SciLife Project

Sarah McComas

March 18, 2016

Abstract

Here will be my abstract it will be cool

1 Introduction

Understanding protein structure is crucial to deepening our knowledge of their function and how diseases affect them and us. One of the most abundant of these are known as transmembrane proteins, which exist across membranes and participate in a variety of function such as cell transport and signaling. Of these transmembrane proteins, most of them are in the form of an alpha helix because their H-bonding properties allow for stability inside the hydrophobic lipid bilayer. About 27% of all proteins produced by humans are classified as an alpha helical transmembrane protein (1), and while understanding their exact structure is difficult, predicting the topology (i.e. where and in what orientation the protein is located in the membrane) gives enough information about how the protein function and potential targets for diseases or drug delivery. Gaining this information with traditional research is difficult and costly, which lead to methods to predict protein structure given the properties of the protein's amino acids. The first method used to do this was known as the Chou Fasman method (2), which analyzed the frequencies of the amino acids' presence in alpha helices and other structures to predict a protein structure with about a 50% accuracy (3). Since the development of machine learning algorithms and incorporation of multiple sequence alignments, the accuracy of these predictions now can reach 90% or higher (4).

Here I have drawn inspiration from such state-of-the-art predictors in creating my own predictor which determines if an alpha helix is inside the membrane or not. As these predictors before mine have, I used a multiple sequence alignment (MSA) profile generated from PSI-BLAST (5) (as well as information from single sequences in order to compare predictor quality) with information about amino acid presence at a certain position in a structure. The MSA profile is crucial in understanding evolutionary information, and can analyze the certain amino acid through evolution and see how frequent it occurs in the structure. I could then use this information as a learning algorithm for a support vector machine, or SVM (6). SVM's use supervised or unsupervised learning to do binary classification, to determine what a positive and negative example for each datapoint look like. In this case, I used an SVM only with supervised learning techniques. This trained machine can then be implemented on new data where the structure is unknown in efforts to predict it. To use the MSA data on a machine learning algorithm is a successful and common practice in todays' top predictors for topology or secondary structure such as TOPCONS, TMHMM, PSIPRED, among many others (4, 7, 8) and is the main reason why these predictors are so successful in their accuracy and efficiency.

Another successful feature of modern computational predictors is the presence of open source information, web servers, and specific databases. Here, all groups can access information quickly and easily and leads to collaboration and therefore an increase in database size and agreements on protein structure prediction. In this way, these groups can build off of each other to improve their own methods and the quality of the data presented. The two main limitations in these predictor qualities are time and accuracy. Now, accuracy is very high but there is still some disagreement between prediction models, which programs such as CASP (9) aim to solve. Many databases continue to separate that which has been computationally generated and what is experimentally proven by traditional methods, but with these improvement programs, one day we will no longer need traditional methods to validate structure.

2 Methods

I extracted features from a database of solely alpha membrane proteins with known topologies to train an SVM in supervised learning. This consisted of about 300 sequences and each entry consisted of the accession number of the protein, the amino acid sequence, and the corresponding position with regards to the membrane (I for inside, O for outside, and M for membrane). In this case, all positions with an 'M' would become a positive example, while 'I' or 'O' would be a negative one. I also assigned each amino acid letter a corresponding number so that it was readable by the SVM. This lead to a standard SVM format, as seen and described in Figure 1.

To increase accuracy in the SVM, I created 6 sets of about 50 sequences each from my database in order to perform cross validation. This is a very important step because otherwise, when the SVM is optimizing the data to try to separate the positive and negative classifications, it will overfit and become too specific to the training example it has been given. In creating a variety of sets in which the SVM can optimize its models, it becomes easier to achieve a good balance between overfitting and achieving a high accuracy of prediction. For this reason, I trained and optimized my SVM on 5 of the 6 datasets to leave one dataset, known as the validation dataset, in which I will test on at the very end of the optimization to ensure that I haven't overfit my predictor.

Because alpha membrane proteins are important, it is not uncommon to see conservation in the protein or sequence and therefore homologs are more than likely present. If they are present and happen to be separated into different cross validation sets, this can report false accuracy when testing the model because the homologous sequences are too similar and it would be as if I had trained the SVM on the same data I tested it. To avoid this, I input my entire database into CD-HIT (10) which reported back the sequences in groups of homologs. My dataset did not have many but did have some so from this, I made my cross validation sets to ensure that if there were homologs between sequences, that they would end up in the same cross validation set.

The SVM was trained in two separate manners. Firstly, the single sequence information was used. This was not expected to create a high quality predictor but it is important to compare to the MSA profiles to this. ******** (GIVE MORE INFO) ************* The MSA profile was created from running PSI-

BLAST (5) locally. This search took several days to complete, but once finished, a position specific scoring matrix, or PSSM, was generated for every amino acid in the sequence. This PSSM correlated to each amino acid's predicted presence in evolution at that certain position. This is the main difference between the MSA data for the SVM and the single sequence information; the single sequence information merely contains a 1 or 0 depending on the amino acid exists at that position in the sequence or not, respectively. The PSSM is therefore much more thorough, but must be transformed first so that the SVM can read it as a feature value (illustrated in Figure 1) and must therefore be between 0 and 1. Before the BLAST output could be read by the SVM, it was first transformed using the sigmoidal function. ************ ILLUSTRATE SIGMOID ***************

There is no true limit on how much one can train an SVM, and here my only real limitation was time. Particularly for the PSI-BLAST output, training the SVM on all of my data would usually take about 8 to 12 hours. I began first by training the SVM on the single sequence data but did not further pursue any optimization. From SVM$^{\text{light}}$ (6), there are many options on which one can optimize beyond the default parameters, which is important for obtaining the best possible results in the predictor. After running default settings for the SVM on my MSA profile, I tried with 3 kernels, which are different algorithms for finding the right means of separating data in an n-dimensional space. I tried these kernels with several parameters as well, which can customize the kernel. This can be, for example, defining a trade-off between misclassification and the simplicity of the separation (the c parameter) or how constrained the model should be (the *gamma* parameter) (11).

ACC and MCC here??

Figure 1- each line began with a target value (a positive or negative number depending on structural position). Show maybe the PSIBLAST output. explain what the PSSM output looks like, the amino acid number, and what a single sequence output would look like instead.

3 Results

Once optimized, the predictor was capable of predicting protein topology with approximately 85% accuracy. These results in comparing the final models can be found in Figure 2 and Table 1.

4 References