Mathematical Biosciences and Engineering

MBE, 21(3): 3798–3815.
DOI: 10.3934/mbe.2024169
Received: 27 November 2023
Revised: 29 January 2024
Accepted: 05 February 2024
Published: 19 February 2024

http://www.aimspress.com/journal/MBE

Research article

SoftVoting6mA: An improved ensemble-based method for predicting DNA N6-methyladenine sites in cross-species genomes

Zhaoting Yin[1], Jianyi Lyu[1], Guiyang Zhang[1], Xiaohong Huang[1], Qinghua Ma[2,3] and Jinyun Jiang[1,*]

[1] College of Information Science and Engineering, Shaoyang University, Shaoyang 422000, China
[2] College of Information Science and Engineering, Hohai University, Nanjing 210000, China
[3] Faculty of Information Technology, University of Jyvaskyla, Jyvaskyla, Finland

* **Correspondence:** Email: tjjjy86@ecjtu.edu.cn.

Abstract: The DNA N6-methyladenine (6mA) is an epigenetic modification, which plays a pivotal role in biological processes encompassing gene expression, DNA replication, repair, and recombination. Therefore, the precise identification of 6mA sites is fundamental for better understanding its function, but challenging. We proposed an improved ensemble-based method for predicting DNA N6-methyladenine sites in cross-species genomes called SoftVoting6mA. The SoftVoting6mA selected four (electron–ion-interaction pseudo potential, One-hot encoding, Kmer, and pseudo dinucleotide composition) codes from 15 types of encoding to represent DNA sequences by comparing their performances. Similarly, the SoftVoting6mA combined four learning algorithms using the soft voting strategy. The 5-fold cross-validation and the independent tests showed that SoftVoting6mA reached the state-of-the-art performance. To enhance accessibility, a user-friendly web server is provided at http://www.biolscience.cn/SoftVoting6mA/.

Keywords: DNA N6-methyladenine; convolution neural network; soft voting; cross-species; feature fusion; webserver

1. Introduction

DNA as one of four major types of macromolecules that is not only a fundamental component of life but also plays a critical role for holding all the generic information about growth, disease, and death. DNA modification is an epigenetics process where specific functional groups are added to the

nucleotide residues. Known DNA modifications include 5-methylcytosine (5mC), 5-hydroxymethylcytosine (5hmC), N4-methylcytosine (m4C), and N6-methyladenine (6mA) [1,2]. The DNA 6mA refers to methylation of the 6th position on the adenines in which the methyl functional group is added to adenine residue with the help of the MT-70 methyltransferase family [3]. For a long period, the 6mA was thought as the most prevalent DNA modification in prokaryotes [4], and present with a low prevalence in the eukaryotes. However, over the last ten years, the 6mA has been discovered in numerous eukaryotes, including *C. elegans* [5,6], rice [7], zebrafish [8], and humans [9]. The 6mA is a reversible modification, which is formed by the writer methyltransferases and is removed by the eraser demethylases. Increasing evidence has shown that DNA 6mA functioned as an epigenetic mark regulating some biological processes such as DNA replication and repair, cell defense, and gene expression [10]. For example, the 6mA was reported to be associated with Alzheimer's disease [11]. DNA 6mA was found to be involved in hepatocellular carcinoma development [12] and to regulate drug resistance of triple negative breast cancer [13]. Therefore, precisely detecting 6mA is of importance to further exploring its functions.

Due to the key roles of 6mA in the cellular process, enormous attention was paid to developing methods for detecting 6mA sites. These methods were grouped into two categories: The laboratory and the bioinformatics methods. The former includes the 6mA-immunoprecipitation sequencing (6mA-IPseq), restriction enzyme-based 6mA sequencing (6mA-REseq), high-performance liquid chromatography coupled with tandem mass spectrometry (HPLC-MS/MS), and single-molecule real-time sequencing (SMRT) [3]. For instance, the 6mA-IPseq used a specific 6mA antibody to enrich methylated genomic fragments. The advantage of this 6mA-IPseq was that the cost was low, and the limitation was that the 6mA antibody preferred the unmodified adenine and thus was unable to precisely locate 6mA. In addition, the 6mA-IPseq suffered from bacterial DNA containing 6mA [3,14]. In the HPLC-MS/MS, the purified DNA samples were first digested by commercial enzymes, then separated by the chromatographic separation system, next ionized by atmospheric pressure ionization (API) techniques, and finally detected by the MS/MS based on mass-to-charge ratio. The HPLC-MS/MS was subject to laborious conditions [3]. Therefore, these laboratory methods are not only cumbersome, time-consuming, and laborious, but also costly.

The bioinformatics method is to exploit the annotated 6mA sequences to learn a classifier and then predict unannotated sequences using the trained classifier. Opposite to the laboratory methods, the bioinformatics methods are simple and high-throughput. With advances in big data and artificial intelligence, more and more attention is given to developing bioinformatics methods for 6mA detection. So far, there are no less than twenty bioinformatics methods developed for 6mA prediction [15–20]. The bioinformatics methods are further divided into the traditional machine learning-based and the deep learning methods. The former depends on both the representations and the learning algorithms [21]. The representations for 6mA sequences include electron–ion-interaction pseudo potential (EIIP), nucleotide chemical property (NCP), dinucleotide physicochemical properties (DPCP), Kmer, trinucleotide physicochemical properties (TPCP), pseudo k-tuple nucleotide composition (PseKNC), One-hot encoding, and ring-functions of hydrogen chemical (RFHC). The learning algorithms included support vector machine (SVM), XGBoost, gradient boosting (GB), and random forest (RF). For example, the iDNA6mA-PseKNC is a bioinformatics method for mouse 6mA site prediction, which used PseKNC to represent DNA sequence and SVM as the learning algorithm [22]. The iDNA6mA-Rice used three types of representations and employed random forest as the learning algorithm [23]. Single representation might be insufficient to characterize 6mA, and a single learning

algorithm might be unable to learn the nature of 6mA. Combining multiple representations or multiple learning algorithms might be an ideal choice. For example, the 6mA-RicePred used four types of representations and employed feature selection to improve predictive performance [24]. The i6mA-Vote integrated multiple classifiers for 6mA sites prediction [25]. The i6ma-stack not only optimized representations from multiple features by recursive feature elimination with cross-validation, but also combined the outputs of SVM, logistic regression (LR), RF, and naive Bayes (NB) as inputs to the final classifier SVM [26]. Compared to traditional machine learning algorithms, deep learning is of powerful fitting ability, especially in the context of big data. The commonly used components of deep learning include convolutional neural network (CNN), recurrent neural network (RNN), long short-term memory (LSTM), attention mechanism, and transformer. For example, the CNN was used alone for 6mA site prediction [17,27,28]. The CNN might be combined with the LSTM for 6mA site prediction [19,29,30]. The various architecture of the CNN and LSTM resulted in various predictive performance. What is the best architecture remained unknown. Le and Ho [31] utilized the transformer and the CNN to detect DNA 6mA sites across species. The predictive ability of deep learning depends on the number of training samples. The small number of samples is easy to cause overfitting. Therefore, we focused on the traditional machine learning algorithms for 6mA prediction.

LightGBM demonstrates its superiority in terms of performance [7,32], as was evidenced by its successful application in predicting viral protein classification by Bao et al. [33]. Therefore, we chose LightGBM as a base model in feature selection. The ensemble learning approach has the potential to enhance prediction performance [34]. We proposed an improved ensemble-based method (called SoftVoting6mA) for predicting DNA N6-methyladenine sites in cross-species genomes. Namely, we combined and optimized representations and multiple learning algorithms to improve predictive performance. The SoftVoting6mA used EIIP, One-hot encoding, Kmer, and PseDNC were used to represent 6mA sequences, and we combined four types of machine learning or deep learning algorithms.

2. Methods

2.1. Benchmark datasets

The cross-species benchmark datasets originated directly from the published work for 6mA-Bert [31]. The benchmark datasets consisted of the training and the independent datasets. The former contained 2500 positive and 2500 negative samples, while the latter comprised 268 positive and 216 negative samples. The process of collecting data was briefly described as follows. Feng et al. [22] compiled the mouse 6mA datasets from the MethSMRT database [35] with the accession in the Gene Expression Omnibus (GEO) GSE71866. After removing the sequences with more than 30 modQV and more than 0.8 sequence identity, Feng et al. obtained 1934 positive and 1934 negative samples. Chen et al. [36] constructed a rice 6mA dataset from the GEO with accession number GSE103145 [7]. Xu et al. [16] fused the mouse and the rice 6mA datasets. Xu et al. further used the cluster programming CD-Hit [37] to reduce sequence homology, and subsequently divided all the samples into the training and the independent datasets. All samples were 41 bp nucleotides long with adenine at the center. Samples with 6mA modification sites were considered as positive samples and otherwise are negative samples. All data can be downloaded from the website: https://github.com/yinzhaoting/Softvoting-6mA.

2.2. Methodology

The framework of our proposed method SoftVoting6mA is shown in Figure 1. Initially, the SoftVoting6mA used as many as 15 types of representations to encode positive or negative samples, namely, EIIP, One-hot encoding, NCP, Kmer, Pseudo Dinucleotide Composition (PseDNC), RCKmer, Dinucleotide-based Auto-Cross Covariance (DACC), Dinucleotide-based Cross Covariance (DCC), DPCP, Accumulated Nucleotide Frequency (ANF), Nucleic Acid Composition (NAC), Dinucleotide-based Auto Covariance (DAC), Trinucleotide-based Auto Covariance (TAC), Trinucleotide-based Auto-Cross Covariance (TACC), and Trinucleotide-based Cross Covariance (TCC). We assessed the predictive performance by 5-fold cross-validation, and chose four top-performing types of representations. Subsequently, we utilized a forward searching strategy to optimize combination of representations. We compared six commonly used machine learning algorithm as well as a CNN-based method. Six machine learning algorithms are LightGBM [38], XGBoost [39], GB [40], RF [41], LR [42], and decision tree (DT) [43]. In these machine learning algorithms, we use default parameters. In the CNN-based method, One-hot encoding was employed as representation [44], followed by a convolutional layer, a pooling layer, and a fully connected layer. Sigmoid was utilized as the activation function for the final fully connected layer, which stand for the probability of predicting as positive samples [45]. We sorted seven algorithms in descending order of performance. We further used backward strategy to optimize algorithms combination by 5-fold cross-validation. Finally, we used optimal combination of representations, ensemble multiple classifiers by soft voting for decision.

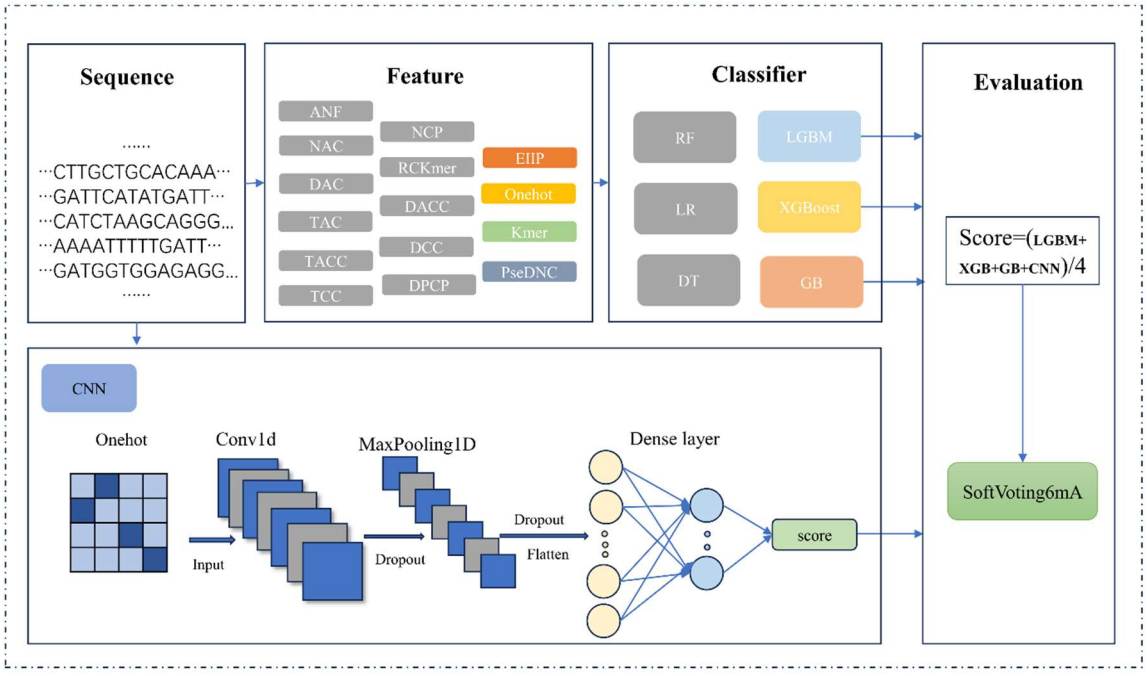

Figure 1. Flowchart of the SoftVoting6mA. The SoftVoting6mA first computed 15 representations, and then selected optimized representations by 5-fold cross validation. Subsequently, the SoftVoting6mA selected the model. Finally, the SoftVoting6mA used the soft voting to combine multiple classifiers for decision.

2.2.1. EIIP

The EIIP [46] is a very simple but much effective numerical representation of nucleotide sequences, which is commonly employed to address function-related problems such as enhancer and N4-methylcytosine prediction [47,48]. In this encoding, each nucleotide was assigned to a value determined by its electron-ion interaction potential, reflecting the nucleotide's electron density. This encoding, as described in Eq (1), was computed as a vector of the same length as the original sequence.

$$V = \left(E_{d_1}, E_{d_2}, \cdots, E_{d_L}\right), \tag{1}$$

where L was the length of this DNA sequence, $d_i (i = 1,2, \ldots, L)$ denote the nucleotides that make up the DNA sequence, E_{d_i} denoted the EIIP value of the nucleotide d_i.The EIIP values of the A, C, G, and T were 0.1260, 0.1340, 0.0806 and 0.1335, respectively.

2.2.2. One-hot encoding

One-hot encoding was initially applied in the field of computer science and later was adopted extensively in the field of bioinformatics as a popular technique for handling categorical data [49]. In the One-hot encoding, only a bit was 1 and others were zeros. In the study, A, C, G, and T were defined as (1,0,0,0), (0,1,0,0), (0,0,1,0), and (0,0,0,1), respectively. The One-hot encoding is a very intuitive representation.

2.2.3. Kmer

Kmer is a widely used sequence-based encoding method in bioinformatics. The Kmer is defined as the occurrence frequency of consecutive k nucleotides in a sequence, which was computed by the Eq (2).

$$f(t) = \frac{N(t)}{N}, \tag{2}$$

where t was an element of the set of consecutive k nucleotides, $N(t)$ was the occurrence number of t, and N was the total number. For example, the sequence "CATCGCAT" can be partitioned into six 3mers: "CAT", "ATC", "TCG", "CGC", "GCA", and "CAT". The frequency of the 3-mer "CAT" is 1/3 and the other five are 1/6. In this study, we applied 1mer, 2mer, and 3mer to compute representation, which resulted in 84-dimensional vectors.

2.2.4. PseDNC

Inspired by the pseudo-amino acid composition (PseAAC) [50] which is a popular representation of protein sequences, Chen et al. [51] proposed PseDNC to represent DNA sequence. The PseDNC fused information about both order as well as frequency and thus made up for the loss of sequence-order information in the Kmer. Due to its powerful representation, the PseDNC was widely used for identifying splicing sites [52], recombination spots[51], and N6-methyladenosine site prediction [53] The PseDNC of a DNA sequence $R_1 R_2 \ldots R_n$ was computed by

$$d_u = \begin{cases} \frac{f_u}{\sum_{i=1}^{16} f_i + w\sum_{j=1}^{\lambda}\theta_j} & 1 \le u \le 16 \\ \frac{w\theta_{u-16}}{\sum_{i=1}^{16} f_i + w\sum_{j=1}^{\lambda}\theta_j} & 16 < u \le 16+\lambda \end{cases}, \quad (3)$$

where f_u denoted the normalized occurrence frequencies of dinucleotides, λ was the user-defined counted ranks ($\lambda \le n-1$), w was a weight factor, and θ_j was calculated by

$$\theta_j = \frac{1}{n-1-j}\sum_{i=1}^{n-1-j}\left[\Theta(R_iR_{i+1}) - \Theta\left(R_{i+1}R_{i+1+j}\right)\right]^2, \quad (4)$$

where $j = 1,2,3,\cdots,\lambda$. The function $\Theta(R_iR_{i+1})$ was defined as the properties of dinucleotide R_iR_{i+1}.

2.2.5. CNN

The CNN is a deep learning technique primarily used for image processing and computer vision tasks. It is a feed-forward neural network with multiple convolutional and pooling layers, as well as fully connected layers. CNN can learn complex patterns and structures. It can also handle input images of different sizes. The convolutional layer uses a set of filters (also called convolutional kernels) to detect pattern. The pooling layer can reduce the complexity of the model while preserving the main characteristic of the input data. The flattening layer followed the pooling layers so that it could link to the fully connected layers. In the final fully connected layer, we use the sigmoid function, which resulted in a probability between 0 and 1. We adopted a grid search approach. Please refer to Tables S1–S3 in the Supplementary file for specific screening. The parameters of each layer of the model are listed in Table 1.

Table 1. The shapes of outputs and the numbers of parameters in the CNN model.

Layers	Output Shape	Parameters
InputLayer	(41, 4)	None
Conv1D	(41, 128)	Number of Filters = 128, Kernel Size = 9
Dropout	(41, 128)	Dropout Ratio = 0.5
MaxPooling1D	(20, 128)	None
Flatten	2560	None
Dropout	2560	Dropout Ratio = 0.5
Dense	8	Number of neurons = 8, activation = 'relu'
Dense	1	Number of neurons = 1, activation = 'sigmoid'

2.3. Soft voting

The voting strategy is widely used in the field of machine learning to integrate multiple classifiers for better prediction accuracy. Balancing the performance between different classifiers, the voting strategy also enable better robustness of the integrated model. There are two categories of voting strategy: Hard voting and soft voting. The former adopted the principle that the minority obeys the majority. For example, if there were 5 classifiers, of which 3 outputted 1, and 2 outputted 0, the final

decision of the hard voting was 1. The soft voting computed average outputting probability of each classifier over all the classifiers. The class with the best average probability was determined as the final class. For example, if there were three classifiers, the outputting probabilities of judging as 1 were 0.9, 0.3 and 0.6, respectively, while the probabilities of judging as 0 were 0.1, 0.7 and 0.4. The average probability for 1 were 0.6, while the average probability for 0 was 0.4. Therefore, the final decision of the soft voting was 1. We used four base classifiers and averaged the prediction probabilities of four classifiers as the final prediction probability, as shown in Figure 2.

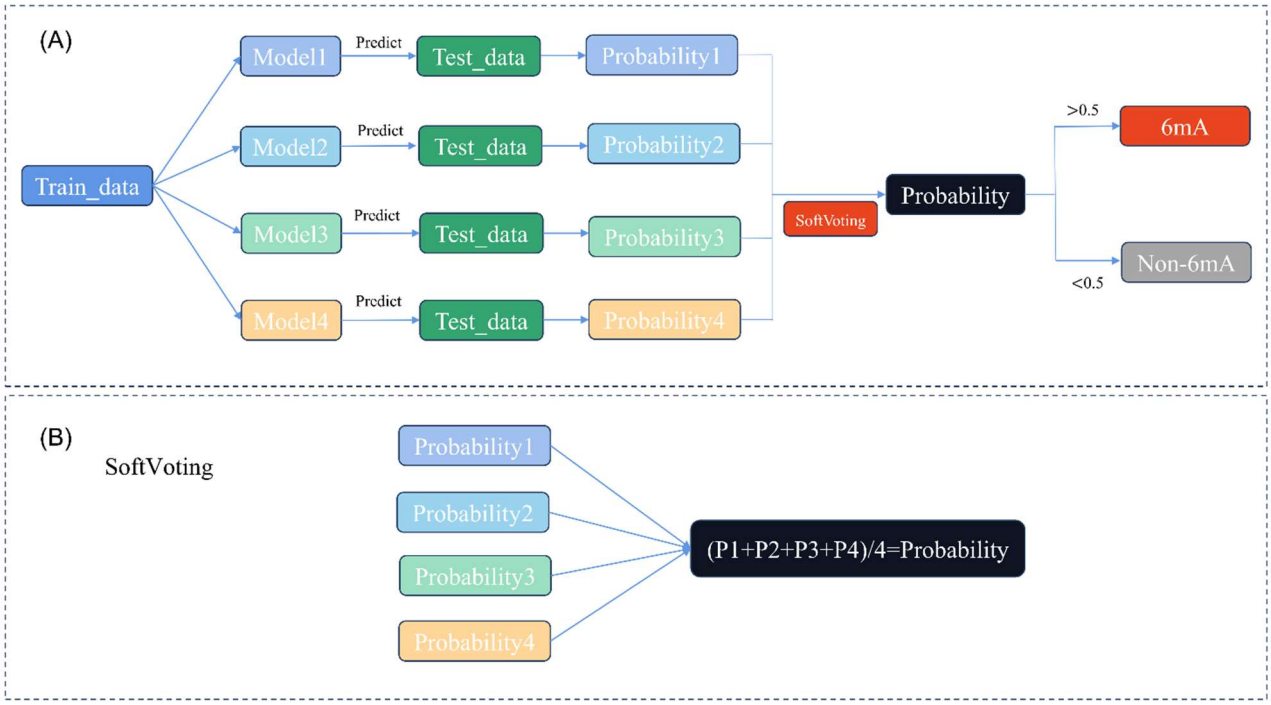

Figure 2. (A) The flowchart of the soft voting process. (B) Specific steps for SoftVoting. P1 stands for Probability1, P2 stands for Probability2, P3 stands for Probability3 and P4 stands for Probability.

2.4. Evaluation metrics

We adopted both the 5-fold cross-validation and the independent test to examine our method. In the 5-fold cross-validation, the training dataset was randomly grouped into five parts, of which four were used for training and one was used for testing. The process was repeated five times until each sample was used for testing only a time and for training four times. In the independent test, the testing data did not appear in the training set. The purpose of the independent test is to examine the generalization ability of our method. We employed sensitivity (SN), specificity (SP), accuracy (ACC), Matthew correlation coefficient (MCC), F1-score, and area under the curve (AUC) to measure performances. The first five metrics were mathematically defined as follows:

$$\mathrm{SN} = \frac{TP}{TP+FN}, \tag{5}$$

$$\mathrm{SP} = \frac{TN}{FP+TN}, \tag{6}$$

$$\mathrm{ACC} = \frac{TP+TN}{TP+FN+FP+T}, \tag{7}$$

$$\mathrm{MCC} = \frac{TP\times TN- \quad \times FN}{\sqrt{(TP+FN)(TP+F \ )(TN+FN)(TN+FP)}}, \tag{8}$$

$$\mathrm{F1-score} = \frac{2\times SN\times SP}{SN+SP}. \tag{9}$$

In the above mathematical formulas, TP, TN, FP, and FN represent true positive, true negative, false positive, and false negative, respectively. The receiver operating characteristic (ROC) curve was drawn by linking true positive rates against false positive rates under various thresholds [54]. The area under the ROC curve (AUC) was also employed to measure the performance, which was computed by

$$AUC = \frac{\sum_{i\in positiveclass} rank_i - \frac{M\times(1+M)}{2}}{M\times N}. \tag{10}$$

Here, M represents the count of positive samples, while N corresponds to the quantity of negative samples.

3. Results and discussion

3.1. Feature combination optimization

An effective representation can significantly enhance the model's performance and generalization capabilities. Hence, selecting an appropriate representation is essential for accurate identification of 6mA modification sites. We calculated 15 common representations of DNA sequences by iLearn [55], i.e., EIIP, One-hot encoding, NCP, Kmer, PseDNC, RCKmer, DACC, DCC, DPCP, ANF, NAC, DAC, TAC, TACC, and TCC, and further examined the performance of various representations by 5-fold cross-validation. We compared four learning algorithms (LightGBM, XGBoost, AdaBoost, and SVM) and 15 representations. The predictive performances were listed in Tables S4–S6 and Table 2. It was evident that the LightGBM outperformed others in terms of ACC. Therefore, we chose LightGBM as the learning algorithm for feature selection. Table 2 displayed SN, SP, ACC, MCC and AUC. The EIIP reached the best ACC (0.798), followed by the One-hot encoding with an ACC of 0.797, by the NCP with an ACC of 0.788, by Kmer with an ACC of 0.680, and then by the PseDNC with the ACC of 0.642. The ACC of DAC, TAC, TACC, and TCC were less than 0.6. We also draw the ROC curves, as shown in Figure 3. The AUC of the EIIP, the One-hot were better, and the AUC of the Kmer, and the PseDNC was worse.

Table 2. The performance of single representations using LightGBM.

Feature Type	SN	SP	ACC	MCC	AUC
EIIP	0.856	0.741	0.798	0.601	0.869
One-hot	0.849	0.746	0.797	0.598	0.869
NCP	0.841	0.734	0.788	0.579	0.866
Kmer	0.645	0.715	0.680	0.361	0.748
PseDNC	0.625	0.660	0.642	0.285	0.710
RCKmer	0.624	0.656	0.640	0.280	0.703
DACC	0.645	0.621	0.633	0.266	0.684
DCC	0.635	0.620	0.628	0.255	0.666
DPCP	0.605	0.650	0.627	0.255	0.697
ANF	0.588	0.652	0.620	0.240	0.664
NAC	0.593	0.626	0.609	0.219	0.656
DAC	0.589	0.590	0.586	0.172	0.631
TAC	0.532	0.587	0.559	0.119	0.585
TACC	0.538	0.579	0.558	0.117	0.590
TCC	0.521	0.580	0.550	0.101	0.573

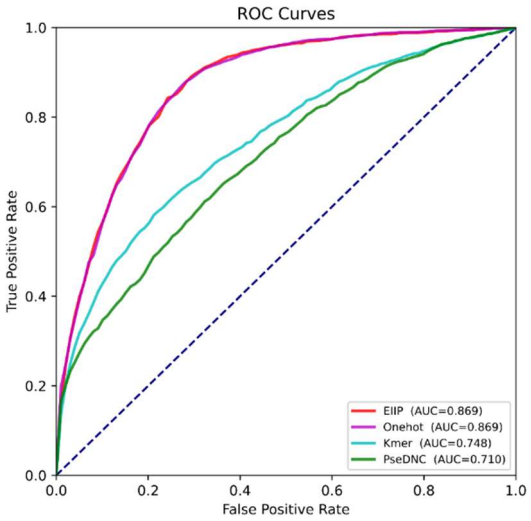

Figure 3. The ROC curves are based on LightGBM of individual features in 5-fold cross-validation.

We chose the top-performing five representations (EIIP, One-hot, NCP, Kmer, and PseDNC) for further fusion. A forward-searching strategy was employed to search for optimal representations of 6mA sequences. We started with the best-performing EIIP as the initial representation and then sequentially added a type of representation one time. The order of adding representations was One-hot, NCP, Kmer, and PseDNC. If the addition of a representation improved the overall performance, it was retained and otherwise it was excluded. As shown in Table 3, the addition of the NCP caused the ACC and the MCC to reduce. Thus, the NCP was removed. The combination of EIIP, One-hot, Kmer, and PseDNC generated the best performance with an ACC of 0.827 and an MCC of 0.656. Consequently, we combined the EIIP, the One-hot, Kmer, and PseDNC as the final representations of the DNA sequence. Figure 4 showed the ROC curves.

Table 3. Performance of combining different representations.

Feature Type	SN	SP	ACC	MCC	AUC
EIIP	0.856	0.741	0.798	0.601	0.869
EIIP +One-hot	0.860	0.744	0.802	0.608	0.871
EIIP+One-hot+NCP	0.846	0.746	0.796	0.596	0.872
EIIP+One-hot+Kmer	0.850	0.795	0.823	0.647	0.892
EIIP+One-hot+Kmer+PseDNC	0.854	0.801	0.827	0.656	0.893
EIIP+One-hot+Kmer+PseDNC+RCKmer	0.851	0.798	0.825	0.651	0.892
EIIP+One-hot+Kmer+PseDNC+DAC	0.853	0.785	0.819	0.640	0.891
EIIP+One-hot+Kmer+PseDNC+DACC	0.849	0.789	0.819	0.639	0.890
EIIP+One-hot+Kmer+PseDNC+DPCP	0.852	0.795	0.824	0.649	0.891
EIIP+One-hot+Kmer+PseDNC+ANF	0.855	0.792	0.824	0.649	0.893
EIIP+One-hot+Kmer+PseDNC+NAC	0.850	0.810	0.825	0.651	0.894
EIIP+One-hot+Kmer+PseDNC+DAC	0.854	0.792	0.823	0.648	0.891
EIIP+One-hot+Kmer+PseDNC+TACC	0.856	0.796	0.826	0.653	0.891
EIIP+One-hot+Kmer+PseDNC+TAC	0.842	0.798	0.820	0.641	0.889
EIIP+One-hot+Kmer+PseDNC+TCC	0.856	0.794	0.825	0.651	0.893

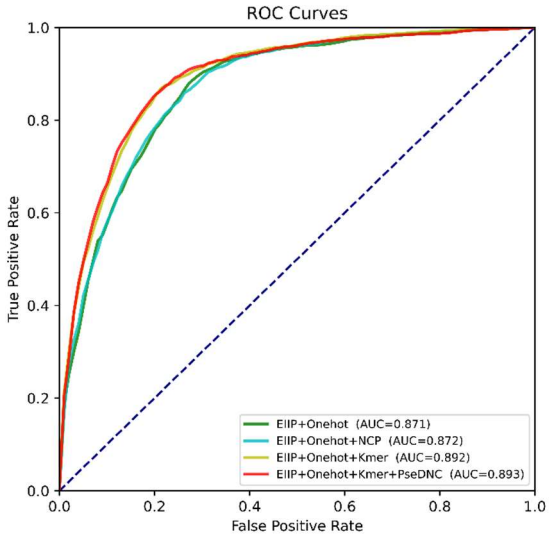

Figure 4. The ROC curves based on LightGBM of the different features in 5-fold cross-validation.

3.2. Model selection

The predictive performance relies not only on representations but also on learning algorithms. We compared six popular machine learning algorithms (LightGBM, XGBoost, GB, RF, DT and LR). Table 4 listed the evaluation metrics. The LightGBM reached the best ACC, the best MCC, and the best AUC, followed by the XGBoost, and then by the GB. The LR and the DT performed worse. Additionally, we built a deep learning-based model (called CNN) for comparison, which consisted of convolutional layers, max-pooling layers, and fully-connected layers. One-hot encoding representation was used as the input for the deep learning model. The CNN was inferior to the LightGBM, the XGBoost, and the GB in terms of ACC, MCC, and AUC.

Table 4. Performance of different machine learning algorithms.

Methods	SN	SP	ACC	MCC	AUC
LightGBM	0.854	0.801	0.827	0.656	0.893
XGBoost	0.842	0.801	0.821	0.643	0.888
GB	0.840	0.795	0.818	0.636	0.889
CNN	0.844	0.768	0.811	0.625	0.887
RF	0.864	0.754	0.809	0.621	0.875
LR	0.772	0.774	0.773	0.547	0.849
DT	0.738	0.727	0.733	0.466	0.733

Compared to a single learning algorithm, the ensemble learning has the potential to enhance the model's predictive capacity and reduce the risk of overfitting. Therefore, we further explored ensemble learning to promote predictive performance using soft voting. We applied a recursive elimination strategy to search for an optimal integration of the learning algorithm. All seven learning algorithms were first combined by soft voting for 6mA prediction. Then, starting from the worst-performing one, learning algorithms were removed from the combination in sequence. If the removal improved performance, the learning algorithm was truly removed and otherwise the learning algorithm was reserved. The process was repeated until no algorithm was removed. As shown in Table 5, the removal of the DT increased ACC by 0.001, and MCC by 0.003, indicating that the DT contributed negatively to the prediction. Subsequently, we removed the DT from the ensemble learning. Similarly, we removed the LR and the RF. The elimination of the CNN, GB, XGBoost, and LightGBM reduced ACC by 0.007, 0.002, 0.002 and 0.003, respectively. Therefore, the CNN, the GB, the XGBoost, and the LightGBM was an optimal combination.

Table 5. Performance of optimal features combined with values of different soft voting models.

Methods	SN	SP	ACC	MCC	AUC
LightGBM+XGBoost+GB+CNN+RF+LR+DT	0.859	0.788	0.823	0.648	0.891
LightGBM+XGBoost+GB+CNN +RF+LR	0.858	0.792	0.824	0.651	0.898
LightGBM+XGBoost+GB+CNN +RF	0.864	0.794	0.829	0.659	0.899
LightGBM+XGBoost+GB+CNN	0.866	0.794	0.830	0.662	0.901
LightGBM+XGBoost+GB	0.852	0.794	0.823	0.647	0.896
LightGBM+XGBoost+CNN	0.861	0.795	0.828	0.657	0.899
LightGBM+GB+CNN	0.865	0.792	0.828	0.659	0.899
XGBoost+GB+CNN	0.859	0.795	0.827	0.655	0.900

3.3. Comparison with state-of-the-art methods

So far, there are no less than 10 machine learning-based methods for 6mA prediction. We compared SoftVoting6mA with the 6mA-Bert [31] by both 5-fold cross-validation and the independent test. As shown in Table 6, the SoftVoting6mA outperformed the 6mA-Bert over the 5-fold cross-validation. The SoftVoting6mA obtained the SN of 0.866 ± 0.026, the SP of 0.793 ± 0.019, the ACC of 0.830 ± 0.016, the MCC of 0.662 ± 0.033, and the AUC of 0.901 ± 0.018, elevating SN by 0.002, SP by 0.105, ACC by 0.054, MCC by 0.011, AUC by 0.06 over the 6mA-Bert, respectively. The F1-

score was 0.836.

The comparison of performance over the independent test was shown in Table 7, the SoftVoting6mA obtained a SN of 0.884, a SP of 0.787, an ACC of 0.841, a MCC of 0.677, and an AUC of 0.901. Compared to the 6mA-Bert, the SoftVoting6mA increased SN by 0.041, SP by 0.056, ACC by 0.048, MCC by 0.097, and AUC by 0.096. The SoftVoting6mA obtained a F1-score of 0.860. We also provide the ROC curves for the 5-fold cross-validation and independent tests as shown in Figure 5A,B.

Table 6. Performance comparison by the 5-fold cross-validation.

Methods	SN	SP	ACC	MCC	AUC
6mA-Bert	0.864	0.688	0.776	0.651	0.841
SoftVoting6mA	0.866±0.026	0.793±0.019	0.830±0.016	0.662±0.033	0.901±0.018

Table 7. Performance comparison by the independent test.

Methods	SN	SP	ACC	MCC	AUC
6mA-Bert	0.843	0.731	0.793	0.580	0.805
SoftVoting6mA	0.884	0.787	0.841	0.677	0.901

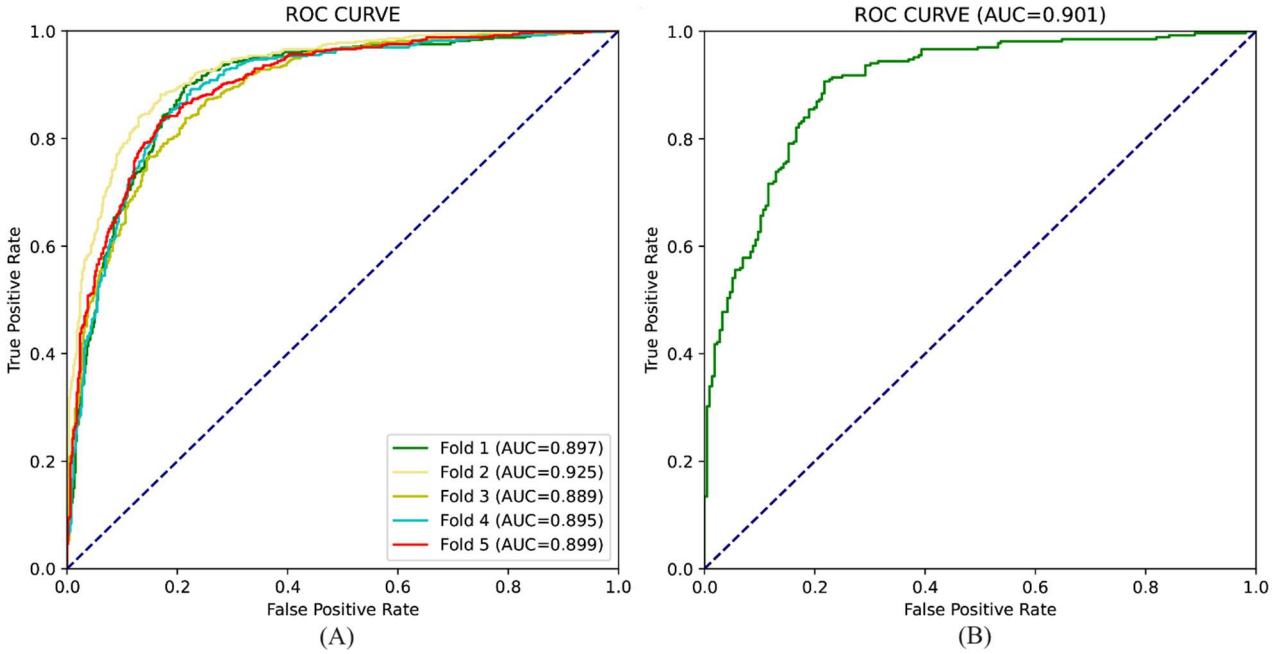

Figure 5. The ROC curves for (A) the 5-fold cross-validation and (B) the independent tests.

For a fair comparison with IDNA6mA-PseKNC [22,28], csDMA [56], and ilM-CNN [28], we used all the samples as the training dataset. The performance over the 5-fold cross-validation was shown in Table 8. Obviously, the SoftVoting6mA outperformed the iDNA6mA-PseKNC, the ilM-CNN, and the csDMA in terms of ACC. The SoftVoting6mA reached the SP of 0.804, the ACC of 0.828, the MCC of 0.656, and the AUC of 0.900, elevating the ACC by 0.063 over iDNA6mA-PseKNC [22], by 0.029 over csDMA [56], and by 0.004 over ilM-CNN [28].

To assess the stability of our method, we utilized external dataset from rice, which comprised 154,000 positive and negative samples. The comparison with Meta-i6mA and ZayyuNet [57,58] were

listed Table 9, The SoftVoting6mA reached an ACC of 0.910, outperforming two other methods.

Table 8. Performance comparison by the 5-fold cross-validation.

Methods	SN	SP	ACC	MCC	AUC
iDNA6mA-PseKNC	0.762	0.769	0.765	0.531	0.844
csDMA	0.863	0.735	0.799	0.603	0.879
ilM-CNN	0.869	0.780	0.824	0.651	0.892
SoftVoting6mA	0.851	0.804	0.828	0.656	0.900

Table 9. Performance comparison on external independent dataset.

Data	SN	SP	ACC	MCC
Meta-i6mA	0.960	0.520	0.740	0.530
ZayyuNet	1.000	0.520	0.760	0.593
SoftVoting6mA	0.900	0.918	0.910	0.809

The training and testing datasets were cross-species dataset. We conducted single species test. As listed in Table S7, the mouse reached the best performance, followed by the rice and then by cross species. This indicated that identifying 6mA site cross species was more difficult than over single species.

3.4. Web server

For users' convenience, we have implemented SoftVoting6mA as a user-friendly web server. Users can access it at http://www.biolscience.cn/SoftVoting6mA/. Users have the option to either upload a FASTA file or directly paste DNA sequences into the text box (Figure 6A). It is mandatory that the length of the submitted sequences is equal to 41. Afterward, users click the submit button and wait for the prediction results, which will be presented in a new interface (Figure 6B). Predictions are made based on a probability threshold of 0.5. If the prediction probability exceeds this threshold, it was decided to be 6mA, and otherwise it was non-6mA. In addition, users have the option to download the datasets from the website.

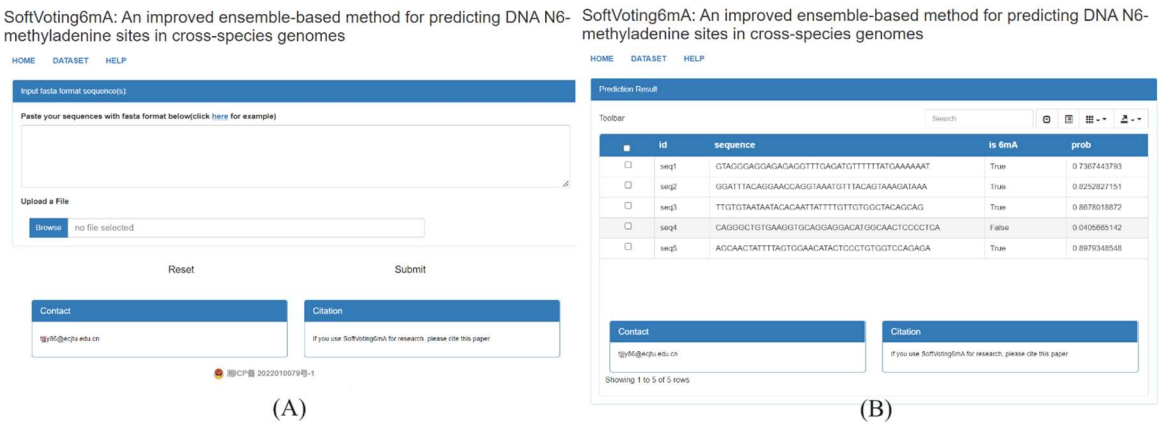

Figure 6. SoftVoting6mA web server (A) interface and (B) results interface.

4. Conclusions

DNA 6mA plays a crucial role in the cellular process. Precisely detecting 6mA sites remains challenging. To address this issue, we proposed a soft voting-based ensemble learning method for predicting DNA 6mA sites cross species, including rice and mice. By optimizing combinations of representations and learning algorithms, we demonstrated remarkable performance and showcased its universality and applicability cross species. To further validate the effectiveness of our model, we conducted validation using an external rice dataset. This validation not only enhanced the reliability of our method in rice but also provided a reliable benchmark for evaluating its performance on different datasets and species. For the convenience of researchers, we developed a user-friendly webserver, allowing free access for predicting 6mA sites. The server not only facilitates easy predictions for researchers but also extends the application across species, providing support for a broader range of biological studies.

In conclusion, we verified the wide applicability of our method through cross-species experiments, offering a fresh perspective on understanding the function and regulation of DNA 6mA in different species. However, we focused solely on two species. In future studies, we will prioritize exploring relevant sites in a broader range of species to gain a more comprehensive understanding of the distribution and function of 6mA sites in biodiversity.

Use of AI tools declaration

The authors declare that they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

The work was supported by Shaoyang University Innovation Foundation for Postgraduate (CX2022SY058).

Conflict of interest

The authors declare that there are no conflicts of interest.

References

1. V. R. Liyanage, J. S. Jarmasz, N. Murugeshan, M. R. Del Bigio, M. Rastegar, J. R. Davie, DNA Modifications: Function and Applications in Normal and Disease States, *Biology*, **3** (2014), 670–723. https://doi.org/10.3390/biology3040670
2. S. Hiraoka, T. Sumida, M. Hirai, A. Toyoda, S. Kawagucci, T. Yokokawa, et al., Diverse DNA modification in marine prokaryotic and viral communities, *Nucleic Acids Res.*, **50** (2022), 1531–1550. https://doi.org/10.1093/nar/gkab1292
3. H. Li, N. Zhang, Y. Wang, S. Xia, Y. Zhu, C. Xing, et al., DNA N6-Methyladenine Modification in Eukaryotic Genome, *Front. Genet.*, **13** (2022), 914404. https://doi.org/10.3389/fgene.2022.914404

4. C. L. Xiao, S. Zhu, M. He, D. Chen, Q. Zhang, Y. Chen, et al., N6-methyladenine DNA Modification in the Human Genome, *Mol. Cell*, **71** (2018), 306–318. e7. https://doi.org/10.1016/j.molcel.2018.06.015
5. E. L. Greer, M. A. Blanco, L. Gu, E. Sendinc, J. Liu, D. Aristizábal-Corrales, et al., DNA Methylation on N6-adenine in C. elegans, *Cell*, **161** (2015), 868–878. https://doi.org/10.1016/j.cell.2015.04.005
6. C. Ma, R. Niu, T. Huang, L. W. Shao, Y. Peng, W. Ding, et al., N6-methyldeoxyadenine is a transgenerational epigenetic signal for mitochondrial stress adaptation, *Nat. Cell Biol.*, **21** (2019), 319–327. https://doi.org/10.1038/s41556-018-0238-5
7. C. Zhou, C. Wang, H. Liu, Q. Zhou, Q. Liu, Y. Guo, et al., Identification and analysis of adenine N 6-methylation sites in the rice genome, *Nat. Plants*, **4** (2018), 554–563. https://doi.org/10.1038/s41477-018-0214-x
8. J. Liu, Y. Zhu, G. Z. Luo, X. Wang, Y. Yue, X. Wang, et al., Abundant DNA 6mA methylation during early embryogenesis of zebrafish and pig, *Nat. Commun.*, **7** (2016), 13052. https://doi.org/10.1038/ncomms13052
9. T. P. Wu, T. Wang, M. G. Seetin, Y. Lai, S. Zhu, K. Lin, et al., DNA methylation on N6-adenine in mammalian embryonic stem cells, *Nature*, **532** (2016), 329–333. https://doi.org/10.1038/nature17640
10. Z. K. O'Brown, E. L. Greer, N6-Methyladenine: A Conserved and Dynamic DNA Mark, *DNA methyltransferases-role funct.*, **945** (2016), 213–246. https://doi.org/10.1007/978-3-319-43624-1_10
11. S. Lv, X. Zhou, Y. M. Li, T. Yang, S. J. Zhang, Y. Wang, et al., N6-methyladenine-modified DNA was decreased in Alzheimer's disease patients, *World J. Clin. Cases*, **10** (2022), 448–457. https://doi.org/10.12998/wjcc.v10.i2.448
12. Q. Lin, J. W. Chen, H. Yin, M. A. Li, C. R. Zhou, T. F. Hao, et al., DNA N6-methyladenine involvement and regulation of hepatocellular carcinoma development, *Genomics*, **114** (2022), 110265. https://doi.org/10.1016/j.ygeno.2022.01.002
13. X. Sheng, J. Wang, Y. Guo, J. Zhang, J. Luo, DNA N6-Methyladenine (6mA) Modification Regulates Drug Resistance in Triple Negative Breast Cancer, *Front. Oncol.*, **10** (2021), 616098. https://doi.org/10.3389/fonc.2020.616098
14. S. Schiffers, C. Ebert, R. Rahimoff, O. Kosmatchev, J. Steinbacher, A.V. Bohne, et al., Quantitative LC–MS Provides No Evidence for m6dA or m4dC in the Genome of Mouse Embryonic Stem Cells and Tissues, *Angew. Chem. Int. Ed.*, **56** (2017), 11268–11271. https://doi.org/10.1002/anie.201700424
15. K. Han, J. Wang, Y. Wang, L. Zhang, M. Yu, F. Xie, et al., A review of methods for predicting DNA N6-methyladenine sites, *Briefings Bioinf.*, **24** (2023), bbac514. https://doi.org/10.1093/bib/bbac514
16. H. Xu, R. Hu, P. Jia, Z. J. B. Zhao, 6mA-Finder: a novel online tool for predicting DNA N6-methyladenine sites in genomes, *Bioinformatics*, **36** (2020), 3257–3259. https://doi.org/10.1093/bioinformatics/btaa113
17. H. Yu, Z. Dai, SNNRice6mA: A Deep Learning Method for Predicting DNA N6-Methyladenine Sites in Rice Genome, *Front. Genet.*, **10** (2019), 1071. https://doi.org/10.3389/fgene.2019.01071

18. M. Tahir, H. Tayara, K. T. Chong, iDNA6mA (5-step rule): Identification of DNA N6-methyladenine sites in the rice genome by intelligent computational model via Chou's 5-step rule, *Chemom. Intell. Lab. Syst.*, **189** (2019), 96–101. https://doi.org/10.1016/j.chemolab.2019.04.007
19. X. Tang, P. Zheng, X. Li, H. Wu, D. Q. Wei, Y. Liu, et al., Deep6mAPred: A CNN and Bi-LSTM-based deep learning method for predicting DNA N6-methyladenosine sites across plant species, *Methods*, **204** (2022), 142–150. https://doi.org/10.1016/j.ymeth.2022.04.011
20. M.M. Hasan, B. Manavalan, W. Shoombuatong, M. S. Khatun, H. Kurata, i6mA-Fuse: improved and robust prediction of DNA 6 mA sites in the Rosaceae genome by fusing multiple feature representation, *Plant Mol. Biol.*, **103** (2020), 225–234. https://doi.org/10.1007/s11103-020-00988-y
21. Z. Abbas, M. ur Rehman, H. Tayara, Q. Zou, K. T. Chong, XGBoost framework with feature selection for the prediction of RNA N5-methylcytosine sites, *Mol. Ther.*, 2023. https://doi.org/10.1016/j.ymthe.2023.05.016
22. P. Feng, H. Yang, H. Ding, H. Lin, W. Chen, K. C. Chou, iDNA6mA-PseKNC: Identifying DNA N6-methyladenosine sites by incorporating nucleotide physicochemical properties into PseKNC, *Genomics*, **111** (2019), 96–102. https://doi.org/10.1016/j.ygeno.2018.01.005
23. H. Lv, F. Y. Dao, Z. X. Guan, D. Zhang, J. X. Tan, Y. Zhang, et al., iDNA6mA-Rice: A Computational Tool for Detecting N6-Methyladenine Sites in Rice, *Front. Genet.*, **10** (2019), 793. https://doi.org/10.3389/fgene.2019.00793
24. Q. Huang, J. Zhang, L. Wei, F. Guo, Q. Zou, 6mA-RicePred: A Method for Identifying DNA N6-Methyladenine Sites in the Rice Genome Based on Feature Fusion, *Front. Plant Sci.*, **11** (2020), 4. https://doi.org/10.3389/fpls.2020.00004
25. Z. Teng, Z. Zhao, Y. Li, Z. Tian, M. Guo, Q. Lu, et al., i6mA-Vote: Cross-Species Identification of DNA N6-Methyladenine Sites in Plant Genomes Based on Ensemble Learning With Voting, *Front. Plant Sci.*, **13** (2022), 845835. https://doi.org/10.3389/fpls.2022.845835
26. J. Khanal, D. Y. Lim, H. Tayara, K. T. Chong, i6mA-stack: A stacking ensemble-based computational prediction of DNA N6-methyladenine (6mA) sites in the Rosaceae genome, *Genomics*, **113** (2021), 582–592. https://doi.org/10.1016/j.ygeno.2020.09.054
27. Z. Abbas, H. Tayara, K. to Chong, SpineNet-6mA: A Novel Deep Learning Tool for Predicting DNA N6-Methyladenine Sites in Genomes, *IEEE Access*, **8** (2020), 201450–201457. https://doi.org/10.1109/ACCESS.2020.3036090
28. A. Wahab, S. D. Ali, H. Tayara, K. T. Chong, iIM-CNN: Intelligent Identifier of 6mA Sites on Different Species by Using Convolution Neural Network, *IEEE Access*, **7** (2019), 178577–178583. https://doi.org/10.1109/ACCESS.2019.2958618
29. C. R. Rahman, R. Amin, S. Shatabda, M. S. I. Toaha, A convolution based computational approach towards DNA N6-methyladenine site identification and motif extraction in rice genome, *Sci. Rep.*, **11** (2021), 10357. https://doi.org/10.1038/s41598-021-89850-9
30. Z. Li, H. Jiang, L. Kong, Y. Chen, K. Lang, X. Fan, et al., Deep6mA: a deep learning framework for exploring similar patterns in DNA N6-methyladenine sites across different species, *PLoS Comput. Biol.*, **17** (2021), e1008767. https://doi.org/10.1371/journal.pcbi.1008767
31. N. Q. K. Le, Q. T. Ho, Deep transformers and convolutional neural network in identifying DNA N6-methyladenine sites in cross-species genomes, *Methods*, **204** (2022), 199–206. https://doi.org/10.1016/j.ymeth.2021.12.004

32. W. Bao, Q. Cui, B. Chen, B. Yang, Phage_UniR_LGBM: phage virion proteins classification with UniRep features and LightGBM model, *Comput. math. methods med.*, **2022** (2022). https://doi.org/10.1155/2022/9470683
33. W. Bao, Y. Gu, B. Chen, H. Yu, Golgi_DF: Golgi proteins classification with deep forest, *Front. Neurosci.*, **17** (2023), 1197824. https://doi.org/10.3389/fnins.2023.1197824
34. W. Bao, B. Yang, B. Chen, 2-hydr_ensemble: lysine 2-hydroxyisobutyrylation identification with ensemble method, *Chemom. Intell. Lab. Syst.*, **215** (2021), 104351. https://doi.org/10.1016/j.chemolab.2021.104351
35. P. Ye, Y. Luan, K. Chen, Y. Liu, C. Xiao, Z. Xie, MethSMRT: an integrative database for DNA N6-methyladenine and N4-methylcytosine generated by single-molecular real-time sequencing, *Nucleic Acids Res.*, **45** (2016), D85–D89. https://doi.org/10.1093/nar/gkw950
36. W. Chen, H. Lv, F. Nie, H. Lin, i6mA-Pred: identifying DNA N6-methyladenine sites in the rice genome, *Bioinformatics*, **35** (2019), 2796–2800. https://doi.org/10.1093/bioinformatics/btz015
37. L. Fu, B. Niu, Z. Zhu, S. Wu, W. J. B. Li, CD-HIT: accelerated for clustering the next-generation sequencing data, *Bioinformatics*, **28** (2012), 3150–3152. https://doi.org/10.1093/bioinformatics/bts565
38. G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, et al., Lightgbm: A highly efficient gradient boosting decision tree, *Adv. neural inf. process. syst.*, **30** (2017), 3149–3157. https://dl.acm.org/doi/10.5555/3294996.3295074
39. T. Chen, C. Guestrin, Xgboost: A scalable tree boosting system, in *Proceedings of the 22nd Acm Sigkdd International Conference on Knowledge Discovery and Data Mining*, Association for Computing Machinery, New York, (2016), 785–794. https://doi.org/10.1145/2939672.2939785
40. A. Natekin, A. Knoll, Gradient boosting machines, a tutorial, *Front. Neurorob.*, **7** (2013), 21. https://doi.org/10.3389/fnbot.2013.00021
41. M. Pal, Random forest classifier for remote sensing classification, *Int. J. Remote Sens.*, **26** (2005), 217–222. https://doi.org/10.1080/01431160412331269698
42. L. G. Grimm, P. R. Yarnold, *Reading and Understanding Multivariate Statistics*, American Psychological Association, Washington, 1995. https://doi.org/10.1152/advan.00006.2004
43. S. R. Safavian, D. Landgrebe, A survey of decision tree classifier methodology, *IEEE Trans. Syst. Man Cybern.*, **21** (1991), 660–674. https://doi.org/10.1109/21.97458
44. J. Inglesfield, A method of embedding, *J. Phys. C: Solid State Phys.*, **14** (1981), 3795. https://doi.org/10.1088/0022-3719/14/26/015
45. S. Albawi, T. A. Mohammed, S. Al-Zawi, Understanding of a convolutional neural network, in *2017 International Conference on Engineering and Technology (ICET)*, Akdeniz University, Antalya, (2017), 1–6. https://doi.org/10.1109/ICEngTechnol.2017.8308186
46. D. Lalović, V. Veljković, The global average DNA base composition of coding regions may be determined by the electron-ion interaction potential, *Biosystems*, **23** (1990), 311–316. https://doi.org/10.1016/0303-2647(90)90013-Q
47. W. He, C. Jia, EnhancerPred2. 0: predicting enhancers and their strength based on position-specific trinucleotide propensity and electron–ion interaction potential feature selection, *Mol. Biosyst.*, **13** (2017), 767–774. https://doi.org/10.1039/C7MB00054E
48. W. He, C. Jia, Q. Zou, 4mCPred: machine learning methods for DNA N4-methylcytosine sites prediction, *Bioinformatics*, **35** (2019), 593–601. https://doi.org/10.1093/bioinformatics/bty668

49. P. Rodríguez, M.A. Bautista, J. Gonzalez, S. Escalera, Beyond one-hot encoding: lower dimensional target embedding, *Image Vision Comput.*, **75** (2018), 21–31. https://doi.org/10.1016/j.imavis.2018.04.004
50. K. C. Chou, Prediction of protein cellular attributes using pseudo-amino acid composition, *Proteins Struct. Funct. Bioinf.*, **43** (2001), 246–255. https://doi.org/10.1002/prot.1035
51. W. Chen, P. M. Feng, H. Lin, K. C. Chou, iRSpot-PseDNC: Identify recombination spots with pseudo dinucleotide composition, *Nucleic Acids Res.*, **41** (2013), e68. https://doi.org/10.1093/nar/gks1450
52. W. Chen, P. M. Feng, H. Lin, K. C. Chou, iSS-PseDNC: Identifying splicing sites using pseudo dinucleotide composition, *Biomed Res. Int.*, **2014** (2014). https://doi.org/10.1155/2014/623149
53. W. Chen, H. Ding, X. Zhou, H. Lin, K. C. Chou, iRNA(m6A)-PseDNC: identifying N6-methyladenosine sites using pseudo dinucleotide composition, *Anal. Biochem.*, **561** (2018), 59–65. https://doi.org/10.1016/j.ab.2018.09.002
54. Z. Cui, S. G. Wang, Y. He, Z. H. Chen, Q. H. Zhang, DeepTPpred: A deep learning approach with matrix factorization for predicting therapeutic peptides by integrating length information, *IEEE J. Biomed. Health. Inf.*, **27** (2023), 4611–4622. https://doi.org/10.1109/jbhi.2023.3290014
55. Z. Chen, P. Zhao, C. Li, F. Li, D. Xiang, Y. Z. Chen, et al., iLearnPlus: a comprehensive and automated machine-learning platform for nucleic acid and protein sequence analysis, prediction and visualization, *Nucleic Acids Res.*, **49** (2021), e60. https://doi.org/10.1093/nar/gkab122
56. Z. Liu, W. Dong, W. Jiang, Z. He, csDMA: an improved bioinformatics tool for identifying DNA 6 mA modifications via Chou's 5-step rule, *Sci. Rep.*, **9** (2019), 13109. https://doi.org/10.1038/s41598-019-49430-4
57. M. M. Hasan, S. Basith, M. S. Khatun, G. Lee, B. Manavalan, H. Kurata, Meta-i6mA: an interspecies predictor for identifying DNA N 6-methyladenine sites of plant genomes by exploiting informative features in an integrative machine-learning framework, *Briefings Bioinf.*, **22** (2021), bbaa202. https://doi.org/10.1093/bib/bbaa202
58. Z. Abbas, H. Tayara, K. T. Chong, ZayyuNet–A unified deep learning model for the identification of epigenetic modifications using raw genomic sequences, *IEEE/ACM Trans. Comput. Biol. Bioinf.*, **19** (2021), 2533–2544. https://doi.org/10.1109/tcbb.2021.3083789

AIMS Press