Title
Analyzing Protein Dynamics Using Dimensionality Reduction

Author
Eryol, Atahan

Publication Date
2015

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA
Merced

Analyzing Protein Dynamics Using Dimensionality Reduction

A Master Thesis submitted in partial satisfaction
of the requirements for the degree of

Master of Science

in

Electrical Engineering and Computer Science

by

Atahan Eryol

Committee in Charge:

Professor Shawn Newsam, Chair

Professor Michael E. Colvin

Professor David Noelle

August 2015

Analyzing Protein Dynamics Using Dimensionality Reduction

by

Atahan Eryol

The Master Thesis of Atahan Eryol is approved and it is acceptable in quality and form for publication on microfilm and electronically:

Professor David Noelle

Professor Michael E. Colvin

Professor Shawn Newsam, Chair

Univeristy of California, Merced

2015

To my family and my dear friends for their endless support.

Acknowledgements

I would like to thank my advisor Prof. Shawn Newsam for the continuous support during my Master's and related research for his guidance, patience, and knowledge. His invaluable advice helped me immensely during all the steps of my research and the writing of my thesis.

I would like to thank the rest of my thesis committee Prof. Michael E. Colvin and Prof. David Noelle, for their comments and opinions.

Besides my advisor and thesis committee, my sincere thanks also goes to Timothy Connolly, who has provided insight, data and knowledge throughout my research.

I would like to thank my family and friends for supporting me spiritually throughout writing this thesis and my education.

Curriculum Vitæ

Atahan Eryol

Education

2015	Master of Science, University of California, Merced (Expected).
2013	Bachelor of Science, Bilkent University, Ankara, Turkey.

Professional Experience

2014 – 2015	Teaching Assistant, University of California, Merced.
2013 – 2014	Research Assistant, University of California, Merced.
2013 – 2013	Software Developer Internship, ASELSAN, Turkey.
2012 – 2013	Web Designer Internship, Santa Barbara Trust for Historic Preservation, Santa Barbara.

Abstract

Analyzing Protein Dynamics Using Dimensionality Reduction

Atahan Eryol

This thesis investigates dimensionality reduction for analyzing the dynamics of protein simulations, particularly disordered proteins which do not fold into a fixed shape but are thought to perform their functions through their movements. Rather than analyze the movement of the proteins in 3D space, we use dimensionality reduction to project the molecular structure of the proteins into a target space in which each structure is represented as a point. All that is needed to do this are the pairwise distances between the protein structures. We can then visualize the projected structures in three dimensions to get a general idea of the dynamics of the protein. We can also measure how well the projection preserves the pairwise distances between structures for a particular target dimension to get an idea of the dimension of the dynamics of the protein in the original space.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Molecular Dynamics simulation is a powerful technique for sampling the motion and structure of proteins and other biomolecules. It is particularly useful for studying disordered proteins which change their structural shape through time. In this work we investigate protein dynamics using dimensionality reduction. One of the major goals of this work is to understand if disordered proteins move in a lower dimensional space than they structurally exist and to what degree they are disordered.

We model proteins as chains of amino acids in which the distance between amino acids is fixed but the angles can vary. A chain of N amino acids has order N degrees of freedom but in reality moves on a lower dimensional manifold.

The dimensionality reduction methods we consider require pairwise structural differences and so we investigate different interstructure distance measures such as root mean squared distance (RMSD), angular distance, etc. We also investigate

different dimensionality reduction techniques such as multidimensional scaling and Isomap. To analyze the effectiveness of the dimensionality reduction we investigate methods of comparing the distances between the structures in the original space and their projected representations as points in the reduced space. Correlation and Kruskals stress are different measures of agreement between the original pairwise distances and the pairwise distances of the points in the reduced space.

Throughout the project, three types of data, fully synthetic, partially synthetic and real proteins, are used to perform the experiments. Totally synthetic data that is comprised of random points in a high dimensional Euclidean space is used to evaluate the correctness of our methods by comparing our outputs to known results. Partially synthetic proteins for which we know the true dimensionality are used to investigate the interstructure distances and the dimensionality reduction methods. Once calibrated, the technique is applied to the real protein data for which we can only hypothesize dimensionality.

1.1 Motivation and Background

Generally in protein biophysics, the focus has been concentrated on the structure and the mechanics of natively folded proteins. With recent developments there has been a growing interest in the movement and dynamics of intrinsically

disordered proteins which do not fold but are thought to perform their functions through their dynamics. As most tools and studies of proteins are focused on their folded state, new analysis and visualization tools are needed to characterize the properties of these unfolded or disordered proteins. The current tools allow us to visualize the 3D motion of these proteins, but as their movement is hard to distinguish with mere observation, our work aims to provide better ways of understanding the movement. The question of whether disordered proteins are limited to lower dimensional dynamics is often asked and by using dimensionality reduction our goal is to provide broader understanding of the space and the dimensions they explore. In this project we investigate whether dimensionality reduction is useful for answering this problem.

The simulation data that contains the movement and coordinates of the atoms that makeup the proteins are generated by computational biologists using molecular dynamics software such as the GROningen MAchine for Chemical Simulations (GROMACS) [1]. Afterwards, the coordinate data is processed using inter-structure distance measures which calculate the pairwise distances. This pairwise distance data is used throughout the project in the form of symmetric pairwise distance matrices.

1.2 Use Case Example

This section provides a full example as a way of introducing our analysis pipeline. The details of this use case will be explained in the rest of the thesis. For the ease of explanation, a disordered protein named Poly-G is used. Initially the protein is modeled using a preset model that explains what atoms the protein consists of and what other characteristics it has. The model of the protein is then simulated using molecular dynamics software which in our case is GROMACS. During the simulation, the protein moves through time and this movement is recorded by the software. Depending on the timestep initially set before the simulation, the structure of the protein is recorded by the software at every t timesteps. The 3D locations of the atoms that make up the protein are recorded at each step. The simulation run by the software finally creates a trajectory that consists of all the structures that are observed during the simulation. For example, if a protein with 4 significant atoms, those that define the molecular chains, is simulated for 1000 ms and the timestep is 1ms, the resulting trajectory will have 1000 structures that make up the movement of the protein. These structures are also called the frames of the trajectory. In each structure (frame), the 3D location of the significant atoms are recorded. Therefore we will

have 3 coordinates in physical XYZ space for each of the significant atoms and in total 12 values that define the structure of the protein at a particular time.

The next step is to create a pairwise distance matrix for dimensionality reduction. In order to create a pairwise distance matrix where each of the entries corresponds to two single frames, a distance measurement is needed which takes the two groups of points that form the structures and outputs a single value that indicates the distance or dissimilarity between those two structures. In this use case, angular distance is chosen. In creating a pairwise distance matrix, each frames' dissimilarity (distance) to every other frame is calculated. For instance, a trajectory that consists of 1000 frames (structures) will result in a 1000x1000 pairwise distance matrix where each element of the matrix is the distance between the corresponding row index and column index of the selected entry. For example, the entry in the matrix in row 25 and column 15 is the distance (dissimilarity) between frames (structures) 25 and 15. It is important to note here that all of the interstructure distance measures used are symmetric, meaning that comparing the distance between structure p1 and p2 will have the same result as comparing p2 and p1. The diagonal consists of zeros as a structures' dissimilarity (distance) to itself is zero. The work up until this point is done by our computational biology collaborators; the rest of the analysis and the dimensionality reduction is the focus of this project.

In the next step, the generated pairwise distance matrix is given to the MDS (multidimensional scaling) algorithm which performs the dimensionality reduction. Multidimensional scaling takes the matrix as an input and tries to reconstruct the input data in a D dimensional Euclidean space where the pairwise distances are preserved as best as possible. The output is a matrix where rows correspond to the input data (in our case structures (frames)) and the columns correspond to the dimensions in the reconstructed space sorted from most significant to the least significant. It is important to note that the points in the output matrix are not the coordinates of the significant atoms of the original protein but represent the structures in a new space. For instance, a trajectory that initially had 1000 frames has a pairwise distance matrix of 1000x1000. After the dimensionality reduction, the resulting matrix will be 1000xD where D is the number of dimensions the data has been reduced to. In this use case, the pairwise distance matrix of Poly-G is reduced to 231 dimensions, therefore the resulting matrix is 1000x231. The 231 dimensions is the number of positive eigenvalues acquired from the MDS transformation.

Finally, the reconstructed data is used to do the analysis on the protein. In the following sections, the analysis methods will be explained in more detail. For this use case, certain attributes can be observed based on the analysis results. For example, 3D visualization in figure 1.1 provides a broad look at the movement of

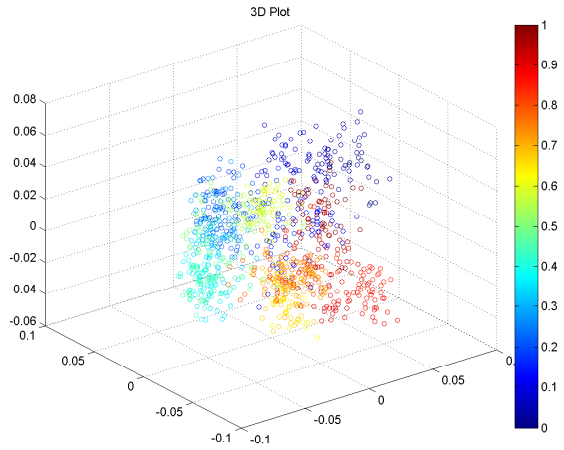


Figure 1.1: 3D plot of Poly-G using angular interstructure distance and MDS. Each point represents a structure. The coordinates of each point is the three most significant dimensions of that point. The color indicates the time.

the protein. It is created by taking the first 3 most significant dimensions of the resulting dimensionality reduced matrix. The values in the first 3 dimensions are taken as the x,y,z coordinates in the plot so that each structure in the input data is reperesented by a point in 3D space. The red points which represent the frames towards the end, between 800-1000, are closer to each other relative to the rest of the points. All the analysis is done either by visualizing the dimensionality reduced data or by comparing the pairwise distances in the output data to the original pairwise distances to try to understand whether the structures can faithfully be represented in the lower dimensional space.

1.3 Basics

In this section, the general workflow is explained step by step in a broad manner. Figure 1.2 illustrates the order of the processes.

1. If the data is synthetic, points or structures are randomly created. If the data is a protein, it is produced using Molecular Dynamics simulations. The movement of a protein is simulated with a given timestep which is generally between 10-100ps. The structures that result from these simulations, which we call frames, are subsampled in order to reduce the computational complexity.

2. The pairwise distances between each of these subsampled frames are calculated using different interstructure distance methods such as root mean square distance, angular, etc. The symmetric pairwise distance matrix, where each entry X_{ij} refers to the distance between the structures at time frames i and j, is generated.

3. This pairwise distance matrix is given as an input to one of the dimensionality reduction methods explained below.

4. The output of the dimensionality reduction is a set of points in a standard Euclidean space where each point corresponds to a structure. These points

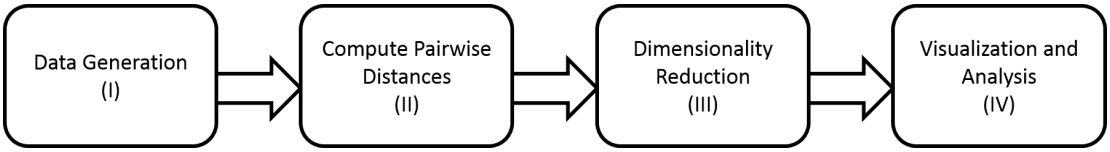


Figure 1.2: Data processing pipeline.

are then visualized in two or three dimensions. They are also analyzed in varying dimensions to see how faithfully they capture the pairwise distances between the original data. The general idea is that if a set of structures can be represented in a D dimensional space, then they have a dimension less than or equal to D.

Chapter 2

Methods

2.1 Introduction

We focus on two aspects of the analysis pipeline in figure 1.2: the interstructure distance computation and the dimensionality reduction. We consider a number of different algorithms for both. Each of the dimensionality reduction methods we consider takes a pairwise distance matrix as an input. Therefore the interstructure distance computation takes a trajectory consisting of the structures (frames) and outputs a symmetric pairwise distance matrix. The best distance measure to calculate distances between two structures is not obvious, therefore several methods are considered and explained in more detail in the following sections. For each trajectory, four different interstructure distance measurements are used to compute the distances between any two given structures. A symmetric pairwise distance matrix is formed with each row corresponding to the distance of the nth

structure to the rest of the structures in the trajectory. In the matrix, the diagonal consists of zeros.

Each of the dimensionality reduction methods takes a symmetric pairwise distance matrix and outputs a matrix that defines where the points corresponding to the input frames lie in a multi-dimensional Euclidean space. A number of different dimensionality reduction methods are considered and compared.

2.2 Interstructure Distance (ISD) Measures

The aim of computing distances between two structures is to have a single value that represents how similar they are. The structures (frames of the trajectory) are initially N points in a 3D space where each point is a bead that represents the backbone atoms of the protein. The beads move in time forming different structures so that we have M structures each with N points in 3D space. The methods mentioned below take all these structures as input and calculate a single value for each of the structure pairs. The output is an MxM pairwise distance matrix where each entry M_{ij} is the value of the distance between the structures i and j.

2.2.1 RMSD (Root Mean Square Distance)

As mentioned before, a protein can be defined by its significant atoms (our beads) and their locations in physical 3D space. RMSD computes the difference between two protein structures using the locations of the significant atoms. This method takes two structures as an input and outputs a single value which indicates the difference between those two structures. The smaller the output the more similar the structures. For this measurement, each structure is fit separately to an ellipse and the center of mass and major/minor axes of the ellipse are calculated. For comparing two structures, the ellipses are aligned and centered. After the alignment, the Euclidean distance between corresponding beads, which are treated as 3D points, is measured. Finally these distances are averaged for a final value. Figure 2.1 illustrates the computation in more detail.

Our RMSD implementation uses the GROMACS 4 library functions resetx and dofit to perform the molecular alignment. All RMSD results presented here use C_α atoms for alignment and distance calculations. If v_i and w_i represent the atoms that form the protein then RMSD is calculated as (after alignment):

$$\mathrm{ISD}_{\mathrm{RMSD}}(\mathbf{v}, \mathbf{w}) = \sqrt{\frac{1}{n}\sum_{i=1}^{n} \|v_i - w_i\|^2} \tag{2.1}$$

$$= \sqrt{\frac{1}{n}\sum_{i=1}^{n} ((v_{ix} - w_{ix})^2 + (v_{iy} - w_{iy})^2 + (v_{iz} - w_{iz})^2)} \tag{2.2}$$

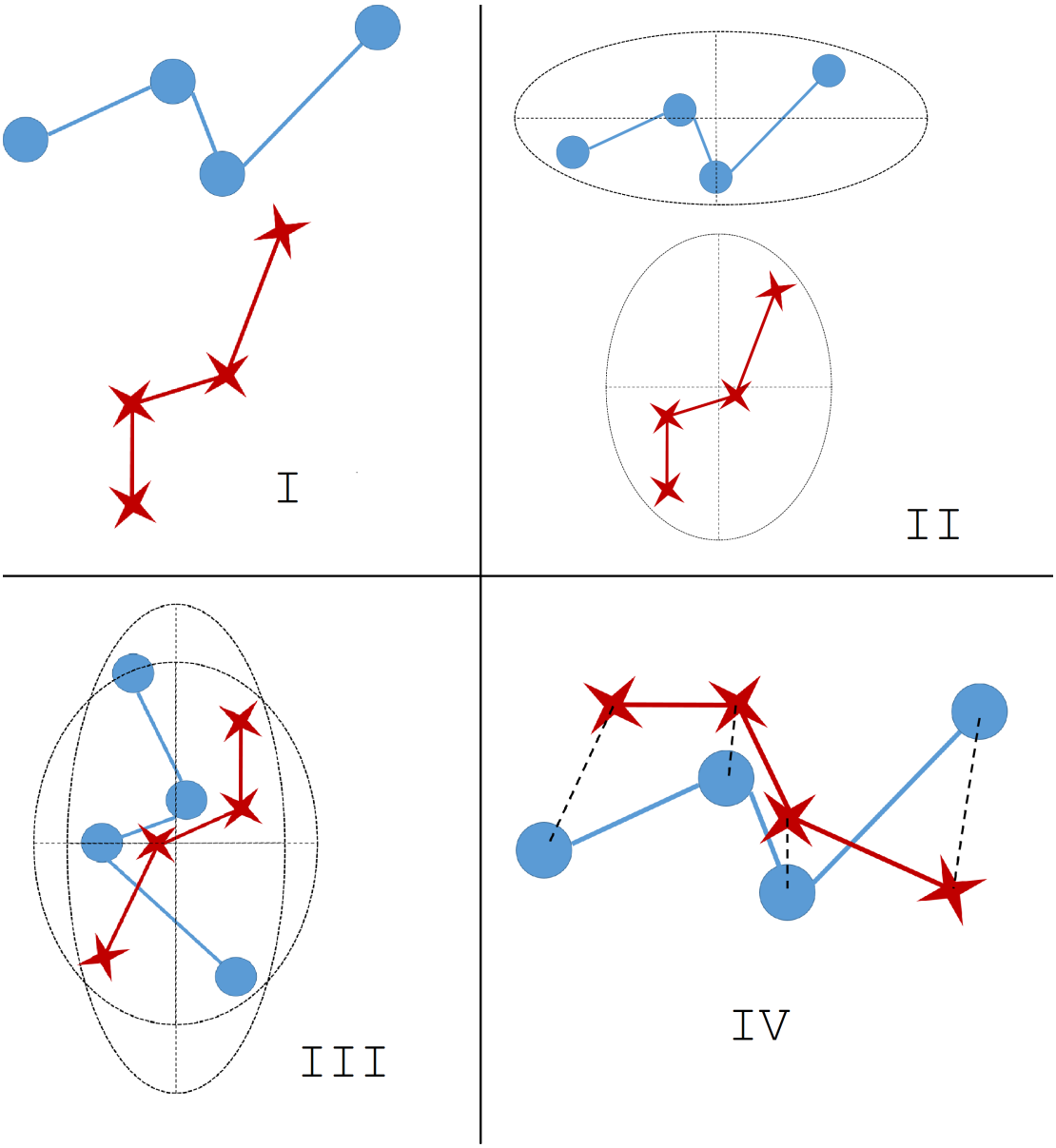


Figure 2.1: (I) Input structures. (II) The structures are fit to ellipses. (III) The ellipses are aligned. (IV) After the alignment, the distances between the beads are calculated and averaged.

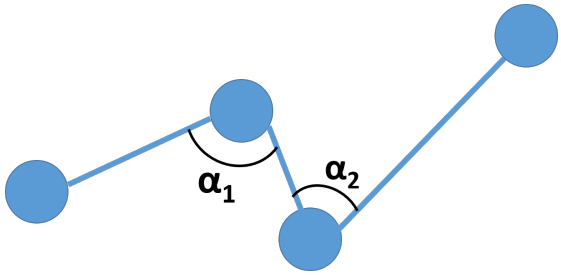


Figure 2.2: The angles α_1 and α_2 illustrate the angles used for the angular ISD. The blue disks are significant atoms forming the protein and the lines between the disks are the fixed length chains that connect the atoms.

2.2.2 Angular

The locations of the significant atoms in a protein is one way to define its structure. However, since the distances between the significant atoms are fixed, the structure can also be characterized by the backbone angles between adjacent atoms. In our context, changes in these angles encode the movement of the protein. However, a change in a single angle (which is considered a small change in our context) can result in a change in the location of multiple significant atoms. This might result in a large dissimilarity for distance measures based on coordinates such as RMSD. We therefore consider an angular based measure.

As shown in figure 2.2, the angle formed by triplets of beads in a single structure is calculated. With this method, each structure has $N-2$ angles that define the structure, where N is the total number of beads that form the structure. For

comparing two structures, the Euclidean distances between corresponding angles are calculated.

Structures are compared by calculating the root-mean-square of the differences between corresponding backbone angles. The reference structure angle θ_{R_i} and the comparison structure angle θ_{S_i} is calculated for each contiguous set of three C_α atoms. Therefore for N beads, there are $N-2$ total backbone angles. Each backbone angle θ_i is calculated using the gmxangle [2] function from the GROMACS 4 library with the two vectors $\overline{C_iC_{i-1}}$ and $\overline{C_iC_{i+1}}$ as inputs where C_i is a bead coordinate. This function calculates θ_i using the equation:

$$\theta_i = \tan^{-1} \left\| \frac{\overline{C_iC_{i-1}} \times \overline{C_iC_{i+1}}}{\overline{C_iC_{i-1}} \cdot \overline{C_iC_{i+1}}} \right\| \tag{2.3}$$

The angular ISD measure is defined as the root-mean-square of the differences between backbone angles rescaled to return a value between zero and one:

$$ISD_{ang} = \frac{1}{\pi} \sqrt{\frac{1}{n-2} \sum_{i=2}^{n-1} (\theta_{S_i} - \theta_{R_i})^2} \tag{2.4}$$

2.2.3 Dihedral Angles

For this measurement, the structure is represented by the dihedral angles between overlapping triplets of beads as shown in figure 2.3. Each triplet of beads creates a plane and the angle between adjacent planes, the dihedral angle, is calculated. A structure with N beads has $N-3$ dihedral angles. For comparing two

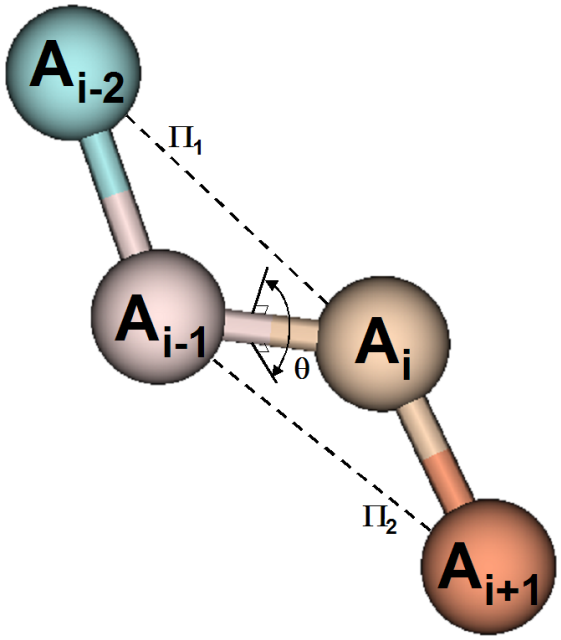


Figure 2.3: Θ is the angle formed by the triplets of beads. π_1 is the plane uniquely defined by the first three beads A_{i-2}, A_{i-1} and A_i. Similarly, π_2 is the plane uniquely defined by the last three beads A_{i-1}, A_i, and A_{i+1}. The dihedral angle, Θ, is defined as the smallest angle between these two planes [5].

structures, the Euclidean distance between the vectors formed by these dihedral angles is calculated.

The backbone dihedral angle made by each set of four beads is determined by choosing two vectors normal to the planes formed by the two contiguous sets of three beads. The two normal vectors are calculated by taking the cross products of the bead coordinates $\overrightarrow{V_1} = \overrightarrow{C_iC_{i-1}} \times \overrightarrow{C_iC_{i+1}}$ and $\overrightarrow{V_1} = \overrightarrow{C_iC_{i+1}} \times \overrightarrow{C_{i+2}C_{i+1}}$. The dihedral angle θ_i between the resultant vectors is calculated using the GROMACS

4 library function gmxangle [2]. The dihedral angle is multiplied by the sign of $\overrightarrow{C_iC_{i-1}} \cdot \overrightarrow{V_2}$ to give a consistent dihedral angle measure with a range of 2π.

$$\theta_i = \left(\frac{\overrightarrow{C_iC_{i-1}} \cdot \overrightarrow{V_2}}{\left|\overrightarrow{C_iC_{i-1}} \cdot \overrightarrow{V_2}\right|} \right) \tan^{-1} \frac{|\overrightarrow{V_1} \times \overrightarrow{V_2}|}{\overrightarrow{V_1} \cdot \overrightarrow{V_2}} \tag{2.5}$$

Since the corresponding dihedral angles in two structures, θ_{S_i} and θ_{R_i} are both bounded by $[-\pi, \pi]$, the difference $\Delta\theta_i = \theta_{S_i} - \theta_{R_i}$ has a range of 4π. An adjustment is made by adding 2π to $\Delta\theta_i$ for $\Delta\theta_i < -\pi$ and subtracting 2π from $\Delta\theta_i$ for $\Delta\theta_i > \pi$. The ISD is defined as the root-mean-square of the differences between the N-3 backbone dihedral angles rescaled to return a value between zero for $\theta_{Ri} = \theta_{Si}$ and a maximum of one.

$$ISD_{dih} = \frac{1}{2\pi} \sqrt{\frac{1}{n-3} \sum_{i=2}^{N-2} {\Delta\theta_i}^2} \tag{2.6}$$

Because of this adjustment, we suspect that during the calculation of dihedral angles between the beads, any dihedral angle that changes more than 180 degrees can be considered an additional dimension. Therefore instead of having N-3 dimensions we end up with having 2*(N-3) dimensions where one half of the circular motion (first 180 degrees change in the angle) is considered a degree of freedom, and the other half of the the circular motion (degree change ranging from 180-360 in the angle) is considered another degree of freedom.

2.3 Dimensionality Reduction Methods

In general, the problem of dimensionality reduction is the transformation of high-dimensional data into a lower dimension without distorting the relations between the data. In an ideal scenario, the reduced space should perfectly reflect all the properties expected to be observed in the original data. There are multiple ways of reducing the dimension of the data such as PCA (principle component analysis), LLE (local linear embedding) or MDS (multidimensional scaling). Methods based on PCA require the original data to explicitly lie in a multidimensional space (one value per axis). In our context of analyzing the movement of proteins, we do not have such an explicit representation but only have the distances between structures. Therefore we will focus on using methods such as MDS that only require pairwise distances.

The overall goal of applying dimensionality reduction to our data is to try understand the dimensionality of the dynamics. The output of MDS and other dimensionality reduction methods we consider is an explicit representation of the structures as points in a multidimensional space. The benefit of the dimensionality reduction techniques we consider is that only pairwise distance matrices are needed as input. This allows us to treat the structures themselves as points in a D dimensional space after dimensionality reduction is performed. That is, the input

is a PxP symmetric pairwise distance matrix, where P is the number of structures and the output is a PxD matrix where D is the target dimension. Each row in the output matrix is the coordinates of one of the input structures in the D dimensional space. We can visualize the structures by setting D=3. We can also use this framework to try determine the true dimension of the input data (the structures) by looking at how well the relations between structures are preserved as we vary (lower) D.

2.3.1 Classical Multidimensional Scaling

MDS is a dimensionality reduction method which aims to place each object in a D dimensional space such that the pairwise distances between the original data and the projected data are preserved as much as possible. The objects in the original space are assigned coordinates in each of the projected dimensions [7]. The input to the method is a symmetric, PxP, zero diagonal matrix where each entry is the difference between structure P1 and P2. The output is a PxD matrix where D refers to the number of dimensions in the projected space and each row is a projected point. The method is able to preserve pairwise distances if the original points are already in a Euclidean space and do not have a dimension greater than

D. The problem can be formulated as the following:

$$\min_{x_1,\ldots,x_I} \sum_{i<j} (\|x_i - x_j\| - \delta_{i,j})^2. \tag{2.7}$$

where x_i is the coordinate of the point i in the projected space and $\delta_{i,j}$ is the original distance between structure i and j [10].

Additionally, the method provides eigenvalues associated with the dimensions. The larger the eigenvalue is for a particular dimension, the larger the impact of that dimension is to the resulting space. Therefore when selecting a target dimension to reduce to, the dimensions corresponding to the largest eigenvalues are selected. When starting in a Euclidean space with no noise, the number of nonzero eigenvalues indicates the dimensionality of the input data. The other eigenvalues (if any) will be zero. Negative eigenvalues indicate that the distance measure used in the original matrix is not Euclidean. It is also important to note that there should be enough instances in the pairwise matrix such that the the original dimensionality can be estimated correctly, meaning that the number instances should be higher than the number of dimensions the objects lie in. The following example further illustrates how MDS works:

Let us assume we are given a distance matrix for 5 cities in the United States, in the following table:

Atl	Chi	Den	Hou	LA
0	587	1212	701	1936
587	0	920	940	1745
1212	920	0	879	831
701	940	879	0	1374
1936	1745	831	1374	0

As it can be seen, the distances matrix is 5x5 and symmetric. Each value corresponds to the distance in miles between the two cities. When we apply classical MDS to this distance matrix we get the following 5x2 output:

X	Y
-791.596226485087	134.256819268446
-544.586489107515	-400.590490571815
361.832113247859	-240.774124034717
-168.167800188666	460.933482163693
1142.51840253341	46.1743131743929

And these are the eigenvalues:

Eigenvalues
2387750.21722908
451061.555638008
2.15888744381498e-10
-451.507829006487
-2537.66503808011

In this case it can be observed that the first two eigenvalues are significantly larger than the rest. This indicates that the first two dimensions of the projected space capture most of the variation in the data. The negative eigenvalues indicate that the distances in the input matrix are not Euclidean. This makes sense as the distances are measured with the curvature of the Earth taken into account. When these 5 points are plotted using the first two dimensions of the projection we can observe that they are placed in correspondence with their actual locations

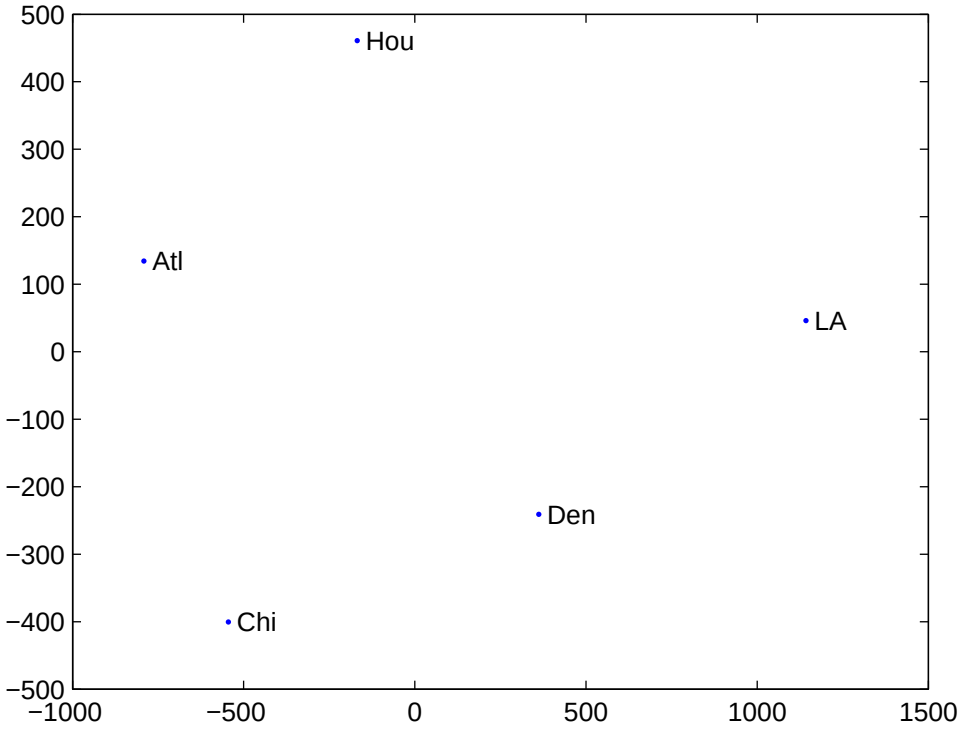


Figure 2.4: Each point represents a city labeled by its name. It is important to note that orientation is arbitrary and in this case north is downwards.

on a map as shown in figure 2.4. The side effect of MDS is that the orientation can be arbitrary, where in this case north is not pointing upwards and the map is rotated.

In summary, the number of non-zero eigenvalues indicates the dimensionality of the input data if the pairwise distances are Euclidean. If the distances are not, such as the case with our pairwise structure distances, then there will be

negative eigenvalues and the dimensionality of the input data will not be clear. Nevertheless, the ordering of the positive eigenvalues should still indicate the significance of the dimensions in the projected space which is important when reducing the dimensionality to just 3 for visualization.

2.3.2 Non-classical Multidimensional Scaling

Similar to classical MDS, nonclassical MDS is used to visualize high dimensional data by reducing its dimensionality. In classical MDS, the goal is to place points in a way such that the distances between them are approximate to the dissimilarities given in the input matrix. In some cases this might be too strict of a requirement and nonclassical MDS allows this constraint to be relaxed. Nonclassical MDS is based on classical MDS with extra iterative post-processing to achieve a better fit of the pairwise distances in the reduced space. It accepts an NxN input distance matrix similar to classical MDS. It starts by applying the classical method as a starting point. After that, it iteratively perturbs the points so as to minimize the difference of the pairwise distances in the original and the reduced space. It does this using gradient decent. For data that is not Euclidean, it can give better results in terms of difference of the pairwise distances compared to the classical method. The target dimensionality is given as an input (unlike classical MDS) and the iterative process to reduce the pairwise differences is done on that

many number of dimensions. The iteration is done until the distances between the resulting points closely resembles the original dissimilarities or a threshold for the number of iterations is reached. In addition to an output matrix consisting of the points in a D dimensional space, the method provides metrics for the goodness of the fit which will be explained further in the results.

Note that while nonclassical MDS preserves the ordering of the pairwise distances the distances might undergo non-linear transformation. That is, distances of different magnitudes might be scaled differently [3].

2.3.3 Isomap

Isomap is a dimensionality reduction method similar to MDS. The major difference is that MDS uses the original pairwise distances, whereas Isomap uses the pairwise geodesic distances to create the low dimensional embedding. K-nearest neighbors are calculated where the distance between neighbors is the original distance. Non-neighbor distances are initially set as infinite, and a graph search algorithm is performed to calculate the distance between any two non-neighboring points. The distance between them is then the sum of the edges of neighboring points along the shortest path. For all geodesic pairwise distances to be calculated, the points with k-nearest neighbors has to form a single connected component. This can be achieved by tuning the parameters of the method, such as increasing k

which the number of neighbors selected initially to construct the graph. After the construction of the graph, classical MDS is run on the geodesic distances to create the lower dimensional space [8]. Isomap has the potential benefit of correctly identifying the dimensionality of non-Euclidean data.

It is important to note that the evaluation methods described in Chapter 4 are not necessarily appropriate for Isomap. These evaluation methods compare the pairwise distances before and after dimensionality reduction. Isomap uses the geodesic distances which clearly distorts the original distances in an attempt to unravel lower dimensional attributes. Despite this handicap, 3D plots based on Isomap dimensionality reduction are useful for visualization. In particular, when combined with temporal information, Isomap can determine when a structure undergoes a large change or when a trajectory revisits itself. This is due to Isomap's neighborhood-based analysis.

Chapter 3

Data

Three types of data are used throughout the project to perform the evaluations.

I Fully synthetic data consisting of random points in a D dimensional Euclidean space.

II A collection of synthetically created structures.

III Trajectories of real proteins.

All of this data is are preprocessed by calculating the pairwise distances for input to the dimensionality reduction techniques.

3.1 Fully Synthetic

The fully synthetic data set consists of random points in a D dimensional space. The symmetric input matrix is constructed using the pairwise Euclidean distances

between the points. This data set allows us to observe how the dimensionality reduction methods perform on data with known dimension and with pairwise distances computed using a Euclidean distance.

We generate 1000 points at random in a 3D space. Figure shows the 1000x1000 symmetric matrix that is the pairwise distances of the randomly generated points. The points are uniformly distributed and therefore the indices do not relate to the distance between points. We expect MDS to be able to perfectly reconstruct the distributions of the 1000 points given the distance matrix. The orientation of the points in the projected space is arbitrary and may not be the same as the original space.

3.2 Synthetic MD Simulations

The synthetic molecular dynamics dataset consists of artificially constructed molecules that exhibit a certain behavior and have known properties such as their dimension. A protein with N beads can be constructed and its trajectory can be created without any other constraints. This type of data allows us to explore the dimensionality captured by different interstructure distance measures particularly in relation to the size of the protein which we can vary. The artificially constructed proteins can be processed into symmetric different pairwise distance

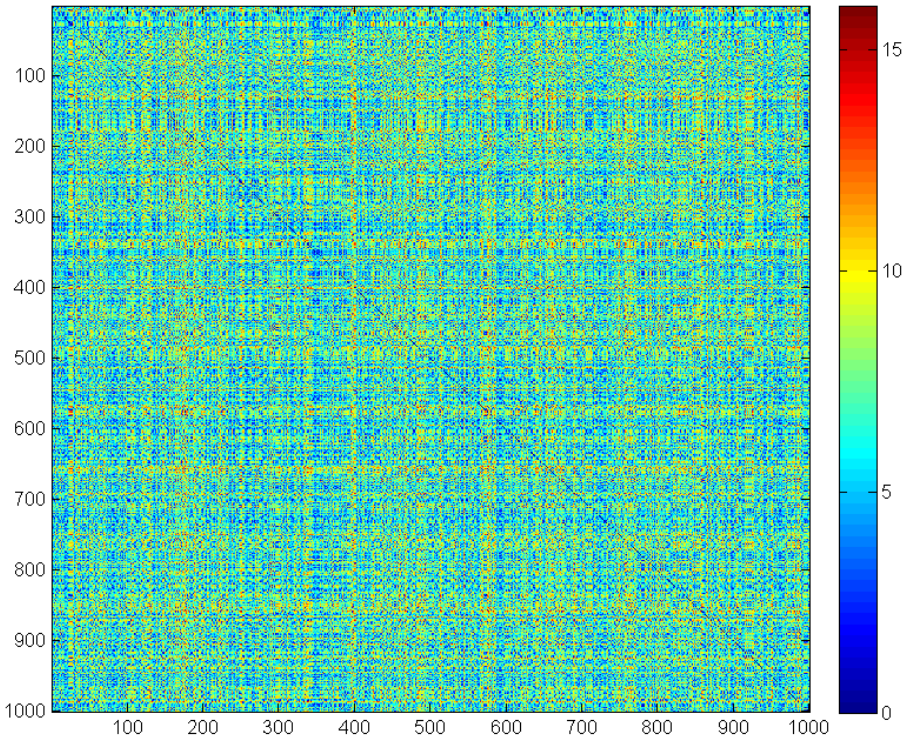


Figure 3.1: Color representation of the pairwise distance matrix of the synthetic points that were created in 3 dimensions. The colorbar represents the values where blue is smaller and red is further apart. The diagonal is dark blue which indicates that the distance between a point and itself is zero.

matrices using any of the distance measures described in chapter 2. We can thus observe which of the interstructure distance measures give more reliable results for a particular dimensionality reduction method. This data set is used to validate the dimensionality reduction and evaluation methods since we have the true dimension of the dynamics of the structures. This data set allows us to calibrate our methods for when we apply them to the real MD simulations.

We refer to the movement of a molecule through time as a trajectory and a snapshot of the molecule at a particular time as a frame. Our synthetic MD data set consists of trajectories in which the frames are generated using what we refer to as a random chain. A random chain consists of N beads with fixed size links but which are otherwise free to move at will. The points are randomly distributed and therefore the frame indices do not relate to the distance between structures.

For the context of this thesis, several synthetic simulations are used with varying sizes. Random moving chains with the following number of beads are used: 10, 30, 50. The pairwise distances of each of the trajectories are calculated using the interstructure distance measures explained in chapter 2. Therefore, as an input to our dimensionality reduction methods, we have 3 pairwise distances matrices for each of the different sized random chains. Table 3.1 provides information regarding the attributes of the random chains and the expected dimensionality with respect to ISD measures. To recall, we expect $3N$ dimensions for RMSD, $N - 2$

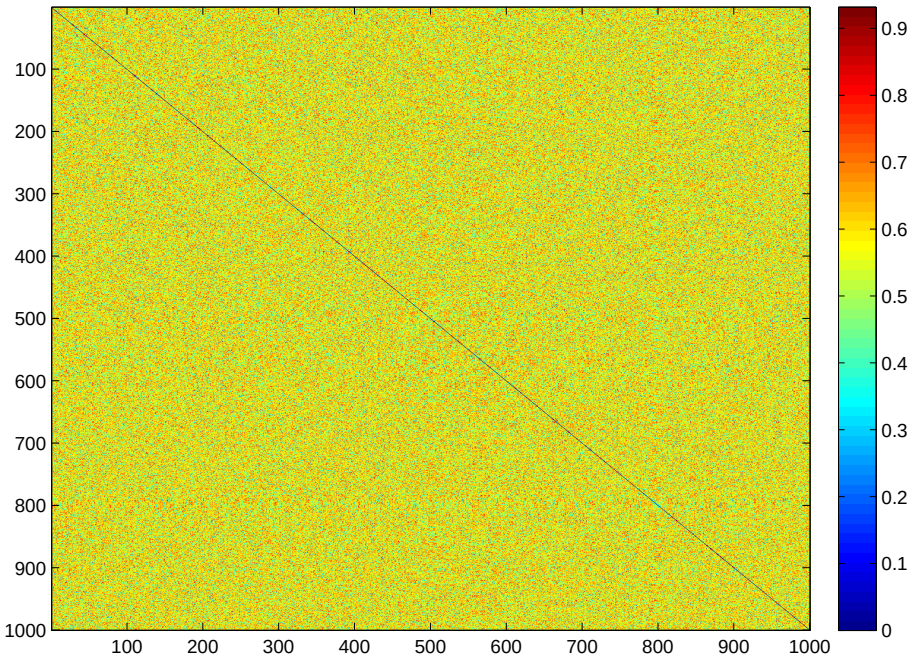


Figure 3.2: Color representation of the pairwise distance matrix of RCn10 using dihedral angles ISD. The colorbar represents the values where blue is smaller and red is further apart. The diagonal is dark blue which indicates that the distance between a point and itself is zero.

Synthetic Trajectory	Trajectory Size	#Of Beads	#Of Angles	#Of Dihedral Angles	Expected for Angular ISD	Expected for Dihedral ISD	Expected for RMSD ISD
RCn10	1000 Frames	10	8	7	8	14	30
RCn30	1000 Frames	30	28	27	28	54	90
RCn50	1000 Frames	50	48	47	48	94	150

Table 3.1: Number of beads, angles and dihedral angles with expected number of dimensions for each random chain.

dimensions for angular and $2N - 6$ dimensions for dihedral angles where N is the number of beads.

3.3 Real Protein Simulations

Our third data set consists of MD simulations of real proteins. The motivation for applying dimensionality reduction to MD simulations of real proteins is to better understand their dynamics. Insight into whether a protein is moving in a lower dimensional subspace can provide clues to its level of disorder.

In real trajectories, one can observe which parts of the trajectory are closer or farther by just looking at the pairwise distance matrices. In the synthetic points, it is evident from the color matrix that the points are randomly distributed and

therefore the frame indices do not relate to the distance between structures. On the other hand, in real trajectories, it is often observed that the trajectory usually does not make large movements in small time steps therefore making the frames that are close by in time also more similar in structure. This is reflected in the pairwise distance matrix where the diagonal region is blueish but the matrix becomes more red for frames that are further apart in time.

The following real protein trajectories are used :

- Poly-A, 1 trajectory with 1000 frames, 50 amino acids
- Poly-G, 1 trajectory with 1000 frames, 50 amino acids
- Poly-Q, 1 trajectory with 1000 frames, 50 amino acids
- GLFG, 5 trajectories all merged into a single set of frames. Each trajectory has 360 frames so 1800 frames in total.
- AXAG, 5 trajectories all merged into a single set of frames. Each trajectory has 360 frames so 1800 frames in total.

Poly-A, Poly-G and Poly-Q are created in GROMACS 4.5.5 using explicit solvent, pressure coupling and amber-99SB-ILDN as the force field. Poly-A, Poly-G and Poly-Q are all disordered proteins with Poly-Q being more disordered than Poly-G and Poly-G more disordered than Poly-A.

Chapter 4

Results

We perform both qualitative and quantitative analysis of the dimensionality reduction results. We qualitatively visualize the trajectories in 3D. We also derive quantitative measures that try get at how well the pairwise distances are preserved for a particular combination of ISD and dimensionality reduction method. The synthetic data allows us to study and calibrate the analysis framework so that it can be then applied to the real data.

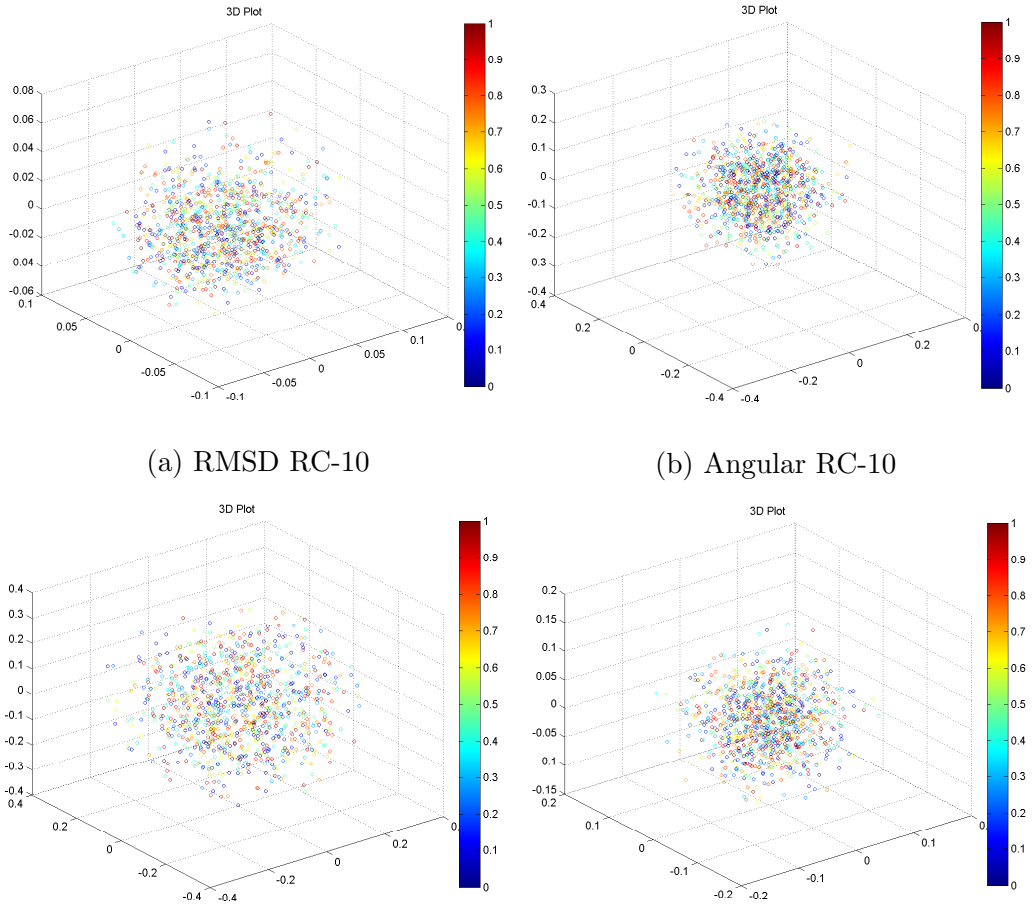
4.1 3D Plots

3D plots are used to visualize the trajectories and to get a basic idea of the motion of the protein throughout its trajectory. It is good for understanding how much of the space it is exploring or if it is revisiting certain states. The plot is derived by the output of the dimensionality reduction for a trajectory. The most significant 3 dimensions are used as the x, y, z coordinates of the plot.

It is important to note here that each of the frames are represented here as a point. Although it is limited to only 3 dimensions, if these 3 dimensions capture most of the variance, it is an accurate representation of the relationships between structures. That is, if two points are close to each other, the structures they represent are considered similar with respect to the ISD measure that was used to create the pairwise distance matrix as an input to the dimensionality reduction.

The 3D plots of the random chains do not vary in any meaningful way between ISD measures (for MDS). This is in part because the structures are independent and do not have any temporal correlation. It is also due to the high dimensionality of the structures: three dimensions clearly cannot capture most of the variance. If we were able to visualize the projected data in a $D+1$ dimensional space where D is dimension of the structures, we might expect to see the data on a D dimensional manifold. But in our case, as it can be seen in figure 4.1 three dimensions do not provide much insight into the movement and behavior of the random chains.

An example of a 3D plot of the protein Poly-A is shown in figure 4.2 with angular distance used as the interstructure distance measure. The colorbar in the plot indicates time. In this plot we can observe that the frames near the end of the trajectory (the red cluster) are closer to each other, which tells us that towards the end of the simulation the protein tends to explore less space relative to the beginning.



(a) RMSD RC-10

(b) Angular RC-10

(c) Dihedral RC-10

(d) Angular RC-30

Figure 4.1: 3D plots using three different distance measures for RC-10 and angular ISD for RC-30. The color bar indicates the index of the structure (from 1 to 1000 but scaled to 0 to 1).

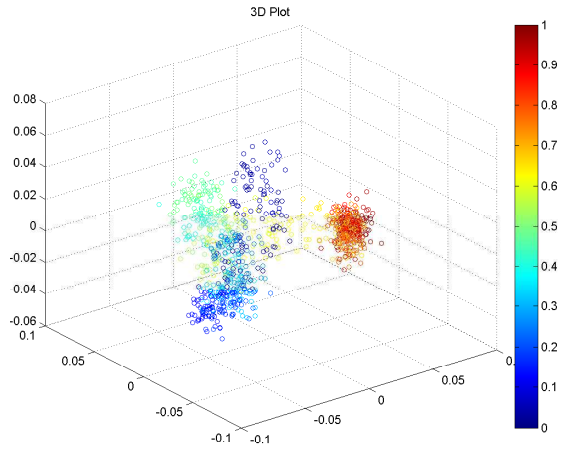
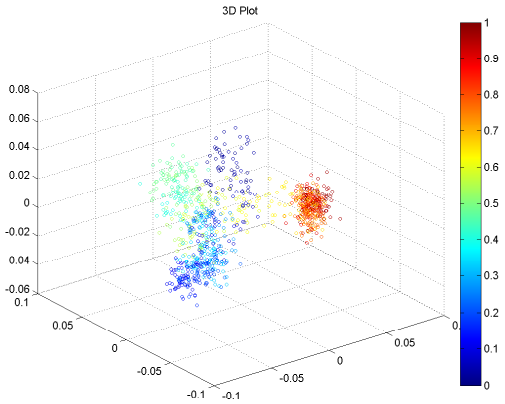


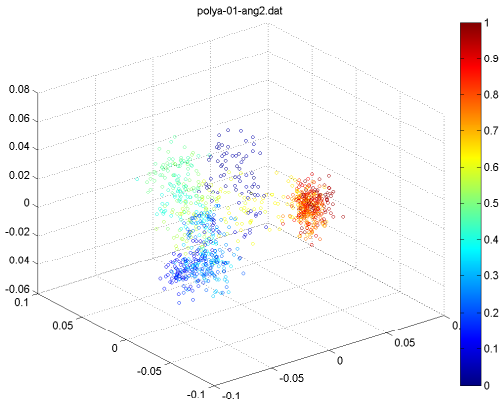
Figure 4.2: 3D plot of Poly-A using angular ISD and MDS. Each point represents a structure. The coordinates of each point is the three most significant dimensions of that point. The color indicates the time.

Based on our observations, the 3D plots based on nonclassical MDS do not differ in any significant way from those based on MDS, as shown in figure 4.3. We do note, however, that the 3D plots based on Isomap can help identify big leaps or drastic changes in a trajectory, as shown in figure 4.4.

It can be argued that in most cases nonclassical MDS does not result in much difference in visualization for the proteins we analyze. Isomap is helpful in identifying big leaps or drastic changes in the movement of the trajectory. It is less sensitive to small changes in structure depending on the number of neighbors selected to construct the geodesic distances.

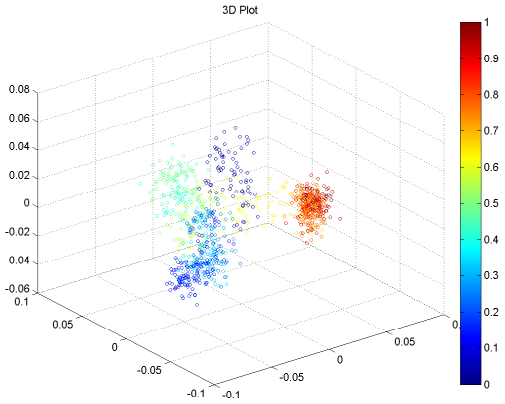


(a) 3D plot of Poly-A using the angular distance measure and MDS.

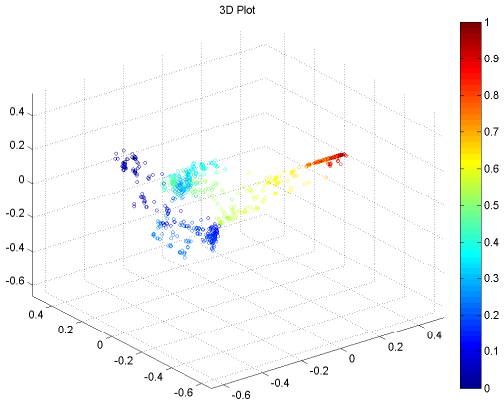


(b) 3D plot of Poly-A using the angular distance measure and nc-MDS.

Figure 4.3: 3D plots of Poly-A protein using angular distance measure presented to compare MDS and nonclassical MDS.

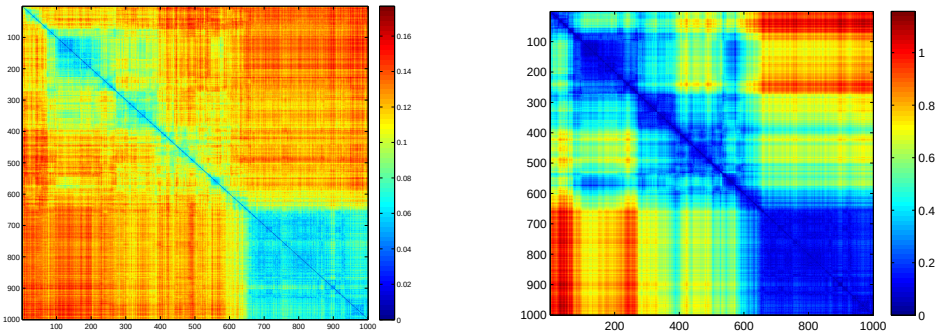


(a) 3D plot of Poly-A using the angular distance measure and MDS.



(b) 3D plot of Poly-A using the angular distance measure and Isomap.

Figure 4.4: 3D plots of Poly-A protein using the angular distance measure comparing MDS and Isomap.



(a) Pairwise distance matrix of Poly-A using the angular distance measure.

(b) Geodesic pairwise distance matrix of Poly-A using the angular distance.

Figure 4.5: Pairwise distance matrices of Poly-A protein using angular distance measure presented to compare MDS and Isomap.

Isomap changes the original pairwise distances by making them geodesic distances as explained in chapter 2. In our case, Isomap enhances the proximity of points, meaning that close points are closer and far points are further apart in the resulting pairwise matrix. Figure 4.5 shows the pairwise distance matrices before and after the geodesic distances have been calculated.

Our framework can also be used to investigate how multiple trajectories (replicates) of the same protein explore structural space in relation to each other. That is, it can help answer the question whether the trajectories overlap in structural space or not. This is accomplished by combining all the structures from multiple trajectories and then visualizing them in the reduced 3D space. Figure 4.6 shows

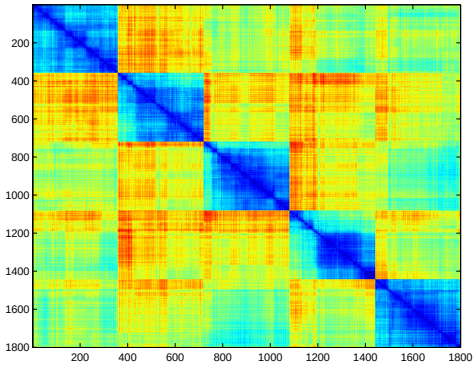
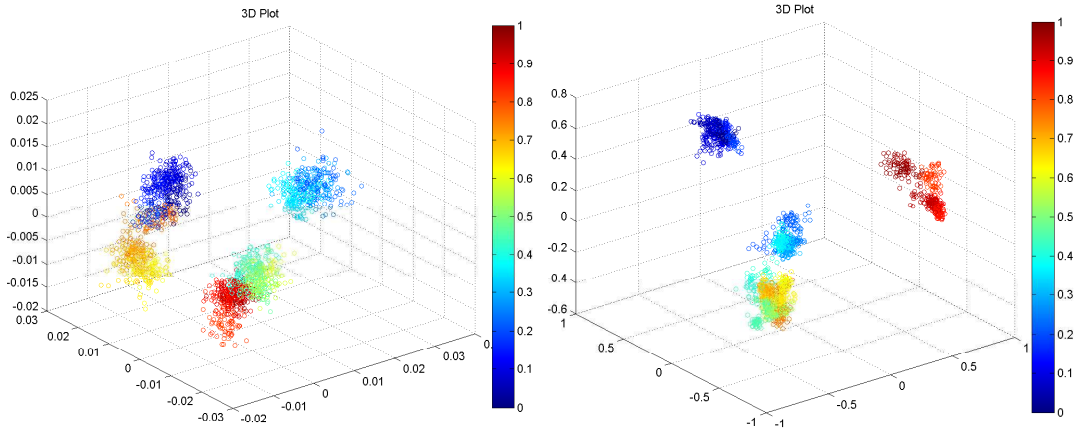


Figure 4.6: The pairwise distance matrix for five GLFG trajectories. The color indicates the distance between structures, red being far and blue being close. This is generated using angular distance measure.

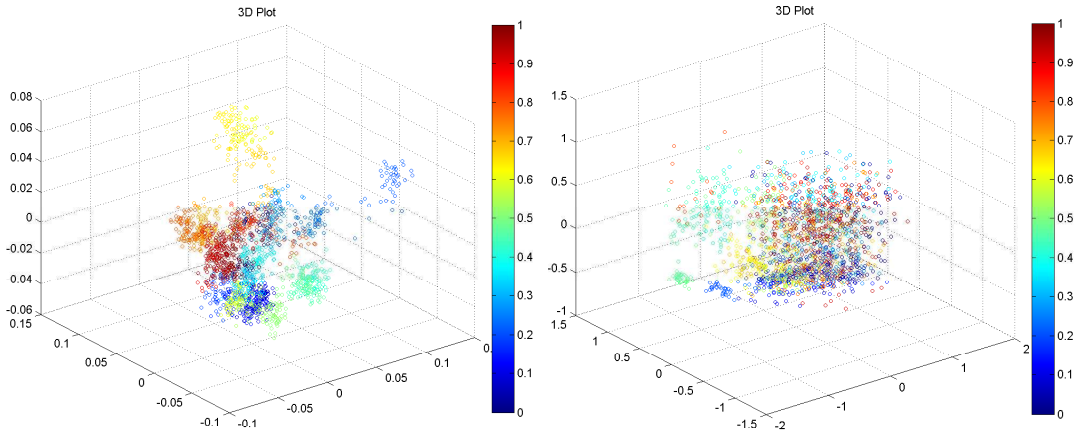
the pairwise distance matrix for five GLFG trajectories using angular distance measure and figure 4.7 visualizes the structures in 3D space after MDS for angular and RMSD distance measures. Figure 4.7 shows that the five trajectories do not overlap. Angular measure performs better in distinguishing all five trajectories, whereas RMSD has two of the trajectories overlapping which results in 4 distinct clusters.

Similar to GLFG, AXAG is analyzed by combining the structures from five trajectories. Unlike GLFG, the individual trajectories cannot be distinguished by observing the 3D plots which is shown in figure 4.8. This confirms the fact that AXAG is more disordered than GLFG.



(a) Angular (b) RMSD

Figure 4.7: 3D plot for five GLFG trajectories using RMSD and angular distance measures for MDS.



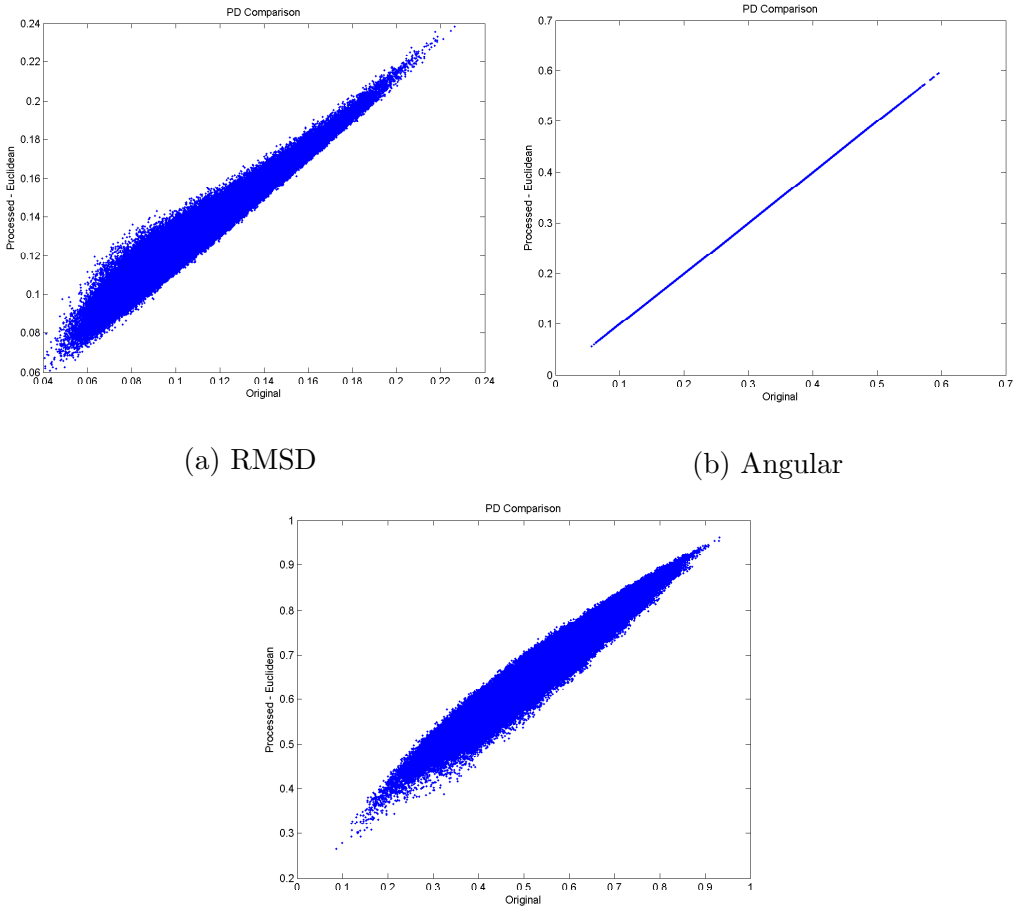
(a) Angular (b) RMSD

Figure 4.8: 3D plot for five AXAG trajectories using RMSD and angular distance measures for MDS.

4.2 Pairwise Distance Comparison

The original pairwise distances and the pairwise distances in the projected space are compared in this evaluation method to observe how well the relative ordering of the distances is preserved. This is done using a plot created as follows. The original distances are sorted and used as the x values. The corresponding pairwise distances between the points in the projected space are calculated using Euclidean distance and used as the y values. If the resulting plot is a one to one and monotonically increasing this indicates that the order of the pairwise distances is preserved. A benefit of this evaluation is that we can observe how well the distances are preserved for a selected target dimension D.

We apply pairwise distance comparison to the RC dataset. Figure 4.9 shows the pairwise distance plots for RC-10 using RMSD, angular and dihedral distance measures. Angular distance measure achieves one-to-one monotonically increasing pairwise distance plot for D dimensions where D is the number of positive eigenvalues. Recall, that in MDS, the maximum number of dimensions is limited by the maximum number of eigenvalues. For RMSD and dihedral distance measures, even when maximum number of dimensions are used for the pairwise distance plots, the results still do not achieve one-to-one correspondence and monotonicity.



(a) RMSD

(b) Angular

(c) Dihedral

Figure 4.9: Pairwise distance plots of 3 different distance measures for RC-10. The x-axis is the original pairwise distances. The y-axis is the pairwise distances in the projected space.

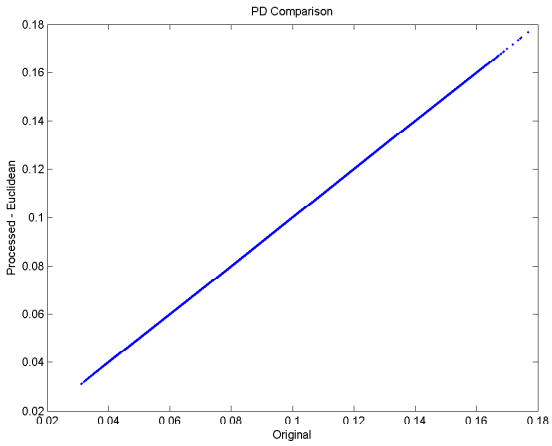


Figure 4.10: The x-axis is the original pairwise distances. The y-axis is the pairwise distances in the projected space. This is for Poly-A using angular ISD and MDS.

Figure 4.10 shows an example pairwise distance comparison plot for Poly-A when reduced to 524 dimensions, which is the number of positive eigenvalues, using MDS with angular distance used as the interstructure distance measure. We can observe that this plot is one to one and monotonically increasing which tells us that the ordering of the pairwise distances is perfectly preserved using 524 dimensions.

Figure 4.11 shows the pairwise distance comparison for Poly-A after using MDS to reduce the dimension to 524 but using the dihedral measure. This is not one-to-one nor monotonically increasing due to the fact that the dihedral ISD is not Euclidean (because of the angle wrap-around).

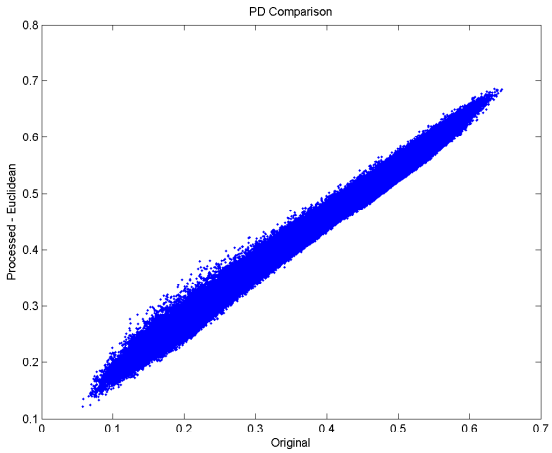


Figure 4.11: The x-axis is the original pairwise distances. The y-axis is the pairwise distances in reduced space. This is for Poly-A using dihedrals and MDS.

While the dihedral ISD does not preserve the ordering of all the pairwise distances, it does do this for most. When a density plot is used to visualize the frequency of points, it can be observed that the concentration is along a one to one monotonically increasing line. This is shown in figure 4.12 using a density plot of the protein Poly-A using MDS and dihedral angles as the distance measure.

4.3 Correlation

The pairwise distance comparison plots in the previous section allowed us to visually determine how well a given combination of ISD and dimensionality reduction methods preserves the relative ordering of and even the relative magnitudes

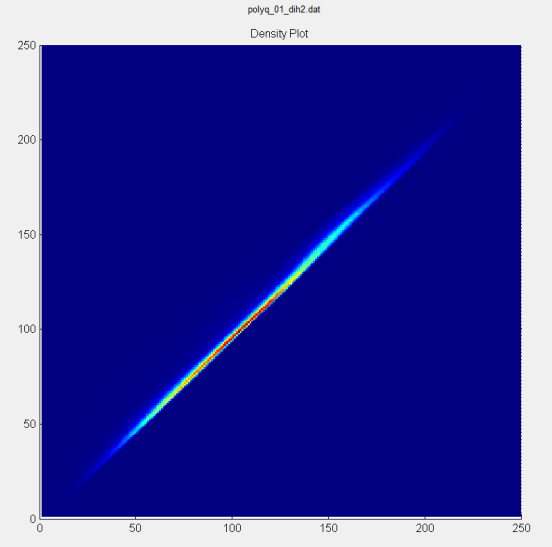


Figure 4.12: Density plot the pairwise distances of Poly-A using the dihedral angles. The x-axis is the original pairwise distances. The y-axis is the pairwise distances in the projected space.

of a trajectory. However, it is visual and therefore qualitative and can only be done for a single target dimension at a time. In this section, we seek to summarize how well the pairwise distances are preserved using a single value so that we can plot it as the target dimension is varied.

Correlation is a measure of how well one set of values linearly predicts another related set (up to a scaling factor). It measures how well a (non-vertical, non-horizontal) line can be fit to the plot of one set of values versus the other. Given a set of x values and a set of corresponding y values, the correlation (coefficient) is computed as [9]

$$\text{Corr} = \frac{n\sum_{i=1}^{n}(x_i y_i) - \sum_{i=1}^{n}(x_i)\sum_{i=1}^{n}(y_i)}{\sqrt{n\sum_{i=1}^{n}(x_i^2) - (\sum_{i=1}^{n}(x_i))^2}\sqrt{n\sum_{i=1}^{n}(y_i^2) - (\sum_{i=1}^{n}(y_i))^2}} \tag{4.1}$$

We can compute the correlation of our pairwise distances comparison plots to estimate how well pairwise distances are preserved for a particular target dimension. The x values in equation 4.1 are taken as the original pairwise distances and the y values are taken as the distance in the reduced space. This value ranges from 0 to 1 where a higher value indicates a better linear fit. We can then plot correlation versus target dimension to get an idea of the dimension of the trajectory.

We first compute and plot correlation versus target dimension for the RC dataset to calibrate the analysis. Figure 4.13 shows the correlation plots for RC-10 after MDS using RMSD, angular and dihedral distance measures. Recall, we

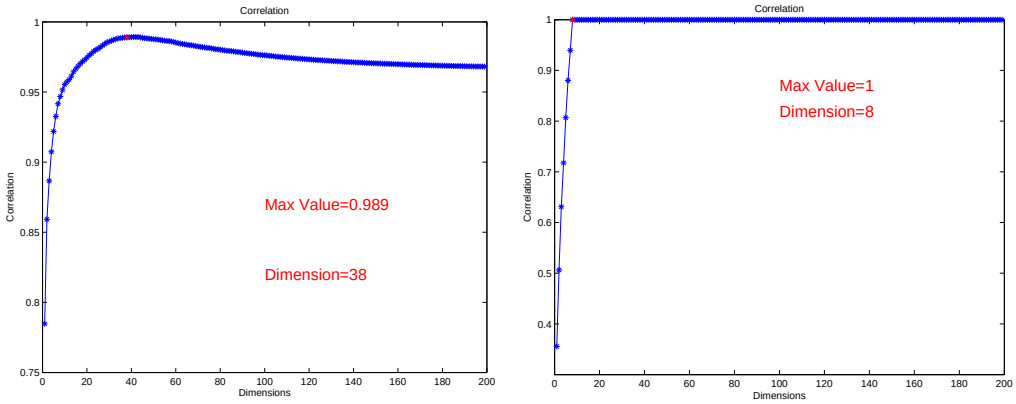
expect the dimension of this trajectory to be 30, 8 and 14 for these measures respectively. We see that the correlation reaches a value of 1 at 8 dimensions for the angular distance measure. Correlation never reaches 1 for dihedral meaning that the relative ordering and scale of the pairwise distances is never preserved by MDS and the dihedral distance measure no matter how large the target dimension. The correlation does reach a maximum value at 14 dimensions, the expected value.

The correlation plot for RMSD is less clear. It does not reach a maximum value at 30 and it actually decreases with increasing dimension.

These plots demonstrate two things. First, that MDS does not preserve pairwise distances when the original distances are computed using RMSD and dihedral distance measures no matter what the target dimension is. This is because RMSD and dihedral are not Euclidean. And, second, that RMSD is poor distance measure particularly when computing the pairwise distances for input dimensionality reduction methods.

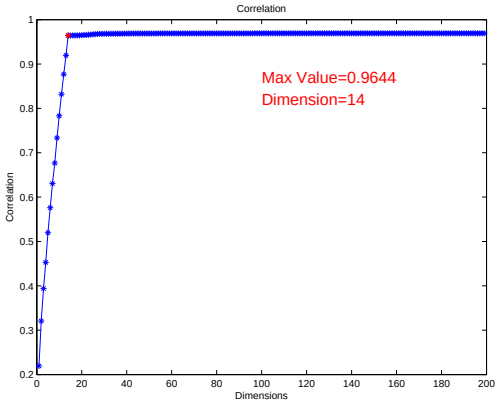
Table 4.1 shows the maximum correlation values for all three RC trajectories for the various distance measures. Table 4.2 shows the maximum correlation values for Poly-A, Poly-G and Poly-Q trajectories for the various distance measures.

Figure 4.14 shows the correlation plots for Poly-A. As is often the case with our problem, even though the analysis framework performed as expected on the



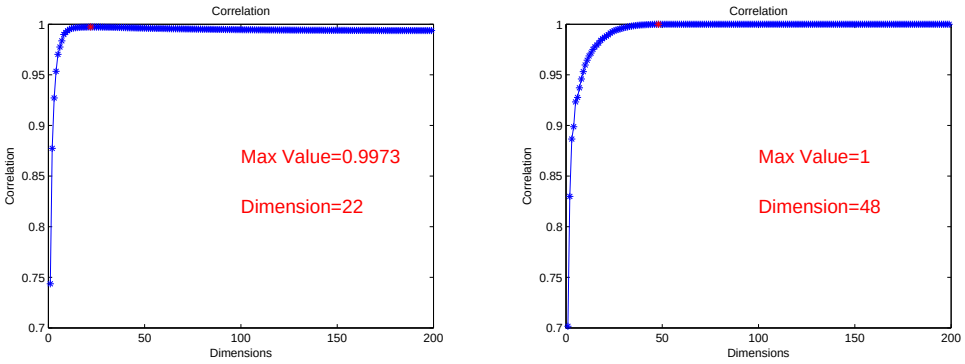
(a) RMSD

(b) Angular



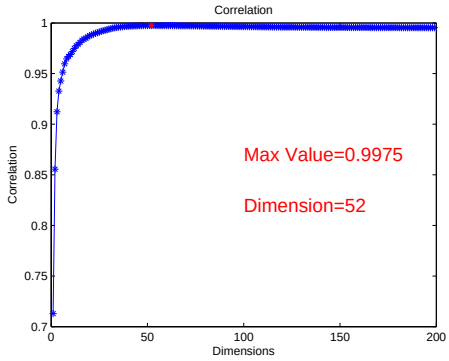
(c) Dihedral

Figure 4.13: Correlation evaluations of 3 different distance measures for RC-10.



(a) RMSD

(b) Angular



(c) Dihedral

Figure 4.14: Correlation evaluations of 3 different distance measures for Poly-A.

Synthetic Trajectory	Trajectory Size	Beads	Dimensions	Max Value
RCn10-RMSD	1000 Frames	10	38	0.9890
RCn10-Ang	1000 Frames	10	8	1
RCn10-Dih	1000 Frames	10	14	0.9644
RCn30-RMSD	1000 Frames	30	45	0.9887
RCn30-Ang	1000 Frames	30	28	1
RCn30-Dih	1000 Frames	30	54	0.9624
RCn50-RMSD	1000 Frames	50	39	0.9903
RCn50-Ang	1000 Frames	50	48	1
RCn50-Dih	1000 Frames	50	94	0.9628

Table 4.1: The maximum correlation values and the corresponding dimensions for random chains of size 10, 30 and 50 using different ISD measures.

Real Trajectory	Trajectory Size	Beads	Dimensions	Max Value
PolyA-RMSD	1000 Frames	50	22	0.9973
PolyA-Ang	1000 Frames	50	48	1
PolyA-Dih	1000 Frames	50	52	0.9975
PolyG-RMSD	1000 Frames	50	30	0.9980
PolyG-Ang	1000 Frames	50	48	1
PolyG-Dih	1000 Frames	50	75	0.9906
PolyQ-RMSD	1000 Frames	50	33	0.9999
PolyQ-Ang	1000 Frames	50	48	1
PolyQ-Dih	1000 Frames	50	29	0.9969

Table 4.2: The maximum correlation values and the corresponding dimensions for Poly-A, Poly-G and Poly-Q using different ISD measures.

synthetic data, the results on real data are rarely straightforward. Angular reaches a correlation of 1 in 48 dimensions as expected.

4.4 Kruskal's Stress Measure

Kruskals stress measure also known as stress-1 is similar to correlation and helps identify the amount of invariance (variance that is not explained) given a

target dimension D. This method is more useful in non-classical MDS where the data is processed using isotonic regression where the points are fit to a monotonically increasing line. It can also be applied to linear classical MDS. Figure 4.15 shows an example plot for Poly-A with angular distances used as the interstructure distance measure. From these plots we can observe that most of the time after three dimensions less than 0.1 percent of the variance is unexplained, which tells us that the 3D plot of this trajectory will reflect most of the data with good accuracy.

The stress measurement is computed as follows:

$$\text{Stress} = \sqrt{\frac{\sum_{i=1}^{n}\sum_{j=1}^{n}(f(x_{ij}) - d_{ij})^2}{\sum_{i=1}^{n}\sum_{j=1}^{n}(d_{ij})^2}} \tag{4.2}$$

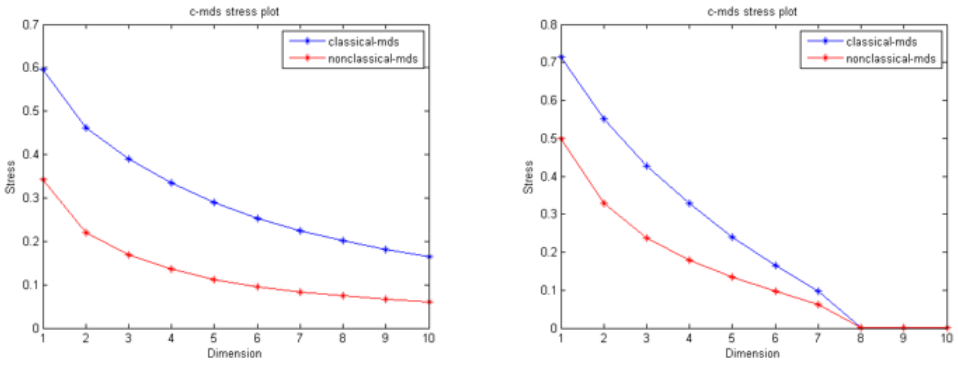
In the equation, d_{ij} refers to the Euclidean distance, across all dimensions, between points i and j in the projected space, $f(x_{ij})$ is some function of the input data, and the divisor is a scaling factor used to keep stress values between 0 and 1. When MDS perfectly projects the input data, $f(x_{ij}) - d_{ij}$ is 0 for all i and j, so stress is zero. Thus, the smaller the stress, the better the representation [4].

The transformation of the input values $f(x_{ij})$ depends on whether its classical or non-classical MDS. In metric scaling, $f(x_{ij}) = x_{ij}$. In other words, the raw input data is compared directly to the projected distances. In non-classical MDS, $f(x_{ij})$ is a weakly monotonic transformation of the input data that minimizes the

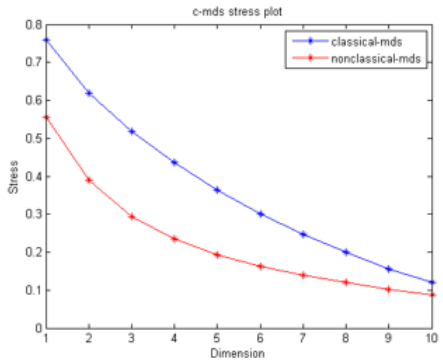
stress function [4]. The monotonic transformation is computed via "monotonic regression", also known as "isotonic regression" [6].

Similar to correlation evaluation, it is beneficial to examine the results of the synthetic random chains using the stress measurement. It can be seen from the plots in figure 4.15 that stress follows the same pattern as correlation. The angular distance measure is the only one that reaches zero stress at the exact dimension of 8. Similarly, both RMSD and the dihedral measure does not reach zero stress where dihedral achieves lowest stress value at 14 dimensions. For the lower dimensions, non-classical MDS achieves a lower stress value, indicating that it can be beneficial to use non-classical MDS if the aim is to represent or evaluate the movement of these random chains in a relatively low dimension.

In the case of Poly-A, the results of the stress evaluation are as expected. For all the interstructure distance measures, non-classical MDS provides lower stress values than classical-MDS. The results of the evaluation can be observed in figure 4.16. This indicates that in lower dimensions, using non-classical MDS captures the movement of the protein and the pairwise distances between structures better than classical MDS.

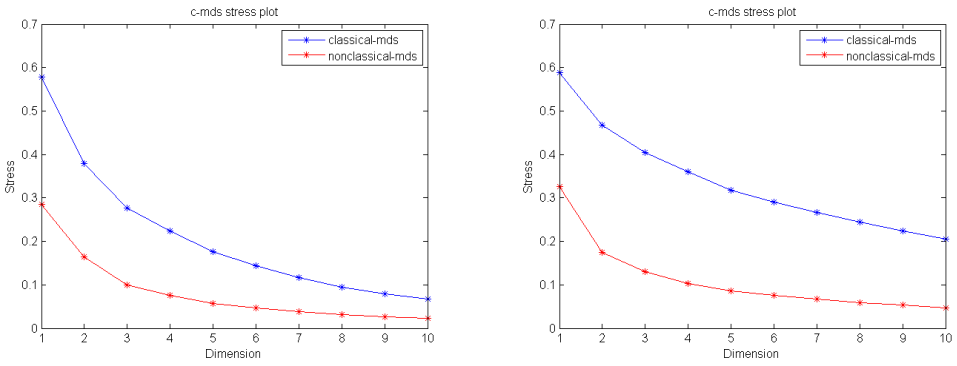


(a) RMSD (b) Angular



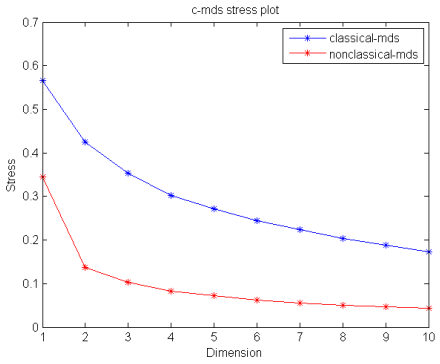
(c) Dihedral

Figure 4.15: Stress evaluations of 3 different distance measures for RC-10 comparing classical and non-classical MDS.



(a) RMSD

(b) Angular



(c) Dihedral

Figure 4.16: Stress evaluations of 3 different distance measures for PolyA comparing classical and non-classical MDS.

Chapter 5

Conclusions

5.1 Overview and Summary

This thesis investigated dimensionality reduction methods for analyzing the dynamics of protein simulations, particularly disordered proteins which do not fold into fixed shapes but are thought to perform their functions through their movements. The question of whether these disordered proteins are limited to lower dimensional dynamics is often asked and by using dimensionality reduction this thesis aimed to provide broader understanding of the space and the dimensions they explore.

The dimensionality reduction methods we considered required pairwise differences and so we investigated different interstructure distance measures of root mean square distance (RMSD), angular distance and dihedral angles distance to compute the distance between two protein structures. We also investigated

different dimensionality reduction techniques: classical multidimensional scaling (MDS), non-classical MDS and Isomap.

The following workflow was used to do the analysis of the proteins:

- The movement of a protein is simulated using Molecular Dynamics simulations. The structures that result from these simulations, which we call frames, is considered the initial data.
- The pairwise distances between each of these frames are calculated using different interstructure distance methods.
- This pairwise distance matrix is given as an input to one of the dimensionality reduction methods.
- The output of the dimensionality reduction is a set of points in a standard Euclidean space where each point corresponds to a structure. These points are then visualized. They are also analyzed in varying dimensions to see how faithfully they capture the pairwise distances between the original data.

The aim is to answer which interstructure distance method and dimensionality reduction method is best for observing degree of disorder and preservation of pairwise distances with respect to a particular dimension. The following conclusions can be reached from the experiments:

- For random chains, angular and dihedral distance measures are informative when combined with MDS.
- For real proteins, angular and in some cases dihedral distance measures are informative when combined with MDS.
- Non-classical MDS is not particularly useful in identifying the degree of disorder, but it can be perform better than MDS if the movement of a protein needs to be constrained and examined in a low dimensional space.
- Isomap is useful in limited cases regarding visualization. If a trajectory's movement can be sufficiently captured in 3 degrees of freedom, the Isomap plot provides good visualization.

Additionally there are points of caution and general take away ideas:

- 3D plots of the results of the dimensionality reduction provide insight into the degree of disorder.
- Proteins that are less disordered tend to have more clustered and distinguishable movement patterns that are reflected in the plots. This may prove useful in the early stages of the analysis and can be used as a starting point.
- The dimension corresponding to the maximum correlation between input and projected distances does not always indicate the total degree of freedom.

- The less sharp the rise in of the correlation graph for a trajectory, the more disordered the movement of that trajectory is. A steady rise in the correlation graph indicates that large number of atoms in the protein are in rapid movement resulting in higher dimensionality.

- Trajectories capture a limited duration of the movement of the protein, therefore throughout the thesis the trajectories used may not fully reflect the general behavior and dynamics of that protein.

Bibliography

[1] About GROMACS. http://www.gromacs.org/About_Gromacs. [Online; accessed 11-May-2015].

[2] Gmx angle - GROMACS manual. http://manual.gromacs.org/programs/gmx-angle.html. [Online; accessed 11-May-2015].

[3] Multidimensional scaling. http://www.mathworks.com/help/stats/multidimensional-scaling.html. [Online; accessed 11-May-2015].

[4] Steve Borgatti. *Multidimensional scaling*. http://www.analytictech.com/networks/mds.htm. [Online; accessed 11-May-2015].

[5] Lydia E. Kavraki. *Representing Proteins in Silico and Protein Forward Kinematics*. OpenStax CNX, 2007.

[6] J.B. Kruskal. Nonmetric multidimensional scaling: A numerical method. *Psychometrika*, 29(2):115–129, 1964.

[7] J.B. Kruskal and M. Wish. *Multidimensional Scaling*. no. 11 in vol. 07. SAGE Publications, 1978.

[8] Joshua B. Tenenbaum, Vin de Silva, and John C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

[9] Wikipedia. Correlation and dependence. http://en.wikipedia.org/wiki/Correlation_and_dependence, 2015. [Online; accessed 12-May-2015].

[10] Wikipedia. Multidimensional scaling. http://en.wikipedia.org/wiki/Multidimensional_scaling, 2015. [Online; accessed 12-May-2015].

Appendix A

Additional Plots

This appendix contains additional plots for Poly-A, Poly-G and Poly-Q. Four plots are shown for each protein and each distance measure: a 3D plot, a correlation plot, a pairwise distance comparison plot and the plot of the eigenvalues that result from the MDS projection. All results are for MDS. The pairwise distance comparison plots are for the maximum dimension identified in the correlation plots.

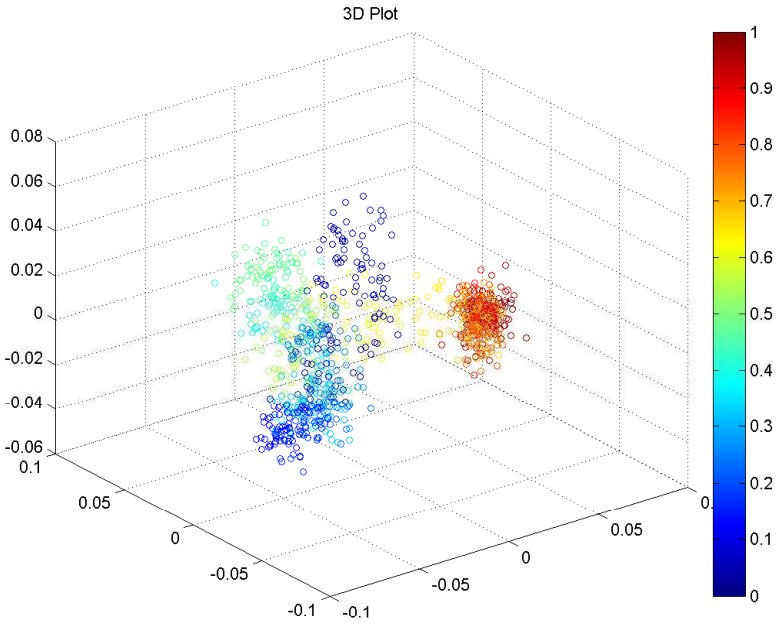


Figure A.1: 3D plot of Poly-A using angular ISD and MDS.

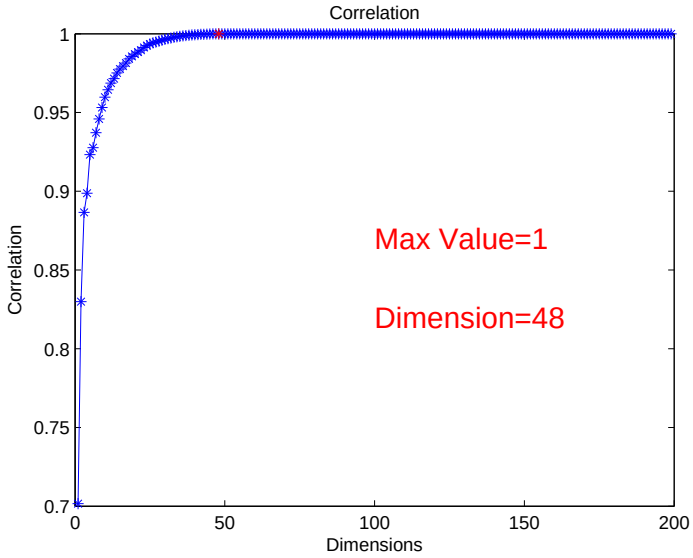


Figure A.2: Correlation plot of Poly-A using angular ISD and MDS.

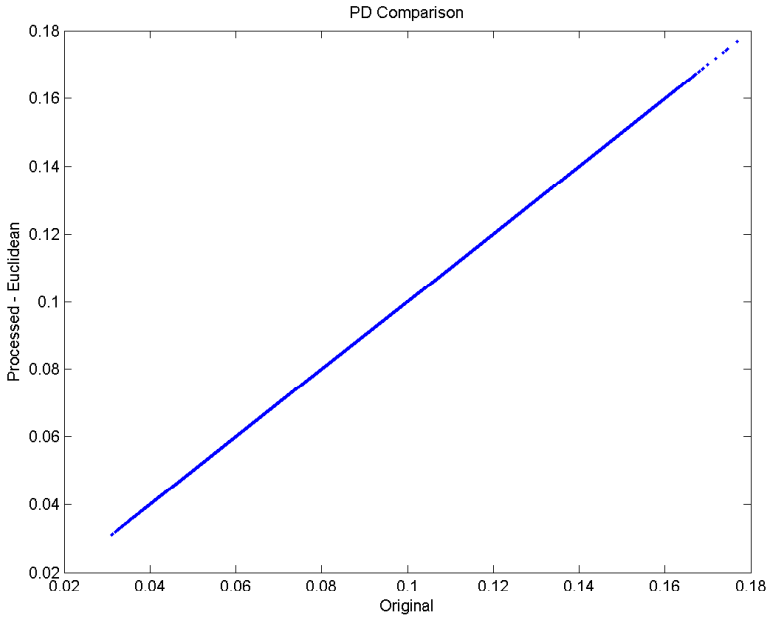


Figure A.3: Pairwise distance plot of Poly-A using angular ISD and MDS.

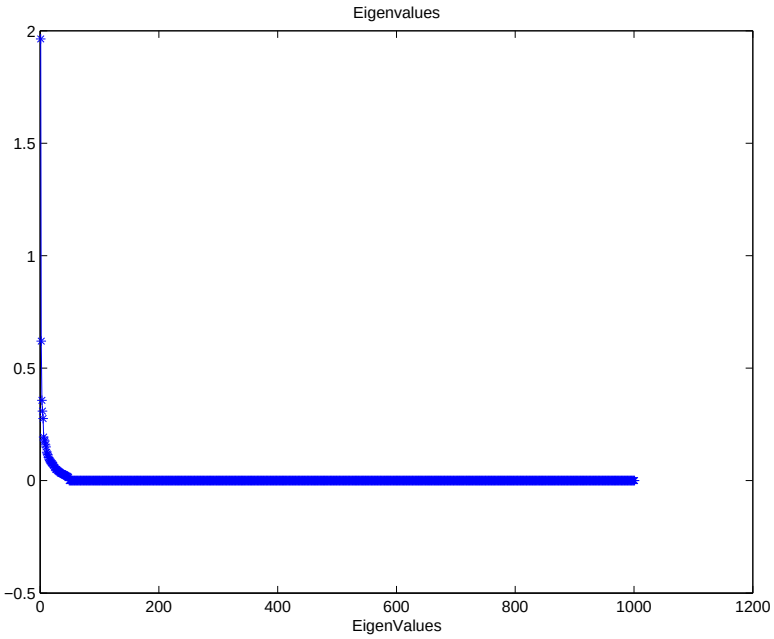


Figure A.4: Eigenvalues of Poly-A using angular ISD and MDS.

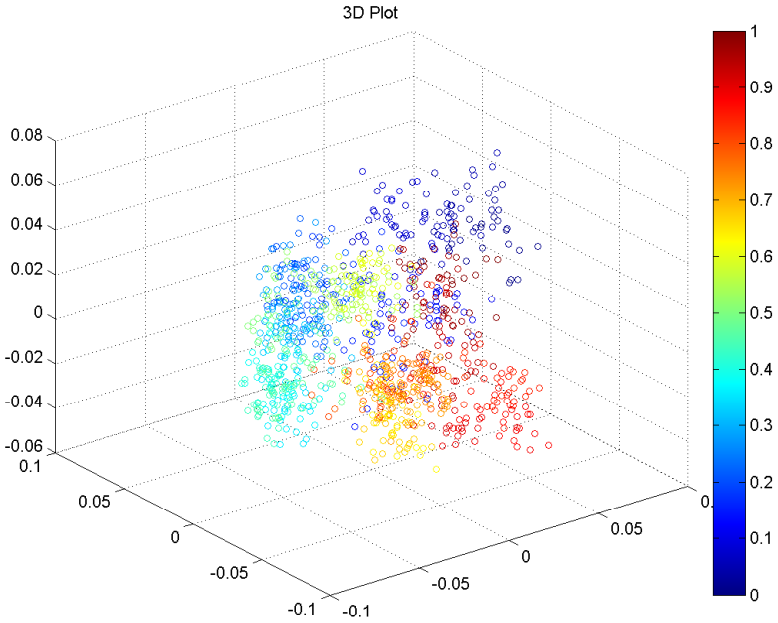


Figure A.5: 3D plot of Poly-G using angular ISD and MDS.

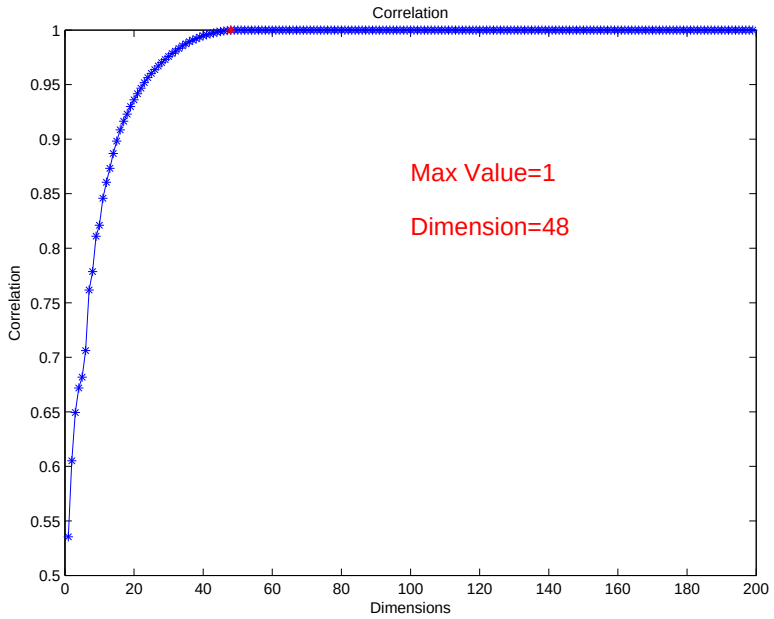


Figure A.6: Correlation plot of Poly-G using angular ISD and MDS.

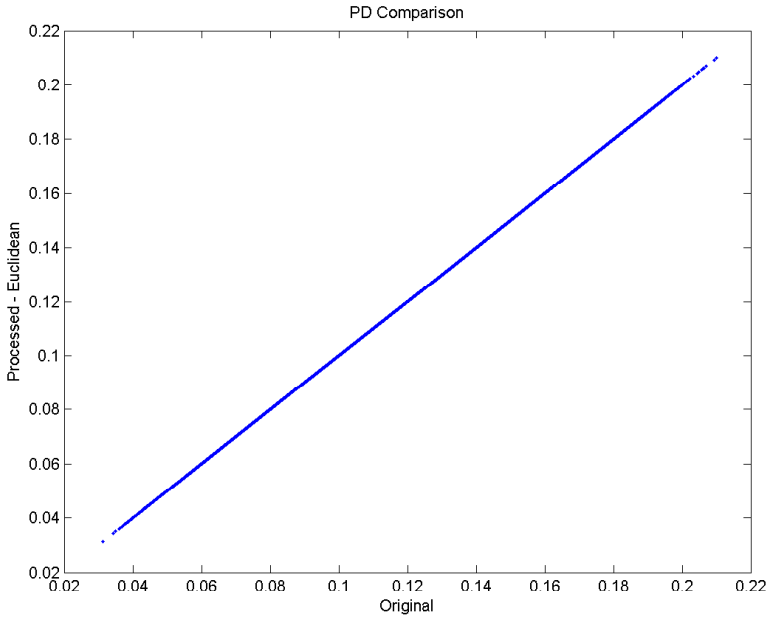


Figure A.7: Pairwise distance plot of Poly-G using angular ISD and MDS.

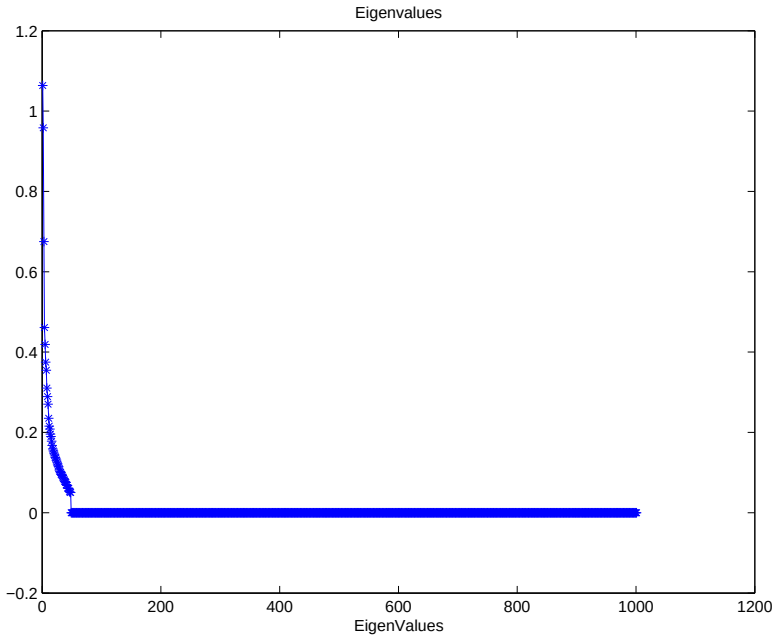


Figure A.8: Eigenvalues of Poly-G using angular ISD and MDS.

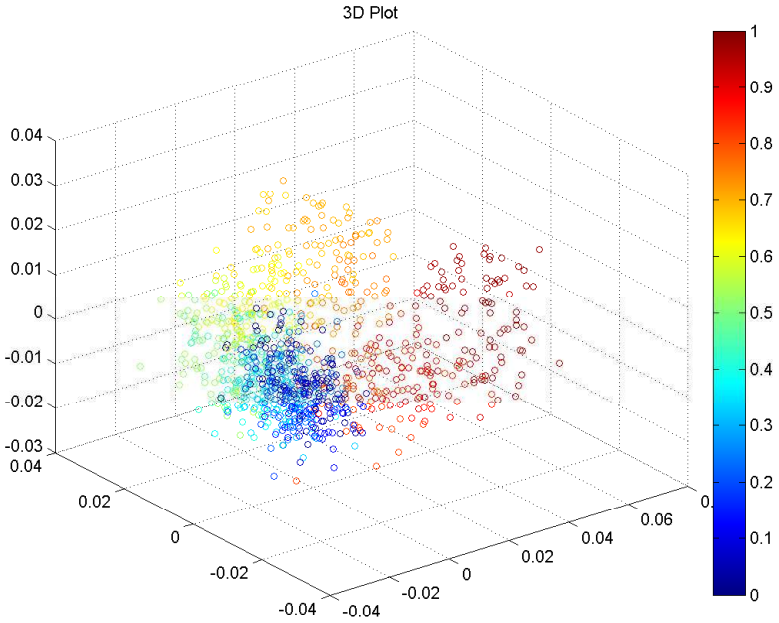


Figure A.9: 3D plot of Poly-Q using angular ISD and MDS.

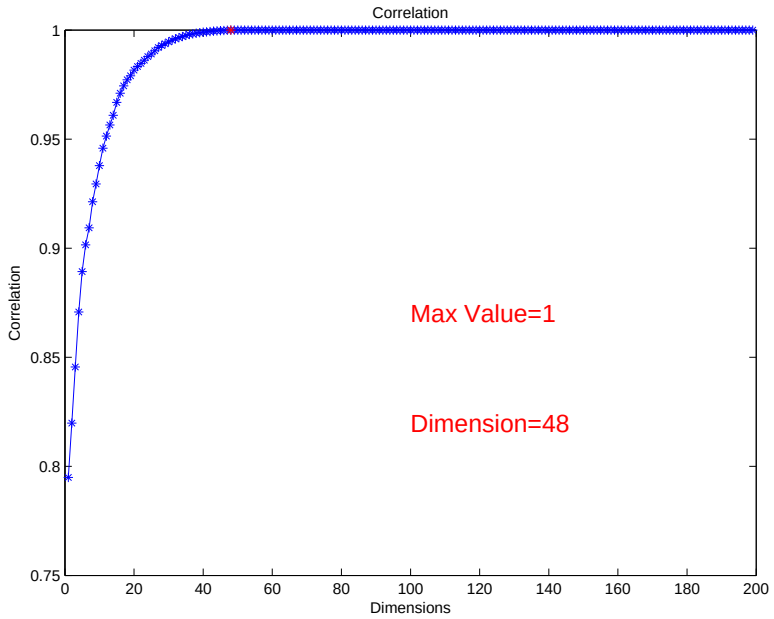


Figure A.10: Correlation plot of Poly-Q using angular ISD and MDS.

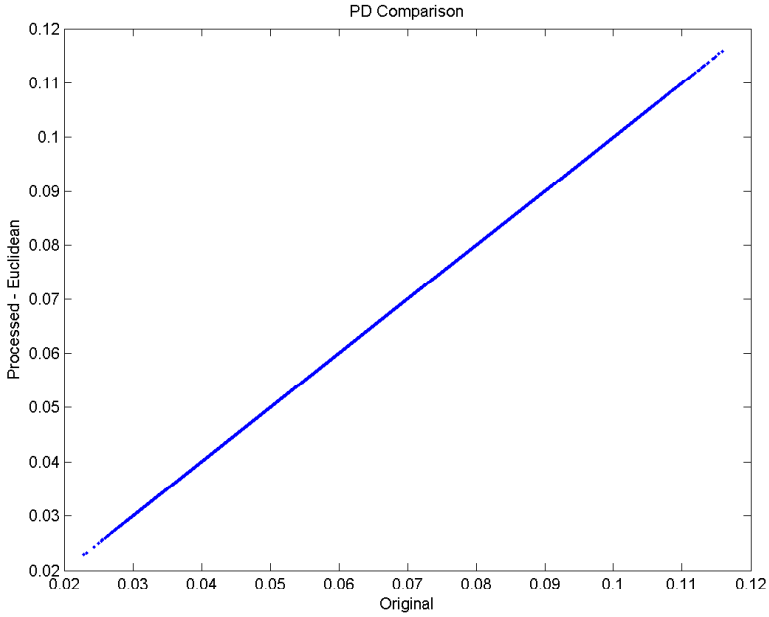


Figure A.11: Pairwise distance plot of Poly-Q using angular ISD and MDS.

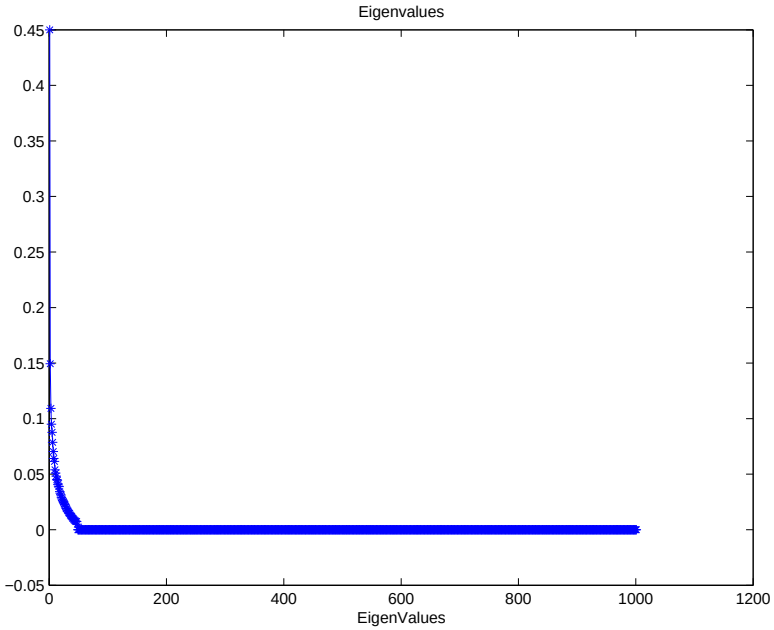


Figure A.12: Eigenvalues of Poly-Q using angular ISD and MDS.

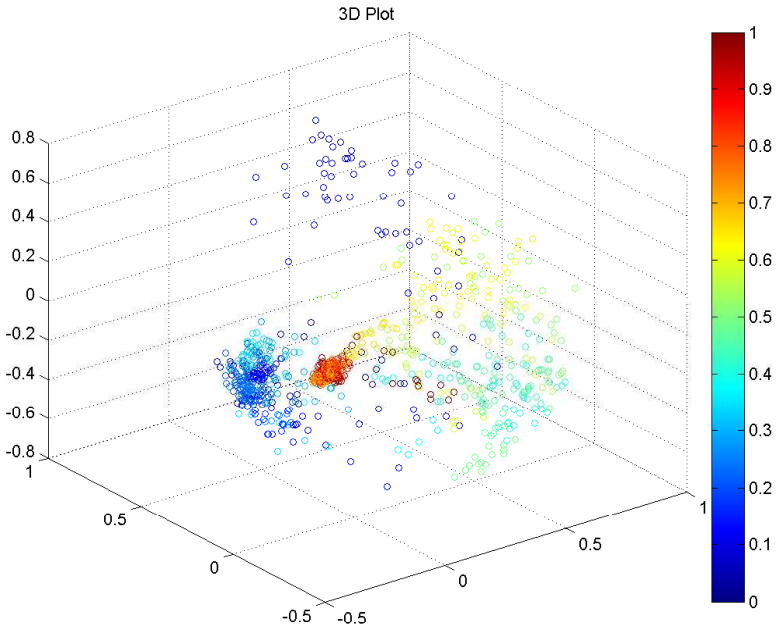


Figure A.13: 3D plot of Poly-A using RMSD ISD and MDS.

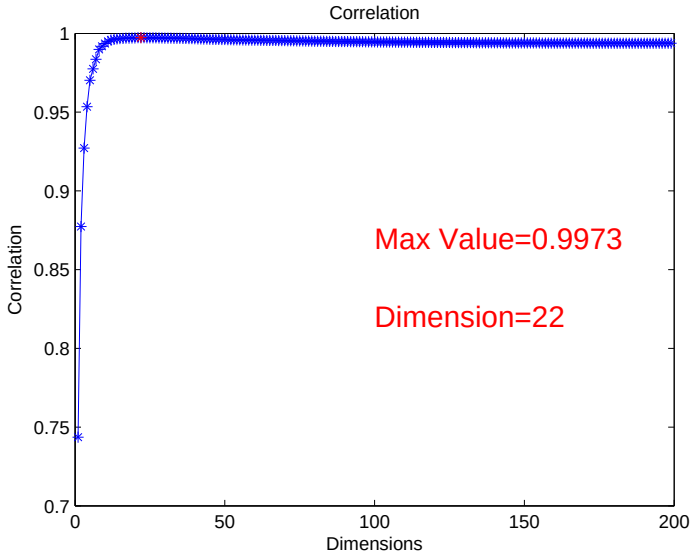


Figure A.14: Correlation plot of Poly-A using RMSD ISD and MDS.

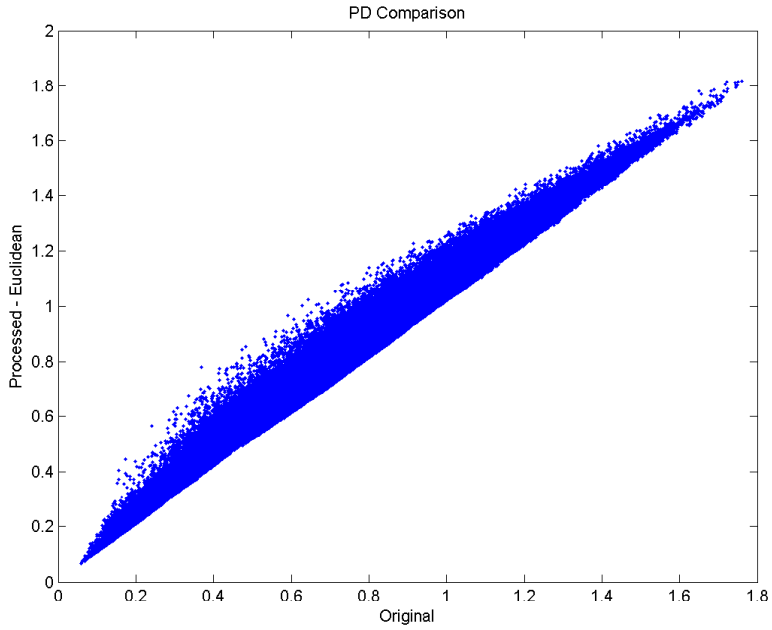


Figure A.15: Pairwise distance plot of Poly-A using RMSD ISD and MDS.

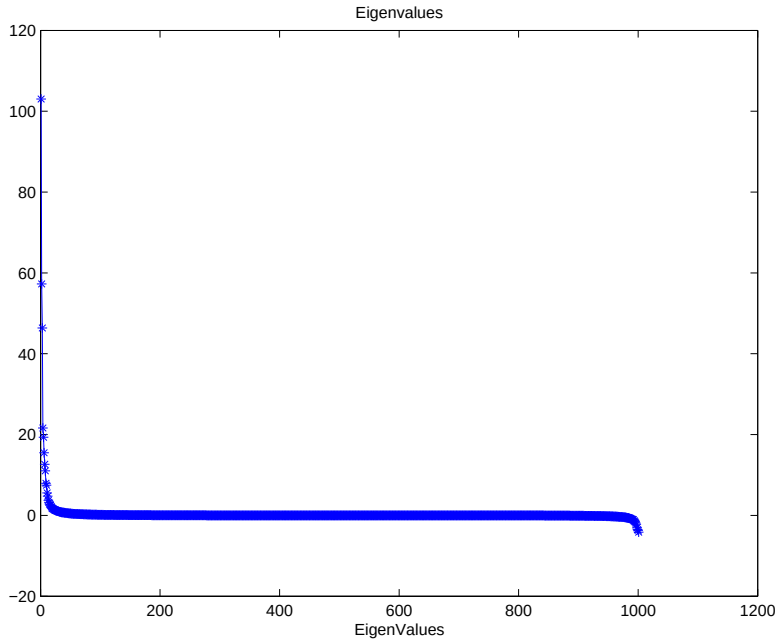


Figure A.16: Eigenvalues of Poly-A using RMSD ISD and MDS.

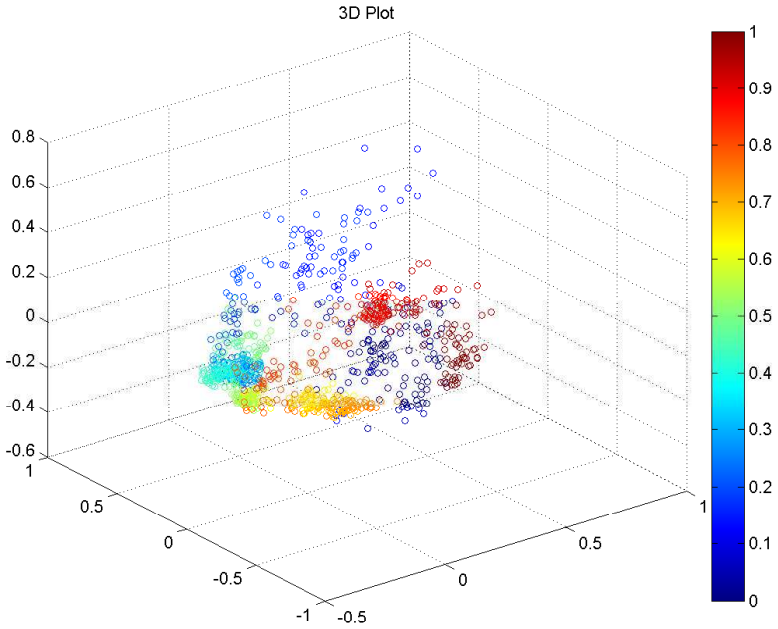


Figure A.17: 3D plot of Poly-G using RMSD ISD and MDS.

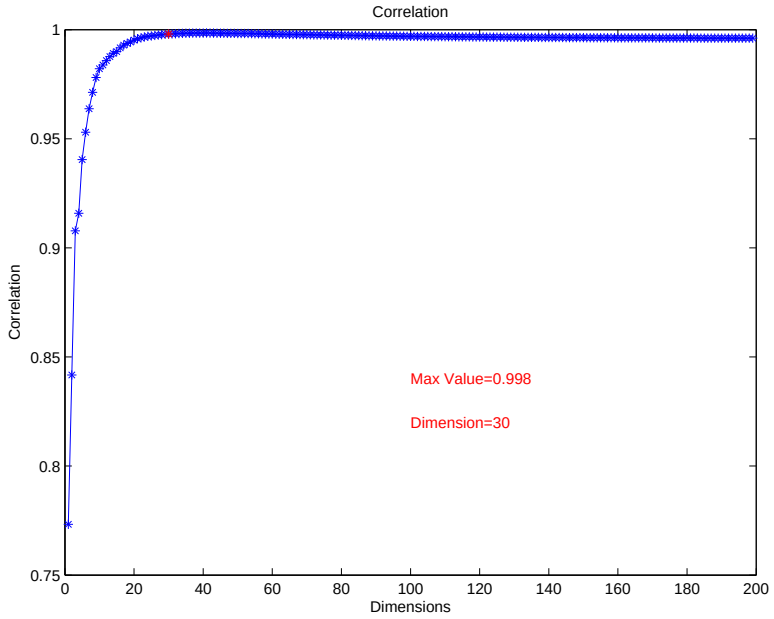


Figure A.18: Correlation plot of Poly-G using RMSD ISD and MDS.

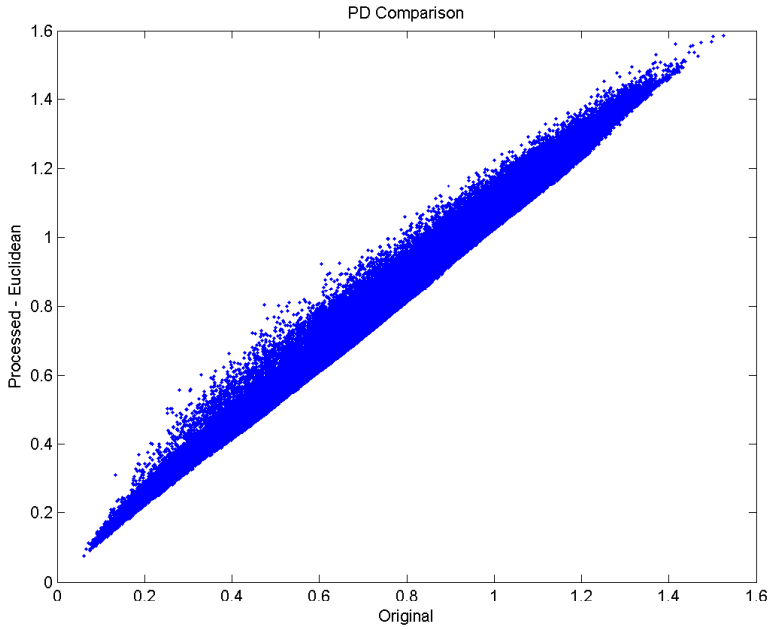


Figure A.19: Pairwise distance plot of Poly-G using RMSD ISD and MDS.

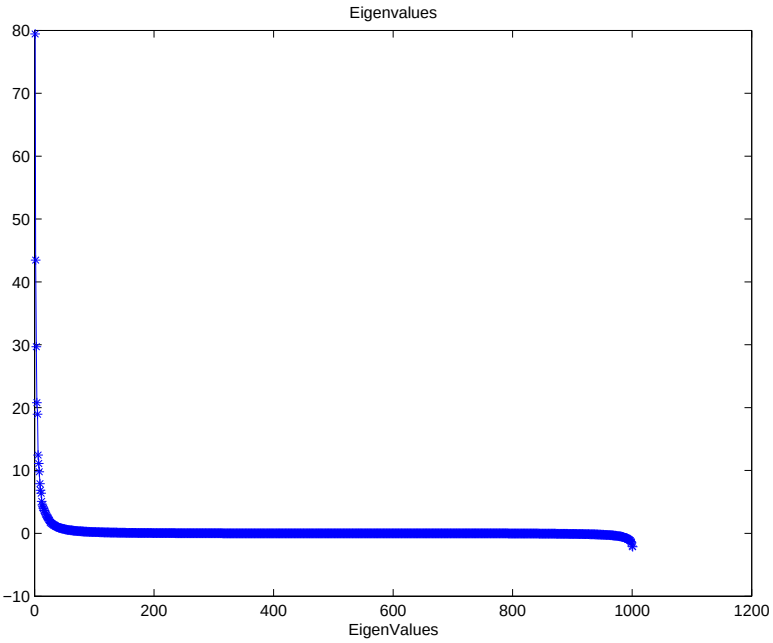


Figure A.20: Eigenvalues of Poly-G using RMSD ISD and MDS.

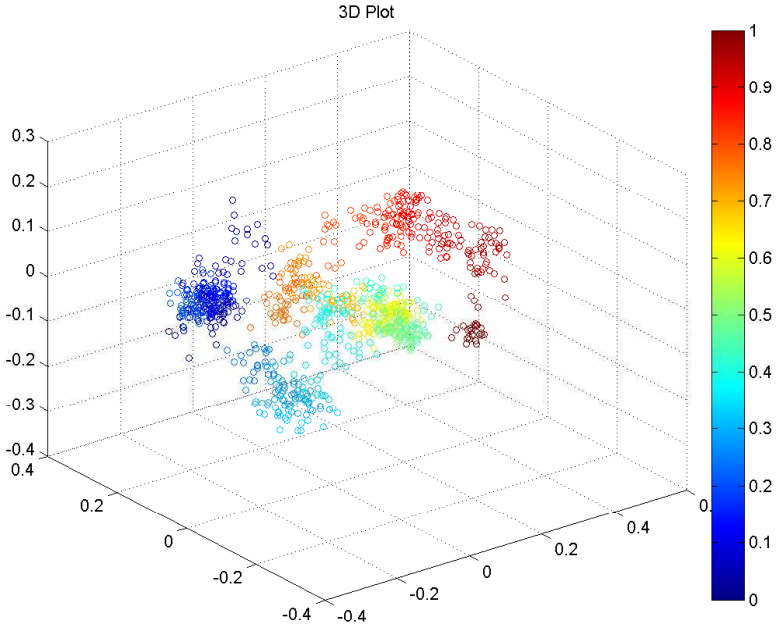


Figure A.21: 3D plot of Poly-Q using RMSD ISD and MDS.

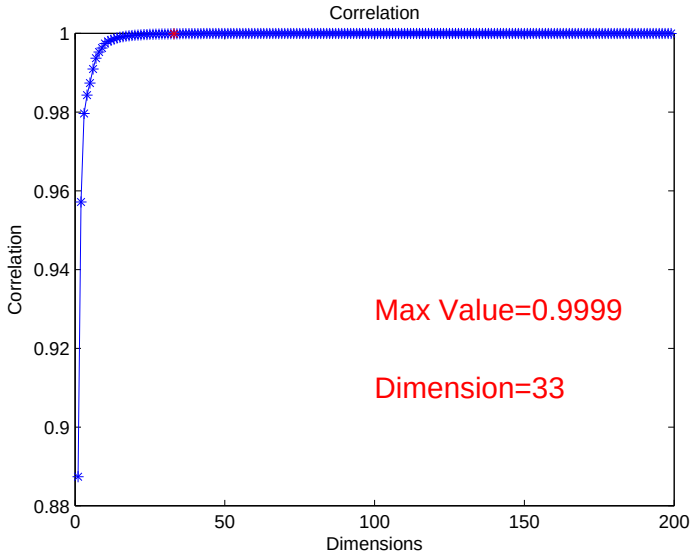


Figure A.22: Correlation plot of Poly-Q using RMSD ISD and MDS.

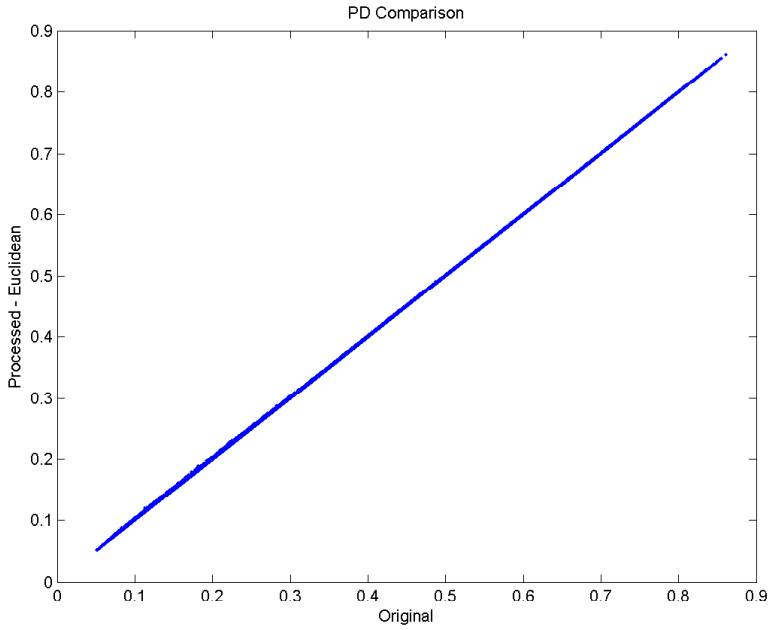


Figure A.23: Pairwise distance plot of Poly-Q using RMSD ISD and MDS.

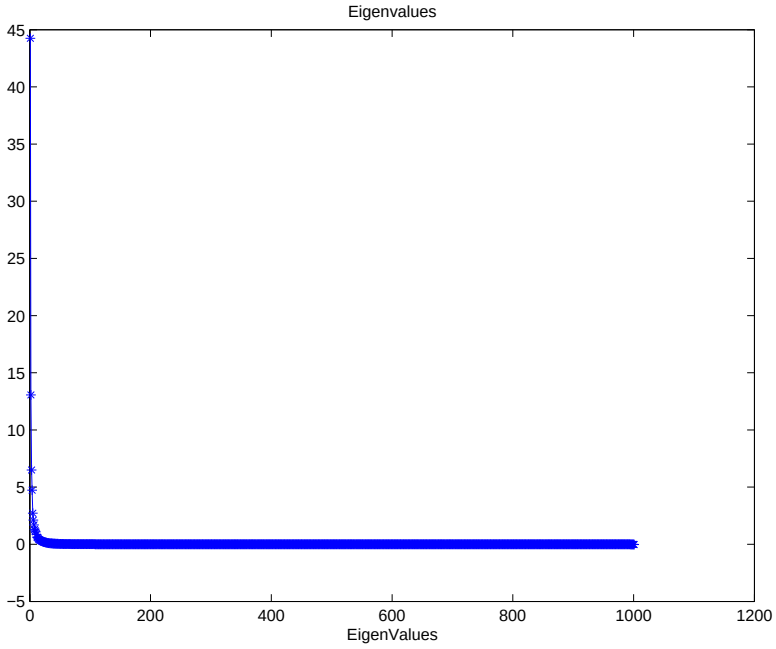


Figure A.24: Eigenvalues of Poly-Q using RMSD ISD and MDS.

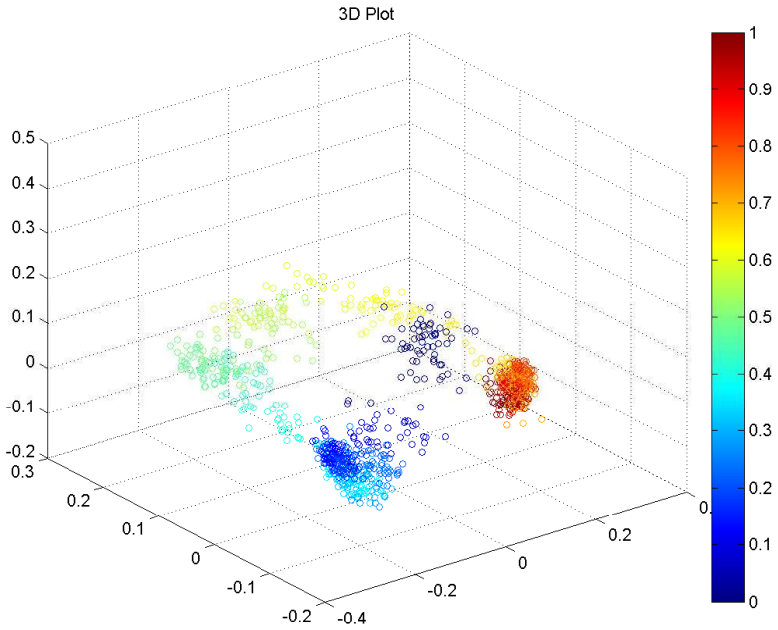


Figure A.25: 3D plot of Poly-A using dihedral ISD and MDS.

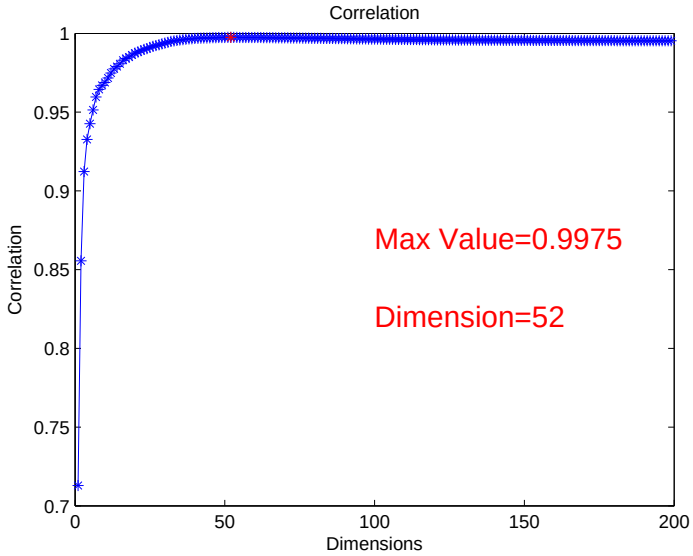


Figure A.26: Correlation plot of Poly-A using dihdral ISD and MDS.

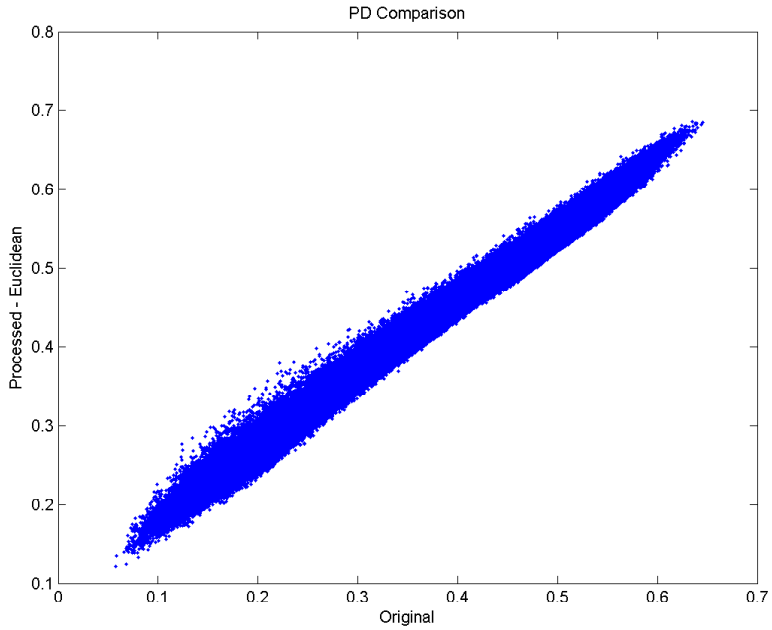


Figure A.27: Pairwise distance plot of Poly-A using dihedral ISD and MDS.

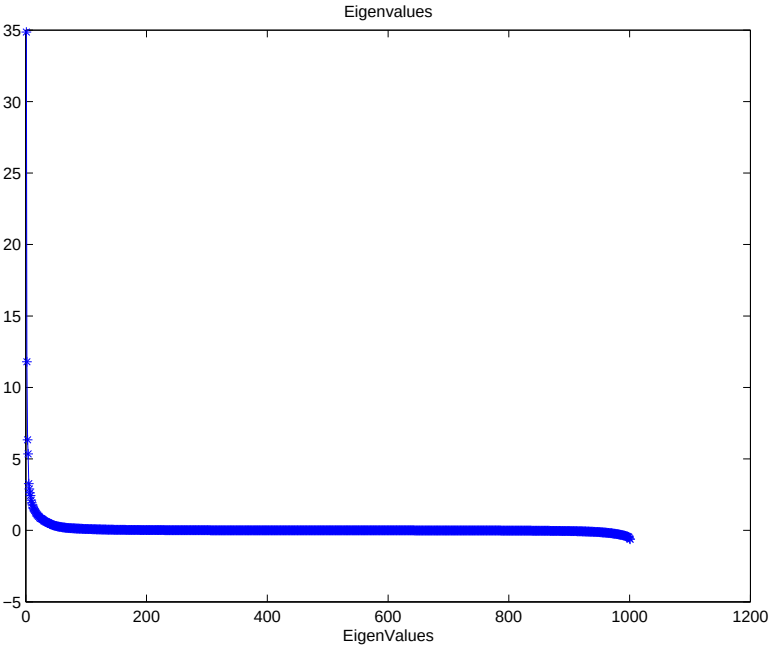


Figure A.28: Eigenvalues of Poly-A using dihedral ISD and MDS.

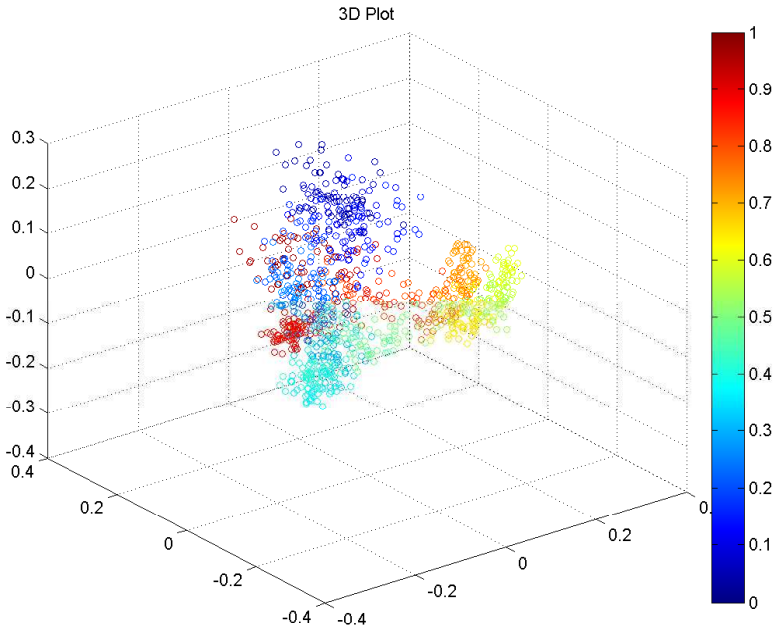


Figure A.29: 3D plot of Poly-G using dihedral ISD and MDS.

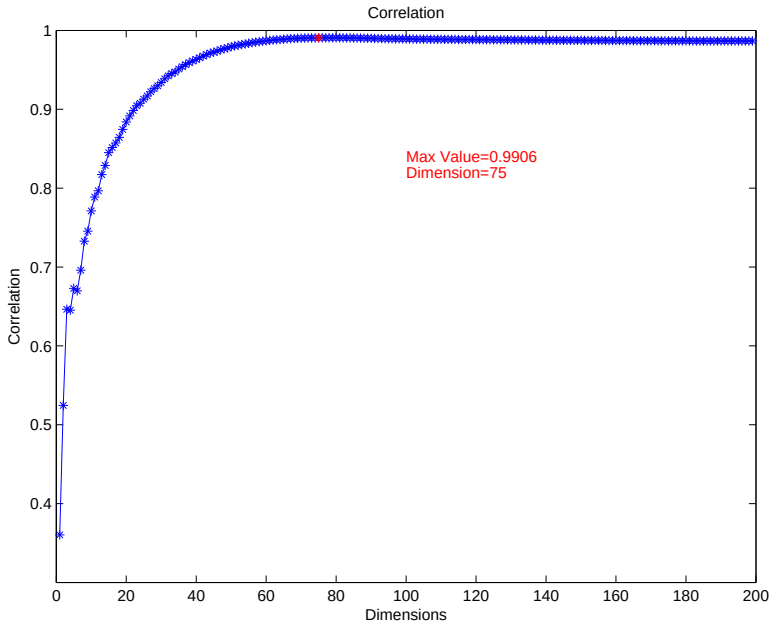


Figure A.30: Correlation plot of Poly-G using dihedral ISD and MDS.

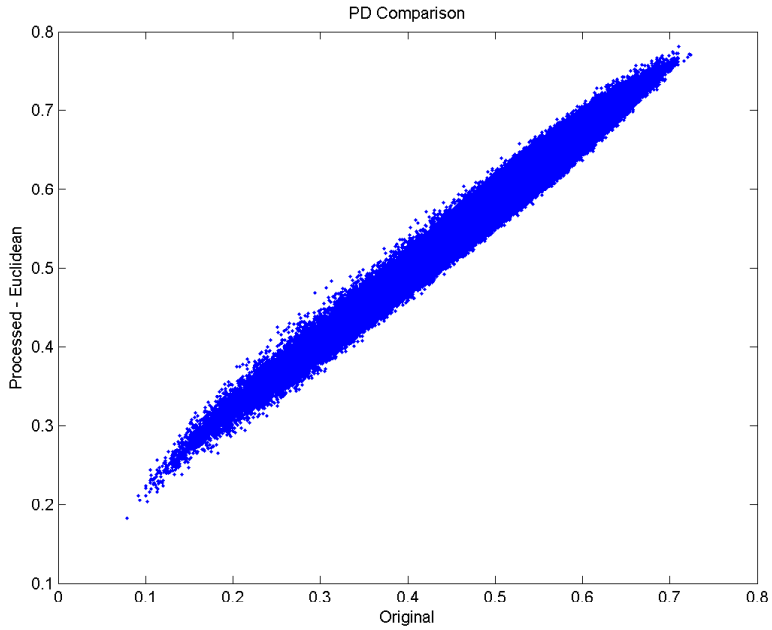


Figure A.31: Pairwise distance plot of Poly-G using dihedral ISD and MDS.

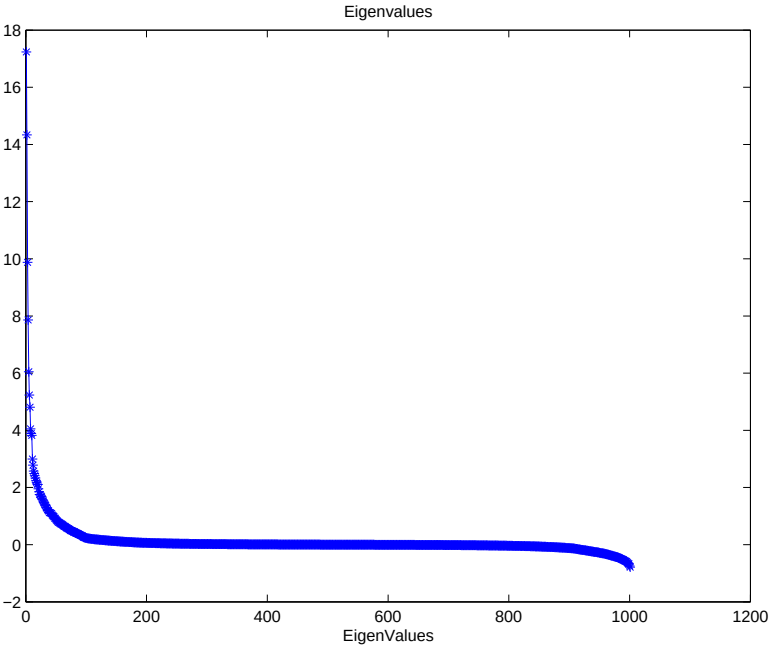


Figure A.32: Eigenvalues of Poly-G using dihedral ISD and MDS.

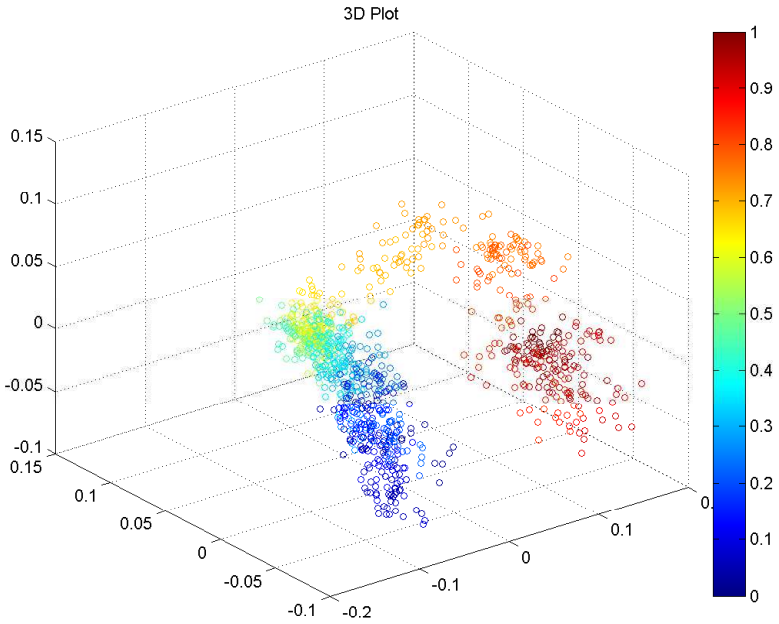


Figure A.33: 3D plot of Poly-Q using dihedral ISD and MDS.

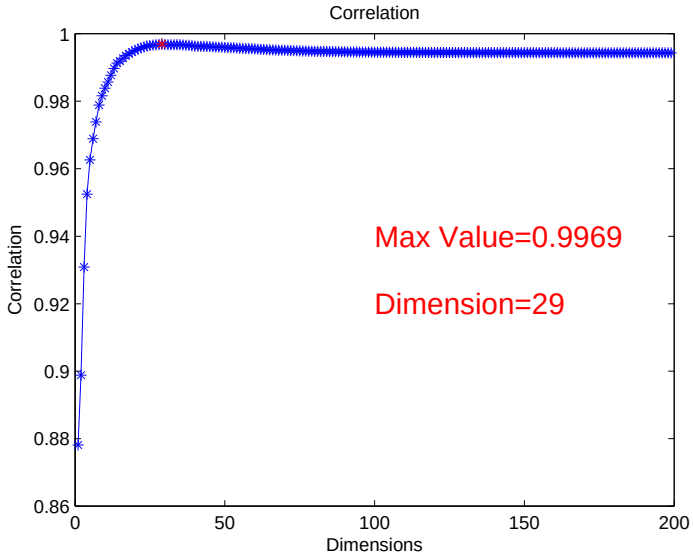


Figure A.34: Correlation plot of Poly-Q using dihedral ISD and MDS.

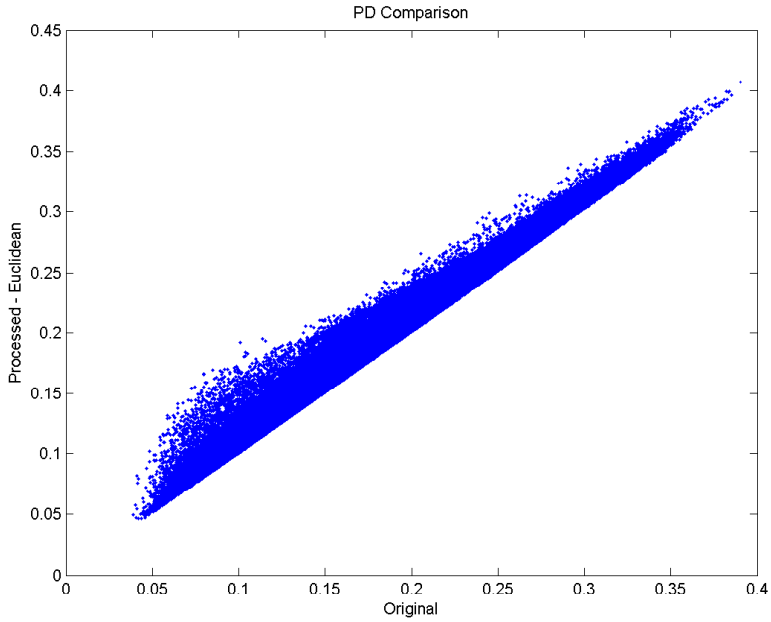


Figure A.35: Pairwise distance plot of Poly-Q using dihedral ISD and MDS.

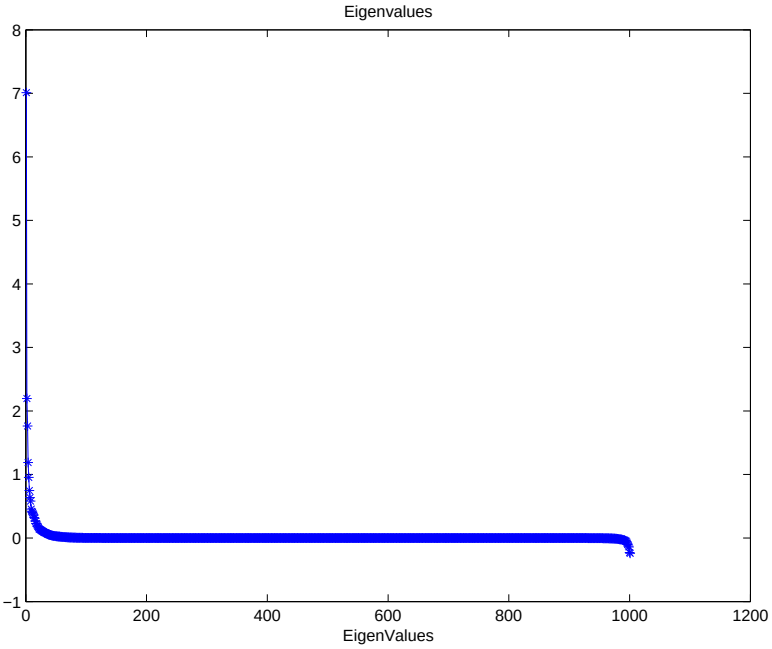


Figure A.36: Eigenvalues of Poly-Q using dihedral ISD and MDS.