Biological Data Project

Single protein domain study

Ling Xuan, Chen
lingxuan.chen@studenti.unipd.it

Reddavide, Matteo
matteo.reddavide@studenti.unipd.it

February 2022

1 Introduction

The goal of this project is to study a single protein domain and create a sequence model in order to provide a functional characterization of the family. Our domain is from *Cyberlindnera mrakii (Yeast) (Williopsis mrakii)* and its Pfam name is *Nitronate monooxygenase* (Pfam ID: PF03060).

The code and all the files used in this project are accessible on GitHub[1].

2 Model creation

In this section we report all the passages used to create our sequence models.

2.1 Blast search

First of all the input sequence is used for a BLAST search against two databases, for this process online BLAST services[2] were used. The selected databases are UniRef50 and UniRef90. Both are used with thresholds for the number of hits, maximum 250 hits, and E-values, lower than 0.01.

2.2 Multiple sequence alignment

Next step was to perform a multiple sequence alignment (MSA) of the previous results, to do so we used three different algorithms from the EMBL-EBI web services:

- T-Coffee[3]
- Clustal Omega[4]
- MUSCLE[5]

In this case the parameters used are the default ones. After the alignment was done, they were all polished using JalView in order to remove redundancies (set to 95) and empty columns, in addition we also removed columns with low occupancy (i.e. lower than 5).

2.3 Models

For each one of the result a Position-Specific Scoring Matrix, *PSSM*, and a Hidden Markov Model, *HMM*, are used to build the models. The process is done using the commands described in the file 'build_pssm_and_hmm' from the command line, the results are stored in 'data/predictions/'.

[1] Accesible at: `https://github.com/MattRosso/Biological-Data/`

[2] Accesible at: `https://www.uniprot.org/blast/`

[3] Accesible at: `https://www.ebi.ac.uk/Tools/msa/tcoffee/`

[4] Accesible at: `https://www.ebi.ac.uk/Tools/msa/clustalo/`

[5] Accesible at: `https://www.ebi.ac.uk/Tools/msa/muscle/`

3 Model evaluation

Ground truth In order to evaluate our results a ground truth is required. We tried with two different methods: firstly we used the InterPro API[6] from which we obtained 26 hits, then we did a UniProt query[7] to retrieve all the proteins, in this case the number of hits was 24. All the 24 proteins found in Uniprot are in the 26 found by the InterPro API. The two extra protein are O34787 and Q2T4N0 which, according to InterPro, does not have the domain, so we ultimately discarded them, basically ending up to use the UniProt query results as our ground truth.
From the InterPro API we also retrieved and then integrated to our ground truth, for each protein, the start and the end position of the domain.

HMM-SEARCH and PSI-BLAST Each of our model is then used to generate predictions about the presence or the absence of the domain inside the model.
In the case of HMM models a HMM-SEARCH is performed against SwissProt using web services from EMBL-EBI[8]. For the PSSMs we used the NCBI web service[9] in order to perform the PSI-BLAST of our models.

3.1 Evaluation

The evaluation of the models is done using as metrics precision, recall, F-score and Matthews Correlation Coefficient. It is also performed in two different steps:

Protein level In this case the evaluation is done on the ability of each model in the retrival of the same proteins.

	Precision	Recall	F-score	MCC
hmmer_tcoffee_uniref50	0.017991	1.00	0.035346	0.133976
psiblast_tcoffee_uniref50	0.046875	0.75	0.088235	0.187398
hmmer_clustalo_uniref50	0.013833	1.00	0.027288	0.117436
psiblast_clustalo_uniref50	0.036000	0.75	0.068702	0.164199
hmmer_muscle_uniref50	0.013165	1.00	0.025988	0.114557
psiblast_muscle_uniref50	0.049315	0.75	0.092545	0.192219
hmmer_tcoffee_uniref90	0.019934	1.00	0.039088	0.141039
psiblast_tcoffee_uniref90	0.082569	0.75	0.148760	0.248776
hmmer_clustalo_uniref90	0.024291	1.00	0.047431	0.155725
psiblast_clustalo_uniref90	0.094241	0.75	0.167442	0.265789
hmmer_muscle_uniref90	0.019884	1.00	0.038993	0.140864
psiblast_muscle_uniref90	0.225000	0.75	0.346154	0.410752
hmmer_muscle_uniref50_threshold	1.000000	1.00	1.000000	1.000000
hmmer_tcoffee_uniref50_threshold	1.000000	1.00	1.000000	1.000000

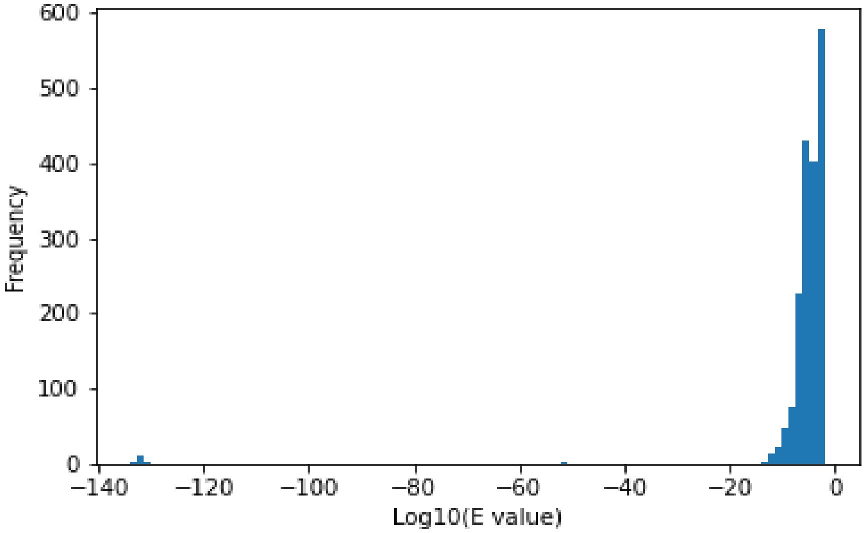
As we can see for each model the precision is low, but the recall is quite high (for HMMSEARCH predictions is always one) implying that even if the number of proteins predicted as having the domain is way higher than the real one, all of them are found by the HMM models.
The last two rows of the table shows that setting the "right" threshold can make some of ours models' precision and recall equal to 1. In fact, (for example in the muscle model) by putting a threshold of an E value smaller than x, where x is a number between 10^{-30} and 10^{-15}, leads to a perfect result. This is particular evident in the histogram below which shows that the majority of retrieved protein have low E value.

[6] https://www.ebi.ac.uk/interpro/api/protein/reviewed/entry/pfam/pf03060?format=json

[7] https://www.uniprot.org with the query: 'family:"nitronate monooxygenase family" AND reviewed:yes'

[8] Accesible at: https://www.ebi.ac.uk/Tools/hmmer/search/hmmsearch

[9] https://blast.ncbi.nlm.nih.gov/Blast.cgi?PROGRAM=blastp&PAGE_TYPE=BlastSearch&LINK_LOC=blasthome



However deciding a classification parameter using the same data that is used to evaluate it would lead to overfitting, so we decided to take in account as *family_sequences* all the protein retrieved by the model *hmmer_muscle_uniref50_tree* that have an e-value smaller or equal that 10^{-9}, preferring to sacrifice some precision to have the possibility of better recall in different databases. In this way the total number of retrieved protein in 95.

Residue level In this second step of our evaluation the focus is on the ability of each model in the prediction of the domain position inside the protein. This is done comparing the position of the model's domain with the ground truth.

	Precision	Recall	F-score	MCC
hmmer_muscle_uniref50	0.978769	0.973057	0.975904	0.635004
hmmer_tcoffee_uniref50	0.974454	0.975664	0.975059	0.601485
hmmer_clustalo_uniref90	0.978843	0.970822	0.974816	0.625390
hmmer_clustalo_uniref50	0.978302	0.968463	0.973357	0.609290
psiblast_muscle_uniref50	0.971734	0.967044	0.969383	0.582193
hmmer_tcoffee_uniref90	0.979286	0.956792	0.967908	0.573702
hmmer_muscle_uniref90	0.976550	0.956543	0.966443	0.544940
psiblast_tcoffee_uniref90	0.968201	0.962883	0.965535	0.531673
psiblast_clustalo_uniref50	0.963111	0.964714	0.963912	0.485977
psiblast_tcoffee_uniref50	0.951291	0.968708	0.959921	0.363066
psiblast_muscle_uniref90	0.945446	0.969208	0.957179	0.285971
psiblast_clustalo_uniref90	0.948479	0.965213	0.956773	0.315304

From the results, sorted by the F-score which is the harmonic mean between precision and recall, again we can conclude that the best predictions are generated by HMM models and in particular by hmmer_muscle_uniref50.

In order to proceed for the other tasks, we decided to download the Swissprot database [10]. We parsed the file using *Biopython* and created a dictionary that has as keys the accession ids and as values the name, the description, the list of GOs and the taxonomy id.

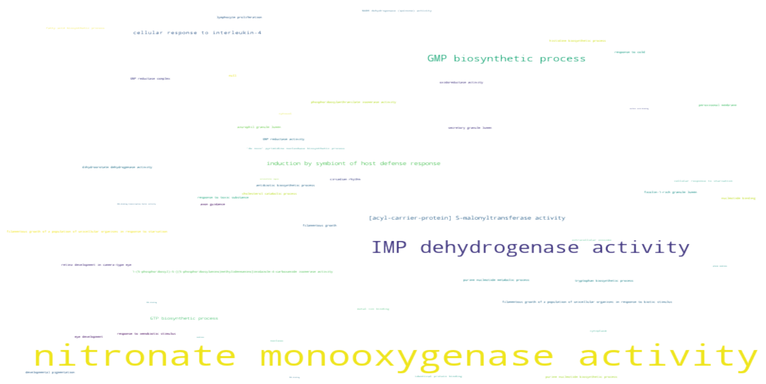
[10] Accesible at :`ftp://ftp.uniprot.org/pub/databases/uniprot/current_release/knowledgebase/complete/uniprot_sprot.xml.gz`

4 Taxonomy

To build the tree in a way that nodes that are on the same rank are plotted at the same height, we preferred to use the API provided by *ebi.ac.uk* to get the full lineage with their associated rank. We plotted only those with {*Hidden: False*} which have also all known rank. The ranks that are used are in order: superkingdom, kingdom, subkingdom, phylum, subphylum, class, order, family, genus and species group.
The node size of the tree is proportional to its abbundance. The tree is plotted using the *plotly* library which in turn uses the Reingold-Tilford tree layout present in the *igraph* library. Given its size, the image is present in the GitHub repository.[11]

5 Function

From GO taken from the SwissProt XML file we studied the characterization of the family sequence dataset. First we created a dictionary of the GO terms, having as key the GO and as value the list of all the proteins that possess that GO. This process was also done using the *go.obo* database, in this way, we could integrate the annotations with their ancestors.
Then for each GO present in our *family_sequences*, we built the confusion matrix where the proteins selected by our model and the remaining proteins of SwissProt were also separated by having or not having the GO term. From these tables, we calculated the fold increase and performed a Fisher test, useful to derive the left, right and two tail p-values.
The most enriched terms are then plotted in a word cloud, and as we expected the most relevant is *nitronate monooxygenase activity*.



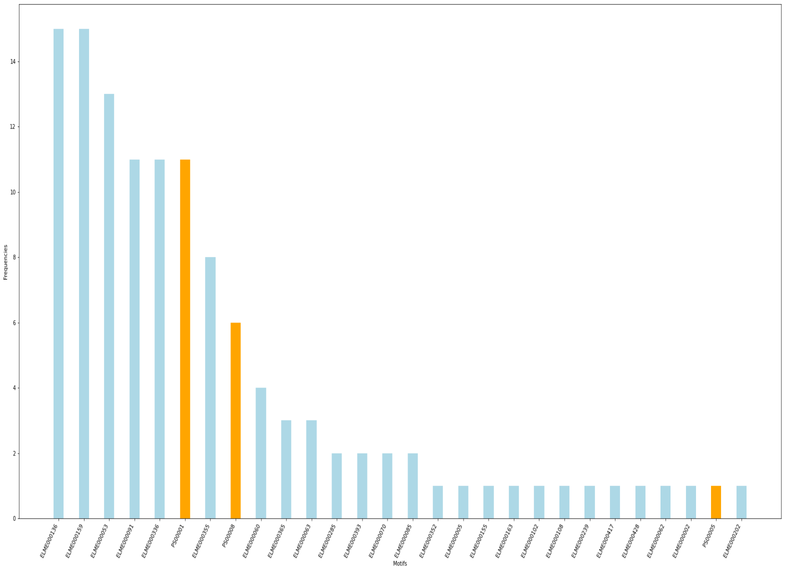
In order to find the most enriched branches, we considered a subset of the previously created tables. We decided to define high level terms those with the depth smaller than 5, the depth intended as the length of the shortest path to the root. Same as above, we calculated the fold increase, sorted them by the p-values. We report the top 10 in the table below.

GO ID	Right-tail	Fold Increase	Namespace	Description	Depth
GO:0016491	1.025126e-77	112.199739	molecular function	oxidoreductase activity	2.0
GO:0009636	1.424325e-29	139.125770	biological process	response to toxic substance	3.0
GO:0000166	3.920993e-28	16.764467	molecular function	nucleotide binding	3.0
GO:0046651	1.141122e-13	175.470369	biological process	lymphocyte proliferation	4.0
GO:0046872	2.192605e-11	5.750041	molecular function	metal ion binding	4.0
GO:0007623	1.221073e-08	49.757628	biological process	circadian rhythm	2.0
GO:0060041	1.243121e-07	123.317199	biological process	retina development in camera-type eye	3.0
GO:0017000	2.527213e-04	34.606491	biological process	antibiotic biosynthetic process	3.0
GO:0036170	9.524927e-03	211.189623	biological process	filamentous growth of a popul...	4.0
GO:0036180	1.002380e-02	200.629078	biological process	filamentous growth of a popul...	3.0

[11] https://github.com/MattRosso/Biological-Data/blob/main/taxonomic_tree.png

6 Linear motifs

Given our *family_sequences*, we firstly compared it with the MobiDB-Lite database[12], which led us to have only 17 out of the 95 protein having a disordered region.
We then downloaded the linear motif from ELM website[13] and ProSite patterns[14], for the Prosite we have taken in account only the patterns with "PA" lines. The Prosite patterns were not in Regex, so in order to parse it, we converted in Regex using the code found in an old *Biopython* distribution[15] Below we show the frequencies of the linear motif found inside our *family_sequences*. Different color used to indicate the two different databases.



[12] https://drive.google.com/file/d/1m7rdFvQiCRizOx54YPk1eMw4qF1iskbz/view
[13] http://elm.eu.org/elms
[14] https://ftp.expasy.org/databases/prosite/prosite.dat
[15] https://home.cc.umanitoba.ca/~psgendb/doc/local/biopython-1.55.old/Bio/Prosite/Pattern.py