"Investigating Amino Acid Enrichments and Patterns: Understanding Biases in Liquid-Liquid Phase Separation"

-Computation User Guide-

Version Control

All programs and computational results were written and obtained using:

-Python **3.11.5**

-Spyder **5.4.3**

-Anaconda Distribution **2.5.2**

-LocalCIDER **0.1.18**

Anaconda Distribution with Spyder available here: <https://www.anaconda.com/download>

LocalCider download and setup guide available here:

<https://pappulab.wustl.edu/localCIDER.html>

In this Guide there will be images for easy step to step reproduction and usage, these images will be from **Spyder Integrated Development Environment**, other properly configured Integrated Development Environments running the same Python version can be used but are not covered visually by this guide, the computational logic still applies.

GitHub file access and download

The Folder containing:

-Python programs used to archive computational results

-Datasets used to achieve computational results

-Computational results

-User Guide to reproduce said Computational results

-The paper "Investigating Amino Acid Enrichments and Patterns: Understanding Biases in Liquid-Liquid Phase Separation."

Is Available here: https://github.com/BPSlab/UnderstandingLLPS.git.

GitHub file access and download

After following the Link to the GitHub repo your navigator of choice should look like this:

A screenshot of a computer

Description automatically generated

Simply click the **Green Code button,** and, in the dropdown menu click **Download ZIP.**

GitHub file access and download

Extract the Downloaded ZIP file to your desired Location, The Resulting Folder is Now ready to access, it is referred in this guide as the **Main Project Directory** (Noted in Red) and when open it presents the following structure:

A screenshot of a computer

Description automatically generated

In the following pages each one of these directories will be elaborated on further.

File Structure

-Datasets-

In this directory are contained all Datasets used to reach the computational results that were not also themselves computational results from Python Programs, here follows a brief description of each:

**-Database.xlsx**: composed of 3 sheets one for whole Proteins, one for Droplet promoting regions and another for the opposite regions.

**-****Database By Families.xlsx**: composed of 6 sheets, one for each family, otherwise same format as **Database.xlsx**.

**-Negative DataBase.xlsx:** composed of one sheet each line contains the uniport id and sequence of respective protein followed by the probability for phase separation and the number of Droplet promoting regions in said protein.

**- Motif Picks.xlsx:** composed of one sheet each line contains a Motif to be analyzed via python program.

**- Motifs By Families.xlsx:** composed of 6 sheets, one for each family, each line contains a Motif to be analyzed via python program.

File Structure

-Programs-

In this directory are contained all Programs used to reach the computational results, here follows a brief description of each, more detailed usage will be in the pages following:

**-** **STATITIAN.py:**

**Input:** Protein Database.xlsx, Negative Database.xlsx, Database By Families.xlsx

**Output:** Protein Stats.xlsx, DPR Stats.xlsx, NODPR Stats.xlsx, Negative Stats.xlsx

**Short Description:** Takes input\_dataframe,output\_dataframe,output\_file\_name and control\_variable and analyses sequences in input\_dataframe saves them in output\_dataframe and outputs a excel sheet to the ‘Results’ Folder with output\_dataframe information that being the size of the sequence, residue count and percentage in said sequence and family and percentage of residues in that family for the sequence.

**-** **CIDER FOR PROTEINS.py:**

**Input:** **Database.xlsx**

**Output:** User Named.xlsx

**Short Description:** Takes input\_dataframe, output\_file\_name and control\_variable and runs CIDER Analysis in sequences in input\_dataframe, and outputs an excel sheet to the ‘Cider for proteins’ Folder.

**-SELECTOR.py:**

**Input:** One of the following: **Database.xlsx, Database By Families.xlsx**.

**Output:** One .xlsx File in a Folder Both named after the length of motif being Analyzed.

**Example for Pentapeptide:**

**Main Project Directory**\Results\Motifs\Pentapeptide\ **Pentapetide.xlsx**

**Short Description:** For a user specified length of motif, takes input\_dataframe and control\_variable and counts Presence and Frequency of motifs of said length in all Sequences of input\_dataframe.

**- FREQUENCY.py:**

**Input:** **Motif Picks.xlsx** and **one** of the following: **Database.xlsx, Database By Families.xlsx.**

(**Main Project Directory**\Results\Motif Frequency\User Defined Folder\MOTIF.xlsx)

One for each Motif in the Motif picks.xlsx File

**Short Description:**

For Motif in Motif Picks.xlsx calls Function that takes a motif initializes two list variables one is the list of the Uniprot IDs of Proteins that the motif appears in, the other is the list of the Uniprot IDs of Proteins that the motif does not appear in, next it cycles through the lines of either:

**Database.xlsx, Database By Families.xlsx or Negative Database.xlsx**

For each sequence, counts Motif appearance and frequency in that sequence. After the last sequence counts the number of sequences the Motif is in and the number of those it is not and writes it to an excel file named after the Motif.

**- COMBINER.py:**

**Input:**  One **Motif Picks.xlsx** and One **Database.xlsx**

**Output:** User Named.xlsx in

(**Main Project Directory**\Results\ Peptide Design)

**Short Description:** Makes a Table, each row contains a list of 3 motifs, the respective Score detailed in **Triple Motif Coexistence Score.docx** and the shortest peptide that those 3 Motifs can assemble. Calculates score for each Motif.

**- CIDER FOR PEPTIDES.py:**

**Input:**  **Motif Combinations.xlsx**

**Output:** User Named.xlsx in

(**Main Project Directory**\ Results\ Cider for peptides)

**Short Description:** For all possible Peptide combinations with score > 10 Makes a Table, each row contains a list of 3 motifs, the respective Score detailed in **Triple Motif Coexistence Score.docx** and one of six peptides that those 3 Motifs can assemble and the CIDER analysis of the respective peptide.

**- MATRITIAN.py:**

**Input:**  **Motif Combinations.xlsx**

**Output:** MATRIX.xlsx in

(**Main Project Directory**\ Results\ Converted Matrix)

**Short Description:** Reads list of peptides in Motif Combinations.xlsx and for each motif list creates a new cell with the peptide score, thus originating a matrix format.

**- FILTER FOR PEPTIDES.py:**

**Input:**  **Peptides Cider.xlsx**

**Output:** Cider for peptides filtered.xlsx in

(**Main Project Directory**\ Results\ Cider for peptides)

**Short Description:** Takes results from Cider for peptides.xlsx and filters based on CIDER parameters.

File Structure

-Results-

In this directory are contained all computational results organized as follows:

A screenshot of a computer

Description automatically generated

Due to GitHub file restrictions, inside of **Motif Frequency** you will find.ZIP Files for results computed with Protein Database and Negative, as such with each Run of the Program

**MANUAL FILE SORTING IS ADVISED**

It is recommended to zip all MOTIF.xlsx files to an archive of your choosing.

Program Usage Guide

-STATITIAN-

**For Regular usage:**

-Run the Program

A screenshot of a computer

Description automatically generated

**For Advanced usage:**

-If you want to Run the program for different data sets you can easily do so if your files match the way our files are Formatted, and you follow the following steps:

-Add a new Data frame from your desired Database excel

A screen shot of a computer code

Description automatically generated

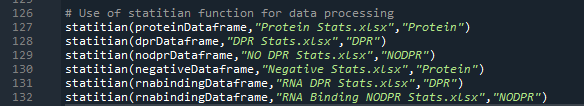
-Add line in format:

statitian(input\_dataframe, output\_file\_name, control\_variable)

**input\_dataframe** is your Dataframe defined in the step above.

**output\_file\_name** is your desired output file name in string format.

**control\_variable** is either "Protein", ”NODPR” or "DPR" and controls if you run statistics in the full protein or just the DPR or NDPR sequences for files formatted like our **Database.xlsx** or **Database By Families.xlsx**



Program Usage Guide

-SELECTOR-

**For Regular usage:**

-Choose what Dataframe to Look for Motifs in, the following Dataframes are already defined and ready to use:

negativeDataframe

dprDataframe

rnabindingDataframe

dnabindingDataframe

chromatinbindingDataframe

regulationDataframe

hydrolaseDataframe

structureDataframe

-Choose what Motif Length to Look for in the sequences, lengths supported go up until Hexapeptides.

(Example usage for structureDataframe and Pentapeptides).



**For Advanced usage:**

-If you want to Run the program for different data sets you can easily do so if your files match the way our files are Formatted, and you follow the following steps:

-Add a new Data frame from your desired Database excel

A screen shot of a computer code

Description automatically generated

-If you want to Run the program for different Motif Lengths follow the following steps:

-Add a new entry in peptideSizeDictionary

A screen shot of a computer

Description automatically generated

-Add line in format:

selector(input\_dataframe,peptide\_size,control\_variable)

**input\_dataframe** is your Dataframe defined in the step above.

**peptide\_size** is your desired Motif length.

**control\_variable** is either "Protein", “NDPR” or "DPR" sequences

Program Usage Guide

-FREQUENCY-

**For Regular usage:**

-Choose what Dataframe to Count Motifs in, the following Dataframes are already defined and ready to use:

proteinDataframe

dprDataframe

nodprDataframe

rnabindingDataframe

dnabindingDataframe

chromatinbindingDataframe

regulationDataframe

hydrolaseDataframe

structureDataframe

- add a line in this format:

makeMotifList(input\_dataframe,output\_path\_name,control\_variable)

**For Advanced usage:**

-If you want to Run the program for a different list of motifs simply replace the Motifs in **Motif picks.xlsx**

Program Usage Guide

-COMBINER-

**For Regular usage:**

-Choose what Sequence Dataframe to Count Motif interactions in, the following Dataframes are already defined and ready to use:

dprDataframe

-Choose what Motif Dataframe to Count Motif interactions for, the following Dataframes are already defined and ready to use:

motifDataframe

Choose one and run main with it both as entry value, together with the control variable and desired output filename.

(Example usage).

****

-Run the program

Program Usage Guide

-CIDER FOR PEPTIDES-

**For Regular usage:**

-Run the program

**For Advanced usage:**

-If you want to Run the program for a different list of motifs in another type of sequence run COMBINER for those then add a line reading the resulting excel:



-Add line in format:

outputMaker(input\_dataframe,output\_filename)

**input\_dataframe** is your Dataframe defined in the step above.

**output\_filename** is your desired output file name in string format.

(Example usage).



-Run the program