Genomics 113 (2021) 582–592

Contents lists available at ScienceDirect

Genomics

journal homepage: www.elsevier.com/locate/ygeno

i6mA-stack: A stacking ensemble-based computational prediction of DNA N6-methyladenine (6mA) sites in the Rosaceae genome

Jhabindra Khanal[a,1], Dae Young Lim[a,b,1], Hilal Tayara[c,*], Kil To Chong[a,b,*]

[a] Department of Electronics and Information Engineering, Jeonbuk National University, Jeonju 54896, South Korea
[b] Advanced Electronics and Information Research Center, Jeonbuk National University, Jeonju 54896, South Korea
[c] School of International Engineering and Science, Jeonbuk National University, Jeonju 54896, South Korea

ARTICLE INFO

Keywords:
Sequence analysis
DNA N6-methyladenine
Machine learning
RFECV
Stacking

ABSTRACT

DNA N6-methyladenine (6 mA) is an epigenetic modification that plays a vital role in a variety of cellular processes in both eukaryotes and prokaryotes. Accurate information of 6 mA sites in the *Rosaceae* genome may assist in understanding genomic 6 mA distributions and various biological functions such as epigenetic inheritance. Various studies have shown the possibility of identifying 6 mA sites through experiments, but the procedures are time-consuming and costly. To overcome the drawbacks of experimental methods, we propose an accurate computational paradigm based on a machine learning (ML) technique to identify 6 mA sites in *Rosa chinensis (R.chinensis)* and *Fragaria vesca (F.vesca)*. To improve the performance of the proposed model and to avoid overfitting, a recursive feature elimination with cross-validation (RFECV) strategy is used to extract the optimal number of features (ONF) subset from five different DNA sequence encoding schemes, i.e., Binary Encoding (BE), Ring-Function-Hydrogen-Chemical Properties (RFHC), Electron-Ion-Interaction Pseudo Potentials of Nucleotides (EIIP), Dinucleotide Physicochemical Properties (DPCP), and Trinucleotide Physicochemical Properties (TPCP). Subsequently, we use the ONF subset to train a double layers of ML-based stacking model to create a bioinformatics tool named 'i6mA-stack'. This tool outperforms its peer tool in general and is currently available at http://nsclbio.jbnu.ac.kr/tools/i6mA-stack/

1. Introduction

Epigenetics refers to heritable phenotype changes that do not directly alter the integrity of a genetic code. N6-methyladenine or 6 mA (the sixth position of the purine ring in adenines) is one of the most important epigenetic modifications in a DNA molecule. It plays a vital role in the regulation of many biological functions, including transgenerational inheritance; gene expression; nucleosome positioning; cell cycle regulation; DNA repair and replication; and the restriction-modification (R-M) system [1,2]. However, these biological functions which result from 6 mA modification in higher eukaryotes, remain largely unclear because 6 mA distribution pattern are rather species-specific, which leads to varying the functional roles. A series of studies have provided some insight to the distributions and biological functions of the important adenine methylation in different species [3–14]. For example, a recent study conducted by Zhao-Yu Liu et al. [15] observed 6 mA distributions in *F.vesca* and *R.chinensis* of the *Rosaceae* family. In addition, a few studies have demonstrated that the 6 mA level in prokaryotes is higher (0.002–2.7%) [16] than that of eukaryotes (0.000006–0.8%) [10,17]. The reasons behind the low presence of adenine methylation (6 mA) sites in eukaryotes still remain unclear, but it may be related to the difference in the size, presence, and length of palindromic sequences; genome complexity: and complicated epigenetic processes.

A number of experimental methods have been proposed to identify 6 mA sites, such as high-performance liquid chromatography (HPLC) separation coupled with tandem mass spectrometry (MS/MS) [18] and a single –molecule real-time (SMRT) sequencing-based technique [19]. However, a couple of problems are found in such experimental methods including antibody detection. Antibody detection is not quantitative and the high chance of recognizing other adenine base modifications, and the outputs of antibody detection and MS/MS could be damaged by microscopic organisms such as bacteria. The SMRT sequencing-based technique is unable to distinguish between 6 mA and another adenine base modification such as N1-Adenine (1 mA) [20]. Furthermore, due to the methods being labor-intensive, time-consuming, and expensive, in

* Corresponding authors at: Department of Electronics and Information Engineering, Jeonbuk National University, Jeonju 54896, South Korea.
E-mail addresses: hilaltayara@jbnu.ac.kr (H. Tayara), kitchong@jbnu.ac.kr (K.T. Chong).
[1] Jhabindra Khanal and Dae Young Lim contributed equally.

https://doi.org/10.1016/j.ygeno.2020.09.054
Received 8 August 2020; Received in revised form 9 September 2020; Accepted 23 September 2020
Available online 01 October 2020
0888-7543/

silico prediction of adenine base modification sites in a genome has appeared to be an alternative approach. Recently, machine learning algorithms have emerged as a promising tool that could resolve these problems and help experimental scientists to identify 6 mA sites.

In recent years, species-specific ML-based research and deep learning-based research has been conducted for the prediction of 6 mA sites [21–26]. However, the existing methods are not sufficient to predict 6 mA sites in the *Rosaceae* genome due to the species-specific nature of the 6 mA distributions, or these methods have not been designed for the *Rosaceae* genome. In addition, to date, there is only a single research article that identified 6 mA sites in the *Rosaceae* genome. It followed by an ML-based method for which fusing multiple feature representations was a key concept [27]; however, the performance of the classifier still remains to improved. Accordingly, we aim to build an effective bioinformatics tool to identify 6 mA sites in the *Rosaceae* genome.

In this paper, we present a novel online DNA 6 mA site prediction tool. We explore different feature encoding schemes and ML techniques, to further improve the prediction accuracy for two species. First, we fuse five encoding schemes into a single set of 1570 feature vectors. Next, the RFECV algorithm is used to extract an ONF (210 features) subset from the 1570-dimensional set, independently for both species. Then, the ONF set is encoded to the original DNA sequences to train four ML-based base-classifiers, and the predicted bonding and non-bonding probabilities of selected base-classifiers are used as inputs to a meta-classifier to get the final model. On independent test data, the developed method offered a significant improvement in prediction accuracy when compared to a previous method. The general workflow for constructing the i6mA-stack is shown in Fig. 1.

2. Materials and methods

2.1. Dataset construction and preprocessing

The positive sequences (6 mA site containing sequences) for *F.vesca* and *R.chinensis* genomes were obtained from the MDR database [15], http://mdr.xieslab.org/. According to W. Chen et al. [28] and Feng et al. [24], a modification QV (modQV) score of 30 or more is the best high-quality for the threshold for labelling the position of a related adenine nucleotide as modified. Thus, in this paper, modQV of ≥ 30 was used to construct the positive dataset. All the positive and negative sequences contained 41-nt with adenine 'A' nucleobase at the center. According to previous researches the best predictive results were achieved with the length of 41-nt sequences [26,27]. The CD-HIT software [29] was used with a cutoff threshold of 70.00% to minimize bias, to remove redundancy in curated sequences, and to delete high sequence similarity. After applying these two processes to construct a reliable dataset, we obtained non-redundant positive sequences of 2313

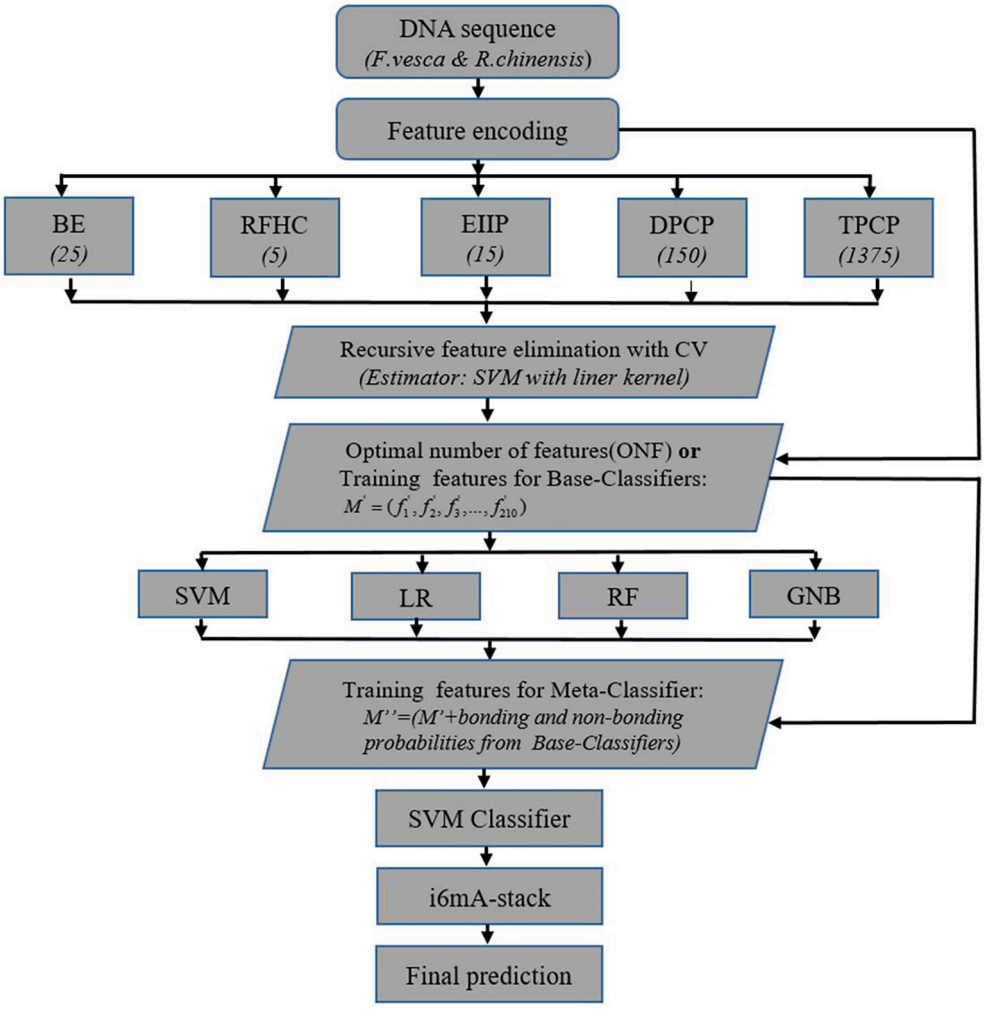


Fig. 1. The framework of the proposed i6mA-stack. For a given DNA sequence of the *Rosaceae* genome, a 41-nt scan window (*w*) is used.

Table 1
Statistical summary of the datasets for the two species.

Genomes	Positive/Negative	Training datasets	Independent datasets
F.vesca	Positive	1966	347
	Negative	1966	347, 1735, 5205
R.chinensis	Positive	813	143
	Negative	813	143, 715, 2145

and 956 6 mA sites for *F.vesca* and *R.chinensis*, respectively. From the two datasets approximately 15.00% of the sequences (347 and 143 6mAs for *F.vesca* and *R.chinensis)* were randomly selected as the independent datasets, and the remaining 85.00% of the sequences (1966 and 813 6mAs for *F.vesca* and *R.chinensis*) were used as the training datasets.

The negative sequences (non-6 mA site containing sequences) for *F.vesca* and *R.chinensis* were obtained from the same genome file where the 6 mA sites (adenine in the center) was not identified by the SMRT. To do this, Pybedtools (a flexible python library) [30] was used to separate the non-6 mA sites (negative sequences) and the known 6 mA sites (positive sequences) in the whole genome of both species. Through this process, we obtained a large number of negative sequences with 'A' in the center. To construct the non-6 mA sequences independently for both species, the negative sequences were randomly selected from the *F.vesca* and *R.chinensis* genomes. To delete the redundant sequences from the negative samples, the CD-HIT software [29] was used with a 70.00% cutoff threshold. The positive and negative datasets are summarized in Table 1. The complete training and independent datasets can be downloaded from http://nsclbio.jbnu.ac.kr/tools/i6mA-stack/

2.2. Feature vector construction

In this study, we used a strategy based on five encoding schemes to encode the sample sequences. These were: Binary Encoding (BE) or Onehot Encoding [31]; Ring-Function-Hydrogen-Chemical Properties (RFHC) [32]; Electron-Ion-Interaction Pseudo Potentials of Nucleotides (EIIP) [33,34]; Dinucleotide Physicochemical Properties (DPCP) [34,35]; and Trinucleotide Physicochemical Properties (TPCP) [36]. A detailed explanation of these feature encoding techniques is presented in the Supplementary Material (Section A).

2.2.1. Recursive feature elimination with cross-validation (RFECV)

Recursive feature elimination (RFE) is a well-known feature selection process that fits a model and removes the weakest features until a specified number of features is reached; however, it is often not known in advance how many features should be selected for the optimal result. To find the optimal number of features (ONF), cross-validation (CV) was used with the RFE method to score different feature subsets and select the best subset of features. In each step, 20% of less important features were eliminated by using five-fold CV method. To implement this section, the RFECV package in *Python* environment was used, it is available at the following link [37] (https://scikit-learn.org). For each 41-nts long sequence, we encoded 205, 41, 123, 240, and 429-dimensional vectors for BE, RFHC, EIIP, DPCP, and TPCP, respectively. For a 41-nts length of sequence, the five types of feature scheme were integrated into a new feature group, which resulted in a total of 1038 features for each sequence. Next, the RFECV method was applied to select the ONF from the integrated (1038) feature set for each sequence. The least important features were eliminated and the final optimized subset contained the 210 most important features. To eliminate the less important features using RFECV, Support Vector Machine (SVM) with linear kernel was used. For the datasets of both species, we have found that SVM was the best performer among the ML-based classifiers adopted in this study. After having fitted SVM, it is possible to access the classifier coefficients. The ONF group could, therefore, be generated by comparing the size of these coefficients to each other. Eliminating the least valuable features in ML plays an important role for, avoiding overfitting, speed up training, reducing computing time and complexity. As a result, this approach ultimately leads to better classification results.

2.3. Base-classifiers and stacking

In order to select base-classifiers to use in the first layer of stacking and meta-classifier to use in the second layer of stacking, we carefully evaluated each of the five types of sequence encoding schemes and the ONF using five conventional ML algorithms: Logistic Regression (LR) [38], Support Vector Machine (SVM) [39], Random Forest (RF) [40], Gaussian Naive Bayes (GNB) [41], and Bagging (BAG) [42]. We observed the ONF to be informative for predicting 6 mA sites, while SVM represented the most powerful classifier. A brief introduction of the five classifiers and their configuration details are given in the Supplementary Material (Section B). All of the above mentioned ML-based classifiers were built and tuned using the Scikit-learn *Python* library for ML [37].

Stacking can combine heterogeneous base classifiers and learns in parallel by training a meta-classifier to generate predictions by drawing from the base-classifier's predictions [43,44]. The approach involvs training the base-classifiers (first-level learners) using the original training dataset, and then generating a new dataset for training the meta-classifier (second-level learner). For the meta-classifier, the outputs of the base-classifiers are used as input features while the original labels are used as labels for the new training data, but this can lead to overfitting. To avoid overfitting produced by the stacking ensemble approach, the method of CV has been used, in which the dataset is split into k folds, and in k successive rounds, $k-1$ fold folds are used to fit the base-classifiers: in each round, the base-classifiers are applied to the single subset that is not used for model.

According to the previous research, to achieve better prediction information for the solution space, it is strictly recommended to use base-classifiers with mutually distinct underlying operating principles [45]. To appropriately select of base-classifiers for a particular situation is a major challenge in stacking ensemble learning. This is because if base classifiers are highly correlated and generate similar outputs, their combination will not create an accurate prediction. In contrast, when base classifiers are independent and make diverse predictions, it can easily guess that the independent errors have better chances to be canceled out. On the other hand, Breiman has claimed that even though the stacking approach works well in practice [46] and has been successfully applied in several machine learning and bioinformatics tasks [47–52], it has not yet been shown to formally satisfy the optimality principle. The detailed flow of the stacking-ensemble algorithm is presented in the three steps in Algorithm 1.

To select the combination of base-classifiers (CB) for the stacked

model according to the different working principles of the base-classifiers, we evaluated five different combinations. These five combinations are as follows:

i. *CB*1: consists of SVM, LR, and RF.
ii. *CB*2: consists of SVM, LR, and BAG.
iii. *CB*3: consists of SVM, LR, and GNB.
iv. *CB*4: consists of SVM, LR, GNB, and RF.
v. *CB*5: consists of SVM, LR, GNB, and BAG.

For this series of combinations of the base-classifiers, we endeavored to combine the base classifiers with differing underlying principles. For example, for CB4 and CB5, the first three base-classifiers SVM, LR, GNB, have different working principles and they are combined with the tree-based algorithms RF and BAG, respectively. Although there are more possible combinations of the base-classifiers, we have presented here the five best performing sets. In each set, all of the combinations of the base-classifiers learn in parallel and combined

Algorithm 1 Staking algorithm

Input: Training dataset $D = \{(x_1, y_1), (x_2, y_2), ..., (x_m, y_m)\}$;
Output: $H(x) = h^{'}(h_1(x), ..., h_T(x))$
Step 1: Learn base-level classifiers, i.e., SVM, LR, RF, GNB
for $t = 1$ to T **do**
    learn h_t based on D
end for
Step 2: Construct new datasets of predictions
for $i = 1$ to m **do**
    $D_h = \{x_i^{'}, y_i\}$, Where $x_i^{'} = \{h_1(x_i), ..., h_T(x_i)\}$
end for
Step 3: Learn a meta-classifier, i.e., SVM
learn $H(x)$ based on D_h
return $H(x)$

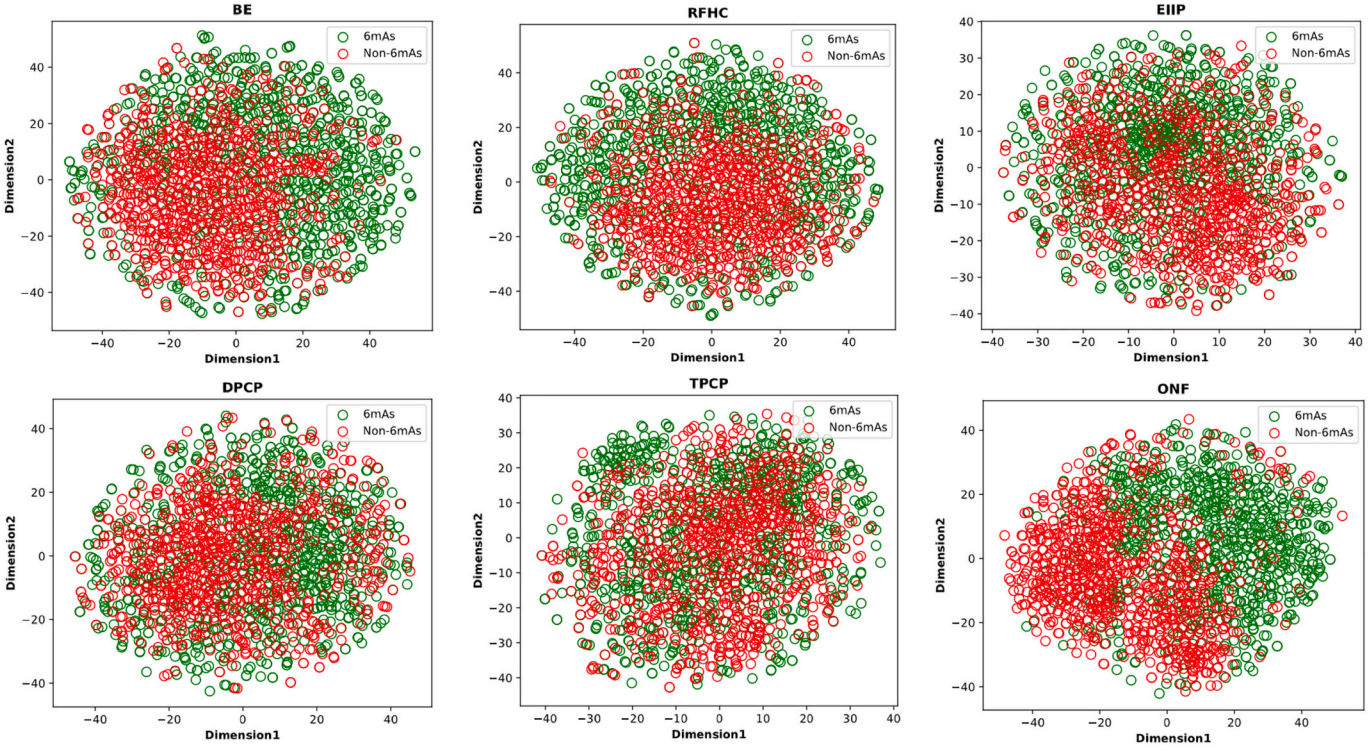


Fig. 2. t-SNE visualizations of the *F.vesca* dataset using the five encoding schemes, i.e., BE, RFHC, EIIP, DPCP, TPCP, and the ONF obtained through the RFECV method. As seen, the t-SNE visualizations based on ONF outperform the other five original encoding schemes in the case for distinguishing between 6mAs and non-6 mAs.

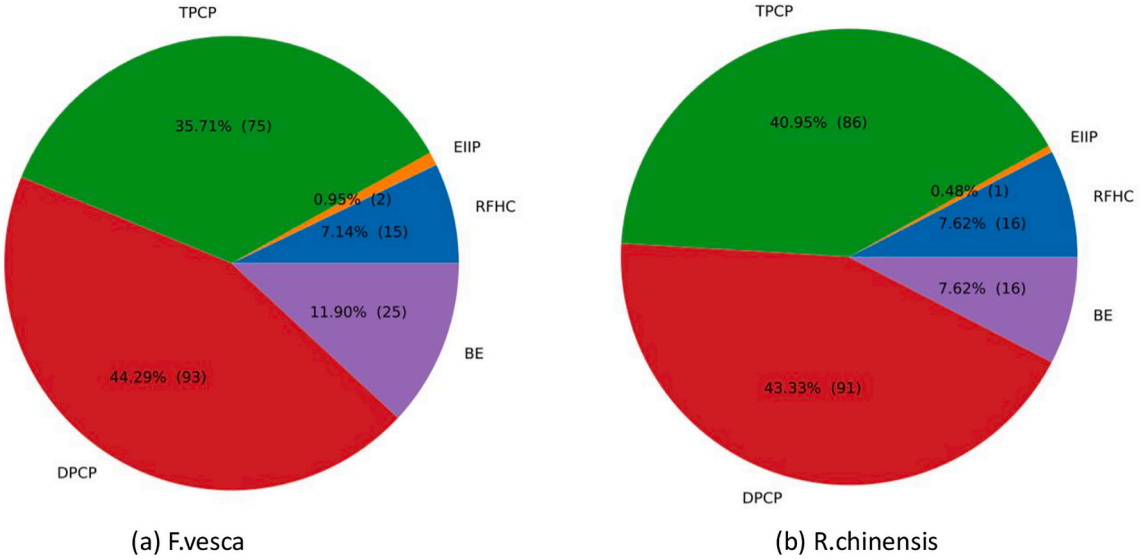


Fig. 3. The pie charts of (a) *F.vesca* and (b) *R.chinensis* show the number of obtained ONF through the RFECV method from five sequence encoding schemes. As seen, the number of selected ONF (210) from BE, RFHC, EIIP, TPCP, and DPCP are 25, 15, 2, 75, and 93, respectively for the *F.vesca* while for the *R.chinensis* those number of ONF (210) are 16, 16, 1, 86, and 91 from BE, RFHC, EIIP, TPCP, and DPCP, respectively.

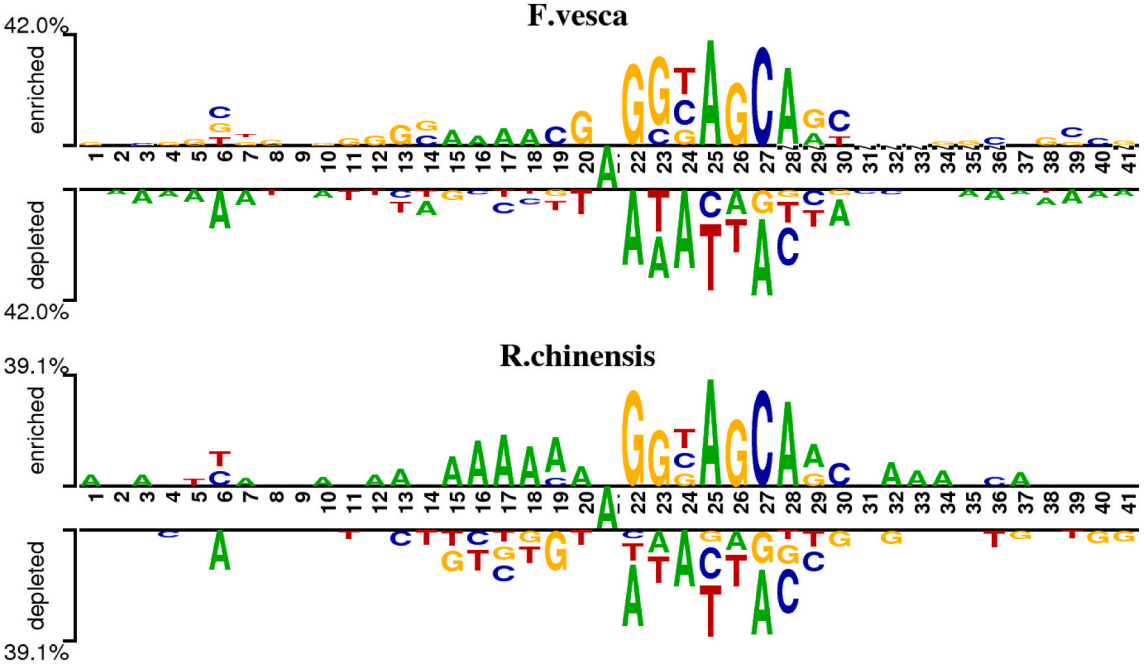


Fig. 4. Nucleotide preferences of 6 mA and non-6 mA-containing sequences of *F.vesca* and *R.chinensis*.

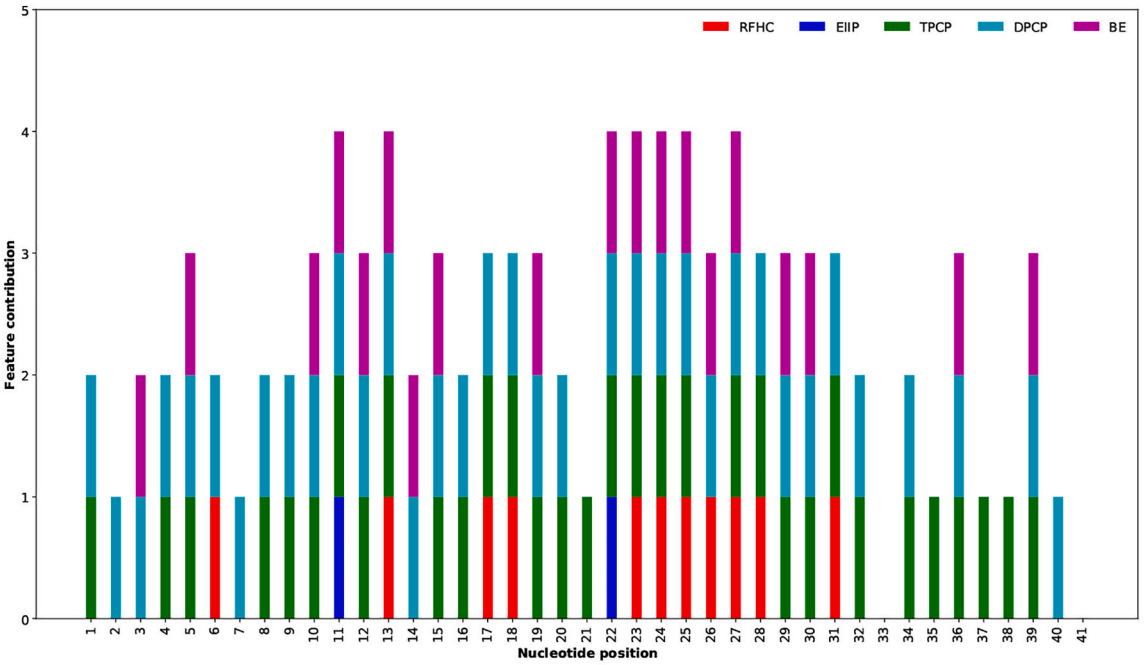
them by a meta-classifier (second-level classifier), it is fair to use the SVM as meta-level classifier according to the representation of the most powerful classifier for the datasets used in this study.

2.4. Performance evaluation

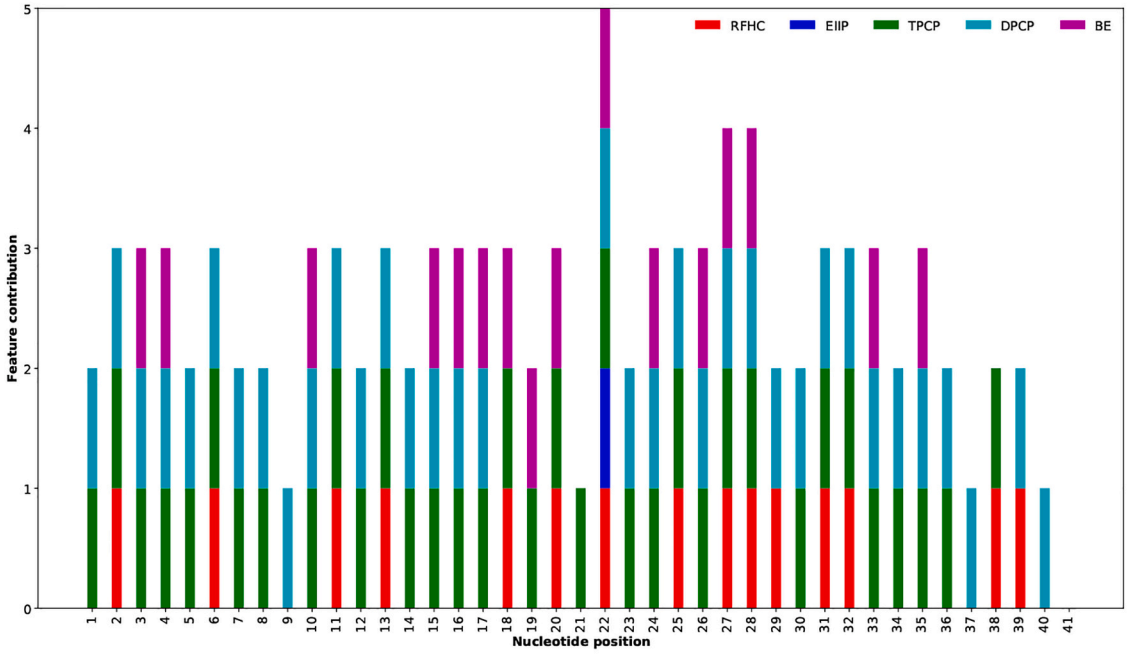
The performance of the i6mA-stack was evaluated by a five-fold CV approach. The predicted performance was evaluated by multiple measurements as described in the Supplementary Material (Section C): i.e., sensitivity (Sn), specificity (Sp), accuracy (ACC), and Mathews correlation coefficient (MCC). These parameters were widely used by other researchers such as: Refs. [53–56]. In addition, we included the area under the curve (AUC) and receiver operating characteristics (ROC) in the performance evaluation. ROC is a probability curve, and AUC stands for the area under the ROC curve, also expressed as AUROC, and it represents a degree of separability. Thus a higher AUC value indicates a more accurate model. For imbalanced class datasets, the precision-recall curve is more suitable than an ROC curve [57]; for this reason, we used the precision (y-axis)-recall (x-axis) for testing our imbalanced independent datasets.

3. Results and discussion

In this section, we first present the performance of all five ML-based algorithms through the five types of feature encoding schemes and the ONF. We demonstrate how the ONF helped to increase the performance of these classifiers. Then, we report the performance of the five stacking frameworks and select the best stacking model as the i6mA-stack bioinformatics tool.



(a)



(b)

Fig. 5. The contribution of the ONF of each encoding scheme at every position in the input sequence for (a) *F.vesca* (b) *R.chinensis*. As seen, the distal nucleotides play an important role in identifying the 6mAs.

3.1. Selection of ONF with RFECV

One of the reasons for constructed the i6mA-stack as a good performer in identifying 6 mA sites in both genomes is due to the ONF group obtained through the different feature learning schemes. To illustrate this, we computed t-distributed stochastic neighbor embedding (t-SNE) [58] in the *Scikit-Python* (https://scikit-learn.org) library and visualized the results in scatter plots for two-dimensional space. The t-SNE is an unsupervised non-linear technique mainly applied for visualizing and exploration high-dimensional data. In simpler words, it is a data arrangement technique in which a cost function is used to optimize a similarity measure between pairs of instances in the low and high-dimensional space. In this study, all the parameters were left at default (n_components = 2, perplexity = 30, learning_rate = 200 and

Table 2
Performance of the SVM classifier with the five encoding schemes and the ONF on the training datasets based on the five-fold CV.

Species	Methods	Sn	Sp	ACC	MCC	AUC
F.vesca	RFHC	0.9100	0.9015	0.9058	0.8116	0.9610
	EIIP	0.7275	0.7358	0.7315	0.4631	0.8040
	TPCP	0.9025	0.8264	0.8651	0.7317	0.9360
	DPCP	0.9250	0.9067	0.9160	0.8320	0.9640
	BE	0.9150	0.8938	0.9045	0.8091	0.9620
	ONF	**0.9325**	**0.9430**	**0.9376**	**0.8753**	**0.9740**
R.chinensis	RFHC	0.8760	0.8645	0.8707	0.7409	0.9330
	EIIP	0.6882	0.7225	0.7046	0.4104	0.7970
	TPCP	0.8058	0.7548	0.7815	0.5617	0.8550
	DPCP	0.8294	0.8000	0.8153	0.6299	0.9060
	BE	0.8760	0.8650	0.8707	0.7400	0.9330
	ONF	**0.9000**	**0.9161**	**0.9079**	**0.8154**	**0.9630**

Note: The best performance value for each metric across different encoding schemes is highlighted in bold.

number of iterations [n_iter] = 1000) for each encoding schemes. Fig. 2 shows the distributions of the 6mAs and non-6 mAs sequences of six feature descriptors in the *F.vesca* dataset. The green and red-colored circles denote the 6mAs and non-6 mAs, respectively. The figure shows that the 6 mA and non-6 mA sequences encoded with the generated ONF (210 features) through the RFECV are easily separable, though a few sequences overlap. We also computed t-SNE distributions for the *R.chinensis* dataset, and this is presented in the Supplementary Material (Fig. S1). It also shows the 6 mA and non-6 mA samples are easier to distinguish when the ONF feature set is encoded in comparison to the five original encoding schemes. In addition, the number of selected ONF through the RFECV from each encoding schemes is presented in Fig. 3.

3.2. Analysis of nucleotide preference of F.vesca and R.chinensis

To analyze the statistically significant differences in position-specific of 6 mA and non-6 mA-containing sequences for *F.vesca* and *R.chinensis*, the two-sample logo software [59] (at a level of p0.05) was used. The logo software has been used in a series of publications to examine the position-specific preference of nucleotide composition in different species [60–62]. We examined that the 'A' nucleotide is located at the 21^{st} position of the DNA sequences having the length of 41-nt as depicted in Fig. 4. In case of *F.vesca*, the 'A' base is enriched at positions 15–18, 25, and 28, while the 'G' base is more enriched at positions 1, 3, 5–14, 20, 22–24, 26, 29–34, 35, 38, 39, and 41. The 'A' base was significantly depleted at positions 1–7, 10, 14, 22–24, 26, 27, 30, and 35–41, while the 'T' base was depleted at positions 8, 11–14, 17–20, 23, 25, 26, 28, 29 and 39. In case of *R.chinensis*, the 'A' base was over-represented at positions 1, 3, 7, 10, 12, 13, 15–20, 25, 28, 29, 32–34, and 37, while the 'C' base was more over-represented at positions 6, 19, 24, 27, 30, and 36 than other nucleotides. The 'G' base was significantly under-represented at positions 15, 17–19, 25, 27, 28, 30, 32, 37, 40, and 41. Over-representation and under-representation of nucleotides at a specific position might have important information for identifying 6 mA sites.

In addition, we show the contribution of the ONF of each encoding scheme at every position in the input sequence as shown in Fig. 5a for the *F.vesca* and Fig. 5b for the *R.chinensis*. In case of both species, we can see that almost all nucleotides in the input sequence contribute in generating the final feature vectore of length 210. These results show that the distal nucleotides contribute in discriminating the 6 mA from non-6 mA.

3.3. Performance of base-classifiers and meta-classifier

Based on the five encoding schemes and the ONF, we first analyze the prediction performance of five widely used ML-based algorithms individually, i.e., SVM, RF, LR, GNB, and BAG. The performance of the five encoding schemes with the five ML-based algorithms is presented in the Supplementary Material (Section E). The results show that the ONF is more accurate in prediction tasks than each of the other five encoding schemes when classified by the ML-based algorithms, and SVM outperforms the other classifiers. Table 2 shows the performance of the five feature encoding schemes and the ONF classified by SVM, and it can clearly be seen that by the ONF features are notably more helpful for achieving higher accuracy than each of the five encoding schemes for both species when classified by SVM. As shown in Fig. 6, the ROC curves indicated that the SVM through the ONF consistently performed better than the other encoding schemes for both species.

Table 3 indicates that the optimized SVM with linear-kernel gives a higher CV accuracy compared to other ML-based methods i.e., RF, LR, GNB, and BAG. In *F.vesca*, the SVM with linear kernel achieved ACC, MCC and AUC values of 0.9376, 0.8753 and 0.9740, respectively, and in *R.chinensis*, the SVM achieved ACC, MCC and AUC values of 0.9161, 0.8154 and 0.9630, respectively. It is worth noting that, LR provides the second-highest performance. In terms of learning principle, SVM and LR are different from each other but they had the best performance. Accordingly, it was justifiable to select the SVM and LR as key base-classifiers in addition to selecting SVM as the meta-classifier. In addition, the algorithms of RF, GNB, and BAG show satisfactory performance with ACC values higher than 83.00%. To identify the optimal combination of base-classifiers, we constructed the five CBs, i.e., CB1, CB2, CB3, CB4, and CB5, as discussed in Section 2.3. Table 4 shows the performance of each combinations for the two species. From Table 4, we note that CB1, CB2, CB3, CB4, and CB5 stacked model sets provide similar performance, with ACC value larger than 0.9300. However, CB4, which includes SVM, LR, GNB, and RF as base-classifiers and SVM as a meta-classifier, provides the highest performance. On the benchmark dataset for the 5-fold CV, CB4 has an Sn of 0.9694, Sp of 0.9321, ACC of 0.9510, MCC of 0.9025, and AUC of 0.9880 for *F.vesca*. For the five-fold CV on the benchmark dataset of *R.chinensis* CB4 provides an Sn of 0.9408, Sp of 0.9394, ACC of 0.9401, MCC of 0.8800, and AUC of 0.9766. As a result, CB4 (the i6mA-stack) was selected as our final 6 mA predictor for both species. Furthermore, the results obtained by the stacked model are remarkably superior to results obtained by the individual classifiers when tested on the training dataset for both species: the comparative AUC curves for the five ML-based classifiers and the stacking model CB4 (the i6mA-stack) are presented in Fig. 7.

3.4. Performance comparison with existing method on the independent dataset

To test whether our i6mA-stack could identify 6 mA sites on an unseen dataset, we ran the model on the independent test datasets. The independent datasets were constructed with different ratios of positive and negative sequences. For *F.vesca* these were: 1:1 [347 positive and 347 negative samples], 1:5 [347 positive and 1735 negative samples], and 1:15 [347 positive and 5202 negative samples]. Datasets with the

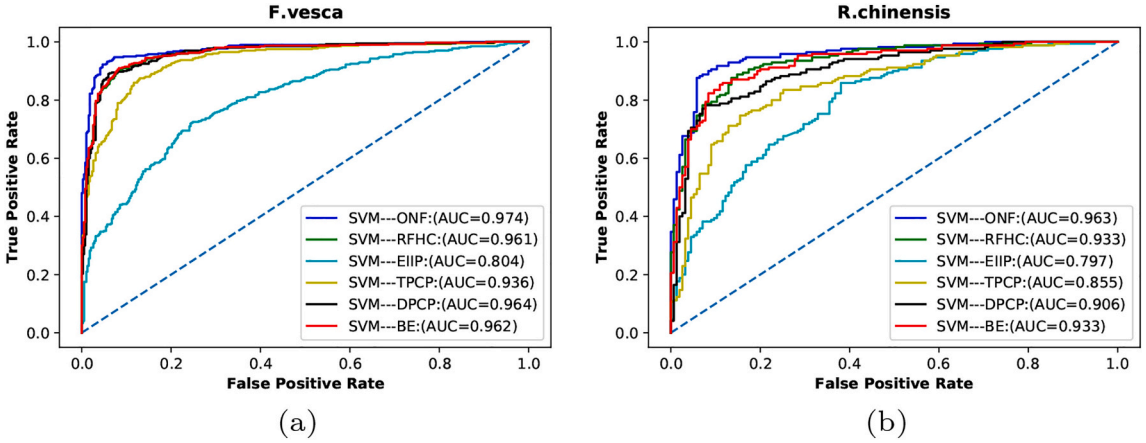


Fig. 6. Roc curves based on the five-fold CV test to assess the predictive performance of the six encoding scheme classified by SVM for the two species. When encoding the original benchmark data with the ONF, AUC values identifying 6 mA in *F.vesca* (a) and *R.chinensis* (b) are 0.974 and 0.963, respectively.

Table 3
Performance of the base-classifiers with the ONF encoding scheme on the training dataset based on the five-fold CV.

Species	Classifiers	Sn	Sp	ACC	MCC	AUC
F.vesca	LR	0.9300	0.9430	0.9363	0.8728	0.9725
	BAG	0.9300	0.9378	0.9338	0.8677	0.9720
	SVM	**0.9325**	**0.9430**	**0.9376**	**0.8753**	**0.9740**
	RF	0.9075	0.9248	0.9160	0.8322	0.9650
	GNB	0.8950	0.8549	0.8753	0.7508	0.9410
R.chinensis	LR	**0.9176**	0.8967	0.9076	0.8149	0.9550
	BAG	0.8941	0.8967	0.8953	0.7905	0.9530
	SVM	0.9000	**0.9161**	**0.9079**	**0.8154**	**0.9630**
	RF	0.8705	0.9161	0.8923	0.7859	0.9510
	GNB	0.8823	0.8645	0.8738	0.7470	0.9340

Note: The best performance value for each metric across different classifiers is highlighted in bold.

Table 4
Performance of stacked models with different set of base-classifiers based on the five-fold CV.

Species	Method	Sn	Sp	ACC	MCC	AUC
F.vesca	CB1	0.9456	0.9335	0.9395	0.8791	0.9730
	CB2	0.9250	**0.9507**	0.9376	0.8757	0.9745
	CB3	0.9300	0.9507	0.9402	0.8806	0.9402
	CB4	**0.9694**	0.9321	**0.9510**	**0.9025**	**0.9880**
	CB5	0.9300	0.9507	0.9402	0.8806	0.9750
R.chinensis	CB1	0.9177	0.9161	0.9169	0.8336	0.9600
	CB2	0.9118	0.9097	0.9198	0.8213	0.9560
	CB3	0.9118	0.8968	0.9045	0.8088	0.9560
	CB4	**0.9408**	**0.9394**	**0.9401**	**0.8800**	**0.9766**
	CB5	0.9176	0.9225	0.9200	0.8398	0.9630

Note: The best performance value for each metric across different stacking models is highlighted in bold.

same ratios were constructed for *R.chinensis*, i.e., 1:1 [143 positive and 143 negative samples], 1:5 [143 positive and 715 negative samples] and 1:15 [143 positive and 2145 negative samples]. The negative sequences did not overlap across ratio groups for both species. Fig. 8 presents the precision-recall curves (PRC) generated by our predictor on all independent datasets for both species.

For a fair comparison, we applied only one method, the i6mA-Fuse, a web-server which was recently constructed by Md. Mehedi Hasan et.al [27], To date, this server is the only web-based server for the identification of 6 mA sites in the *Rosaceae* genome. Although, there are several online tools for the prediction of 6 mA sites in different species,

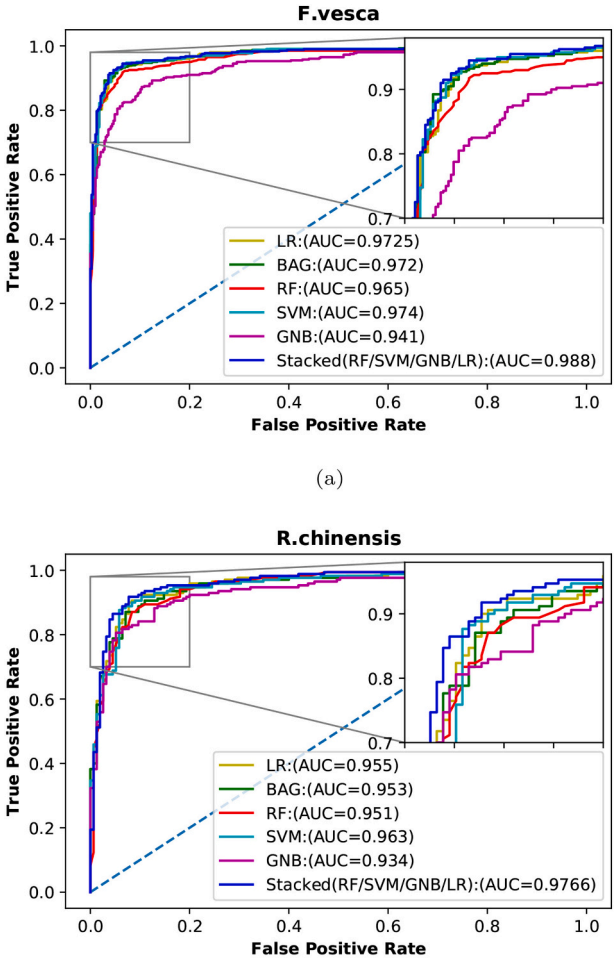


Fig. 7. ROC curves of LR, BAG, RF, SVM, GNB and stacked classifiers based on the benchmark datasets using five-fold CV for two species: (a) *F.vesca* (b) *R.chinensis*. The subfigures in the top-right corners of both figures show the magnified version of ROCs for selected regions.

these tools were not included in this study due to species-specific nature of 6 mA sites. We used the same positive/negative ratios of the independent datasets on the i6mA-Fuse web-server. The performance of

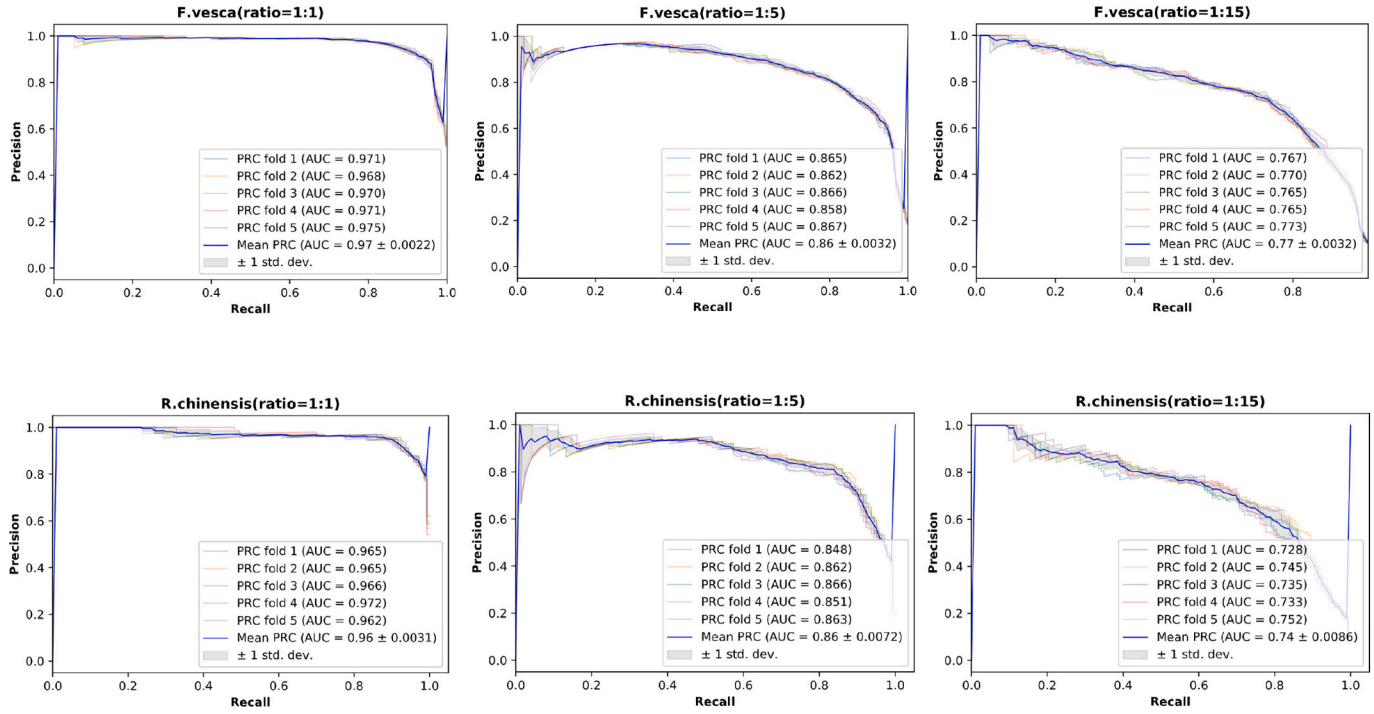


Fig. 8. Comparison of PRC generated by the i6mA-stack on the different ratios of the imbalanced independent test dataset for *F.vesca* and *R.chinensis*.

Table 5
The performance of the i6mA-stack and i6mA-Fuse on the independent datasets with different ratios.

Species	Method	Sn	Sp	ACC	MCC	AUC
F.vesca	**i6mA-Fuse**					
	ratio of [1:1]	0.9379	0.8889	0.9148	0.8292	0.9520
	ratio of [1:5]	0.9128	0.8828	0.8991	0.7965	0.9510
	ratio of [1:15]	0.9142	0.8516	0.8850	0.7734	0.9400
	i6mA-stack					
	ratio of [1:1]	0.9106	0.9711	0.9409	0.8834	0.9700
	ratio of [1:5]	0.9106	0.9711	0.9510	0.8801	0.9600
	ratio of [1:15]	0.9186	0.9481	0.9293	0.8593	0.9600
R. chinensis	**i6mA-Fuse**					
	ratio of [1:1]	0.9181	0.9028	0.9111	0.8209	0.9510
	ratio of [1:5]	0.9075	0.8973	0.9028	0.8044	0.9488
	ratio of [1:15]	0.8870	0.8904	0.8885	0.7758	0.9075
	i6mA-stack					
	ratio of [1:1]	0.9412	0.9281	0.9344	0.8688	0.9700
	ratio of [1:5]	0.9235	0.9028	0.9140	0.8268	0.9700
	ratio of [1:15]	0.9023	0.9028	0.9025	0.8038	0.9600

the i6mA-Fuse and i6mA-stack is presented in Table 5. For *F.vesca*, the i6mA-Fuse yielded an Sn of 0.9379, Sp of 0.8889, ACC of 0.9148, MCC of 0.8292, and AUC of 0.9520 for the 1:1 ratio independent dataset. These parameters were improved with the i6mA-stack, with the exception of Sn: Sp increased by 8.22%, ACC increased by 2.61%, MCC increased by 5.42%, and AUC increased by 1.8%. For *R.chinensis* the i6mA-stack model outperformed Sn by 2.31%, Sp by 2.53%, ACC by 2.33%, MCC by 4.79%, and AUC by 1.9%. For the independent dataset with ratio of 1:5, the performance was improved, except sensitivity, in *F.vesca* by 8.83%, 5.19%, 8.45%, and 0.9% in terms of specificity, accuracy, MCC, and AUC, respectively, while the performance was improved in *R.chinensis* by 1.6%, 0.55%, 1.12%, 2.24% and 2.24% in terms of sensitivity, specificity, accuracy, MCC, and AUC, respectively. Similarly, when predictions were evaluated based on the independent dataset of *F.vesca* with the ratio of 1:15, the i6mA-stack outperformed the i6mA-Fuse by 0.44%, 9.65%, 4.43%, 8.59%, and 2% based on Sn, Sp, ACC, MCC, and AUC, respectively, while for the *R.chinensis*, the i6mA-stack outperformed the i6mA-Fuse in Sn by 1.53%, Sp by 1.24%, ACC by 1.4%, MCC by 2.8%, and AUC by 5.25%. Based on the independent test results of the imbalanced datasets, our method add robust and stable against the increasing ratios of the imbalanced dataset while the i6mA-Fuse does not perform well against imbalanced dataset.

These results indicate that the proposed model, i6mA-stack, improves on the existing method and is thus a promising predictor in the field of computational biology and related academia. The superior performance of the i6mA-stack can be explained by the following aspects: reduction of dataset redundancy, selection of the ONF through different encoding schemes, and appropriate combinations of the base-learners as well as the meta-learner. On the other hand, although significant effort have been made towards theoretical analysis of stack-ensemble [63,64], a full understanding of the mechanism of the stack-ensemble model has yet to be attained. It should be noted that our approach is intended for combining base-classifiers that are heterogeneous (diverse in characters) and strong (i.e. each of the base-classifiers performs relatively well in its own right), rather than homogeneous (of the same kind) and weak.

In addition, it is worth to mentioning that a recent paper (HLPpred-Fuse) [65] was used an ensemble approach to predict hemolytic peptide and its activity, it can be noticed that the out of six base-classifiers (RF, ERT, SVM, GB, AB, and KNN), some classifiers share the same underlying operating principle (homogeneous): for example, the RF and the ERT are the tree-based algorithms. Our method (used heterogeneous principle), therefore, is different than the methods similar to HLPpred-Fuse.

4. Web-server implementation

For convienience, we have developed a user-friendly and publicly accessible web-server using Python and Flask library. To use the web-server, the step-by-step user guide should be followed.

i. Please click on the link to reach the web-server at http://nsclbio.jbnu.ac.kr/tools/i6mA-stack/

ii. Users should select a species from the check list.

iii. Users can either paste or type the query DNA sequences into

the input box, or upload sequences in FASTA format.

iv. Click the 'submit sequences' button to get the results: the query results will be shown on the result page after a short time.

In addition, datasets, codes, trained models, and the ONF identification are available at https://github.com/Jhabindra-bioinfo2020/i6mA-stack

5. Conclusion

By fusing five different DNA sequence encoding schemes and extracting the best features through RFECV, a new stacking based machine learning method, named i6mA-stack, was developed for the prediction of 6 mA sites in *F.vesca* and *R.chinensis*. Our proposed model employs an ensemble of base learners, such as SVM, LR, RF, and GNB, to generate output for the SVM as the meta-learner. In summary, this study followed three main steps: training the linear SVM for extracting the most useful evolutionary information features from the RFECV method, using these to train the base-learner as the first step of stacking, and combining the output of the base-learner with the help of the linear SVM as the meta stage or second stage of stacking. As a result, the second stage SVM (meta-learner) of the i6ma-stack produced an ACC of 95.10%, MCC of 0.9025, and AUC of 0.9880 on the training dataset for *F.vesca*, and an ACC of 94.01%, MCC of 0.8800, and AUC of 0.9766 for *R.chinensis*. A comparison with an alternative method indicates that our proposed method provides good results based on the independent test datasets and could serve as a promising tool for predicting 6mAs in a DNA sequence. Moreover, we compared the stacking model with the performance of the individual ML-based algorithms adopted in this study, and the results showed that the i6mA-stack outperformed on the benchmark dataset. As a final point, we hoped that a reliable 6 mA sites identification and validation in cooperation with the stacking-based ML method will pave the way to more efficient computer-aided methylation sites prediction.

Funding

This work was supported in part by the National Research Foundation of Korea(NRF) grant funded by the Korea government (MSIT) (No. 2020R1A2C2005612), in part by the Brain Research Program of the National Research Foundation (NRF) funded by the Korean government (MSIT) (No. NRF-2017M3C7A1044816), and in part by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (No. 2019R1A6A3A01094685).

Declaration of Competing Interest

The authors declare no conflict of interest.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.ygeno.2020.09.054.

References

[1] T. Phillips, et al., The role of methylation in gene expression, Nat. Educ. 1 (1) (2008) 116.
[2] Z.K. O'Brown, E.L. Greer, N6-methyladenine: a conserved and dynamic dna mark, DNA Methyltransferases-Role and Function, Springer, 2016, pp. 213–246.
[3] Y. Fu, G.-Z. Luo, K. Chen, X. Deng, M. Yu, D. Han, Z. Hao, J. Liu, X. Lu, L.C. Doré, et al., N6-methyldeoxyadenosine marks active transcription start sites in chlamydomonas, Cell 161 (4) (2015) 879–892.
[4] E.L. Greer, M.A. Blanco, L. Gu, E. Sendinc, J. Liu, D. Aristizábal-Corrales, C.-H. Hsu, L. Aravind, C. He, Y. Shi, Dna methylation on n6-adenine in c. elegans, Cell 161 (4) (2015) 868–878.
[5] G. Zhang, H. Huang, D. Liu, Y. Cheng, X. Liu, W. Zhang, R. Yin, D. Zhang, P. Zhang, J. Liu, et al., N6-methyladenine dna modification in drosophila, Cell 161 (4) (2015) 893–906.
[6] Z. Liang, G. Yu, J. Liu, Y. Geng, J. Mao, D. Wang, J. Zhou, X. Gu, The n (6)-adenine methylation in yeast genome profiled by single-molecule technology, J. Genet. Genomics 45 (2018) 223–225.
[7] M.J. Koziol, C.R. Bradshaw, G.E. Allen, A.S. Costa, C. Frezza, J.B. Gurdon, Identification of methylated deoxyadenosines in vertebrates reveals diversity in dna modifications, Nat. Struct. Mol. Biol. 23 (1) (2016) 24.
[8] J. Liu, Y. Zhu, G.-Z. Luo, X. Wang, Y. Yue, X. Wang, X. Zong, K. Chen, H. Yin, Y. Fu, et al., Abundant dna 6ma methylation during early embryogenesis of zebrafish and pig, Nat. Commun. 7 (1) (2016) 1–7.
[9] C. Zhou, C. Wang, H. Liu, Q. Zhou, Q. Liu, Y. Guo, T. Peng, J. Song, J. Zhang, L. Chen, et al., Identification and analysis of adenine n 6-methylation sites in the rice genome, Nat. Plants 4 (8) (2018) 554–563.
[10] T.P. Wu, T. Wang, M.G. Seetin, Y. Lai, S. Zhu, K. Lin, Y. Liu, S.D. Byrum, S.G. Mackintosh, M. Zhong, et al., Dna methylation on n 6-adenine in mammalian embryonic stem cells, Nature 532 (7599) (2016) 329–333.
[11] B. Yao, Y. Cheng, Z. Wang, Y. Li, L. Chen, L. Huang, W. Zhang, D. Chen, H. Wu, B. Tang, et al., Dna n6-methyladenine is dynamically regulated in the mouse brain following environmental stress, Nat. Commun. 8 (1) (2017) 1–10.
[12] Z. Liang, L. Shen, X. Cui, S. Bao, Y. Geng, G. Yu, F. Liang, S. Xie, T. Lu, X. Gu, et al., Dna n6-adenine methylation in arabidopsis thaliana, Dev. Cell 45 (3) (2018) 406–416.
[13] Q. Zhang, Z. Liang, X. Cui, C. Ji, Y. Li, P. Zhang, J. Liu, A. Riaz, P. Yao, M. Liu, et al., N6-methyladenine dna methylation in japonica and indica rice genomes and its association with gene expression, plant development, and stress responses, Mol. Plant 11 (12) (2018) 1492–1508.
[14] C.-L. Xiao, S. Zhu, M. He, D. Chen, Q. Zhang, Y. Chen, G. Yu, J. Liu, S.-Q. Xie, F. Luo, et al., N6-methyladenine dna modification in the human genome, Mol. Cell 71 (2) (2018) 306–318.
[15] Z.-Y. Liu, J.-F. Xing, W. Chen, M.-W. Luan, R. Xie, J. Huang, S.-Q. Xie, C.-L. Xiao, Mdr: an integrative dna n6-methyladenine and n4-methylcytosine modification database for rosaceae, Hortic. Res. 6 (1) (2019) 1–7.
[16] D. Dunn, J. Smith, The occurrence of 6-methylaminopurine in deoxyribonucleic acids, Biochem. J. 68 (4) (1958) 627–636.
[17] M.A. Gorovsky, S. Hattman, G.L. Pleger, [6n] methyl adenine in the nuclear dna of a eucaryote, tetrahymena pyriformis, J. Cell Biol. 56 (3) (1973) 697–701.
[18] S. Frelon, T. Douki, J.-L. Ravanat, J.-P. Pouget, C. Tornabene, J. Cadet, High-performance liquid chromatography- tandem mass spectrometry measurement of radiation-induced base damage to isolated and cellular dna, Chem. Res. Toxicol. 13 (10) (2000) 1002–1010.
[19] B.A. Flusberg, D.R. Webster, J.H. Lee, K.J. Travers, E.C. Olivares, T.A. Clark, J. Korlach, S.W. Turner, Direct detection of dna methylation during single-molecule, real-time sequencing, Nat. Methods 7 (6) (2010) 461.
[20] G.-Z. Luo, M.A. Blanco, E.L. Greer, C. He, Y. Shi, Dna n 6-methyladenine: a new epigenetic mark in eukaryotes? Nat. Rev. Mol. Cell Biol. 16 (12) (2015) 705–710.
[21] L. Kong, L. Zhang, i6ma-dncp: computational identification of dna n6-methyladenine sites in the rice genome using optimized dinucleotide-based features, Genes 10 (10) (2019) 828.
[22] Q.F. Huang, J. Zhang, 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.
[23] S. Basith, B. Manavalan, T.H. Shin, G. Lee, Sdm6a: a web-based integrative machine-learning framework for predicting 6ma sites in the rice genome, Mol. Therapy Nucleic Acids 18 (2019) 131–141.
[24] 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 (1) (2019) 96–102.
[25] 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.
[26] W. Chen, H. Lv, F. Nie, H. Lin, i6ma-pred: identifying dna n6-methyladenine sites in the rice genome, Bioinformatics 35 (16) (2019) 2796–2800.
[27] 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. (2020) 1–10.
[28] W. Chen, H. Yang, P. Feng, H. Ding, H. Lin, idna4mc: identifying dna n4-methylcytosine sites based on nucleotide chemical properties, Bioinformatics 33 (22) (2017) 3518–3523.
[29] L. Fu, B. Niu, Z. Zhu, S. Wu, W. Li, Cd-hit: accelerated for clustering the next-generation sequencing data, Bioinformatics 28 (23) (2012) 3150–3152.
[30] R.K. Dale, B.S. Pedersen, A.R. Quinlan, Pybedtools: a flexible python library for manipulating genomic datasets and annotations, Bioinformatics 27 (24) (2011) 3423–3424.
[31] J. Khanal, I. Nazari, H. Tayara, K.T. Chong, 4mccnn: identification of n4-methylcytosine sites in prokaryotes using convolutional neural network, IEEE Access 7 (2019) 145455–145461.
[32] L. Wei, R. Su, S. Luan, Z. Liao, B. Manavalan, Q. Zou, X. Shi, Iterative feature representations improve n4-methylcytosine site prediction, Bioinformatics 35 (23) (2019) 4930–4937.
[33] A.S. Nair, S.P. Sreenadhan, A coding measure scheme employing electron-ion interaction pseudopotential (eiip), Bioinformation 1 (6) (2006) 197.
[34] C. Jia, Q. Yang, Q. Zou, Nucpospred: predicting species-specific genomic nucleosome positioning via four different modes of general pseknc, J. Theor. Biol. 450 (2018) 15–21.
[35] B. Liu, K. Li, D.-S. Huang, K.-C. Chou, Ienhancer-el: identifying enhancers and their strength with ensemble learning approach, Bioinformatics 34 (22) (2018)

3835–3842.
[36] B. Manavalan, S. Basith, T.H. Shin, L. Wei, G. Lee, Meta-4mcpred: a sequence-based meta-predictor for accurate dna 4mc site prediction using effective feature representation, Mol. Therapy Nucleic Acids 16 (2019) 733–744.
[37] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al., Scikit-learn: machine learning in python, J. Mach. Learn. Res. 12 (2011) 2825–2830.
[38] T. Hastie, R. Tibshirani, J. Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Springer Science & Business Media, 2009.
[39] V.N. Vapnik, An overview of statistical learning theory, IEEE Trans. Neural Netw. 10 (5) (1999) 988–999.
[40] L. Breiman, Random forests, Mach. Learn. 45 (1) (2001) 5–32.
[41] G. H. John, P. Langley, Estimating Continuous Distributions in Bayesian Classifiers, arXiv preprint arXiv:1302.4964.
[42] L. Breiman, Bagging predictors, Mach. Learn. 24 (2) (1996) 123–140.
[43] D.H. Wolpert, Stacked generalization, Neural Netw. 5 (2) (1992) 241–259.
[44] Z.-H. Zhou, Ensemble Methods: Foundations and Algorithms, CRC press, 2012.
[45] J. Tang, S. Alelyani, H. Liu, Data Classification: Algorithms and Applications, Data Mining and Knowledge Discovery Series, CRC Press, 2014, pp. 37–64.
[46] L. Breiman, Stacked regressions, Mach. Learn. 24 (1) (1996) 49–64.
[47] E. Frank, M. Hall, L. Trigg, G. Holmes, I.H. Witten, Data mining in bioinformatics using weka, Bioinformatics 20 (15) (2004) 2479–2481.
[48] Q. Hu, C. Merchante, A.N. Stepanova, J.M. Alonso, S. Heber, A stacking-based approach to identify translated upstream open reading frames in arabidopsis thaliana, International Symposium on Bioinformatics Research and Applications, Springer, 2015, pp. 138–149.
[49] A. Mishra, P. Pokhrel, M.T. Hoque, Stackdppred: a stacking based prediction of dna-binding protein from sequence, Bioinformatics 35 (3) (2019) 433–441.
[50] S. Nagi, D.K. Bhattacharyya, Classification of microarray cancer data using ensemble approach, Network Model. Anal. Health Inform. Bioinforma. 2 (3) (2013) 159–173.
[51] J. Wang, H. Zheng, Y. Yang, W. Xiao, T. Liu, Preddbp-stack: prediction of dna-binding proteins from hmm profiles using a stacked ensemble method, Biomed. Res. Int. 2020 (2020) 1–8.
[52] P. Charoenkwan, C. Nantasenamat, M.M. Hasan, W. Shoombuatong, Meta-ipvp: a sequence-based meta-predictor for improving the prediction of phage virion proteins using effective feature representation, J. Comput. Aided Mol. Des. (2020) 1–12.
[53] H. Tayara, M. Tahir, K.T. Chong, Identification of prokaryotic promoters and their strength by integrating heterogeneous features, Genomics 112 (2) (2020) 1396–1403.
[54] W. Alam, S. D. Ali, H. Tayara, K. T. Chong, A cnn-based rna n6-methyladenosine site predictor for multiple species using heterogeneous features representation, IEEE Access.
[55] J. Khanal, H. Tayara, K.T. Chong, Identifying enhancers and their strength by the integration of word embedding and convolution neural network, IEEE Access 8 (2020) 58369–58376.
[56] M. M. Hasan, B. Manavalan, W. Shoombuatong, M. S. Khatun, H. Kurata, i4mc-mouse: improved identification of dna n4-methylcytosine sites in the mouse genome using multiple encoding schemes, Comput. Struct. Biotechnol. J.
[57] T. Saito, M. Rehmsmeier, The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets, PLoS One 10 (3) (2015) e0118432.
[58] L.V.D. Maaten, Accelerating t-sne using tree-based algorithms, J. Mach. Learn. Res. 15 (1) (2014) 3221–3245.
[59] V. Vacic, L.M. Iakoucheva, P. Radivojac, Two sample logo: a graphical representation of the differences between two sets of sequence alignments, Bioinformatics 22 (12) (2006) 1536–1537.
[60] W. Chen, X. Song, H. Lv, H. Lin, Irna-m2g: identifying n2-methylguanosine sites based on sequence-derived information, Mol. Therapy Nucleic Acids 18 (2019) 253–258.
[61] B. Manavalan, S. Basith, T.H. Shin, D.Y. Lee, L. Wei, G. Lee, et al., 4mcpred-el: an ensemble learning framework for identification of dna n4-methylcytosine sites in the mouse genome, Cells 8 (11) (2019) 1332.
[62] M.M. Hasan, B. Manavalan, M.S. Khatun, H. Kurata, i4mc-rose, a bioinformatics tool for the identification of dna n4-methylcytosine sites in the rosaceae genome, Int. J. Biol. Macromol. 157 (2020) 752–758.
[63] L.I. Kuncheva, C.J. Whitaker, Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy, Mach. Learn. 51 (2) (2003) 181–207.
[64] K.M. Ting, I.H. Witten, Issues in stacked generalization, J. Artif. Intell. Res. 10 (1999) 271–289.
[65] M.M. Hasan, N. Schaduangrat, S. Basith, G. Lee, W. Shoombuatong, B. Manavalan, Hlppred-fuse: improved and robust prediction of hemolytic peptide and its activity by fusing multiple feature representation, Bioinformatics 36 (11) (2020) 3350–3356.