APPLYING MACHINE LEARNING ALGORITHMS AND DEVELOPING COMPUTATIONAL TOOLS FOR ANALYZING DNA, RNA, PROTEIN SEQUENCING AND EMR DATA

BY

SHAOPENG GU

A thesis submitted in partial fulfillment of the requirements for the

Master of Science

Major in Mathematics – Statistics Specialization

South Dakota State University

2019

APPLYING MACHINE LEARNING ALGORITHMS AND DEVELOPING COMPUTATIONAL TOOLS FOR ANALYZING DNA, RNA, PROTEIN SEQUENCING AND EMR DATA

SHAOPENG GU

This thesis is approved as a creditable and independent investigation by a candidate for the Master of Science in Mathematics – Statistics Specialization degree and is acceptable for meeting the thesis requirements for this degree. Acceptance of this thesis does not imply that the conclusions reached by the candidate are necessarily the conclusions of the major department.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Xijin Ge, PhD.

Thesis Advisor Date

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Kurt Cogswell, PhD.

Head, Mathematics & Statistics Date

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Dean, Graduate School Date

ACKNOWLEDGEMENTS

At beginning, I would like to give my endless thanks to my advisor, Dr. Xijin Ge, who encouragingly guided me through this research topic. Thanks for having me as your graduate student, lab member and giving a lot of valuable suggestions to complete this work. As an experienced and professional researcher in data science and bioinformatics area, his brilliant insights help me solve many detailed problems. In addition, Dr. Ge provides me such excellent opportunities to learn and develop webserver, especially for laboratory information management system. In the past year, Dr. Ge sent many tutorials and lent valuable textbooks of bioinformatics to help me understand materials well.

Besides my advisor, I would like to thank my former advisor, Dr. Qin Ma for guiding me to bioinformatics area. As a student who graduated from Electrical Engineering major, lack of biology knowledge is my top challenge during my master’s study. In the past two years, Dr. Ma taught me many things not only relate to research skills but also relate to my future career. Thanks to Dr. Ma’s patience and encouragement, I obtain enough time and passion to move forward for the degree.

I also want to thank my co-workers for this research topic: Dr. Bin Yu and his student, Cheng Chen. Dr. Yu brought many ideas and suggestion for enhancing the quality of my works. Cheng Chen help me to understand biological information of DNA and RNA sequencing data through feature extraction process.

I would like to give a special thanks to the department head, Dr. Cogswell for helping me review all materials covered by core courses of the degree.

Thanks, my committee, Dr. Rhoda Burrows. Thank you for being my committees, thanks for your precious time and valuable comments.

My sincere thanks also go the all BMBL members. Dr. Adam McDermaid provides many insightful pieces of advice regarding background of bioinformatics and future career opportunities. Anjun Ma, one of my best friends, taught me a lot of biological knowledge. Cankun Wang, Minxuan Sun, Yiran Zhang and Yirong Wang, thank you all for your valuable comments.

Thanks to Dr. Jing Zhao, our cooperation in analyzing medical data guided me to healthcare area. Without this valuable opportunity, getting a job offer from healthcare system will hardly happen.

Finally, infinite gratitude to my sweet family. My wife, Huayinyin Wang, thanks for your accompanying and strong supports for my daily life. My daughter, Caroline, you brought my life more colors.

CONTENTS

[LISTS OF FIGURES viii](#_Toc21013155)

[LIST OF TABLES ix](#_Toc21013156)

[ABSTRACT 10](#_Toc21013157)

[CHAPTER 1: Introduction for Sequencing Data Analysis 13](#_Toc21013158)

[**1.1 Gene Expression** 13](#_Toc21013159)

[**1.2 Machine Learning in Sequencing Data Analysis** 14](#_Toc21013160)

[**1.3 Feature Extraction of Sequencing Data** 15](#_Toc21013161)

[**1.4 Feature Selection** 16](#_Toc21013162)

[**1.5 Dimensionality Reduction** 16](#_Toc21013163)

[**1.6 Models Construction** 17](#_Toc21013164)

[**1.7 Sequencing Data Analysis Tool** 18](#_Toc21013165)

[CHAPTER 2: ALLFUTARES – An Integrated Python Package for DNA, RNA and Protein Sequencing Data Analysis 20](#_Toc21013166)

[**2.1 Overall Design of ALLFEATURE** 20](#_Toc21013167)

[**2.2 Detailed Methods in ALLFEATURE** 21](#_Toc21013168)

[2.2.1 Feature Extraction 22](#_Toc21013169)

[2.2.2 Feature Selection and Dimensionality Reduction 25](#_Toc21013170)

[2.2.3 Models Construction 28](#_Toc21013171)

[2.2.4 Cross-validation and Models Evaluation 28](#_Toc21013172)

[**2.3 Application of ALLFEATURE** 29](#_Toc21013173)

[2.3.1 DNA N6-methyladenine Sites Prediction 30](#_Toc21013174)

[2.3.2 RNA N6-methyladenine Sites Prediction 35](#_Toc21013175)

[**2.4 Summary and Conclusion** 38](#_Toc21013176)

[CHAPTER 3: Introduction for EMR Data and CKD 39](#_Toc21013177)

[**3.1 Chronic Kidney Disease and eGFR** 39](#_Toc21013178)

[**3.2 Machine Learning in EMR Data Analysis** 41](#_Toc21013179)

[CHAPTER 4: Predicting Outcomes of Chronic Kidney Disease from EMR Data Based on Random Forest Regression 43](#_Toc21013180)

[**4.1 Methods** 43](#_Toc21013181)

[4.1.1 Data Acquisition 43](#_Toc21013182)

[4.1.2 Data Pre-processing 44](#_Toc21013183)

[**4.2 Construction of Random Forest Regression Model** 46](#_Toc21013184)

[**4.3 Assessment of model performance** 47](#_Toc21013185)

[4.3.1 Goodness-of-fit 47](#_Toc21013186)

[4.3.2 Discrimination 47](#_Toc21013187)

[**4.4 Results** 47](#_Toc21013188)

[**4.5 Conclusion and Discussion** 50](#_Toc21013189)

[REFERENCES 51](#_Toc21013190)

[APPENDIX 1: ALLFUEATRE Tutorial 62](#_Toc21013191)

[APPENDIX 2: DNA and RNA Feature Extraction Methods Description 67](#_Toc21013192)

[APPENDIX 3: Protein Feature Extraction Methods Description 76](#_Toc21013193)

[APPENDIX 4: Plan of Study 93](#_Toc21013194)

[APPENDIX 5: Curriculum Vitae 94](#_Toc21013195)

# LISTS OF FIGURES

[Figure 1. General designing idea of an integrated sequencing data analysis pipeline tool.. 19](#_Toc20918752)

[Figure 2. The overflow of ALLFEATURE.. 21](#_Toc20918753)

[Figure 3. The boxplot (A) and ROC curves (B) of DNA N6-methyladenine sites using various classifiers based on combined feature vectors 56](#_Toc20918754)

[Figure 4. The boxplot (A) and ROC curves (B) of DNA N6-methyladenine sites using various classifiers with ExtraTree feature selection method. 58](#_Toc20918755)

[Figure 5. The boxplot (A) and ROC curves (B) of DNA N6-methyladenine sites using various classifiers with Fisher Score feature selection method. 59](#_Toc20918756)

[Figure 6. The boxplot (A) and ROC curves (B) of DNA N6-methyladenine sites using various classifiers with MRMR feature selection method. 60](#_Toc20918757)

[Figure 7. The boxplot (A) and ROC curves (B) under different classifiers on RNA N6-methyladenine sites based on combined feature vectors 61](#_Toc20918758)

[Figure 8. The boxplot (A) and ROC curves (B) under different classifiers on RNA N6-methyladenine sites dataset via SVM-RFE feature selection. 63](#_Toc20918759)

# LIST OF TABLES

[Table 1. List of 16 DNA feature extraction methods and 12 RNA feature extraction methods 22](#_Toc21012349)

[Table 2. List of 32 Protein feature extraction methods and their description 24](#_Toc21012350)

[Table 3. Feature Selection and Dimensionality Reduction Methods 26](#_Toc21012351)

[Table 4. Predictor and covariate data type breakdown 43](#_Toc21012352)

[Table 5. Hyperparameters used in the Random Forest Regression for the default and optimized models. 48](#_Toc21012353)

# ABSTRACT

APPLYING MACHINE LEARNING ALGORITHMS AND DEVELOPING COMPUTATIONAL TOOLS FOR ANALYZING DNA, RNA, PROTEIN SEQUENCING AND EMR DATA

SHAOPENG GU

2019

The appearance of next generation sequencing (NGS) technology have significantly improved the quantities and qualities of biological sequences (DNA, RNA and protein). This faster and cheaper modern sequencing technology revolutionizes genomic research field and trigger explosive growth of DNA, RNA and protein sequencing data. The sequencing data contains valuable information for current biological researchers to explore. How to study the structure and function from the perspective of biological sequences is significant for biological research, disease diagnosis, biotechnology and many other areas. However, analyzing sequencing data with human inspection alone is hardly possible. Thus, indispensable computational tools are widely used in sequencing data analysis.

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. The analysis of biological sequence can be mainly regarded as binary and multi-class prediction tasks, including DNA N6-methyladenosine site, DNA N4-methylcytosine site, RNA N6-methyladenosine site, RNA-binding protein identification, protein function site, protein fold recognition, protein-protein interaction prediction and many others.

For the process to construct predictors, the first step is feature extraction, which transform the character sequence data into numeric vectors of the same length. 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. There are many popular models can be used for researchers to analyze the structure and function of sequencing data, such as XGBoost, support vector machine (SVM) and k nearest neighbors (KNN) and etc. Inspired by this, several computational tools or web servers have been released, including PROFEAT, Pse-in-one, PyFeat, POSSUM, iFeature, BioSeq-Analysis, iLearn and BioSeq-Analysis2.0.

However, some challenges still remain in the biological sequencing data analysis. First, overfitting and time-consuming issues often exist during performing accurate prediction in machine learning. Based on our observation, extracted feature vectors from feature extraction step often displays high dimensionality. They often contain a lot of noisy features which can cause poor prediction and time-consuming issues while modeling. Secondary, most computational tools or web servers only focus on one individual step instead of integrating all functionalities for sequencing data analysis. There are some computational tools which contain multiple steps but some newer released feature selection methods and classifiers algorithms are not included. Finally, these computational tools lack deep learning methods which are widely used for classification recently.

Feature selection is the step to overcome the challenge of overfitting by only selecting those features that contribute most to the prediction and removing other noisy features. In addition, for some feature vectors with extremely high dimensionalities, dimension reduction methods are often involved in the sequencing data analysis to pre-filter vectors to reduce the time cost and improve classification performance.

Inspired by above discussion, I present a new python pipeline tool, **ALLFEATURE**, that integrated 20 feature selection methods, total of 16 dimensionality reduction methods and 13 prediction/classification models. In addition, our pipeline also contains the step of feature extraction to generate total of 60 different models of features from DNA, RNA and protein sequencing data.

Applying machine learning algorithms for Electronic medical record (EMR) data analysis is also included in this dissertation. Chronic kidney disease (CKD) is prevalent across the world and well defined by an estimated glomerular filtration rate (eGFR). The progression of kidney disease can be predicted if the future eGFR can be accurately estimated using predictive analytics. Thus, I present a prediction model of eGFR that was built using Random Forest regression. The dataset includes demographic, clinical and laboratory information from a regional primary health care clinics. The final model included eGFR, age, gender, body mass index (BMI), obesity, hypertension and diabetes, which achieved a mean coefficient of determination of 0.95. The estimated eGFRs were used to classify patients into CKD stages with high macro-averaged and micro-averaged metrics.

# CHAPTER 1: Introduction for Sequencing Data Analysis

## **1.1 Gene Expression**

The process to determine which information from gene can be used for synthesizing a functional gene product is called gene expression. Gene products are often proteins, a molecule that perform a job in the cell. But the products also can be a functional RNA in non-protein coding genes such as transfer RNA or small nuclear RNA genes. The gene expression process mainly contains two steps that are transcription and translation. The DNA sequence of a gene is transcribed into RNA in the transcription step. In translation step, the sequence of transcribed RNA is translated to protein, the amino acid sequence of a polypeptide. Therefore, the gene expression can be considered as a translation between DNA, RNA and protein that using to determine the cell functions.

There are many technologies that used to detect expression of genes include microarray and NGS. Microarray is the first high-throughput technology that introduced in the late of 1990s. With this technology, researchers are able to monitor and detect the expression of thousands of genes at the same time via printing microscope slides in defined positions that containing many known DNA sequences or genes [1]. Reference genome and transcriptome are required, however, to perform microarrays analysis. Therefore, its application limited to such organisms which having the well-sequenced genome.

Next-generation sequencing can also be called second generation sequencing which providing an advanced technology with many advantages: ultra-high throughput, speed, scalability and friendlily cost. With NGS, the duration for sequencing an entire human genome is reduced from a decade to a single day [2] and its cost dropped from $300000 to less than $1000 [3]. 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 [4]. Unlike the microarrays, NGS have 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 much lower variation. NGS is a powerful tool or platform which can sequence of thousands to millions of DNA molecules simultaneously [5]. It provides researchers a precious opportunity to rapidly sequence whole genomes and study the structure and function from the perspective of biological sequences [6, 7]. Analyzing sequencing data help researches to explore disease diagnosis [8-10], biotechnology [11] and many other areas.

## **1.2 Machine Learning in Sequencing Data Analysis**

It is impossible to analyze sequencing data by human inspection alone. In addition, the cost of wet-laboratory experiment is very high. Therefore, sequencing data analysis completely rely on computational tools. Recently, the rapidly development of machine learning provide a lot of new solutions to many difficult problems. Applying machine learning algorithms and integrating them in computational tools became a popular trends to explore structural and functional properties of DNA, RNA and protein sequencing data [12]. One of popular research interests is to consider sequencing data analysis as a two-class or multi-class prediction tasks [13, 14] including DNA N6-methyladenosine site [15], DNA N4-methylcytosine site [16], RNA N6-methyladenosine site [17], RNA-binding protein identification [18], protein function site [19], protein fold recognition [20, 21], protein-protein interaction prediction [22-24], etc. Besides popular supervised algorithms for classification and regression, unsupervised machine learning methods also can be applied to sequencing data analysis to reduce dimensionality of the data.

## **1.3 Feature Extraction of Sequencing Data**

Billions of short raw reads can be generated for each sample through NGS and stored in sequencing data files that often in FASTQ or FASTA format [25]. One popular sequencing data analysis concept is to find some relationship between information that is contained in sequencing data and its known functions or expressions. Therefore, sequencing data analysis can be considered as a classification or prediction problem that is, using gene information from sequencing data to predict one of its function or expression so that exploring their relationship.

Raw reads of sequencing data cannot directly used for classification modeling purpose. Thus, feature extraction is required to transform reads in sequencing data to mathematical or numerical data matrix as the first step of sequencing data analysis [26]. There are four main different types of feature extraction concepts include extracting features based on sequence, physicochemical property, evolution and structure. These four feature extraction concepts will focus on different characters of sequencing data and generate completely distinguished feature vectors.

## **1.4 Feature Selection**

Feature selection is one of the most popular topics in the current artificial intelligence era. With a rapidly increasing number of machine learning algorithms has been introduced and applied in multiple areas, selecting most essential features and eliminating noisy features to reach accurate and efficient performances becomes a new challenge [27]. The advantages of feature selection include efficiently reduce the machine learning algorithm training time, computational resources consuming, the complexity of models, overfitting and improve the accuracy of models by selecting appropriate features or subset from a big dataset.

Feature selection is a necessary step for constructing classification models based on DNA, RNA and protein feature vectors [28]. These extracted feature vectors often show high dimensionality, especially some protein feature vectors display extremely high dimensionality which can cause time-consuming issue and poor accuracy of prediction. Therefore, selecting those features that contribute most to functional or structural prediction is an essential step in sequencing data analysis pipeline. There are many popular and powerful feature selection algorithms can be used to overcome this challenge such as: Chi squared test [29], SVM-RFE [30], Lasso [31], Pearson correlation [32], ReliefF [33], etc.

## **1.5 Dimensionality Reduction**

Besides many supervised feature selection methods, some unsupervised dimensionality reduction such as K-means [34], PCA [35] and TSNE [36], are introduced recently. Clustering methods do not require label vectors to perform dimensionality reduction. In addition, compare to the supervised feature selection method that normally calculating and ranking feature importance based on label vectors, clustering methods focus on data itself and often perform faster execution.

Even though the objective of this work is to accurately predict a function or structure of sequencing data, unsupervised methods are required in this analysis pipeline for some feature vectors with extremely high dimensionality. Based on our observation, implementing feature selection method for those vectors will take more than 24 hours even 48 hours. Therefore, clustering methods provide a much faster method to filter the data before constructing prediction models.

## **1.6 Models Construction**

Classification or prediction is a supervised learning approach to classify new observation based on the given input data in machine learning. This dataset can have two classes (binary) like indicating whether the person is male or female. In other case, the dataset can have multiple classes’ information like indicating what color of coat the person is worn. One classification algorithm can be called as a classifier. A classifier needs to be trained at first using the dataset then it can be used for prediction. A trained classifier will provide a classification or label for same type of data.

There are some popular and well-developed classification algorithms that are widely used in many different fields, such as SVM [37], RandomForest [38], LightGBM [39], XGBoost [40], Adaboost [41] and KNN [42], etc. Every classifier has its own characters thus there is no the best classifier but only appropriate classifier. Therefore, training multiple classifiers simultaneously can help researchers to find better classifier with high accuracy for a specific dataset.

## **1.7 Sequencing Data Analysis Tool**

There are some computational tools for sequencing data analysis available in public. Some tools focus only on extracting features from one or more types of sequencing data. For instance, repDNA is a python package to only generate various types of feature vectors from DNA sequences [43]. Pse-in-one 2.0 [44], PyFeat [45] and PROFEAT [46] are generation tools for extracting different feature models from DNA, RNA, and protein sequences. To my knowledge, there are three computational tools: IFeature [47], iLeran [48] and BioSeq-Analysis2.0 [49] that integrating multiple steps for sequencing data analysis, but the integrated classifier algorithms and feature selection methods are not sufficient and updated. In addition, deep learning is very powerful computational tools for classification tasks via layer by layer learning [50]. The popular deep learning methods such as autoencoder network, deep neural network (DNN), convolutional neural network (CNN), and recurrent neural network (RNN) show convincing performances for prediction [51]. However, the packages above lack the usages of deep learning theories.

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, classification and deep learning algorithms is needed.

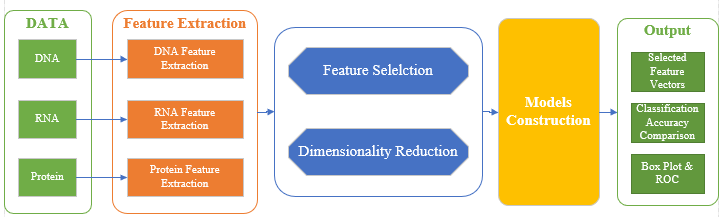
A general integrated DNA, RNA and Protein sequencing data analysis pipeline tool should contain at least four sections: feature extraction, feature selection, dimensionality reduction and model construction. The outputs of the pipeline tool should include feature vectors after the step of feature selection, comparison of classification accuracies between predictors and models evaluation, **Figure 1**.

Figure 1. General designing idea of an integrated sequencing data analysis pipeline tool. The tool will take DNA, RNA or protein sequencing data with FASTA format and then go through steps of feature extraction, feature selection, dimensionality reduction and models construction to generate outputs.

# CHAPTER 2: ALLFUTARES – An Integrated Python Package for DNA, RNA and Protein Sequencing Data Analysis

Here, I present **ALLFEATURE**, a comprehensive Python pipeline tool that integrating multiple steps to analyze DNA, RNA, and protein sequencing data: feature extraction, feature selection, dimensionality reduction and models construction.

## **2.1 Overall Design of ALLFEATURE**

**ALLFEATURE** integrated 20 feature selection methods, 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 total of 60 different models of features from DNA, RNA and protein sequencing data. The flowchart of python pipeline tool is shown in **Figure 2**. Compared with other software packages or webservers, the proposed ALLFEATURE has following advantages: (i) sufficient feature selection methods and dimension reduction methods, including regularization-based, statistic-based, information-based, tree-based and recursive feature elimination-based approaches. (ii) Ten machine learning methods and three deep learning approaches: deep neural network (DNN) [52], convolutional neural network (CNN) [53], and recurrent neural network (RNN) [54]. (iii) More abundant graphical display results, including box figure, ROC curves, etc.

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

Figure 2. The overflow of ALLFEATURE. The python package contains feature extraction, feature selection, machine learning, deep learning, and performance evaluation. The input are the DNA, RNA and protein sequences with FASTA format, or the high-dimension matrix with csv format. The outputs of pipeline will provide generated feature vectors, prediction accuracy comparison and suggestion of best model for researchers.

## **2.2 Detailed Methods in ALLFEATURE**

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

 (1)

where  represents the  residue. And the biological sequence contains the important and effective information of structure and function. How to analyze the attributes of sequence and construct the useful predictor 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 etc.; (ii) Feature selection (dimension 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.

### 2.2.1 Feature Extraction

**ALLFEATURE** directly extracts features from DNA, RNA or protein sequences based on the total of 60 different types of feature extraction methods. The step of feature extraction consists 16 feature extraction methods for DNA and 12 feature extraction methods for RNA; 32 feature extraction methods for protein sequences, which can be shown in **Table 1** and **Table 2**, respectively.

Table 1. List of 16 DNA feature extraction methods and 12 RNA feature extraction methods

|  |  |  |  |
| --- | --- | --- | --- |
| **DNA Feature Extraction Methods** | **RNA Feature Extraction Methods** | **Extraction Method Description** | |
| Kmer | Kmer | | DNA or RNA sequence are represented as the occurrence frequencies of k neighboring nucleic acids [55, 56] |
| Reverse Compliment Kmer (RCKmer) | Reverse Compliment Kmer (RCKmer) | | A variant of Kmer descriptor by removing the reverse compliment Kmer [55, 57] |
| Pseudo Dinucleotide Composition (PseDNC) | Pseudo Dinucleotide Composition (PseDNC) | | Incorporating the contiguous local sequence-order and global sequence-order information [58] |
| Pseudo k-tuple Nucleotide Composition (PseKNC) | - | | Extending the PseDNC by incorporating k-tuple nucleotide composition [59] |
| Dinucleotide Based Auto Covariance (DAC) | Dinucleotide Based Auto Covariance (DAC) | | Measuring the correlation of the same physicochemical index between two dinucleotides separated by lag along the sequence [60, 61] |
| Dinucleotide Based Cross Covariance (DCC) | Dinucleotide Based Cross Covariance (DCC) | | Measuring the correlation of two different physicochemical indices between two dinucleotides separated by lag nucleic acids [60, 61] |
| Dinucleotide Based Auto-cross Covariance (DACC) | Dinucleotide Based Auto-cross Covariance (DACC) | | Combining of DAC and DCC [60-61, 43] |
| Trinucleotide Based Auto Covariance (TAC) | - | | Measuring the correlation of the same physicochemical index between trinucleotides separated by lag nucleic acids [43] |
| Trinucleotide Based Cross Covariance (TCC) | - | | Measuring the correlation of two different physicochemical indices between two trinucleotides separated by lag nucleic acids [43] |
| Trinucleotide Based Auto-Cross Covariance (TACC) | - | | Combining of TCC and TACC [43] |
| Nucleic Acid Composition (NAC) | Nucleic Acid Composition (NAC) | | Calculating the frequency of each nucleic acid type in nucleotide sequence [48] |
| Di-Nucleotide Composition (DNC) | Di-Nucleotide Composition (DNC) | | Containing 16 NAC descriptors [48] |
| Tri-Nucleotide Composition (TNC) | Tri-Nucleotide Composition (TNC) | | Containing 64 NAC descriptors [48] |
| zCurve Mathematical Formula (zCurve) | zCurve Mathematical Formula (zCurve) | | Calculating three components in three axis in genomic sequence analysis [45] |
| monoMonoKGap Theoretical Description (MonoKGap) | monoMonoKGap Theoretical Description (MonoKGap) | | Calculating features based on value of kgap [45] |
| monoDiKGap Theoretical Description (MonoDiKGap) | monoDiKGap Theoretical Description (MonoDiKGap) | | Calculating features based on value of 4 \* kgap [45] |

Table 2. List of 32 Protein feature extraction methods and their description

|  |  |  |
| --- | --- | --- |
| **Protein Feature Extraction** | **Extraction Method Description** | |
| Amino Acid Composition (AAC) | | Calculating the frequencies of 20 kinds of amino acids [62] |
| Dipeptide Composition(DC) | | transforming the variable length of proteins to fixed length feature vectors [62] |
| Composition of K-Spaced Amino Acid Pairs (CKSAAP) | | Extracting important intrinsic correlation information of protein sequences in multidimensional space [63-65] |
| Grouped Dipeptide Composition (GDC) | | A variation of the DPC descriptor which generates 25 descriptors [66] |
| Grouped Tripeptide Composition (GTC) | | Another variation of TPC descriptor which generates 125 descriptors [66] |
| Conjoint Triad (CT) | | Calculating the frequency of occurrence of each triad [67] |
| K-Spaced Conjoint Triad (KSCTriad) | | Combining CT and considers the continuous amino acid units that are separated by any *k* residues [68] |
| Composition (C)  Transition (T)  Distribution (D) | | Calculating composition descriptors  Calculating transition descriptors  Calculating distribution descriptors [69-71] |
| Encoding Based on Grouped Weight (EBGW) | | Capturing the continuity and discontinuity features based on grouped weight coding [72] |
| Auto Covariance (AC) | | Measuring the correlation of the same property between two residues separated by distance of *l* [73] |
| Moreau-Broto autocorrelation (Morean-Broto) | | Measuring the physiochemical and position information between two amino acid [74] |
| Moran Autocorrelation (Moran) | | Measuring the physiochemical information of adjacent amino acid [75] |
| Geary Autocorrelation (Geary) | | Measuring the physiochemical information and generate positive values [76, 77] |
| Quasi-Sequence-Order (QSO) | | Obtaining the sequence distribution patters for a specific physicochemical property [78] |
| Pseudo-Amino Acid Composition (PseAAC) | | Extracting the physicochemical information and sequence order information [79, 80] |
| Amphiphilic Pseudo-Amino Acid Composition (APAAC) | | Extracting the type-2 pseudo amino acid composition [79, 80] |
| Amino Acid Composition PSSM (ACC-PSSM) | | Calculating process of amino acid composition PSSM [81, 82] |
| Dipeptide Composition PSSM (DPC-PSSM) | | Extracting the sequence-order information in the PSSM [82] |
| Bi-gram PSSM (Bi-PSSM) | | Calculating the frequency of the transition between amino acids [83] |
| Auto Covariance PSSM (AC-PSSM) | | Measuring the correlation of the same property between two residues separated by lag [84] |
| Pseudo PSSM (PsePSSM) | | Calculating the PsePSSM feature vector according to the pseudo amino acid composition [85] |
| AB-PSSM | | Calculating feature vector based on averaged PSSM over blocks [86] |
| Secondary Structure Composition (SSC) | | Calculating feature based normalized count of frequency of the structural motifs present at the amino-acid residue positions [87] |
| Accessible Surface Area composition (ASA) | | Calculating feature based on normalized sum of accessible surface area [87] |
| Torsional Angles Composition (TAC) | | Calculating features based four different types of torsional angles [87] |
| Torsional Angles bigram (TA-bigram) | | Calculating feature based on the bigram of the torsional angles [87] |
| Structural Probabilities bigram (SP-bigram) | | Calculating feature based on structural probabilities for each position of amino acid residue [87] |
| Torsional Angles Auto-Covariance (TAAC) | | Calculating feature from the torsional auto-covariance [87] |
| Structural Probabilities Auto-Covariance (SPAC) | | Calculating feature from the structural probabilities [87] |

### 2.2.2 Feature Selection and Dimensionality Reduction

The dimension of feature vectors after feature extraction and feature fusion could be extremely high. Take 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 descriptor using physicochemical properties to extract sequence information. As we can kown, the AAindex database contains 554 physicochemical propertyies. High dimensions feature vectors usually contain some redundant and noisy features, which may increase memory storage and run time. Especially, this case may lead to overfitting and performance degradation. It is desirable to apply feature selection and dimensionality reduction to eliminate redundancy and noise information. Therefore, ALLFEATURE integrated a step of feature selection and dimensionality reduction methods in **Table 3**. Selecting those features which contribute most to the prediction can effectively improve prediction accuracy and reduce the implementation time. These feature selection methods rank features according to their feature importance score for prediction of structure and function. Only wanted number of top-ranking features will be selected in the step of feature selection. Dimensionality reduction methods project the raw feature space to new low-dimension feature space, which could mine the linear or nonlinear relationship, eliminate redundancy and retain effective feature information.

Table 3. Feature Selection and Dimensionality Reduction Methods

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Feature Selection Method** | | **Description** | **Dimensionality Reduction Method** | **Description** |
| Lasso | Using Lasso liner model to recursively eliminate features [31, 88] | | K-means | Clustering data by separating samples in n groups of equal variances [34] |
| ElasticNet | Using ElasticNet model to recursively eliminate features [89] | | T-SNE | Visualizing high-dimensional data [36] |
| L1-SVM | Using SVM with L1 penalty model to recursively eliminate features [90] | | Principal Component Analysis (PCA) | Linear dimensionality reduction using singular value decomposition [35] |
| CHI2 | Retrieving best features based on test [29, 91] | | Kernel PCA (KPCA) | Non-linear dimensionality reduction through use of kernels [35] |
| Pearson Correlation (PC) | Retrieving best features based on Pearson correlation [32] | | Locally linear embedding (LLE) | Reducing projection of data which preserves distances within local neighborhoods [105] |
| ExtraTree | Using ExtraTree model to recursively eliminate features [92] | | Truncated Singular Value Decomposition (TSVD) | Linear dimensionality reduction by means of truncated singular value decomposition [106] |
| xgBosst | Using xgBoost model to recursively eliminate features [40. 93] | | Non-negative matrix factorization (NMF) | Reducing dimension by finding two non-negative metrices [107] |
| SVM-RFE | Using linear SVM model to recursively eliminate features [30, 100] | | Multi-dimensional Scaling (MDS) | Reducing dimension by modeling data as distances in a geometric space [108] |
| LOG-RFE | Using Logistic Regression model to recursively eliminate features [94] | | Independent Component Analysis (ICA) | Reducing dimension by finding components with some sparsity [109] |
| Mutual Information (MI) | Retrieving best features based mutual information [95] | | Factor Analysis (FA) | Reducing dimension by performing a maximum likelihood estimate [110] |
| Minimum Redundancy Maximum Relevance (MRMR) | Selecting features that still having high correlation to the classification variable [96] | | Agglomerate Feature (AF) | Recursively merges feature instead of samples [111] |
| Joint Mutual Information (JMI) | Retrieving best features based joint mutual information [97] | | Gaussian Random Projection (GRP) | Reducing the dimension by projecting the original input space using the Gaussian distribution [112] |
| Maximum Relevance Maximum Distance (MRMD) | Retrieving best features by measuring relevance and redundancy between features [98] | | Sparse Random Projection (SRP) | Reducing dimension by projecting the original input space using a sparse random matrix [113] |
| ReliefF | Retrieving best features by calculating and ranking a feature score for each feature [33] | | Autoencoder | Reducing the dimension using encode and decode neural network [114] |
| Trace Ratio | Retrieving best features by calculating the corresponding score in trace ratio form [99] | | Gaussian Noise Autoencoder (GNA) | Corrupting input before being passed to autoencoder neural network [115] |
| Gini Index | Retrieving best features by constructing the measure function based on Gini-Index [100] | | Variational Autoencoder (VA) | Neural network can be trained with stochastic gradient descent [116] |
| Spectral Feature Selection (SPEC) | Retrieving best features based on structure induced [101] | | - | - |
| Fisher Score | Retrieving best features based on scores of features under the Fisher criterion [102] | | - | - |
| T Score | Retrieving best features based on their t-score [103] | | - | - |
| Information Gain (IG) | Retrieving best features based on their information gain [104] | | - | - |

### 2.2.3 Models Construction

A lot of analysis of biological sequence could be regarded as classification tasks in bioinformatics and computational biology, so machine learning and deep learning methods are key step for predictor construction. SVM is a kernel-based classifier through constructing the optimal hyperplane. RandomForest is a tree-based and widely used machine learning method. SVM and RandomForest are commonly used in the area of prediction tasks. Recently, gradient boosting decision tree [116], XGBoost and LightGBM are demonstrated to be excellent classifiers via gradient boosting algorithm. **ALLFEATURE** integrated 10 popular machine learning methods SVM, KNN, RandomForest, LightGBM, XGBoost, Adaboost [117], ExtraTree, gaussian Naïve Bayes [119], gradient boosting. In recent years, deep learning frameworks provide effective solutions for the analysis of DNA, RNA and protein sequence. Deep learning can mine essential sequence represent information via hierarchical structure, which can accurately, effectively and better perform the sequence prediction tasks. **ALLFEATURE** integrated three deep learning methods, including deep neural network (DNN), convolutional neural network (CNN), and recurrent neural network (RNN).

### 2.2.4 Cross-validation and Models Evaluation

In this paper, stratified K-Folds cross-validator is used for obtaining classification accuracy and plotting ROC curves. Each dataset will be split to five datasets for testing performance of each predictors. All models are evaluated using classification accuracy that reflect the fraction of correct predictions:

(2)

Most structural and functional of sequences predictions are binary classification and the accuracy can be calculated by:

(3)

where TP, TN, FP and FN in the above equations represent true positive, true negative, false positive and false negative, respectively.

(4)

where i means ith classes.

## **2.3 Application of ALLFEATURE**

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 N6-methyladenosine sites, RNA N 6-methyladenosine sites and protein-protein interactions datasets to evaluate the validation of **ALLFEATURE**. The prediction results can be automatically and easily generated. And the model performance could be comparable and even higher than the state-of-the-art approaches, which indicate our proposed python packages are useful and powerful when analyzing DNA, RNA and protein biological sequences.

### 2.3.1 DNA N6-methyladenine Sites Prediction

(6mA) is one kind of post-replication modification occurring in a wide range of DNA sequences [120]. In prokaryotes, 6mA has been found to be associated with a wide range of biological processes such as DNA replication, repair, transcription, and cellular defense. 6mA site-containing sequences were taken from the genome of *Mus musculus* in the MethSMRT database, including 1934 positive samples and 1934 negative samples. Firstly, we used Kmer, Psednc, binary, TNC and MonoKGap five feature extraction methods to construct initial feature vectors, multi-information fusion can represent more effective information of DNA sequences. The extracted feature vector was used to construct prediction models include SVM, KNN, RandomForest, LightGBM, XGBoost, Adaboost, Bagging, ExtraTree, gaussian Naïve Bayes, gradient boosting, DNN, CNN and RNN predictors, **Figure 3**.

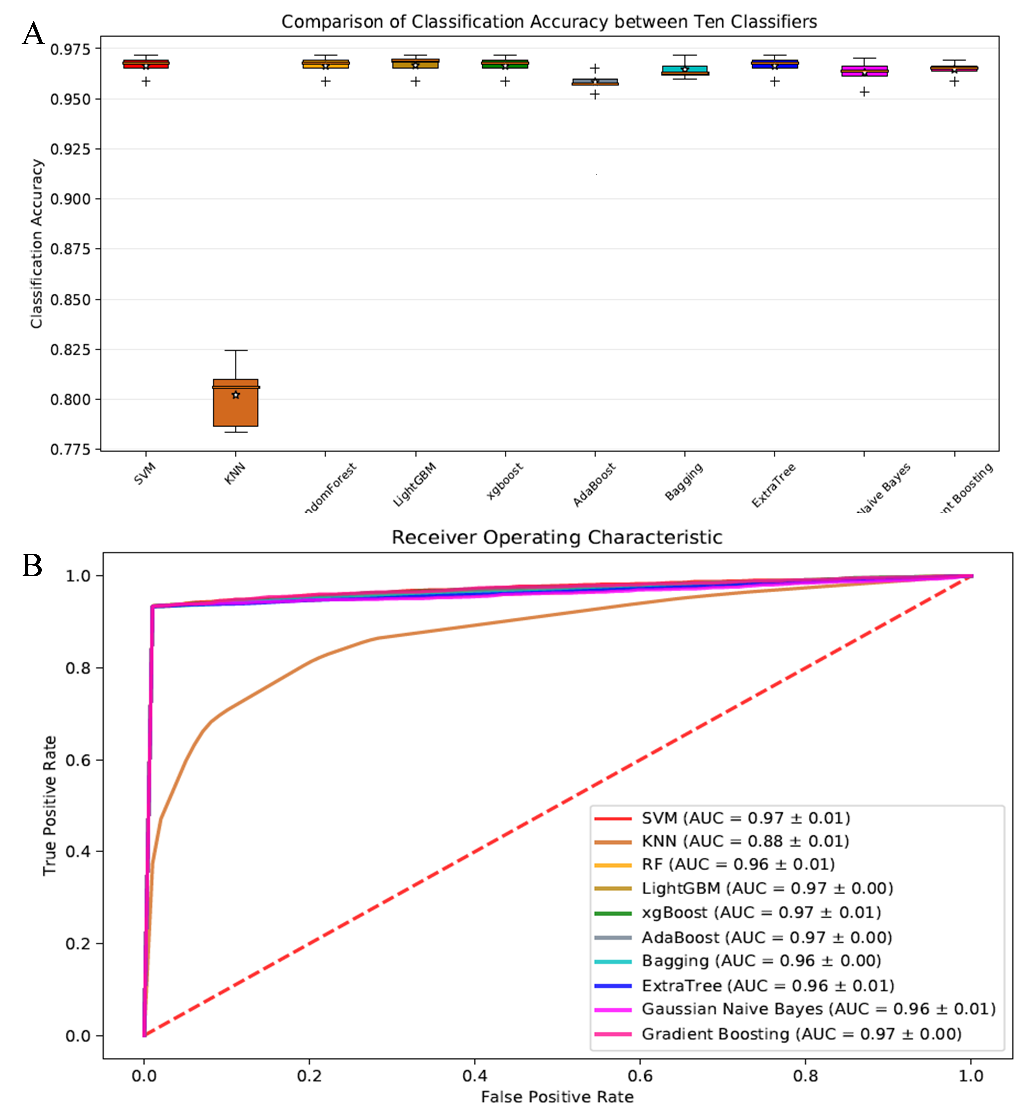


Figure 3. The boxplot and ROC curves of DNA N6-methyladenine sites using various classifiers (A) 13 classifiers all achieve satisfactory accuracy, and SVM, RandomForest, XGBoost, DNN, RNN obtain superior performance than other classifiers. (B) The ROC curves of 13 classifier indicate ALLFEATURE can obtain accurate prediction results.

Next, all Lasso, ElasticNet, L1-SVM, CHI2, Pearson Correlation, ExtraTree, XGBoost, SVM-RFE, LOG-RFE, Mutual Information, Minimum Redundancy Maximum Relevance, Joint Mutual Information, Maximum-Relevance-Maximum-Distance, ReliefF, Trace Ratio, Gini index, Spectral Feature Selection, Fisher Score, T Score, Information Gain methods are employed to fulfill dimensionality reduction. Finally, the optimized feature vectors via the process of feature selection or dimensionality reduction are fed into those predictors.

To our observation and comparison, when fusing Kmer, Psednc, binary, TNC and MonoKGap to extract features, ExtraTree, Fisher Score, MRMR feature selection methods can achieve better prediction performance. In addition, the execution time was only one third of execution time with original data. The boxplot and ROC curves under different classifiers via the operation of ExtraTree, Fisher Score and MRMR feature selection can be shown in **Figure 4**, **5** and **6**, respectively.

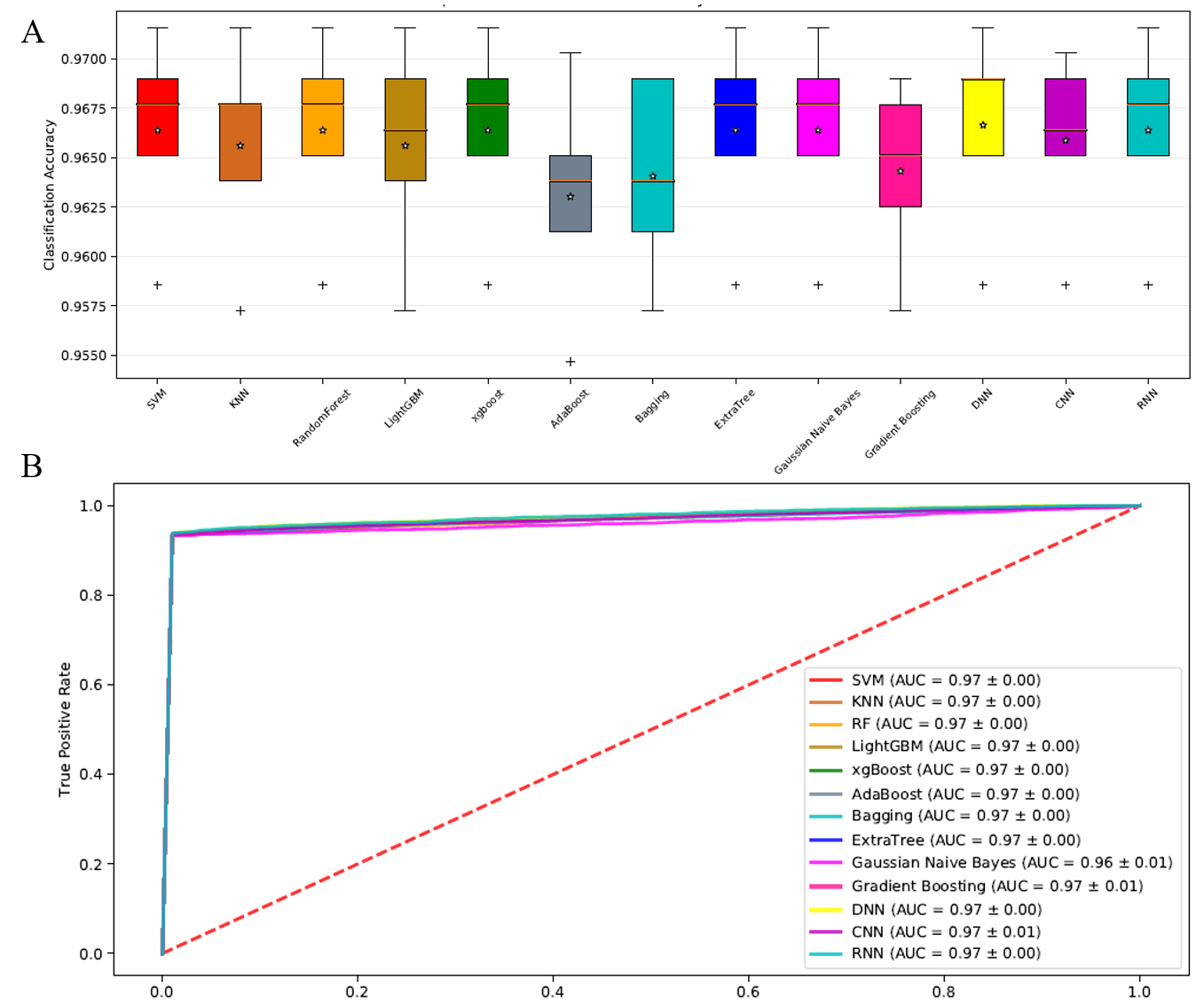


Figure 4. The boxplot and ROC curves of DNA N6-methyladenine sites using various classifiers with ExtraTree feature selection method.

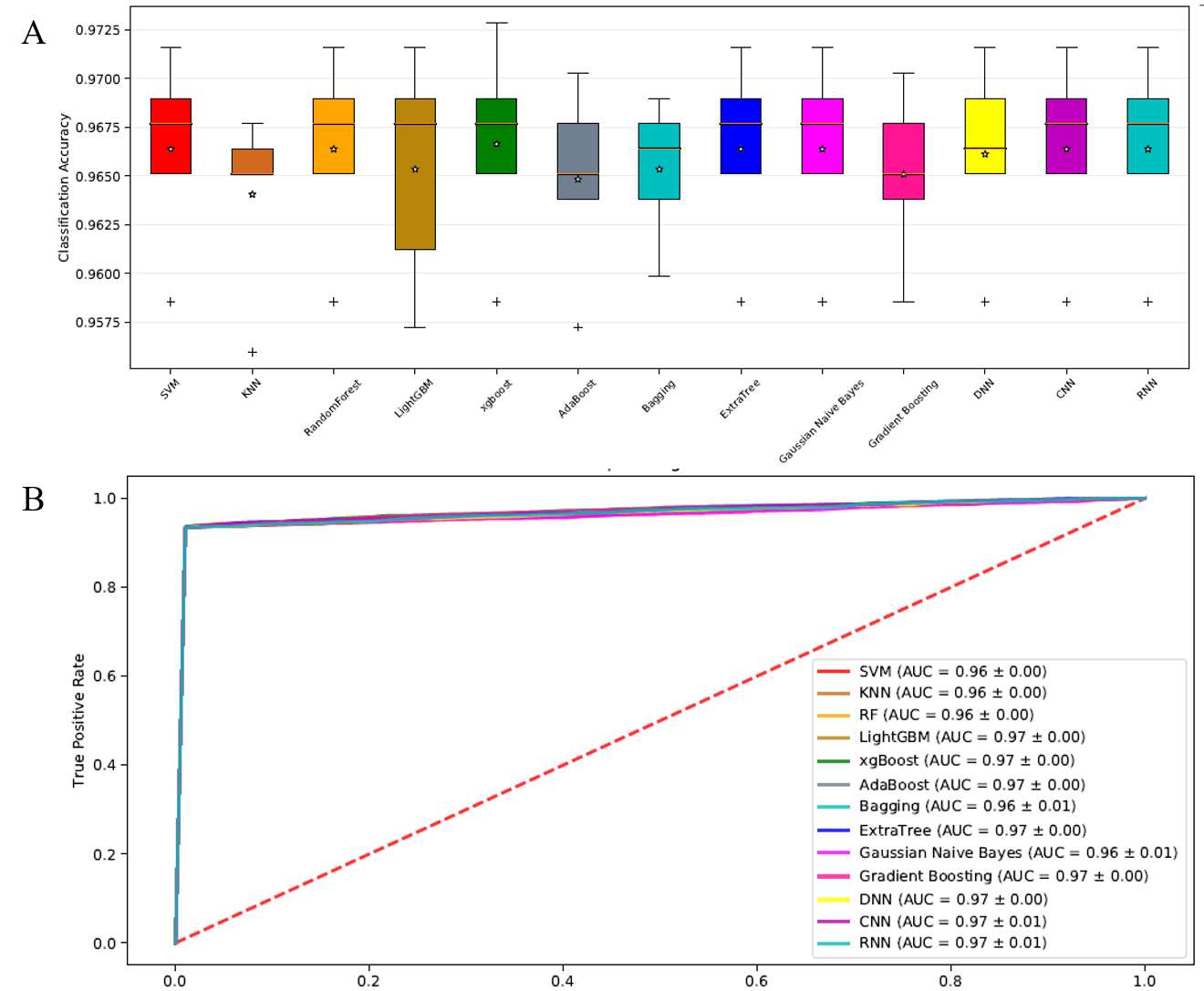


Figure 5. The boxplot and ROC curves of DNA N6-methyladenine sites using various classifiers with Fisher Score feature selection method.

G:\feature_selection_package\正文\Figure 2.emf

Figure 6. The boxplot and ROC curves of DNA N6-methyladenine sites using various classifiers with MRMR feature selection method.

### 2.3.2 RNA N6-methyladenine Sites Prediction

N6-methyladenosine (m6A) refers to methylation of the adenosine nucleotide acid at the nitrogen-6 position. It plays an important role in a series of biological processes, such as splicing events, mRNA exporting, nascent mRNA synthesis, nuclear translocation and translation process [17]. It is highly desirable to build up an effective predictive model to identify RNA N6-methyladenine sites. In this paper, the dataset contains 2260 sequences, where 1130 represents true methyladenosine sites, and the remaining 1130 are false methyladenosine sites. Firstly, we used Kmer, Psednc, binary, TNC and MonoKgap five feature extraction methods to construct initial feature vectors. Different feature information complements with each other, and multi-information fusion can obtain fully represent the feature information, leading to better model performance and elaborating biological feature information, **Figure 7**.

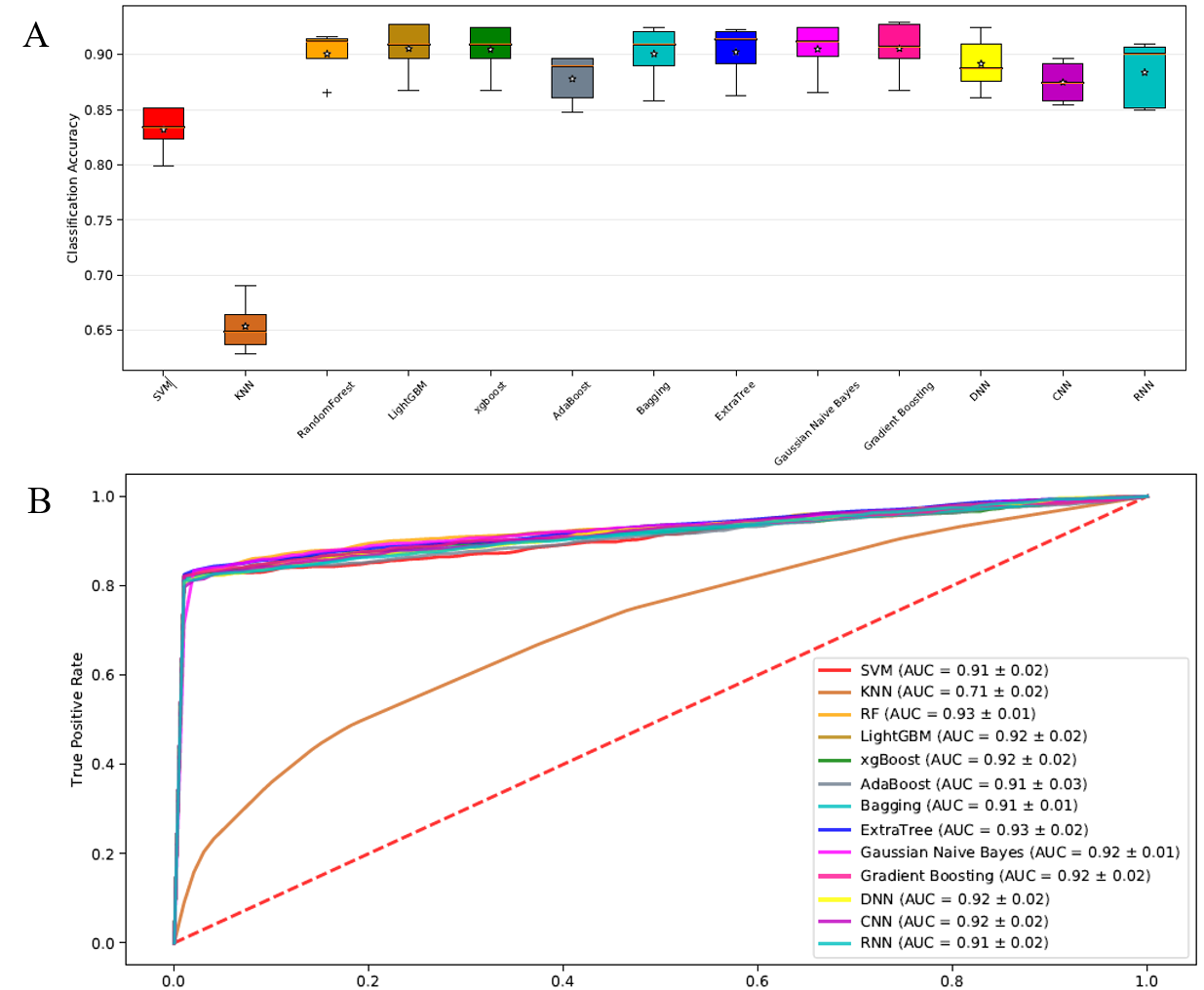


Figure 7. The boxplot and ROC curves under different classifiers on RNA N6-methyladenine sites dataset. (A) The boxplot show machine learning and deep learning method can better predict RNA N6-methyladenine sites. (B) The 13 classifiers all achieve the good true positive rate at corresponding false positive value, and the AUC values are also high.

In order to evaluate the effectiveness of feature selection, we using Lasso, ElasticNet, L1-SVM, CHI2, Pearson Correlation, ExtraTree, XGBoost, SVM-RFE, LOG-RFE, Mutual Information, Minimum Redundancy Maximum Relevance, Joint Mutual Information, Maximum-Relevance-Maximum-Distance, ReliefF, Trace Ratio, Gini index, Spectral Feature Selection, Fisher Score, T Score, Information Gain to determine optimal feature subset, which can eliminate the redundant and noisy features, retain valuable features and reduce run time. The predictive performance of SVM-RFE via different classifiers can be shown in **Figure 8**.

G:\feature_selection_package\正文\Figure3.emf

Figure 8. The boxplot and ROC curves under different classifiers on RNA N6-methyladenine sites dataset via SVM-RFE feature selection.

## **2.4 Summary and Conclusion**

In this chapter, I present a python package and web server for feature extraction, feature selection, machine learning, deep learning and model evaluation called ALLFEATURE. We also test the proposed package using DNA N6-methyladenine sites, RNA N6-methyladenine sites and protein-protein interactions dataset. Experimental results indicate ALLFEAURE is more useful, effective and accurate compared with other state-of-the-art approaches. We integrated 20 types feature selection methods and 16 types dimensionality reduction approaches to increase computational efficiency and construct better generalization model. Especially, in order to build up biological sequence predictor, ALLFEAURE integrates SVM, KNN, RandomForest, LightGBM, XGBoost, Adaboost, ExtraTree, gaussian naïve bayes, gradient boosting, DNN, CNN and RNN, 13 types classifiers to construct sequence analysis model. It is anticipated that **ALLFEATURE** can be useful tools for bioinformatics and computational biology.

# CHAPTER 3: Introduction for EMR Data and CKD

## **3.1 Chronic Kidney Disease and eGFR**

The increasing incidence of chronic kidney disease (CKD) in the United States and around the world lays an enormous burden on healthcare [121, 122]. By December 2015, there were 703,243 prevalent patients with End Stage Renal Disease (ESRD), with the unadjusted incident rate of 378 per million population [123]. In 2017, there were approximately 500,000 patients on different dialysis modalities (91% are on hemodialysis), 20,000 received transplants [123]. Treatments that are effective in patients with advanced CKD also increase health care costs and lead to adverse effects [124]. Thus it is essential to identify earlier stage CKD and prevent its progression to ESRD [125]. However, the biggest challenge is that most people do not have any signs or symptoms in the early stages of CKD and go undetected until an advanced stage [126].

Early identification and targeted intervention of CKD have attracted considerable attention from clinicians and researchers since both have the potential to reduce the number of patients progressing to ESRD and lower the mortality rate related to CKD and associated healthcare costs [127]. With the growing availability of Electronic Medication Record (EMR) data, various computational predictive models for disease progression have been developed to facilitate the decision-making process of health care providers [124, 128, 129]. Choi et al. have classified disease progression models into two categories based on the extent of targeted diseases: models focusing on a specific disease and those focusing on a broader range of conditions [130]. Among those disease-specific progression models, some are validating specific hypotheses of disease progression based on experts’ knowledge [124, 11,12], while others are driven by application of advanced statistical methods [13–15]. Approaches that can be generalized to model the progression of multiple diseases have been proposed, where statistical methods and machine learning techniques are widely used [16–18]. For kidney disease, different models have been developed in predicting CKD stages to ESRD over time and in predict variations of GFR in patients [7,9,19–22].

Estimated glomerular filtration rates (eGFRs) have been implemented in primary care to assist the early detection and staging of CKD [26,27]. The eGFR formula [28] is:

(5)

where eGFR (estimated glomerular filtration rate) = mL/min/1.73 m2; SCr (standardized serum creatinine) = mg/dL, κ = 0.7 (females) or 0.9 (males), α = −0.329 (females) or −0.411 (males), min = indicates the minimum of SCr/κ or 1, max = indicates the maximum of SCr/κ or 1, and age = years.

Although routine reporting eGFR had positive effects in clinical practice, including prevention of CKD progression and reduction of CKD related complications, there are still concerns in its negative effects caused by over diagnosis [26]. Other than reporting eGFR, studies have begun using an alternative measurement, such as eGFR decline derived from eGFR to evaluate and predict CKD progression [29,30]. Studies have investigated the association between eGFR change and ESRD risk and mortality risk respectively, where age and gender factors have been taken into account [29,31,32]. Higher eGFR decline levels were proved to be associated with greater hazard ratios of ESRD in several clinical trials [33–35]. However, a smaller percentage of eGFR change, which is a reflection of the short-term treatment effect of kidney disease, is underexamined [29].

## **3.2 Machine Learning in EMR Data Analysis**

The application of statistical models and machine learning techniques have been rapidly-growing in estimating health and disease outcomes [23]. Cerqueira et al. developed a model using the Cox proportional hazard regression in predicting the risks that pre-dialysis pediatric patients progress to ESRD from CKD [9]. Decruyenaere et al. compared the performances of machine learning methods with logistic regression in predicting the occurrence of delayed renal graft in renal transplant patients [24]. Their results showed that linear support vector machine outperformed logistic regression in sensitivity. Kumar compared six machine learning classifiers (Random Forest, Sequential Minimal Optimization, NaiveBayes, Radial Basis Function, Multilayer Perceptron Classifier, and SimpleLogistic) in CKD classification and identified that Random forest outperformed the other classifiers [25].

The future renal function of a CKD patient can be predicted if their GFR variations can be predicted since GFR is the best test in measuring the level of kidney function [3,7,36]. Consequently, the time to reach GFR thresholds corresponding to stages of CKD can be anticipated. An integrated intelligent fuzzy expert system has been used in predicting future GFR based on selected clinical variables and demonstrated reliable accuracy [7]. However, there is still a lack of efficient methods for predicting the individual level timeframe of CKD progression [37]. Specifically, Random Forest Regression, featured with a reduction in overfitting and less variance, has not been used to predict the progression of renal function yet. This study predicted future eGFR values using Random Forest regression based on real-world EMR data representing the general population in the upper Midwest. The main aim of this study is to propose an efficient and reliable clinical tool that allows us to identify at an earlier stage and preemptively suggest the preventive strategies that can attenuate the development of this challenging disease in patients that reside in our agricultural communities.

# CHAPTER 4: Predicting Outcomes of Chronic Kidney Disease from EMR Data Based on Random Forest Regression

## **4.1 Methods**

### 4.1.1 Data Acquisition

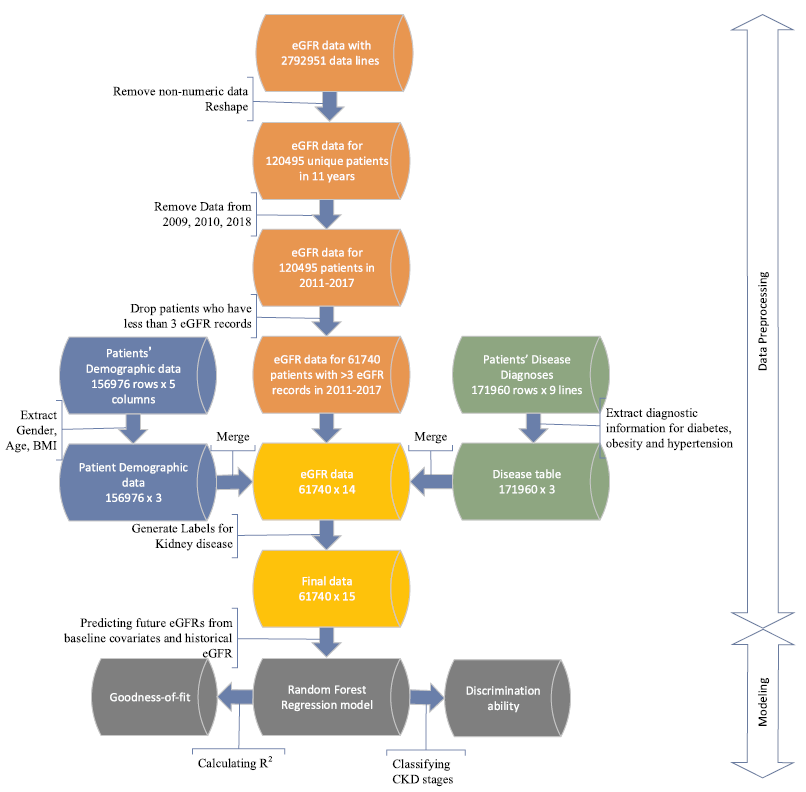
The dataset used in this study comes from real-world clinical data. We built up a cohort consisting of 120,495 patients aged from 20 to 80 in Sioux Falls, SD, region that receiving primary care from Sanford Health. By consulting with the nephrologist, we pulled out data elements influencing GFR variations for this cohort from the comprehensive Sanford EMR database for years 2009–17. None of the identifiable information was extracted to protect patients' privacy. We are focusing on the progression of CKD, so only the “clinical” encounter data was included. Those data elements contain patients’ eGFR records for years 2009–17, the ICD-10 codes [38] for CKD, Hypertension, Diabetes, and Obesity, and their demographic information comprising Age, Gender, and Race. A detailed description of the data elements is given in **Table 4**.

Table 4. Predictor and covariate data type breakdown

|  |  |  |
| --- | --- | --- |
| **Feature** | | **Data elements** |
| **Predictor**  eGFR | All clinical encounter eGFR data with testing dates were pulled out for each patient | |
| **Covariates**  Age  Gender  Race/Ethnicity  BMI  Hypertension  Diabetes  Obesity | Continuous  Categorical  Categorical  Continuous  Flagged for each patient (ICD-10: I10, I11, I12, I13, I15, I16)  Flagged for each patient (ICD-10: E08, E09, E10, E11, E13)  Flagged for each patient (ICD-10: E66.9) | |

### 4.1.2 Data Pre-processing

The extracted data were formatted into three separate tables: (1) eGFR table with rows representing patients and columns containing eGFR for multiple years; (2) Demographic table consisting of demographic information; and (3) Disease table composed of diagnosis status of hypertension, diabetes, and obesity. The processing of these data tables is illustrated in Figure 1 and described below.



1. The eGFR table has 120,495 unique patients and 10 columns, each of which representing eGFR records in years 2009–18. First, the non-numeric eGFR records (e.g. “>90”)) were considered as missing data and marked as “NA.” For patients with more than one eGFR values in a specific year, the median of these values was calculated and kept for that year in the table.
2. More than 95% eGFR records are missing in 2009 and 2010, so data from these two years were omitted. Since the data in 2018 was not complete when the data was extracted, we also excluded the records in this year. Patient lines were removed from the data if they have no more than three available records from 2011 to 2017. The final eGFR table has 61,740 unique patients and 7 years eGFR data for each patient with at least three eGFR values.
3. Next, the different CKD stages were determined by eGFR values in the physical laboratory. Therefore, the CKD stages true labels were created using eGFR. The minimum eGFR value in each of the years between 2011 and 2017 was evaluated first, and then the CKD stages labels were produced based on the following equation:

(6)

1. The true labels were also merged into eGFR matrix based on their index (patient ID).
2. The current eGFR matrix includes 61,740 unique patients, and each patient has 7 years eGFR values from 2011 to 2017 and labels for the CKD stage from 1 to 5. The final data table was created by merging the eGFR table with the demographic table and the disease table by matching their patient IDs.

## **4.2 Construction of Random Forest Regression Model**

The longitudinal design of this study enables the estimation the future eGFR value from the past eGFR values adjusted by clinical covariates. We selected Random Forest regression as the primary model because of its efficiency and accuracy to predict 1 year, 2 years and 3 years eGFRs from the historical eGFR records between years 2011–14.

*Baseline covariates and predictors*: The variables included in the analysis were baseline eGFR, age, gender, ethnicity, body mass index (BMI), hypertension, diabetes, obesity.

*Outcome:* eGFR values in the year 2015, 2016, and 2017 were considered as the outcome variable. This is based on the consensus that GFR is the best measure of kidney function [36].

*Model development:* the inputs of this model are the attributes of the *i*th patient denoted by a vector *Xi =* (*xi*1,*…*, *xin*) which includes eGFR values from multiple years and other covariates listed in Table 1. The output is the future eGFR for the *i*th patient denoted by *Gij* where *j* indicating a future year.

In the computational experiment, we used the processed dataset with 61,740 unique patients. For building the model in predicting eGFR of 2015, the patient must have recorded eGFR in 2015, and at least two recorded eGFR between 2011 and 2014. Similar requirements were used in predicting eGFR of 2016 and 2017. Other years’ eGFR values were imputed and filled by the median eGFR value of each patient. All models were built using scikit-learn package [39]. The parameters of Random Forest Regressor were determined using the grid-search method. Only two parameters, number of estimators and maximum number of features, were tuned because they can determine numbers of trees in forest and how the tree will split and grow. We also randomly split the dataset and repeat the training process five times with different sets to avoid over fitting for our models.

## **4.3 Assessment of model performance**

### 4.3.1 Goodness-of-fit

The model fit of the proposed Random Forest Regression was measured using the coefficient of determination *R*2 to show how well the fitted eGFR value approximates the real eGFR value. *R*2 is a measure used to represent the percent of variation explained, i.e., the proportion of variance in the dependent variable that can directly be attributed to variance in the independent variables. An *R*2 of 1 would indicate all changes we see in the dependent variable are caused by changing our independent variables, whereas an *R*2 of 0 means no such direct impact. We also checked the residual plot since randomly distributed residuals indicate the model fits the data well.

### 4.3.2 Discrimination

The estimated eGFR values were used to classify patients into different

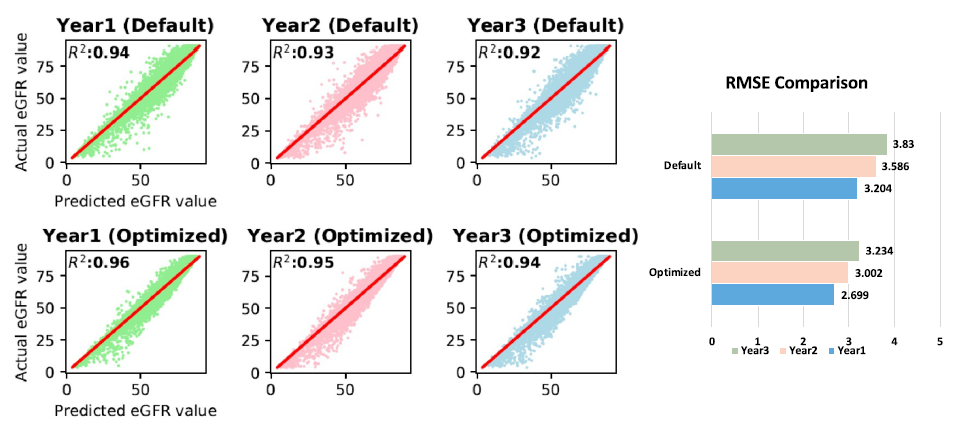
CKD stages based on Eq. (1). Both micro-average and macroaverage were generated to illustrate the classification accuracy of the Random Forest model.

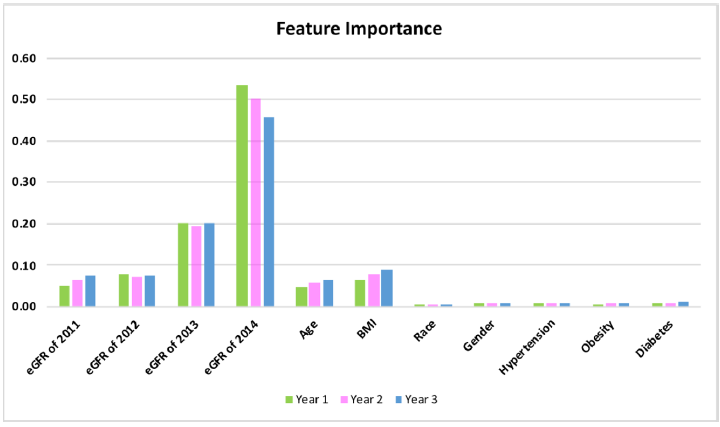
## **4.4 Results**

In Random Forest regression analysis, the predicting accuracy was enhanced by optimizing the values of hyperparameters, where the default values and the optimized values of the hyperparameters were shown in **Table 5**. The predicted versus observed eGFR values in years 1–3 were plotted for both the default and optimized hyperparameters in Fig. 2. The *R*2 was increased from default to optimized hyperparameters in each of the three years. The Root of Mean Squared Error (RMSE) in Fig. 2 illustrated that the optimized hyperparameters provided a more accurate prediction that the default values. It is also worse noticing that the prediction accuracy decreased over time. With the optimal parameters, we further examined the importance of the features included in the analysis whose results were given in Fig. 3. It is not surprising that previous eGFR records played essential roles than other features since eGFR is decreasing continuously over time. Although the information of age and BMI are considered in estimating GFR using the eGFR formula [28], predictions based solely on the previous eGFR are not sufficient. Age and BMI, as illustrated in Fig. 3, still contribute to 4.7–9% to the future three years of eGFR respectively. All the other features, including Race, Gender, Obesity, Hypertension, and Diabetes, accounted for a total of 2.7–3.9% of the variances.

Table 5. Hyperparameters used in the Random Forest Regression for the default and optimized models.

|  |  |  |
| --- | --- | --- |
|  | **Default** | **Optimized** |
| # of trees  Max depth  Max sample split  Min samples leaf  Max features  Bootstrap | 10  None  2  1  11  True | 100  None  2  1  8  True |





## **4.5 Conclusion and Discussion**

In this study, we proposed a model in predicting future eGFR values, which is based on Random Forest regression that can efficiently learn from the real world EMR data and accurately predict future patient outcomes. We validated this model on an EMR dataset extracted from a health system located in the Great Plains. The computational experiment achieved an average *R*2 of 0.95 over three years with small variation. And an 88% Macro Recall and a 96% Macro Precision by averaging over three years were obtained by dividing patients into different CKD stages using estimated eGFRs. Besides, we identified the crucial features that contribute to the variation of future eGFRs, which include recent eGFR records, Age and BMI. Therefore, our proposed predictive model of eGFR has excellent potential to be developed into a clinical decision support tool to assist doctors in providing preventive advice to patients.

One of the limitations of this work is that only patients with numeric eGFR records were included, which exclude those patients without CKD symptoms in the study period. However, those excluded patients can serve as a control group whose clinical information can be incorporated into the predictive model to adjust the parameter estimations. Also, the current study only contained historical eGFRs, demographic characteristics, and relevant disease diagnoses. Studies have shown that an individual's genetic and phenotypic characteristics both affect their risk in developing kidney disease, including genetic mutations, a family history, gender, ethnicity, age, obesity, socioeconomic status, smoking, nephrotoxins, acute kidney injury, diabetes mellitus, and hypertension [41]. Thus we are planning to address those issues in future studies to improve the practicability of the predictive model of eGFR in support of patient care.

# REFERENCES

1. Selvaraj, S. and J. Natarajan, *Microarray Data Analysis and Mining Tools.* Bioinformation, 2011. **6**(3): p. 95-99.
2. 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.
3. Cost of Next-Generation Sequencing. (n.d.). Retrieved from <https://www.illumina.com/science/technology/next-generation-sequencing/beginners/ngs-cost.html>.
4. GenBank and WGS Statistics. (n.d.). Retrieved from https://www.ncbi.nlm.nih.gov/genbank/statistics/.
5. Garber, M., et al., *Computational methods for transcriptome annotation and quantification using RNA-seq.* Nature methods, 2011. **8**(6): p. 469-477.
6. Wang, Z., M. Gerstein, and M. Snyder, *RNA-Seq: a revolutionary tool for transcriptomics.* Nature reviews genetics, 2009. **10**(1): p. 57-63.
7. Ozsolak, F. and P.M. Milos, *RNA sequencing: advances, challenges and opportunities.* Nature reviews genetics, 2011. **12**(2): p. 87-98.
8. 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.
9. Navin, N., et al., *Tumour evolution inferred by single-cell sequencing.* Nature, 2011. **472**(7341): p. 90-4.
10. Xu, X., et al., *Single-cell exome sequencing reveals single-nucleotide mutation characteristics of a kidney tumor.* Cell, 2012. **148**(5): p. 886-95.
11. D’Agaro, E. (2017). NGS genome annotation profiling using data analysis workflows. *Journal of Biotechnology*, *256*. doi: 10.1016/j.jbiotec.2017.06.039.
12. 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.
13. 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.
14. 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.
15. 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.
16. Sasnauskas, G. (2018). DNA binding domain of restriction endonuclease McrBC in complex with N4-methylcytosine DNA. doi: 10.2210/pdb6gcf/pdb.
17. Lynch, S., & Kool, E. (2015). N6-Methyladenosine RNA. doi: 10.2210/pdb2mvs/pdb.
18. 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.
19. 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.
20. Fischer, D., & Eisenberg, D. (2008). Protein fold recognition using sequence-derived predictions. *Protein Science*, *5*(5), 947–955. doi: 10.1002/pro.5560050516.
21. 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.
22. 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.
23. 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.
24. Constructing Reliable Protein-Protein Interaction (PPI) Networks. (2017). *Computational Prediction of Protein Complexes from Protein Interaction Networks*, 15. doi: 10.1145/3064650.3064653.
25. Zhang, H. (2016). Overview of Sequence Data Formats. *Methods in Molecular Biology Statistical Genomics*, 3–17. doi: 10.1007/978-1-4939-3578-9\_1.
26. Feature Extraction and Classification. (2014). *Medical Diagnosis Using Artificial Neural Networks*, 159–169. doi: 10.4018/978-1-4666-6146-2.ch011.
27. Deep Learning in Python: Preventing Overfitting. (2019). doi: 10.4135/9781526493453.
28. 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.
29. An Introduction to the Chi-Square Test. (2017). doi: 10.4135/9781473980525.
30. 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.
31. Missing-Value, M.-V. (n.d.). The Lasso for Linear Models. *Statistical Learning with Sparsity*, 22–43. doi: 10.1201/b18401-4.
32. Applications Of Pearson Correlation To Measurement Theory. (n.d.). *Correlation and Regression*, 67–96. doi: 10.4135/9781412983815.n4.
33. 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.
34. 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.
35. 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.
36. 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.
37. Qi, X., Silvestrov, S., & Nazir, T. (2017). Data classification with support vector machine and generalized support vector machine. doi: 10.1063/1.4972718.
38. Vanicek, T. (2019). Classification in major depressive disorder using randomForest and various cortical and subcortical gray matter measures. *Intrinsic Activity*, *7*(Suppl. 1). doi: 10.25006/ia.7.s1-a3.49.
39. Wang, D., Zhang, Y., & Zhao, Y. (2017). LightGBM. *Proceedings of the 2017 International Conference on Computational Biology and Bioinformatics - ICCBB 2017*. doi: 10.1145/3155077.3155079.
40. Supplemental Information 3: Performance of XGBoost predictive models. (n.d.). doi: 10.7717/peerj.3230/supp-3.
41. Zhang, Y., & He, P. (2010). A revised AdaBoost algorithm: FM-AdaBoost. *2010 International Conference on Computer Application and System Modeling (ICCASM 2010)*. doi: 10.1109/iccasm.2010.5623209.
42. 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.
43. 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.
44. 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.
45. 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.
46. 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.
47. 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.
48. 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.
49. 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.
50. Ketkar, N. (2017). Introduction to Deep Learning. *Deep Learning with Python*, 1–5. doi: 10.1007/978-1-4842-2766-4\_1.
51. Deep Learning in Python: Different Types of Deep Learning Network. (2019). doi: 10.4135/9781526493439.
52. 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.
53. Convolutional Neural Network. (2018). *Encyclopedia of Social Network Analysis and Mining*, 418–418. doi: 10.1007/978-1-4939-7131-2\_100208.
54. Gelenbe, E. (1992). Learning in the Recurrent Random Neural Network. *Neural Networks*, 1–12. doi: 10.1016/b978-0-444-89330-7.50004-1.
55. 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.
56. Lee, D., Karchin, R. and Beer, M.A. (2011) Discriminative prediction of mammalian enhancers from DNA sequence. Genome research, 21, 2167-2180.
57. 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.
58. 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.
59. 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.
60. 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.
61. 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.
62. 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.
63. 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.
64. 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.
65. 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.
66. 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.
67. 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.
68. Göktepe, Y.E. and Kodaz, H. (2018) Prediction of protein-protein interactions using an effective sequence based combined method. Neurocomputing, 303, 68-74.
69. 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.
70. 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.
71. 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.
72. 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.
73. 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.
74. Horne, D.S. (1988) Prediction of protein helix content from an autocorrelation analysis of sequence hydrophobicities. Biopolymers, 27, 451-477.
75. 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.
76. 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.
77. Lin, Z. and Pan, X.M. (2001) Accurate prediction of protein secondary structural content. J. Protein Chem., 20, 217-220.
78. 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.
79. Chou K.C. (2001) Prediction of protein cellular attributes using pseudo-amino acid composition. Proteins, 43, 246-255.
80. Chou, K.C. (2005) Using amphiphilic pseudo amino acid composition to predict enzyme subfamily classes. Bioinformatics, 21, 10-19.
81. 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.
82. 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.
83. 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.
84. 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.
85. 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.
86. 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.
87. 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.
88. 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.
89. 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.
90. 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.
91. 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.
92. Tree-Structured Classifier. (n.d.). *SpringerReference*. doi: 10.1007/springerreference\_66000.
93. 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.
94. Introduction to the Logistic Regression Model. (2005). *Applied Logistic Regression*, 1–30. doi: 10.1002/0471722146.ch1.
95. Lajevardi, S. M., & Hussain, Z. M. (2009). Feature selection for facial expression recognition based on mutual information. *Exhibition*. doi: 10.1109/ieeegcc.2009.5734265.
96. Zhao, X.-M. (2013). Maximum Relevance/Minimum Redundancy (MRMR). *Encyclopedia of Systems Biology*, 1191–1192. doi: 10.1007/978-1-4419-9863-7\_432.
97. 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.
98. 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.
99. 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.
100. 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.
101. Zhao, Z. A., & Liu, H. (2011). Spectral Feature Selection for Data Mining. doi: 10.1201/b11426.
102. 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.
103. 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.
104. 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.
105. 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.
106. Garnadi, A. D. (2017). Kompresi Citra Menggunakan Truncated Singular Value Decomposition (TSVD), Sebuah Eksplorasi Numerik. doi: 10.31227/osf.io/uqrwe.
107. 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.
108. Chen, Y. (n.d.). Multi-dimensional scaling and MODELLER based evolutionary algorithms for protein model refinement. doi: 10.32469/10355/43046.
109. R., G. (2012). Introduction: Independent Component Analysis. *Independent Component Analysis for Audio and Biosignal Applications*. doi: 10.5772/52324.
110. Data Reduction: Factor Analysis and Cluster Analysis. (n.d.). *Statistical Methods for Geography*, 193–209. doi: 10.4135/9781849209953.n10.
111. Clustering: Practice - Agglomerative Clustering. (2018). doi: 10.4135/9781526469205.
112. 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.
113. 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.
114. 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.
115. 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.
116. Hsu, W.-N., & Glass, J. (2018). Scalable Factorized Hierarchical Variational Autoencoder Training. *Interspeech 2018*. doi: 10.21437/interspeech.2018-1034.
117. 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.
118. 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.
119. Padmanaban, H. (n.d.). Comparative Analysis of Naive Bayes and Tree Augmented Naive Bayes Models. doi: 10.31979/etd.n7jg-e3uh.
120. Li, Z., & Wang, H. (2016). DNA N6-methyladenine modification: a new role for epigenetic silencing in mammalian. *National Science Review*. doi: 10.1093/nsr/nww052.
121. J. Coresh, et al., Prevalence of chronic kidney disease in the United States, JAMA

298 (17) (2007) 2038–2047.

1. A.C. Webster, et al., Chronic kidney disease, Lancet 389 (10075) (2017) 1238–1252.
2. [United States Renal Data System, USRDS Annual Data Report: Epidemiology of](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0003) [Kidney Disease in the United States, National Institutes of Health, National Institute](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0003) [of Diabetes and Digestive and Kidney Diseases, Bethesda, MD, 2017.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0003)
3. [N. Tangri, et al., A predictive model for progression of chronic kidney disease to kidney failure, JAMA 305 (15) (2011) 1553–1559.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0004)
4. [F. Locatelli, L.D. Vecchio, P. Pozzoni, The importance of early detection of chronic](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0005) [kidney disease, Nephrol. Dial. Transplant. 17 (11) (2002) 2–7.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0005)
5. Centers for Disease Control and Prevention, Chronic Kidney Disease Surveillance System—United States. Age-adjusted prevalence of CKD stages 1-4 by gender 1999- 2012. [cited 2016 December 6]; Available from: <http://www.cdc.gov/ckd>.
6. [J. Norouzi, et al., Predicting renal failure progression in chronic kidney disease using integrated intelligent fuzzy expert system, Comput. Math. Methods Med. 2016 (2016) 1–9.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0006)
7. [M. Taal, B. Brenner, Predicting initiation and progression of chronic kidney disease: developing renal risk scores, Kidney Int. 70 (10) (2006) 1694–1705.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0007)
8. [D.C. Cerqueira, et al., A predictive model of progression of CKD to ESRD in a predialysis pediatric interdisciplinary program, Clin. J. Am. Soc. Nephrol. 9 (4) (2014) 06630613.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0008)
9. [E. Choi, et al., Doctor AI: predicting clinical events via recurrent neural networks, Machine Learning for Healthcare Conference, 2016.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0009)
10. [W. De Winter, et al., A mechanism-based disease progression model for comparison of long-term effects of pioglitazone, metformin and gliclazide on disease processes underlying type 2 diabetes mellitus, J. Pharmacokinet. Pharmacodyn. 33 (3) (2006)](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0010) [313–343.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0010)
11. [K. Ito, et al., Disease progression meta-analysis model in Alzheimer's disease,](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0011) [Alzheimer's Dementia 6 (1) (2010) 39–53.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0011)
12. [Y.-Y. Liu, et al., Longitudinal modeling of glaucoma progression using 2-dimen-](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0012)

[sional continuous-time hidden Markov model, International Conference on Medical](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0012) [Image Computing and Computer-Assisted Intervention, Springer, 2013.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0012)

1. [C.H. Jackson, et al., Multistate Markov models for disease progression with classi-](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0013)

[fication error, J. R. Stat. Soc. Ser. D 52 (2) (2003) 193–209.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0013)

1. [J. Zhou, et al., Modeling disease progression via fused sparse group lasso, Proceedings of the 18th ACM SIGKDD international Conference on Knowledge](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0014) [Discovery and Data Mining, ACM, 2012.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0014)
2. [X. Wang, D. Sontag, F. Wang, Unsupervised learning of disease progression models,](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0015)

[Proceedings of the 20th ACM SIGKDD International Conference on Knowledge](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0015) [Discovery and Data Mining, ACM, 2014.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0015)

1. [E. Choi, et al., Constructing disease network and temporal progression model via context-sensitive Hawkes process, Data Mining (ICDM), 2015 IEEE International Conference on, IEEE, 2015.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0016)
2. [R. Ranganath, et al., The Survival Filter: Joint Survival Analysis with a Latent Time Series, UAI, 2015.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0017)
3. [G.T. Obrador, et al., Establishing the global kidney disease prevention network (KDPN): a position statement from the National Kidney Foundation, Am. J. Kidney Dis. 57 (3) (2011) 361–370.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0018)
4. [P. Rucci, et al., A clinical stratification tool for chronic kidney disease progression rate based on classification tree analysis, Nephrol. Dial. Transplant. 29 (3) (2013)](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0019)

[603–610.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0019)

1. [L.A. Stevens, et al., Assessing kidney function—measured and estimated glomerular filtration rate, New Engl. J. Med. 354 (23) (2006) 2473–2483.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0020)
2. [F. Gaspari, et al., Performance of different prediction equations for estimating renal](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0021)

[function in kidney transplantation, Am. J. Transplant. 4 (11) (2004) 1826–1835.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0021)

1. [M. Wainberg, et al., Deep learning in biomedicine, Nat. Biotechnol. 36 (9) (2018)](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0022) [829.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0022)
2. [A. Decruyenaere, et al., Prediction of delayed graft function after kidney trans- plantation: comparison between logistic regression and machine learning methods,](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0023) [BMC Med. Inform. Decis. Mak. 15 (1) (2015) 83.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0023)
3. [M. Kumar, Prediction of chronic kidney disease using random forest machine](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0024) [learning algorithm, Int. J. Comput. Sci. Mob. Comput. 5 (2) (2016) 24–33.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0024)
4. [A. Levin, P.E. Stevens, Early detection of CKD: the benefits, limitations and effects on prognosis, Nat. Rev. Nephrol. 7 (8) (2011) 446.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0025)
5. [A.S. Levey, C. Becker, L.A. Inker, Glomerular filtration rate and albuminuria for detection and staging of acute and chronic kidney disease in adults: a systematic](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0026) [review, JAMA 313 (8) (2015) 837–846.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0026)
6. [A.G. Abraham, et al., Longitudinal formulas to estimate GFR in children with CKD,](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0027) [Clin. J. Am. Soc. Nephrol. 4 (11) (2009) 1724–1730.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0027)
7. [J. Coresh, et al., Decline in estimated glomerular filtration rate and subsequent risk of end-stage renal disease and mortality, JAMA 311 (24) (2014) 2518–2531.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0028)
8. [R.M. Perkins, et al., GFR decline and mortality risk among patients with chronic kidney disease, Clin. J. Am. Soc. Nephrol. 6 (2011) 00470111.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0029)
9. [S.I. Hallan, et al., Age and association of kidney measures with mortality and end-](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0030)

[stage renal disease, JAMA 308 (22) (2012) 2349–2360.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0030)

1. [D. Nitsch, et al., Associations of estimated glomerular filtration rate and albumi-](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0031) [nuria with mortality and renal failure by sex: a meta-analysis, Bmj 346 (2013) f324.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0031)
2. [H.J.L. Heerspink, et al., Estimated GFR decline as a surrogate end point for kidney failure: a post hoc analysis from the Reduction of End Points in Non–Insulin-](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0032)

[Dependent Diabetes With the Angiotensin II Antagonist Losartan (RENAAL) study](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0032) [and Irbesartan Diabetic Nephropathy Trial (IDNT), Am. J. Kidney Dis. 63 (2) (2014)](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0032) [244–250.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0032)

1. [L.A. Inker, et al., GFR decline as an alternative end point to kidney failure in clinical trials: a meta-analysis of treatment effects from 37 randomized trials, Am. J. Kidney Dis. 64 (6) (2014) 848–859.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0033)
2. [T. Greene, C. Teng, J. Ying, Validity and statistical power of alternative eGFR-based](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0034)

[endpoints: a report from an NKF FDA Workshop, J. Am. Soc. Nephrol. 24 (2013)](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0034) [151A.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0034)

1. [M. Baumgarten, T. Gehr, Chronic kidney disease: detection and evaluation, Am.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0035)

[Family Phys. 84 (10) (2011) 1138.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0035)

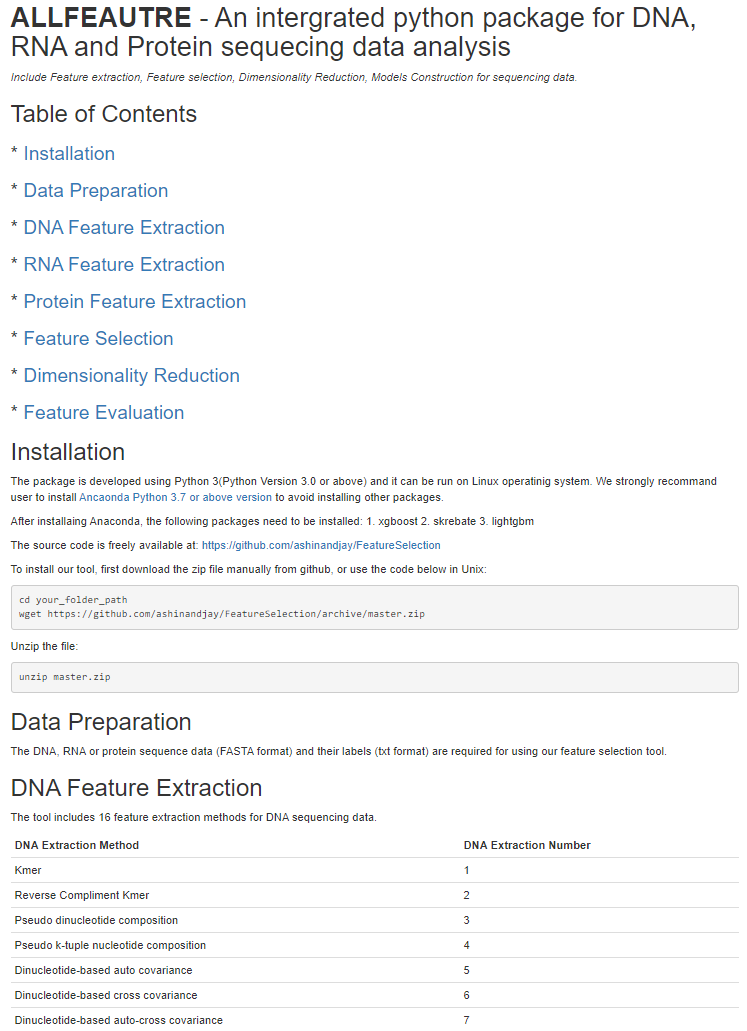
[A. Yadollahpour, Applications of expert systems in management of chronic kidney disease: a review of predicting techniques, Oriental J. Comput. Sci. Technol. 7 (2)](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0036) [(2014) 306–315.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0036)

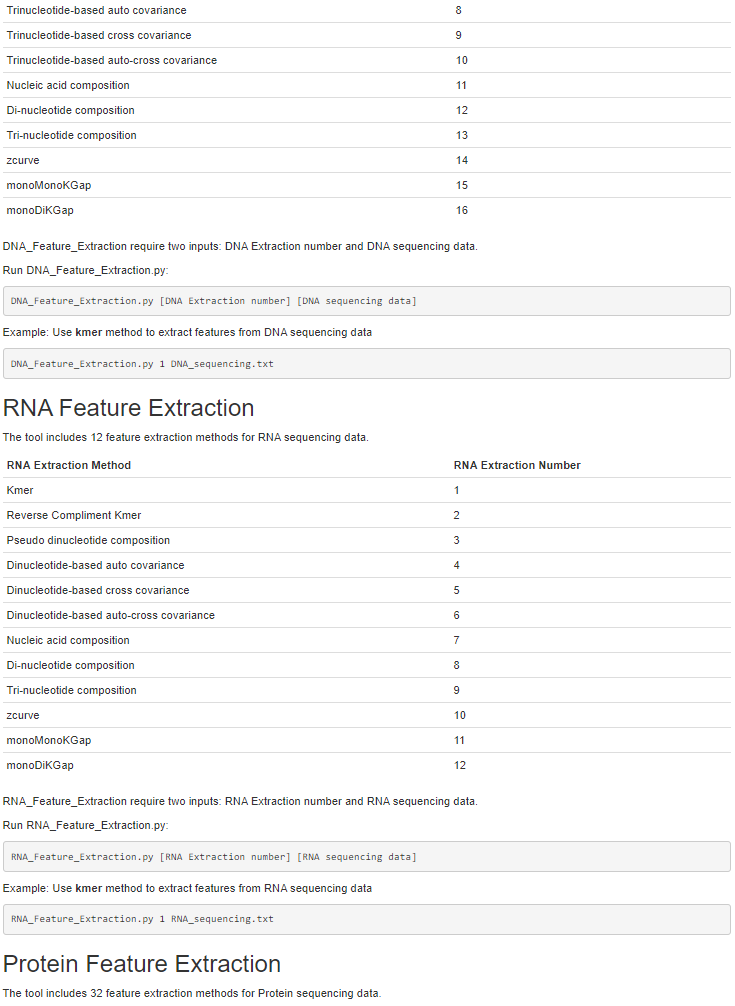
1. [W.H. Organization, ICD-10: International statistical Classification of Diseases and](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0037) [Health-Related Problems, WHO, Geneva, 1992.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0037)
2. [F. Pedregosa, et al., Scikit-learn: machine learning in Python, J. Mach. Learn. Res.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0038)

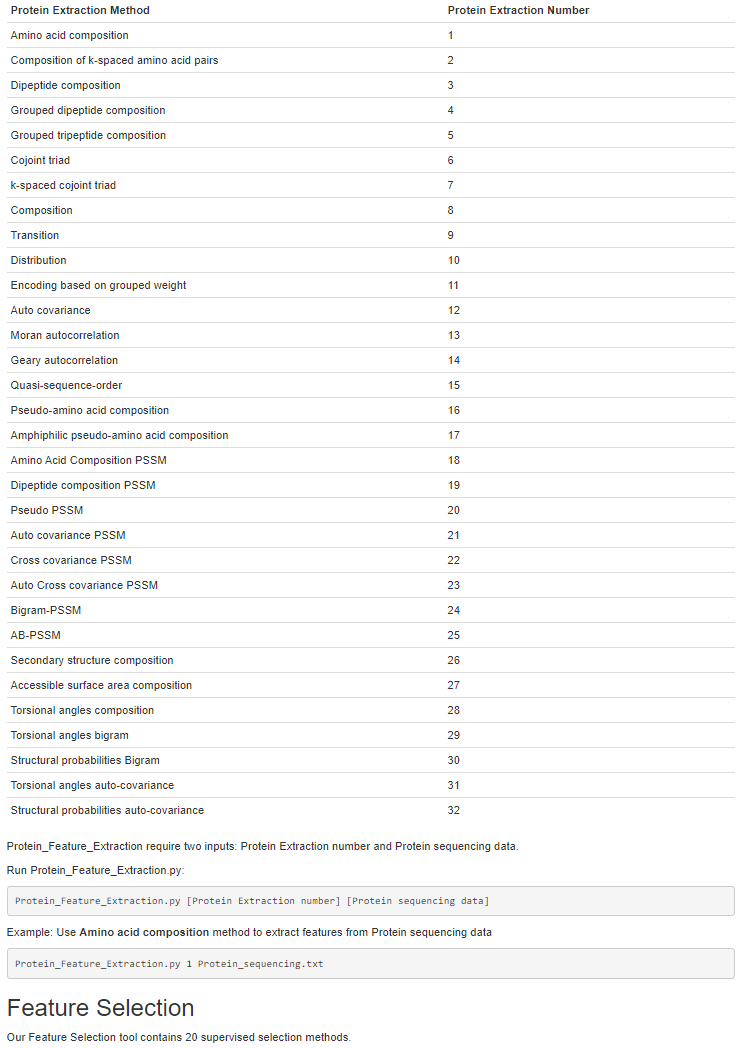
[12 (Oct) (2011) 2825–2830.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0038)

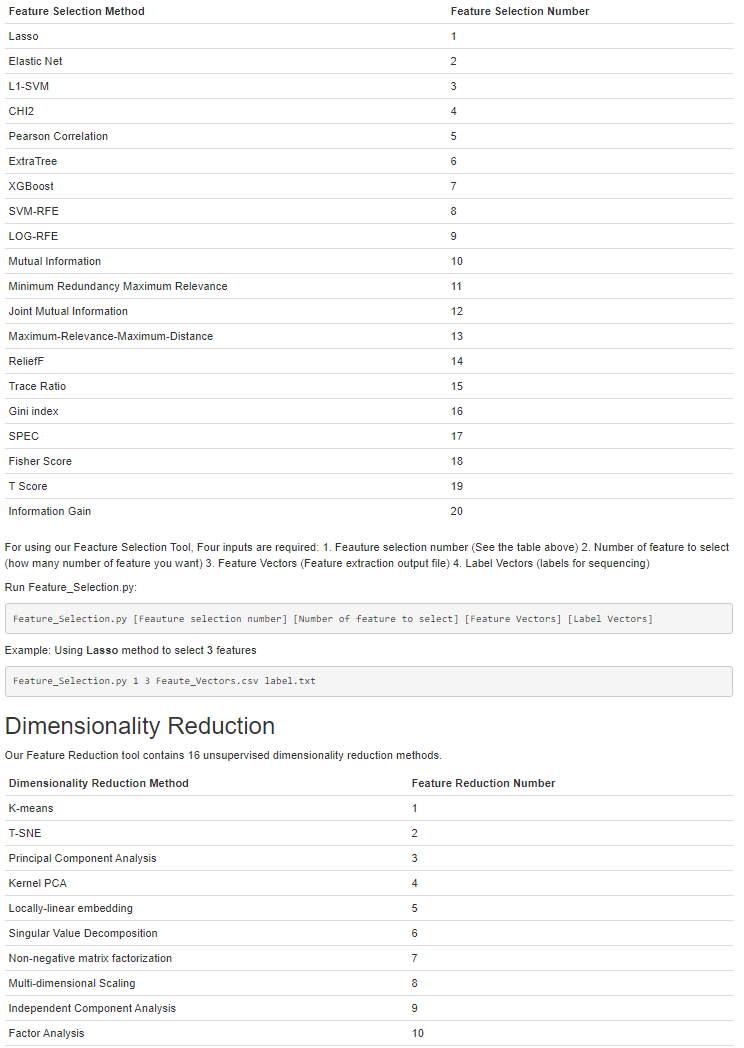
1. [X.Q. Du, et al., DeepPPI: boosting prediction of protein-protein interactions with](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0039) [deep neural networks, J. Chem. Inf. Model. 57 (6) (2017) 1499–1510.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0039)
2. [R. Kazancioğlu, Risk factors for chronic kidney disease: an update, Kidney Int. Suppl. 3 (4) (2013) 368–371.](http://refhub.elsevier.com/S0025-5564(18)30304-3/sbref0040)

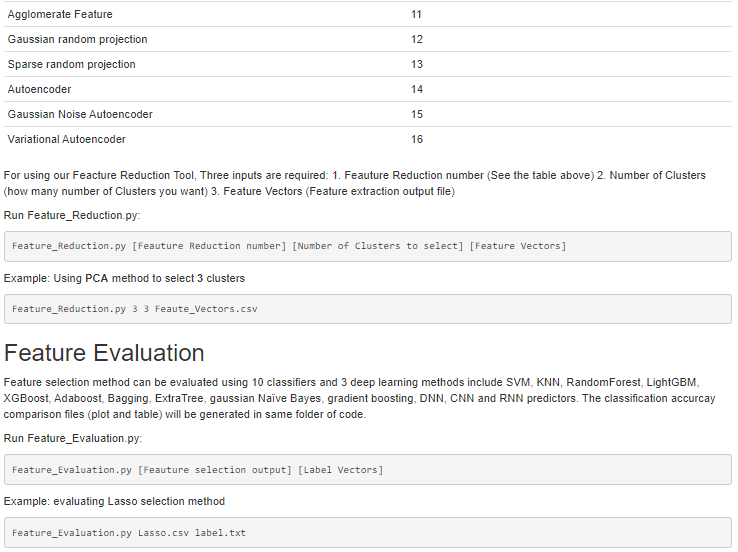
# APPENDIX 1: ALLFUEATRE Tutorial











# APPENDIX 2: DNA and RNA Feature Extraction Methods Description

1. Kmer

For Kmer descriptor, the DNA sequences are represented as the occurrence frequencies of  neighboring nucleic acids. The Kmer (k=3) descriptor can be calculated as:

 (7)

where  is the number of kmer type , while  is the length of a nucleotide sequence.

1. Reverse Compliment Kmer (RCKmer)

The reverse compliment kmer is a variant of kmer descriptor, in which the kmers are not expected to be strand specific. For instance, for a DNA sequence, there are 16 types of 2-mers (i.e. 'AA', 'AC', 'AG', 'AT', 'CA', 'CC', 'CG', 'CT', 'GA', 'GC', 'GG', 'GT', 'TA', 'TC', 'TG', 'TT'), ‘TT’ is reverse compliment with ‘AA’. After removing the reverse compliment kmers, there are only 10 distinct kmers in the reverse compliment kmer approach ('AA', 'AC', 'AG', 'AT', 'CA', 'CC', 'CG', 'GA', 'GC', 'TA').

1. Pseudo dinucleotide composition (PseDNC)

PseDNC is an approach incorporating the contiguous local sequence-order information and the global sequence-order information into the feature vector of the DNA sequence. Given a DNA sequence D, the PseDNC feature vector of D is defined:



 (8)

where  is the normalized occurrence frequency of dinucleotides in the RNA sequence, the parameter  is an integer, representing the highest counted rank (or tier) of the correlation along a RNA sequence, w is the weight factor ranged from 0 to 1,  is called the correlation factor that reflects the sequence-order correlation between all the most contiguous dinucleotides along a RNA sequence, which is defined:

 (9)

where the correlation function is given by

 (10)

where  is the number of physicochemical indices, in this approach, 6 indices reflecting the local RNA structural properties are employed to generate the PseDNC feature vector,  represents the numerical value of the  physicochemical index of the dinucleotide .

1. Pseudo k-tuple nucleotide composition (PseKNC)

PseKNC extends the PseDNC approach by incorporating *k*-tuple nucleotide composition. Given a DNA sequence D, the feature vector of D is defined:



 (11)

where  is the number of the total counted ranks (or tiers) of the correlations along a DNA sequence;  is the frequency of oligonucleotide that is normalized to ; is a weight factor; is given by

 (12)

which represents the  structural correlation factor between all the  most contiguous dinucleotides. The correlation function  is defined by

 (13)

where  is the number of physicochemical indices, in this study, 6 indices reflecting the local DNA structural properties are employed to generate the PseKNC feature vector;  represents the numerical value of the  physicochemical index for the dinucleotide .

1. Dinucleotide-based auto covariance (DAC)

The DAC measures the correlation of the same physicochemical index between two dinucleotides separated by a distance of lag along the sequence, which can be calculated as:

 (14)

where  is a physicochemical index,  is the length of the RNA sequence  means the numerical value of the physicochemical index  for the dinucleotide  at position ,  is the average value for physicochemical index  along the whole sequence:

In such a way, the length of DAC feature vector is , where  is the number of physicochemical indices, and  is the maximum of sequence.

1. Dinucleotide-based cross covariance (DCC)

Given a DNA sequence, the DCC approach measures the correlation of two different physicochemical indices between two dinucleotides separated by lag nucleic acids along the sequence, which can be calculated by:

 (15)

where  are two different physicochemical indices,  is the length of the DNA sequence,  is the numerical value of the physicochemical index  for the dinucleotide .  is the average value for physicochemical index value  along the whole sequence:

 (16)

In such a way, the length of the DCC feature vector is ,  is the maximum of sequence; N is the number of physicochemical indices.

1. Dinucleotide-based auto-cross covariance (DACC)

DACC is a combination of DAC and DCC. Therefore, the length of the DACC feature vector is , where N is the number of physicochemical indices and  is the maximum of sequence.

1. Trinucleotide-based Auto Covariance (TAC)

The Trinucleotide-based auto covariance (TAC) encoding measures the correlation of the same physicochemical index between trinucleotides separated by lag nucleic acids along the sequence, and can be calculated as:

 (17)

where  is a physicochemical index,  is the length of the DNA sequence  means the numerical value of the physicochemical index  for the dinucleotide  at position , is the average value for physicochemical index  along the whole sequence. The dimension of TAC feature vector is , where  is the number of physicochemical indices, and  is the maximum of sequence.

1. Trinucleotide-based Cross Covariance (TCC)

The trinucleotide-based cross covariance (TCC) encoding measures the correlation of two different physicochemical indices between two trinucleotides separated by nucleic acids along the sequence. The TCC encoding can be calculated as:

 (18)

where  are two different physicochemical indices,  is the length of the DNA sequence,  is the numerical value of the physicochemical index  for the dinucleotide .  is the average value for physicochemical index value  along the whole sequence:

 (19)

In such a way, the length of the TCC feature vector is ,  is the maximum of sequence; N is the number of physicochemical indices.

1. Trinucleotide-based auto-cross covariance (TACC)

The trinucleotide-based auto-cross covariance (TACC) encoding is a combination of TAC and TACC encoding. Thus, the dimension of the TACC encoding is , where N is the number of physicochemical indices and  is the maximum of DNA sequence.

1. Nucleic Acid Composition (NAC)

The Nucleic Acid Composition (NAC) encoding calculates the frequency of each nucleic acid type in a nucleotide sequence. The frequencies of all 4 natural nucleic acids (i.e. “ACGT or U”) can be calculated as:

 (20)

where *N*(*t*) is the number of nucleic acid type *t*, while *N* is the length of a nucleotide sequence.

1. Di-Nucleotide Composition (DNC)

The Di-Nucleotide Composition gives 16 descriptors. It is defined as:

 (21)

where is the number of di-nucleotide represented by nucleic acid types *r* and *s*.

1. Tri-Nucleotide Composition (TNC)

The Tri-Nucleotide Composition gives 64 descriptors. It is defined as:

 (22)

where is the number of tri-nucleotide represented by nucleic acid types *r, s* and *t*.

1. zCurve Mathematical Formula (zCurve)

Z-curve theory is often used in genomic sequence analysis. It has got three components in three axis. They are defined as following.

 (23)

Three features will generate using the zCurve method.

1. MonoKGap Theoretical Description (MonoKGap)

When -kgap=n then the  features will exist for DNA and RNA but  features will exist for protein. When -kgap=1, feature structure will be X\_X.

When -kgap=2, feature structure will be X\_X, and X\_\_X. When -kgap=3, feature structure will be X\_X. X\_\_X, and X\_\_\_X. Described with appropriate examples: When -kgap=1 then only sixteen features will exist for DNA and RNA but four hundred (400) features will exist for protein. Features will be numbers of A\_A, A\_C, A\_G, A\_T, C\_A, C\_C, C\_G, C\_T, G\_A, G\_C, G\_G, G\_T, T\_A, T\_C, T\_G, and T\_T of the whole sequence of DNA respectively. When -kgap=2 then only thirty-two features will exist for DNA and RNA but eight hundred (800) features will exist for protein. Features will be numbers of A\_A, A\_C, A\_G, A\_T, C\_A,C\_C, C\_G, C\_T, G\_A, G\_C, G\_G, G\_T, T\_A, T\_C, T\_G,T\_T, A\_\_A, A\_\_C, A\_\_G, A\_\_T, C\_\_A, C\_\_C, C\_\_G, C\_\_T, G\_\_A, G\_\_C, G\_\_G, G\_\_T, T\_\_A, T\_\_C, T\_\_G, and T\_\_T of the whole sequence of DNA respectively.

When -kgap=3 then only forty-eight features will exist for DNA and RNA, but one thousand and two hundred (1,200) features will exist for protein. Features will be numbers of A\_A, A\_C, A\_G, A\_T, C\_A,C\_C, C\_G, C\_T, G\_A, G\_C, G\_G, G\_T, T\_A, T\_C, T\_G,T\_T, A\_\_A, A\_\_C, A\_\_G,A\_\_T, C\_\_A, C\_\_C, C\_\_G, C\_\_T, G\_\_A, G\_\_C, G\_\_G, G\_\_T, T\_\_A, T\_\_C, T\_\_G, T\_\_T, A\_\_\_A, A\_\_\_C, A\_\_\_G, A\_\_\_T, C\_\_\_A, C\_\_\_C, C\_\_\_G, C\_\_\_T, G\_\_\_A, G\_\_\_C, G\_\_\_G,G\_\_\_T, T\_\_\_A, T\_\_\_C, T\_\_\_G, and T\_\_\_T of the whole sequence of DNA respectively.

1. MonoDiKGap Theoretical Description (MonoDiKGap)

When kgap=n then the  features will exist for DNA and RNA, but features will exist for protein.

When -kgap=1, feature structure will be X\_XX.

When -kgap=2, feature structure will be X\_XX, and X\_\_XX.

When -kgap=3, feature structure will be X\_XX, X\_\_XX, and X\_\_\_XX. Described with appropriate examples:

When -kgap=1 then only sixty-four (64) features will exist for DNA and RNA, but eight thousand (8,000) features will exist for protein. Features will be numbers of A\_AA, A\_AC,A\_AG, A\_AT, A\_CA, A\_CC, A\_CG, A\_CT, A\_GA, A\_GC, A\_GG, A\_GT, A\_TA, A\_TC, A\_TG,A\_TT, C\_AA, C\_AC, C\_AG, C\_AT, C\_CA, C\_CC, C\_CG, C\_CT, C\_GA, C\_GC, C\_GG, C\_GT,C\_TA, C\_TC, C\_TG, C\_TT, G\_AA, G\_AC, G\_AG, G\_AT, G\_CA, G\_CC, G\_CG, G\_CT, G\_GA,G\_GC, G\_GG, G\_GT, G\_TA, G\_TC, G\_TG, G\_TT, T\_AA, T\_AC, T\_AG, T\_AT, T\_CA, T\_CC,T\_CG, T\_CT, T\_GA, T\_GC, T\_GG, T\_GT, T\_TA, T\_TC, T\_TG, and T\_TT of the whole sequence of DNA respectively.

When -kgap=2 then only hundred and twenty-eight (128) features will exist for DNA and RNA, but sixteen thousand (16,000) features will exist for protein. Features will be numbers of A\_AA, A\_AC,A\_AG, A\_AT, A\_CA, A\_CC, A\_CG, A\_CT, A\_GA, A\_GC, A\_GG, A\_GT, A\_TA, A\_TC, A\_TG,A\_TT, C\_AA, C\_AC, C\_AG, C\_AT, C\_CA, C\_CC, C\_CG, C\_CT, C\_GA, C\_GC, C\_GG, C\_GT,C\_TA, C\_TC, C\_TG, C\_TT, G\_AA, G\_AC, G\_AG, G\_AT, G\_CA, G\_CC, G\_CG, G\_CT, G\_GA,G\_GC, G\_GG, G\_GT, G\_TA, G\_TC, G\_TG, G\_TT, T\_AA, T\_AC, T\_AG, T\_AT, T\_CA, T\_CC,T\_CG, T\_CT, T\_GA, T\_GC, T\_GG, T\_GT, T\_TA, T\_TC, T\_TG,T\_TT, A\_\_AA, A\_\_AC, A\_\_AG, A\_\_AT, A\_\_CA, A\_\_CC, A\_\_CG, A\_\_CT, A\_\_GA, A\_\_GC,A\_\_GG, A\_\_GT, A\_\_TA, A\_\_TC, A\_\_TG, A\_\_TT, C\_\_AA, C\_\_AC, C\_\_AG, C\_\_AT, C\_\_CA,C\_\_CC, C\_\_CG, C\_\_CT, C\_\_GA, C\_\_GC, C\_\_GG, C\_\_GT, C\_\_TA, C\_\_TC, C\_\_TG, C\_\_TT,G\_\_AA, G\_\_AC, G\_\_AG, G\_\_AT, G\_\_CA, G\_\_CC, G\_\_CG, G\_\_CT, G\_\_GA, G\_\_GC, G\_\_GG,G\_\_GT, G\_\_TA, G\_\_TC, G\_\_TG, G\_\_TT, T\_\_AA, T\_\_AC, T\_\_AG, T\_\_AT, T\_\_CA, T\_\_CC,T\_\_CG, T\_\_CT, T\_\_GA, T\_\_GC, T\_\_GG, T\_\_GT, T\_\_TA, T\_\_TC, T\_\_TG, and T\_\_TT of the whole sequence of DNA respectively.

# APPENDIX 3: Protein Feature Extraction Methods Description

Type 1: Sequence information

1. Amino acid composition (AAC)

The amino acid composition is the fraction of each amino acid type within a protein. The amino acid composition gives 20 features and the fractions of all 20 natural amino acids are calculated as

 (24)

where  is the number of the amino acid type  and  is the length of the sequence.

1. Dipeptide composition (DC)

The dipeptide composition is used to transform the variable length of proteins to fixed length feature vectors.A dipeptide composition has been used earlier by Grassmann et al. and Reczko and Bohr for the development of fold recognition methods.We adopt the same dipeptide composition-based approach in developing a deep neural networks-based method for predicting protein-protein inter-action.The dipeptide composition gives a fixed pattern length of 400.Dipeptide composition encapsulates information about the fraction of amino acids as well as their local order.The dipeptide composition is defined as

 (25)

wherethe number of dipeptide represented by amino acid typeand.

1. -gap dipeptide composition introduces (g-GapDC)

The -gap dipeptide composition introduces (-Gap DC) extracts important intrinsic correlation information of protein sequences in multidimensional space. For a protein ,  of the sequence length C, the calculation process of the -Gap DC can be expressed as

 (26)

 (27)

where  represents the number of residues in the primary structure of the two amino acids, represents the number of occurrences of the -th feature in the -gap dipeptide, and  represents the frequency at which the -th feature in the -gap dipeptide appears in the sequences. When , there is no gap between two adjacent amino acid residues, and when , it means that one amino acid residue is between two adjacent amino acid residues. From equation (9), a protein sequence can be represented as a 400-dimensional feature vector.

1. Grouped di-peptide composition (GDC)

The Grouped di-peptide composition encoding is another variation of the DPC descriptor. It is composed of a total of 25 descriptors that are defined as:

 (28)

where  is the number of the di-peptide type , and  is the length of the sequence.

1. Grouped tri-peptide composition (GTC)

The Grouped tri-peptide composition encoding is also a variation of TPC descriptor, which generates 125 descriptors, defined as:

 (29)

where  is the number of the tri-peptide type , and  is the length of the sequence.

1. Conjoint triad (CT)

First, 20 amino acids are clustered into seven classes based on dipoles and volumes of side chains. Considering the interaction between the amino acid and its vicinal amino acids, the three continuous amino acids are regarded as a unit, so that we can obtain  triad types. We calculate the frequency of occurrence of each triad called . Then the 343-dimensional feature vector is obtained according to equation (6).

 (30)

1. *k*-Spaced Conjoint Triad (KSCTriad)

The *k*-spaced conjoint triad descriptor is based on the conjoint ctriad descriptor,

which not only calculates the numbers of three continuous amino acid units, but also considers the continuous amino acid units that are separated by any *k* residues (The default maximum value of *k* is set to 5). For example, AxRxT is a 1-spaced triad. Thus, the dimensionality of the KSCTriad encoded feature vector is 343 (*k*+1).

1. Composition, transition and distribution (CTD)

8.1 Composition

For CTD, taking hydrophobicity as an example, all amino acids are classified into three categories: polar, neutral, and hydrophobic. The replacement sequence consists of three types, and the composition descriptors of the polar, neutral, and hydrophobic residues of the protein can be calculated as follows:

 (31)

where  is the number of amino acid type  in the coding sequence and  is sequence length.

8.2 Transition

The transition descriptor first converts the original sequence into a replacement sequence, and T includes three characteristics, the dipeptide composition frequency from the polar group to the neutral group and the composition frequency from the neutral group to the polar group. Transitions between the neutral group and the hydrophobicity and these between hydrophobic group and the polar group are defined in the same way. The T descriptor is defined as follows:

 (32)

Where  and  are the  and  dipeptide frequency, respectively.  is the sequence length.

8.3 Distribution

For each group (polar, neutral and hydrophobic), the D descriptor can generate five values. We obtain the position of the first, 25%, 50%, 75% and 100% of the specific encoded group sequence and then divided the position by the whole sequence. Given sequence MTTTVPKVFAFHEF. It can be represented as '32223213323213' according to Hydrophobicity\_PRAM900101. '1' represents polar, '2' represents neutral, '3' represents hydrophobicity. Take '3' for example, there are 6 residues encoded '3'. The first '3' is 1. The second '3' is . The third '3' is . The fourth '3' is . The fifth '3 is . The position in the first, the second, the third, the fourth, the fifth '3' of whole sequence are 1, 1, 8, 9, 14, respectively. So the distribution descriptor for '3' are ,,,,.

The C descriptor generates a 39-dimensional feature vector, the T descriptor generates a 39-dimensional feature vector, and the D descriptor generates a 195-dimensional feature vector. For each protein sequence, the CTD generates a 273-dimensional feature vector.

1. Encoding based on grouped weight coding (EBGW)

Studies have shown that different amino acids have different physical and chemical properties. The EBGW can capture the sequence and physicochemical information based on grouped situation. These amino acids are classified into four categories:

neutral and hydrophobic amino acids 

neutral and polarity amino acids 

acidic amino acids 

basic amino acids .

Thus, we can get three combinations, each of which can partition the 20 amino acid residues into two disjoint group:  vs , or vs, and  vs . Let be a protein sequence, we can transform it into three binary sequences by three homomorphic mapswhich are defined as follows:

 (33)

 (34)

 (35)

where , and we call as characteristic sequence of the protein sequence.

Then , , are three binary sequences of length *L*. These sequences are divided into a number of sub-sequences of increasing length successively. A fixed parameter *N* is set and the sub-sequence can be expressed as, where represents the integer operator. Calculate the frequency of 1 in each sub-sequence, each  can be converted into an *N*-dimensional feature vector. To sum up, for a protein sequence *P* with length *L*, a 3*N*-dimension vector can be obtained.

Type 2: Physicochemical information

1. Auto covariance (AC)

Suppose a protein sequence P with *L* amino acid residues; i.e.

 (36)

where  represents the amino acid residue at the sequence position 1,  the amino acid residue at position 2 and so forth.

The AC approach measures the correlation of the same property between two residues separated by a distance of along the sequence, which can be calculated as:

 (37)

where is a physicochemical index, is the length of the protein sequence, means the numerical value of the physicochemical index for the amino acid  at position ,  is the average value for physicochemical index along the whole sequence. In such a way, the length of AC feature vector is , where *N* is the number of physicochemical indices extracted from AAindex; ** is the maximum of sequence.

1. Moreau-Broto autocorrelation (Morean-Broto)

Moreau-Broto autocorrelation descriptor is defined as:

 (38)

where , and  indicate the  and the  amino acids of the protein sequence, respectively.  and indicate the normalized physicochemical values of  and . The  is the parameter that needs to be adjusted.

1. Moran autocorrelation (Moran)

Moran autocorrelation descriptor is defined as:

 (39)

where  represents the mean value of whole protein sequence for specific physicochemical property.

1. Geary autocorrelation (Geary)

Geary autocorrelation descriptor is defined as:

 (40)

1. Quasi-sequence-order descriptors (QSO)

The sequence order features can also be used for representing amino acid distribution patterns of a specific physicochemical property along protein or peptide sequence. These descriptors are derived from both the Schneider-Wrede physicochemical distance matrix and the Grantham chemical distance matrix between each pair of the 20 amino acids. Theth rank sequence-order-coupling number is defined as

 (41)

whereis the distance between the two amino acids at positionand. Maxlag is the maximum lag and the length of the protein must be not less than maxlag. The maxlag is equal to 30 in the experiment. For each amino acid type, the type-1 quasi-sequence-order descriptor can be defined as

 (42)

where is the normalized occurrence of amino acid type-1 and is a weighting factor. The type-2 quasi-sequence-order is defined as

 (43)

In addition to the Schneider-Wrede physicochemical distance matrix used by Chou et al., another chemical distance matrix by Grantham is also used here. The sequence-order features produce a total ofdescriptors.

1. Pseudo-amino acid composition (PseAAC)

Pseudo-amino acid composition are utilized to extract the physicochemical information. Pseudo-amino acid composition (PseAAC) represents the composition information of the protein and sequence order information. The feature vector of PseAAC can be represented:

 (44)

where the first 20-dimentional feature vector represents the amino acid composition information, and the latter  dimensional vector represents the order information of protein sequence.  is the length of amino acid sequence.

  (45)

where  is the normalized frequency of 20 amino acids in protein.  is the layer sequence correlation factor calculated according to the equation (2). Sequence correlation factor can be obtained from the hydrophobicity, hydrophilicity, and side-chain mass of the amino acid.

1. Amphiphilic pseudo amino acid composition (APAAC)

APAAC (http://www. csbio. sjtu. edu. cn/bioinf/PseAAC/type2. htm) are also called type-2 pseudoamino acid composition. The definitions of these qualities are similar to PAAC descriptors. First, two variables are derived from the original hydrophobicity valuesand hydrophilicity valueof 20 amino acids

 (46)

 (47)

whereand,the hydrophobicity and hydrophilicity correlation functions, are defined respectively as

 (48)

where sequence order factors can be defined as

 (49)

 (50)

 (51)

Then a set of descriptors called "Amphiphilic Pseudo Amino Acid Composition” are defined as

 (52)

 (53)

whereis the weight factor and is taken as, andis equal to 30 in the experiment. So, we produce a total ofdescriptors.

Type 3: Evolutionary information

In order to obtain the evolutionary information of the amino acid sequence, all the protein sequences in the dataset are compared with the non-redundant database SwissProt using the PSI-BLAST program. The program can search sequences based on the iterative BLAST search method. Evolutionary information in the position specific scoring matrix (PSSM) plays an important role in biological system analysis.

During the running process, the parameter E-value threshold of PSI-BLAST is set to 0.001, the maximum number of iterations is set to 3, and the remaining parameters are set by default. Then PSSM of each protein sequence is obtained. For a protein sequence whose length is , PSSM is shown in equation (49).

  (54)

where each row of the PSSM represents a log likelihood score for amino acid substitutions occurring at corresponding positions in the query sequence. Where  represents the  position of query sequence being mutated to type  during evolution process. The scores are positive integers or negative integers. A positive integer indicates that more mutations have occurred in the alignment and a negative integer indicates that fewer substitutions have occurred in the alignment.

1. Amino acid composition PSSM (AAC-PSSM)

In this calculating process of amino acid composition PSSM (AAC-PSSM), the PSSM is standardized by the logic function. PSSM elements are map to the interval [0,1].

  (55)

PSSM are converted to feature vector by AAC-PSSM via equation (51)

 (56)

where ,  represents the composition information of the  amino acid residue, which is the average score of  amino acid in PSSM.

1. Dipeptide composition PSSM (DPC-PSSM)

The PSSM contains evolutionary information. ACC-PSSM only represents the composition information from PSSM, and loses the order information, which is insufficient to fully represent the evolutionary information. Dipeptide composition PSSM (DPC-PSSM) can reflect the sequence-order information in the PSSM, which converts the character signal into the numerical signal, and the extracted feature vector can be expressed as

 (57)

where , the dimension of DPC-PSSM is 400.

1. Bi-gram PSSM (Bi-PSSM)

For Bi-gram PSSM, the frequency of the transition from the  amino acids to the  amino acids is calculated:

 (58)

Therefore, there are 400 possible cases for , then the Bi-gram PSSM eigenvector for each protein sequence is:

 (59)

1. Auto covariance PSSM (AC-PSSM)

AC-PSSM can transform the PSSMs of different lengths into fixed-length vector. The AC variable measures the correlation of the same property between two residues separated by a distance of lag along the sequence, which can be calculated as:

 (60)

whereis one of the residues,is the length of the protein sequence,is the PSSM score of amino acidat position,is the average score for amino acid  along the whole sequence:

 (61)

In such a way, the number of AC variables can be calculated as, where  is the maximum of sequence.

1. Cross covariance PSSM (CC-PSSM)

CC-PSSM can transform the PSSMs of different lengths into fixed-length vectors. The CC variable measures the correlation of two different properties between two residues separated by lag along the sequence, which can be calculated by:

 (62)

whereare two different amino acids andis the average score for amino acidalong the sequence. Since the CC variables are not symmetric, the total number of CC variables is.

1. Auto-cross covariance PSSM (ACC-PSSM)

ACC-PSSM as one of the multivariate modeling tools, can transform the PSSMs of different lengths into fixed-length vectors by measuring the correlation between any two properties. ACC results in two kinds of variables: AC between the same property, and cross-covariance (CC) between two different properties. Each protein sequence is represented as a vector of either AC variable or ACC variable that is a combination of AC and CC.

1. Pseudo PSSM (PsePSSM)

According to the pseudo amino acid composition, we obtain the PsePSSM feature vector:

  (63)

where, each protein sequence can generate the dimensional feature vector. The first 20-dimensional vector represents the composition information of the PSSM matrix, and the remaining  dimensional feature vector represents the order evolutionary information. PsePSSM can transform an inconsistent protein sequence into a consistent numerical vector by feature extraction.

1. AB-PSSM

AB-PSSM is based on the averaged PSSM profiles over blocks, each with 5 percent of a sequence. Thus, a protein sequence, regardless of its length, is divided into 20 blocks and each block consists of 20 features (derived from the 20 columns in PSSMs). Mathematically, for the  block, the feature  is a  dimensional feature vector, which is generated by using the following equation:

 (64)

where  is the size of the  block, which is 5 percent of the length of a sequence and Pej i is a  vector extracted from the PSSM profile at the  position in the  block. For each sequence, there are a total of 20 blocks; therefore, the final feature is a 400-dimensional vector.

Type 4: Structural information

1. Secondary structure composition (SSC)

This feature is the normalized count or frequency of the structural motifs present at the amino-acid residue positions. There are three types of motifs: *α*-helix (H), *β*-sheet (E) and random coil (C). SPIDER2 returns a vector *SS* of dimension *L* × 1 containing this information. Thus, we can define this feature as following:

 (65)

where, *L* is the length of the protein and

 (66)

where, *SSj* is the structural motif at position *j* of the protein sequence and *fi* is one of the 3 different motif symbols.

1. Accessible surface area composition (ASA)

The accessible surface area composition is the normalized sum of accessible surface area defined by:

 (67)

where ASA is the vector of accessible surface area of dimension *L* × 1 containing the values of accessible surface area for all the amino acid residues.

1. Torsional angles composition (TAC)

Four different types of torsional angles: *ϕ*, *ψ*, *τ* and *θ* are returned by SPIDER2 for each residue. First, we convert each of them into radians from degree angles and then take sign and cosine of the angles at each residue position. Thus, we get a matrix of dimension. We denotethis matrix by *T*. Torsional angles composition is defined as

 (68)

1. Torsional angles bigram (TA-bigram)

The Bigram for the torsional angles is similar to that of the PSSM matrix and is defined as:

 (69)

1. Structural probabilities bigram (SP-bigram)

Structural probabilities for each position of the amino-acid residue are given in the SPD2 file as a matrix of dimension *L* × 3, which we denote by *P*. The Bigram of the structural probabilities is similar to that of PSSM matrix and is defined as:

 (70)

1. Torsional angles auto-covariance (TAAC)

This feature is also derived from the torsional angles and is defined as:

 (71)

1. Structural probabilities auto-covariance (SPAC)

This feature is also derived from the structural probabilities and is defined as:

 (72)

# APPENDIX 4: Plan of Study

# APPENDIX 5: Curriculum Vitae