**BIOINFORMATICS**

**Professor- Bingyun Sun**

**Research Assistant- Amritpal Singh**

**System Requirement: -**

* Linux, Windows, Mac Os
* Python 3+ required.

**Python Libraries: -**

* Pandas, NumPy, Re(Regular Expression operators), Matplotlib, Sys(System-specific parameters and functions).

**Downloading data from UniProt website:-**

* Go to <https://www.uniprot.org/>
* Search Humans, then click on reviewed articles.
* Customize results table and select Entry, Entry name, Protein names, Genes names, Length, Organism, Mass, Status, Disulfide bond, Glycosylation.
* Make sure to select all these columns if not considered then program will give error. If interested in more fields then add them along with the above columns.
* Download the file in either excel format or tab-separated format.

**Running the File: -**

* Start terminal and change the directory to place where all files are present.

**Downloaded file is in excel format: -**

* Run this command in terminal- **python3 Bioinformatics\_excel.py input.xlsx**

**Downloaded file is in tab-separated format:-**

* Run this command in terminal- **python3 Bioinformatics\_tab.py input.tab**

Here **‘input.xlsx’** and ‘**input.tab’** are those files which are downloaded from Uniprot website.

**Summary: -**

The main strategy is to work with the sulfide position and glycosylation positions and to figure out how the glycosylation position are inside sulfide bonds, outside sulfide bonds, N-terminus distance, C-terminus distance and so on. So, the first stage is to manage data and the second stage is to do data analysis.

**Bioinformatics (Excel/Tab) Code: -**

In the Bioinformatics file there is a lot of data cleaning and ETL (Extract-Transform-Load) is done. The new data frame created to store all input data.

* **'Organism’** column only contain data which contains **‘Homo sapiens’** as we are only interested in Human species.
* **‘get\_sulfide\_value’** function used to get all sulfide pairs.
* ‘**get\_gly\_value’ and ‘getGlycoNLinked’** function used to get glycosylation positions.
* Using loops and properties of the Python library 're' to get values and finding positions between sulfide bond and glycosylation.
* Calculating inter-bond Distance
* Calculating intra-bond distance
* Numbers of pairs of sulfide bonds
* First sulfide pair to N-terminus distance
* Last sulfide pair to C-terminus distance
* Calculate average distance of intra-bond distance.
* Calculating sulfide pairs score.

**Sub Cases: -**

Calculating average distance of those pairs of disulphide bonds that have Glycosylation Inside.

* Using the ‘loop’ if the glycosylation position is inside disulphide pair then that particular disulfide pair position changed to ‘nil’. The reason for that it will be easier to find those pairs which do not have a glycosylation position inside.
* Then from that data we took the only pairs which not equal to ‘nil’.
* Calculating length between sulfide bonds from above data.
* Convert those length from String to the integer values.
* Calculating the average length of sulfide bonds which do not have glycosylation inside.

Outputted the data in excel form.