SeqFea-Learn: A Python pipeline tool for feature extraction, feature selection, machine learning and deep learning based on DNA, RNA and protein sequence data

Name Surname[1,*], Name Surname[2] and Name Surname[2]

[1] Department, Institution, Town, State, Postcode, Country
[2] Department, Institution, Town, State, Postcode, Country

User Manual

CONTENTS

Introduction

SeqFea-Learn is a Python pipeline tool for analyzing DNA, RNA, and protein sequencing data that integrated 19 feature extraction methods for DNA, 15 feature extraction methods for RNA, 32 feature extraction methods for Protein sequencing data, 21 feature selection methods, 15 dimensionality reduction methods, 7 clustering methods, 5 sampling methods, 10 classification methods, and 3 deep learning methods. This document will provide user a full details tutorial.

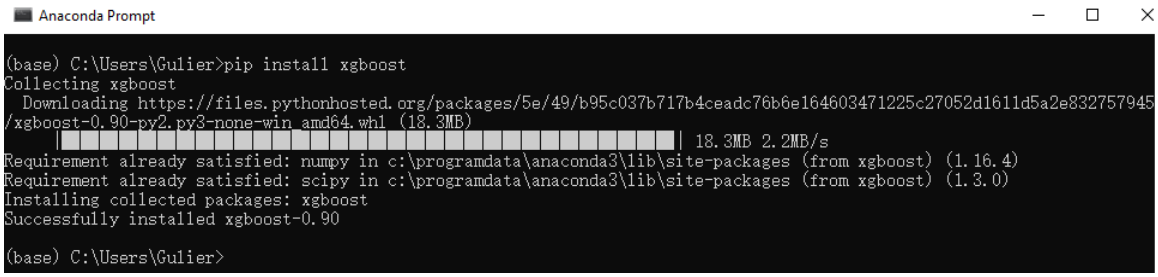
Installation

The tool is developed using Python 3 (Python Version 3.0 of above) and it can be run on both **Windows** and **Linux** operating system. We strongly recommend user to install Anaconda Python 3.6 or above version to avoid installing other packages. Anaconda can be freely downloaded from https://www.anaconda.com/distribution/#download-section. After installing Anaconda, the following packages need to be installed:

1. Xgboost
2. Skrebate
3. Lightgbm
4. Tensorflow
5. Keras
6. Imblearn

For Windows, open your Anaconda Prompt, then run the following codes:

```
pip install xgboost
pip install skrebate
pip install lightgbm
pip install tensorflow
pip install keras
pip install imblearn
```



The source code is freely available at

To install our tool, download the Zip file from Github and unzip the file to the location you want.

For Linux, download Anaconda for Linux from the website above and load your Anaconda module, then run the following codes:

```
module load anaconda
pip install xgboost
pip install skrebate
pip install lightgbm
pip install tensorflow
pip install keras
pip install imblearn
```

Use the code below to install the package:

```
cd your_folder_path
wget https://github.com/ashinandjay/FeatureSelection/archive/master.zip
```

Unzip the file:

```
unzip master.zip
```

Data preparation

SeqFea-Learn accepts DNA, RNA, and Protein for sequencing data and they must be FASTA Format.

```
>P_1
TGATTTTCAGTTTTCTCGCCATGTTCGAGGTCCTACAGTTT
>P_2
GAAAATCACGGAAAATGAGAAGTACACACTTTGGGACATGA
>P_3
ATGTCCACTGTAGGACGTGGAGTATGGCAAGAAAACTGAAA
>P_4
TACACACTTTAGGACATGAAATAAAGCGAGGAAAACTGAAA
>P_5
ATTGAAAATGAGAAACATACAGTTGACGACTTGAAAAATGA
```

SeqFea-Learn accepts label of sequencing data as TXT format.

```
>P_1
1
>P_2
1
>P_3
1
>P_4
1
>P_5
1
```

DNA feature extraction

SeqFea-Learn contains 19 DNA feature extraction methods. For convenience, we assign each method a method number that help user to execute the code. The details of each method was shown in our supplementary documents. This code will generate extracted feature vector in CSV format.

Feature Extraction Method	Method Number
Kmer	1
Reverse compliment kmer (RCKmer)	2
Nucleic acid composition (NAC)	3
Di-nucleotide composition (DNC)	4
Tri-Nucleotide Composition (TNC)	5
Binary encoding (BE)	6
zCurve Mathematical Formula (zCurve)	7
Dinucleotide based auto covariance (DAC)	8
Dinucleotide based cross covariance (DCC)	9
Di-nucleotide based auto-cross covariance (DACC)	10
Tri-nucleotide based auto covariance (TAC)	11
Tri-nucleotide based cross covariance (TCC)	12
Tri-nucleotide based auto-cross covariance (TACC)	13
Position-specific dinucleotide propensity (PSDNP)	14
Position-specific trinucleotide propensity (PSTNP)	15
MonoKGap theoretical description (MonoKGap)	16
MonoDiKGap theoretical description (MonodiKGap)	17

Pseudo di-nucleotide composition (PseDNC)	18
Pseudo k-tuple nucleotide composition (PseKNC)	19

For windows, open your Anaconda Prompt and follow this code to execute DNA feature extraction:

```
cd your_folder_path
python DNA_Feature_Extraction.py [DNA Extraction number] [DNA sequencing data]
```

For Linux, follow this code to execute DNA feature extraction:

```
cd your_folder_path
module load anaconda
python DNA_Feature_Extraction.py [DNA Extraction number] [DNA sequencing data]
```

Example: use Kmer method to extract features from DNA sequencing data

```
python DNA_Feature_Extraction.py 1 DNA_sequencing.fast
```

RNA feature extraction

SeqFea-Learn contains 15 RNA feature extraction methods. For convenience, we assign each method a method number that help user to execute the code. The details of each method was shown in our supplementary documents. This code will generate extracted feature vector in CSV format.

Feature Extraction Method	Method Number
Kmer	1
Reverse compliment Kmer (RCKmer)	2
Nucleic acid composition (NAC)	3
Di-nucleotide composition (DNC)	4
Tri-Nucleotide Composition (TNC)	5
Binary encoding (BE)	6
zCurve Mathematical Formula (zCurve)	7
Dinucleotide based auto covariance (DAC)	8
Dinucleotide based cross covariance (DCC)	9
Di-nucleotide based auto-cross covariance (DACC)	10
Position-specific dinucleotide propensity (PSDNP)	11
Position-specific trinucleotide propensity (PSTNP)	12
MonoKGap theoretical description (MonoKGap)	13
MonoDiKGap theoretical description (MonodiKGap)	14
Pseudo di-nucleotide composition (PseDNC)	15

For windows, open your Anaconda Prompt and follow this code to execute RNA feature extraction:

```
cd your_folder_path
python RNA_Feature_Extraction.py [DNA Extraction number] [DNA sequencing data]
```

For Linux, follow this code to execute RNA feature extraction:

```
cd your_folder_path
module load anaconda
python RNA_Feature_Extraction.py [RNA Extraction number] [RNA sequencing data]
```

Example: use DNC method to extract features from RNA sequencing data

```
python RNA_Feature_Extraction.py 4 RNA_sequencing.fast
```

Protein feature extraction

SeqFea-Learn contains 31 Protein feature extraction methods. For convenience, we assign each method a method number that help user to execute the code. The details of each method was shown in our supplementary documents. This code will generate extracted feature vector in CSV format.

Protein feature extraction methods	Method Number
Amino acid composition (AAC)	1
Dipeptide composition (DC)	2
Composition of k-spaced amino acid pairs (CKSAAP)	3
Grouped dipeptide composition (GDC)	4
Grouped tripeptide composition (GTC)	5
Conjoint triad (CT)	6
K-spaced conjoint triad (KSCTriad)	7
Composition (C)	8
Transition (T)	9
Distribution (D)	10
Encoding based on grouped weight (EBGW)	11
Auto covariance (AC)	12
Moreau-Broto autocorrelation (Morean-Broto)	13
Moran autocorrelation (Moran)	14
Geary autocorrelation (Geary)	15
Quasi-sequence-order (QSO)	16
Pseudo-amino acid composition (PseAAC)	17

Amphiphilic pseudo-amino acid composition (APAAC)	18
Amino acid composition PSSM (AAC-PSSM)	19
Dipeptide composition PSSM (DPC-PSSM)	20
Bi-gram PSSM (Bi-PSSM)	21
Auto covariance PSSM (AC-PSSM)	22
Pseudo PSSM (PsePSSM)	23
AB-PSSM	24
Secondary structure composition (SSC)	25
Accessible surface area composition (ASA)	26
Torsional angles composition (TAC)	27
Torsional angles bigram (TA-bigram)	28
Structural probabilities bigram (SP-bigram)	29
Torsional angles auto-covariance (TAAC)	30
Structural probabilities auto-covariance (SPAC)	31

For windows, open your Anaconda Prompt and follow this code to execute Protein feature extraction:

```
cd your_folder_path
python Protein_Feature_Extraction.py [Protein Extraction number] [Protein sequencing data]
```

For Linux, follow this code to execute Protein feature extraction:

```
cd your_folder_path
module load anaconda
```

```
python Protein_Feature_Extraction.py [Protein Extraction number] [Protein
sequencing data]
```

Example: use AAC-PSSM method to extract features from DNA sequencing data

```
python RNA_Feature_Extraction.py 19 Protein_sequencing.fasta
```

Feature Selection

SeqFea-Learn contains 21 feature selection methods. For convenience, we assign each method a method number that help user to execute the code. The details of each method was shown in our supplementary documents. This code will generate selected feature vector in CSV format.

Feature selection methods	Method Number
LASSO	1
Elastic net (EN)	2
L1-SVM	3
L1-LR	4
Extra-Trees-RFE	5
XGBosst-RFE	6
SVM-RFE	7
LR-RFE	8
Mutual information (MI)	9
Minimum redundancy maximum relevance (MRMR)	10
Joint mutual information (JMI)	11
Maximum relevance maximum distance (MRMD)	12
Information gain (IG)	13
Chi-square test (CHI2)	14
Pearson correlation (Pearson)	15
ReliefF	16
Trace Ratio	17

Gini Index	18
Spectral Feature Selection (SPEC)	19
Fisher Score	20
T-Score	21

For windows, open your Anaconda Prompt and follow this code to execute feature selection method:

```
cd your_folder_path
python Feature_Selection.py [Feature selection number] [Number of feature to select] [Feature Vectors] [Label Vectors]
```

For Linux, follow this code to execute feature selection method:

```
cd your_folder_path
module load anaconda
python Feature_Selection.py [Feature selection number] [Number of feature to select] [Feature Vectors] [Label Vectors]
```

Example: use LASSO method to select 50 features from extracted feature vector

```
python Feature_Selection.py 1 50 Kmer_out.csv label.txt
```

Dimensionality Reduction

SeqFea-Learn contains 15 dimensionality reduction methods. For convenience, we assign each method a method number that help user to execute the code. The details of each method was shown in our supplementary documents. This code will generate dimension reduced feature vector in CSV format.

Dimensionality Reduction Methods	Method Number
Principal component analysis (PCA)	1
Kernel PCA (KPCA)	2
Locally linear embedding (LLE)	3
Multi-dimensional scaling (MDS)	4
t-distributed stochastic neighbor embedding (T-SNE)	5
Truncated singular value decomposition (SVD)	6
Non-negative matrix factorization (NMF)	7
Gaussian random projection (GRP)	8
Sparse random projection (SRP)	9
Independent component analysis (ICA)	10
Factor analysis (FA)	11
Agglomerate feature (AF)	12
Autoencoder	13
Gaussian noise autoencoder	14
Variational autoencoder	15

For windows, open your Anaconda Prompt and follow this code to execute dimensionality reduction method:

```
cd your_folder_path
python Feature_Reduction.py [Feature reduction number] [Number of dimension to select] [Feature Vectors]
```

For Linux, follow this code to execute dimensionality reduction method:

```
cd your_folder_path
module load anaconda
python Feature_Reduction.py [Feature reduction number] [Number of dimension to select] [Feature Vectors]
```

Example: use T-SNE method to reduce to 50 dimension from extracted feature vector

```
python Feature_Reduction.py 5 50 Kmer_out.csv
```

Clustering method

SeqFea-Learn contains 7 clustering methods. For convenience, we assign each method a method number that help user to execute the code. The details of each method was shown in our supplementary documents. This code will generate vector in CSV format after applying clustering method.

Clustering methods	Method Number
K-means	1
Spectral Clustering	2
Gaussian Mixture Clustering	3
Affinity Propagation Clustering	4
Mean Shift	5
DBSCAN	6
OPTICS	7

For windows, open your Anaconda Prompt and follow this code to execute clustering method:

```
cd your_folder_path
python Feature_Clustering.py [Feature Clustering number] [Feature Vectors]
```

For Linux, follow this code to execute dimensionality reduction method:

```
cd your_folder_path
module load anaconda
python Feature_Clustering.py [Feature Clustering number] [Feature Vectors]
```

Example: use DBSCAN clustering method to create clusters from extracted feature vector

```
python Feature_ Clustering.py 6 Kmer_out.csv
```

Sampling method

SeqFea-Learn contains 5 sampling methods. For convenience, we assign each method a method number that help user to execute the code. The details of each method was shown in our supplementary documents. This code will generate a vector in CSV format after applying sampling method.

Sampleing methods	Method Number
Random over sampling (ROS)	1
Synthetic minority oversampling technique (SMOTE)	2
Adaptive synthetic (ADASYN)	3
Random under sampling (RUS)	4
Neighbourhood cleaning rule (NCR)	5

For windows, open your Anaconda Prompt and follow this code to execute sampling method:

```
cd your_folder_path
python Feature_Sampling.py [Feature Sampling number] [Feature Vectors]
```

For Linux, follow this code to execute sampling method:

```
cd your_folder_path
module load anaconda
python Feature_Sampling.py [Feature Sampling number] [Feature Vectors]
```

Example: use NCR sampling method to sample extracted feature vector

```
python Feature_ Sampling.py 5 Kmer_out.csv
```

Model Evaluation

SeqFea-Learn integrated 13 classification methods. Execute this code will construct all 13 classification methods to compare performance of each classifier. This code will generate box plot of classification accuracies, plot of ROC curves of all 13 classification methods, and a classification accuracies table in CSV format.

For windows, open your Anaconda Prompt and follow this code to execute model evaluation:

```
cd your_folder_path
python Feature_Evaluation.py [Feature Vectors] [Label Vectors]
```

For Linux, follow this code to execute model evaluation:

```
cd your_folder_path
module load anaconda
python Feature_Evaluation.py [Feature Vectors] [Label Vectors]
```

Example: execute model evaluation based on a vector that using kmer feature extraction method and Lasso feature selection method

```
python Feature_Evaluation.py Kmer_Lasso_out.csv DNA_label.txt
```

Feature Prediction

SeqFea-Learn integrated 13 classification methods to make predictions. For convenience, we assign each classifier a method number that help user to execute the code. This code will generate a prediction results, ROC curve based on the input dataset.

Classifier	Method Number
Support vector machine (SVM)	1
K-nearest neighbor (KNN)	2
Random forest (RF)	3
Extremely randomized trees (Extra-Trees)	4
Gradient boosting decision tree (GBDT)	5
XGBoost	6
LightGBM	7
Bagging classifier (Bagging)	8
AdaBoost	9
Gaussian Naïve Bayes (GNB)	10
Deep neural network (DNN)	11
Convolutional neural network (CNN)	12
Recurrent neural network (RNN)	13

For windows, open your Anaconda Prompt and follow this code to execute feature prediction:

```
cd your_folder_path
python Feature_Prediction.py [Feature Vectors] [Label Vectors]
```

For Linux, follow this code to execute model evaluation:

```
cd your_folder_path

module load anaconda

python Feature_Prediction.py [classification method number] [Feature Vectors]
[Label Vectors]
```

Example: execute random forest for a vector which extracted using Kmer feature extraction method and Lasso feature selection method

```
python Feature_Prediciton.py 3 Kmer_Lasso_out.csv DNA_label.txt
```