

MMPred User-Manual

Filippo Guerri

September 23, 2024

1 Introduction

This user guide provides instructions for using **MM-Pred**. The program can be used only as a command line tool in Linux environment.

2 Installing the software

The software is compatible with python3.8 or later versions. Download the folder from [link](#) than run the following commands to create a virtual environment (this steps will ensure that the necessary packages to run the program are available). The installation of the virtual environment has to be performed only once.

```
python3 -m venv venv
source venv/bin/activate
pip install -r requirements.txt
```

Once the virtual environment has been installed a first time and the folder "venv" has been created, it has to be activated every time a new session is initialized, running the command:

```
source venv/bin/activate
```

If the user wants to include NetMhcIIpan in the analysis, the software has to be downloaded from [IEDB](#) (section MHC-II binding predictions - Download). Also, to include the Blast alignment in the analysis Blast+ has to be installed. Run the following script to install Blast+:

```
wget ftp://ftp.ncbi.nlm.nih.gov/blast/executables/blast+/2.12.0/ncbi-
blast-2.12.0+-x64-linux.tar.gz
```

```
tar -xzf ncbi-blast-2.12.0+-x64-linux.tar.gz
```

3 Run Epitope Prediction

```
python3 MMPred.py -q QUERY -a ALLELE
```

- **QUERY (mandatory):** The fasta file to which epitope predictions is applied.
- **ALLELE (mandatory):** A 1-column txt file with the identifier of the alleles.

This script applies the CNNPEPPRED prediction for the alleles specified in ALLELE to the protein sequences in QUERY.

```
python3 MMPred.py -q ... -a ... -n NETMHCIIPAN_PATH
```

- **NETMHCIIPAN_PATH (optional):** The relative or absolute path for the folder "mhc_ii" of the NetMhciipan software.

This script applies both CNNPEPPRED and NETMHCIIPAN (Ba and EI) predictions for the alleles specified in ALLELE to the protein sequences in QUERY.

```
python3 MMPred.py ... -m MODE
```

- **MODE (optional):** Can be either "protein" or "peptide." It specifies if the QUERY file contains full-length protein sequences or small peptide/epitopes.

If MODE is set to "protein" the program will identify a 9-mer core for each window of size W of the sequences in QUERY.

If MODE is set to "peptide" the program predicts one 9-mer core for each sequence in QUERY.

DEFAULT = protein

```
python3 MMPred.py ... -m protein -w W
```

- **W (optional):** The window size for the prediction. DEFAULT = 15. Used only when MODE = protein and only if the alignment is performed (see below).

```
python3 MMPred.py ... -r RESULTS_FOLDER
```

- **RESULTS_FOLDER (optional):** The name of the results folder.

4 Run Epitope Prediction with Alignment

```
python3 MMPred.py -b BLAST_PATH -q QUERY -t TARGET -a ...
```

- **BLAST_PATH (mandatory):** Relative or absolute path to the blast folder "ncbi-blast-2.12.0+".

- **QUERY** (mandatory): Fasta file.
- **TARGET** (mandatory): Fasta file.

QUERY is aligned against TARGET. Epitope prediction is then applied to the TARGET's sequences that show a significant alignment with QUERY.

```
python3 MMPred.py -b BLAST_PATH -q QUERY -t TARGET -a ... -afp AF_PAR -afv AF_VAL
```

- **AF_PAR** (optional): Parameter chosen to filter alignment, possible values: "evalue", "bitscore". DEFAULT = evalue.
- **AF_VAL** (optional): The cutoff to filter the alignments. DEFAULT = 0.05.

```
python3 MMPred.py ... -alg_mode ALG_MODE
```

- **ALG_MODE** (mandatory): Is the alignment mode, can be either "blastp" or "psiblast".

```
python3 MMPred.py ... -alg_mode psiblast -pssm_comp_db EPITOPES_DB -n_core N
```

- **EPITOPES_DB** (mandatory when using psiblast): Is the fasta file containing the set of sequences against which a psiblast search is performed to compute the PSSM.
- **N** (optional): number of core to be used to ue parallele computation of the PSSM, can be used only when using psiblast.

5 To Run the Pipeline from the Parameter File

```
python3 MMPred.py -getPF PARAM_FILE_NAME
```

- **PARAM_FILE_NAME**: The name of the empty parameter file. An empty parameter file named PARAM_FILE_NAME is generated using this script. Instructions on how to use it are in the file itself.

```
python3 MMPred.py -PF PARAM_FILE_NAME
```

- Runs the pipeline with the parameters specified in PARAM_FILE_NAME.

6 Output

The program will generate a folder named RESULTS_FOLDER which contains a file named "PRED_SUMMARY.csv" with a table summarizing all the results.

6.1 Without-alignemnt

When alignment is not applied, "PRED_SUMMARY.csv" columns are:

- **core:** sequence of the predicted core (9-residues)
- **query:** ID of the query sequence as in the fasta file (QUERY) in input
- **start:** start of the predicted core
- **end:** end of the predicted core
- **method:** epitope prediction method
- **score:** prediction score
- **rank:** prediction %Rank
- **allele:** allele used in the prediction

6.2 with Alignment

When alignment is applied, the "PRED_SUMMARY.csv" columns are:

- **core:** sequence of the predicted core (9-residues)
- **target_seq_id:** ID of the target sequence, as specified in the fasta file (TARGET) in input.
- **target_window_start:** starting position of the window of size W extracted from the target sequence after the alignment.
- **target_window_end:** ending position of the window of size W extracted from the target sequence after the alignment.
- **target_alg_start:** starting position of the alignment for the target sequence.
- **target_alg_end:** ending position of the alignment for the target sequence.
- **target_core_start:** starting position of the predicted core. This position is relative to the window of size W.
- **target_core_end:** ending position of the predicted core. This position is relative to the window of size W.
- **ident:** identity obtained in the alignment
- **eval:** E-value obtained in the alignment
- **bitscore:** Bit-score obtained in the alignment
- **target_aligned_seq:** aligned target sequence, .i.e is the sequence from position target_alg_start to target_alg_end

- **query_seq_id**: ID of the query sequence, as specified in the fasta file (QUERY)
- **query_alg_start**: starting position of the alignment for the query sequence
- **query_alg_end**: ending position of the alignment for the query sequence
- **query_aligned_seq**: aligned query sequence, i.e. is the sequence from position query_alg_start to query_alg_end.
- **method**: epitope prediction method
- **score**: prediction score
- **rank**: prediction %Rank
- **allele**: allele used in the prediction