DATA MODELLING - PROJECT 1

Internal Dynamics of Modular Protein Structures

Gus Breese, Ben Davies, Mark James, Vladimír Macko, Matthew Ramcharan

Abstract

A data set consisting of modular proteins with user defined shapes has been provided for the purpose of analysis. Given 10 additional perturbations per protein, patterns can be observed about its flexibility relating to itself and its respective neighbours. This analysis and understanding of the dataset allows one to rank modules according to the rigidity of the structure and apply this to predict the behaviour of modules within a given protein. A table of distributions of modules has been created. Using this, 77 out of 90 modules are correctly predicted according to their distributions, proving the method to be valid. With a larger sample size of training and test data the predicted distributions could be more accurate, and the correctness of the method could be reinforced.

Under the supervision of:

Oscar Benjamin

6th November 2017

Contents

1 Introduction

1.1 The Task/Aims

The last decades have seen a growing interest towards the discovery and manufacturing of nanomaterials. These materials are characterized by precise nanometer-scale structures that are responsible for specific optical, electric or mechanical properties. Nanomaterials have a broad spectrum of applications ranging from energy storage to diagnostics. In biology, control at the nanometer scale is commonplace and structures are formed mainly by proteins, a basic type of molecule produced in living organisms. Nowadays, proteins can be computationally designed, paving the way to the production of biological structures with defined characteristics that could expand the range of available nano materials. The task is to predict the flexibility of a protein and its modules based on its topology.

A dataset consisting of modular proteins with user defined shapes has been collected. Each protein consists of a chain of modules, which, in turn, are defined as chains of residues. The positions of these residues are described using 3D coordinates. Dynamics simulations are applied to these proteins, resulting in stochastic 'perturbations', i.e. displacement of the residue positions. The displacement across the different perturbations, defined by the coordinate change, for specific proteins acts as metrics for measuring a modules flexibility. Analysis of the perturbations will be performed using a course grained approach. The rigidity and flexibility of the modules will thus be determined. Additionally, analysis of the neighbouring modules will be performed to investigate their influence on the flexibility of the module in question. The outcomes of the analysis of initial dataset will be used to develop an automated predictive model, capable of ranking modules according to their flexibility. The overall goal of the predictive model is to approximate the expected deformation of a protein without the need for time-consuming simulations.

2 Data Processing

The training data given consists of Protein Data Bank (.PDB) files, a textual file format describing the three-dimensional structures of molecules [1]. For the purposes of determining displacement a PDB file contains the 3D coordinates of the atoms in each module for the given protein. The data is produced from a designed protein using Elfin [2], which is then perturbed by a stochastic process that is not required to be understood for the purpose of this project. Each protein has been provided with ten perturbations as PDB files.

2.1 Root-mean-square derivation of atomic positions (RMSD)

RMSD is the most commonly used quantitative measure of the geometric similarity between two superimposed protein structures. The unit of RMSD is Ångström (Å). RMSD is calculated via

$$\begin{aligned} RMSD(\mathbf{v},\mathbf{w}) &= \sqrt{\frac{1}{n}\sum_{i=1}^{n}||v_i - w_i||^2} \\ &= \sqrt{\frac{1}{n}\sum_{i=1}^{n}(v_ix - w_ix)^2 + (v_iy - w_iy)^2 + (v_iz - w_iz)^2} \end{aligned} \tag{1}$$

where v_i and w_i are two sets of n points in vector space.

RMSD is often calculated for the 'backbone' heavy atoms Carbon, Nitrogen and Oxygen. Every residue contains a central carbon atom, called Cα, that can be used to describe the position of the whole residue. Thus we consider a coarse grain residue representation: a single point per residue (the Cα) as the "centre of mass" [3]. As each residue uses the carbon alpha atom as its central point, the RMSD values from these atoms can be considered to be the displacement for each residue. Hence the RMSD can be used as the metric for measuring the flexibility of a protein and its modules.

2.2 Kabsch algorithm

Using the Kabsch algorithm [4], it is possible to transform the perturbed protein data to be aligned with the original proteins in order to minimise and accurately measure the RMSD. The Kabsch algorithm translates the data or "re-centers" it about a common centroid, then computes the optimal rotational matrix as seen in Fig. 1 and Fig. 2. The translation is calculated using the root-means-square deviation of centroids, and the rotational matrix is calculated by performing a singular value decomposition. The Kabsch algorithm provides a consistent way of calculating the RMSD between the original protein and its different perturbations.

2.3 Kernel density estimation

The kernel density estimation is a non-parametric method to estimate the probability density function of a random variable. Fundamentally, the kernel density estimation is used for data smoothing, in which case inferences about the population are made based on the data sample [5]. Given a univariate, independent and identically distributed sample drawn from an unknown density f,

$$(x_1, x_2, ..., x_n), \tag{2}$$

one can estimate the shape of the function f, using

$$\hat{f}_h(x) = \frac{1}{n}\sum_{i=1}^{n} K_h(x - x_i) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right), \tag{3}$$

where K is the kernel - a non negative function that integrates to one. The bandwidth h is a smoothing parameter. Generally, one wants to define h as small as the data will allow to maximise information gain. Theoretically, there are many ways in which the bandwidth can be optimised if the density function is unknown. One which is commonly used when density function is known to have a Gaussian base is a rule-of-thumb bandwidth estimator, widely known as Silvermans rule of thumb [6]. The formula is given by

$$h = \left(\frac{4\hat{\sigma}^5}{3n}\right)^{\frac{1}{5}} \approx 1.06\hat{\sigma}n^{\frac{-1}{5}}. \tag{4}$$

where $\hat{\sigma}$ is the standard deviation of the samples. While easy to compute, problems can arise if the distribution being estimated is not close to being normal. Once the bandwidth has been selected it is possible to run the function through our data. The kernel density estimation is closely related to histograms, but can include continuity and smoothness [7] by using a suitable kernel as shown in Fig. 4. The KDE is only used for visualisation purposes. This is because large amount of histograms in a single plot are unreadable whereas the KDE representation shows the same information without as much visual noise. Violin plots shown in Figs. **??** and 3 use the kernel density estimation.

3 Single Module Analysis

The analysis of the training data is conducted with the ultimate goal of determining a valid method for automating the prediction of the flexibility of any protein structure consisting of previously trained modules. It is desirable to gain an understanding of whether modular flexibility is consistent between the perturbations of a single protein, as well as between proteins globally. The findings discussed in this section determine whether the flexibility of a module can be assessed based on the entire dataset, which consists of many different proteins and their perturbations, or just within the perturbations of the protein in question.

It is therefore sensible to first determine whether module behaviour is consistent between the perturbations of a single protein. To do this analysis was performed on every protein whereby the modules in each perturbation were ranked according to their RMSD values as shown by the two examples in Fig. 6. This shows there to be a number of modules that are consistently among the most and least flexible in each perturbation. This implies within the same protein the modules behave relatively consistently. Subsequently, this suggests that the topology may be used to predict flexibility, because between perturbations the chain of modules remains unchanged.

In order to investigate whether a module behaves consistently between proteins it is therefore useful to analyse the distribution of the RMSD for a module across the whole dataset. To achieve this the data is plotted as a histogram, an example is shown in Fig. 7. The first thing to notice here is that the overall distribution of the module appears, visually, to follow something that could be approximated to a normal distribution. The assumption of normality is supported by the central limit theorem [8], due to there being many samples of this module across the whole dataset. Due to this, it can be inferred that modular behaviour can be described across the whole dataset under one distribution which in turn suggests that predictions can be made based on the whole dataset.

3.1 Triplets Analysis

As suggested in the previous section the topology of a protein may be affecting the dynamics of its modules. With regards to protein, a triplet refers to three modules in a row. For the purposes of this project the assumption has been made that the order of the modules within these triplets is important. This is a sensible assumption because the modules aren't symmetric with regards to the ordering of the residues that they contain. The module that is being investigated is always central within the triplet. Ordered triplets were investigated because they are the simplest robust factor that most obviously differentiates the central module, given the structure of the protein.

Figure 8 show KDE probability distribution functions for all possible triplets found in the whole training dataset for given central module. The distributions of the triplets vary around the distribution of the central module. This implies that the triplets may be used to predict the flexibility of the central module. This can also be seen in an another example in the Fig. 3 where different triplets follow different distributions.

3.1.1 Shapiro-Wilk test

Given the stochastic nature of the perturbations, predicting the RMSD of a module with absolute accuracy is difficult. The process is be made easier, however, if it is possible to approximate RMSD distribution of each module as being Gaussian. Hence this is a very useful assumption to make, though it is essential to test its validity.

The Shapiro-Wilk test takes a sample of data with a null hypothesis that the sample came from a normally distributed population. The test statistic is used to ascertain the p-value, with a confidence level of 5% being the accepted standard. If the p-value is less than 0.05 the null hypothesis is rejected thus the data is not normal, otherwise the hypothesis cannot be rejected. The test is therefore used to identify triplets that follow Gaussian distribution. P-value can be considered precise only for smaller datasets.

Before conducting the test the impact of outliers on the perceived normality of the data should be considered. Making predictions for the behaviour of a module is far simpler should a normally distributed dataset be available, so it is undesirable to reject the normality of a dataset based on a very small number of extreme outliers. The method proposed to tackle this is median absolute deviation (MAD) [9]. The data is ordered with regards to its deviation from the median, and then the median of this new dataset is taken, this is the median absolute deviation. Outliers are then screened using $\frac{x_i - median(x_i)}{MAD}$, where x_i is each data point. If this outputs a value higher than a chosen threshold then the data is considered to be an outlier. For the purposes of this model a value of 3.5 was chosen, which given that most of the modules have an average RMSD of less than one means that only extreme outliers will be ignored. Ignoring outliers means that it is sensible to now consider the mean rather than the median when ranking the flexibility of the modules and their triples, so this is the parameter that will now be preferred for these rankings.

In the single module analysis it was mentioned that the central limit theorem is used to confirm the normality of the modular distributions. The reason for this is that when applied to individual modules the Shapiro-Wilk test is inaccurate. The reason for this is because the test becomes very sensitive to small deviations from normality for large datasets [10], and many of the modules have hundreds of datapoints, with a couple having multiple thousand. Hence the central limit theorem is relied on to assume the normality of these large datasets distributions. This can be verified simply through visually analysing the distributions, such as in Fig. 7.

Generally the triplets have significantly less data to be modelled on than the individual modules, hence the Shapiro-Wilk test is suitable to verifying the normality of these distributions. Many of these distributions sit quite far from the distributions of their relative central modules, implying that there is a factor that is differentiating these modules from the others. An obvious conclusion would be to suggest that this factor is the neighbouring modules. However, first this data must be proven to be statistically significantly different from the rest, and even then without more data and further tests it is difficult to say with certainty that this difference is caused by the neighbouring modules.

3.1.2 Mann-Whitney U test

In order to determine whether a given triplet distribution should be further analysed it is important to test if it is statistically significantly different, i.e. whether the data is different enough from the central module distribution and whether there is enough of this data to consider it to follow a separate distribution [11].

The Mann-Whitney U test first ranks all the data regardless of the group that they belong to, i.e. the triplet being tested or any of the other triplets for the same module. The rank for each item of data in each group is then totalled together, with the groups being assigned T_1 and T_2 and the number of data points in each group being referred to as N_1 and N_2 respectively. The U_1 and U_2 values are then calculated to be:

$$U_1 = R_1 - \frac{N_1(N_1+1)}{2} \tag{5}$$

$$U_2 = R_2 - \frac{N_2(N_2+1)}{2}. \tag{6}$$

The smaller of these 2 values is then compared against the Mann-Whitney U table to determine whether the data is statistically significantly different at the 5% level.

Applying this method to the test data used in this project results in 190 of the 372 unique triplets being identified as statistically significantly different from the central module. That is clearly a considerable proportion of the triples that differ from the overall trend for their corresponding central module. Thus, given that the sample size is relatively large, the question of false positives must be raised before the result can be fully analysed. The question of false positives relates to the family-wise error rate, which is the probability of coming to at least one false conclusion in a series of hypothesis tests [12], calculated by

$$FWE \leq 1 - (1 - \alpha_{IT})^c, \tag{7}$$

where α_{IT} = alpha level for an individual test (0.05 for our tests), c = Number of comparisons. There are 372 tests, so the familywise error can be approximated to approximately 1. There will be at least one false positive result. Due to this a correction must be applied, in this case the Holm–Šidák method is chosen, an extension of the Holm-Bonferroni method.

3.1.3 Corrections by Holm–Šidák method

The Holm-Bonferroni method controls the familywise error rate - the probability that one or more Type I errors (false positives) will occur. The Šídák modification of the Holm test makes it a more powerful test, especially when there are many comparisons.

Applying this method to the results of the Mann Whitney U test results in 75 of the 372 unique triplets still identified as statistically significantly different from central module. These 75 triplets must be further analysed: have the Shapiro-Wilk test applied to them to ascertain whether its distribution is Gaussian. If a triplet follows a Gaussian distribution then the mean and standard deviation can be used to describe the predicted distribution. Otherwise, the RMSD is predicted using the distribution of the central module.

4 Prediction model

4.1 Method

The original goal of this project was to produce an automated model that could accurately predict the flexibility of any given protein structure. Thus, in order to analyse each module it is sensible to utilise the data that has already been gathered using the techniques discussed earlier in the report. First, each triplet in the structure is checked. If the data for that triplet exists as a normal distribution and has been found to be statistically significantly different from the overall module distribution, then the RMSD range for that module is predicted based on the mean and standard deviation of its triplet distribution. If the distribution of the data for that triplet isn't either normally distributed or statistically significantly different, then the RMSD range is predicted based on the mean and standard deviation of the distribution for the central module.

The model was then used to predict the RMSD for the modules in 6 protein structures that were not present within the original training data, i.e. the test data. These 6 proteins consist only of modules that were present in the proteins found in the training data. These predictions were then compared against the average RMSD taken from a set of simulated perturbations for each protein. A prediction is considered to be correct if the average RMSD sits within one ± standard deviation of the predicted RMSD.

4.2 Results & Discussion

The test data consists of the 3 proteins made up of 20 modules each, and 3 made up of 10 modules each, thus there are 90 modules in total. Of the 90 modules that were tested 86% were correctly predicted (± standard deviation of the predicted RMSD), which given the limitations of the dataset is a good outcome. The 13 modules for which the RMSD was incorrectly predicted are displayed in Fig. 9. Of the 13 that were incorrectly predicted 9 came from predictions made based on the overall distribution of the central module, and 4 came from predictions made based on the distribution of the triplet itself. Given that out of all 90 modules only 12 were predicted based on the distribution of their triplet this seems to imply that the predictions based on the single module distribution are more successful. However, a closer look at the average standard deviation of the triplet distributions versus the overall module distributions suggests an alternative interpretation.

The average standard deviation of the predictions made from the module distributions is 0.214, while the corresponding average for the predictions made from triplets is 0.078. Therefore, predictions made based on the distribution of a triplet need to be nearly 3 times more accurate compared to those made from the modules' distribution. Looking at the 4 triplet points that were incorrectly predicted, the average standard deviation drops to 0.055. This implies that the reason that the data is being incorrectly predicted is because the distributions of these particular triplets have particularly low standard deviation. This is backed up by inspection of Fig. 9 where 3 of the 4 triplets see the actual average sit just outside of the predicted range. Meanwhile, the average standard deviation of the modules also reduces slightly, to 0.152, though Fig. 9 illustrates that even with a significantly wider standard deviation many of the predictions would still not encompass the actual value.

Referring again to Fig. 9, the points predicted from triplets generally sit far closer to the actual value than the predictions taken from the module distributions. This is backed up numerically, as the average error of the points taken from the triplet is 0.0930 compared to 0.3830 for the module predictions. This shows that although, proportionally, the rate at which the triplet distributions inaccurately predict the RMSD is higher than that of the module distributions, the triplets still make predictions that sit far closer to the absolute value, based on the difference of the means. This could imply two things. Firstly, it is likely that the data is overfitted to a relatively small dataset given that on average there are no more than 40 examples of each unique triple throughout the whole dataset. More data would tune the model to predict a more accurate mean and standard deviation. The second conclusion that could be implied is that the standard deviation is simply inadequate for estimating the range of the RMSDs. The data presented in Figs. 9 and 10 demonstrates the accuracy with which the triplet distributions must predict the mean.

Across all the predictions made by the model, the average error in the RMSD taken from the triplets was only 0.0492, compared to 0.105 for the module distributions. It is unfortunate that there is insufficient data to predict more of the triplets based on their unique triplet distribution, however this data still gives a strong indication that the predictions made from the triplet distributions are superior to those made from the module distribution. It is interesting to note that even the average error of the 4 triplet distributions that were falsely predicted are better than the average error of the module distribution predictions as a whole. Thus the conclusion to be drawn here must be that identifying the unique distribution of each triplet is essential to maximising the precision of the predictions. To achieve this it is therefore hugely important to have a training dataset that is extensive for every triplet, so that no triplets are considered to be statistically insignificant due to a lack of data.

5 Conclusion

A primary goal of the original brief was for the final model to be able to rank modules according to flexibility. On a modular level this was demonstrated to be a relatively simple task as seen in Fig. 5. However, the characteristic that neighbours are important to the RMSD, and therefore flexibility, of a module offers potential for an even more comprehensive ranking. A preliminary ranking of the 35 triplets that were proven to have unique distributions is provided in Table 1. Having such data available can be hugely useful. A module that is generally very stiff can display flexible characteristics in the presence of certain neighbours, this ranking highlights that behaviour.

In terms of the models ability to predict the flexibility of a module, it is very adaptable to the data that it has available. When the data for a module is limited the model makes allowances and is more conservative in its predictions, however for cases where more data is available the model makes remarkably precise predictions to an impressive degree of accuracy. The data generated during each stage of the analysis of this project has strongly suggested that there is a link between the neighbours that a module has and its resulting flexibility. This relationship was utilised to the benefit of the predictive model, and has the potential to be further expanded upon given a larger database.

5.1 Project Limitations and Further Improvements

The main limitation to the predictive model is the limited database that it has from which to draw its conclusions, however this has a relatively simple and obvious solution in that it simply requires more data. With a larger dataset more unique triplet distributions should be discovered, and the distributions of those triplets that have already been revealed should be fine-tuned with more accurate means and standard deviations. In itself the use of standard deviation to define the probable range for the prediction is flawed for relatively small datasets. The model has shown itself to work accurately for modules where data is sufficient in volume, and this gives confidence that this methodology could be extended to larger data sets and still be precise.

An importance of ordered triplets was assumed, because the modules themselves are not symmetric. This assumption could be tested, i.e. by calculating statistical significance between a triplet and its reverse – e.g. triplet ABC and CBA. If the null hypothesis is not rejected, the triplet and its reverse can be considered to be drawn from the same distribution. Thus the order does not matter. Inversely statistical significance would imply that order is important. This test can be only performed if the dataset contains enough datapoints for the triplet and its reverse. This was not the case for the initial training data supplied.

The method lacks a way of detecting protein chain breaking. In order to investigate this further more research would be required into outliers. Once again this comes down to a limited dataset. The training dataset did not contain enough outliers that would warrant research into this further model. Thus for the purpose of improving the prediction of the modules that are normally distributed the outliers have been disregarded.

References

[1] K. H. Helen Berman and H. Nakamura, 'Announcing the worldwide protein data bank', *Nature Structural Biology*, vol. 10, no. 980, 2003. DOI: doi:10.1038/nsb1203-980.

[2] C. T. Yeh, T. J. Brunette, D. Baker, S. McIntosh-Smith and F. Parmeggiani, 'Elfin: An algorithm for the computational design of custom three-dimensional structures from modular repeat protein building blocks', *J. Struct. Biol.*, Sep. 2017.

[3] I. Kufareva and R. Abagyan, 'Methods of protein structure comparison', in *Homology Modeling: Methods and Protocols*, A. J. W. Orry and R. Abagyan, Eds., 1st ed., ser. Methods in Molecular Biology. Totowa, NJ, USA: Humana Press, 2012, vol. 857, pp. 231–257, ISBN: 978-1-61779-588-6. DOI: 10.1007/978-1-61779-588-6_10.

[4] P. J. A. Cock, T. Antao, J. T. Chang, B. A. Chapman, C. J. Cox, A. Dalke, I. Friedberg, T. Hamelryck, F. Kauff, B. Wilczynski and M. J. L. de Hoon, 'Biopython: Freely available python tools for computational molecular biology and bioinformatics', *Bioinformatics*, vol. 25, no. 11, pp. 1422–1423, 2009. DOI: 10.1093/bioinformatics/btp163.

[5] E. Parzen. (2007). On estimation of a probability density function and mode, [Online]. Available: https://projecteuclid.org/euclid.aoms/1177704472.

[6] S. J. Sheather *et al.*, 'Density estimation', *Statistical Science*, vol. 19, no. 4, pp. 588–597, 2004.

[7] D. W. Scott, 'On optimal and data-based histograms', *Biometrika*, vol. 66, no. 3, pp. 605–610, 1979.

[8] A. DasGupta, 'Normal approximations and the central limit theorem', in *Fundamentals of Probability: A First Course*. New York, NY: Springer New York, 2010, pp. 213–242, ISBN: 978-1-4419-5780-1. DOI: 10.1007/978-1-4419-5780-1_10. [Online]. Available: https://doi.org/10.1007/978-1-4419-5780-1_10.

[9] C. C. Peter J. Rousseeuw, 'Alternatives to the mean absolute deviation', *Journal of the American Statistical Association*, 1993.

[10] A. Field, 'Discovering statistics using spss', vol. 3, p. 144, 2009.

[11] H. B. Mann and D. R. Whitney, 'On a test of whether one of two random variables is stochastically larger than the other', *Ann. Math. Statist.*, vol. 18, no. 1, pp. 50–60, Mar. 1947. DOI: 10.1214/aoms/1177730491. [Online]. Available: https://doi.org/10.1214/aoms/1177730491.

[12] H. Abdi. (2007). The bonferonni and šidák corrections for multiple comparisons, [Online]. Available: http://www.utdallas.edu/~herve/Abdi-Bonferroni2007-pretty.pdf.

[13] Charnley. (2017). Calculate root-mean-square deviation (rmsd) of two molecules using rotation, github. version 84de1b2, [Online]. Available: https://github.com/charnley/rmsd.

A Appendix

The scripts used to process the data and produce the graphics for this report where produced in Python 3.6.2 and are available on request of Vladimír Macko.

.

Triplet name	Mean (Å)	Standard deviation (Å)
(D79, D79, EMPTY)	0.130032	0.048064
(D14_j1_D18, D18, EMPTY)	0.130074	0.012260
(D18, D18, EMPTY)	0.133084	0.019406
(EMPTY, D79, D79)	0.145275	0.023763
(EMPTY, D79_j1_D54, D54)	0.210915	0.044481
(D14_j1_D54, D54, EMPTY)	0.220513	0.094041
(D14_j2_D71, D71, D71)	0.266763	0.042943
(D79, D79_j2_D14, EMPTY)	0.284457	0.042806
(D79_j2_D14, D14_j1_D54, EMPTY)	0.293078	0.080115
(D14_j2_D79, D79, EMPTY)	0.296512	0.059738
(EMPTY, D18, D18_j1_D14)	0.324841	0.046065
(D14_j1_D79, D79_j1_D54, D54)	0.343692	0.028808
(EMPTY, D14, D14_j1_D14)	0.346507	0.047399
(D14_j4_D79, D79_j1_D54, D54)	0.347814	0.030218
(EMPTY, D14_j2_D54, D54)	0.349417	0.090989
(EMPTY, D18_j1_D14, D14)	0.357951	0.089053
(D79, D79_j2_D14, D14_j1_D79)	0.361407	0.062070
(D54_j1_D79, D79_j2_D14, D14_j2_D71)	0.367345	0.065558
(D14_j3_D54, D54_j1_D79, EMPTY)	0.367392	0.086949
(D54, D54_j1_D79, EMPTY)	0.368658	0.090293
(D14_j5_D79, D79_j1_D54, D54)	0.369269	0.039885
(D54_j1_D79, D79_j2_D14, D14_j2_D14)	0.370325	0.052355
(D14_j1_D54, D54_j1_D79, EMPTY)	0.388292	0.057229
(EMPTY, D14_j5_D79, D79)	0.394315	0.079548
(D79_j1_D54, D54_j1_D79, EMPTY)	0.394843	0.078949
(D14_j2_D54, D54_j1_D79, EMPTY)	0.396385	0.058600
(D54_j1_D79, D79_j2_D14, D14_j1_D79)	0.402732	0.068056
(D18_j1_D14, D14_j5_D79, D79)	0.433201	0.046478
(D14_j5_D79, D79_j1_D54, D54_j1_D79)	0.448912	0.054305
(D79, D79, D79_j1_D54)	0.460837	0.077214
(D79, D79_j1_D54, EMPTY)	0.473253	0.079871
(D79_j1_D54, D54_j1_D79, D79_j2_D14)	0.482501	0.084975
(D14_j2_D14, D14_j2_D79, EMPTY)	0.494349	0.012626
(D79_j1_D54, D54_j1_D79, D79_j1_D54)	0.534839	0.080066
(D18_j1_D14, D14_j1_D54, D54)	0.540107	0.094872
(D14_j3_D54, D54_j1_D79, D79_j1_D54)	0.549673	0.070905
(D14_j1_D79, D79, D79)	0.596138	0.029681
(D14_j4_D79, D79, D79_j1_D54)	0.611621	0.085686
(D49_j1_D79, D79, D79)	0.615440	0.038187
(D18_j1_D14, D14_j1_D54, EMPTY)	0.620441	0.075077
(D18, D18_j1_D14, D14_j4_D79)	0.621086	0.107049
(D49_j1_D79, D79, D79_j1_D54)	0.625034	0.078440
(D14_j1_D18, D18_j1_D14, D14_j1_D18)	0.639212	0.080499
(EMPTY, D14_j4_D79, D79_j2_D14)	0.654302	0.191001
(D14, D14_j3_D54, D54_j1_D79)	0.670545	0.194271
(EMPTY, D14_j2_D54, D54_j1_D79)	0.680931	0.165541
(D79_j2_D14, D14_j5_D79, D79_j1_D54)	0.743073	0.100918
(EMPTY, D79, D79_j2_D14)	0.891264	0.057291
(EMPTY, D79, D79_j1_D54)	0.947472	0.056881
(D79_j2_D14, D14_j4_D79, D79)	1.068192	0.126333

Table 1: Statistically significant triplets that have been tested to be normal. Ordered by the predicted mean RMSD.

List of Figures

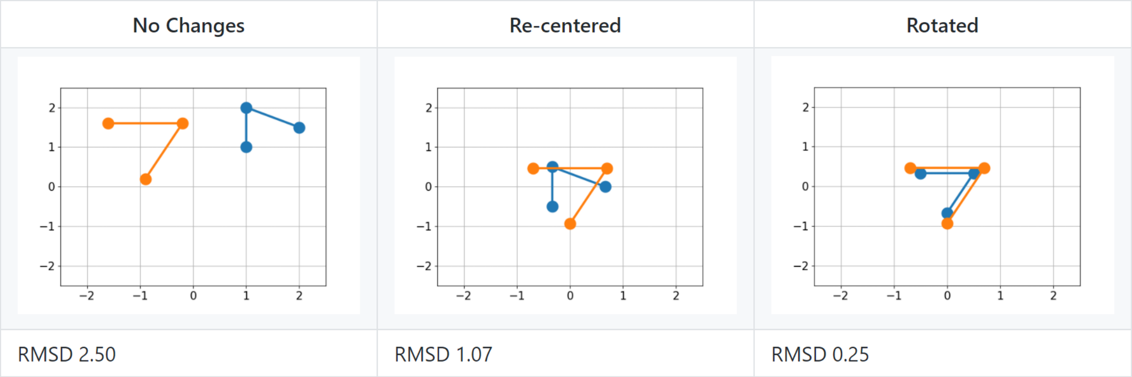
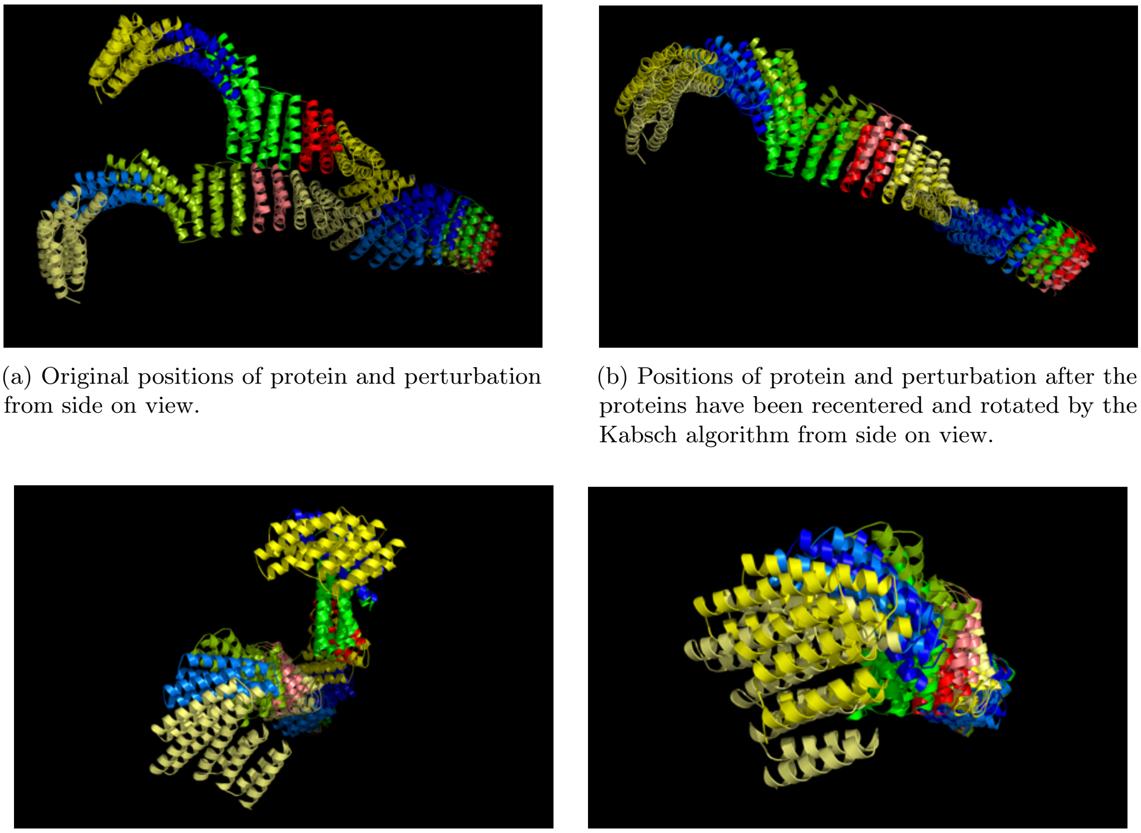


Figure 1: A simplified demonstration of what the Kabsch algorithm does on a three module protein (orange) and a perturbation (blue) [13]. The Kabsch algorithm re-centers the two proteins about a common centroid, then rotates the proteins to minimise RMSD.



(a) Original positions of protein and perturbation from side on view.

(b) Positions of protein and perturbation after the proteins have been recentered and rotated by the Kabsch algorithm from side on view.

(c) Original positions of protein and perturbation from straight on view.

(d) Positions of protein and perturbation after the proteins have been recentered and rotated by the Kabsch algorithm from straight on view.

Figure 2: Representations of the Kabsch algorithm performed on a protein and one of its perturbations in the test data using PyMOL.

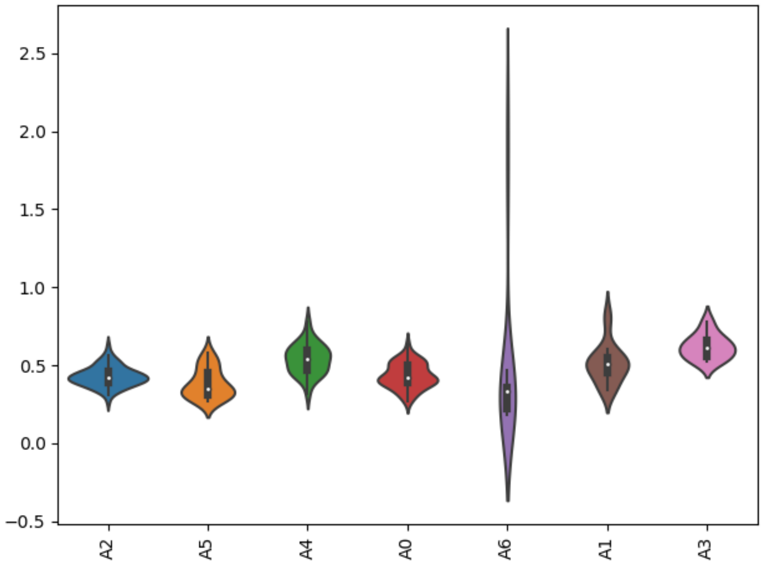


Figure 3: Violin plot showing the module D14_j1_D54 represented by its triples' distributions. This figure separates the module into its triples and plots their distributions. It shows D14_j1_D54 has seven distinct triples all with varying distributions that come together to create the overall distribution for D14_j1_D54 that is shown in Fig. **??**

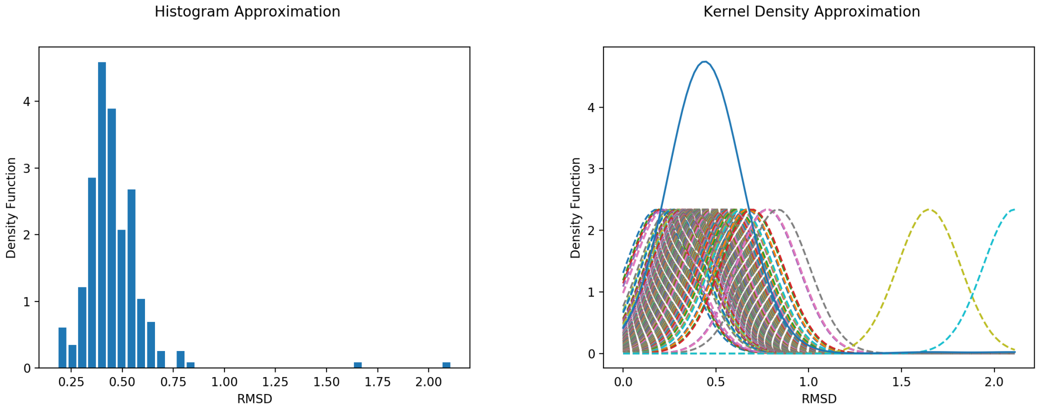


Figure 4: Comparison between two types of approximation of the distribution of the RMSD values for the module D14_j1_D54. The left hand side represents the data points as a histogram whereas the right hand side creates a normal distribution for each data point. These smaller kernels are then used to create the larger overall Kernel Density Approximation i.e. the blue line.

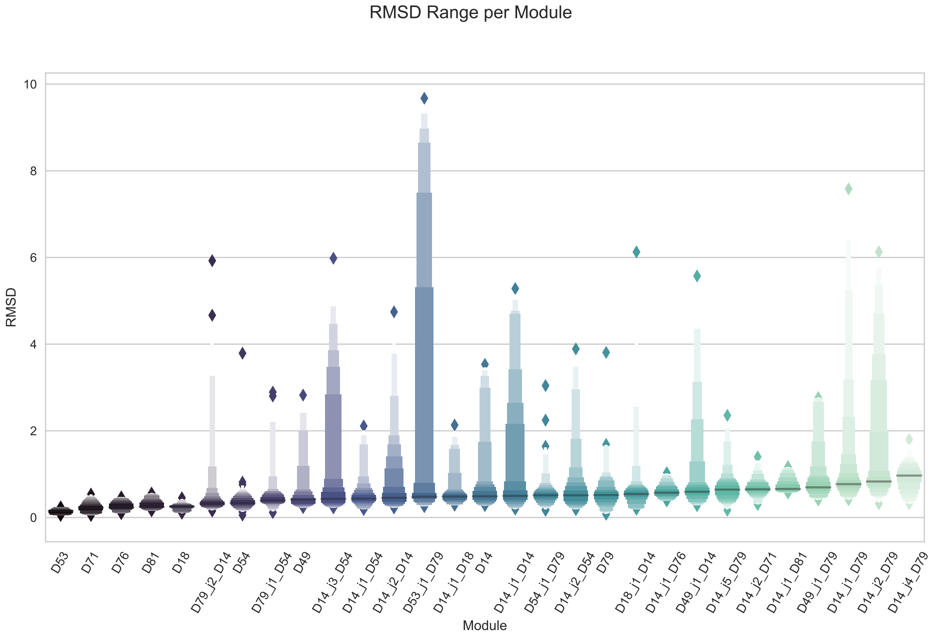


Figure 5: An LV plot, showing the distribution of RMSD values for every module seen in the data. Due to the presence of large outliers (diamonds) which can have disproportionate effects on the mean, the modules are sorted by their median value. The range of which these values lie is also visible. Each decrease in the width of the plot represents a point on the graph. Modules such D53_j1_D79 have many perturbations where the RMSD exceeds the 5 threshold, and could be classed as a higher risk from this visualisation. D53, the furthest to the left and therefore the module with the smallest median, has a small range and therefore could be deemed a low risk module.

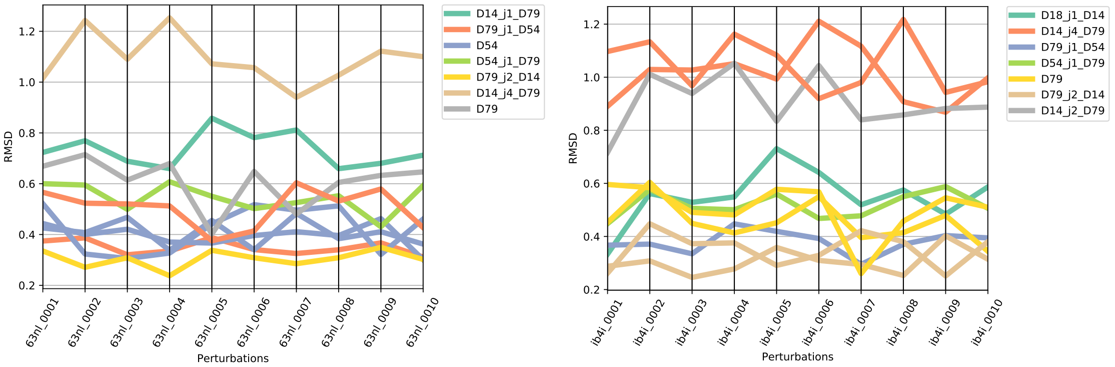


Figure 6: Line Plots showing module behaviour across different perturbations for the same protein. Each line represents how a module's RMSD values change between perturbations. Notice that the two figures suggest that module behaviour is consistent in a specific protein string as the module ranking remains consistent across the perturbations.

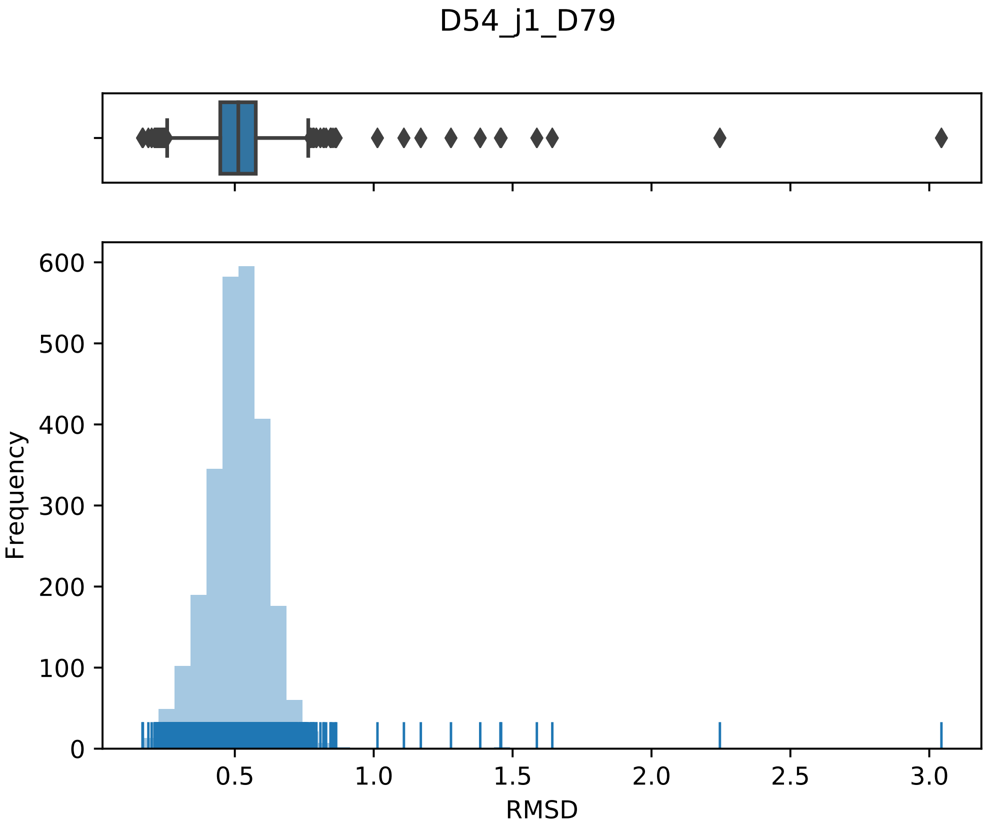


Figure 7: The distribution of the RMSDs of modules in a single protein, D54_j1_J79, can be shown as a histogram. 150 values are binned to demonstrate the normal distribution by the central limit theorem. In addition, the individual RMSD values are marked, showing the locations of several outliers to the distribution

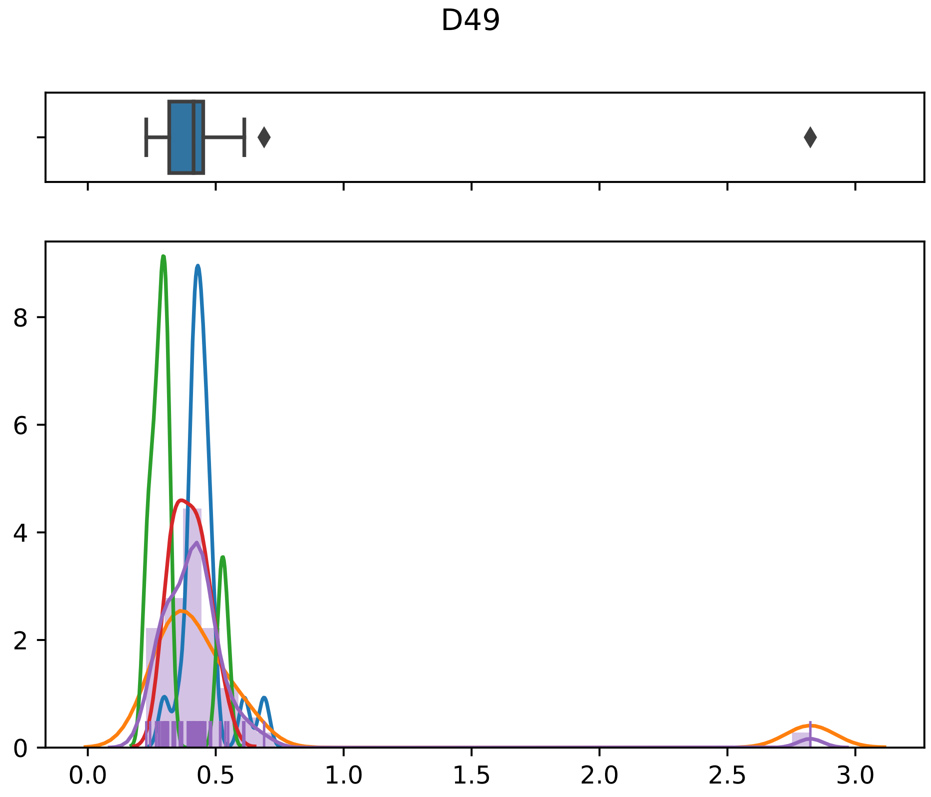


Figure 8: Estimated probability density functions (using KDE) of every triplet with a given central module D49. The distribution of the central module is shown as a histogram in the same plot.

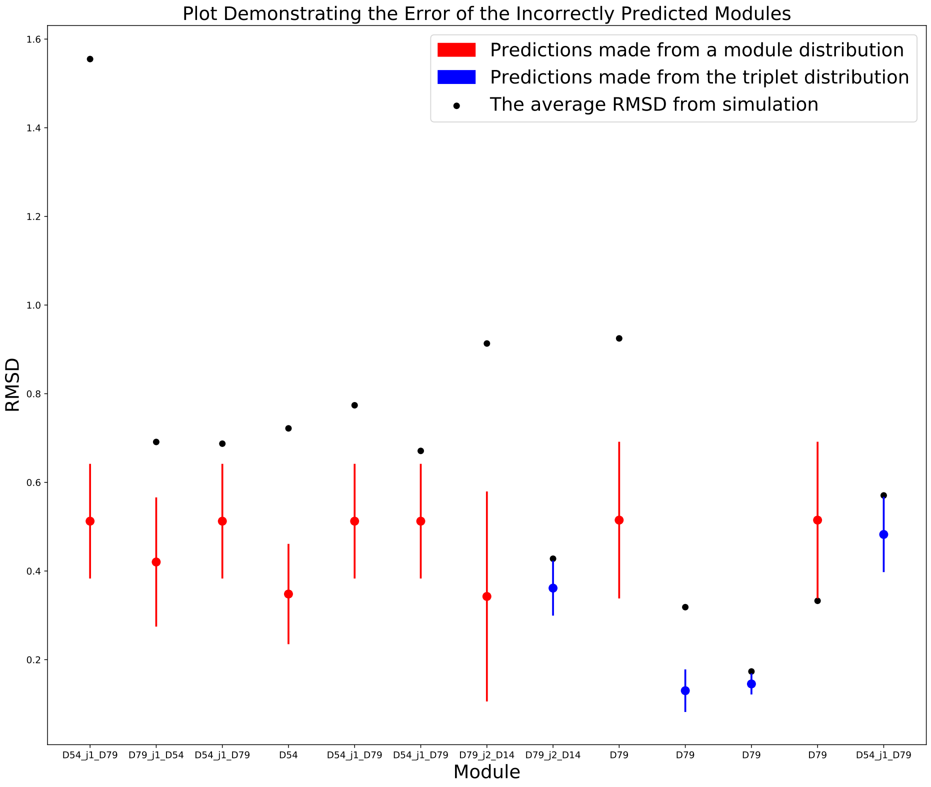


Figure 9: An illustration of the absolute error in the incorrectly predicted modules. The absolute error in the predicted triplets is not large, however due to a smaller sample size in the triplets, the standard deviations are small and therefore these estimates are considered out of the target range.

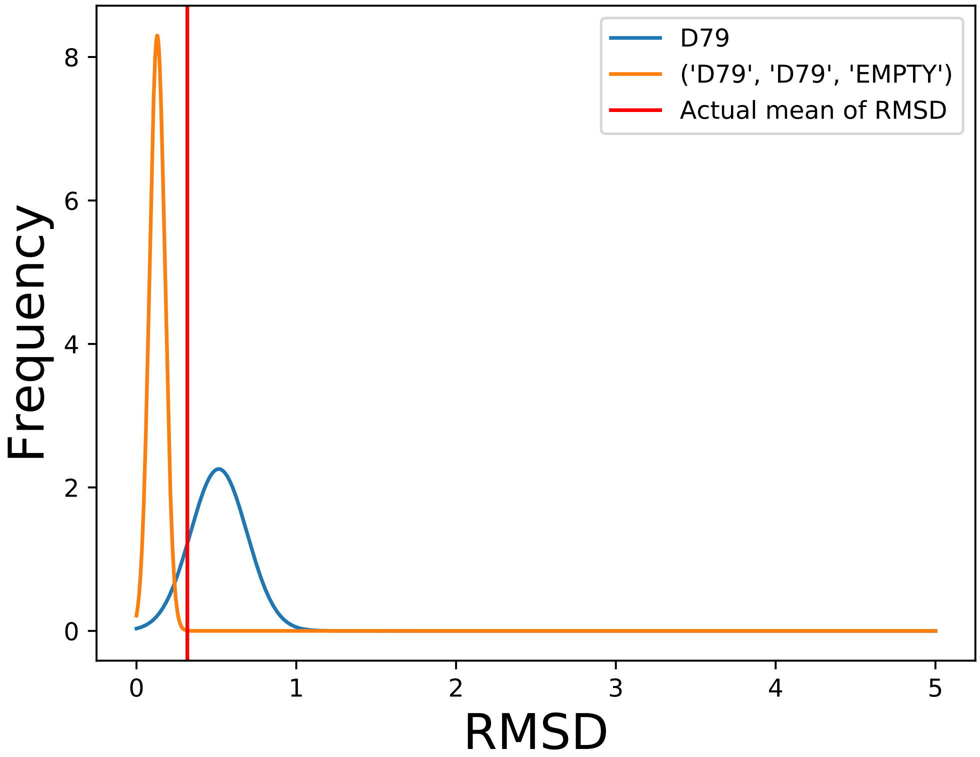


Figure 10: The mean of the triplet distribution is closer to the actual value compared to the central module distribution, but it is outside one standard deviation for the triplet. This suggest that using a standard deviation for identifying successful prediction is problematic.