Predicting Possible Oligomerization States of Protein Sequences

Joel Joseph K B[1] & Agney K Rajeev[2]
[1]Department of Biological Sciences and [2]Department of Physical Sciences
National Institute of Science Education and Research Bhubaneswar
P.O. Jatni, Khurda 752050, Odisha, India
joeljoseph.kb@niser.ac.in & agneyk.rajeev@niser.ac.in

Abstract

Protein oligomerization state is the number of polypeptide chains that constitute a protein's quaternary structure. Predicting the protein oligomerization state from its primary amino acid sequence can give a lot of insights into its function and structure; it can also make designing further experiments easier, especially in the case of early experiments in novel proteins. Here we have demonstrated three kNNs that use three different distance parameters. All these parameters are exclusively derived from a protein's primary amino acid sequence. The kNN models have shown accuracies ranging from 58% to 70% for a 10-class classification; accuracies are found using 10-fold cross-validation.

1 Introduction

Protein structure can be divided into four levels: Primary, Secondary, Tertiary, and Quaternary. Among these, Quaternary structure is the final functional form of a protein, formed by one or more sub-units of polypeptide chains. The quaternary structure can be classified into Monomer, Dimer, Trimer, Tetramer, etc., based on the number of polypeptide sub-units present as well as Homo-Oligomer or Hetero-Oligomer based on the type of polypeptide chains involved. This classification is called the Oligomerization state of a protein.

Oligomerization state holds vast significance in the field of proteomics. The function of a protein is heavily influenced by its quaternary structure, and hence, knowledge of the oligomerization state is important in the functional analysis of proteins. Several diseases and the development of drug targets mandate information on the oligomerization state to target specific protein complexes and understand the effects of mutations. Contemporary methods of identifying the oligomerization state require various experimental procedures such as X-ray crystallography, Analytical UltraCentrifugation (AUC), etc., which are costly and time-consuming.

We propose a machine learning algorithm to predict the most probable oligomerization state of a polypeptide chain from its Amino Acid sequence.

2 Related works

We base our work primarily on ideas presented by Hong-Bin Shen and Kuo-Chen Chou on QuatIdent: A Web Server for Identifying Protein Quaternary Structural Attributes by Fusing Functional Domain and Sequential Evolution Information[1], and by Simeon *et al* on osFP: a web server for predicting the oligomeric states of fluorescent proteins[2].

QuatIdent involves the use of Functional Domain information (FunD) and Pseudo position-specific score matrix (PsePSSM) generated from the sequence to predict the oligomerization state using

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

an OET-kNN algorithm. They constructed a two-level model, with the first level predicting the number of polypeptide chains, i.e., Monomer, Dimer, etc., and the second level predicting the type of polypeptide chains, i.e., Homo or Hetero. The first level was reported to have an accuracy of 71.1%, and the same for the second level was 84% - 96%.

osFP is designed to classify fluorescent proteins as Monomers and Oligomers(proteins with more than one sub-unit). They used Amino acid descriptors as sequence features to train a decision-tree algorithm to an excess of 80% accuracy.

3 Datasets and Baseline models

We have curated a dataset of 148,820 unique amino acid sequences in FASTA format (representation of amino acid sequence where each amino acid is represented by an English alphabet) and their corresponding possible oligomerization states from the RCSB Protein Data Bank containing around 200,000 proteins. We have removed sequences with any ambiguous amino acids (E.g., the letter B can represent either Aspartic Acid or Asparagine).

For each FASTA sequence, we calculate the following descriptors using the BioPython ProtParam (refer to the Documentation) module:

molecular_weight, isoelectric_point, aromaticity, gravy, instability_index, helix_fraction, turn_fraction, sheet_fraction, net_charge, molar_extinction_coefficient (Cysreduced), molar_extinction_coefficient (CysCys bond), aromatic_amino_acids, polar_amino_acids, basic_amino_acids, acidic_amino_acids, unique_amino_acids, charge_at_pH_1, charge_at_pH_14

We also generate a Pseudo Amino Acid Composition (PseAAC), which is a vector representation of an Amino Acid sequence containing information regarding the composition as well as the position of its constituent Amino Acids. It was introduced by Kuo-Chen Chou in 2001.

We create three kNN models to classify each amino acid sequence to one of ten oligomerization states (Monomer, Homo 2-mer, Homo 3-mer, Homo 4-mer, Hetero 2-mer, Hetero 3-mer, Hetero 4-mer, and Oligomer). Each kNN uses a different distance parameter to find the nearest neighbor for classification.

Our first kNN uses the classical idea that proteins with similar amino acid sequences would have a similar structure and thus similar quaternary structure. Thus this algorithm uses sequence similarity derived from the PairwiseAligner(refer to the Documentation) module of Biopython. This sequence similarity is then used as a distance parameter, i.e. more similar a sequence is with each other, the closer they are.

Our second kNN uses the vector constructed from the descriptors mentioned above. The distance parameter is then calculated as the Euclidean distance between these vectors.

Our third kNN uses the Pseudo Amino Acid Composition of a protein's sequence. Here, the distance parameter is also calculated as the Euclidean distance between the PseAAC vectors. We chose the weight parameter(w) of PseAAC (Appendix I) to be 0.05 and the amino acid functions to be the hydrophilicity value, hydrophobicity value, and the side chain mass.

4 Experiments

4.1 Finding the best value for k

Finding the best value of k where the accuracy is maximum is crucial. to do this, we created a validation dataset using the train_test_split() function from scikitlearn. The validation dataset used was of size 1% from the training dataset. It was made sure that after the construction of the validation dataset, the data in the validation dataset was not present in the training dataset. This validation dataset was then used to plot an accuracy vs k-value graph to find the k where accuracy was maximum. These graphs are shown in Figure 1.

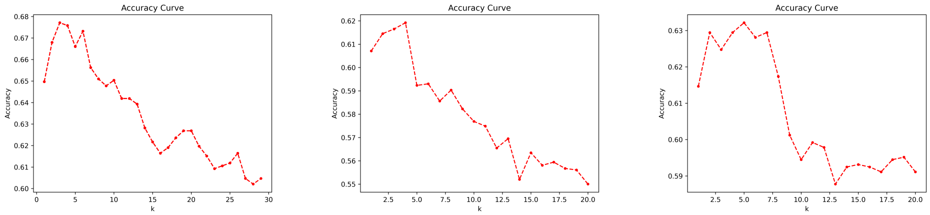

Figure 1: accuracy vs k curve for sequence similarity, descriptors and PseAAC respectively

4.2 Finding the best value for vector dimension(l) in PseAAC

l is another hyperparameter that influences the dimension of the PseAAC vector. We use a similar 1% validation set to determine the value of l.

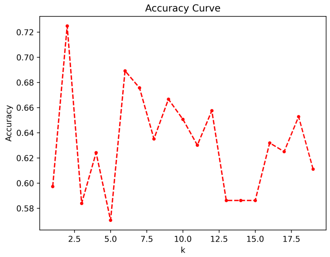

Figure 2: accuracy vs l curve

4.3 Determining the accuracy of the models

The accuracy of the various implementations of kNN was derived using 10-fold cross-validation. the accuracies were determined as follows:

Model	Accuracy (%)	Value of k	Distance parameter
kNN1	70.68	3	Sequence Similarity
kNN2	58.88	4	Descriptors
kNN3	62.06	5	Pseudo AAC

5 Further plans

From the experiments performed using kNN, we have identified different possible feature spaces other than just sequence similarity that can be used to predict protein oligomeric states. Our next goal is to develop a carefully curated independent test database from UniProt with equal representation from all the classes of oligomeric states to evaluate our models more reliably. We intend to build a Deep Learning model that can take advantage of all the mentioned feature spaces found from previous experiments to predict possible oligomeric states with much better accuracy. We also need to research proper embeddings for the Neural Network.

References

[1] Shen, H., & Chou, K. (2009). QuatIdent: a web server for identifying protein quaternary structural attribute by fusing functional domain and sequential evolution information. Journal of Proteome Research, 8(3), 1577–1584. https://doi.org/10.1021/pr800957q

[2] Simeon, S., Shoombuatong, W., Anuwongcharoen, N., Preeyanon, L., Prachayasittikul, V., Wikberg, J. E. S., & Nantasenamat, C. (2016). osFP: a web server for predicting the oligomeric states of fluorescent proteins. Journal of Cheminformatics, 8(1). https://doi.org/10.1186/s13321-016-0185-8

A Appendix

A.1 Pseudo Amino Acid Composition (PseAAC)

The Amino Acid Composition of a protein is a vector given by

$$F = [f_1, f_2, ..., f_{20}] \tag{1}$$

where f_u is the number of occurrences of the u^{th} amino acid in the sequence The PseAAC vector is given as

$$P = [p_1, p_2, ..., p_{20}, p_{21}...., p_{20+l}] \tag{2}$$

where

$$p_u = \begin{cases} \frac{f_u}{\sum_{i=1}^{20} f_i + w \sum_{k=1}^{l} \tau_k} \\ \frac{w \tau_{u-20}}{\sum_{i=1}^{20} f_i + w \sum_{k=1}^{l} \tau_k} \end{cases} \tag{3}$$

$$\tau_k = \frac{1}{L-k} \sum_{i=1}^{L-k} J_{i,i+k} \tag{4}$$

$$J_{i,i+k} = \frac{1}{\Gamma} \sum_{q=1}^{\Gamma} [\Phi_q(R_{i+k}) - \Phi_q(R_i)]^2 \tag{5}$$

L: Length of sequence, R_i: i^{th} Amino acid of the sequence, Φ_q: q^{th} function of an amino acid (E.g., Hydrophobicity, Hydrophilicity, etc.)