**Feature-Analysis: a Python pipeline tool for feature extraction, feature selection, machine learning and deep learning based on DNA, RNA and protein sequence data**

**ALLFEATURE: an integrated Python pipeline tool for extracting, selecting, modeling and evaluating features of DNA, RNA and Proteins**

**sequencing data**

**Authors…**

**ABSTRACT**

The faster and cheaper modern sequencing technology revolutionizes the genomic research and triggers explosive growth of DNA, RNA and proteins sequencing data. Using bioinformatics tool to analyze and process the DNA, RNA and protein sequence data efficiently and accurately has become a popular and challenging task in genomics and proteomics fields. With the development of experimental methods and statistical learning, an integrated and user-friendly tool that containing the state-of-the-art data mining methods are needed. Here, we propose **ALLFEATURE**, a comprehensive Python pipeline tool that integrating multiple steps: feature extraction, dimensionality reduction, feature selection, machine learning and deep learning models construction to analyze DNA, RNA, protein sequencing data. We used DNA enhancers, RNA N6-methyladenosine sites and protein-protein interactions datasets to evaluate the validation of the package. **ALLFEATURE** can effectively perform biological sequence analysis and predictor construction. The Python pipeline tool can be download **…**

**Key words**: Sequencing data analysis; feature extraction; feature selection; machine learning; deep learning

# INTRODUCTION

The appearance of next-generation sequencing (NGS) technology has significantly improved the quantities and qualities of biological sequences (DNA, RNA, and protein). With NGS, the duration for sequencing an entire human genome is reduced from a decade to a single day [1] and its cost dropped from $300000 to less than $1000 [2]. The most recent released version, 232 of GenBank in NCBI contains 213,387,758 sequences and WGS in NCBI includes 1,022,913,321 sequences [3]. Unlike the microarrays, NGS has the ability to analyze and deep investigate the transcriptome for any species because it does not require species or transcript-specific probes. In addition, the technology provides much higher resolution and accuracy with a much lower variation. NGS is a powerful tool or platform which can sequence of thousands to millions of DNA molecules simultaneously [4]. It provides researchers a precious opportunity to rapidly sequence whole genomes and study the structure and function from the perspective of biological sequences [5, 6]. Analyzing sequencing data help researches to explore disease diagnosis [7-9], biotechnology [10] and many other areas.

It is impossible to analyze sequencing data by human inspection alone. Therefore, sequencing data analysis deeply relies on computational tools. Recent research shows that machine learning methods and algorithms have been heavily applied to explore structural and functional properties of DNA, RNA and protein sequencing data [11]. The analysis of biological sequence can be mainly regarded as binary and multi-class prediction tasks [12, 13], including DNA N6-methyladenosine site [14], DNA N4-methylcytosine site [15], RNA N6-methyladenosine site [16], RNA-binding protein identification [17], protein function site [18], protein fold recognition [19, 20], protein-protein interaction prediction [21-23], etc. For the process to construct predictors, the first step is feature extraction, which transforms the character sequence data into numeric vectors of the same length [24]. The feature information in sequencing data can be classified as sequence-based, physicochemical property-based, evolution-based and structure-based biological information. After extracting feature vectors, selecting a suitable classifier algorithm is another step for the prediction model construction. Many popular models can be used for researchers to analyze the structure and function of sequencing data, such as XGBoost [25], support vector machine (SVM) [26] and k nearest neighbors (KNN) [27], etc.

With the development of bioinformatics, some web servers and standalone packages are developed, including repDNA, Propy, PROFEAT, Pse-in-one, PyFeat, POSSUM, iFeature for only feature extraction, BioSeq-Analysis, iLearn and BioSeq-Analysis2.0 for both feature extraction and predictor construction. At the same time, FeatureSelect and scikit-feature are designed for only feature selection. Although many bioinformatics tools are developed, there are still some obstacles need to be conquered. (i) The existing tools mainly fulfill individual step in the process of biological sequence analysis, and this is not enough to study biological sequences. Most computational tools or web servers only focus on one individual step instead of integrating all functionalities for sequencing data analysis. For instance, repDNA is a python package to only generate various types of feature vectors from DNA sequences [29]. Pse-in-one 2.0 [30], PyFeat [31] and PROFEAT [32] are generation tools for extracting different feature models from DNA, RNA, and protein sequences. (ii) Overfitting issues often exist during performing accurate predictions in machine learning [28]. Based on our observation, extracted feature vectors from the feature extraction step often displays high dimensionality, especially some protein feature vectors display extremely high dimensionality which can cause time-consuming issues and poor accuracy of prediction. (iii) The available packages lack of feature selection and dimensional dimensionality reduction methods, and ignore some state-of-the-art machine learning methods. (v) Deep learning is a very powerful computational tool for classification tasks via layer by layer learning [36]. However, the packages above lack the usages of deep learning theories.

Especially, high dimensionality will consume memory storage, significantly increase execution time of model construction and hurt prediction accuracy. Feature selection methods just select those features that contribute most to predictions, including information-based, tree-based, regularization-based, statistic-based methods. Some popular and powerful feature selection algorithms are widely used such as the Chi-squared test [39], SVM-RFE [40], Lasso [41], Pearson correlation [42], ReliefF [43], etc. In addition, for some feature vectors with extremely high dimensionalities, unsupervised dimension reduction methods are often involved in the sequencing data analysis to pre-filter vectors to reduce the time cost of models construction. Dimensionality reduction can project raw feature space with high dimensionality to a new feature space via the linear or non-linear combination. Dimensionality reduction and feature selection both can reduce model’s complexity, computational resource cost and execution time, prevent overfitting issue and improve accuracy of prediction to provide more reliable predictions.

Inspired by above discussion, we propose a new Python pipeline, **ALLFEATURE**, that integrated 20 feature selection methods, a total of 16 dimensionality reduction methods and 13 prediction/classification models. In addition, our pipeline tool also contains the step of feature extraction to generate a total of 60 different models of features from DNA, RNA and protein sequencing data, **Figure 1**. Compared with other software packages or webservers, the proposed ALLFEATURE has the following advantages: (i) A user-friendly software package to fulfill the sequence analysis via feature extraction, feature selection, dimensionality reduction, machine learning, deep learning and predictor evaluation. (ii) Sufficient feature selection methods and dimension reduction methods, including regularization-based, statistic-based, information-based, tree-based and recursive feature elimination-based approaches, which could determine the optimal feature subset and eliminate noisy, unrelated, and redundant information. (iii) Ten popular machine learning methods are provided and based on these methods, the user can choose appropriate classifier to predict the sequence structure and function. In this way, the better predictor can be established through optimizing feature fusion, feature selection and classifier construction. (iv) Three deep learning approach: deep neural network (DNN) [44], convolutional neural network (CNN) [45], and recurrent neural network (RNN) [46] are provided. (v) More abundant graphical display results, including box figure, ROC curves, etc.

G:\feature_selection_package\正文\flowchart\flowchart (第二版).tif

**Figure 1.** The pipeline of ALLFEATURE. The python package contains feature extraction, feature selection, machine learning, deep learning, and performance evaluation. The input is the DNA, RNA or protein sequences with FASTA format or the high-dimension matrix with csv format. The outputs of the pipeline will provide generated feature vectors, prediction accuracy comparison, and suggestion of the best model for researchers.

# MATERIALS AND METHODS

The DNA, RNA and protein biological sequence  with  residues can be regarded as:

 (1)

where  represents the  residue. And the biological sequence contains important and effective information of structure and function. How to analyze the attributes of sequence and construct useful predictors based on machine learning is necessary. The commonly steps for modeling the predictor based on DNA, RNA and protein sequence: (i) Feature extraction, which obtain the composition information, physiochemical information and evolutionary information and structural information; (ii) Feature selection and dimensionality reduction, which remove redundancy and noise from the extracted feature vectors and retain clean, effective and understandable feature information; (iii) Machine learning and deep learning, which could predict the structure and function of sequencing data via predictors; (iv) Models evaluation, which generate the prediction results of tables and figures.

## Feature extraction

**ALLFEATURE** directly extracts features from DNA, RNA or protein sequences based on a total of 60 different types of feature extraction methods. The step of feature extraction consists of 16 feature extraction methods for DNA, **Table 1**; 12 feature extraction methods for RNA, **Table 2**; 32 feature extraction methods for protein sequences, **Table 3**.

**Table 1**. List of 16 DNA feature extraction methods and their descriptions

|  |  |
| --- | --- |
| **DNA Feature Extraction Methods** | **Method Description** |
| Kmer | DNA sequence is represented as the occurrence frequencies of k neighboring nucleic acids [47, 48] |
| Reverse Compliment Kmer (RCKmer) | A variant of Kmer descriptor by removing the reverse compliment Kmer [47, 49] |
| Pseudo Dinucleotide Composition (pseDNC) | Incorporating the contiguous local sequence-order and global sequence-order information [50] |
| Pseudo k-tuple Nucleotide Composition (pseKNC) | Extending the pseDNC by incorporating k-tuple nucleotide composition [51] |
| Dinucleotide Based Auto Covariance (DAC) | Measuring the correlation of the same physicochemical index between two dinucleotides separated by lag along the sequence [52, 53] |
| Dinucleotide Based Cross Covariance (DCC) | Measuring the correlation of two different physicochemical indices between two dinucleotides separated by lag nucleic acids [52, 53] |
| Dinucleotide Based Auto-cross Covariance (DACC) | Combining of DAC and DCC [52-53, 29] |
| Trinucleotide Based Auto Covariance (TAC) | Measuring the correlation of the same physicochemical index between trinucleotides separated by lag nucleic acids [29] |
| Trinucleotide Based Cross Covariance (TCC) | Measuring the correlation of two different physicochemical indices between two trinucleotides separated by lag nucleic acids [29] |
| Trinucleotide Based Auto-Cross Covariance (TACC) | Combining of TCC and TACC [29] |
| Nucleic Acid Composition (NAC) | Calculating the frequency of each nucleic acid type in nucleotide sequence [34] |
| Di-Nucleotide Composition (DNC) | Containing 16 NAC descriptors [34] |
| Tri-Nucleotide Composition (TNC) | Containing 64 NAC descriptors [34] |
| zCurve Mathematical Formula (zCurve) | Calculating three components in three axis in genomic sequence analysis [31] |
| monoMonoKGap Theoretical Description (MonoKGap) | Calculating features based on value of kgap [31] |
| monoDiKGap Theoretical Description (MonodiKGap) | Calculating features based on value of kgap [31] |

**Table 2**. List of 12 RNA feature extraction methods and their descriptions

|  |  |  |
| --- | --- | --- |
| **RNA Feature Extraction Methods** | **Method Description** | |
| Kmer | | RNA sequence is represented as the occurrence frequencies of k neighboring nucleic acids [47, 48] |
| Reverse Compliment Kmer (RCKmer) | | A variant of Kmer descriptor by removing the reverse compliment Kmer [47, 49] |
| Pseudo Dinucleotide Composition (pseDNC) | | Incorporating the contiguous local sequence-order and global sequence-order information [50] |
| Dinucleotide Based Auto Covariance (DAC) | | Measuring the correlation of the same physicochemical index between two dinucleotides separated by lag along the sequence [52, 53] |
| Dinucleotide Based Cross Covariance (DCC) | | Measuring the correlation of two different physicochemical indices between two dinucleotides separated by lag nucleic acids [52, 53] |
| Dinucleotide Based Auto-cross Covariance (DACC) | | Combining of DAC and DCC [52-53, 29] |
| Nucleic Acid Composition (NAC) | | Calculating the frequency of each nucleic acid type in nucleotide sequence [34] |
| Di-Nucleotide Composition (DNC) | | Containing 16 NAC descriptors [34] |
| Tri-Nucleotide Composition (TNC) | | Containing 64 NAC descriptors [34] |
| zCurve Mathematical Formula (zCurve) | | Calculating three components in three axis in genomic sequence analysis [31] |
| monoMonoKGap Theoretical Description (MonoKGap) | | Calculating features based on value of kgap [31] |
| monoDiKGap Theoretical Description (MonodiKGap) | | Calculating features based on value of 4 \* kgap [31] |

**Table 3**. List of 32 Protein feature extraction methods and their descriptions

|  |  |  |
| --- | --- | --- |
| **Protein Feature Extraction** | **Extraction Method Description** | |
| Amino Acid Composition (AAC) | | Calculating the frequencies of 20 kinds of amino acids [54] |
| Dipeptide Composition(DC) | | Transforming the variable length of proteins to fixed length feature vectors [54] |
| Composition of K-Spaced Amino Acid Pairs (CKSAAP) | | Extracting important intrinsic correlation information of protein sequences in multidimensional space [55-57] |
| Grouped Dipeptide Composition (GDC) | | A variation of the DPC descriptor which generates 25 descriptors [58] |
| Grouped Tripeptide Composition (GTC) | | Another variation of TPC descriptor which generates 125 descriptors [58] |
| Conjoint Triad (CT) | | Calculating the frequency of occurrence of each triad [59] |
| K-Spaced Conjoint Triad (KSCTriad) | | Combining CT and considers the continuous amino acid units that are separated by any *k* residues [60] |
| Composition (C)  Transition (T)  Distribution (D) | | Calculating composition descriptors  Calculating transition descriptors  Calculating distribution descriptors [61-63] |
| Encoding Based on Grouped Weight (EBGW) | | Capturing the continuity and discontinuity features based on grouped weight coding [64] |
| Auto Covariance (AC) | | Measuring the correlation of the same property between two residues separated by distance of *l* [65] |
| Moreau-Broto autocorrelation (Morean-Broto) | | Measuring the physiochemical and position information between two amino acid [66] |
| Moran Autocorrelation (Moran) | | Measuring the physiochemical information of adjacent amino acid [67] |
| Geary Autocorrelation (Geary) | | Measuring the physiochemical information and generate positive values [68, 69] |
| Quasi-Sequence-Order (QSO) | | Obtaining the sequence distribution patters for a specific physicochemical property [70] |
| Pseudo-Amino Acid Composition (pseAAC) | | Extracting the physicochemical information and sequence order information [71, 72] |
| Amphiphilic Pseudo-Amino Acid Composition (APAAC) | | Extracting the type-2 pseudo amino acid composition [71, 72] |
| Amino Acid Composition PSSM (ACC-PSSM) | | Calculating process of amino acid composition PSSM [73, 74] |
| Dipeptide Composition PSSM (DPC-PSSM) | | Extracting the sequence-order information in the PSSM [74] |
| Bi-gram PSSM (Bi-PSSM) | | Calculating the frequency of the transition between amino acids [75] |
| Auto Covariance PSSM (AC-PSSM) | | Measuring the correlation of the same property between two residues separated by lag [76] |
| Pseudo PSSM (psePSSM) | | Calculating the psePSSM feature vector according to the pseudo amino acid composition [77] |
| AB-PSSM | | Calculating feature vector based on averaged PSSM over blocks [78] |
| Secondary Structure Composition (SSC) | | Calculating feature based normalized count of frequency of the structural motifs present at the amino-acid residue positions [79] |
| Accessible Surface Area composition (ASA) | | Calculating feature based on normalized sum of accessible surface area [79] |
| Torsional Angles Composition (TAC) | | Calculating features based four different types of torsional angles [79] |
| Torsional Angles bigram (TA-bigram) | | Calculating feature based on the bigram of the torsional angles [79] |
| Structural Probabilities bigram (SP-bigram) | | Calculating feature based on structural probabilities for each position of amino acid residue [79] |
| Torsional Angles Auto-Covariance (TAAC) | | Calculating feature from the torsional auto-covariance [79] |
| Structural Probabilities Auto-Covariance (SPAC) | | Calculating feature from the structural probabilities [79] |

## Feature selection and dimensionality reduction

The dimension of feature vectors after feature extraction and feature fusion could be extremely high. Take the composition of k-spaced amino acid pairs (CKSAAP) for example, when the parameter, the size of dimension could be 1200. At the same time, autocorrelation descriptors using physicochemical properties to extract sequence information. The AAindex database contains 554 physicochemical properties. Therefore, ALLFEATURE integrated steps of feature selection and dimensionality reduction. Selecting those features which contribute most to the prediction can effectively improve prediction accuracy and reduce the implementation time. Only wanted the number of top-ranking features will be selected in the step of feature selection, **Table 4**. Dimensionality reduction methods can be applied for extremely high dimensionality dataset to reduce data preparation time for model constructions, **Table 5**.

**Table 4**. Feature selection methods and their descriptions

|  |  |
| --- | --- |
| **Feature Selection Method** | **Description** |
| Lasso | Using Lasso liner model to recursively eliminate features [41, 80] |
| Elastic-Net | Using Elastic-Net model to recursively eliminate features [81] |
| L1-SVM | Using SVM with L1 penalty model to recursively eliminate features [82] |
| CHI2 | Retrieving best features based on test [39, 83] |
| Pearson Correlation (PC) | Retrieving best features based on Pearson correlation [42] |
| Extra-Tree | Using Extra-Tree model to recursively eliminate features [84] |
| XGBosst | Using XGBoost model to recursively eliminate features [25. 85] |
| SVM-RFE | Using linear SVM model to recursively eliminate features [40, 86] |
| LOG-RFE | Using Logistic Regression model to recursively eliminate features [87] |
| Mutual Information (MI) | Retrieving best features based mutual information [88] |
| Minimum Redundancy Maximum Relevance (MRMR) | Selecting features that still having high correlation to the classification variable [90] |
| Joint Mutual Information (JMI) | Retrieving best features based joint mutual information [91] |
| Maximum Relevance Maximum Distance (MRMD) | Retrieving best features by measuring relevance and redundancy between features [92] |
| ReliefF | Retrieving best features by calculating and ranking a feature score for each feature [43] |
| Trace Ratio | Retrieving best features by calculating the corresponding score in trace ratio form [93] |
| Gini Index | Retrieving best features by constructing the measure function based on Gini-Index [94] |
| Spectral Feature Selection (SPEC) | Retrieving best features based on structure induced [95] |
| Fisher Score | Retrieving best features based on scores of features under the Fisher criterion [96] |
| T Score | Retrieving best features based on their t-score [97] |
| Information Gain (IG) | Retrieving best features based on their information gain [106] |

**Table 5**. Dimensionality Reduction Methods and their descriptions

|  |  |
| --- | --- |
| **Dimensionality Reduction Methods** | **Description** |
| K-means | Clustering data by separating samples in groups of equal variances [98] |
| T-SNE | Visualizing high-dimensional data [99] |
| Principal Component Analysis (PCA) | Linear dimensionality reduction using singular value decomposition [100] |
| Kernel PCA (KPCA) | Non-linear dimensionality reduction through use of kernels [100] |
| Locally linear embedding (LLE) | Reducing projection of data which preserves distances within local neighborhoods [101] |
| Truncated Singular Value Decomposition (TSVD) | Linear dimensionality reduction by means of truncated singular value decomposition [102] |
| Non-negative matrix factorization (NMF) | Reducing dimension by finding two non-negative metrices [103] |
| Multi-dimensional Scaling (MDS) | Reducing dimension by modeling data as distances in a geometric space [104] |
| Independent Component Analysis (ICA) | Reducing dimension by finding components with some sparsity [105] |
| Factor Analysis (FA) | Reducing dimension by performing a maximum likelihood estimate [107] |
| Agglomerate Feature (AF) | Recursively merges feature instead of samples [108] |
| Gaussian Random Projection (GRP) | Reducing the dimension by projecting the original input space using the Gaussian distribution [109] |
| Sparse Random Projection (SRP) | Reducing dimension by projecting the original input space using a sparse random matrix [110] |
| Autoencoder | Reducing the dimension using encode and decode neural network [111] |
| Gaussian Noise Autoencoder (GNA) | Corrupting input before being passed to autoencoder neural network [112] |
| Variational Autoencoder (VA) | Neural network can be trained with stochastic gradient descent [113] |

## Model construction

A lot of analysis of biological sequence could be regarded as classification tasks in bioinformatics and computational biology. Thus, selecting suitable machine learning and deep learning methods are a key step for predictor construction. SVM is a kernel-based classifier through constructing the optimal hyperplane. RandomForest (RF) is a tree-based and widely used machine learning method. SVM and RF are commonly used in the area of prediction tasks. Recently, gradient boosting decision tree (GBDT) [114], XGBoost and LightGBM are demonstrated to be excellent classifiers via gradient boosting algorithm. **ALLFEATURE** integrated 10 popular machine learning methods SVM, KNN, RF, LightGBM, XGBoost, Adaboost [115], Extra-Tree, gaussian Naïve Bayes (GNB) [116], GBDT. In recent years, deep learning frameworks provide effective solutions for the analysis of DNA, RNA and protein sequence. Deep learning can explore essential sequences represent information via hierarchical structure, which can accurately, effectively handle sequencing prediction tasks. **ALLFEATURE** integrated three deep learning methods, including deep neural network (DNN), convolutional neural network (CNN), and recurrent neural network (RNN).

## Cross-validation and Model evaluation

After the step of feature extraction, feature selection or dimensionality reduction, the predicting model can be constructed. In ALLFEATURE, K-fold cross-validation, leave-one-out cross-validation and hold-out evaluation methods are all used for evaluating classification models. Meanwhile, in order to measure the designed statistical prediction model, sensitivity is also the recall rate (SE), specificity (SP), accuracy (ACC), Matthew's correlation coefficient (MCC), precision (PRE), and F-measure (F1) are sued to evaluate the model performance.

 (2)

 (3)

 (4)

 (5)

 (6)

 (7)

where TP, TN, FP and FN in the above equations represent true positive, true negative, false positive and false negative, respectively. The Receiver Operating Characteristic (ROC) curve, the Precision-Recall (PR) curve and their corresponding AUC (area under the ROC curve) and AUPR (area under the PR curve) are also important indicators for evaluating the predictor.

# RESULTS AND DISCUSSION

## ALLFEATURE Python Tool and webserver

ALLFEATURE Python pipeline tool can be implemented in both Linux and Windows operating system. The download link and tutorial can be found in <https://github.com/ashinandjay/FeatureSelection>

The Webserver is in developing and will be released to public soon.

## Application of ALLFEATURE

Our user-friendly tool will generate feature vectors from raw sequencing data in FASTA format, apply feature selection method and construct predicting model based on user’s choice. For testing the usage of our pipeline tool, three prediction tasks were performed for DNA, RNA and protein sequences, respectively. In this paper, we use DNA enhancers, RNA N6-methyladenosine sites and protein-protein interactions datasets to evaluate the validation of models. These prediction performances are comparable and even more effective than the state-of-the-art approaches, which indicate our proposed python package are competitive for analysis of biological sequences.

## Task 1: DNA enhancers prediction

DNA enhancers data plays an important role for analyzing gene expression and to identify enhancers is a challenging task [120]. The dataset contains 1484 enhancer samples and 1484 non-enhancer samples. We applied five DNA feature extraction methods: PSTNP, RCKmer, pseDNC, TNC and MonoKGap to construct predictors. In addition, we also found that fusing these feature descriptors as one mixed descriptor can effectively represent the information of the dataset and improve classification performance. The highest classification accuracies of 13 predictors are shown in **Table 6.**

**Table 6**. Predicting accuracies of different feature descriptors

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Feature Extraction Methods** | **PSTNP** | **RCKmer** | **pseDNC** | **TNC** | **MonoKGAP** | **Five Descriptors Fusion** |
| **Highest Classification Accuracy of 13 Predictors** |  |  |  |  |  |  |

According to Table 6, we used fused feature vector to implement the step of feature selection. All of 20 Feature Selection methods are applied and these selected feature vectors are used for constructing predicting models to find the best one. Based on our observation and comparison, the selected feature vector using Extra-Tree feature selection method can achieve better prediction performance, **Figure 2**. And the execution time of modeling is significantly reduced compare to the original feature vector, **Table 7**.

**Table 7**. Comparison of number of features and predicting modeling execution time

|  |  |  |
| --- | --- | --- |
|  | **Fused feature vector** | **Selected feature vector based Extra-Tree** |
| **Number of Features** | **1296** | **50** |
| **Execution Time** | **34m 42s** | **4m 5s** |

D:\feature_selection_python_package\正文\图片1.emf

**Figure 2.** The boxplot of classification accuracies (A) and ROC curves (B) of DNA enhancers using various classifiers with Extra-Tree feature selection method. (A) 13 classifiers all achieve satisfactory accuracy, and SVM, DNN, RNN obtain superior performance than other classifiers. (B) The ROC curves of 13 classifier indicate DNN and RNN achieved better results.

## Task 2: RNA N6-methyladenine sites prediction

N6-methyladenosine (m6A) refers to methylation of the adenosine nucleotide acid at the nitrogen-6 position. It is highly related to series of biological processes, such as splicing events, mRNA exporting, nascent mRNA synthesis, nuclear translocation and translation process [17]. Therefore, constructing an effective predicting model to identify RNA N6-methyladenine sites becomes an essential task. In this paper, the m6A dataset contains 2614 sequences, where 1307 represents true methyladenosine sites, and the remaining 1307 are false methyladenosine sites. Instead of using single feature extraction method, the fused feature vector shows a better classification performance, **Table 8**. We used pseDNC, PSTNP, Kmer, binary and DNC five feature extraction methods to construct initial feature vectors. In order to evaluate the effectiveness of feature selection, we applied all feature selection method to determine optimal feature subset. The feature subset using ReliefF feature selection method display better predictions, **Figure 3**.

**Table 8**. Predicting accuracies of different feature descriptors

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Feature Extraction Methods** | **PSTNP** | **Kmer** | **pseDNC** | **DNC** | **Binary** | **Five Descriptors Fusion** |
| **Highest Classification Accuracy of 13 Predictors** |  |  |  |  |  |  |

**Table 9**. Comparison of number of features and predicting modeling execution time

|  |  |  |
| --- | --- | --- |
|  | **Fused feature vector** | **Selected feature vector based Extra-Tree** |
| **Number of Features** | **182** | **50** |
| **Execution Time** | **6m 13s** | **4m 26s** |

D:\feature_selection_python_package\正文\图片2.emf

**Figure 3.** The boxplot accuracies (A) and ROC curves (B) under different classifiers on RNA N6-methyladenine sites dataset via ReliefF feature selection. (A) The boxplot of 13 classifiers and deep learning methods achieve better performance and the KNN is the worst. (B) The ROC curves of 13 classifier and DNN, CNN and RNN obtain the best prediction performance.

## Task 3: Protein-protein interactions prediction, protein site prediction

The analysis of protein-protein interactions (PPIs) can help to understand the protein function, construct the complete interactome and study the signaling pathways. The limitations of experiments lead to the development of computational prediction methods. These approaches can help us to encode, integrate and predict the PPIs. In this section, we use *S. cerevisae* collected by Guo et al to evaluate ALLFEATURE, and *S. cerevisae* includes 5594 PPIs samples and 5594 non-PPIs samples. For feature extraction, we fused CTDC, CTDT, CTDD, EBGW, Geary, PseAAC, PsePSSM, abPSSM to obtain the feature representation information. After comparing all selected feature vectors’ predicting performance. The XXX feature selection method show a better performance, **Figure 4**.

**Table 9**. Predicting accuracies of different feature descriptors

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Feature Extraction Methods** |  |  |  |  |  |  |
| **Highest Classification Accuracy of 13 Predictors** |  |  |  |  |  |  |

**Table 10**. Comparison of number of features and predicting modeling execution time

|  |  |  |
| --- | --- | --- |
|  | **Fused feature vector** | **Selected feature vector based Extra-Tree** |
| **Number of Features** |  |  |
| **Execution Time** |  |  |

# CONCLUSION

With the rapid increase of DNA, RNA and protein sequence data, the analysis and process of biological sequence is urgently needed. In this paper, we introduce an intuitive and comprehensive Python package and web server called ALLFEATURE to perform steps of feature extraction, feature selection, dimensionality reduction and model construction to predict the sequence structure and function of unseen samples. ALLFEATURE for the first time integrated 20 types of feature selection methods and 16 kinds of dimensionality reduction approaches to deal with dimensionality disaster and prevent overfitting issues. ALLFEATURE also offers 10 popular classifiers and 3 deep learning frameworks to satisfy users’ need. The tool will generate visible results to provide user clear idea to compare and select the best classifier. For further test the validity of ALLFEATURE, we perform three predicting tasks: DNA enhancers, RNA N6-methyladenine sites and protein-protein interactions prediction. Integrated feature selection and dimensionality reduction methods reduce as much as 80% modeling time. The classification performances indicate ALLFEAURE is an effective and accurate biological sequencing pipeline tool for bioinformatics and computational biology compared with other state-of-the-art approaches.

# REFERENCE

1. Behjati, S., & Tarpey, P. S. (2013). What is next generation sequencing? *Archives of Disease in Childhood - Education & Practice Edition*, *98*(6), 236–238. doi: 10.1136/archdischild-2013-304340.
2. Cost of Next-Generation Sequencing. Retrieved from https://www.illumina.com/science/technology/next-generation-sequencing/beginners/ngs-cost.html.
3. GenBank and WGS Statistics. Retrieved from https://www.ncbi.nlm.nih.gov/genbank/statistics/.
4. Garber, M., et al., *Computational methods for transcriptome annotation and quantification using RNA-seq.* Nature methods, 2011. **8**(6): p. 469-477.
5. Wang, Z., M. Gerstein, and M. Snyder, *RNA-Seq: a revolutionary tool for transcriptomics.* Nature reviews genetics, 2009. **10**(1): p. 57-63.
6. Ozsolak, F. and P.M. Milos, *RNA sequencing: advances, challenges and opportunities.* Nature reviews genetics, 2011. **12**(2): p. 87-98.
7. Prince, M.E., et al., *Identification of a subpopulation of cells with cancer stem cell properties in head and neck squamous cell carcinoma.* Proc Natl Acad Sci U S A, 2007. **104**(3): p. 973-8.
8. Navin, N., et al., *Tumour evolution inferred by single-cell sequencing.* Nature, 2011. **472**(7341): p. 90-4.
9. Xu, X., et al., *Single-cell exome sequencing reveals single-nucleotide mutation characteristics of a kidney tumor.* Cell, 2012. **148**(5): p. 886-95.
10. D’Agaro, E. (2017). NGS genome annotation profiling using data analysis workflows. *Journal of Biotechnology*, *256*. doi: 10.1016/j.jbiotec.2017.06.039.
11. Libbrecht, M. W., & Noble, W. S. (2015). Machine learning applications in genetics and genomics. *Nature Reviews Genetics*, *16*(6), 321–332. doi: 10.1038/nrg3920.
12. Karchin, R., Karplus, K., & Haussler, D. (2002). Classifying G-protein coupled receptors with support vector machines. *Bioinformatics*, *18*(1), 147–159. doi: 10.1093/bioinformatics/18.1.147.
13. Cai, C. (2003). SVM-Prot: web-based support vector machine software for functional classification of a protein from its primary sequence. *Nucleic Acids Research*, *31*(13), 3692–3697. doi: 10.1093/nar/gkg600.
14. Feng, P., Yang, H., Ding, H., Lin, H., Chen, W., & Chou, K.-C. (2019). iDNA6mA-PseKNC: Identifying DNA N6-methyladenosine sites by incorporating nucleotide physicochemical properties into PseKNC. *Genomics*, *111*(1), 96–102. doi: 10.1016/j.ygeno.2018.01.005.
15. Sasnauskas, G. (2018). DNA binding domain of restriction endonuclease McrBC in complex with N4-methylcytosine DNA. doi: 10.2210/pdb6gcf/pdb.
16. Lynch, S., & Kool, E. (2015). N6-Methyladenosine RNA. doi: 10.2210/pdb2mvs/pdb.
17. Han, L. Y. (2004). Prediction of RNA-binding proteins from primary sequence by a support vector machine approach. *Rna*, *10*(3), 355–368. doi: 10.1261/rna.5890304.
18. Lee, J. S., & Ondrechen, M. J. (2011). Electrostatic Properties for Protein Functional Site Prediction. *Protein Function Prediction for Omics Era*, 183–196. doi: 10.1007/978-94-007-0881-5\_10.
19. Fischer, D., & Eisenberg, D. (2008). Protein fold recognition using sequence-derived predictions. *Protein Science*, *5*(5), 947–955. doi: 10.1002/pro.5560050516.
20. Yan, K., Xu, Y., Fang, X., Zheng, C., & Liu, B. (2017). Protein fold recognition based on sparse representation based classification. *Artificial Intelligence in Medicine*, *79*, 1–8. doi: 10.1016/j.artmed.2017.03.006.
21. Han, M., Song, Y., Qian, J., & Ming, D. (2018). Sequence-based prediction of physicochemical interactions at protein functional sites using a function-and-interaction-annotated domain profile database. *BMC Bioinformatics*, *19*(1). doi: 10.1186/s12859-018-2206-2.
22. Lo, S. L., Cai, C. Z., Chen, Y. Z., & Chung, M. C. M. (2005). Effect of training datasets on support vector machine prediction of protein-protein interactions. *Proteomics*, *5*(4), 876–884. doi: 10.1002/pmic.200401118.
23. Constructing Reliable Protein-Protein Interaction (PPI) Networks. (2017). *Computational Prediction of Protein Complexes from Protein Interaction Networks*, 15. doi: 10.1145/3064650.3064653.
24. Feature Extraction and Classification. (2014). *Medical Diagnosis Using Artificial Neural Networks*, 159–169. doi: 10.4018/978-1-4666-6146-2.ch011.
25. Supplemental Information 3: Performance of XGBoost predictive models. (n.d.). doi: 10.7717/peerj.3230/supp-3.
26. Qi, X., Silvestrov, S., & Nazir, T. (2017). Data classification with support vector machine and generalized support vector machine. doi: 10.1063/1.4972718.
27. Kramer, O. (2013). K-Nearest Neighbors. *Dimensionality Reduction with Unsupervised Nearest Neighbors Intelligent Systems Reference Library*, 13–23. doi: 10.1007/978-3-642-38652-7\_2.
28. Deep Learning in Python: Preventing Overfitting. (2019). doi: 10.4135/9781526493453.
29. Liu, B., Liu, F., Fang, L., Wang, X., & Chou, K.-C. (2014). repDNA: a Python package to generate various modes of feature vectors for DNA sequences by incorporating user-defined physicochemical properties and sequence-order effects. *Bioinformatics*, *31*(8), 1307–1309. doi: 10.1093/bioinformatics/btu820.
30. Liu, B., Wu, H., & Chou, K.-C. (2017). Pse-in-One 2.0: An Improved Package of Web Servers for Generating Various Modes of Pseudo Components of DNA, RNA, and Protein Sequences. *Natural Science*, *09*(04), 67–91. doi: 10.4236/ns.2017.94007.
31. Muhammod, R., Ahmed, S., Farid, D. M., Shatabda, S., Sharma, A., & Dehzangi, A. (2019). PyFeat: a Python-based effective feature generation tool for DNA, RNA and protein sequences. *Bioinformatics*, *35*(19), 3831–3833. doi: 10.1093/bioinformatics/btz165.
32. Li, Z. R., Lin, H. H., Han, L. Y., Jiang, L., Chen, X., & Chen, Y. Z. (2006). PROFEAT: a web server for computing structural and physicochemical features of proteins and peptides from amino acid sequence. *Nucleic Acids Research*, *34*(Web Server). doi: 10.1093/nar/gkl305.
33. Chen, Z., Zhao, P., Li, F., Leier, A., Marquez-Lago, T. T., Wang, Y., … Song, J. (2018). iFeature: a Python package and web server for features extraction and selection from protein and peptide sequences. *Bioinformatics*, *34*(14), 2499–2502. doi: 10.1093/bioinformatics/bty140.
34. Chen, Z., Zhao, P., Li, F., Marquez-Lago, T. T., Leier, A., Revote, J., … Song, J. (2019). iLearn: an integrated platform and meta-learner for feature engineering, machine-learning analysis and modeling of DNA, RNA and protein sequence data. *Briefings in Bioinformatics*. doi: 10.1093/bib/bbz041.
35. Liu, B., Gao, X., & Zhang, H. (2019). BioSeq-Analysis2.0: an updated platform for analyzing DNA, RNA and protein sequences at sequence level and residue level based on machine learning approaches. *Nucleic Acids Research*. doi: 10.1093/nar/gkz740.
36. Ketkar, N. (2017). Introduction to Deep Learning. *Deep Learning with Python*, 1–5. doi: 10.1007/978-1-4842-2766-4\_1.
37. Deep Learning in Python: Different Types of Deep Learning Network. (2019). doi: 10.4135/9781526493439.
38. Introduction to Feature and Gene Selection. (2011). *Feature Selection and Ensemble Methods for Bioinformatics*, 117–122. doi: 10.4018/978-1-60960-557-5.ch008.
39. An Introduction to the Chi-Square Test. (2017). doi: 10.4135/9781473980525.
40. Wang, J., Shan, G., Duan, X., & Wen, B. (2011). Improved SVM-RFE feature selection method for multi-SVM classifier. *2011 International Conference on Electrical and Control Engineering*. doi: 10.1109/iceceng.2011.6058060.
41. Missing-Value, M.-V. (n.d.). The Lasso for Linear Models. *Statistical Learning with Sparsity*, 22–43. doi: 10.1201/b18401-4.
42. Applications Of Pearson Correlation To Measurement Theory. (n.d.). *Correlation and Regression*, 67–96. doi: 10.4135/9781412983815.n4.
43. Yang, F., Cheng, W., Dou, R., & Zhou, N. (2011). An improved feature selection approach based on ReliefF and Mutual Information. *International Conference on Information Science and Technology*. doi: 10.1109/icist.2011.5765246.
44. Komorowski, J., Kurzejamski, G., & Sarwas, G. (2019). DeepBall: Deep Neural-Network Ball Detector. *Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*. doi: 10.5220/0007348902970304.
45. Convolutional Neural Network. (2018). *Encyclopedia of Social Network Analysis and Mining*, 418–418. doi: 10.1007/978-1-4939-7131-2\_100208.
46. Gelenbe, E. (1992). Learning in the Recurrent Random Neural Network. *Neural Networks*, 1–12. doi: 10.1016/b978-0-444-89330-7.50004-1.
47. Noble, W.S., Kuehn, S., Thurman, R., Yu, M., and Stamatoyannopoulos, J. (2005) Predicting the in vivo signature of human gene regulatory sequences. Bioinformatics, 21, i338-343.
48. Lee, D., Karchin, R. and Beer, M.A. (2011) Discriminative prediction of mammalian enhancers from DNA sequence. Genome research, 21, 2167-2180.
49. Gupta, S., Dennis, J., Thurman, R.E., Kingston, R., Stamatoyannopoulos, J. and Noble, W.S. (2008) Predicting human nucleosome occupancy from primary sequence. PLoS computational biology, 4, e1000134.
50. Chen, W., Feng, P.M., Lin, H. and Chou, K.C. (2013) iRSpot-PseDNC: identify recombination spots with pseudo dinucleotide composition. Nucleic Acids Res, **41**, e68.
51. Guo, S.H., Deng, E.Z., Xu, L.Q., Ding, H., Lin, H., Chen, W. and Chou, K.C. (2014) iNuc-PseKNC: a sequence-based predictor for predicting nucleosome positioning in genomes with pseudo k-tuple nucleotide composition. Bioinformatics, **30**, 1522-1529.
52. Dong, Q., Zhou, S. and Guan, J. (2009) A new taxonomy-based protein fold recognition approach based on autocross-covariance transformation. Bioinformatics, 25, 2655-2662.
53. Guo Y., Yu L., Wen Z. and Li, M. (2008) Using support vector machine combined with auto covariance to predict protein-protein interactions from protein sequences. Nucleic Acids Res. 36, 3025-3030.
54. Bhasin, M. and Raghava, G.P. (2004) Classification of nuclear receptors based on amino acid composition and dipeptide composition. J. Biol. Chem., 279, 23262-23266.
55. Chen, K., Jiang Y., Du, L., Kurgan, L. and (2009) Prediction of integral membrane protein type by collocated hydrophobic amino acid pairs. J. Comput. Chem., 30, 163-172.
56. Chen, K., Kurgan, L. and Rahbari, M. (2007) Prediction of protein crystallization using collocation of amino acid pairs. Biochem. Biophys. Res. Commun., 355, 764-769.
57. Chen, K., Kurgan, L.A. and Ruan, J. (2007) Prediction of flexible/rigid regions from protein sequences using k-spaced amino acid pairs. BMC Struct. Biol., 7, 25.
58. Lee, T.Y., Lin, Z.Q., Hsieh, S.J., Bretaña, N.A. and Lu, C.T. (2011) Exploiting maximal dependence decomposition to identify conserved motifs from a group of aligned signal sequences. Bioinformatics, 27, 1780-1787.
59. Shen, J., Zhang, J., Luo, X., Zhu, W., Yu, K., Chen, K., Li Y. and Jiang, H. (2007) Predicting protein-protein interactions based only on sequences information. Proc. Natl. Acad. Sci. U. S. A., 104, 4337-4341.
60. Göktepe, Y.E. and Kodaz, H. (2018) Prediction of protein-protein interactions using an effective sequence based combined method. Neurocomputing, 303, 68-74.
61. Cai, C.Z., Han, L.Y., Ji, Z.L., Chen, X., and Chen, Y. Z. (2003) SVM-Prot: Web-based support vector machine software for functional classification of a protein from its primary sequence. Nucleic Acids Res., 31, 3692-3697.
62. Cai, C.Z., Han, L.Y., Ji, Z.L. and Chen, Y.Z. (2004) Enzyme family classification by support vector machines. Proteins, 55, 66-76.
63. Dubchak, I., Muchnik, I., Holbrook, S.R. and Kim, S.H. (1995) Prediction of protein folding class using global description of amino acid sequence. Proc. Natl. Acad. Sci. U. S. A., 92, 8700-8704.
64. Zhang, Z.H., Wang, Z.H., Zhang, Z.R. and Wang, Y. X. (2006) A novel method for apoptosis protein subcellular localization prediction combining encoding based on grouped weight and support vector machine. FEBS Lett., 580, 6169-6174.
65. Guo Y., Yu L., Wen Z. and Li, M. (2008) Using support vector machine combined with auto covariance to predict protein-protein interactions from protein sequences. Nucleic Acids Res. 36, 3025-3030.
66. Horne, D.S. (1988) Prediction of protein helix content from an autocorrelation analysis of sequence hydrophobicities. Biopolymers, 27, 451-477.
67. Sokal, R.R. and Thomson, B.A. (2006) Population structure inferred by local spatial autocorrelation: an example from an Amerindian tribal population. Am. J. Phys. Anthropol, 129, 121-131.
68. Feng, Z.P. and Zhang, C.T. (2000) Prediction of membrane protein types based on the hydrophobic index of amino acids. J. Protein Chem., 19, 269-275.
69. Lin, Z. and Pan, X.M. (2001) Accurate prediction of protein secondary structural content. J. Protein Chem., 20, 217-220.
70. Du, X., Sun, S., Hu, C., Yao, Y., Yan, Y. and Zhang Y. (2017) DeepPPI: boosting prediction of protein-protein interactions with deep neural networks. J. Chem. Inf. Model, 57, 1499-1510.
71. Chou K.C. (2001) Prediction of protein cellular attributes using pseudo-amino acid composition. Proteins, 43, 246-255.
72. Chou, K.C. (2005) Using amphiphilic pseudo amino acid composition to predict enzyme subfamily classes. Bioinformatics, 21, 10-19.
73. Liu, T., Geng, X., Zheng, X., Li, R. and Wang, J. (2012) Accurate prediction of protein structural class using auto covariance transformation of PSI-BLAST profiles. Amino acids, 42, 2243-2249.
74. Liu, T., Zheng, X. and Wang, J. (2010) Prediction of protein structural class for low-similarity sequences using support vector machine and PSI-BLAST profile, Biochimie, 92, 1330-1334.
75. Saini, H. Raicar, G., Lal, S.P. Dehzangi, A., Imoto, S. and Sharma, A. (2016) Protein fold recognition using genetic algorithm optimized voting scheme and profile bigram. J Softw., 11, 756–767.
76. Dong, Q., Zhou, S. and Guan, J. (2009) A new taxonomy-based protein fold recognition approach based on autocross-covariance transformation. Bioinformatics, 25, 2655-2662.
77. Shen, H.B., and Chou, K.C. (2007) Nuc-PLoc: a new web-server for predicting protein subnuclear localization by fusing PseAA composition and PsePSSM. Protein Eng. Des. Sel. 20, 561.
78. Jeong, J.C., Lin, X. and Chen, X.W. (2011) On position-specific scoring matrix for protein function prediction. IEEE/ACM transactions on computational biology and bioinformatics, 8, 308-315.
79. Rayhan, F., Ahmed S., Shatabda S., Farid D.M., Mousavian Z., Dehzangi A. and Rahman M.S. (2017) iDTI-ESBoost: identification of drug target interaction using evolutionary and structural features with boosting. Sci. Rep., 7, 17731.
80. Libal, U. (2011). Feature selection for pattern recognition by LASSO and thresholding methods - a comparison. *2011 16th International Conference on Methods & Models in Automation & Robotics*. doi: 10.1109/mmar.2011.6031338.
81. Teisseyre, P. (2017). CCnet: Joint multi-label classification and feature selection using classifier chains and elastic net regularization. *Neurocomputing*, *235*, 98–111. doi: 10.1016/j.neucom.2017.01.004.
82. Maldonado, S., Montoya, R., & López, J. (2016). Embedded heterogeneous feature selection for conjoint analysis: A SVM approach using L1 penalty. *Applied Intelligence*, *46*(4), 775–787. doi: 10.1007/s10489-016-0852-5.
83. Liu, H., & Setiono, R. (n.d.). Chi2: feature selection and discretization of numeric attributes. *Proceedings of 7th IEEE International Conference on Tools with Artificial Intelligence*. doi: 10.1109/tai.1995.479783.
84. Tree-Structured Classifier. (n.d.). *SpringerReference*. doi: 10.1007/springerreference\_66000.
85. Li, J. (2018). Analysis of the Rise and Fall of International Futures Based on Xgboost Algorithm. *Finance*, *08*(05), 211–220. doi: 10.12677/fin.2018.85025.
86. Test, E., Zigic, L., & Kecman, V. (2013). Feature ranking using Gini index, scatter ratios, and nonlinear SVM RFE. *2013 Proceedings of IEEE Southeastcon*. doi: 10.1109/secon.2013.6567380.
87. Introduction to the Logistic Regression Model. (2005). *Applied Logistic Regression*, 1–30. doi: 10.1002/0471722146.ch1.
88. Lajevardi, S. M., & Hussain, Z. M. (2009). Feature selection for facial expression recognition based on mutual information. *Exhibition*. doi: 10.1109/ieeegcc.2009.5734265.
89. Zhao, X.-M. (2013). Maximum Relevance/Minimum Redundancy (MRMR). *Encyclopedia of Systems Biology*, 1191–1192. doi: 10.1007/978-1-4419-9863-7\_432.
90. Zhao, X.-M. (2013). Maximum Relevance/Minimum Redundancy (MRMR). *Encyclopedia of Systems Biology*, 1191–1192. doi: 10.1007/978-1-4419-9863-7\_432.
91. Liu, H., & Ditzler, G. (2017). A fast information-theoretic approximation of joint mutual information feature selection. *2017 International Joint Conference on Neural Networks (IJCNN)*. doi: 10.1109/ijcnn.2017.7966441.
92. Shirzad, M. B., & Keyvanpour, M. R. (2015). A feature selection method based on minimum redundancy maximum relevance for learning to rank. *2015 AI & Robotics (IRANOPEN)*. doi: 10.1109/rios.2015.7270735.
93. Wang, H., Yan, S., Xu, D., Tang, X., & Huang, T. (2007). Trace Ratio vs. Ratio Trace for Dimensionality Reduction. *2007 IEEE Conference on Computer Vision and Pattern Recognition*. doi: 10.1109/cvpr.2007.382983.
94. Test, E., Zigic, L., & Kecman, V. (2013). Feature ranking using Gini index, scatter ratios, and nonlinear SVM RFE. *2013 Proceedings of IEEE Southeastcon*. doi: 10.1109/secon.2013.6567380.
95. Zhao, Z. A., & Liu, H. (2011). Spectral Feature Selection for Data Mining. doi: 10.1201/b11426.
96. A Hybrid Feature Selection Method Based On Fisher Score And Genetic Algorithm. (2016). *Journal of Mathematical Sciences: Advances and Applications*, *37*(1), 51–78. doi: 10.18642/jmsaa\_7100121627.
97. Polat, K., & Güneş, S. (2009). A new feature selection method on classification of medical datasets: Kernel F-score feature selection. *Expert Systems with Applications*, *36*(7), 10367–10373. doi: 10.1016/j.eswa.2009.01.041.
98. Suryavanshi, A. S. (2016). A Survey Paper on Modified Approach for Kmeans Algorithm. *International Journal of Emerging Trends in Science and Technology*. doi: 10.18535/ijetst/v3i02.04.
99. Priam, R. (2018). Symmetric Generative Methods and tSNE: A Short Survey. *Proceedings of the 13th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*. doi: 10.5220/0006684303560363.
100. A Comparative Studyon Kernel PCA and PCA Methods for Face Recognition. (2016). *International Journal of Science and Research (IJSR)*, *5*(5), 1844–1847. doi: 10.21275/v5i5.nov163830.
101. Sun, X., & Lu, Y. (2016). Locally Linear Embedding based on Rank-order Distance. *Proceedings of the 5th International Conference on Pattern Recognition Applications and Methods*. doi: 10.5220/0005658601620169.
102. Garnadi, A. D. (2017). Kompresi Citra Menggunakan Truncated Singular Value Decomposition (TSVD), Sebuah Eksplorasi Numerik. doi: 10.31227/osf.io/uqrwe.
103. Larsen, J. S., & Clemmensen, L. K. H. (2015). Non-negative Matrix Factorization for Binary Data. *Proceedings of the 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management*. doi: 10.5220/0005614805550563.
104. Chen, Y. (n.d.). Multi-dimensional scaling and MODELLER based evolutionary algorithms for protein model refinement. doi: 10.32469/10355/43046.
105. R., G. (2012). Introduction: Independent Component Analysis. *Independent Component Analysis for Audio and Biosignal Applications*. doi: 10.5772/52324.
106. Dhir, C. S., Iqbal, N., & Lee, S.-Y. (2007). Efficient feature selection based on information gain criterion for face recognition. *2007 International Conference on Information Acquisition*. doi: 10.1109/icia.2007.4295788.
107. Data Reduction: Factor Analysis and Cluster Analysis. (n.d.). *Statistical Methods for Geography*, 193–209. doi: 10.4135/9781849209953.n10.
108. Clustering: Practice - Agglomerative Clustering. (2018). doi: 10.4135/9781526469205.
109. Fong, H., Zhang, Q., & Wei, S. (2007). Image Reconstruction Based on Sub-Gaussian Random Projection. *Fourth International Conference on Image and Graphics (ICIG 2007)*. doi: 10.1109/icig.2007.28.
110. Kim, Y., & Toh, K.-A. (2008). Sparse random projection for efficient cancelable face feature extraction. *2008 3rd IEEE Conference on Industrial Electronics and Applications*. doi: 10.1109/iciea.2008.4582897.
111. Yu, J., Huang, Y., & Liu, C. (2017). Classification method for uncertain data based on sparse denoising autoencoder neural network. *Advances in Modelling and Analysis B*, *60*(1), 210–223. doi: 10.18280/ama\_b.600113.
112. Gualtierotti, A. F. (2015). Likelihoods for Signal Plus Gaussian Noise Versus Gaussian Noise. *Detection of Random Signals in Dependent Gaussian Noise*, 1087–1160. doi: 10.1007/978-3-319-22315-5\_17.
113. Hsu, W.-N., & Glass, J. (2018). Scalable Factorized Hierarchical Variational Autoencoder Training. *Interspeech 2018*. doi: 10.21437/interspeech.2018-1034.
114. Al-Timemy, A. H. (2017). Boosting-based decision tree for improved screening of vibroarthrographic signals. *2017 Fourth International Conference on Advances in Biomedical Engineering (ICABME)*. doi: 10.1109/icabme.2017.8167551.
115. Grossmann, E. (n.d.). AdaTree: Boosting a Weak Classifier into a Decision Tree. *2004 Conference on Computer Vision and Pattern Recognition Workshop*. doi: 10.1109/cvpr.2004.296.
116. Padmanaban, H. (n.d.). Comparative Analysis of Naive Bayes and Tree Augmented Naive Bayes Models. doi: 10.31979/etd.n7jg-e3uh.