SSR_VDFDA

SSR_VDFDA is a tool designed to establishes a DNA fingerprint database based on whole genome SSR characteristics and is used to predict variety for unknown individuals. It consists of three parts: model construction, model evaluation and individual variety identification.

Requirement:

- minia
- misa (only misa.pl is needed)
- python 3.9.1

Make sure they are all installed and added to the environment variables.

Installation

Shell

```
conda install -c oldcat931 ssr_vdfda
```

Usage

1 SSR_VDFDA_model_build.py

This script integrates three key steps: 1) assembly of sequencing data using Minia, 2) extraction of SSR information using MISA, and 3) screening of SSR information and model construction. Users can optionally execute the first two steps by themselves if desired.

Shell

```
SSR_VDFDA_model_build.py -s 1 -e 3 -pp 0.8 -index index_file -o output_path
```

The following parameters are essential:

- **-s**: Specifies the start stage.

- **-e**: Specifies the end stage.
- **-pp**: Indicates the polymorphism (as described in the main text of the paper).
- **-o**: Designates the output path for storing the final identification model and SSR lists.
- **-index**: Refers to the index file.

We have two index files. The primary index file, detailed in the table below, contains two columns. The first column lists the variety for each individual, while the second column provides the corresponding absolute path for the input file of Minia. Please note that the paths are absolute paths.

Plain Text

```
1     /Rho/data/W-1-01
1     /Rho/data/W-1-02
2     /Rho/data/W-2-01
2     /Rho/data/W-2-02
3     /Rho/data/W-3-01
3     /Rho/data/W-3-02
4     /Rho/data/W-4-01
4     /Rho/data/W-4-02
5     /Rho/data/W-5-01
5     /Rho/data/W-5-02
6     /Rho/data/W-6-01
6     /Rho/data/W-6-02
7     /Rho/data/W-7-01
7     /Rho/data/W-7-02
8     /Rho/data/W-8-01
8     /Rho/data/W-8-02
```

When Minia's input includes multiple files, an index file is used to specify the input files. This is referred to as the second-level index file. For example, the second-level index file for "/Rho/data/W-1-01" is shown in the below table. Since our files are paired-end sequencing files, each line in this file represents one end of a sequencing pair. If you are working with different types of files, you can modify the index file accordingly. If you have already completed the Minia assembly, you do not need the second-level index file.

Plain Text

```
/Rho/data/W-1-01_R1.fq
/Rho/data/W-1-01_R2.fq
```

The first-level index file is always necessary under any conditions.

The following parameters are optional, users can adjust them according to your computer configuration:

- **-misap**: Specifies the number of MISA processes to use; the default is 1
- **-miniap:** Specifies the number of minia processes to use; the default is 1
- **-miniac:** Specifies the number of cores for Minia processes; the default is 1

Next, I will explain the output files generated by this software. For example, if your index file is named "index_file" and your polymorphism indicator is "0.8" (please refer to our main text for details), the output files include the following:

- all_data.csv;
- polymorphism_ssr0.8.csv;
- polymorphism_ssr_list.pkl;
- predict_model.pkl.

Shell

```
output_path/index_file_0.8/
                 └── all_data.csv
                 └── polymorphism_ssr0.8.csv
                 └── polymorphism_ssr_list.pkl
                 └── predict_model.pkl
```

2 VDFDA_tSNE_Kmeans.py

Shell

```
VDFDA_tSNE_Kmeans.py -index index_file -i polymorphism_ssr.csv
```

The following parameters are essential:

- **-index_file** Specifies the index file, which should be the same as the one used in `SSR_VDFDA_model_build.py`
- **-i**: Specifies the input file, which is the `polymorphism_ssr.csv` generated by `SSR_VDFDA_model_build`.

Additionally, after running `SSR_VDFDA_model_build.py`, the output information will include the code used for generating the validation results.

3 SSR_VDFDA_model_predict.py

Shell

```
SSR_VDFDA_model_predict.py -i -d -index -k
```

The following parameters are essential:

- **-i**: Specifies the SSR information file for the individual whose variety is to be predicted.
- **-d**: Indicates the folder containing the KNN recognition model file and the filtered SSR list, which are the final results from the previous step.
- **-index**: Refers to the same index file used in the first step.
- **-k**: Represents the number of nearest neighbors (k) in the KNN prediction.

Here is an example:

Shell

```
SSR_VDFDA_model_predict.py -i W-1-10.contigs.fa.misa -d /out/index_file_6_27.txt_0.8 -index index_file_6_27.txt -k 37
```

Please note that the prediction results are not returned directly. Instead, due to variations in the number of individuals per variety within the model, the output will list the top *k* closest points to the predicted individual based on distance. For example, if *K* is set to 5, the output will display the 5 nearest points, sorted by proximity to the predicted individual.