Spring 2020 CS123A Bioinformatics Project Report

Project Title: PSSM Replication

NAME: Azael Zamora

ABSTRACT

The following report will describe the process of implementing the PSSM (Position-Specific Scoring Matrix) algorithm that will utilize a pseudo count in order to help overcome the relatively lack of data from an input of sequences. Performance of the PSSM is accomplished by performing the necessary steps to create a PSSM that will also include a pseudocount value. Data that was utilized for this project are protein sequences that are available in the NCBI database, that were acquired by running a PSI-BLAST on the accession number NP_000509, and were used as the input sequences for the algorithm to create a PSSM. Results indicate that the approach to this project is somewhat similar to the approach that is done by PSI-BLAST since yielded similar values in the PSSMs. The impact of the results that were gathered from this project, provide further support in how utilizing a pseudocount can potentially help overcome the lack of data from a small set of sequences or from relatively similar protein sequences. The implementation of providing pseudocounts that are derived from a substitution matrix can significantly impact the data that is computed from the algorithm, since it allows for other amino acids to potentially occur in a specific position of a sequence as opposed to only having the most frequent amino acid occurring in a similar sequence. Future work for this implementation of the algorithm, will revolve around potentially using the PSSM results from the project to aid in the selection of new similar protein sequences.

Key Words: PSSM (Position-Specific Scoring Matrix), PFM (Position-Frequency Matrix), PPM (Position-Probability Matrix), Pseudo count, BLOSUM-62 Substitution Matrix, PSI-BLAST, NCBI, PSSM Viewer

Table of Contents

List of Figures

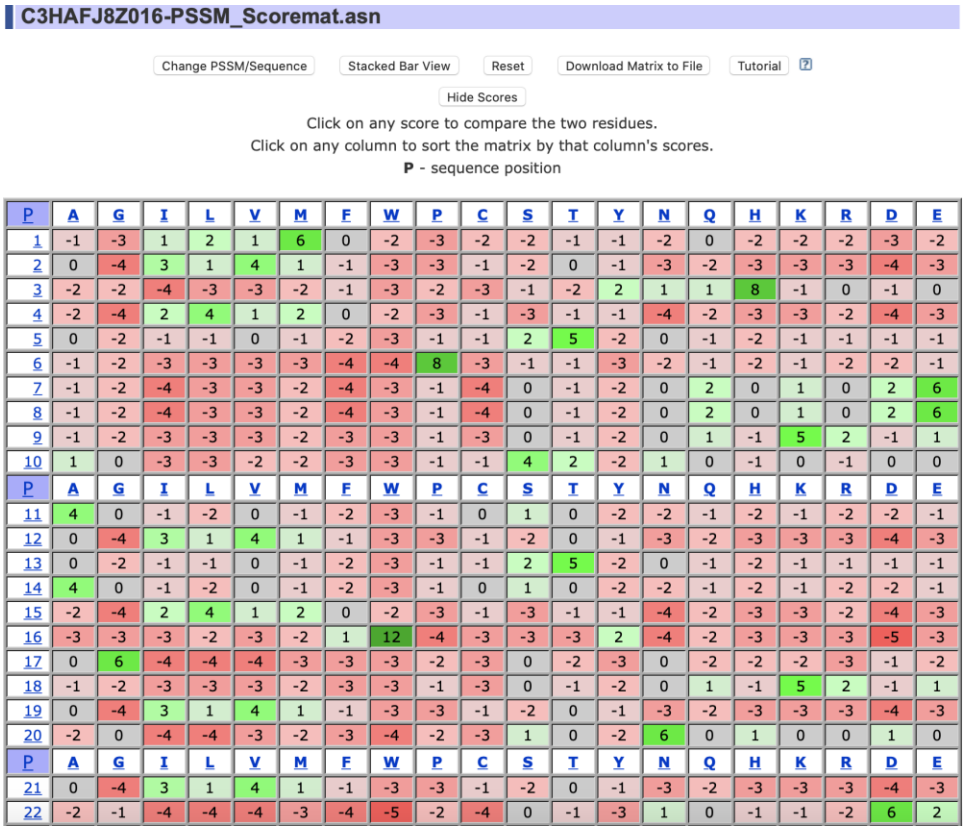
INTRODUCTION

The problem that will be addressed is how pseudocounts can help overcome the lack of data when creating a PSSM. For reference, in order to generate a PSSM, a set of sequences is required which are aligned by a common reference either resulting from a database search or a multiple alignment (1). A common problem that the PSSM can have from using similar sequences is the lack of true reside preferences since the particular amino acid type that commonly occurs in a position can be mainly used to affect the overall generation of the PSSM with very little information for each position. One way around the lack of data is to use dependent pseudocount values similarly to the PSI-BLAST which utilizes a substitution scoring matrix for pseudocounts when generating a PSSM (2). Based on how pseudocounts can potentially help overcome the lack of data when generating a PSSM, the task of this project is to implement a PSSM algorithm that will utilize pseudocount values that will be dervived from the BLOSUM-62 substitution scoring matrix to provide a better distribution value of an amino acid appearing in a position of the sequence. The reason as to why I have decided to use the BLOSUM-62 matrix for pseudocount values in the PSSM, is due to the fact that the PSI-BLAST has accomplished a similar approach in using a scoring matrix as pseudocount values, which will be used to compare the overall results from the PSSM that will be implemented for this project.

BACKGROUND

Given that the project will mainly focus on implementing a PSSM algorithm that utilizes the BLOSUM-62 Substitution Matrix for pseudocount values, a similar work that has been accomplished is the PSI-BLAST. The PSI-BLAST utilizes a scoring matrix in order to accomplish a standard BLASTP in order to obtain a list of similar sequences, and from there the scoring matrix and the list of sequences are used in order to create a PSSM (3). With the PSI-BLAST, the scoring matrix is utilized as pseudocount distribution, and with each iteration that occurs, the scoring matrix has a more accurate distribution based on the data from which they were derived (1). Once an iteration is completed after running the initial BLASTP, a PSSM will be generated based on the query sequence length, and will an *n* by 20 matrix in which n is the number of rows form the query sequence, and there will be 20 columns which is a column for each amino acid that could potentially occur in the position (3). Since the PSI-BLAST list of sequences will be similar to the query sequence, the scoring matrix that the program utilizes provides different pseudocount values for each amino acid per position, in order to overcome lack of amino acid occurrences, and provide further data for the PSSM to contain. As such, by utilizing a scoring matrix and the query of sequences from the BLASTP, each iteration that is accomplished in the PSI-BLAST has the possibility of adding additional similar sequences

that is determined by the PSSM that was generated using the previous set of sequences. The end result of this project will be a PSSM that will be in similar dimensions to the PSSM that is generated in the PSI-BLAST, and will utilize the Blosum scoring matrix for pseudocount values similarly to how the PSI-BLAST generates a PSSM. In short, the PSI-BLAST's approach to generating a PSSM and using a scoring matrix for pseudocounts is similar to what the project will aim to accomplish, and as such the PSI-BLAST will be used as reference to determine whether or not the project will yield ideal results.



C3HAFJ8Z016-PSSM_Scoremat.asn

Change PSSM/Sequence | Stacked Bar View | Reset | Download Matrix to File | Tutorial

Hide Scores

Click on any score to compare the two residues.
Click on any column to sort the matrix by that column's scores.
P - sequence position

P	A	G	I	L	V	M	F	W	P	C	S	T	Y	N	Q	H	K	R	D	E
1	-1	-3	1	2	1	6	0	-2	-3	-2	-2	-1	-1	-2	0	-2	-2	-2	-3	-2
2	0	-4	3	1	4	1	-1	-3	-3	-1	-2	0	-1	-3	-2	-3	-3	-3	-4	-3
3	-2	-2	-4	-3	-3	-2	-1	-3	-2	-3	-1	-2	2	1	1	8	-1	0	-1	0
4	-2	-4	2	4	1	2	0	-2	-3	-1	-3	-1	-1	-4	-2	-3	-3	-2	-4	-3
5	0	-2	-1	-1	0	-1	-2	-3	-1	-1	2	5	-2	0	-1	-2	-1	-1	-1	-1
6	-1	-2	-3	-3	-3	-3	-4	-4	8	-3	-1	-1	-3	-2	-1	-2	-1	-2	-2	-1
7	-1	-2	-4	-3	-3	-2	-4	-3	-1	-4	0	-1	-2	0	2	0	1	0	2	6
8	-1	-2	-4	-3	-3	-2	-4	-3	-1	-4	0	-1	-2	0	2	0	1	0	2	6
9	-1	-2	-3	-3	-3	-2	-3	-3	-1	-3	0	-1	-2	0	1	-1	5	2	-1	1
10	1	0	-3	-3	-2	-2	-3	-3	-1	-1	4	2	-2	1	0	-1	0	-1	0	0
P	A	G	I	L	V	M	F	W	P	C	S	T	Y	N	Q	H	K	R	D	E
11	4	0	-1	-2	0	-1	-2	-3	-1	0	1	0	-2	-2	-1	-2	-1	-2	-2	-1
12	0	-4	3	1	4	1	-1	-3	-3	-1	-2	0	-1	-3	-2	-3	-3	-3	-4	-3
13	0	-2	-1	-1	0	-1	-2	-3	-1	-1	2	5	-2	0	-1	-2	-1	-1	-1	-1
14	4	0	-1	-2	0	-1	-2	-3	-1	0	1	0	-2	-2	-1	-2	-1	-2	-2	-1
15	-2	-4	2	4	1	2	0	-2	-3	-1	-3	-1	-1	-4	-2	-3	-3	-2	-4	-3
16	-3	-3	-3	-2	-3	-2	1	12	-4	-3	-3	-3	2	-4	-2	-3	-3	-3	-5	-3
17	0	6	-4	-4	-4	-3	-3	-3	-2	-3	0	-2	-3	0	-2	-2	-2	-3	-1	-2
18	-1	-2	-3	-3	-3	-2	-3	-3	-1	-3	0	-1	-2	0	1	-1	5	2	-1	1
19	0	-4	3	1	4	1	-1	-3	-3	-1	-2	0	-1	-3	-2	-3	-3	-3	-4	-3
20	-2	0	-4	-4	-3	-2	-3	-4	-2	-3	1	0	-2	6	0	1	0	0	1	0
P	A	G	I	L	V	M	F	W	P	C	S	T	Y	N	Q	H	K	R	D	E
21	0	-4	3	1	4	1	-1	-3	-3	-1	-2	0	-1	-3	-2	-3	-3	-3	-4	-3
22	-2	-1	-4	-4	-4	-3	-4	-5	-2	-4	0	-1	-3	1	0	-1	-1	-2	6	2

Figure 1: The PSSM file that is generated by the PSI-BLAST, viewed in PSSM Viewer

DATA COLLECTED / ACCESSED

For the purpose of this project, the type of data that I needed was a list of sequences that are closely similar to one another, and would also need the PSSM file that can be downloaded after running the second iteration of the PSI-BLAST on the proteins that were selected. For the main focus of the project, protein sequences are the type of data that will be utilized. The amount of data that I believe that would suffice would

be around 10 protein sequences as a small data set, and I have also utilized another data set containing 50 protein sequences in order to test the objective of the project on a slightly bigger data set. The relationship and relevance between the type of data is crucial for the overall objective of the project, since the PSSM algorithm requires a set of sequences that are preferably closely similar to each other. The relevance of utilizing the PSSM that is obtained from the PSI-BLAST, is to compare results from the project, and determine if there are similar results in both the implementations of the PSSM algorithm. The data was obtained by running a PSI-BLAST on the hemoglobin protein for homo sapiens, which has an accession number of NP_000509.1 from the NCBI database. As show below, the ten sequences that are selected to run the second iteration of the PSI-BLAST are the sequences that provide the necessary data in an input file for the project. Once the second iteration of the PSI-BLAST is finished, there will be the option to download the PSSM file which is created from the 1st of the proteins that were initially similar to the protein that was selected.

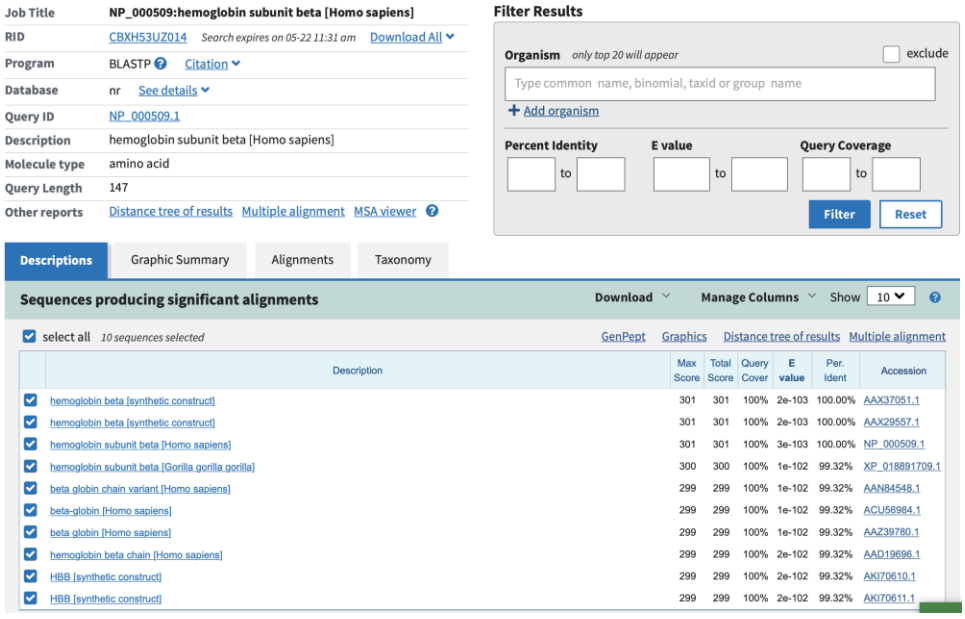


Figure 2: The 10 sequences that are utilized in the small data set, from the PSI-BLAST

APPROACH AND METHOD

The approach that I have taken, is to implement the Position-Specific Scoring Matrix (PSSM) algorithm with a pseudo count that will vary according to each different amino acid. The pseudo count will be based from the BLOSUM-62 Substitution Matrix to aid in overcoming the lack of data from a small set of sequences. For reference, I will compare the results of the implemented PSSM matrix to the NCBI PSSM Viewer of the two input files where one file contains 10 sequences, while the other file will contain 50 sequences which include the 10 sequences from the previous file. The importance of using the PSSM Viewer to compare results from the PSSM implementation, is to compare the similarities and differences of each cell value for each position, and to see how utilizing the BLOSUM-62 Matrix as pseudo counts can help provide more data, since the set of sequences will be relatively small.

The first step of the project is to save the sequences in a matrix format in which each position is a column position from the sequences. Afterwards, the following step is to create the Position-Frequency Matrix which will contain the number of occurrences of the 20 amino acids per position for all of the sequences that are provided. An important thing to note is that as the PFM (Position-Frequency Matrix) is filled out, a consensus sequence should also be created which will contain the most frequent occurring amino acid in each position. The reason why the consensus sequence is created as the PFM is filled out, is due to the fact that the consensus sequence will be used to access the BLOSUM-62 Substitution Matrix for the pseudo counts. Once the PFM has been filled out, the PFM will be an *n* by 20 matrix, in which n represents the number of positions from the sequences, and the consensus sequence will have been created with n number of positions.

The next step of the approach will be to create the PPM otherwise known as the Position-Probability Matrix. Since the PPM will contain the probability of each amino acid occurrence rate per position, there is a high chance of various amino acids with zero occurrences per position. For the purpose of this project, to overcome the lack of data which in this case, is the number of occurrences for an amino acid, a pseudo count will be included to the amino acids with zero occurrences in a position by utilizing the BLOSUM-62 Substitution Matrix. As each amino acid column cell is accessed per position, if a cell has a value that is greater than 0, which means that there is an occurrence of an amino acid, then the frequency of the amino acid will be divided by the sum of the number of sequences that are being used and 20, which represents the 20 different amino acids. Should the cell have a value that is equal to 0, then a pseudo count

value of s(a, b) will be used which is the substitution score of the ath position of the consensus sequence and the bth column of the amino acid that is currently being calculated. The cell value will contain the quotient of the pseudo count that is divided by the sum of the number of sequences and 20 times the pseudo count value. Once each of the cell values has been updated with the pseudo count, the last step of the PPM is to obtain the log value of the absolute cell value divided by 0.05, which is the probability of each amino occurring in each position. The reason why the value 0.05 is utilized as opposed to the actual frequency value of each amino acid, is due to the fact that for this project, we are assuming that each amino acid has equal probability. Once this step has been accomplished, the Position-Probability Matrix will have been calculated, and will be able to proceed into the final step of the PSSM algorithm.

The final step is to create the Position-Specific Scoring Matrix based from the data that was obtained from the PPM. For the objective of this project, to further overcome the lack of data, a pseudo count will be added to each cell entry from the PPM to the PSSM. The pseudo count will vary from column and position, since the pseudo count will be derived from the BLOSUM-62 Substitution Matrix, such that a cell entry in the PSSM will be the sum of the cell from the PPM and the substitution value of the consensus sequence at the current position along with the current amino acid column value in the PPM. As each cell in the PSSM is being calculated, the corresponding cell value will be rounded to the nearest whole number. Once each cell of the PSSM has been calculated and rounded, the PSSM has been completed for this project, and the final result will be formatted similar to a PSSM file that can be viewed using the NCBI PSSM Viewer.

Position	A	G	I	L	V	M	F	W	P	C	S	T	Y	N	Q	H	K	R	D	E
1	−1	−3	1	2	1	6	0	−1	−2	−1	−1	−1	−1	−2	0	−2	−1	−1	−3	−2
2	0	−3	3	1	5	1	−1	−3	−2	−1	−2	0	−1	−3	−2	−3	−2	−3	−3	−2
3	−2	−2	−3	−3	−3	−2	−1	−2	−2	−3	−1	−2	2	1	0	9	−1	0	−1	0
4	−1	−4	2	5	1	2	0	−2	−3	−1	−2	−1	−1	−3	−2	−3	−2	−2	−4	−3
5	0	−2	−1	−1	0	−1	−2	−2	−1	−1	1	6	−2	0	−1	−2	−1	−1	−1	−1
6	−1	−2	−3	−3	−2	−2	−4	−4	8	−3	−1	−1	−3	−2	−1	−2	−1	−2	−1	−1
7	−1	−2	−3	−3	−2	−2	−3	−3	−1	−4	0	−1	−2	0	2	0	1	0	2	6
8	−1	−2	−3	−3	−2	−2	−3	−3	−1	−4	0	−1	−2	0	2	0	1	0	2	6
9	−1	−2	−3	−2	−2	−1	−3	−3	−1	−3	0	−1	−2	0	1	−1	6	2	−1	1
10	1	0	−2	−2	−2	−1	−2	−3	−1	−1	5	1	−2	1	0	−1	0	−1	0	0
11	5	0	−1	−1	0	−1	−2	−3	−1	0	1	0	−2	−2	−1	−2	−1	−1	−2	−1
12	0	−3	3	1	5	1	−1	−3	−2	−1	−2	0	−1	−3	−2	−3	−2	−3	−3	−2
13	0	−2	−1	−1	0	−1	−2	−2	−1	−1	1	6	−2	0	−1	−2	−1	−1	−1	−1
14	5	0	−1	−1	0	−1	−2	−3	−1	0	1	0	−2	−2	−1	−2	−1	−1	−2	−1
15	−1	−4	2	5	1	2	0	−2	−3	−1	−2	−1	−1	−3	−2	−3	−2	−2	−4	−3

Figure 3: Picture on how the output looks like, file is called “10_sequences_output.txt”

EVALUATION OF RESULTS

Based on the PSSM file that was generated as the result of the algorithm for this project, I feel like the goal in determining if a pseudo count derived from a scoring matrix could help improve the lack of data was accomplished to some degree. Utilizing the pseudo count for this project was able to provide further data for the probability score of an amino acid in a specific position. The PSSM that was generated as the output result of the program was compared to the PSSM file that was generated from the PSI-BLAST in which, the file "BZ05K4R2014-PSSM_Scoremat.asn_matrix.txt" is used to compare the PSSM output file of 10 sequences, and the file "C3HAFJ8Z016-PSSM_Scoremat.asn_matrix.txt" is used to compare the PSSM output file of 50 sequences. By comparing the PSSM output file to the file that was downloaded from the PSSM Viewer, the output file is somewhat similar to the file that was provided by the viewer. By assessing the similarities between both the respective files to some degree, the implemented PSSM algorithm that has used the BLOSUM-62 Substitution Matrix for pseudo count values, is able to help overcome the lack of the data since the sequences that were used for this project were between 98-100% similar to the NP_000509 protein sequence. Based on these results, the main question of whether or not a substitution scoring matrix can be utilized as a pseudo count is answered, since the PSSM was able to provide further data on a small set of sequences. At first, it was relatively confusing to determine how the substitution matrix can be used as a pseudo count, but once a consensus sequence was implemented in the algorithm, the values of the matrix could be accessed using both the current position of the consensus sequence and the current column position of an amino acid as a key to access the values. Once the substitution matrix was implemented, the analysis in comparing results became much clearer to understand throughout the course of this project.

CONCLUSION AND DISCUSSION

To summarize, based on the results that were obtained from the implemented PSSM algorithm that utilized the BLOSUM-62 Substitution Matrix values as pseudo count values, I can conclude from the results that utilizing a pseudo count can potentially overcome the lack of data. The way that the PSSM algorithm was approached for this project, if the pseudo count was not present throughout the algorithm, then the PSSM would contain little information about the probability that a certain amino acid could occur in a position since the data that was used for this project was relatively small. With the addition of the pseud count, the potential lack of data in the PSSM was overcome, which in turn will aid in providing several amino acids in each position a possibility of appearing

their corresponding position. Based on the approach that was taken for this project, the results that were obtained from the PSSM indicates that using a pseudo count that corresponds to the values in the BLOSUM-62 Matrix can help in overcoming the lack of data for a small set of sequences or for a set of sequences that may not provide sufficient information about the probability of certain amino acids appearing a position. By comparing the results from the project to those that were obtained to the NCBI PSSM Viewer from the PSI-BLAST, the analysis that was done for this project can help advance the scientific community in a way that other substitution matrices could be utilized as potential pseudo counts to overcome the lack of data from a set of sequences for the PSSM.

FUTURE WORK

Given that the PSSM implementation utilized values from the BLOSUM-62 Substitution Matrix for the pseudo counts, the next step would be to try to implement the PSSM in such a way that it could be a deciding factor in choosing closely related proteins. With respect to how the PSI-BLAST utilizes a PSSM after the 2nd iteration to further select closely related proteins, the next step would be to have a file that could contain a list of potential sequences that could be added based on the level of similarity that is determined by the PSSM. From there, the PSSM that was implemented would be taken into consideration in choosing a set of additional proteins from the file, and from there using the updated set of sequences, update the PSSM with the new sequences that were added. The way that the PSSM would be utilized in the manner that is discussed is by creating a consensus sequence that is contains the most frequent occurring amino acid in each position. Afterwards, given the list of potential sequences, the consensus will be utilized in order to determine the amount of similarity between the sequences and pick those that have around a 95% similarity and append it to the list of similarly related protein sequences. Then, it would repeat the process of the PSSM algorithm with the newly added sequences in order to update the PSSM cell values for each amino acid occurrence rate in a position. Once the newly updated PSSM has been established, the following step would be to compare the results of the newly PSSM created from the project, to the newly updated PSSM from running the 3rd iteration of the PSI-BLAST using the NCBI PSSM Viewer. Another thing to note would also be to compare the sequences that were added as a result from the 3rd iteration of the PSI-BLAST to the sequences added from the implemented PSSM project. Overall, the next steps of the project should it be further worked on, would require further extensive research in order to potentially have similar results, and to make sure that further implementations of the project is correctly done.

REFERENCES

1. Baum, J., & Zvelebil, M. (2007). Patterns, Profiles, and Multiple Alignments. *Understanding Bioinformatics,* 165-177

2. Agarwala, R., Altschul, S., Gertz, E., Schäffer, A., & Yu, Y. (2008). PSI-BLAST pseudocounts and the minimum description length principle

3. Hu, G., & Kurgan, L. Sequence Similarity Searching