in the original data there is not a dimensionality reduction in the PCA transformation because we keep both variables. This means the data is completely reconstructed in the new representation and that the PCA transformation, when not loosing any information, corresponds to a rotation of the data.

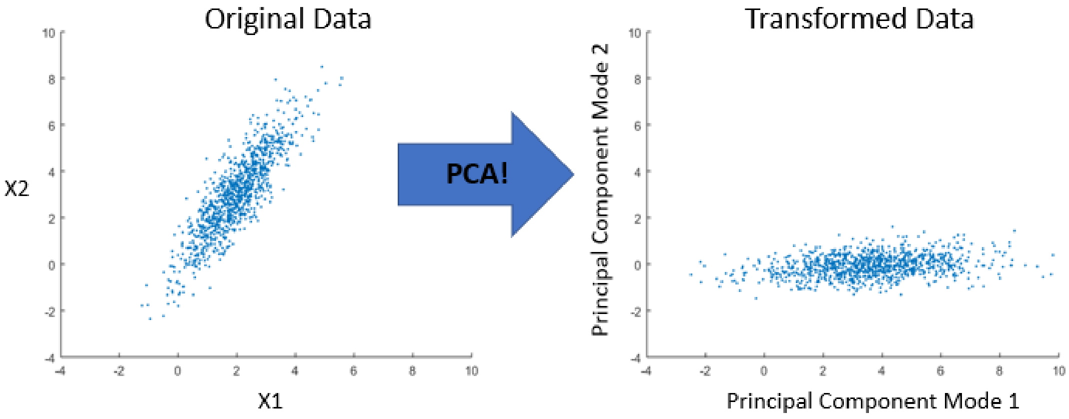


Figure 3.2: Schematic of PCA algorithm which takes two dimensional gaussian data generated in MATLAB and computes the directions of maximum variance, and then projects the data into these axes. The result is a two dimensional rotation as the data is already two dimensional to begin with but now the gaussian data aligns with the x-axis.

A geometrical interpretation of PCA can be borrowed from the moment of inertia. When a three-dimensional object rotates it has "special axes" along which the object will naturally rotate. All rotations of an object can be described as a linear combination of these principal axes, and thus they define an ideal basis set in which the problem of describing rotations can be described. PCA is a method of finding the principal axes of any data set, but it chooses the directions based on variance (i.e. the second moments in a moment of inertia matrix). The directions themselves (often referred to as collective features or Principal Component modes) are linear combinations of the original degrees of freedom which are given by the eigenvectors of the covariance matrix and the eigenvalues are the total variance in that direction.[34] Principal component analysis calculates these eigenvectors and eigenvalues and selects the top N modes (where N is defined by the user) associated with the largest eigenvalues. The selected subspace of the covariance matrix defined by these eigenvectors contains the majority of important information about the dataset called the essential subspace. The dataset is then reconstructed by projecting out all information not contained in this space and analyzed. The projection of a (mean centered) sample into one of the principal axes is called a principal component (PC).

$$PC_i(t) = (\vec{C(t)} - < \vec{C} >) \cdot \vec{V_i}. \tag{3.2}$$

$PC_i(t)$ represents the projection of sample $C(t)$ projected into the ith eigenvector, V_i. This number represents a score of how much a sample correlates to that principal component mode. The major benefit of this is that it has been shown that high dimension data can be reconstructed in much smaller dimensions while only losing a relatively small amount of information (variance) [34][33]. The low dimensional representation of PC's of data can be plotted in two- or three-dimensional plots to reveal features in the data which may have been too complex to see in high dimensions. These dots can be especially revealing clustering within a data set, and when it does the PC's will blur together, and thus we refer to these plots as "fuzzball" plots.

The selection of the number of features N which comprise the essential subspace can be determined via two methods, both having some validity thus the method for use is up to the analyst. The first method is to define a cumulative variance threshold which represents the minimum percentage of the total variance in the data which the essential subspace ought to be able to reproduce. Some studies cite that 70-80 percent of variance can be described in the top 10 to 20 PCA modes [34][33]. An alternative method, called Cattell's scree test[35], is a graphical method which relies on analysis of the scree plot. The scree plot is a plot of all the eigenvalues in descending order, so named for its resemblance to the scree that accumulates by falling rocks at the bottom of cliffs. The scree plot is most often characterized by a sharp falling of the eigenvalues (the cliff) in the early modes and then an abrupt leveling out (the scree) in the later modes. The kth feature at which the scree begins to level out is often thought to signal the end of the essential subspace. If PCA is being done of the correlation matrix a different way of identifying how many features comprise the essential subspace is to take the eigenvalues larger than 1.[33] For our project, in most cases we used Cattell's test because setting a cumulative variance requirement often required many more modes than necessary. This is because the full feature space is very large, 789 dimensions, and while most of the variance is indeed found in the first few modes the less variant modes had little variance distributed over many modes. Often to get a cumulative variance of 70 percent, upwards of 60 modes were necessary with 45-50 percent being in the top one or two modes. Meanwhile Cattell's test would bring the essential subspace down to 15 to 20 dimensions which agrees with other essential dynamics studies of proteins.

When performing Essential Dynamics analysis with PCA the input variables are the x-y-z coordinates of atoms from MD simulation. Often this view is coarse grained to the carbon alphas to represent the motion of a whole residue as it has been traditionally held that this is detailed enough to obtain the pertinent large-scale collective motions.[12] So for an N residue long protein there will be 3N degrees of freedom. Even with this level of coarse graining the dimensionality of protein dynamics data can be very large, as an example beta-lactamase has 263 residues and so 789 degrees of freedom. Structures for analysis can be obtained via any method, MD, Geometrical Simulation, NMR, etc. as long as the Cartesian coordinates can be retrieved. These conformations need not represent a time sequence of structures. This data is arranged into a matrix called the data matrix, A, where each column represents the coordinates of a single conformation and each row is a degree of freedom. In order to remove global translational and rotational degrees of freedom a reference structure should be defined, and all the conformations structurally aligned to it. This structure is commonly chosen to be either any conformation from the dataset or an average conformation across all structures. We chose the first structure generated in our MD simulations to be our reference. The reference centered A matrix can be defined as Z.

$$Z = A - Ref \tag{3.3}$$

Using the Z matrix the covariance matrix can be defined as the following with M as the number of samples:

$$C = \frac{1}{M-1}(Z^T Z). \tag{3.4}$$

Next one would obtain the eigenvalues and eigenvectors by diagonalizing the covariance matrix. After diagonalization the eigenvalues will be the entries of the diagonal matrix Λ and the eigenvectors are the column entries in the matrix V.[12]

$$\Lambda = V^T C V \tag{3.5}$$

The eigenvectors represent "directions" in the data set with the most variance. Because the loadings of the PCA eigenvectors can be mapped back to the original x-y-z coordinates of the protein, it is possible to extract a physical motion to which the vector corresponds. The covariance matrix is positive definite and symmetric and thus the set of eigenvectors returned constitutes a complete orthonormal vector space. In analogy to quantum mechanics, the motions obtained by PCA modes are a complete basis set of collective motions for the protein in which all motions the protein undergoes can be represented. [36] When dimensionality is reduced, this corresponds to filtering out all of the motions with small variance, or just the small motions, and the large motions left over can be studied individually.[37]

In general PCA, the next step is to create the low dimensional PC representation of the data, however for Essential Dynamics another form of PC's, displacement vector projections (DVPs), are sometimes used.[33] The only difference between PC and DVP is that a DVP is centered to some reference coordinates instead of the mean conformation. Subtracting off a physical reference turns the trajectory into a sequence of displacement vectors (DVs) instead of mean fluctuations. Physical meaning can be interpreted from DVPs when we consider that the PCA modes are themselves weighted linear combinations of the original features in the covariance matrix, and this can then be thought of as a displacement vector for all the Carbon alpha atoms. Thus the inner product of a DV and a PCA mode measures the similarity between the displacement of the conformation in question and the collective motion. If they are correlated the resulting PC or DVP will be proportionately large positive number, and if it is uncorrelated it will be negative.[12] Thus DVPs and PCs are measures of how much of each of the basis collective motions are present in each conformation. The smaller subspace of the essential motions can be analyzed via graphing as scatter plots (called fuzzball plots) and creating probability distributions of the samples along each PC mode. Comparisons between different proteins can be made through DVPs when pooling is applied.

In order to compare DVPs of two proteins directly it is necessary to employ a method for combining the data called pooling. Pooling combines the data sets (trajectories) for each protein into one set before computing the covariance matrix and pushing through the rest of PCA. We consider this pooling to give us the average covariance across the different sets pooled together which will allow us to pull out the essential dynamics common to all of the data. These will be represented in a single subspace of eigenvectors in which the data can be projected into to allow for direct comparison of DVPs. By this method of comparison we look to see if the DVPs cluster by protein type, indicating that the two proteins have different behavior along their essential motions. Overlapping of DVPs for different proteins conversely implies that the protein has similar expressions of these essential motions.

When pooling data it should be noted that for comparing dynamics of different proteins the mean centering should be done on the combined data. This shifts the mean of the whole dataset to zero, but not the means of the individual protein trajectories which capture how projections for each protein spread into the hyper plane. If mean centering is done on each protein trajectory individually before centering then all of the projections onto PCA modes will be centered at the origin (or the reference) and less comparative information is able to be extracted.

DVPs can also be used to calculate projections of the proteins free energy landscape (FEL).[38] The actual FEL exists in a high dimensional space and can be parameterized in many ways to make projected landscapes in 1, 2, and 3 dimensions, however the methods of parameterizing can often lead to artifacts in the landscape. The DVPs provide a way to parameterize the FEL so that it will describe the largest motions of a protein, although even this method can lead to artifacts if the PC modes used do not describe enough of the variance in the motion.[32] For a N dimensional FEL a probability distribution function must be parameterized using the N DVPs. This is done by binning the space containing the DVP scatter plot and calculating the probability a scatter point will be in each bin. The free energy for each bin can then be calculated by the formula from statistical mechanics, where $FE(DVP_1, DVP_2, ..., DVP_3)$ is the free energy and $P(DVP_1, DVP_2, ..., DVP_3)$ is the probability:

$$FE(DVP_1, DVP_2, ..., DVP_3) = -k_b T ln(P(DVP_1, DVP_2, ..., DVP_3)) \tag{3.6}$$

3.2.1 PCA Types

Typically the set of degrees of freedom comprising the essential dynamics of a protein is the Cartesian coordinates of all the carbon-alpha atoms selected, however any set of degrees of freedom can work. For example it is possible to select subsets of any types of atoms of interest to perform essential dynamics studies on proteins using Distance Pair PCA (dpPCA)[33][39], Dihedral Angle PCA (dhPCA)[39][40], and coarse grained contact based PCA[41]. In this work we will explore a new method to perform PCA by tracking the displacement of each carbon alpha (or atom) from frame to frame. We call this method Displacement PCA (dispPCA). The methodology for dispPCA is almost identical to Cartesian PCA, except there is an extra step in the preparation of the data to convert the data from Cartesian coordinates to displacements. To do this one takes the data matrix of coordinates, where each column represents the x,y,z coordinates of each frame from MD, and sequentially subtracts it from following frame to get the displacement between the alpha carbons of each residue from frame to frame. We call this a one frame displacement however this can be generalized to an N frame displacement by subtracting each frame from the Nth later frame.

Our main motivation for using dispPCA over cPCA is that displacements retain only information about the internal dynamics, losing the structural displacement information. Structural alignment between the proteins will be needed in the beginning (before finding the displacements) in order to make sure all the displacements can be compared on the same footing. Normal cPCA makes use of mean centered coordinates, Z, and not the un-centered coordinates for two reasons, (1) the covariance matrix is made of displacements from the mean, yielding information about the fluctuations of coordinates not coordinates themselves, and (2) projecting the actual coordinates into the subspace would cause residues further from the origin to have larger DVP values. dispPCA is already displacements and the displacements do not get large depending where they are in reference to the origin of the coordinate system. Because of the implicit dependence on the reference structure in cPCA the choice of reference structure may bias the results, such that it could cause PCA to overwhelmingly show differences between crystal structures while drowning out the sequence differences, a problem we call "crystal memory". Note that dispPCA does not require a reference structure when making the covariance matrix or the DVPs minimizing these issues.

3.3 Discriminant Analysis

Principal Component Analysis is considered an unsupervised learning technique, meaning that there is no training portion of the algorithm. PCA looks at the data given to it, determines the directions of largest variance, and determines via metrics such as RMSIP or fuzzball plots whether or not classes can be discriminated. Another method for discriminating classes within data is called Fisher's Discriminant Analysis (DA). As a supervised learning method, unlike PCA, the algorithm must be trained on a subset of data which has its class labels. Class label in this case are the mutations which are expressed by a structure, TEM-1, TEM-2 or TEM-52. The concept of DA is fundamentally different than PCA. DA is looking for the direction in the data set which separates the data the best. DA partitions further into two major algorithms: Linear Discriminant Analysis (LDA) and Quadratic Discriminant Analysis (QDA). QDA is the most general form of this algorithm and LDA is a special case explained below. LDA and QDA are very useful for classifying data and is considered a "standard" for supervised learning classification because of its effectiveness and simplicity. An advantage that these algorithms have over other Machine Learning Classification techniques is that they have an analytic solution.

The main purpose of DA is classification. To do this task a set of discriminant functions is formed from the training data which makes use of the covariance.[42] The idea was introduced by Fisher in the 30's.[43] The goal is to optimize these functions so that they satisfy Fisher's criteria[44]:

$$\lambda = argmax(\frac{W^T S_B W}{W^T S_W W}) \tag{3.7}$$

which looks for the directions which simultaneously maximizes variance between classes (The between class scatter matrix cS_B) and minimizes variance within the class (The within class scatter matrix S_W). S_B and S_W can be calculated per class and is given by[45]:

$$S_B^i = (\mu_i - \mu)(\mu_i - \mu)^T \tag{3.8}$$

$$S_W^i = \frac{1}{N_i - 1}\sum_j^{N_i}\sum_k^{N_i}(x_j - \mu_i)(x_k - \mu_i)^T \tag{3.9}$$

x is a vector representing a single sample from the data, μ_i represents the means of class i, μ is the mean of all of the data in all classes, N_C is the number of classes in the data set, and N_i is the number of samples in class i. The total S_B is the weighted average of all the individual between class scatter matrices while the total S_W is the sum of all the within class scatter matrices. A benefit of DA solutions is that they are unique for a given set of training data, because the discriminant functions which optimize Fisher's criteria only depend on the covariance of the training data.

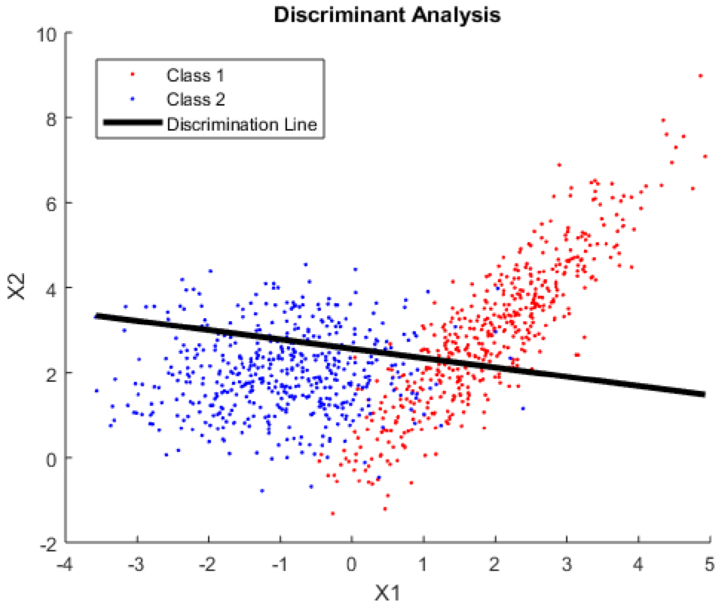


Figure 3.3: LDA is applied on two sets of 2-D random Gaussian data colored in red and blue. The black line represents the direction of largest discrimination in the data. When projected onto the line the data will be maximally separated and clustered by class.

To illustrate how Fisher's criteria works we showed a simple case with random Gaussian data in Figure 3.3. In this example we had two 2-dimensional classes which are clearly separable by inspection. The two classes are separated roughly in the x direction, however, due to the fact that the class 1 comes underneath class 2, the x direction would not be the best discriminator because the classes would overlap when projected onto the line. Fisher's criteria, often referred to by the phrase "maximizing the between class variance to within class variance", looks to find the optimal direction of discrimination with two conditions. First it looks to separate the classes as far from each other as possible by considering the means (maximizing the between class variance). Second its looks for the directions which minimize the spread in the distributions of the classes to ensure the classes overlap as little as possible on the discriminating direction (minimizing the within class variance). The means of the classes in Figure 3.3 are separated in the x direction, with a possibly slight upward slope, yet the spread in class 1 is far from minimized in this direction. Because class 1 is much narrower in a downward sloping direction, the algorithm weights this fact two and finds the combination of these directions which satisfy Fisher's criteria.

The discriminant functions can be derived by starting with the Gaussian assumption of the data. The probability that class C_k contains the sample x_i is[42]:

$$P(x_i|C_k) = \frac{1}{\sqrt{(2\pi)^n|\Sigma|}}e^{-\frac{1}{2}(x_i-\mu_k)^T S_W^{-1}(x_i-\mu_k)} \tag{3.10}$$

Thus the probability that a sample is classified as class k would be given by, $P(C_k|x_i) = P(C_k)P(x_i|C_k)$, based on Bayesian probability. $P(C_k)$ is the prior probability of class k in the data, often given as $\pi = N_k/N_{tot}$ The idea is to build a discriminant function for each class in the data set using the means and covariance matrix of each class's training data which will minimize the discriminant function[45]. This is what makes DA a supervised learning algorithm. Minimizing the discriminant function is equivalent to maximizing the probability $P(c_k|x_i)$ and by convention you will have as many discriminant functions as you have classes, and each sample is run through all of the functions. The function which returns the lowest score for the sample is the class which DA will predict for it.

In its most general form, without other limiting assumptions besides the Gaussian requirement, the discriminant function represents QDA.

$$D_k^{QDA}(\mu_k, \Sigma_k) = (x_i - \mu_k)^T \Sigma_k^{-1}(x_i - \mu_k) + ln(|\Sigma_k|) - 2ln(\pi_k) \tag{3.11}$$

Each term in the discriminant function can be interpreted for how it scores the classification of the sample. The first term is the equivalent of Fisher's criteria. The $x_i - \mu_k$ measures how far a sample is from the mean of a class which can be thought of as a between class distance for that sample and $\Sigma_k = S_W^k$. Thus the first term can be equivalent to the ratio of between to within class variance. The larger this term is the more discrimination that is achieved, and the smaller it is the more likely a sample is to be in that class. The third term is largely irrelevant but biases the discriminant functions when one class might be sampled better than another. By subtracting it says that the larger the prior probability of a class the more likely a sample is to be classified that class.

The middle term is important and is the major difference between QDA and LDA[45]. This terms says that a more variant class is more likely to contain the sample. The assumption which changes QDA to LDA is that the covariance in each class is the same, such that $|\Sigma|$ is the same for all classes. In this case the second term in the discriminant function can be dropped all together because it would contribute to all classes equally. The LDA discriminant function is

$$D_k^{LDA}(\mu_k, \Sigma_k) = (x_i - \mu_k)^T \Sigma_k^{-1}(x_i - \mu_k) - 2ln(\pi_k). \tag{3.12}$$

There is a second approach to LDA in which the algorithm can be used for dimensionality reductions. This method directly computes the directions which best discriminate the data instead of evaluating samples on a discrimination boundary like the discriminant functions do. This method takes Fisher's criteria as an optimization problem which solves for the W vectors which maximize the equation. In this form the problem can be recast into an eigenvalue problem and the method becomes similar to PCA.

$$S_B v = \lambda S_W v \tag{3.13}$$

S_B and S_W are the between and within class scatter matrices for the dataset, λ is an eigenvalue, and v is the corresponding eigenvector. This does not take the form of a typical eigenvalue problem, where there is only a matrix on one side, but this problem is called the generalized eigenvalue problem[46]. This can be solved in a similar way where the result is a set of eigenvalues and an orthogonal set of eigenvectors. Here the eigenvalues give a score to the discrimination power of the corresponding eigenvector. The eigenvectors are the directions along which the data is maximally discriminated. As in PCA, the eigenvalues and eigenvectors are sort ordered in descending order and the largest set are chosen. Unlike PCA where there is some uncertainty to how many vectors are found in the essential subspace of the data, the discriminant subspace determined by LDA will always be $N_c - 1$ large. In this way LDA can also be used for dimensionality reduction. Figure 3.3 illustrates how LDA can find the direction which optimizes Fisher's criteria to discriminate the data.

When using LDA and QDA for classification it is also important to understand the co-linearity problem. Co-linearity refers to when a data set contains redundant degrees of freedom which can be represented as a linear combination of other variables in the system. When this happens a degenerate null space can form in the data leading to eigenvalues with values of 0 or close to 0. This makes methods which rely on the inverse of the covariance matrix limited due to the fact that matrices with eigenvalue spectra like this will be unstable when inverted.[47] The usual approach around this problem is to take a pseudo inverse using singular value decomposition. Unfortunately, non-zero but small eigenvalues cause numerical instability in practice. This is especially a problem for Molecular Dynamics simulations because of the large dimensional space needed to describe protein motions, and thus some pre-processing must be performed before LDA or QDA can be used for analysis.

The solution used in this work is to apply a filtering algorithm to the covariance matrix before taking its inverse. The algorithm takes a spectral decomposition of the covariance and isolates the smallest eigenvalues.[48] It takes a specified cutoff which is a percent of the total trace, usually no more than 20 percent, and replaces the set of smallest eigenvalues which constitute percent of the total trace with their average. This effectively removes the null space so that no eigenvalue is numerically close to zero while preserving the total variance in the data. Another important aspect of this filtering is that it does not remove any directions from the data, it just replaces the null space with a degenerate space. These two features allow the covariance matrix to retain as much of its original information as possible after filtering without resorting to artificial shrinkage methods such as fully removing the small eigenvalue subspace.[48] After filtering, the inverse of the covariance matrix exists because the null space is removed and the determinant is no longer 0. After filtering, the effect of collinearity and numerical instability is greatly reduced and we are free to use LDA and QDA.

CHAPTER 4: CARTESIAN PCA

First we analyze cPCA on results for multiple beta-lactamase datasets. We performed this PCA on the entire set of 263 residues in what we call the global scheme, as well as the subsets of residues with biological importance and across the various pooling schemes. We pooled the data by crystal structure ("protein pooling") meaning that all 10000 frames for each of the 24 MD runs was collected into a single trajectory, by sequence ("TEM pooling") meaning that all 80000 frames of each sequence was collected into a single trajectory, and by set ("set pooling") meaning that all 48000 frames consisting of 2000 frame sets of each of the 24 runs was collected into a single trajectory. This allows us to compare how the proteins differentiate in their internal dynamics as well as check to see the consistency of dynamics throughout the whole MD run and across the starting structures. In addition we broke the TEM pooling up by set also giving 5 sets of 16000 frames per sequence, and we also pooled all 240000 frames of data together.

Table 4.1: Summary of pooling schemes used

Pooling Name	Description	# of frames per Pool
Protein Pooling	Pools all frames of same crystal structure AND sequence	10000
TEM Pooling	Pools all frames of same sequence (all crystal structures)	80000
Set Pooling	Pools all trajectories together in increments of 2000 frames	48000

Before analyzing the PCA, we can look at the root mean square fluctuation (RMSF) across each protein, presented in Figure 4.1. RMSF is a measure of how atom ("residue" for the sake of the coarse graining done in this work) fluctuates across the whole simulation.[32]

$$RMSF = \sqrt{\frac{1}{M}\sum_{i=1}^{M}(C_i^{q,x} - < C_i^{q,x} >)^2 + (C_i^{q,y} - < C_i^{q,y} >)^2 + (C_i^{q,z} - < C_i^{q,z} >)^2} \tag{4.1}$$

$C_i^{q,x}$ would refer to the x component of the qth alpha-carbon atom in the protein and M is the number of conformations in the simulation. RMSF lets us look at how a particular residue varies across the simulation. Residues on the protein which are more variant have greater mobility could possibly indicate places of interest. These resides might inform on functionally relevant sub-sests of residues for analysis. The RMSFs for the simulations are shown in figure 4.1. From the RMSF we can see that certain sets of residues along the sequence are in general more mobile. One observation made by comparing the RMSF between the different sequences for the same starting structure is that there is not an absolute trend across starting structures in which we can say, for example, that TEM-1 has a higher RMSF than TEM-52. Rather, there is not consistency in rank ordering of the RMSF. As an example in the 1ERQ run TEM-1 has a large spike around residue 156 which over shadows TEM-2 and TEM-52, in the 1XPB run TEM-2 spikes, and in the 1JWP run the TEM-52 spikes.

Aside from these magnitude fluctuations the overall shape of each curve is relatively constant. In some of the simulations two large spikes appear in the TEM-52 RMSF around residues Glu239 and Asp252. This could be related to the mutation at the 238 location. Certainly the Glu239 may be affected as it neighbors the mutation site, however the spikes are not consistent across all the runs and this would explain neither the increased RMSF of Asp252 which is not near any of the mutations nor the increase in TEM-2. When averaging the RMSF across all runs by sequence (shown in Figure 4.2) results are more similar. TEM-1 has a slightly lower RMSF than the other two mutants over most of the sequence. The spikes at 239 and 252 do appear for TEM-2 and TEM-52 on the averaged RMSF plot, and it constitutes the most drastic difference between the curves. Unfortunately discrimination by just RMSF is not possible, however differences in RMSF will be used as the basis of further analysis later in this chapter. It is interesting that TEM-2, which is functionally similar to TEM-1, follows the TEM-52 RMSF more often than TEM-1, especially at the large spikes.

4.1 Global PCA

First we performed cPCA on the global set of 263 residues for all sequences. We looked at the covariance plots to see if we could find any defining features. We started by considering the reduced covariance matrices, that combine the x,y,z covariance of each residue into a single residue level covariance, because of TEM pooling. The average reduced covariance matrix per sequence is presented in Figure 4.3. We note first in 4.3(a) that the TEM-1 covariance matrix is sparser than the other two indicating that there is less correlated motions between the residues than in the mutants. In all three plots we see a stripe of anti-covariance around residue 220 which separates the molecule into two regions which confine correlated motions within themselves. The block in the upper left exhibits motion that is anti-correlated to the rest of the protein, accentuated by the darker red areas on the plots. The anti-covariance is less pronounced in TEM-1 sequence. Comparing different sequences per starting structure yields a similar pattern.

Next we quantify how the variance is distributed in the essential subspace. Figure 4.4(a) shows all of the individual scree plots from the 24 protein pooling trajectories, and in (b) the average scree plot of each sequence. Overall the TEM-52 sequence runs have greater variance in their first modes, however there is quite a bit of mixing between TEM-1 and TEM-52. TEM-2 is in the middle and its first eigenvalue varies across the full range between the highest TEM-52 and the lowest TEM-1 top eigenvalue. The closeness of all of these scree plots is demonstrated by taking the average over all the individual protein scree plots. Each sequence's average scree plot has similar behavior, although the first mode of TEM-52 is somewhat higher. These average scree plots can be compared with the scree plots in Figure 4.4(c) which were derived from TEM pooling. This pooling average is similar to averaging the dynamics across each MD run individually. We have also shown in (d) the scree plots resulting from pooling the sets of 2000 frames together for all 24 runs to investigate the stability of the dynamics over the dataset. We expect for well equilibrated MD data that the scree, and thus the variance, to be similar across all of the frames. Sets 1, 2, and 3 cluster into similar variance as do sets 4 and 5, however they do not cluster with each other. The trace of the covariance matrix, which is also equal to the total variance captured in the molecular motions of the simulation for the pooled sets 1 through 5 are 179.8625, 171.8401, 177.1466, 200.1359, and 213.1509 respectively, indicating that on average there was an increase in global mobility toward the end of the simulations. However, the RMSD plots indicate that this can difference is attributed to a few outlier runs. Just using variance and scree plots alone we are unable to distinguish the different sequences.

Next we looked at fuzzball plots to see how the various sequences distribute themselves along their PC modes. Here we only present the first two PC modes, which is typical in PCA, because they should show the most of the original data's features as they contain the largest amount of variance. Lower PCA modes, while being significant tend not to provide good low dimensional visualization for comparing data as the PC distributions become tighter and more centered on 0 as we go further into the subspace. When comparing fuzzballs it is important to make sure the data is properly aligned to the same reference structure and that all the data to be compared is being projected into the same set of eigenvectors. For this reason the pooled datasets had PCA performed on them to extract the essential subspaces and the coordinates were projected manually in MATLAB.

First we pooled all the data, all sequences for all crystal structures, into one trajectory and split it into 5 segments representing sets of 2000 frames of each MD run. In Figure 4.5 the fuzzball plots are shown for Set 1. In Figure 4.5(a), it is easy to see there is not discrimination between sequence as the different colored fuzzballs are scattered on top of each other. In (b) the figure shows the same fuzzball colored by the different crystal structures. In this case there was no distinguishing features coming from the starting structure either. With all crystal structures and sequences there was considerable overlap between the MD simulations. This indicates that the configurational space over all 24 simulations was fairly constrained and shared, however in later sets we saw the colors spread out in the PCA feature space. We posit that this spreading is due to differences in the starting crystal structure. We asked the question as to whether or not PCA was discriminating the sequence dynamics or just the differences in starting structure. For this reason we also consider each sequence on its own using the TEM pooling technique. We projected the data into its own subspace to see if all the fuzzballs clustered indicating that all simulations for the same sequences were sharing essentially the same major dynamics. The three fuzzballs, colored by crystal structure, are shown in Figure 4.6.

The largest average motions of all TEM-1 runs were fairly similar and the fuzzballs are essentially centered on each other, although all the individual structures cover a wide range along the PC mode. In contrast the TEM-2 and TEM-52 have more separable crystal structure fuzzballs. In both there seem to be two regions, one in which most of the initial crystal structures cluster, and another region to the right in which one or two structures are protruding from the main cluster. Another interesting feature is that the main clustering of the different crystal structures is in DVP 1 (spread in x axis) and they spread out in DVP 2 (spread in y axis). Because the first PC mode represents the most variance and from Figure 4.4 we can say that the motion represented by PC 2 was more different among the structures while they shared the most variant motion. All of this would suggest that the dynamics of each

TEM are fairly similar, however some of the crystal structures such as 1LHY (green) were outliers. While these structures showed dynamics that were differentiated from the rest of the simulations, this was only present in only these crystal structures and not consistent across sequences.

Finally, when considering the essential dynamics of the different sequences we looked at the RMSIP between subspaces. Starting with the subspaces describing each of the twenty four MD runs, Figure 4.7(a) amd (b) shows the RMSIP between all combinations of the 24 essential subspaces. It is difficult to distinguish between sequences when considering each run separately until 10 dimensional subspaces were compared. This was made easier to visualize by averaging all of the 8 by 8 blocks comparing all runs of 2 different sequences. This revealed that on average TEM-1 and TEM-2 (with wild type function) have more similar dynamics than either compared to TEM-52 (ESBL function). This suggests that some of the dynamics picked up in the entire subspace, although not discriminating in the first two modes as shown in the fuzzball plots, may be distinguished in the mutants and possibly functionally relevant. One issue that could be raised about a conclusion such as this is the low average overlap that TEM-52 proteins have with each other. In Figure 4.7(b) it can be see that this average RMSIP of TEM-52 with TEM-52 is lower than that of TEM-1 or TEM-2 with themselves, however it is still visibly larger than the average of TEM-52 with any other sequence.

We consider also the RMSIP from comparing the TEM pooling scheme. We broke the TEM pooled trajectories into 5 sets of 2000 frames and computed the essential subspace for each, similar to the set pooling, these are shown in Figure 4.7(c) and (d). Immediately we noted that comparing each individual set per sequence gave more consistent results than considering each MD run. The TEM-1 sets in particular were far more similar. The band of high RMSIP along the middle indicates that although each sequence had fairly consistent essential dynamics across all the simulations, the sets that were closer to each other temporally (i.e. Set 1 and 2 as opposed to Set 1 and 5) were also more similar. This indicates that the global dynamics drift over time, yet differences in the essential dynamics between the sequences is maintained. This can be seen even stronger in the bottom right panel which gives the average of each 5 by 5 box comparing one sequence to another. Unfortunately this only discriminates the sequences themselves and does not imply that the dynamics of TEM-1 and TEM-2, the functionally similar proteins, are similar.

While no discrimination could be obtained through analyzing fuzzball projections, differences in RMSIP across sequences are observed. This suggests that functional motions comprise of multiple PC modes. The RMSIP plots were often not able to be discriminated until the subspace contained at least 4 dimensions. By analyzing overlaps of single modes we found that often even between same sequence runs the overlap of the top eigenvectors in the essential subspaces were low, while the cumulative overlaps between the eigenvectors in the low dimensional subspaces remained high. If only the top two dimensions are considered in fuzzball plots then information contained by higher dimensional combinations of the modes is washed out. Ideally it would be nice to look at higher dimensional representations of fuzzball plots but that poses a visualization challenge, which leaves us to consider pairwise 2-D combinations or triplet 3-D combinations.

4.2 Subset cPCA

The lack of discrimination between mutations in global cPCA led to the selection of several subsets discussed above to to analyze with PCA. In global PCA, the collective motions of all the residues are being considered in the covariance, thus many motions which have no relevance to the functional motions of the protein are introducing noise which could drown out any functional signal. The subsets chosen were informed by biological inference and are reported in Table 4.2.

We also chose a subset of residues which was informed directly from differences in RMSF across the MD simulations. To do this we considered the average RMSF across all the runs (Figure 4.2) (including all crystal structures) and took the residues which had a difference in RMSF of larger than .25 angstroms between the average wild type TEM-1 and ESBL TEM-52. The differences are shown in Figure 4.8. This set contained 41 residues: 52, 53, 58, 99, 100, 102, 103, 104, 105, 106, 107, 108, 110, 128, 129, 130, 131, 158, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 196, 197, 215, 216, 228, 238, 239, 240, 241, 252, 253, 254. It was particularly interesting that three areas emerged from this analysis without any prior information. The omega loop residues, 167-177 were included, as well as the residues around the E104K and G238S mutations were identified. These mutations, as described above, are attributed in the literature as responsible for the extended spectrum resistance of the TEM-52 enzyme.[8] The M182T mutation which acts as a global stabilizer for the protein is notably missing from this set. This same analysis was performed for the RMSF for TEM-1 and TEM-2. Not surprising, many of the same residues were identified for TEM-2 and TEM-52, however there were a few residues not included in this set that was in the set from the TEM-1 to TEM-52 comparison. Finally, of the mechanistic residues (70, 73, 130, 166, and 234) only 130 was directly expressed in

Table 4.2: Summary of PCA subsets to be investigated

Subset Name	**Number of Residues**	**Residue Numbers**
Global	263	26-288
Active Site + Omega Loop	25	70, 73, 130, 131, 136, 163-178, 179, 233, 234
Functional Site (Active+1)	21	69, 70, 73, 129, 130, 131, 132, 135, 136, 137, 163, 164, 166, 167, 178, 179, 180, 232, 233, 234, 235
Mechanistic + Omega Loop	19	70, 73, 130, 163-178, 234
Omega Loop[17]	16	163 - 178
Active Site[17]	10	70, 73, 130, 131, 136, 164, 166, 179, 233, 234
Mechanistic Site[18][25]	5	70, 73, 130, 166, 234

this set. Residues 70, 73, and 234 were not in the set nor were any of their bordering residues, however the residues around 166 in the omega loop were.

Also in the TEM-52 set, residues 215, 216, 240, and 241 were included. It has also been suggested that the pushing of these areas out of the active site is important for the function of TEM-52.[21]

It was interesting to see that the mutation sites for extended spectrum activity appeared in both sets, while the mutation site for the Q39K mutation which takes TEM-1 to TEM-2 was not included. Like the M182T mutation on TEM-52, it has no bearing on the function of the enzyme. It might be thought that these types of mutations work cooperatively with the functional mutations to increase activity because the increase of fluctuations is in areas not near the mutations.[25] It is known that M182T thermodynamically stabilizes beta-lactamases structure but because Q39K is expressed without any functional mutations, its effect is to increase mobility in regions of functionally important residues or mutations.

4.2.1 Active Site and Omega Loop

First we consider the subset of the active site and omega loop together. In figure 4.9 we present the covariance matrices for each mutation. The most prominent difference between the mutations is that there is significantly more covariant motion in the TEM-2 and TEM-52 enzymes. The covariance matrices have been reordered such that the active site and omega loop residues are grouped together. Because of this, covariance within each area can be seen as well as the covariance between the areas too. Within the active site TEM-52 has strongly covariant motions between the primary Serine 70 residue and the 233-234 residue pair and this is the most significantly covariant motion in this region. Note that in all three mutants, the Serine 70 residue has a substantial anti-covariant motion with the active site region around residue 166 in the omega loop. Overall this shows that some communication between the Serine 70 and the other active site residue is important.

The covariance in the omega loop is not too different between the mutants except that the features of the TEM-2 and TEM-52 mutants are much more prominent. This may not indicate different motions themselves, but that the mobility of the residues are much larger. This conclusion would be consistent with the idea that the TEM-2 mutation increases mobility in the omega loop region. This would make sense especially for TEM-2 in the context of the literature as it is thought that the mutation increases the mobility of this loop.[25] The other interesting feature in the covariance matrices is the communication overall between the active site and omega loop. For the TEM-1 the motions between these two regions are nearly uncorrelated except for the 70-166 covariance. In contrast the two mutants have more structured motions between the Serine 70, Aspartic Acid 233, and Lysine 234 residues in the active site and many of the residues which make up the omega loop. This is more prominent in the ESBL mutants and indicates that this coordinated motion might be important for opening up the binding pocket which is capped by the omega loop.

Next we analyze how the variance distributes itself in the PCA modes. The scree plot for all three mutants from TEM pooling as well as for all the individual runs is shown in Figure 4.10. In the figure showing all the scree plots obtained form each protein pooling run there is a clear outlier in the data which corresponds to the 1LHY crystal structure which can be seen in Figure 4.10(b). Ignoring this outlier, an interesting difference in the variance can be seen between TEM-1 and TEM-52. The wild type has consistently lower variance in almost all of its higher PCA modes, which when zoomed in can be clearly seen in Figure 4.10(c). Interestingly TEM-2 acts like a wild card when being compared to the other two mutants, meaning it sometimes mimics characteristics of TEM-1 and sometimes of TEM-52. The total variance in the TEM-2 motions take more toward the TEM-52 and in the 10 dimensions the scree plot showed that both mutants had around 10 percent more of its total variance explained than the TEM-1. This is most likely due to the fact that the first eigenvalue of both is much higher. Although TEM-2 starts high, it has a much steeper scree than TEM-52 and quickly (within the first 2 modes) rejoins the TEM-1 lines. In this way it seems that TEM-2 has one motion which is starkly more variant than the wild type but then the rest of its collective motions are about the same scale. In Figure 4.10(a), the average scree was obtained from the TEM pools broken into sets, and error bars were found. From these error bars it can be seen that although the variance in the enzymes looks different, the difference is not statistically significant.

In coordination with these differences in the variance, the fuzz ball plots shown in Figure 4.11 show more interesting features. First we can look at the DVPs as colored by mutation. There is not a clear separation in the fuzz balls, however its interesting to note that the conformations for TEM-2 and TEM-52 spread out a lot more than TEM-1 in PC mode 1. The points seem to cover more of the same range along PC mode 2. Although the separation by mutation is not observed, by looking at the individual crystal structures we note that there is much more overlap between the fuzzballs. This indicates that by focusing on specific parts of the protein we reduced bias from the starting structures. The biologically relevant regions have motions that are more conserved across different structures, while all other motions masked relevant motions in the essential subspace of the global protein. This happens because motions related to the omega loop and active site are less mobile than other regions. This highlights a flaw in the PCA dogma which states that modes with the greatest variance are most relevant.

Because the PC loadings can be mapped directly back to the x-y-z coordinates, we can look at what are called the squared modes, which combine the displacements of each reside in all three directions into a single score to see which residues are taking part in each mode. The squared modes for these PCAs are shown in Figure 4.13. When the motions are decomposed by residue, we find that the motion described in PC 1 is primarily in the omega loop, although residues 233 and 234 also take part. This is not surprising and lends more evidence that the mutations allow more mobility to the omega loop, especially since the DVPs suggest that the TEM-1 is more constrained in this motion. It is also worth noting that the second mode seems to be most prevalent around the mechanistic site residues. Because the conformations explored this direction somewhat equally, this implies that this motion is fairly conserved regardless of the mutation.

Finally for this subset we can show how the RMSIP between the essential subspaces changes as the size of the essential subspace increase. As before we used the essential subspaces obtained from the TEM pooling scheme as they represent "averages" over all the crystal structures. Based on Cattel's criteria we should cut off the essential subspace around 3-5 dimensions, however we show the RMSIP for the dynamics at 3, 7, and 15 to illustrate how the subspace saturates. We can see from these plots that the differences in the subspaces are actually pretty small when considering even only 7 dimensions of motion. The TEM-1 and TEM-2 have a slight similarity, although in the further detailed analysis of the essential subspaces there was not a clear differentiation between the mutations.

4.2.2 RMSF Chosen Subset

In this section we present the results of PCA from the subset chosen by looking at differences in RMSFs of the MD simulations. This subset was chosen by analyzing the mobility along the backbone, and not informed by biology like the other subsets, thus it will presumably consist of the largest differences between the protein. We expect that if cPCA will find anything in the data set we have, then this set will be ideal because PCA is a variance based algorithm.

As before we consider how the residues are working together via the covariance, presented in Figure 4.16. Once again the TEM-1 (Figure 4.16(a)) has considerable less structured motion than the mutants. In the functionally similar TEM-2 (Figure 4.16(b)) covariant motions have been increased, but the most distinctive feature is the appearance of anti-covariance between the E104 mutation site which is not present in this sequence, the mechanistic S130 residue, and the latter half of the omega loop. Again it was surprising to find that the functional mutation sites from the TEM-52 mutant had such a high covariance with other residues in the TEM-2 enzymes. The increase in fluctuations of the Serine 130 might indicate that it plays a more important role in TEM-2 than in either TEM-1 or TEM-52 as the variance between it and the omega loop are less pronounced. The most distinctive feature in TEM-52's covariance (Figure 4.16(c)) for this subset is the increase in anti correlated motions between the latter residues around 238 to 250 with themselves and with the other functional mutated Lysine at residue 104.

The scree plots from this subset show that the variance in the fluctuations of these residues are statistically significant. For all the sets of TEM pooling the variance in the first eigenvalues for TEM-1 was very small, and the error in the first eigenvalues of neither mutant was large enough to overlap each other. Up to the first 4 modes there is no overlap in the errors on the TEM-1 vs either mutant. This gives convincing evidence that the major effects of the mutations is to introduce more mobility into the protein, especially around the mutation sites and some of the mechanistic residues. It is interesting to see that the individual runs which were pooled by crystal structure and TEM had more mixing than the combined runs. This might be again attributed to differences from the crystal structures.

The projections for this subset revealed additional insight. First we employed SET pooling. The fuzzball plots are shown in Figure 4.17. It was observed that the TEM-1 projections stayed in a relatively small area of the conformation space in the first DVP, while the TEM-2 and TEM-52 projections spread into a much larger area as the simulations progressed. By calculating the overlap between the top two overlaps across all sets, we can be confident that the top two essential motions maintain similarity for the entire simulation. It was interesting that between set 3 and set 4 the eigenvectors basically switched, the top eigenvector for the first three sets becoming the second eigenvector for the last two, and vice versa for the other eigenvector. However throughout the entire simulation the space defined by the top two eigenvectors stayed around 90 percent similar. By looking at the square modes across all the sets, we can attribute the motion which causes this discrimination to residues 215 (Lysine), 216 (Valine), 252 (Aspartic Acid), and 253 (Glycine), as both eigenvector 1 and 2 spike at these residues and are very low elsewhere.

We also looked at another method of making projections, because interesting results were found when all of the data was pooled. The modes when compared by overlaps were different than the ones specified in the set pooling, however, the squared modes show that the motions occur at the same residues. Because squared modes consider all the x-y-z motions at once while the overlaps treat them differently this would imply that it is the same 4 residues which are undergoing the motion, but in a different direction. The projections are shown in Figure 4.18. These projections are different than the ones from SET pooling in that the TEM-52 DVPs spread far into the conformational space. It was also interesting to see that in this case the TEM-1 has a more diverse configurational space than the TEM-2 mutant. We also show a three dimensional reconstruction of the data in Figure 4.18(d). This shows that the TEM-2 is certainly the most constrained even in the top three dimensions.

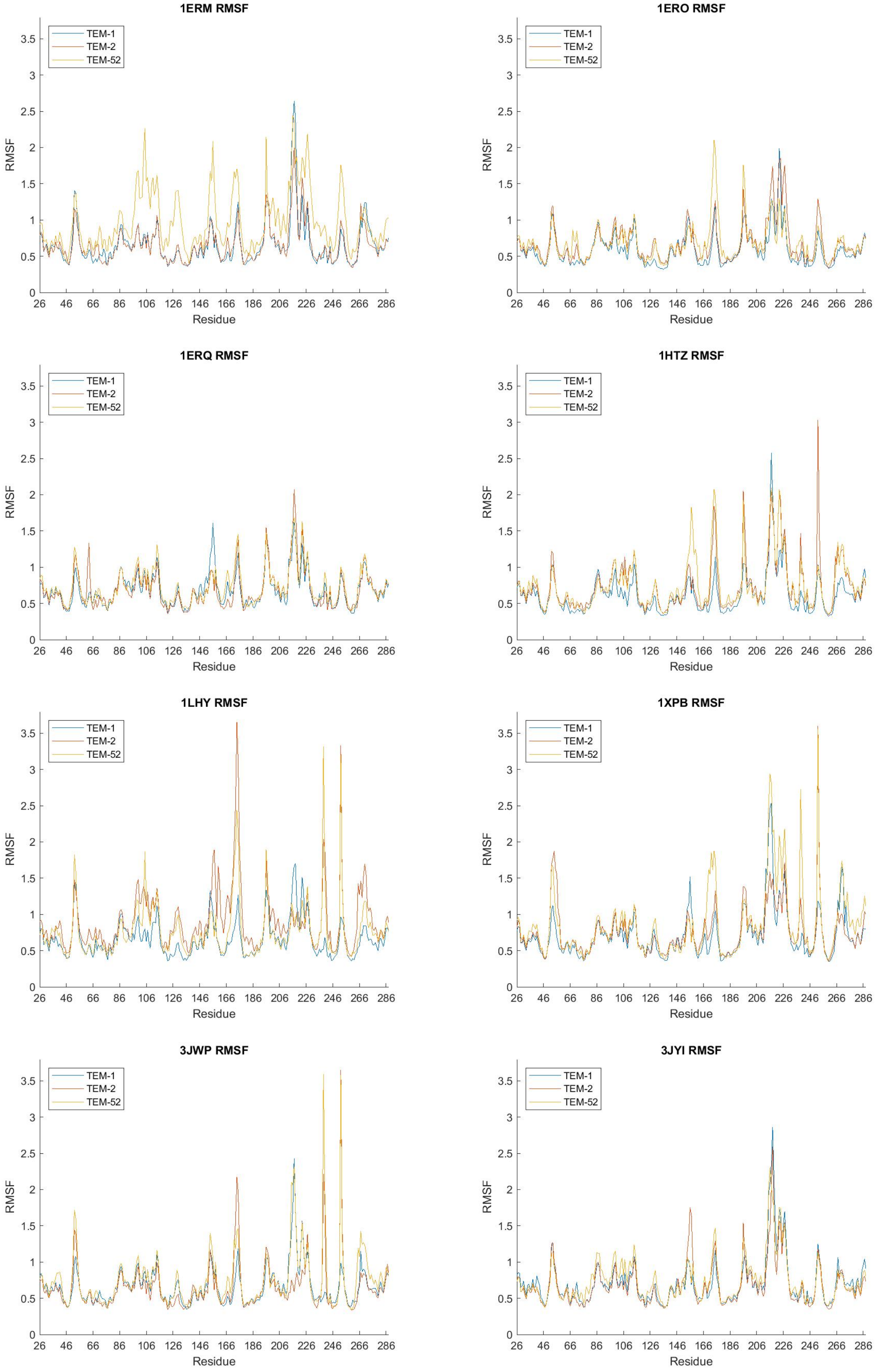


Figure 4.1: RMSFs for all 24 simulations.

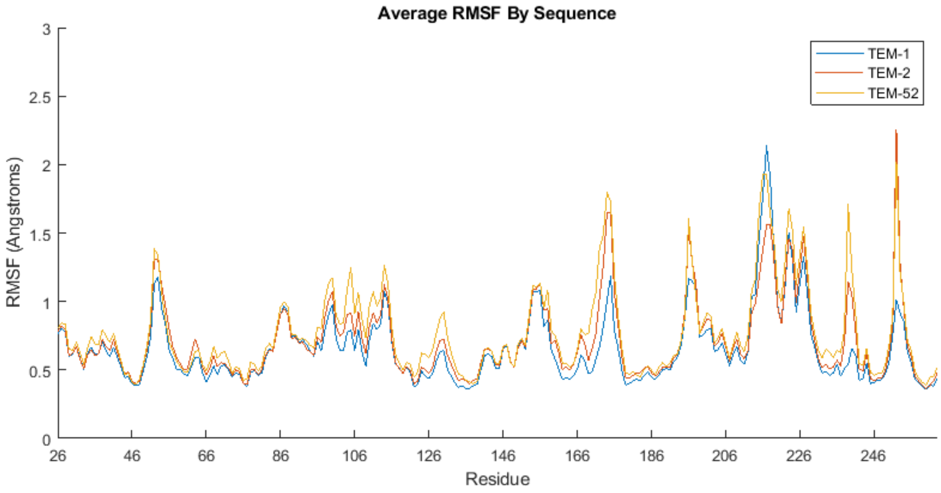
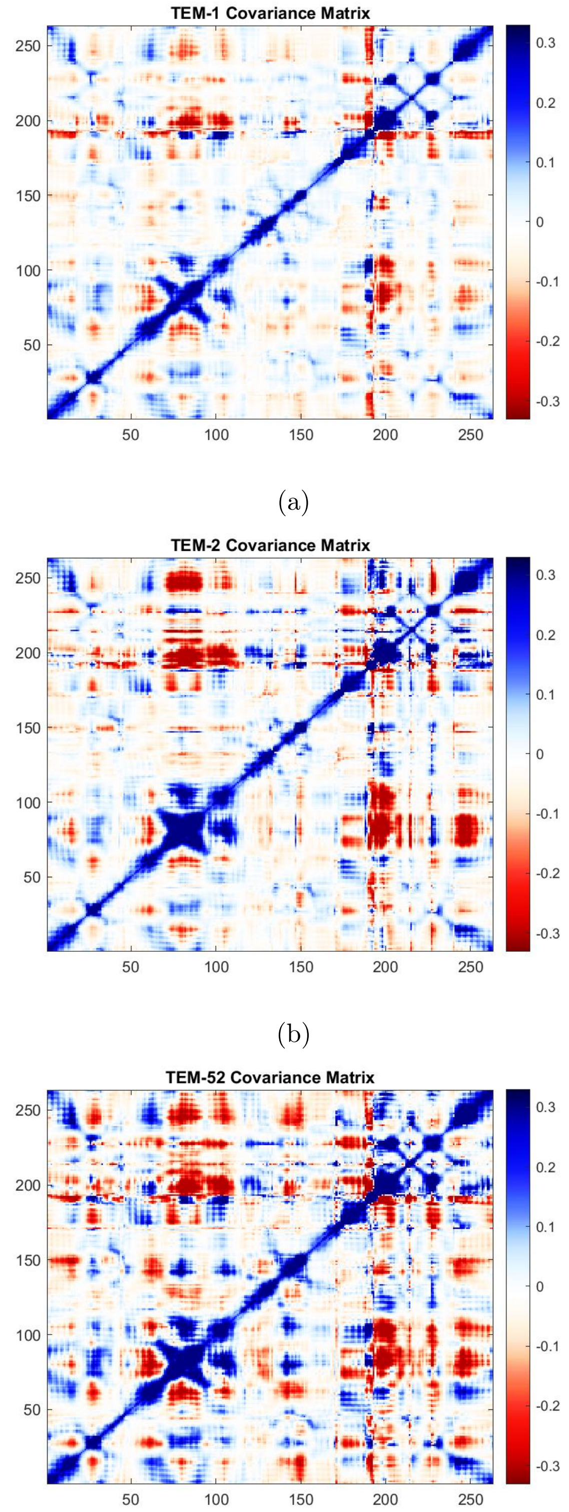


Figure 4.2: Average RMSF for each sequence

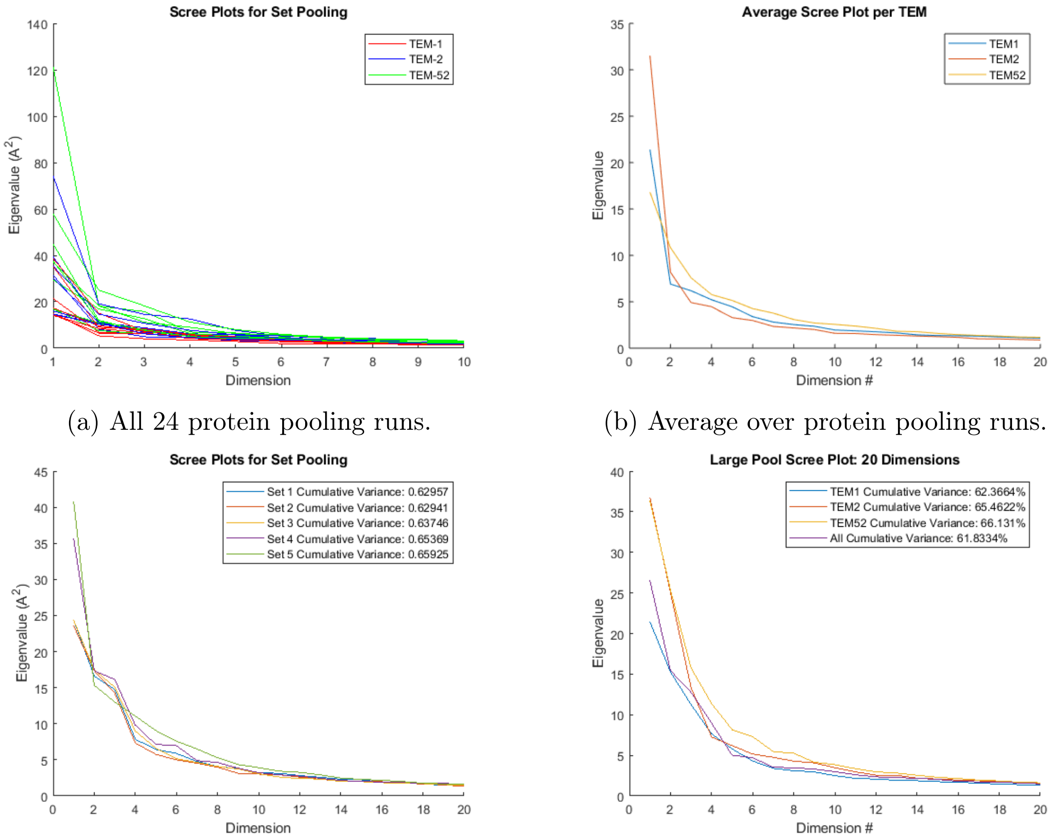


(a)

(b)

(c)

Figure 4.3: Covariance matrix of each for each sequence using TEM pooling.



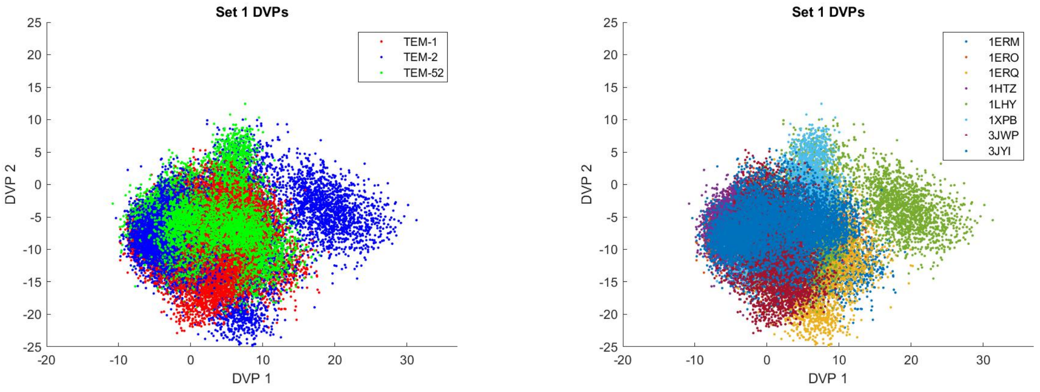
(a) All 24 protein pooling runs.

(b) Average over protein pooling runs.

(c) Set pooling.

(d) TEM pooling and zl data pooled.

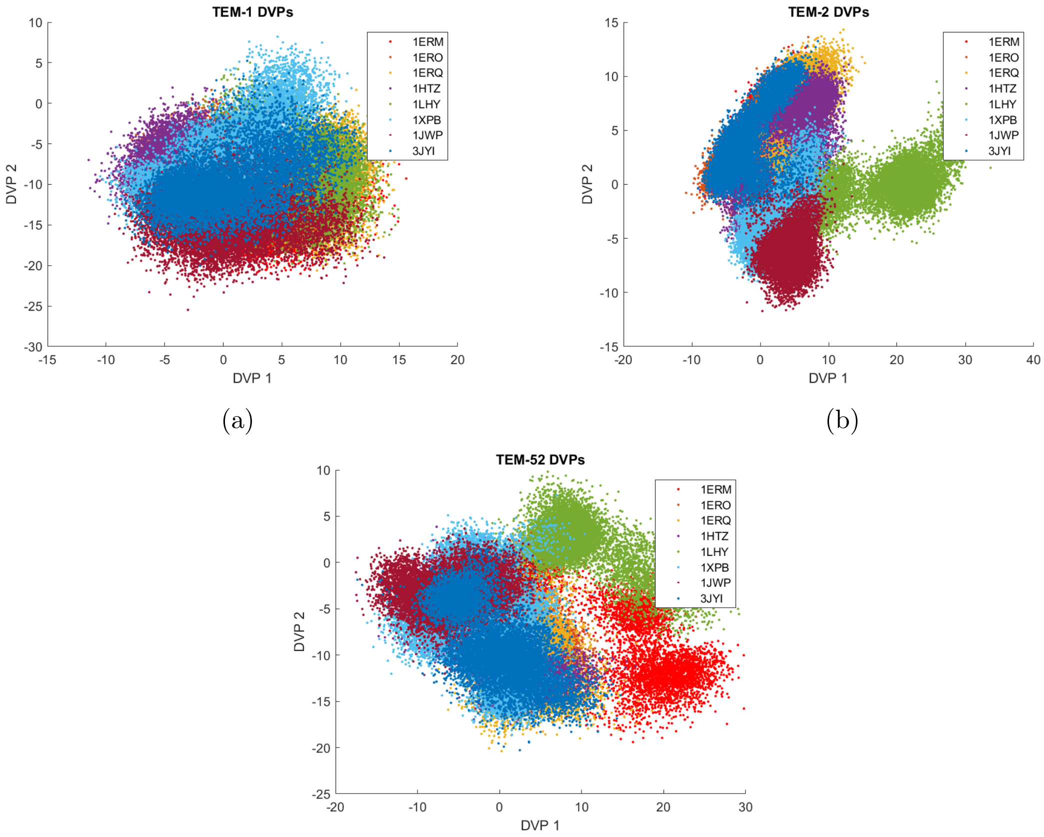
Figure 4.4: Scree plots for cPCA pooling schemes. Percent of total variance represented by scree is given in legend for (c) and (d)



(a) Coloring by Sequence.

(b) Coloring by Crystal Structure.

Figure 4.5: Fuzzball plots for the first 2000 frames of each simulation pooled together, projected into its own essential subspace.

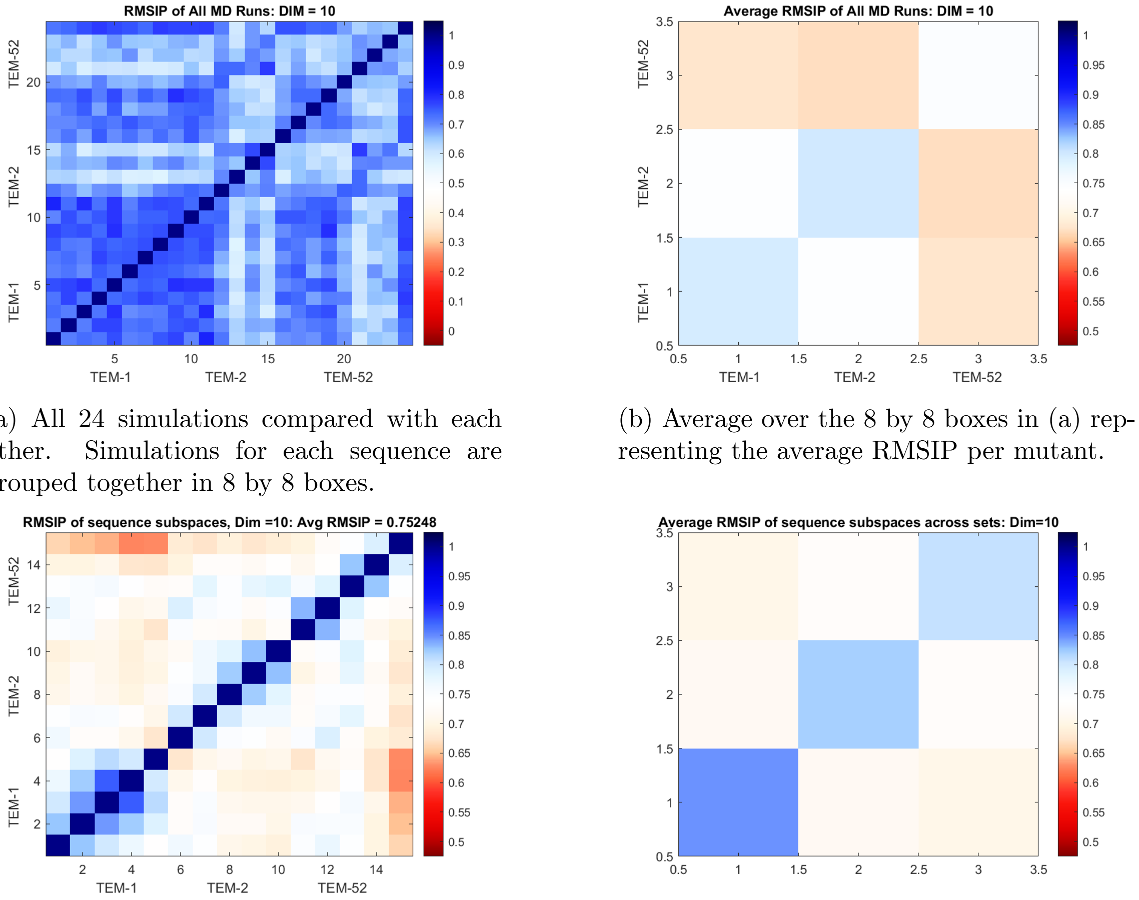


(a)

(b)

(c)

Figure 4.6: Fuzzballs of all MD for each sequence projected into its own essential subspace.



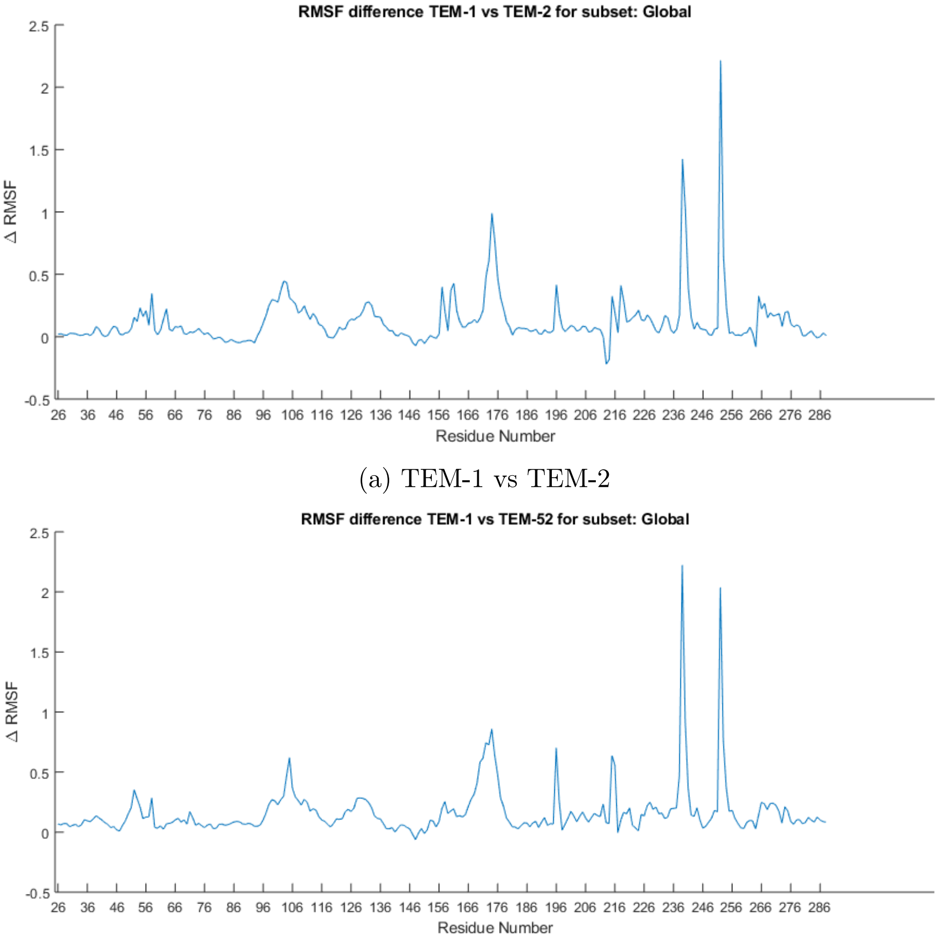
(a) All 24 simulations compared with each other. Simulations for each sequence are grouped together in 8 by 8 boxes.

(b) Average over the 8 by 8 boxes in (a) representing the average RMSIP per mutant.

(c) RMSIP of from all set pooling. All sets for the same sequences are grouped in 5 by 5 boxes.

(d) Average over the 5 by 5 boxes in (c) representing the average RMSIP per mutant.

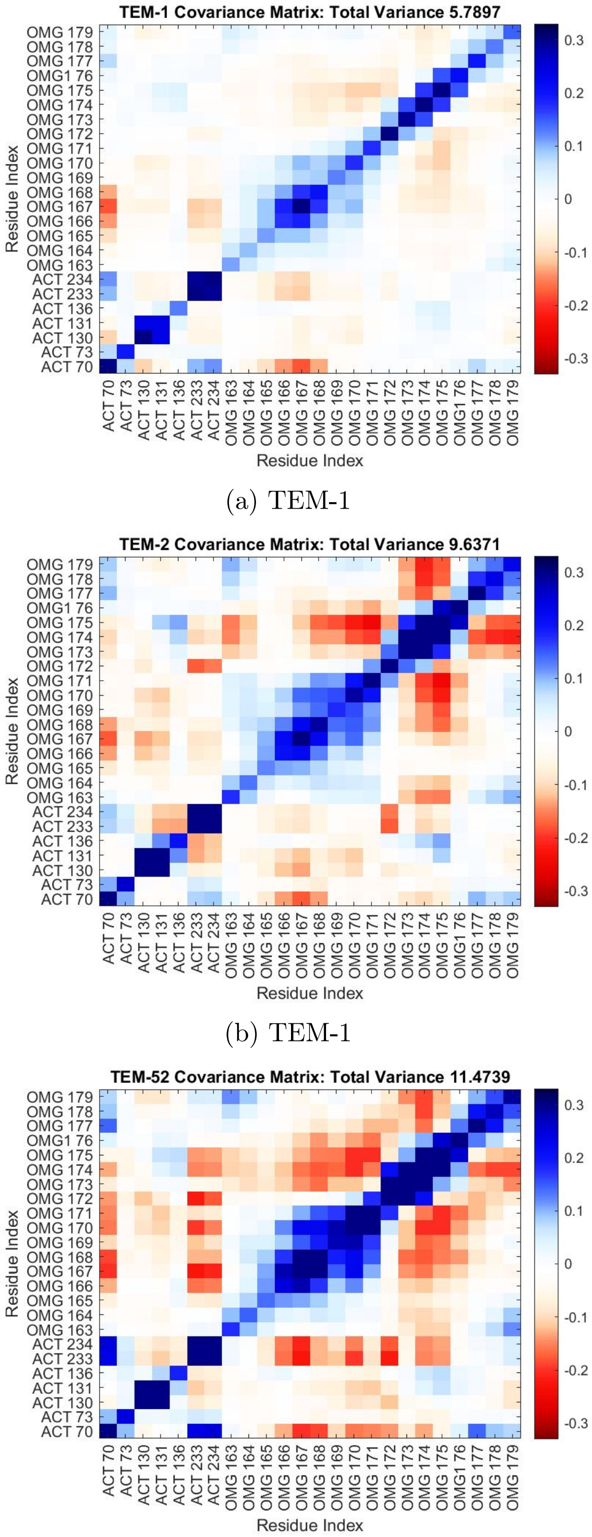
Figure 4.7: RMSIP plots protein dynamics captured in cPCA.



(a) TEM-1 vs TEM-2

(b) TEM-1 vs TEM-52

Figure 4.8: Differences in RMSF between TEM-1 and its mutants.

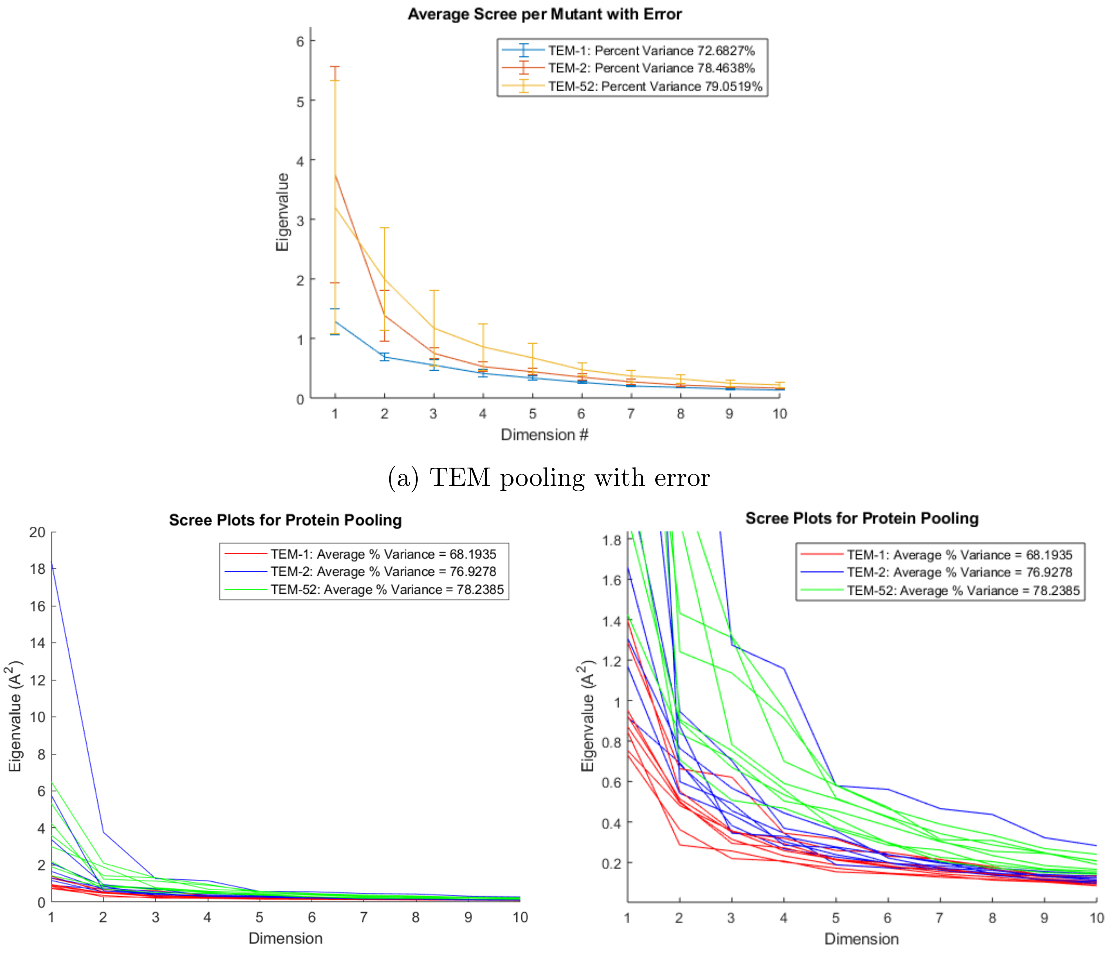


(a) TEM-1

(b) TEM-1

(c) TEM-1

Figure 4.9: Covariance matrix of each for each sequence using TEM pooling for the active site and omega loop subset.

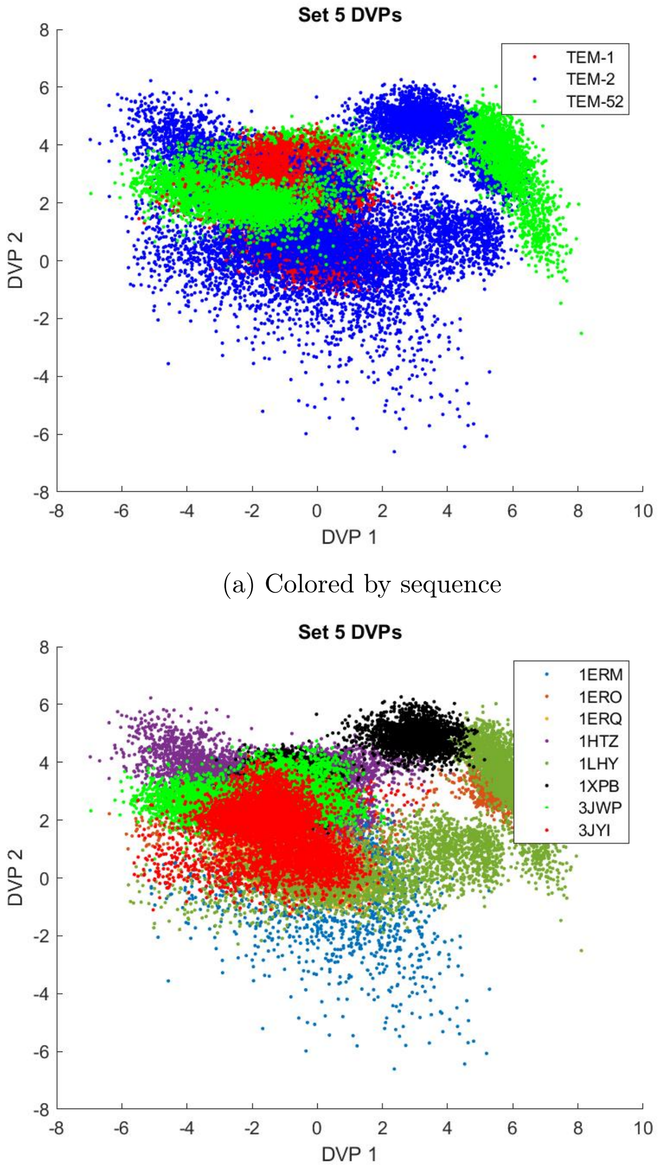


(a) TEM pooling with error

(b) PROT pooling for all sets.

(c) Zoomed in to Figure (b)

Figure 4.10: Scree plots for active site and omega loop cPCA.



(a) Colored by sequence

(b) Colored by crystal structure

Figure 4.11: DVP for the last 2000 frames of all MD runs.

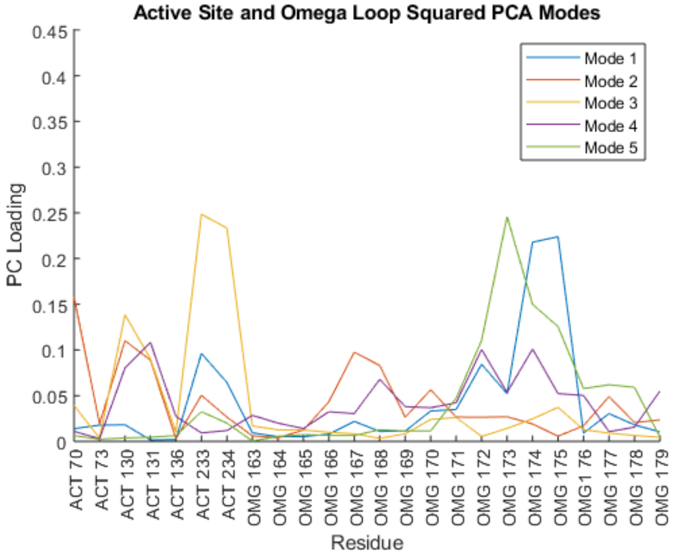


Figure 4.12: Square Modes showing the PC loadings for the DVPs presented in figure 4.11.

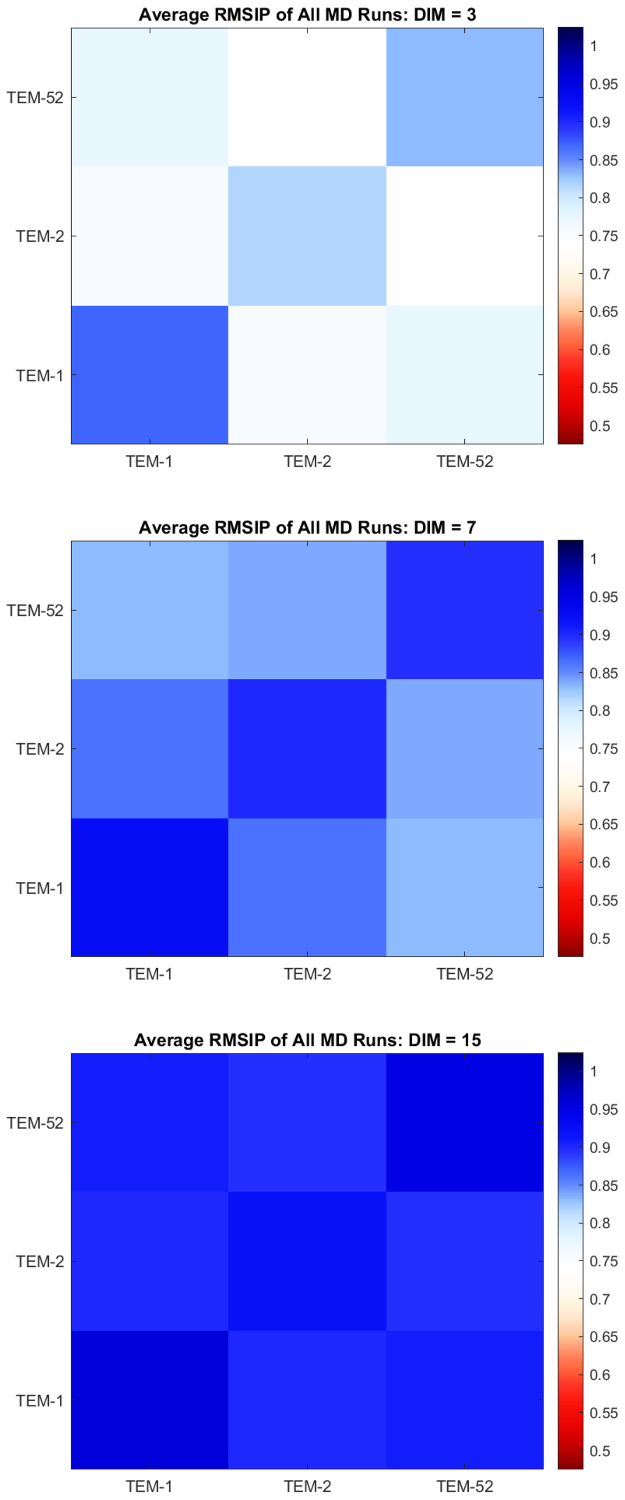


Figure 4.13: RMSIP between the essential subspaces obtained by breaking the TEM pooling into 5 sets of 2000 frames. RMSIP shown at 3, 7, and 15 dimensions.

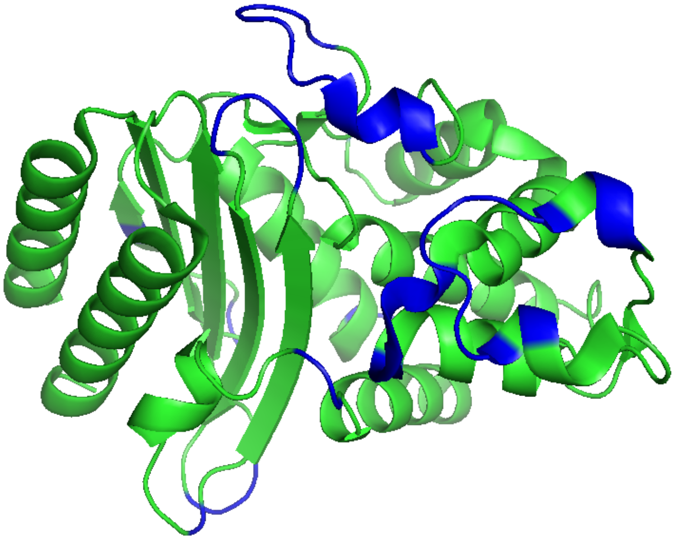
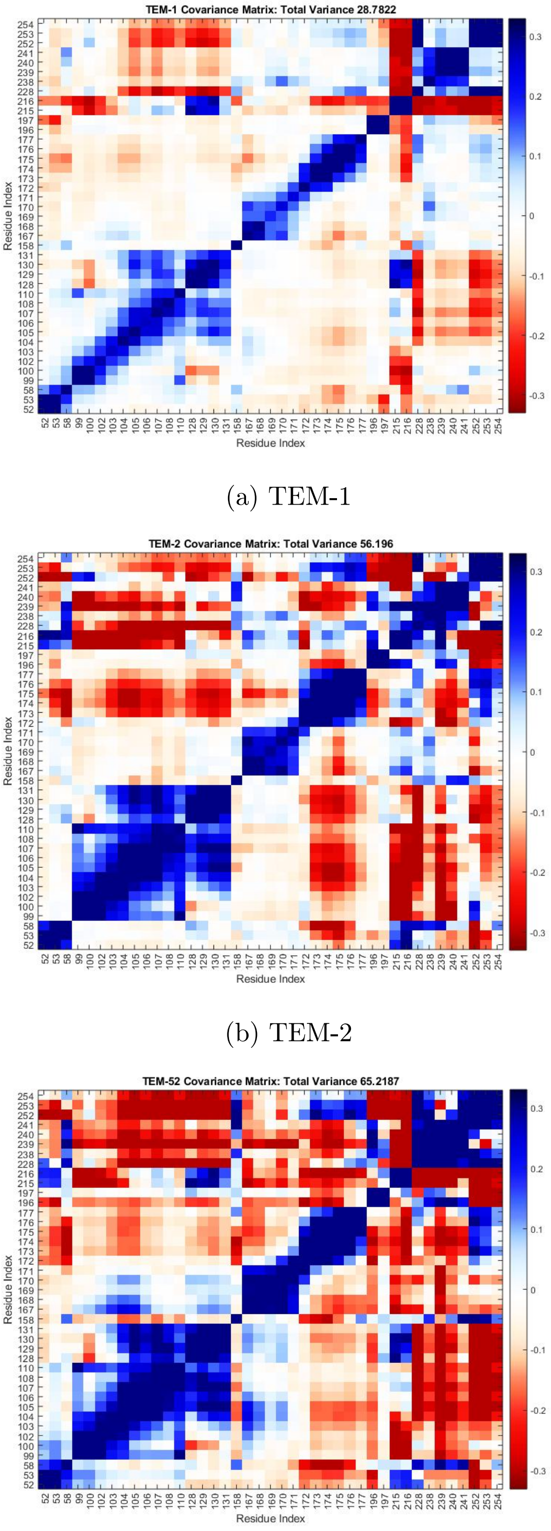


Figure 4.14: Beta-lactamase colored by the site identified by RMSF analysis in blue.

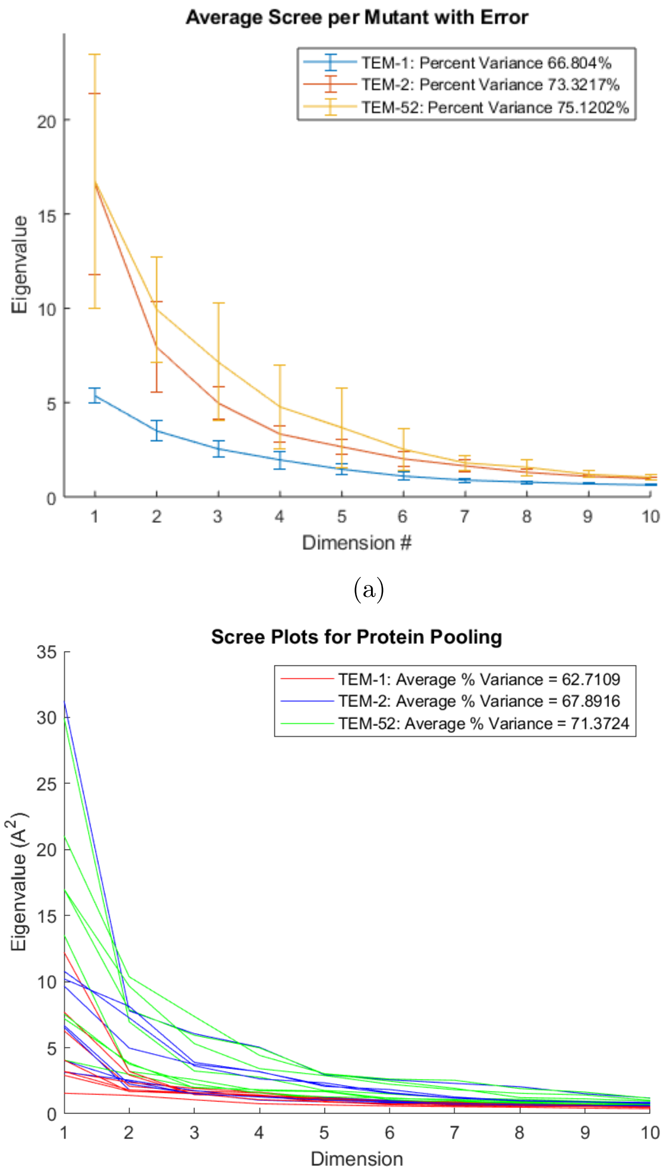


(a) TEM-1

(b) TEM-2

(c) TEM-52

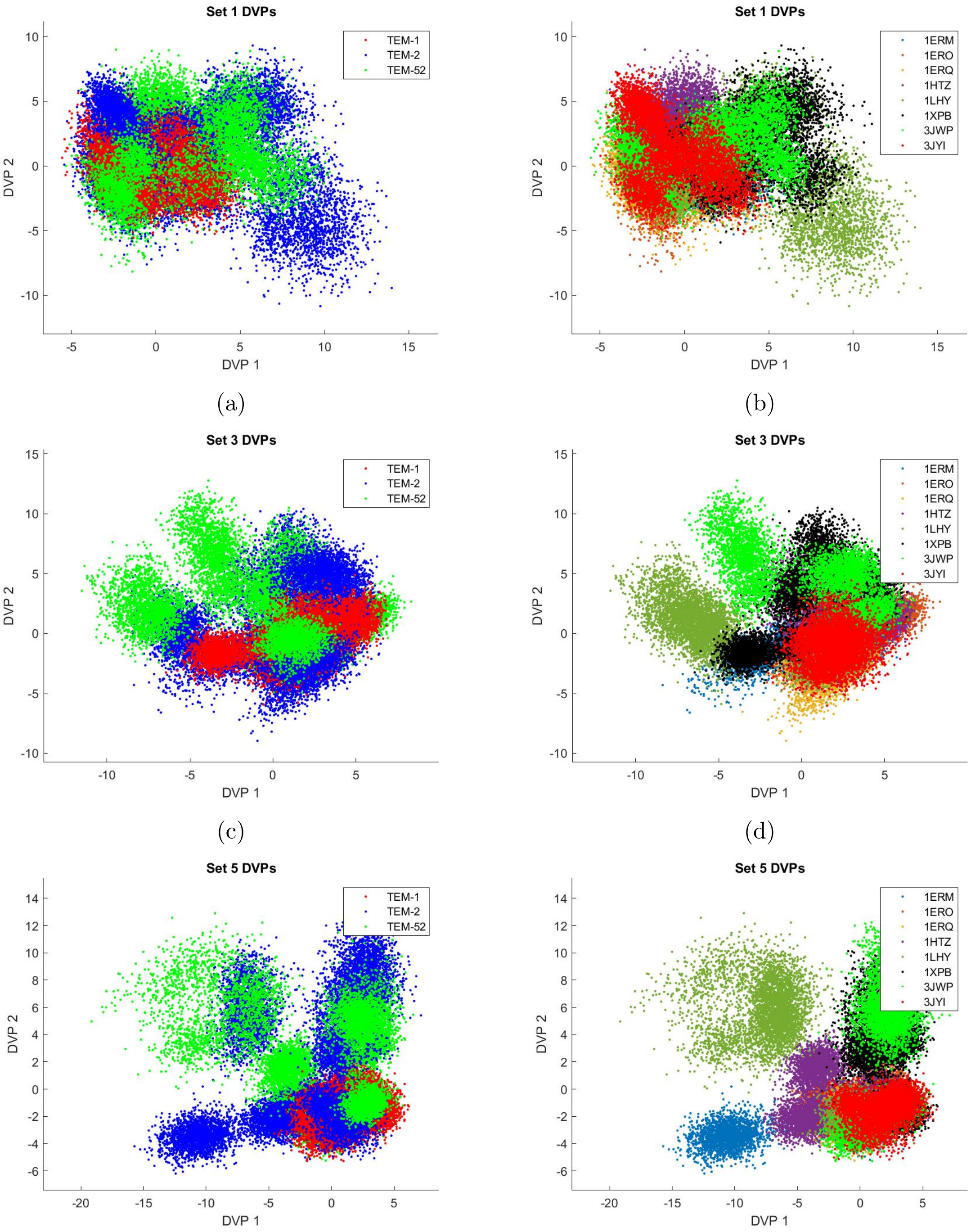
Figure 4.15: Covariance plots from RMSF subsets



(a)

(b)

Figure 4.16: Scree plots from RMSF subsets. Top is average scree form TEM pooling and bottom is all scree plots from PROT pooling.



(a)

(b)

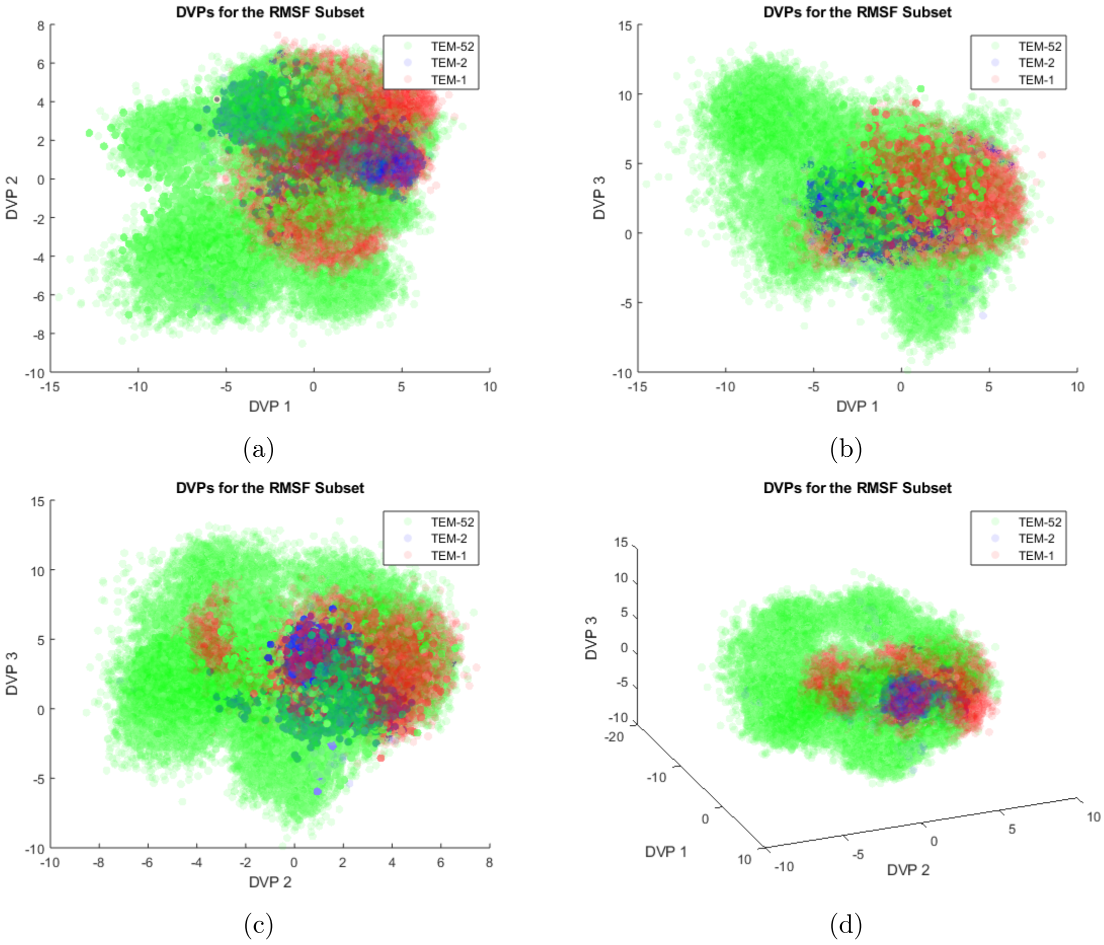
(c)

(d)

(e)

(f)

Figure 4.17: DVPs for RMSF subset. DVPs obtained through SET pooling, including all crystal structures. (a) and (b) are Set 1, (c) and (d) are Set 3, and (e) and (f) are Set 5.



(a)

(b)

(c)

(d)

Figure 4.18: DVPs for RMSF subset from pooling all the data together. DVPs are shown with faded dots so that all the projections can be seen. In (d) a 3-dimensional representation is shown

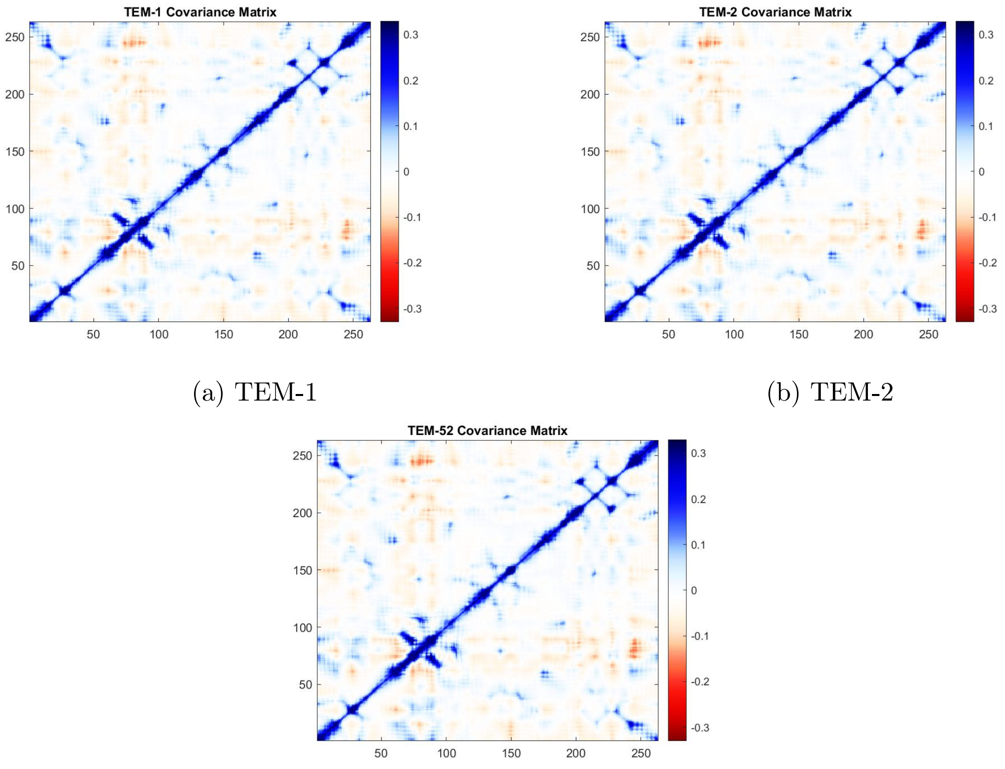
CHAPTER 5: DISPLACEMENT PCA

In this chapter we will present the results of our new PCA method, dispPCA. dispPCA uses as its input degrees of freedom, the successive displacements between atoms in the simulations instead of the Cartesian coordinates. Again, the motivation for this was to further remove coordinate bias from the essential motions.

5.1 Global dispPCA

In this section we present the results of displacement PCA on the entire beta-lactamase protein. In Figure 5.1 we show the covariance matrices for the displacement's covariance using TEM pooling. It turned out that all three covariance plots were extremely similar. This would imply that the introduction of point mutations did not drastically change the displacements, nor how the displacements affected the total dynamics of the protein. This is a little surprising seeing as cPCA showed that there are large differences in the variance in the proteins motions in coordinate space. In comparison to the covariance plots which resulted from cPCA in Figure 4.3 which showed significant correlations between the residues from only positional information, the dispPCA covariance matrices are very sparse, the displacements of residues were not as correlated to each other as the residues actual positions. The highly covariant band along the diagonal is to be expected as residues which are directly linked would be expected to move together, similar to what was observed in the omega loop region of the covariance matrix for cPCA.

These similarities in displacement information can be further inferred from the scree plots in Figure 5.2. In Figure 5.2(a) the scree plots from all 24 MD runs have been overlaid and colored by sequence. Unlike in cPCA the different sequences did



(a) TEM-1

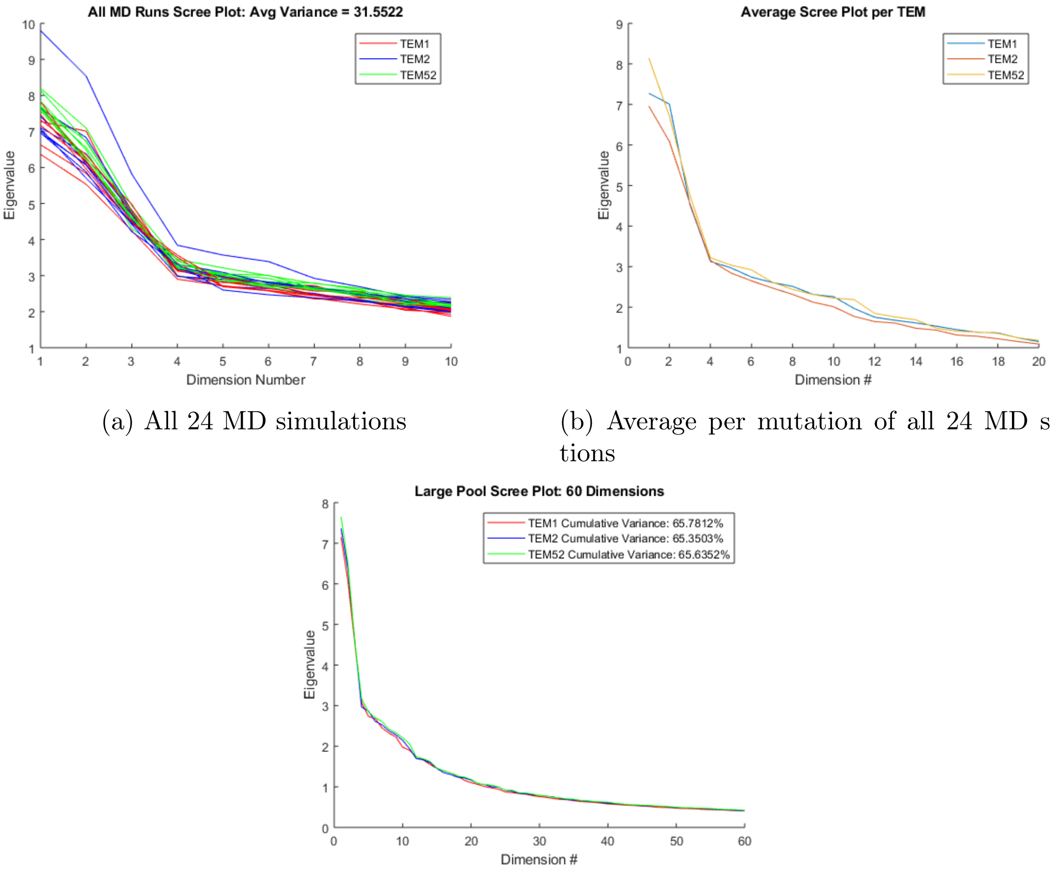
(b) TEM-2

(c) TEM-52

Figure 5.1: Sequence covariance matrices for displacements

not seem to distinguish themselves in terms of variance. In fact the average scree plot per sequence for all runs (Figure 5.2(b)) are almost identical. Different MD runs, as well as different sequences did not make much impact on the overall variance in the displacements. It was also interesting to note that the scree plots had a different overall shape than those from cPCA. In the top left figure one can compare the "scree" to the cPCA analogue and note that displacement scree falls off slower than coordinate scree. For this reason to reach the same percent of total variance as cPCA, dispPCA must take more modes, this can be seen in the scree resulting from TEM pooling (Figure 5.2(c)), which notes that for all three sequences to reach about 65 percent of the total variance 60 modes are needed in the essential subspace. This is drastically higher than 10-15 needed for cPCA.

To see if there was any discrimination within the PCA modes we turned to DVPs



(a) All 24 MD simulations

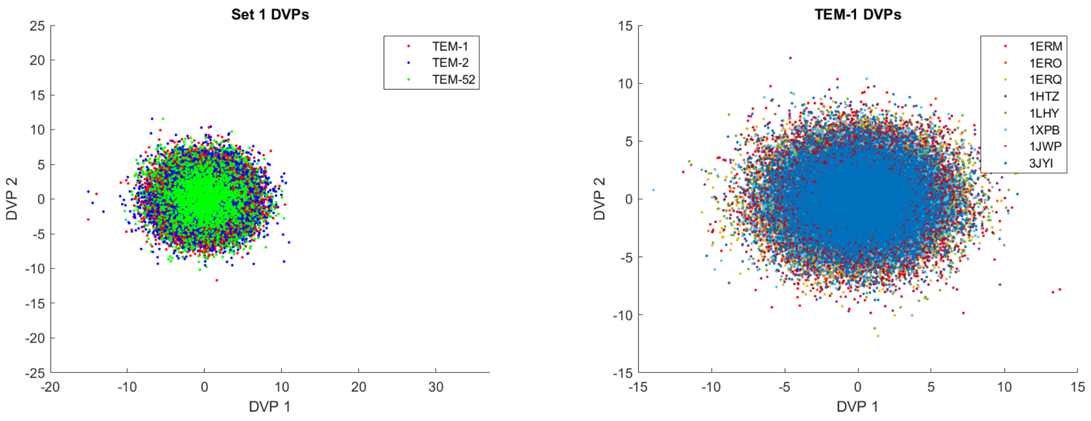
(b) Average per mutation of all 24 MD simulations

(c) TEM pooling scree plots

Figure 5.2: Scree plots for dispPCA results.

and RMSIP again. In all cases of pooling for fuzzball plots we saw the same result. The conformations distributed themselves fairly uniformly along their DVPs. Not only that, but all sequences and crystal structures appeared at the exact same place too. We could not discriminate between sequence, nor could we discriminate between crystal structure, as demonstrated in Figure 5.3.

Even without seeing discrimination, we looked at the squared modes for these displacements as with the cPCA DVPs. The squared modes are shown in Figure 5.4. As with the covariance matrices, one is hard pressed to find differences in the eigenvectors for dispPCA. The most variant displacement identified by mode 1 do not seem to highlight any specific residues, however some of the latter modes do. The only mode which seems any different between the mutations is actually mode



(a)

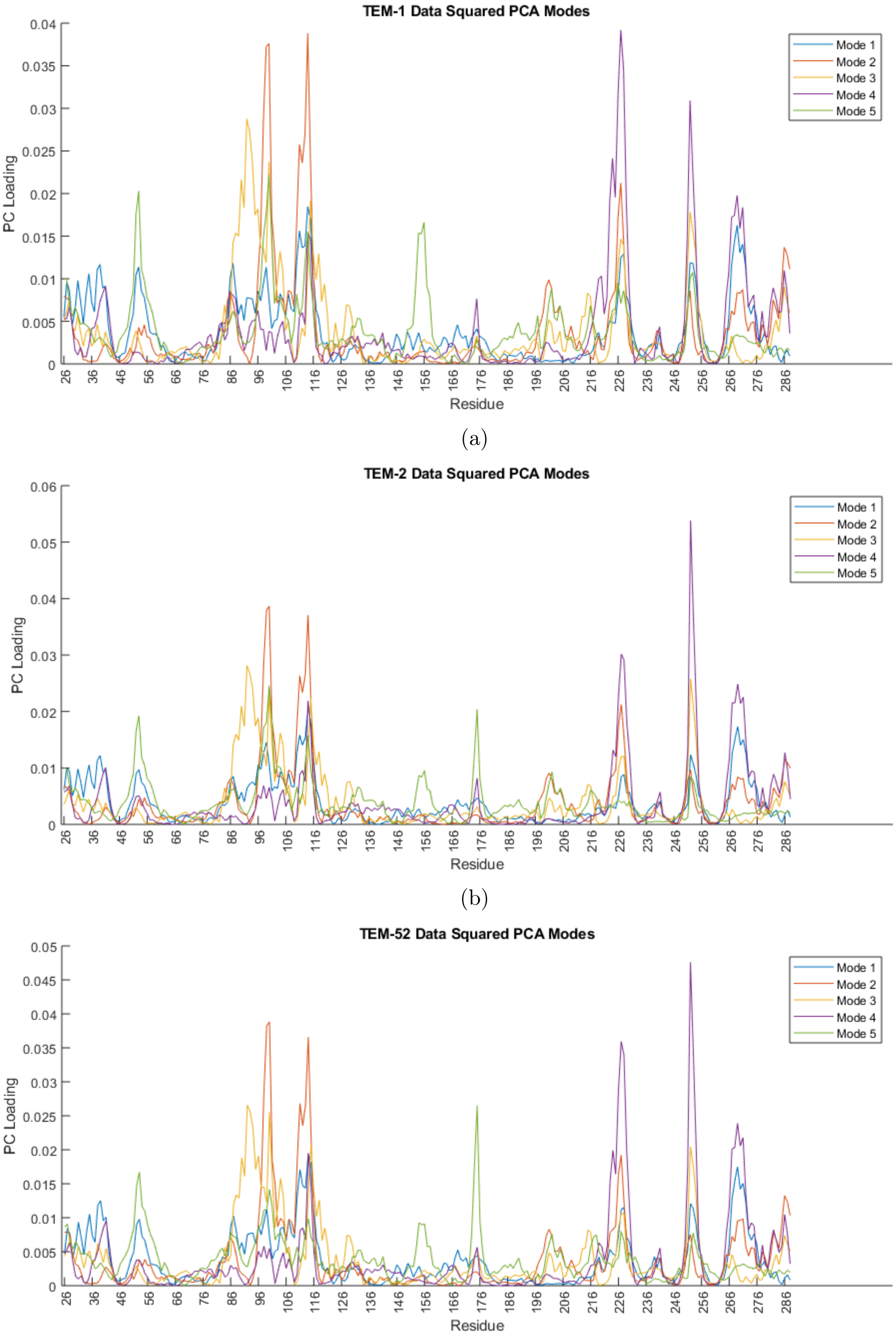
(b)

Figure 5.3: dispPCA Fuzzball Plots, (a) set 1 pooled colored by sequence, (b) TEM-1 pooled colored by crystal structure

5. Around residue 150 the eigenvectors seem to put a greater importance on this residue for TEM-1. Overwhelmingly, however when overlaps are considered, the first 3 modes are over 99 percent similar to each other. This indicates that the exact same motion, reguardless of the fluctuation in the squared modes, was pulled out by PCA. In addition to this we also looked for similarity of the dynamics identified thoughout the MD simulations. We did the same overlap analysis on the SET pooled eigenvectors and found also that all the sets agree with each other to greater than 99 percent similarity.

We repeated this analysis of dispPCA and the essential dynamics extracted from it with the subsets as with cPCA. For brevity only a summary of the results will be provided. By reducing the dynamics to that of a subset, we find that there is not any further discriminating features in the scree plot or the projections. The fuzzballs are all concentric and spherical as with the global DVPs shown above. The RMSIP for the essential subspaces does not dip below 97 percent and as such the essential dynamics can be considered the same.

This would seem to provide a null result, however this confirmed that the displacement information superseded structure information. This is an interesting de-



(a)

(b)

(c)

Figure 5.4: Squared modes form PCA eigenvectors. Top is TEM-1, middle is TEM-2, bottom is TEM-52.

velopment to essential dynamics analysis which at the moment still is limited to distinguishing structural differences, as can be seen in many of the cPCA fuzzball plots. Only in some subsets of residues using coordinate degrees of freedom do we see hints of the sequence dynamics presenting themselves. In future work a systematic study of the step size in displacement PCA should be explored.

CHAPTER 6: SUPERVISED LEARNING

In light of the results from PCA, in this section we explore the use of other clustering and classifications methods for identifying differences in the mutations of beta-lactamase. We have focused our efforts on two common methods for discriminant analysis and tested a new method that was developed specifically for identifying discriminating features across a set of molecular dynamics simulations, called supervised projective learning for orthogonal congruencies, or SPLOC. The S in SPLOC gives away the major difference between the methods of this chapter and PCA, that is that these methods employ supervised learning. This means that they take in class information about a training dataset to build a classifier. In this TEM-1 and TEM-2 are in the same WT-class, and TEM-52 is in the extended spectrum class.

6.1 LDA/QDA

Discriminant analysis (DA) can be a powerful tool for supervised classifications of data. As discussed in the methods chapter, PCA classifications are made based on variance. DA uses Fisher's criteria to maximize the ratio of between class to within class variance. The assumptions of DA is that the data is approximately Gaussian and that there is no collinearity.[49] To test the Gaussian assumption we looked at the distributions of the raw x-y-z coordinates from the simulations. The distributions were generated by a non parametric method which combines maximum entropy and maximum likelihood methods to estimate probability density functions.[50] By inspection most of the degrees of freedom looked appropriately Gaussian and the worst two cases have been shown in Figure 6.1. The most important characteristic we are looking for is that the data has a single mode and the probability for finding data far

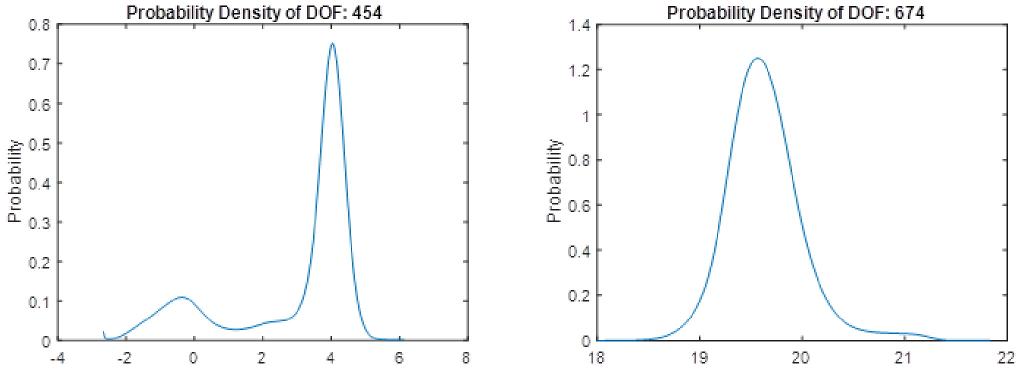
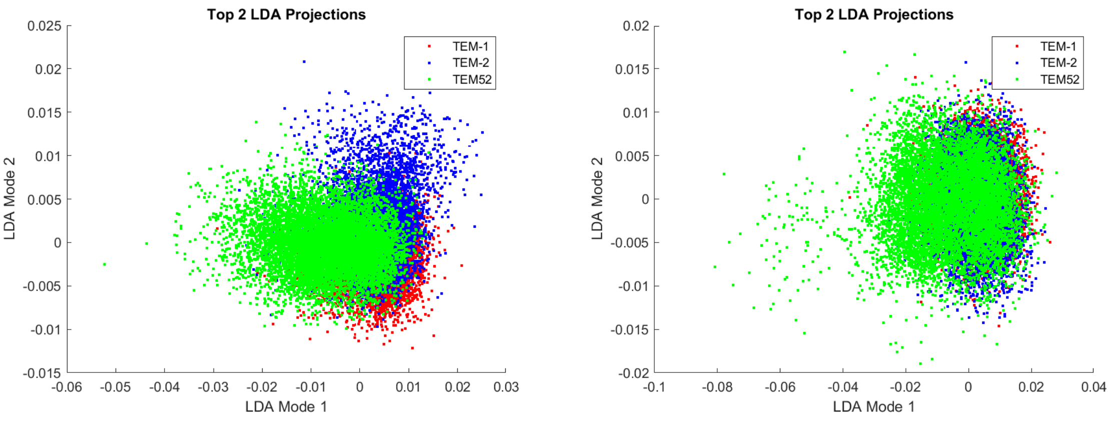


Figure 6.1: Probability density functions for two "bad" degrees of freedom. Notice that one is multi-modal, and the other has a small tail. Even these distributions would be considered fine for DA because they both fall to zero fast enough.

from the mean drops rapidly. In some cases multi-modal distributions are considered fine for DA.[49].

The second condition required some pre-processing of the data before classification. As previously explained in order for the data to not be collinear we had to apply a filtering algorithm to remove the small eigenvalues. The effect of this filtering on the classification statistics will be explored in this section.

Training and testing sets for the data were created by selecting random conformations from the MD trajectories. The selection had two schemes to help validate the accuracy of the classifier. First we pulled M_{Train} random conformation samples from each of the 8 crystal structure MD simulations for each mutant, resulting in $8 * M_{Train}$ samples for the training set, and then a smaller number M_{Test} from all the crystal structure simulations, all mutants is collected into a set of test conformations to classify. These sets are referred as "Full" training schemes because they have representatives from all of the starting structures. It is hoped that, like in PCA, by including all the structures in the all of the training sets we can average out all of the differences of the structures themselves so that what DA identifies is differences in internal dynamics. The second scheme that we use is the same process, but we choose 4 of the crystal structures at random and allow the training data to only come



(a)

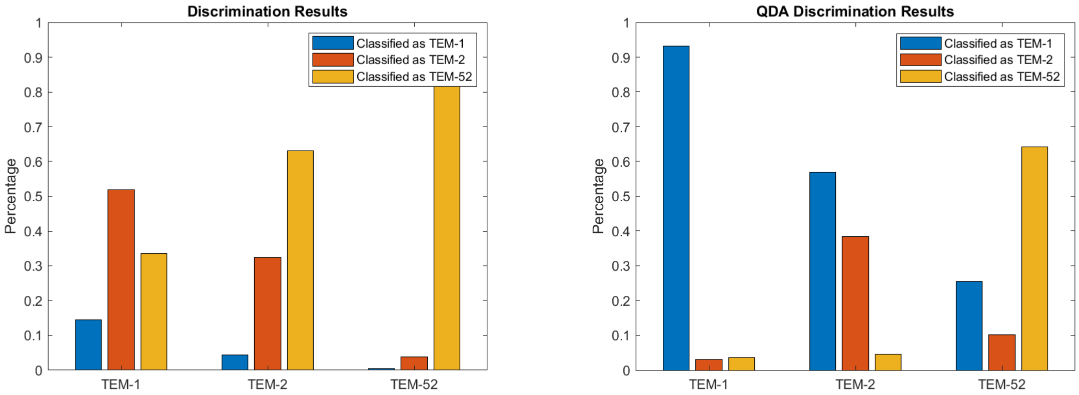
(b)

Figure 6.2: LDA fuzzball plots showing the first two modes. (a) full scheme (b) half scheme.

from their simulations. We call this the "Half" scheme. Then the test data is selected from the remaining structure MD simulations. In this way we investigated whether signal detected from the previous Full scheme was due to over-training on the specific crystal structures represented in the training set. For both of these schemes we attempted to keep the statistics per mutation as constant as possible by keeping the number of total samples in each training set constant. Doing this meant that for the half scheme we chose to take twice as many samples from each simulation chosen.

For simplicity and because we saw the most reduction of crystal structure information, we focused our efforts on the active site and omega loop regions. When we attempted LDA we looked at results from both the discriminant functions and the eigenvector formulation we found that discrimination was not clearly definitive. For the eigenvector method, in which we could get "discrimination modes" and fuzzball plots as in PCA, we used a training set of 2400 structures per mutation with 300 coming from each starting structure's MD simulation for the full scheme and 600 for the half scheme. Testing was done on 8000 structures which was split by mutation. For all of the cross validation sets used we saw that the TEM-1 and TEM-2 separated slightly from TEM-52, however the bulk of the distributions overlapped heavily implying that any discrimination would be weak. This weak discrimination was reflected in the eigenvalues which are proportional to the discrimination power of the modes. Only in one of the cross validation sets did the eigenvalues exceed 1, even in the top modes of each set. When comparing the two schemes, full and half, it was clear based on the fuzzball plots that the results of LDA was dependant on the starting structure of the simulations, as the discrimination between the wild type function enzymes and the ESBL enzyme was always in the top mode for the full scheme but was sometimes pushed to the second mode or not present at all for the half scheme. This does imply again that the differences in crystal structure play a larger role in determining the internal dynamics of a protein, however the slight discrimination implies that when all the structures are averaged over in the full scheme these differences can be removed. One quirk of LDA is that because it depends on a generalized eigenvalue problem, and that the matrices involved are not always symmetric, there is no condition which forces the eigenvectors to be symmetric, and thus discriminant subspaces are not created and cannot be directly compared.

LDA and QDA in the form of discriminant functions does not create eigenvector solutions, and so we have evaluated the use of these algorithms by attempting to classify the test data with the train functions. Because we know the classes of the test data beforehand we can quantify the percent of the test data which is correctly classified. When we applied LDA and QDA to the data sets, in both schemes we saw somewhat opposing results. The results for LDA are shown in Figure 6.3(a) and QDA in Figure 6.3(b). The QDA scoring function overall seemed to bias the TEM-1 and the LDA seemed to bias TEM-52. In the context of what the scoring functions do this makes some sense. In PCA we showed convincingly that one of the defining features of the TEM-52 dynamics is that there is more mobility in the functional motions, therefore the total variance in the coordinate data is also larger than the other two. We see that the main difference in the scoring function for these methods is that one



(a)

(b)

Figure 6.3: Classifications of MD simulation by discriminant analysis. Along the x axis the classifications are sorted in to three groups of three bars, each group representing all the structures which are actually classified by sequence, and each bar gives what the data was classified. (a) LDA, (b) QDA. Both with .05 cutoff.

takes into account the variance (QDA) while the other makes the assumption that the variance between the classes is equal (LDA). Because the means of the classes are close, as can clearly be seen in the PCA and LDA fuzzballs, it is easy for LDA to mistake the TEM-1 and TEM-2 structures as the TEM-52. Because of this LDA preferentially likes to classify structures as TEM-52. QDA on the other hand does take into account the variance in the data. Because the TEM-1 distributions are so tight, the scoring function tends to favor it in classifications.

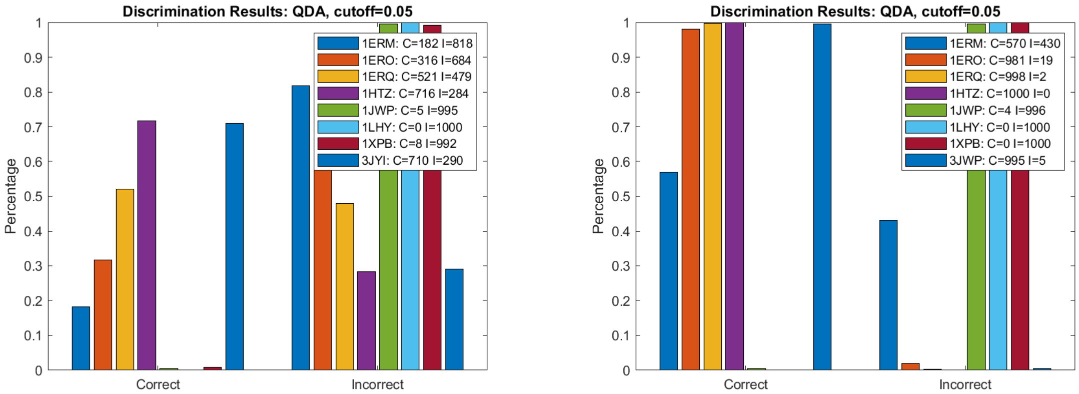
LDA seems to also be invariant to whether or not the full or half scheme is being used. This would be good news for its utility but its classification rate was not good across the four validation sets used. Most often LDA would do the best job identifying TEM-52 or TEM-2, although for larger sample size while training (up 12,000 structures per class) TEM-2's correct classification rate saturated at around 50 percent. This is better than random, however if LDA had completely eliminated TEM-1 as an option due to its significantly smaller variance then this is approximately random. According to the PCA predictions TEM-2 and TEM-52 are more variant and this would be in agreement.

In opposition to this is the results of QDA. QDA tended to favor TEM-1 and would most often classify TEM-2 as TEM-1. This is in agreement to the previously determined functional groups, wild type and ESBL. Recall that the reason for including TEM-2 in this data set was to act as a control group. In total TEM-1 was correctly identified in the test data around 90 percent of the time, while TEM-52 was correctly identified around 65 percent of the time. These results were invariant to raising and lowering the number of samples the models were trained on. Seeing as the wild type and the ESBL were correctly identified more than 60 percent of the time, and that TEM-2 was considered closer to the wild type QDA gave the best classification result. The only down side to this is that when moving from the full to the half scheme the classification rates go down. In two of the 4 validation sets the classification rate was still above 60 percent for both, however, QDA did not do as well over all with this training scheme. As with LDA this highlights that the dynamics associated with the starting structures for the MD simulations will often drown out motions associated with the sequence. Because of the large variability in the results for the half scheme, when we switch the random crystal structures chosen to be in the training set, we would hypothesize that some of the structures are more different than the others in term of dynamics.

The way that we have applied discriminant analysis with the filtering of the covariance matrix allowed for a hyper parameter to be introduced. This parameter controls how much of the trace of the covariance to replace with its mean. The reason for the introduction of this parameter is explained in the method sections, but it allows us to avoid the multicollinearity problem and invert the covariance matrix by removing the small eigenvalues. While adjusting this parameter in order to optimize the classifier we found that the results were almost independent of the choice of cutoff. It seems that it is only important to remove at least the smallest eigenvalues. The way to choose this is to consider the numerical accuracy of your eigenvalues and remove all eigenvalues which contribute less than that much, as you cannot be sure that the the numerical value of the the eigenvalue is correct.[48] However, because of the invariance of our classification results we chose to set the numerical cutoff at 5 percent of the total trace which would cut off around 40 out of the 72 eigenvalues and replace them with their mean. This was consistent across the all levels of training.

Although differences between TEM-1 and TEM-52 were not prominent, we can still say that due to our results with the full scheme there is something distinguishing between the wild type and ESBL sequence in DA. As with PCA we found that whatever the discriminating motions that QDA found were, they were not based fully on the differences in the covariance matrices. Because LDA favored the large large variance, and QDA favored the small variance, then there might exist some happy medium between the two methods. Because the scoring function only differs by the term of the determinate of the covariance matrix, one could imagine a method in which the effect of this could be controlled by a hyper parameter in the method. This type of approach already exists under the name Regularized Discriminant Analysis [51]. One direction of future work is to see how classification can be tuned by applying such algorithms in order to get the optimal classifier for wild type and ESBL beta-lactamase.

Finally we wanted to consider whether or not the MD data could be classified via the starting structure from their simulations. The reason for doing this was that in all the previous analysis we saw that the most different dynamics were primarily determined by the crystal structure it came from. This problem is termed crystal memory because it seems like the protein "remembers" its initial conformation throughout the simulation. As long as this crystal memory persists recovering the sequence dynamics which separates wild type from ESBL beta-lactamase enzymes will be challenging. In the same methodology as before we created 8 training sets from each of the 8 crystal structures which are represented in our data, and we sampled equally from all three



(a) Active site and omega loop

(b) Global protein

Figure 6.4: Classifications for MD conformations by starting structure. Classifications sorted by correct and incorrect along the x axis and the bars representing correct or incorrect classifications for a single crystal structure. (a) Active site and omega loop, (b) global protein.

mutations per training set. We chose to take 5000 samples per structure giving 15,000 total samples per training set, ensuring statistics were sufficient. We selected 1000 samples from the remaining data to be test set. We then performed QDA on these training sets, and computed classification counts on a set of test data.

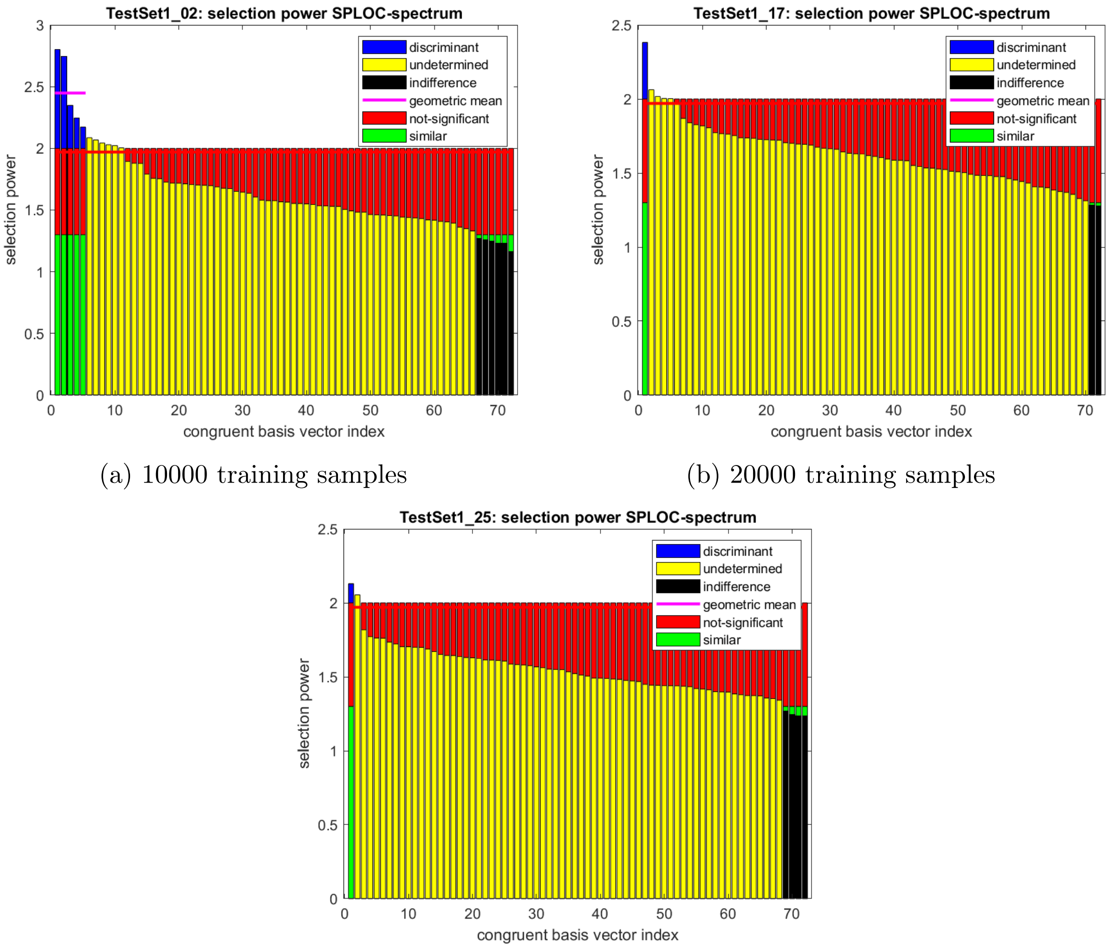
The results from this were partially as expected and partially surprising. They are shown in Figure 6.4, for the active site and omega loop (a) and the global protein (b). We found that for the global protein, the classifications were much higher, with almost 100 percent accuracy for four of the crystal structures, although the 1JWP, 1LHY, and 1XPB had zero correct classifications. 1ERM was the only structure to be in between, fluctuating around 50 percent accuracy across the validation sets. Although this is far from 100 percent, it is still about 4 times better than random guessing considering that there were 8 classes and thus is distinguishable in QDA. Overall QDA did a good job in distinguishing 5 out of the 8 crystal structures, showing conclusively that the crystal memory was preserved throughout most of the simulations. Importantly, when we went to the subset informed by biological inference the distinguishing characteristics of many of the crystal structures went away. The structure of these regions is more conserved across the whole protein, which is to be expected, and thus we can validate our earlier claim that by moving to subsets we reduce the effect of the structure dynamics.

6.2 SPLOC

In this section we present the results from a novel supervised learning method called SPLOC, Supervised Projective Learning for Orthogonal Congruencies. This program was developed outside of this project and as such the details of the algorithm will not be presented here, but will be presented in a future work. SPLOC takes labeled simulation data for a functional and nonfunctional system and extracts the most discriminating features between them. The algorithm takes some inspiration from QDA and neural networks, and uses an algorithm which takes in to account differences in the means and variances of the classes. Discrimination is confirmed in two ways to make sure that any signal found is significant. The first way involves making sure that the mode has sufficient signal beyond noise. The second way takes a "voting consensus" and cross validates that the mode has discrimination across all of the training data to achieve statistical significance. One of the major benefits to SPLOC is that it is an eigenvector based method like PCA and LDA so we can get discrimination modes which can be linked back to the original x-y-z degrees of freedom to allow us visualization. The PCA modes are often visualized as normal modes, however SPLOC filters the non-discriminate motions out of the original data via projections thus allowing direct visualization. SPLOC creates a complete set of eigenvectors, therefore the statistical subspace analysis is also usable. Finally, as an improvement on LDA/QDA, which looks only for discrimination between the classes, SPLOC will find collective modes which are conserved as well as discriminate allowing one to probe the similarities and differences. In order for a mode to be put into either the discriminate or indiscriminate subspaces it must pass both of the tests above. If it fails either one or both it is placed in an undetermined space. If SPLOC cannot find something definitively discriminate or indiscriminate in the training data, then all of the modes will be undetermined. The SPLOC package is distributed as a library of functions in MATLAB allowing the users to do a wide array of analysis of MD simulations.

For the subsequent analysis with SPLOC we have focused on looking at the active site and omega loop region. To make the training sets for SPLOC we broke all 24 500 nanosecond runs into 5 sets of 2000 random samples, giving a total of 120 2000 sample sets. These are randomly selected by mutation to make the training sets, and completely randomly selected to be in the test set for classification. We always made sure to keep the number of representatives in each training set as equal as possible and attempted to make sure that no crystal structure was over represented. We chose three levels of training, 10000, 20000, and 40000 samples, to test the consistency of the solutions with increase in training size. The only other adjustable parameter for SPLOC is called the voting threshold, which determines the minimum amount of agreement needed within the data set to show that a mode is discriminate or indiscriminate. We chose to leave this parameter at the default. The default voting consensus is dependent on the number of samples used to train, and was .615 for the sets with 10000 samples, .594 for 20000 samples, and .578 for 40000 samples.

To test the consistency of each training set we created an ensemble of 26 SPLOC runs. SPLOC solutions are not symmetric upon exchanging train labels. That is when inputting the training data one must be labeled the "Functional" and the other "Non-Functional" as the class separators, and the resulting answer is not the same if you flip these class names. This is because SPLOC looks at differences in probability distributions of the data, and it has an easier time identifying a tight distribution from a wide one than a wide distribution from a tight one. In the PCA analysis we saw extensively that TEM-1 had a much tighter distribution than TEM-52 and thus we chose to make it the functional data. Attempts to use TEM-52 as functional did



(a) 10000 training samples

(b) 20000 training samples

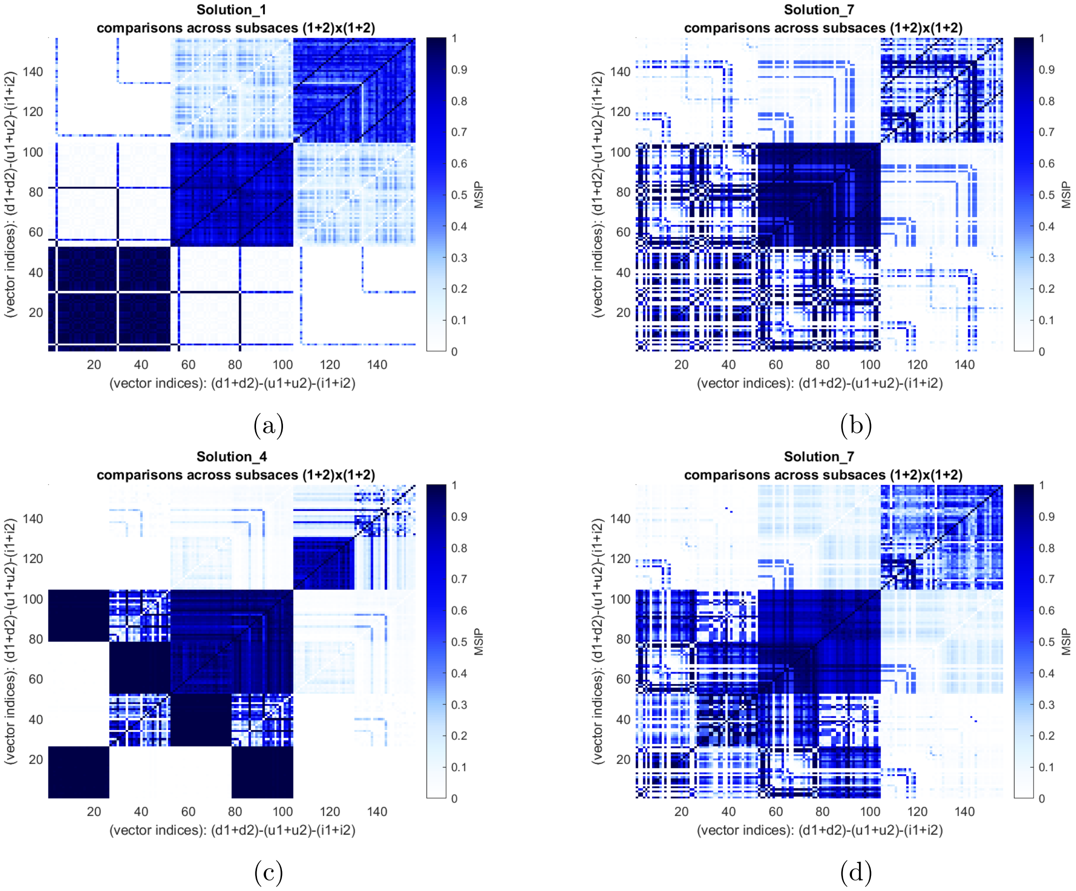
(c) 40000 training samples

Figure 6.5: SPLOC basis vector results sorted by selection power for (a) 10000 samples, (b) 20000 samples, and (c) 40000 samples.

not show any discrimination.

In total we found that with 7 ensemble runs (26*7 total SPLOCs) we only had 8 solutions which did not contain a discrimination solutions. In the active site plus the omega loop region there is 72 degrees of freedom, thus 72 modes are returned by the program, and most often SPLOC would only return a single discriminate mode for the TEM-1 to TEM-52 comparison. There was one set out of the validation data sets which produced up 6 modes of discrimination regularly. This set used the lowest number of training sets, 5 (10000 samples), and although it was consistent across all the ensemble runs it was not seen anywhere else in the validation data. The number of conserved modes between the TEM-1 and TEM-52 was variant, even within the same training set. Sometimes there could be as few as 2, while in other sets there were as many as 35.

In all this variation we can see again that, especially for smaller training sets, the discriminate motions were highly dependent on what structures were present in the training set. When the training was increased we saw a much more consistent number of discriminate and conserved modes. From the 26 runs in the ensemble that used our largest training set we saw 4 runs not produce any solution and two runs out of the remaining 24 identified 2 collective modes as discriminate. The selection power of the discriminate modes, which corresponds to the signal beyond noise of the modes, is not very high, although they are over the threshold for SPLOC. It is also interesting to note that although only one or two modes were returned as discriminant on average for the larger training sets, there were other modes which were significant in selection power. These modes, however, did not have the statistical significance to be pushed into the discriminant subspace. The 4 runs that did not have any discrimination also noticed a large increase in the indiscriminate subspace. All the other runs which found a single mode had around 3-5 indiscriminate modes. Some examples of the vector spaces returned by SPLOC are shown in Figure 6.5.



(a)

(b)

(c)

(d)

Figure 6.6: MSIP of SPLOC basis vectors. Plots can be divided into large 3 by 3 grids. Each box in the grid compares the 26 subspaces of each ensemble run (discriminate, undetermined, or indifferent respectively) with the a subspace from another ensemble run. (a) 10000 run with itself, (b) 40000 run with itself, (c) 10000 run with a 20000 run, (d) 40000 run with a 20000 run. The 3 by 3 grid is clearly visible in (a).

While every run more or less brought up something discriminate, we used the subspace analysis to look at how consistent the various discriminate spaces were. Instead of the usual RMSIP used with PCA, we show the MSIP because it provides better definition in the plots. Some of the MSIP plots which exemplify the data are shown in Figure 6.6. The clear large "boxes" correspond to the the different vector spaces: discriminate, unknown, and indiscriminate. For example the lower left large "box" in the lower left corner of an MSIP grid corresponds to discriminate subspaces with discriminate subspaces. Overall the plots show MSIP between the different subspaces within an ensemble with each other within the same ensemble (Figure 6.6 (a) and (b)) or between two ensembles (Figure 6.6 (c) and (d)). We can see that the first plot of 10000 samples with itself shows the most ideal results. All of the discriminate subspaces have a high similarity with themselves, and the other two spaces have significantly higher similarity with themselves too. If every run in the ensemble gave the same answer we would expect to see a block diagonal matrix like this.

The other plots are not nearly as block diagonal, but we do see that in most cases the indiscriminate space has very little in common with the other spaces. Further, we note that in most of the cases where there is a low MSIP in the discriminant space there is a high MSIP for the undetermined space. Keeping in mind that for the higher training levels the discriminant space only had a single mode, there were often one or more modes in the undetermined space which had a high signal beyond noise but not enough consensus to be pushed into the discriminate subspace. If these modes are significant in describing functional differences between the mutants, we could hypothesize that some discriminate motions may have been flipping between the discriminate space and the undetermined space. This would account for the complementary discriminant and undetermined subspaces in the MSIP plots and should be investigated further in the future by changing the voting threshold.

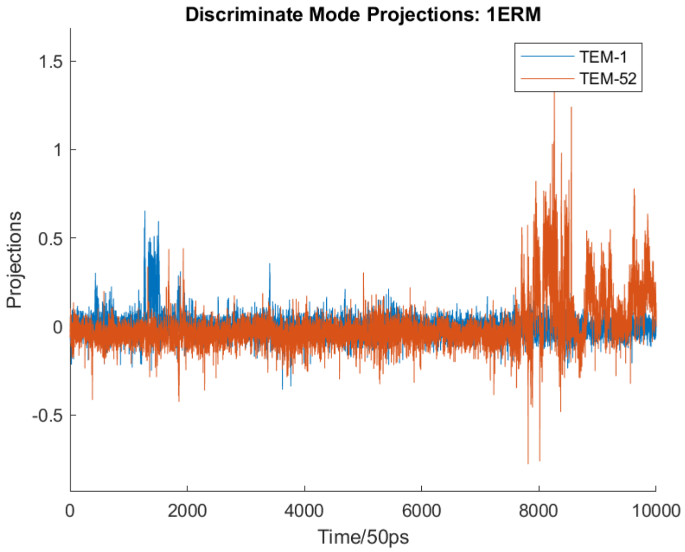


Figure 6.7: Discriminant SPLOC modes projected onto the TEM-1 and TEM-52 for the 1ERM starting structure runs. The TEM-1 and TEM-52 had similar dynamics in this discriminate motion for the first 400ns of the trajectory. TEM-52 "activates" for the last 100ns.

One way to analyze the presence of these discriminate motions in the dynamics of the MD simulation trajectories is to look at the time sequence of DVPs along the trajectory. We used the mode returned from our largest training set (40000 samples per sequence). Previously the two dimensional fuzzball plots have been shown, but because only 1 mode was returned as discriminant for most of the runs we have chosen to analyze this 1-Dimensional representation. This analysis revealed an interesting feature of the discriminant motions, the motions are not always "active" during the dynamics in the trajectory. In Figure 6.7 this effect can be seen. For the first 400ns of the trajectory (8000 frames) the projections of the enzymes motion into this mode were similar and very small for the most part, indicating that the motion was not present in either the wild type or the ESBL mutation. At around 400ns the TEM-52 "activates" this motions and the TEM-1 does not indicating that the dynamics have shifted. It should be noted that TEM-1 did have a short lived activation of this motion early in the trajectory, however for the most part it stays inactive. From this we see that the motions identified as the most discriminate between the wild type and ESBL may not be unique to either enzyme, but one molecule may exhibit the motion more often than the other or the motion could be easily induced in the ESBL mutation. This was observed in all of the TEM-1 and TEM-52 simulation comparisons. Because the motion activates and inactivates within a trajectory we saw the discriminate motion active at different points in the different trajectories, although in one of the trajectories the discriminate motion never activated. In all cases the motion was active more often in the TEM-52 than the TEM-1 however.

As stated, one of the benefits of SPLOC is that we can visualize the motions defined by the the discriminate modes. We used the same mode shown in Figure 6.7, filtering out any motion not defined from the last 2000 frames of 1ERM trajectory, where the discriminate mode was active, directly. The most distinctive motion is that in the TEM-52 the omega loop moves far more, although the actual motion is similar in direction, the amplitude is much larger. The actual amplitudes of both motions is small, however SPLOC was still able to pick up on the relative differences. In Figure 6.8 we show the beta-lactamase enzyme colored to show where SPLOC found discriminate motions on the protein. One clear effect in the motions of the protein is that whenever the omega loop makes a large motion the 130/131 residues respond by being pushed away. This is also coupled with a large deformation of the alpha helix containing the pivotal Serine 70 residue. The discriminate mode clearly shows that the ESBL TEM-52's active site has more mobility, which would be in agreement with claims in the literature.[21]

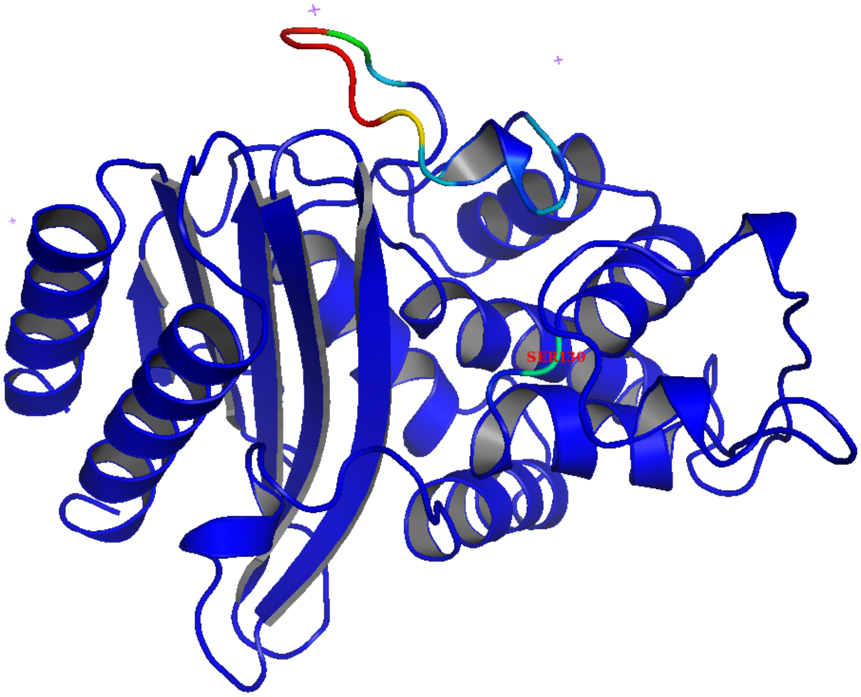


Figure 6.8: Beta-lactamase colored to show where the discriminant motions found by SPLOC happen. The more red the color the larger the mobility. The large red area is the omega loop, and the Serine 130 residue has also been labeled as SPLOC identified this site also.

CHAPTER 7: CONCLUSIONS

In this work we have explored a wide variety of methods for identifying the differences in dynamics of different proteins. Beginning with the common Cartesian PCA method we determined that the global dynamics was similar across the three mutants and discrimination of functional differences was not possible. However, we did detect discrimination by crystal structure which we termed the problem of crystal memory where the starting structure from MD is a major determining factor for the dynamics of the protein throughout the simulation. To resolve this problem we focused on various subsets of residues expected to be functionally relevant. This successfully removed or de-emphasized crystal memory while boosting signal to noise by reducing the number of variables in the covariance matrix. Some functional discrimination was achieved. Specifically we noted that the major difference in the dynamics of TEM-1 and TEM-52 proteins was an increase of of amplitude in motion of TEM-52 in the active site and omega loop regions, although the direction of motion was conserved. The omega loop seemed to be an important location for this increase in mobility, as well as showing that the Serine 130 may play a role in the extended spectrum activity of TEM-52. We also saw that when the subset was identified by RMSF differences key regions around the mutation showed increases in mobility, especially around the G238S mutation responsible for the extended spectrum resistance.

We also explored the utility of supervised machine learning methods for classifying wild type from ESBL mutants based on the MD data. We focused first on the common LDA/QDA methods, applied to the functionally relevant subset of residues. Upon de-noising the covariance matrix, the data could be classified. LDA was not a good method due to the differences in covariance between the different mutations, however QDA gave classifications of greater than 60 percent, which showed that there is certainly some differences in the structure ensembles from MD. We also validated that moving to subsets does decrease the effects of the crystal memory.

Finally we tested a new machine learning algorithm, SPLOC, that aims to identify collective motions that maximally differentiates functional motions between the TEM-1 and TEM-52 mutations. We restricted the analysis on the subsets of residues that were expected to show functional relevance. Through visualization of these motions, we confirmed the differentiating characteristics found using PCA on these functionally relevant subsets of residues. We saw a increase in mobility in the TEM-52 molecule, while we also noticed some cooperation between the omega loop, the Serine 130 residue, and the Serine 70 residue. Specifically whenever the omega loop becomes more mobile, the Serine 130 was pushed away and the alpha helix deformed. This motion was amplified in the TEM-52. Through analysis of DVPs we noted that the dynamics were not always present in the MD simulation trajectories.

Overall the results of this study agree with what is claimed in the literature. We found that the omega loop is an important feature when it comes to the extended spectrum capabilities of beta-lactamase. We see that the motion of this loop may influence the motions of the active site residues, specifically the Serine 70 and 130 which are responsible for the catalysis of beta lactams. From PCA we also saw that some residues outside of the omega loop and active site, including the functional mutation sites on TEM-52 (Glu104Lys and Gly238Ser) and and some other residues such as Lysine 215, Valine 216, and Aspartic Acid 252 could play a role in the ESBL properties too. The two functional mutation sites are located just under the omega loop and could be interacting with the residues in the loop to increase this motion. The other residues mentioned above are far from the omega loop but residues 215 and 216 do border the active site. A combination of interactions of these residues could be responsible for allowing the larger and more rigid extended spectrum beta lactam antibiotics [21] into the binding pocket by increasing the mobility of the omega loop and active site residues. A reshaping or widening of the binding pocket has been suggested by several sources.[24]

Looking to the future, we hope to continue studying beta-lactamase to further identify motions that may be conserved across all ESBL mutants. We have identified several directions for this endeavor. The first consideration is to simulate holo structures which include the beta-lactam in the binding pocket. We hope to simulate each mutation with at least two differentiating antibiotics, where one increases activity and the other not. An example would be simulating TEM-1 and TEM-52 respectively with penicillin and then again with cefotaxime. Penicillin would be chosen to represent the wild type resistance and cefotaxime would be chosen because of TEM-52's large binding affinity for it. We could also then compare the dynamics of the docked complexes with the unbound structures presented in this work to compare their essential dynamics and elucidate the induced dynamics caused by the ligand interaction.

More work should also be done in the direction of supervised learning methods. It was observed in this work that the LDA and QDA algorithms overshot each other in opposite directions, suggesting that Regularized Discriminant Analysis (RDA) may be an interesting approach. RDA modifies the scoring function of QDA to decrease the effect of the determinate of the covariance. Introducing this hyper-parameter is expected to tune the algorithm appropriately to maximize how well we can classify the structures. Further development of SPLOC is already in progess.

REFERENCES

[1] H. C. Neu, "The crisis in antibiotic resistance," *Science*, vol. 257, no. 5073, pp. 1064–1073, 1992.

[2] T. Ali, I. Ali, N. A. Khan, B. Han, J. Gao, *et al.*, "The growing genetic and functional diversity of extended spectrum beta-lactamases," *BioMed research international*, vol. 2018, 2018.

[3] M. S. Wilke, A. L. Lovering, and N. C. Strynadka, "β-lactam antibiotic resistance: a current structural perspective," *Current opinion in microbiology*, vol. 8, no. 5, pp. 525–533, 2005.

[4] S. Shaikh, J. Fatima, S. Shakil, S. M. D. Rizvi, and M. A. Kamal, "Antibiotic resistance and extended spectrum beta-lactamases: Types, epidemiology and treatment," *Saudi Journal of Biological Sciences*, vol. 22, no. 1, pp. 90 – 101, 2015. Special issue: Biological Aspects of Global Health Issues.

[5] K. Bush, G. A. Jacoby, and A. A. Medeiros, "A functional classification scheme for beta-lactamases and its correlation with molecular structure.," *Antimicrobial agents and chemotherapy*, vol. 39, no. 6, p. 1211, 1995.

[6] K. Bush and G. A. Jacoby, "Updated functional classification of β-lactamases," *Antimicrobial agents and chemotherapy*, vol. 54, no. 3, pp. 969–976, 2010.

[7] K. Bush, "Past and present perspectives on β-lactamases," *Antimicrobial agents and chemotherapy*, vol. 62, no. 10, pp. e01076–18, 2018.

[8] M. L. Salverda, J. A. G. De Visser, and M. Barlow, "Natural evolution of tem-1 beta-lactamase: Experimental reconstruction and clinical relevance," *FEMS Microbiology Reviews*, vol. 34, no. 6, pp. 1015–1036, 2010.

[9] K. P. F.R.S., "Liii. on lines and planes of closest fit to systems of points in space," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 2, no. 11, pp. 559–572, 1901.

[10] H. Hotelling, "Relations between two sets of variates," *Biometrika*, vol. 28, no. 3/4, pp. 321–377, 1936.

[11] J. Shlens, "A tutorial on principal component analysis," *CoRR*, vol. abs/1404.1100, 2014.

[12] A. Amadei, A. B. M. Linssen, and H. J. C. Berendsen, "Essential dynamics of proteins," *Proteins: Structure, Function, and Bioinformatics*, vol. 17, no. 4, pp. 412–425, 1993.

[13] F. K. Majiduddin, I. C. Materon, and T. G. Palzkill, "Molecular analysis of beta-lactamase structure and function," *International journal of medical microbiology*, vol. 292, no. 2, pp. 127–137, 2002.

[14] C. Micheletti, "Comparing proteins by their internal dynamics: Exploring structure–function relationships beyond static structural alignments," *Physics of life reviews*, vol. 10, no. 1, pp. 1–26, 2013.

[15] E. Fonzé, P. Charlier, Y. To'Th, M. Vermeire, X. Raquet, A. Dubus, and J.-M. Frere, "Tem1 β-lactamase structure solved by molecular replacement and refined structure of the s235a mutant," *Acta Crystallographica Section D: Biological Crystallography*, vol. 51, no. 5, pp. 682–694, 1995.

[16] C. Jelsch, F. Lenfant, J. Masson, and J. Samama, "β-lactamase tem1 of e. coli crystal structure determination at 2.5 a resolution," *FEBS letters*, vol. 299, no. 2, pp. 135–142, 1992.

[17] D. Verma, D. J. Jacobs, and D. R. Livesay, "Variations within class-a β-lactamase physiochemical properties reflect evolutionary and environmental patterns, but not antibiotic specificity," *PLOS Computational Biology*, vol. 9, 07 2013.

[18] E. M. Lewandowski, K. G. Lethbridge, R. Sanishvili, J. Skiba, K. Kowalski, and Y. Chen, "Mechanisms of proton relay and product release by class a β-lactamase at ultrahigh resolution," *The FEBS journal*, vol. 285, no. 1, pp. 87–100, 2018.

[19] J. Farmer, F. Kanwal, N. Nikulsin, M. C. Tsilimigras, and D. J. Jacobs, "Statistical measures to quantify similarity between molecular dynamics simulation trajectories," *Entropy*, vol. 19, no. 12, p. 646, 2017.

[20] X. Wang, G. Minasov, J. Blázquez, E. Caselli, F. Prati, and B. K. Shoichet, "Recognition and resistance in tem β-lactamase," *Biochemistry*, vol. 42, no. 28, pp. 8434–8444, 2003.

[21] M. Page, "Extended-spectrum β-lactamases: structure and kinetic mechanism," *Clinical Microbiology and Infection*, vol. 14, pp. 63–74, 2008.

[22] X. Wang, G. Minasov, and B. K. Shoichet, "The structural bases of antibiotic resistance in the clinically derived mutant -lactamases tem-30, tem-32, and tem-34," *Journal of Biological Chemistry*, vol. 277, no. 35, pp. 32149–32156, 2002.

[23] C. Poyart, P. Mugnier, G. Quesne, P. Berche, and P. Trieu-Cuot, "A novel extended-spectrum tem-type β-lactamase (tem-52) associated with decreased susceptibility to moxalactam inklebsiella pneumoniae," *Antimicrobial agents and chemotherapy*, vol. 42, no. 1, pp. 108–113, 1998.

[24] M. C. Orencia, J. S. Yoon, J. E. Ness, W. P. Stemmer, and R. C. Stevens, "Predicting the emergence of antibiotic resistance by directed evolution and structural analysis," *Nature Structural and Molecular Biology*, vol. 8, no. 3, p. 238, 2001.

[25] D. Shcherbinin, M. Y. Rubtsova, V. Grigorenko, I. Uporov, A. Veselovsky, and A. Egorov, "The study of the role of mutations m182t and q39k in the tem-72 β-lactamase structure by the molecular dynamics method," *Biochemistry*

(Moscow), Supplement Series B: Biomedical Chemistry, vol. 11, no. 2, pp. 120–127, 2017.

[26] A. Philippon, R. Labia, and G. Jacoby, "Extended-spectrum beta-lactamases.," *Antimicrobial agents and chemotherapy*, vol. 33, no. 8, p. 1131, 1989.

[27] H. J. Berendsen, D. van der Spoel, and R. van Drunen, "Gromacs: a message-passing parallel molecular dynamics implementation," *Computer physics communications*, vol. 91, no. 1-3, pp. 43–56, 1995.

[28] K. Lindorff-Larsen, S. Piana, K. Palmo, P. Maragakis, J. L. Klepeis, R. O. Dror, and D. E. Shaw, "Improved side-chain torsion potentials for the amber ff99sb protein force field," *Proteins: Structure, Function, and Bioinformatics*, vol. 78, no. 8, pp. 1950–1958, 2010.

[29] W. L. Jorgensen, J. Chandrasekhar, J. D. Madura, R. W. Impey, and M. L. Klein, "Comparison of simple potential functions for simulating liquid water," *The Journal of chemical physics*, vol. 79, no. 2, pp. 926–935, 1983.

[30] H. J. Berendsen, J. v. Postma, W. F. van Gunsteren, A. DiNola, and J. Haak, "Molecular dynamics with coupling to an external bath," *The Journal of chemical physics*, vol. 81, no. 8, pp. 3684–3690, 1984.

[31] S. Nosé and M. Klein, "Constant pressure molecular dynamics for molecular systems," *Molecular Physics*, vol. 50, no. 5, pp. 1055–1076, 1983.

[32] E. Papaleo, P. Mereghetti, P. Fantucci, R. Grandori, and L. De Gioia, "Free-energy landscape, principal component analysis, and structural clustering to identify representative conformations from molecular dynamics simulations: the myoglobin case," *Journal of molecular graphics and modelling*, vol. 27, no. 8, pp. 889–899, 2009.

[33] C. C. David, E. R. A. Singam, and D. J. Jacobs, "Jed: a java essential dynamics program for comparative analysis of protein trajectories," *BMC bioinformatics*, vol. 18, no. 1, p. 271, 2017.

[34] I. T. Jolliffe and J. Cadima, "Principal component analysis: a review and recent developments," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 374, no. 2065, p. 20150202, 2016.

[35] R. B. Cattell, "The scree test for the number of factors," *Multivariate behavioral research*, vol. 1, no. 2, pp. 245–276, 1966.

[36] H. J. Berendsen and S. Hayward, "Collective protein dynamics in relation to function," *Current opinion in structural biology*, vol. 10, no. 2, pp. 165–169, 2000.

[37] A. Amadei, A. B. Linssen, B. L. De Groot, and H. J. Berendsen, "Essential degrees of freedom of proteins," in *Modelling of biomolecular structures and mechanisms*, pp. 85–93, Springer, 1995.

[38] G. G. Maisuradze, A. Liwo, and H. A. Scheraga, "Relation between free energy landscapes of proteins and dynamics," *Journal of chemical theory and computation*, vol. 6, no. 2, pp. 583–595, 2010.

[39] M. Ernst, F. Sittel, and G. Stock, "Contact-and distance-based principal component analysis of protein dynamics," *The Journal of Chemical Physics*, vol. 143, no. 24, p. 12B640_1, 2015.

[40] A. Altis, P. H. Nguyen, R. Hegger, and G. Stock, "Dihedral angle principal component analysis of molecular dynamics simulations," *The Journal of chemical physics*, vol. 126, no. 24, p. 244111, 2007.

[41] R. J. Lindsay, J. Siess, D. P. Lohry, T. S. McGee, J. S. Ritchie, Q. R. Johnson, and T. Shen, "Characterizing protein conformations by correlation analysis of coarse-grained contact matrices," *The Journal of chemical physics*, vol. 148, no. 2, p. 025101, 2018.

[42]

[43] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Annals of eugenics*, vol. 7, no. 2, pp. 179–188, 1936.

[44] S. Mika, G. Ratsch, J. Weston, B. Scholkopf, and K.-R. Mullers, "Fisher discriminant analysis with kernels," in *Neural networks for signal processing IX: Proceedings of the 1999 IEEE signal processing society workshop (cat. no. 98th8468)*, pp. 41–48, Ieee, 1999.

[45] A. Tharwat, T. Gaber, A. Ibrahim, and A. E. Hassanien, "Linear discriminant analysis: A detailed tutorial," *AI communications*, vol. 30, no. 2, pp. 169–190, 2017.

[46] M. Welling, "Fisher linear discriminant analysis. department of computer science, university of toronto," tech. rep., Technical Report, 2005.

[47] T. Næs and B.-H. Mevik, "Understanding the collinearity problem in regression and discriminant analysis," *Journal of Chemometrics: A Journal of the Chemometrics Society*, vol. 15, no. 4, pp. 413–426, 2001.

[48] A. S. Ettayapuram Ramaprasad, S. Uddin, J. Casas-Finet, and D. J. Jacobs, "Decomposing dynamical couplings in mutated scfv antibody fragments into stabilizing and destabilizing effects," *Journal of the American Chemical Society*, vol. 139, no. 48, pp. 17508–17517, 2017.

[49] P. A. Lachenbruch and M. Goldstein, "Discriminant analysis," *Biometrics*, pp. 69–85, 1979.

[50] J. Farmer and D. Jacobs, "High throughput nonparametric probability density estimation," *PloS one*, vol. 13, no. 5, p. e0196937, 2018.

[51] J. H. Friedman, "Regularized discriminant analysis," *Journal of the American statistical association*, vol. 84, no. 405, pp. 165–175, 1989.