

Protein Conservation Program Manual

Version 1.0

November 19, 2023

Contents

1. Overview
2. Help manual
 1. Inputs
 1. Working directory
 2. Taxon
 3. Protein family
 4. Confirm number of sequences and species
 5. Motifs scanning parameters
 2. Preliminary results and outputs
 1. Protein sequence conservation analysis
 2. Motifs scanning
 3. Secondary structure prediction
 4. Phylogenetic tree analysis
 5. Results display
 6. Example outputs
 7. Interpreting outputs
3. Maintenance manual
 1. Code structure
 2. Functions
 1. Input validation
 2. Data retrieval
 3. Programs
 1. EMBOSS Plotcon
 2. EMBOSS Patmatmotifs
 3. EMBOSS Garnier
 4. PhyML
4. References

1. Overview

The purpose of this program is to perform a protein sequence conservation analysis. The program utilizes applications from EMBOSS (European Molecular Biology Open Software Suite), including Plotcon, Patmatmotifs, and Garnier, and the package PhyML. Given a user-defined subset of the taxonomic tree and the name of a protein family, the program will determine the level of conservation in the protein sequences in the dataset using Plotcon. The protein sequences included in the dataset are retrieved from the NCBI Protein database and exclude predicted and hypothetical sequences. Plotcon takes the sequence alignment created using Clustal Omega and plots the similarity found in each position in a given window of the protein sequence. The multiple sequence alignment is used in protein conservation analysis since it allows the visualization of differences and similitudes within the sequences of the protein of interest. Next, the sequences will be scanned using Patmatmotifs. This is to establish if motifs from the PROSITE database are associated with the dataset. When studying unknown proteins, it is useful to determine if their sequences contain motifs. Motifs are patterns of short sequences that are associated with specific functions (1). The PROSITE database comprises over a thousand protein families and domains. Following this, the secondary structure prediction of the protein sequences will be performed with Garnier. The program applies the original Garnier Osguthorpe Robson algorithm for the predictions (2). Given that the location of domains in a sequence relates to the physical properties of the protein that lead to its stabilization and different functions (e.g., membrane association) (3), studying the secondary structure of proteins is of great importance. Lastly, a phylogenetic tree will be constructed with PhyML. PhyML estimates the maximum likelihood of phylogenies from the protein sequence alignment. Phylogenetic trees show the evolutionary relations between sequences and can be used to predict residues with critical functions (4). The phylogenetic tree will be restricted to a maximum of 100 sequences since the computational time grows exponentially as the number of sequences increases. Each step and output are mentioned while running the analysis. The program is written for a Python3 interpreter.

2. Help manual

To execute the program, go to the directory that contains the file `protein_conservation_analysis.py` and type in the command-line:
`./protein_conservation_analysis.py`

The following welcome message will be displayed on screen:

```
-----  
Welcome,  
  
The purpose of this program is to perform a protein sequence conservation analysis.  
  
You will be asked to provide the name of a subset of the taxonomic tree and the name of a protein family you are interested in.  
The program will also ask to provide a path to a directory to store the results.  
The program will carry out the following analysis:  
  
    - Protein sequence conservation analysis (EMBOSS program: plotcon)  
    - Identification of motifs from the PROSITE database (EMBOSS program: patmatmotifs)  
    - Prediction of protein secondary structure (GOR method) (EMBOSS program: garnier)  
    - Phylogenetic tree construction (PhyML)  
  
-----
```

2.1 Inputs

Example inputs are shown on every step, otherwise, they require an integer as the answer or confirmation by typing y or n, for yes and no.

2.1.1 Working directory

The program requires a directory to store all program and output files. There are two options, the program can create a directory named Protein_analysis in the current working directory, or the user can specify a valid path with a chosen name for the directory. If a directory with the given name in the specified path already exists, the program will delete it. WARNINGS are issued by the program to highlight this. Enter 1 or 2 on the command-line to proceed.

```
A directory will be created to store all the files required for the program and to store the results.
How would you like to proceed?
  1. The program will create the Protein_Analysis directory in the current directory
  2. Enter the path and name of the new directory

WARNING: If a directory with the same name is present in the given path, the program will delete it.

Enter a number from the given options:
2
```

Directory names that contain spaces will not be considered.

```
WARNING: If a directory with the same name is present in the given path, the program will delete it.

Enter the path and the name of the directory to store the program files:
(The new directory name must contain only letters, numbers or _, e.g., /home/ICA2/Protein_analys)
/home/s1234567/Protein analysis

The input is not valid.

WARNING: If a directory with the same name is present in the given path, the program will delete it.

Enter the path and the name of the directory to store the program files:
(The new directory name must contain only letters, numbers or _, e.g., /home/ICA2/Protein_analys)
█
```

2.1.2 Taxon

The program requires to enter either a taxon name or the NCBI taxon ID for the analysis and then to confirm the selection.

```
Would you like to provide the taxon name or the NCBI taxon id?
  1. Taxon name
  2. NCBI taxon ID

Enter a number from the given options:
1

Which subset of the taxonomic tree do you want to analyze?
(e.g., birds, or g-proteobacteria)
birds

Searching for taxon name matching input...

Do you want to analyze Aves(birds)? y/n
y█
```

```
Would you like to provide the taxon name or the NCBI taxon id?
  1. Taxon name
  2. NCBI taxon ID

Enter a number from the given options:
  2

Enter the ID of the taxonomic subset you would like to analyze:
  8782

Searching for taxon ID matching input...

Do you want to analyze Aves(birds)? y/n
  y
```

2.1.3 Protein family

The program requires to enter the name of the protein family for the analysis. The name must be written as a singular noun.

```
Enter the name of the protein family you would like to analyze:
(write the protein family name in the singular form, e.g., catalase)
  glucose-6-phosphatase

Searching for sequences to analyze...
Wait for confirmation to proceed with analysis.
```

If no results are found for the given protein family and taxon, the program gives the option to modify the query or exit the program.

```
Enter the name of the protein family you would like to analyze:
(write the protein family name in the singular form, e.g., catalase)
  glucose-6-phosphata

Searching for sequences to analyze...
Wait for confirmation to proceed with analysis.

No results were found for the query.

How do you want to proceed?
  1. Modify query.
  2. Exit program.

Enter a number from the given options:
  █
```

2.1.4 Confirm number of sequences and species

Next, the program will ask to confirm the number of sequences and species to include in the analysis. Predicted and hypothetical sequences are excluded from the query.

```
There are 82 sequences associated with the query.

How do you want to proceed?
  1. Continue with analysis.
  2. Modify query.
  3. Exit program.

Enter a number from the given options:
  1

Wait for confirmation to proceed with analysis.
```

```
There are 82 sequences from 75 species in the dataset.
```

```
How do you want to proceed?
```

1. Continue with analysis.
2. Modify query.
3. Exit program.

```
Enter a number from the given options:
```

```
1
```

If the query includes more than 1,000 sequences, a message will be prompted to ask the user how to proceed.

```
WARNING: There are more than 1000 sequences associated with the query.
```

```
The program can analyze up to 1000 sequences.
```

```
Refine your search or continue analysis with the first 1000 sequences in the query.
```

```
How do you want to proceed?
```

1. Continue with analysis.
2. Modify query.
3. Exit program.

```
Enter a number from the given options:
```

```
1
```

```
Wait for confirmation to proceed with analysis.
```

```
WARNING: There are 429 species in the dataset.
```

```
How do you want to proceed?
```

1. Continue with analysis.
2. Modify query.
3. Exit program.

```
Enter a number from the given options:
```

```
1
```

2.1.5 Motifs scanning parameters

The application Patmatmotif can include in the report simple post-translational modification sites. The user must select an option to proceed. It is the last input required from the user.

```
While determining whether any known motifs are associated with the sequences, do you want to include simple post-translational modifications in the results?
```

1. Yes
2. No

```
Enter a number from the given options:
```

```
1
```

Once all the input is completed, the following message will display on screen:

```
STARTING ANALYSIS...
```

```
Depending on the size of the dataset, it can take up to several minutes to complete all analysis.  
Preliminary results will be shown on the screen.
```

For example, a given dataset with 1,000 sequences of 1,000 aa would take ~20min to complete. Another dataset with 80 sequences of 500 aa would take ~8min to complete.

2.2 Preliminary results and outputs

The starting of each analysis will be displayed on screen as well as the name of the output files generated in each step. Some analysis will display preliminary or summary results. The following sections contain images of a query for glucose-6-phosphatase proteins from birds.

2.2.1 Protein sequence conservation analysis

```
STARTING PROTEIN SEQUENCE CONSERVATION ANALYSIS

Plot conservation of a sequence alignment
Created glucose_6_phosphatase_Aves_taxID8782_plotcon.pdf
Plot conservation of a sequence alignment
Created glucose_6_phosphatase_Aves_taxID8782_plotcon.1.png

The conservation plot will be displayed on screen once ALL analysis are FINISHED.

PROTEIN SEQUENCE CONSERVATION ANALYSIS FINISHED

-----
```

2.2.2 Motifs scanning

The results files contain an EMBOSS report with information on the location and score of the motifs found in the dataset. The final csv file contains all counts per sequence, indicating sequence number in the dataset, sequence accession number and species as well.

```
STARTING MOTIFS SCANNING

Summary of motifs found in dataset:

count  AMIDATION  ASN_GLYCOSYLATION  CAMP_PHOSPHO_SITE  CK2_PHOSPHO_SITE  MYRISTYL  PKC_PHOSPHO_SITE
mean   0.914634    1.134146           0.780488           1.195122          7.902439   2.329268
std    0.391301    0.437789           0.445121           0.710399          1.037682   1.155287
min    0.000000    0.000000           0.000000           0.000000          4.000000   0.000000
25%    1.000000    1.000000           1.000000           1.000000          8.000000   2.000000
50%    1.000000    1.000000           1.000000           1.000000          8.000000   2.000000
75%    1.000000    1.000000           1.000000           2.000000          8.000000   3.000000
max    2.000000    2.000000           2.000000           3.000000          14.000000  6.000000

The following files were created:

- Patmatmotifs_result_files (directory with files with motif scanning results for each sequence)
- Associated_motifs.txt (file with motif scanning with results from all sequences)
- Motifs_counts.csv (file with data frame with counts per motif per sequence)
- Motifs_counts.png (file with plot of total motifs counts)
- Motifs_counts_graphs (directory with plots of motifs counts per sequence)

MOTIFS SCANNING FINISHED

-----
```

2.2.3 Secondary structure prediction

The results file contains an EMBOSS report with the secondary structure prediction.

```
STARTING SECONDARY STRUCTURE PREDICTION

Predict protein secondary structure using GOR method
Error: ajSeqTypeCheckIn: Sequence must be protein sequence without BZ U X or *: found bad character 'X'

The following file was created:

- glucose-6-phosphatase_Aves_taxID8782_secondary_structure.garnier (file with secondary structure
sequence)

SECONDARY STRUCTURE PREDICTION FINISHED

-----
```

2.2.4 Phylogenetic tree analysis

If the dataset includes more than 100 sequences, the phylogenetic tree will only consider the first 100 sequences according to the order in the sequence alignment from Clustal Omega. PhyML will display information regarding tree construction while running. This is the most time-consuming analysis in the program. Due to limitations in the number of characters for the species name, the tree shows the beginning of the species name and ends with the number of the sequence in the dataset, which can be access on the cvs file with the motifs counts.

```

STARTING PHYLOGENETIC TREE ANALYSIS

Running the analysis on 64 CPUs..

Command line: /usr/lib/phyml/bin/phyml-mpi -i Protein_alignment.phy -d aa

=====
. Sequence filename: Protein_alignment.phy
. Data type: aa
. Alphabet size: 20
. Sequence format: interleaved
. Number of data sets: 1
. Nb of bootstrapped data sets: 0
. Compute approximate likelihood ratio test: yes (aBayes branch supports)
. Model name: LG
. Proportion of invariable sites: 0.000000
. Number of subst. rate catgs: 4
. Gamma distribution parameter: estimated
. "Middle" of each rate class: mean
. Amino acid equilibrium frequencies: model
. Optimise tree topology: yes
. Starting tree: BioNJ
. Add random input tree: no
. Optimise branch lengths: yes
. Minimum length of an edge: 1e-08
. Optimise substitution model parameters: yes
. Run ID: none
. Random seed: 1700404550
. Subtree patterns aliasing: no
. Version: 3.3.3:3.3.20190909-1
. Byte alignment: 1
. AVX enabled: no
. SSE enabled: no
=====

379 patterns found (out of a total of 528 sites).

271 sites without polymorphism (51.33%).

Computing pairwise distances...

Building BioNJ tree...

Note: taxon 'S_mage138' is a duplicate of taxon 'S_mend36'.
Note: taxon 'S_humbo39' is a duplicate of taxon 'S_mend36'.
Note: taxon 'S_demer37' is a duplicate of taxon 'S_mend36'.
Note: taxon 'P_mantar31' is a duplicate of taxon 'P_papua19'.
Note: taxon 'M_antip34' is a duplicate of taxon 'M_antip35'.
Note: taxon 'M_antip33' is a duplicate of taxon 'M_antip35'.
Note: taxon 'M_antip32' is a duplicate of taxon 'M_antip35'.
Note: taxon 'E_rlm62' is a duplicate of taxon 'E_robo35'.
Note: taxon 'E_chry21' is a duplicate of taxon 'E_pachy26'.
Note: taxon 'E_albos27' is a duplicate of taxon 'E_minor28'.
Note: taxon 'C_anna46' is a duplicate of taxon 'C_anna47'.


This analysis requires at least 69 MB of memory space.

Score of initial tree: -6796.01

```

[illegible]

2.2.5 Results display

At the end of the data processing, three images will be shown, first the protein conservation plot, then the overall count of the motifs found in the dataset, and lastly, the phylogenetic tree. The icon  will show on the task bar, the user must click on it to display an image and close it to see the next one. Once all images are closed, the program ends.

STARTING RESULTS DISPLAY

It can take up to minutes for the plots to display, please wait.

(Image 1/3) Displaying protein conservation plot ...

(Image 2/3) Displaying motifs found in dataset ...

(Image 3/3) Displaying phylogenetic tree ...

RESULTS DISPLAY FINISHED

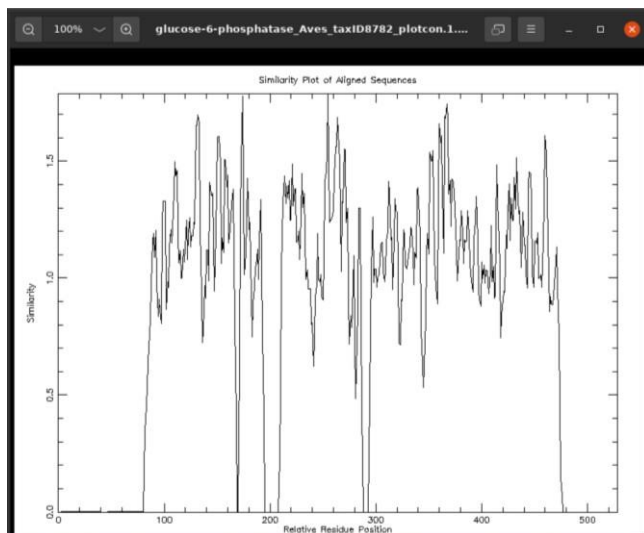
EXITING PROGRAM

2.2.6 Example outputs

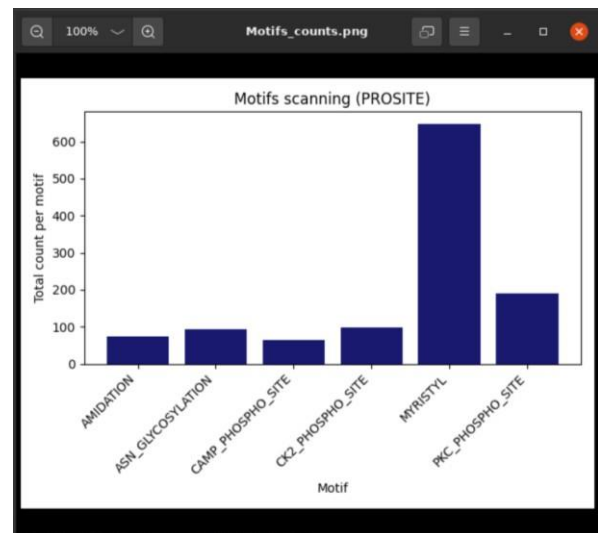
Besides the files mentioned in the messages on screen, a fasta file with all the sequences included in the analysis and a file with the protein sequences alignment in msf format were created. Contents of working directory:

```
Associated_motifs.txt
glucose-6-phosphatase_Aves_taxID8782.fasta
glucose-6-phosphatase_Aves_taxID8782_plotcon.1.png
glucose-6-phosphatase_Aves_taxID8782_plotcon.pdf
glucose-6-phosphatase_Aves_taxID8782_secondary_structure.garnier
glucose-6-phosphatase_Aves_taxID8782_Tree.pdf
glucose-6-phosphatase_Aves_taxID8782_Tree.png
Motifs_counts.csv
Motifs_counts_graphs
Motifs_counts.png
Patmatmotifs_result_files
Protein_alignment.msf
Protein_alignment.phy
Protein_alignment.phy_phyml_stats.txt
Protein_alignment.phy_phyml_tree.txt
```

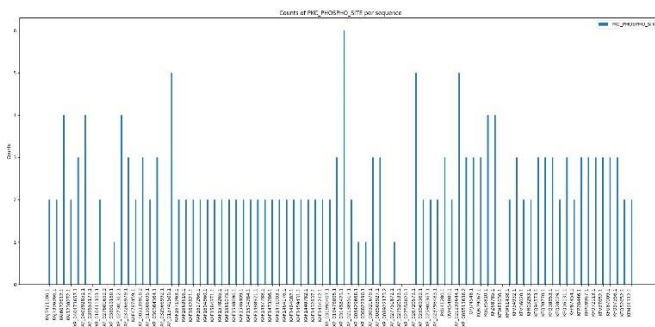
Example protein sequence conservation plot



Example overall motifs counts



Example motif counts per sequence



Example csv file with counts per motif

	AMIDATION	ASN_GLYCOSYLATION	CAMP_PHOSPHO_SITE	CK2_PHOSPHO_SITE	MYRISTYL	PKC_PHOSPHO_SITE	Accession	Species
Seq_1	1	1	1	2	8	2	KA17421106.1	Willisornis vidua
Seq_2	1	1	1	2	8	2	KA17396366.1	Pitangus sulphuratus
Seq_3	1	1	1	1	8	4	KA16072612.1	Aix galericulata
Seq_4	0	1	0	2	8	2	KA11230272.1	Lamprolaima superbus
Seq_5	1	1	1	2	8	3	XP_040473657.1	Falco naumanni
Seq_6	1	1	1	0	8	4	XP_040392851.1	Cygnus olor
Seq_7	0	1	0	2	9	0	XP_039553177.1	Passer montanus
Seq_8	1	2	1	3	7	2	XP_010411101.1	Corvus cornix cornix
Seq_9	1	2	1	2	8	0	XP_027601652.2	Pipra filicauda
Seq_10	1	1	0	2	8	1	XP_038019160.1	Motacilla alba alba
Seq_11	1	1	1	0	8	4	XP_027301312.1	Anas platyrhynchos
Seq_12	1	1	1	2	8	3	XP_037266979.1	Falco rusticolus
Seq_13	1	1	1	2	8	2	KAF4787459.1	Turdus rufiventris
Seq_14	1	1	0	1	8	3	XP_005139910.2	Melospittacus undulatus
Seq_15	0	2	0	2	8	2	XP_015506695.1	Parus major
Seq_16	1	2	0	1	7	3	XP_030364364.1	Strigops habroptila

Example patmatmotif file

```
# Sequence: KAF1404242.1      from: 1   to: 358
# HitCount: 14
#
# Full: No
# Prune: No
# Data_file: /usr/share/EMBOSS/data/PROSITE/pro
#
#=====

Length = 4
Start = position 96 of sequence
End = position 99 of sequence

Motif = ASN_GLYCOSYLATION

TDYYSNTSAPEIQQ
  |  |
 96 99

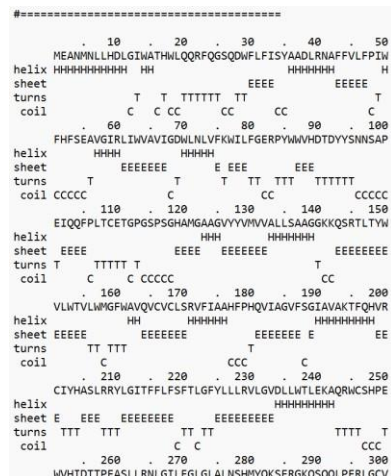
Length = 4
Start = position 141 of sequence
End = position 144 of sequence

Motif = CAMP_PHOSPHO_SITE

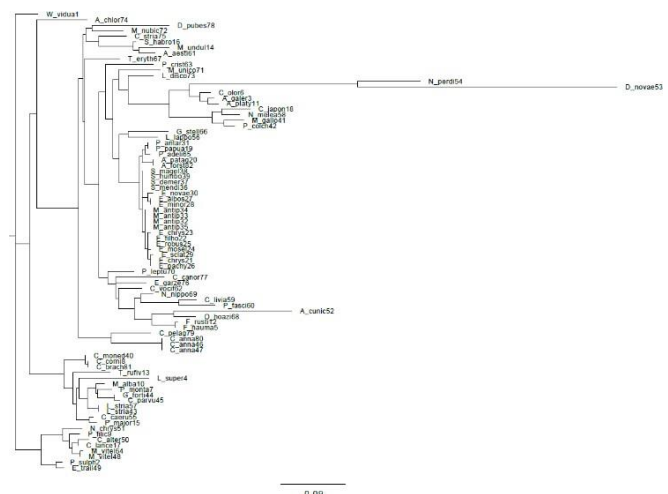
STAMGKKQSRTLKY
  |  |
141 144

Length = 3
Start = position 284 of sequence
End = position 286 of sequence
```


Example secondary structure prediction



Example phylogenetic tree



2.2.7 Interpreting outputs

Protein sequence conservation (Plotcon): Plotcon allows to visualize the quality along the sequence alignment, qualitatively assessing the conserved regions in the proteins from the dataset. The program utilizes a sliding-window method where the average similarity of residues within that window is plotted. Since the program is design to work on distinct datasets, the default parameter of window size 4 was used for all analysis. Only Plotcon graphs with same window size are comparable since the similarity score units are sensitive to this value (5). Higher values represent higher conservation between sequences.

Motifs scanning (Patmatmotifs): The motifs found from the PROSITE database were summarized in a csv file with all the counts per sequence and individual graphs with counts were created. The user can access the .patmatmotifs files from each sequence (or the master file) to gain insight on the location, length and aa from a specific motif. The motifs can give insight into unknown protein function. There are tools available online to visualize the location of the motifs in each sequence.

Secondary structure prediction (Garnier): The Garnier application has a maximum accuracy of 65%, while other software can reach 80% (2). For secondary structure prediction, a consensus should be made by employing different programs.

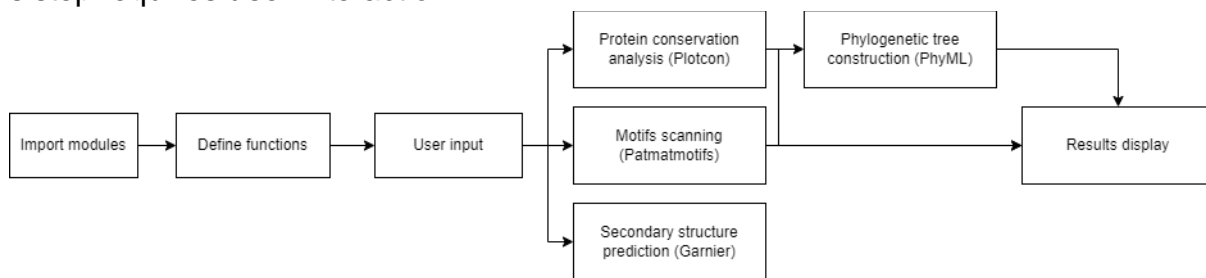
Phylogenetic tree(PhyML): The tree statistics file contains information on the analysis setting and the likelihood (log-likelihood) of the model constructed for the dataset according to the maximum likelihood phylogenetic model (6). Since the sequences included in the dataset were not filtered to eliminate duplicated species, the phylogenetic tree may include nodes with multiple sequences from the same species. However, the statistics file includes the actual number of taxa analysed. Higher log-likelihoods indicate trees that make the data most probable. The log-likelihoods are comparable for trees constructed from the same alignment but not from others. Resources can be found online on how to interpret phylogenetic trees.

3. Maintenance manual

3.1 Code structure

The code written for a Python3 interpreter contains the following elements in the given order:

- Modules: Imports all the modules required for the program.
- Functions: Defines all the functions to be used. Aimed at input validation and data retrieval.
- User input: The program starts with introduction for user and asks for all the input required to perform all analysis. This section calls all the functions previously defined.
- Analysis execution: The program executes one analysis at the time. Shows preliminary results on the terminal.
- Results display: After analysis execution, the program displays resulting images on screen. This step requires user interaction.



3.2 Functions

3.2.1 Input validation

Functions were defined for input validation in order to restrict user to give suitable answers when possible. These functions were integrated in a final function that would ask for all the input and return the values needed for data retrieval.

Working directory

To determine the directory where the program would store all output files, error-traps were created so the user could not enter a directory name that was either the current working directory or a parent directory since the program would delete it and create the a new one with the output files.

Taxon and taxon ID

For the taxon name, regular expressions were used to limit the type of input. Accepted input: Single words with or without a hyphen, or single word ending with a dot if more words are present. Words must be separated by one space. For the taxon ID, only non negative integers are accepted.

Protein family

Regular expressions were also used to filter protein family name. Only alphanumeric characters, spaces and hyphens are allowed.

User confirmation

At several points, the program asks the user to confirm their choices and the dataset for the analysis (chosen species, number of sequences, number of species and motif scanning parameter). This is done by asking to enter y or n or selecting from given numbered options. A couple of functions were defined for this purpose. *While loops* were used to retrieve an answer from the given options. *If statements* were used for y/n answers.

3.2.2 Data retrieval

The input from the user (taxon name/ID and protein family name) was used to query NCBI databases using EDirect (Entrez Direct). Extensive documentation on the use of Entrez and EDirect can be found at <https://www.ncbi.nlm.nih.gov/books/NBK25499/#chapter4.Introduction> and <https://www.ncbi.nlm.nih.gov/books/NBK179288/>. The taxonomy database was used to query the user input for taxon name or ID, and the protein database to query the chosen taxon and protein family and retrieve the protein sequences. *While loops* were used to ask input in case no results were found in the query (due to poor typing or no results for the specified taxon and protein family). EDirect queries can fail due to problems with the NCBI server, so a try/except scheme was placed to retry the queries until the results are found. The try/except scheme was used to retrieve a valid taxon name and ID, number of protein sequences in the query, protein sequences for the dataset, and accession number and species of the protein sequences. If the number of hits of the given taxon name/ID and protein family is larger than 1,000 sequences, the program will restrict the query to the first 1,000 sequences in the query if the user decides to proceed with the analysis. This parameter is taken into account for the retrieval of accession numbers and species as well.

3.2 Programs

3.2.1 EMBOSS Plotcon

The program does a subprocess call to run the protein sequence alignment of the dataset using clustalo. The output alignment is saved in tree order for later use when creating the phylogenetic tree. Once the alignment is done, the program does a subprocess call to run plotcon. The default window size of 4 was used, but the next version of the program will give the option to the user to specify the value, keeping 4 as the default. Once all analysis are done, the program will display on screen the conservation plot generated on this step.

3.2.2 EMBOSS Patmatmotifs

The patmatmotif program requires individual fasta files to look for motifs present in the given protein sequence. Given that the EDirect query produces a fasta file that contains all the sequences, an iteration over this file was used to create temporary fasta files with only one sequence. Patmatmotif was run on this fasta file and a patmatmotif file with the accession number of the protein sequence was generated. Inside this iteration, the program opens the patmatmotf file and uses findall from the re module to look for all the motifs present in this sequence. This information is stored in a dictionary (total_motifs) with sequences number as keys and the motifs as values. The contents from all the individual patmatmotifs files are also stored in a txt file. Next versions of the program will take the starting and ending positions of

the motifs and plot them along the protein sequence. A set of the motifs in the query is defined and the dictionary created in the previous step (`total_motifs`) is used to create a data frame to store all results. Accession numbers and species are added for easy identification of sequences of interest (final data frame used for the csv output file). Using `matplotlib.pyplot`, graphs with overall counts and specific motifs counts are generated. The overall counts graph is displayed at the end of the program.

3.2.3 EMBOSS Garnier

The program does a subprocess call to run the garnier program using the fasta file with all the protein sequences in the dataset.

3.2.4 PhyML

Given the computational time of phylogenetic tree construction, the program will restrict this analysis to 100 sequences. If the dataset contains more, the order from the Clustal Omega alignment generated for the sequence conservation analysis is used to generate a new fasta file with the first 100 sequences. This fasta file is now used to generate a new alignment using Clustal Omega but the output is required in phy format. The new alignment is used to construct the phylogenetic tree. The character length for the species name in the tree is limited. This is why a partial species name and the number of the sequence in the dataset is used to identify the species in the resulting tree. Different rules were created trying to maximize the length of the partial name according to the species name. Once the PhyML output files are generated, the program does a subprocess call to run the figtree program to create a visual representation of the tree. This image is displayed at the end of the program.

4. References

1. Xiong J. Protein motifs and domain prediction (Chapter 7) - essential bioinformatics [Internet]. Cambridge University Press; [cited 2023 Nov 20]. Available from: <https://www.cambridge.org/core/books/essential-bioinformatics/protein-motifs-and-domain-prediction/E17046CB1CD04184A828D8BAC2D222AF>
2. EMBOSS. Garnier [Internet]. [cited 2023 Nov 20]. Available from: <https://emboss.sourceforge.net/apps/release/6.6/emboss/apps/garnier.html>
3. Palaniappan A, Jakobsson E. Fourier analysis of conservation patterns in protein secondary structure. *Computational and Structural Biotechnology Journal*. 2017; 15: 265–70. doi:10.1016/j.csbj.2017.02.002
4. Atas H, Tuncbag N, Doğan T. Phylogenetic and other conservation-based approaches to predict protein functional sites. *Methods in Molecular Biology*. 2018; 1762: 51–69. doi:10.1007/978-1-4939-7756-7_4
5. EMBOSS. Plotcon [Internet]. [cited 2023 Nov 20]. Available from: <https://emboss.sourceforge.net/apps/release/6.6/emboss/apps/plotcon.html>
6. Guindon S, Dufayard J-F, Lefort V, Anisimova M, Hordijk W, Gascuel O. New algorithms and methods to estimate maximum-likelihood phylogenies: Assessing the performance of phyml 3.0. *Systematic Biology*. 2010;59(3):307–21. doi:10.1093/sysbio/syq010