Abc4pwm: Affinity Based Clustering for Position Weight Matrices

Documentation

MARCH 16, 2023

OMER ALI
Omerali.0191@gmail.com

Table of Contents

Abc4pwm: Affinity Based Clustering for Position Weight Matrices

Introduction:

A new tool for the clustering of Transcription Factors is designed that work on RNAseq, ChIPseq, Human Transfaction factors, and other biological datasets to give easy interpretation for biologists. It is python package designed in Python 3 and the dependencies are given in 'requirements.txt' file.

Abstract

Background

Transcription factor (TF) binding motifs are identified by high throughput sequencing technologies as means to capture Protein-DNA interactions. These motifs are often represented by consensus sequences in form of position weight matrices (PWMs). With ever-increasing pool of TF binding motifs from multiple sources, redundancy issues are difficult to avoid, especially when every source maintains its own database for collection. One solution can be to cluster biologically relevant or similar PWMs, whether coming from experimental detection or in silico predictions. However, there is a lack of efficient tools to cluster PWMs. Assessing quality of PWM clusters is yet another challenge. Therefore, new methods and tools are required to efficiently cluster PWMs and assess quality of clusters.

Results

A new Python package Affinity Based Clustering for Position Weight Matrices (abc4pwm) was developed. It efficiently clustered PWMs from multiple sources with or without using DNA-Binding Domain (DBD) information, generated a representative motif for each cluster, evaluated the clustering quality automatically, and filtered out incorrectly clustered PWMs. Additionally, it was able to update human DBD family database automatically, classified known human TF PWMs to the respective DBD family, and performed TF motif searching and motif discovery by a new ensemble learning approach.

Conclusion

This work demonstrates applications of abc4pwm in the DNA sequence analysis for various high throughput sequencing data using ~ 1770 human TF PWMs. It recovered known TF motifs at gene promoters based on gene expression profiles (RNA-seq) and identified true TF binding targets for motifs predicted from ChIP-seq experiments. Abc4pwm is a useful tool for TF motif searching, clustering, quality assessment and integration in multiple types of sequence data analysis including RNA-seq, ChIP-seq and ATAC-seq.

Publication:

This work is published and can be found here.

Installation:

1. Download the package using following command:

```
wget https://github.com/Omer0191/abc4pwm/archive/refs/heads/master.zip
```

2. Change to home directory of the downloaded file.
    a. It is highly recommended to create a separate virtual environment for the package so that any libraries compatibility can be avoided. This can be done using following commands:
    b. Create virtual environment and activate it. (We use miniconda here as an example)

```
conda create -venv env_abc4pwm
conda activate env_abc4pwm
```

    c. Python 3 is required for this package. Install dependencies. This can be done by using the *requirements.txt* file provided with the package.

```
pip install -r requirements.txt
```

    d. Now all dependencies should have been installed if the above command run smoothly. Package can be install using the following command:

```
python setup.py install
```

Package should be installed and running by now. An environment file abc4pwm_environment.yml file is also provided if some user wants to directly import the virtual environment.

Usage:

The following command can be used for the tasks. Help menu with every task and parameters can be seen by typing -h, --help after any command:

abc4pwm [-h] [task]
Tasks: cleandatabase_for_classification, classification, clustering, representative_motif, quality_assessment, visualize, plot_cluster_motifs, text_tfdb, searching, conversion, ensemble_learning, ensemble_investigate

Pipeline to produce all results:

You should run the pipeline in the following order to generate all necessary files:
1- **'cleandatabase_for_classification':** In the first step you should generate clean database to classify inputs files.
2- **'classification':** In this step classify files according to their DBD by adding labels.
3- **'clustering ':** In the step apply clustering on output folder produced by step 2.

4- **'quality_assessment':** This step calculates quality of clusters and remove bad quality pwms from clusters.
5- **'representative_motif':** This step prepares a representative motif of the cluster.
6- **'plot_cluster_motifs':** This will plot the motifs inside cluster in along with the names in a pdf file. Also generate a text report.

Modules:

This package provides following functionalities for sequence analysis.

1. cleandatabase_for_classification
2. classification
3. clustering
4. representative_motif
5. quality_assessment
6. visualize
7. plot_cluster_motifs
8. text_tfdb
9. conversion
10. ensemble_learning
11. ensemble_investigate
12. conversion
13. searching

1. Database For Classification:

This module creates a uniform naming database of the input TFs along with their respective DNA Binding Domain information from various other databases.

usage: abc4pwm cleandatabase_for_classification [-h]
[--pwm_files_directory FOLDER]
[--read_new NUMBER]

Arguments:

Parameter	Type	help
--pwm_files_directory	FOLDER	This folder should contain the input pwm files in .mlp format

optional arguments:

Parameter	Type	help

--read_new	NUMBER	Select 1 if you want a new updated read from sources(internet Connection Required). Default 0
-h, --help	string	show help message and exit

2. Classification

This module assigns DNA binding domain information to the input motifs.

usage: abc4pwm classification [-h] [--pwm_files_directory FOLDER]
[--output_directory FOLDER]
[--original_pwm_files_directory FOLDER]
[--load_new_db NUMBER]

Arguments:

Parameter	Type	help
--pwm_files_directory	FOLDER	This folder should contain the input pwm files in . mlp format
--output_directory	FOLDER	This folder should point to folder where files should go after getting labels of DBD
--original_pwm_files_directory	FOLDER	This folder should contain a copy of input pwm files in . mlp format

optional arguments:

Parameter	Type	help
--load_new_db	NUMBER	This should be 1 if you want to download a new update db from sources 0 by default
-h, --help	string	show help message and exit

3. Clustering

This module clusters the input motifs given in PWM format. This module provides functionality to perform clustering on motifs divided into their respective DBDs as well as motifs without any DBD information. It uses Affinity Propagation Clustering and do not need any seed cluster number.

usage: abc4pwm clustering [-h] [--dbd_folders_directory FOLDER]
[--output_directory FOLDER]
[--in_dbd NUMBER]
[--minimum_pwms_in_dbd NUMBER]
[--max_processors NUMBER]
[--seed Number]

[--damp Number]
[--max_iter Number]
[--convergence_iter Number] [--preference Number]

Arguments:

Parameter	Type	help
--dbd_folders_directory	FOLDER	This folder should contain DBD folders. Output of classifcation_pwm should be this task input.
--output_directory	FOLDER	This folder should point to folder.
--path_to_txt	FOLDER	This path will have the summary out file.

optional arguments:

Parameter	Type	help
--in_dbd	NUMBER	This should be 0 if you want to cluster all together (non DBD) 1 by default
--minimum_pwms_in_dbd	NUMBER	minimum number of pwms in a dbd to be clusteredDefault value is 5
--path_to_txt	FOLDER	This path will have the summary out file.
--max_processors	NUMBER	maximum number of processors for paralle processingDefault value is 5
--seed	Number	Seed for random selection of cluster center. Input 1 to fix. Default Seed is 0.
--damp	Number	Damping factor (between 0.5 and 1) is the extent to which the current value is maintained relative to incoming values (weighted 1 - damping). This in order to avoid numerical oscillations when updating these values (messages).
--max_iter	Number	Maximum number of iterations.
--convergence_iter	Number	Number of iterations with no change in the number of estimated clusters that stops the convergence.
--preference	Number	Preferences for each point - points with larger values of preferences are more likely to be chosen as exemplars. The number of exemplars, ie of clusters, is influenced by the input preferences value. If the preferences are not passed as arguments, they will be set to the median of the input similarities.

4. Quality Assessment

Quality assessment module provides a combination of several statistical metrics for the quality assessment of clustered PWMs.

usage: abc4pwm quality_assessment [-h] [--dbd_folders_directory FOLDER]
[--output_directory FOLDER]
[--dbd_for_plotting FOLDER]
[--load_new_assesment <class 'bool'>]
[--mean_threshold NUMBER]
[--z_score_threshold NUMBER]
[--top_occurrences NUMBER]
[--occurrences_threshold NUMBER]

Arguments:

Parameter	Type	help
--dbd_folders_directory	FOLDER	This folder should contain clustered DBD folders. Output of clusterings should be this task input
--out_path_for_qa_clusters	FOLDER	This folder should point to output folderwhere quality assessed clusters will be stored.
--output_folder_for_text_report	FOLDER	Specify a folder where report in txt file will be stored.

optional arguments:

Parameter	Type	help
--output_path_for_quality_assessment_file	FOLDER	folder where quality assessment .json file will be stored.Default, data/in/
--load_new_assesment	NUMBER	1 if you want to do new assesment, 0 if you want load existing assesment
--mean_threshold	NUMBER	mean threshold for uncertain clustersDefault value is 0.80

--z_score_threshold	NUMBER	max negative threshold of zscore for similarity values of pwmDefault value is -1.0
--top_occurrences	NUMBER	This value corresponds to occurrence of a pwm less than a threshold z-scoreValue is between 0 to 1. Default value is 0.15
--occurrences_threshold	NUMBER	This value corresponds to threshold of occurrence from top occurrencesValue is between 0 to 1. Default value is 0.05

5. Representative Motif:

This module will create a representative motif of any given multiple motifs list. Here it creates representative motif of the clusters produced.

usage: abc4pwm representative_motif [-h]
[--path_to_clusters FOLDER]
[--dbd string]
[--clusters string]
[--ic NUMBER]
[--best_match_initial_motif NUMBER]
[--mean_threshold NUMBER]
[--z_score_threshold NUMBER]
[--top_occurrences NUMBER]
[--occurrences_threshold NUMBER]

Arguments:

--path_to_clusters	FOLDER	This folder should contain the clusters folders
--clusters	string	This argument should be cluster numbers as string For example, 0,1,2,3,4 if you want to plot these clusterswrite 'all' if you want to make representative for all clusters

optional arguments:
-h, --help show this help message and exit

--dbd	string	Default value is 'selected'. Representative of clusters pathmentioned in --path_to_clusters parameter will be calculated. Write 'all' if representative calculation of all dbd and all clusters is required.
--best_match_initial_motif	NUMBER	This should be 0 if you want initial motif to be random 1 by default.
--mean_threshold	NUMBER	mean threshold for uncertain clustersDefault value is 0.80
--z_score_threshold	NUMBER	max negative threshold of zscore for similarity values of pwmDefault value is -1.0
--top_occurrences	NUMBER	This value corresponds to occurrence of a pwm less than a threshold z-scoreValue is between 0 to 1. Default value is 0.15
--occurrences_threshold	NUMBER	This value corresponds to threshold of occurrence from top occurrencesValue is between 0 to 1. Default value is 0.05
--ic	NUMBER	Information Content for trimming edges.Default value is 0.4

6. Plotting Motifs of Clusters:

Module to plot list of PWMs (motifs) given in a folder. run representative motif task before running this task. It will produce a pdf with all motifs and their representative motif at the top. All relevant information is also stored in a text file for further uses and analysis.

usage: abc4pwm plot_cluster_motifs [-h]
[--path_to_clusters FOLDER]
[--output_folder FOLDER]
[--clusters string]
[--dbd string]

Arguments:

Parameter	Type	help
--path_to_clusters	FOLDER	This folder should contain clusters of a DBD, Write all if you want to plot all dbds.
--output_folder	FOLDER	This folder should contain plots of clusters in pdf

-clusters	string	This arguement should be cluster numbers as string For example, 0,1,2,3,4 if you want to plot these clusters. write 'all' if you want to plot all clusters

Optional Arguments:

--dbd	string	Default value is 'selected'. Clusters pathmentioned in --path_to_clusters parameter will be printed Write 'all' if plotting of all dbd and all clusters is required.

7. Visualize

Module for the visualization.

usage:

abc4pwm visualize [-h]
[--path_to_folder_of_assessment_file FOLDER]
[--path_to_folder_of_DBDs FOLDER]
[--output_folder FOLDER]
[--dbd_for_plot FOLDER] [--task string]

Arguments:

--path_to_folder_of_assessment_file	FOLDER	folder path from where quality assessment file should be taken
--path_to_folder_of_DBDs	FOLDER	this folder should contain clustered DBD folders
--output_folder	FOLDER	folder where visualization output should be saved
--dbd_for_plot	FOLDER	path of dbd which is needed to be visualized. Write 'all' if you want to plot for all dbds.

optional arguments:

--task	string	Specify visualization task. For example, boxplot, pichart, etc Default is boxplot.

8. Transcription Factor Database:

This module creates a database of TFs list with their respective DBD information gathered from various sources and stores in a text file for further uses in analysis.

usage:

```
abc4pwm text_tfdb [-h]   [--pwm_files_directory FOLDER]
                         [--output_directory FOLDER]
```

Arguments:

--pwm_files_directory	FOLDER	This folder should contain the input pwm files in . mlp format
--output_directory	FOLDER	This folder should to output folder. Boxplot will go to this folder

9. Searching

This module can be used to search a given motif against a database of motifs and produce a pdf of top n matched motifs along with similarity score.

usage:

```
abc4pwm searching [-h]
                  [--pwm file]
                  [--db_path path or list]
                  [--output_directory FOLDER]
                  [--db_type string]
                  [--db_format string]
                  [--top_n NUMBER]
                  [--tf_name string]
                  [--input_count NUMBER]
                  [--db_count NUMBER]
                  [--db_file_type String]
                  [--input_file_type String]
                  [--input_prob Number]
                  [--db_prob Number]
```

Arguments:

--pwm	file	position weight matrix file (motif) which you want to search. .mlp format
--db_path	path or list	path to clustered dbds according to the hierarchy of abc4pwm.if db_type=list then then this paramter should be list of pwms, against whcih you are searching the pwm

--output_directory	FOLDER	This folder should to output folder. search_result.pdf output file will be stored here.

optional arguments:

--tf_name	string	If you want to search specific tf in a folder then use this parameter
--db_type	string	If database for comparison is folder hierarchy like abs4pwm then this willbe db_type=path. Write list if providing list of pwms for comparison
--db_format	string	If database for comparison have format, please mention. Supported formats are abc4pwm, Tranfac, Jaspar. default is abc4pwm
--top_n	NUMBER	Number of top matches from the database. Default 5
--input_count	NUMBER	1 if input file contains values in counts
--db_count	NUMBER	1 if database file contains values in counts
--db_file_type	String	mention the extension of file type. e.g., .mlp, .txt
--input_file_type	String	mention the extension of file type. e.g., .mlp, .txt
--input_prob	Number	1 if input file contains values in probabilities
--db_prob	Number	1 if databas file contains values in probabilities

10. Conversion

This module is used for the conversion of input formats. We provide service to convert 'transfac' and 'jaspar' formats to abc4pwm input format.

usage:

abc4pwm conversion [-h]
[--pwm_files_directory FOLDER]
[--in2out string]
[--output_folder FOLDER]

optional arguments:

--pwm_files_directory	FOLDER	This folder should contain the input pwm files in . mlp format which you want to convert
--in2out	string	Specify conversion like the following. 'abc4pwm2transfac' 'transfac2abc4pwm' 'abc4pwm2jaspar' 'jaspar2abc4pwm'
--output_folder	FOLDER	folder where converted files should be saved

11. Ensemble Learning

This module performs TF prediction from protein DNA interaction experiments.
usage:
abc4pwm ensemble_learning [-h] [--opt_dependence Number]
[--numP Number] [--opt_numOfWeakReads Number]
[--number_of_genes Number]
[--expFile File path]
[--opt_weak_expFile File path]
[--opt_seqFile File path]
[--opt_out File path.] [--opt_loops Number]
[--opt_min_L Number] [--opt_max_L Number]
[--opt_iteration Number]
[--opt_p_value float] [--opt_strand Number]
[--opt_normalization Number]
[--max_processors Number]

Arguments:

--expFile File	path	Strong Expression file path
--opt_seqFile	path	Strong sequence file path.

optional arguments:

--opt_dependence	Number	Define dependence, Default is 0.
--numP	Number	number of times random selections will be done. Default is 15
--opt_numOfWeakReads	Number	number of weak read if any. Default 0.
--number_of_genes	Number	number of genes to be selected from input. Default is 200
--opt_weak_expFile	File path	Weak expression file path.
--opt_out	File path.	output folder path.
--opt_loops	Number	Number of loops to repeat calculations. Default is 3.
--opt_min_L	Number	Minimum length for predicted pwm (motif), Default is 9.
--opt_max_L	Number	Maximum length for predicted pwm (motif). Default is 9.
--opt_iteration	Number	Number of iterations. Default is 500.
--opt_p_value	float	p value. Default is 0.0001
--opt_strand	Number	Strand. Default is 0
--opt_normalization	Number	Normalization value. Default is 2.

--max_processors	Number	Define maximum number of processors for parallel computing. Default is mp.cpu_count()
--seed Number	Number	Seed for random selection. Default seed is 0(random).

12. Ensemble Investigate:

This modules compares the results and investigates the quality of predictions.

usage:

abc4pwm ensemble_investigate [-h]
[--path_to_predicted_files . mlp file]
[--db_folder path or list]
[--output_folder FOLDER]
[--tf_name string]
[--db_type string]
[--top_n Number]
[--dst_for_bad_pwms path]
[--mean_threshold NUMBER]
[--z_score_threshold NUMBER]
[--top_occurrences NUMBER]
[--occurrences_threshold NUMBER]
[--ic_for_rep NUMBER]
[--min_pwms_in_cluster Number]
[--db_format string]
[--input_count NUMBER]
[--db_count NUMBER]
[--db_file_type String]
[--input_file_type String]
[--input_prob Number]
[--db_prob Number]
[--qa Number]
[--seed Number]
[--damp Number]
[--max_iter Number]
[--convergence_iter Number]
[--preference Number]

Arguments:

--path_to_predicted_files	.mlp files	Folder which contain predicted files.
--db_folder	path or list	path to clustered dbds according to the hierarchy of abc4pwm. If db_type=list then then this paramter should be list of pwms, against whcih you are searching the pwm

--output_folder	FOLDER	This folder should to output folder. search_result.pdf output file will be stored here

Optional Arguments:

--tf_name	String	If you want to search specific tf in a folder then use this parameter
--db_type	String	If database for comparison is folder hierarchy like abs4pwm then this will be db_type=path. Write list if providing list of pwms for comparison
--top_n	Number	Specify how many top matches from database is required. Default 2
--min_pwms_in_cluster	Number	Number of minimum acceptable pwms in a cluster made from predicted pwms. Default 3
--db_format	string	If database for comparison have format, please mention.Supported formats are abc4pwm, Transfac, Jaspar. default is abc4pwm
--input_count	NUMBER	1 if input file contains values in counts
--db_count	NUMBER	1 if database file contains values in counts
--db_file_type	String	mention the extension of file type. e.g., .mlp, .txt
--input_file_type	String	mention the extension of file type. e.g., .mlp, .txt
--input_prob	Number	1 if input file contains values in probabilities
--db_prob	Number	1 if database file contains values in probabilities
--ic	NUMBER	Information Content for trimming edges.Default value is 0.4
--qa	Bool	1 if quality assessment of predicted files (clusters) is need, 0 is by default.
--seed	Number	Seed for random selection of cluster center. Input 1 to fix. Default Seed is 0.
--damp	Number	Damping factor (between 0.5 and 1) is the extent to which the current value is maintained relative to incoming values (weighted 1 - damping). This in order to avoid numerical oscillations when updating these values (messages).
--max_iter	Number	Maximum number of iterations.
--convergence_iter	Number	Number of iterations with no change in the number of estimated clusters that stops the convergence.
--preference	Number	Preferences for each point - points with larger values of preferences are more likely to be chosen as exemplars. The number of exemplars, ie of clusters, is influenced by the input preferences value. If the preferences are not passed as arguments, they will be set to the median of the input similarities.

13. Example

Examples of all modules are given in demo folder.