**Project Course -**

**- A method proposal and review of Alpha helical trans-membrane topology prediction methods.**

**Abstract - Information about my dataset:-**

Recent advances in computing technology have allow for

The dataset is of membrane alpha helices. The data file is in 3 lines per a protein sequence with a protein id, protein sequence and the feature sequence. The features have 3 possibilities Inside the cell, in the membrane or outside the membrane

**Introduction**

**- Background – of features**

* Alpha helical trans-membrane proteins have a key role in life science for many reasons.
* Yet despite this important role their structural characterization remains low in the various protein databases, as their structures are difficult determine experimentally. Advances have been made with new technologies
* Problems - physiochemical features of transmembrane – feature discrimination of feature of the secondary.
  + - * Motif, singal peptides signals re-entrants

**-Background of methods and predictors**

* **TM topology predictors**
* **Methods** include HMM-TM [11], HMMTOP [12] and TMHMMfix[13]. Tools such as SOSUI [14] and PRED-CLASS [15] are designed to discriminate between globular and TM proteins, while others such as PRED-TMBB [16] specialise in the discrimination and prediction of beta-barrel TM pro-teins. A key element when constructing any prediction method is the use of a high quality data set for both training and validation purposes.

**- Other**

**Here I propose a strategy using a predictor strategy. I will predictor creation, training, predicting, optimization, validating and benchmarking.**

**Method Overview:**

**Preparation:**

* Created a bash script and a git hub repository which has structure for my project

**Main objects**

* Isolation and extraction of sequence IDs, peptide sequences and topology features with parsing of the provided training dataset file to other files
* Transforming the of the features from sequences your dataset - Encoding
* Create cross-validated sets. Protein level and window level corss validation
* Train a SVM using single sequence information, using sklearn
* Check different window sizes for the inputs
* **Add evolutionary information by running psi-blast and extracting the information**
* **Train a SVM using multiple sequence information**
* **Optimize the performance of the SVM**
* Analyze the results and compare it to previous work
* **Use random forests and a simple decision tree and compare the performance with the SVM performance.**
* **Extract the data from 50 other proteins and test the performance**
* Review the state of art for your predictor
* **Write a report**

**In-depth steps:**

**Project Preparation:**

* **Created a bash script and a git hub repository which has structure for my project**
* **Setup of a VirtualMachine and installation of Ubunta on local computer**
* **Setup of programming environment – Anaconda package**

**Training Dataset characteristic:**

**313 alpha helical trans-membrane protein sequence.**

**Three topology features consisting of Inside, Membrane and Outside.**

- id labels → idlist

- sequences → seqlist

**- features → feat\_list**

**Inputs**

**1. Extract the feature from your dataset**

**Parsing: -** A new python script was created and label *parse.py*. It objective is to open my data file at the beginning of the script and parse the file by sorting through the dataset text file *membrane-alpha.3line.txt* and separating the elements line by line. The separation of the different elements was done by the **s**plitting the elements into their own list with the condition if/else of the remainder by the modulator of 3.

- id labels → idlist

- sequences → seqlist

- features → feat\_list

The lists were indexed with the use of an enumeration of the list.The different elements were simulatanoesly written to a output file in the forloop – *idlist.txt*, *seqlist.txt* and *feat\_list.txt*.

Note: I am to decide if to leave this as a script or if to define it as a function. This is a question I going to decide on as I code more as I want to make it easy to reuse code and also to create flow in the final program.

**2. Create cross-validated sets.**

Separating into train and test datasets:

To extract the features using the sci-kit learn OneHotencoder or the Dictvectorizer into a sparse matrix format on which I can use for an input into sklearn.

Either Using the cross\_valid\_sort in scikit learn

I will need to divide the dataset into 3 or 5 different file on which to use for cross-validation in order to train and test a SVModel.

**SVM training**

* Train a SVM using single sequence information, using sklearn

Train a SVM on sequence info with sklearn. After which the model will be tested on a test set.

* Check different window sizes for the inputs

**Optimization**

* Add evolutionary information by running psi-blast and extracting the information
* Train a SVM using multiple sequence information
* Optimize the performance of the SVM

**Benchmarking**

* Analyze the results and compare it to previous work
* Use random forests and a simple decision tree and compare the performance with the SVM performance.
* Extract the data from 50 other proteins and test the performance

to be continued…………

**Results:**

**Discussion:**

**References:**

- [Scikit-learn: Machine Learning in Python](http://jmlr.csail.mit.edu/papers/v12/pedregosa11a.html), Pedregosa et al., JMLR 12, pp. 2825-2830, 2011.

- Bash - Unix etc <http://swcarpentry.github.io/shell-novice/>

- Git <http://swcarpentry.github.io/git-novice>

- Python <http://swcarpentry.github.io/python-novice-inflammation/>

- [Transmembrane protein topology prediction using support vector machines](http://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-10-159)  
- [Protein Structure Prediction using Support Vector Machine](https://www.google.se/url?sa=t&rct=j&q=&esrc=s&source=web&cd=3&cad=rja&uact=8&ved=0ahUKEwiFiYDd3q_SAhUqDMAKHXIECQsQFggyMAI&url=http%3A%2F%2Fairccse.org%2Fjournal%2Fijsc%2Fpapers%2F2112ijsc06.pdf&usg=AFQjCNEkEni07XAjnNqkFXQ51SilTRFrkQ&sig2=OSUE53raGY7oqx5bQJTZXA)   
- [Membrane protein structural bioinformatics](https://www.ncbi.nlm.nih.gov/pubmed/22075226)  
- [Improving transmembrane protein consensus topology prediction using inter-helical interaction](http://www.sciencedirect.com/science/article/pii/S0005273612001885)  
- [A Web-Server for Machine Learning Prediction of Protein Functional Families from Sequence Irrespective of Similarity](http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0155290)

-[Jones DT. (1999) Protein secondary structure prediction based on position-specific scoring matrices. J. Mol. Biol. 292: 195-202.](http://www.idealibrary.com/)

-Noble WS (2009) A Quick Guide to Organizing Computational Biology Projects. PLoS Comput Biol 5(7): e1000424.

-Chih-Wei Hsu, Chih-Chung Chang, and Chih-Jen Lin. A Practical Guide to Support Vector Classification. http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf

- DataCamp.com – sklearn, python, git - cheatsheetshttps://www.datacamp.com/community/blog/scikit-learn-cheat-sheet#gs.eNo17Bc