**Implementing Linear Regression for Pairwise Protein Structural Similarity Prediction**

**Rahul Alapati**1,\*,+**, Sritika Chakladar**1,+**, Anuj Gupta**1,+**, Vineet Nayak**1,+ **and Austin Ream**2,+

1Graduate Student, Computer Science and Software Engineering, Auburn University, 36830, USA 2Undergraduate Student, Computer Science and Software Engineering, Auburn University, 36830, USA

\* rza0037@auburn.edu

+these authors contributed equally to this work

**ABSTRACT**

A protein’s three dimensional structure is very important to its function. We developed a program that predicts protein structural similarity. The program implements the linear regression algorithm for prediction. The features used for prediction include information about amino acid relative position, secondary structure proportions, and amino acid solvent accessibility proportions.

**1. Introduction**

Single chain protein structure is comprised of primary structure, secondary structure and tertiary structure. Primary structure refers to the sequence of amino acids in the chain. Secondary structure describes the local foldings that occur along the backbone of the amino acid chain. Tertiary structure is the overall three dimensional structure of the protein. Their is also another structural component of proteins known as quaternary structure which applies to proteins with more than one amino acid chain; however, we are only considering single chain proteins for this project.

Protein structure is responsible for its biological function. Understand a protien’s structure allows us to infer how it works, and how we might be able to modify it. Creating accurate models of proteins can be a very costly process. Being able to predict the similarity between two proteins may be able to give insight into how they work.

**2. Methods**

Our raw data includes two types of files, .fasta and .pbd. Each fasta file contains a protein’s amino acid sequence in FASTA format. Each fasta file has an associated .pdb file that describes the protein’s 3d structures in PDB format. The pdb files are used with a program called TM-align to generate similarity values between proteins. These values are used as training labels for the training of our model. As part of our feature generation, we generate a position specific scoring matrix (.pssm) for each .fasta file.

Our model utilizes linear regression that takes in 50 features and produces a predicted similarity score. The first 40 features are generated from each proteins associated .pssm file (20 from each). The features are created by doing a column-wise average on the matrix. Each of these values is brought down to value between zero and one by dividing each of the values by 100. The next six features are generated by another predictor which predicts whether each amino acid residue is part of a helix, strand or coil. Thus we take the predicted values to summarize the proportion of each of the these classes in each protein. The proportion of each class is then used as a feature (three per protein). The last four features relay the proportions of amino acids in each protein that are buried or exposed to the surrounding solvent. These values are also generated from another machine learning predictor.

**2.1 Linear Regression**

Prediction is done with linear regression by multiplying each feature by an associated weight and adding up the results to get the prediction. Optionally one can add an additional bias weight that does not get multiplied by anything. Training the weights is done using gradient decent. The gradient is calculated by taking the partial derivative of a cost function with respect to each weight. The cost is the squared difference between the predicted value and the actual value. For batch processing, the cost is calculated for several training examples at one, where as stochastic processing calculates the cost for one training example at a time.

After we have calculated the gradient of the cost with respect to each weight, we can use this to nudge each weight down the gradient by a ratio of the gradient. This means that the stronger the gradient is, the more we adjust the weight. We further control the amount of adjustment with per-determined learning rate. We continue adjusting the weights until either the cost function has reached a minimum or the cost has reached an acceptable minimum.

**3. Results**

...

**4. Discussion**

...

**5. References**

1. Wikipedia contributors. (2018, March 21). Protein structure. In Wikipedia, The Free Encyclopedia. Retrieved 19:23, March 26, 2018, from <https://en.wikipedia.org/w/index.php?title=Protein_structure&oldid=831674619>
2. Pauling, L., Corey, R. B., & Branson, H. R. (1951). The structure of proteins: two hydrogen-bonded helical configurations of the polypeptide chain. Proceedings of the National Academy of Sciences, 37(4), 205-211.
3. Alberts, B., Johnson, A., Lewis, J., Raff, M., Roberts, K., & Walter, P. (2002). The shape and structure of proteins.
4. Anfinsen, C. B. (1973). Principles that govern the folding of protein chains. Science, 181(4096), 223-230.