Send Orders for Reprints to reprints@benthamscience.net

RESEARCH ARTICLE

An Improved Computational Prediction Model for Lysine Succinylation Sites Mapping on *Homo sapiens by* Fusing Three Sequence Encoding Schemes with the Random Forest Classifier

Samme Amena Tasmia[1], Fee Faysal Ahmed[2], Parvez Mosharaf[1], Mehedi Hasan[3] and Nurul Haque Mollah[1,*]

[1]*Bioinformatics Lab., Department of Statistics, Rajshahi University, Rajshahi-6205, Bangladesh;* [2]*Department of Mathematics, Jashore University of Science and Technology, Jashore, Bangladesh;* [3]*Department of Bioscience and Bioinformatics, Kyushu Institute of Technology, Fukuoka, Japan*

ARTICLE HISTORY

Received: May 20, 2020
Revised: December 13, 2020
Accepted: January 06, 2021

DOI:
10.2174/1389202922666210219114211

Abstract: ***Background*:** Lysine succinylation is one of the reversible protein post-translational modifications (PTMs), which regulate the structure and function of proteins. It plays a significant role in various cellular physiologies including some diseases of human as well as many other organisms. The accurate identification of succinylation site is essential to understand the various biological functions and drug development.

***Methods*:** In this study, we developed an improved method to predict lysine succinylation sites mapping on *Homo sapiens by* the fusion of three encoding schemes such as binary, the composition of *k*-spaced amino acid pairs (CKSAAP) and amino acid composition (AAC) with the random forest (RF) classifier. The prediction performance of the proposed random forest (RF) based on the fusion model in a comparison of other candidates was investigated by using 20-fold cross-validation (CV) and two independent test datasets were collected from two different sources.

***Results*:** The CV results showed that the proposed predictor achieves the highest scores of sensitivity (SN) as 0.800, specificity (SP) as 0.902, accuracy (ACC) as 0.919, Mathew correlation coefficient (MCC) as 0.766 and partial AUC (pAUC) as 0.163 at a false-positive rate (FPR) = 0.10 and area under the ROC curve (AUC) as 0.958. It achieved the highest performance scores of SN as 0.811, SP as 0.902, ACC as 0.891, MCC as 0.629 and pAUC as 0.139 and AUC as 0.921 for the independent test protein set-1 and SN as 0.772, SP as 0.901, ACC as 0.836, MCC as 0.677 and pAUC as 0.141 at FPR = 0.10 and AUC as 0.923 for the independent test protein set-2. It also outperformed all the other existing prediction models.

***Conclusion*:** The prediction performances as discussed in this article recommend that the proposed method might be a useful and encouraging computational resource for lysine succinylation site prediction in the case of human population.

Keywords: Protein sequences, lysine succinylation site, prediction, encoding schemes, feature selection, random forest, fusion model.

1. INTRODUCTION

Succinylation is one type of reversible protein post-translational modifications (PTMs) in which a succinyl group (-CO-CH_2-CH_2-CO_2H) is transferred to a protein molecule at its lysine (K) residue. This modification is transpired in many protein molecules to change their structure and functions in both eukaryotic and prokaryotic organisms [1-3] and altered lysine's charge from +1 to −1 (at physiological pH) and produced comparatively large structural moiety (100 Da), bigger than acetylation (42 Da) or methylation (14 Da) [4]. It is a central regulatory PTM in many biological processes and plays a significant role in various cellular physiologies of human as well as many other organisms [2, 4, 5]. Recently, several empirical methods have been built to detect succinylated protein including high-performance liquid chromatography assays, chromatography-mass spectrometry and spectrophotometric assays [6, 7]. In the previous years, different massive proteomic technologies were widely developed to detect succinylated protein in various organisms, including *Escherichia coli*, *Mycobacterium tuberculosis*, *Toxoplasma gondii*, *Saccharomyces cerevisiae*, *Homo sapiens* and *Mus musculus*, and currently in plants [1, 4, 8-14]. However, the experiential technologies are often time-consuming, cost-effective and difficult to detect exact modifications of protein.

There were few computational predictors in the literature for predicting succinylated protein using a web server [14-18]. Hasan *et al.* developed two predictors termed SuccinSite2.0 [14] and SuccinSite [15] with the combination of RF classifier scores based on the amino acid frequency and properties. Huang et al developed a computational predictor, named CNN-SuccSite, which has been developed based on deep learning architectures with different encoding schemes [16]. Recently, Ning *et al.* developed HybridSucc using

*Address correspondence to this author at the Bioinformatics Lab., Department of Statistics, Rajshahi University, Rajshahi-6205, Bangladesh;
E-mail: mollah.stat.bio@ru.ac.bd

1389-2029/21 $65.00+.00

Group-based Prediction System (GPS) *via* diverse encoding systems including k-space amino acid pair composition (CKSAAP), amino acid index (AAindex) physicochemical properties and pseudo amino acid composition (PseAAC) [17]. More recently, Hasan *et al*. also suggested a predictor termed GPSuc, by combining five sequence encoding schemes *i.e.* profile-based composition of *k*-spaced amino acid pairs (pCKSAAP), Amino acid composition (AAC), AAindex, binary amino acid codes (BE), and position-specific scoring matrix *(*PSSM). They used the selected feature vectors and random forest (RF) classifier to build the predictor [18]. Their sensitivity was reported to be less than 70%. So, further improvement was expected that can supplement the existing Lysine succinylation site prediction techniques. In this study, an attempt was made to develop an improved prediction model by fusing few encoding schemes with the machine learning approaches to predict lysine succinylation sites mapping on *Homo sapiens*.

We introduced the necessary materials and methods for the development of the proposed computational procedure in section 2. The summary results and their discussions were given in sections 3 and 4, respectively, and section 4 presents the conclusion of this study.

2. MATERIALS AND METHODS

2.1. Data Sources and Descriptions

In this study, we considered two protein datasets. Both datasets were collected from the Uni-ProtKB/Swiss-Prot and NCBI protein sequence databases. The dataset-1 contained experimentally validated 1700 lysine succinylated protein sequences mapping on *Homo sapiens*. The dataset-2 contained experimentally validated 704 lysine succinylated protein sequences mapping on human pathogen *Histoplasma capsulatum*. The succinylated sites for both datasets are also known as positive windows and the remaining lysine residues in the protein sequence for both datasets were considered as the non-succinylated sites (negative windows) as suggested [15, 18, 19]. Dataset-1 was used to develop the prediction model and dataset-2 was used to investigate the performance of the prediction model as described in detail in the next subsection 2.2.

2.2. Data Preparation and Overview on the Development of the Proposed Computational Predictor

The prediction performance not only depends on the estimation of the model parameters but also depends on the selection of tuning parameters like CD-HIT (Cluster Database at High Identity with Tolerance) threshold [20-23], window size of protein sequence, ratio of positive and negative windows, encoding scheme, features and classifier. The appropriate value of CD-HIT threshold (CHT), window size (WS) and ratio of positive and negative windows depends on the dataset [15-18, 24-26]. As for example, Hasan and Kurata (2018) [18] used CHT=30%, WS=41 and ratio=1:2; Manavalan *et al.* (2018) [25] used CHT=30%, WS=41 and ratio=1:2; Chen *et al.* (2019) [26] used CHT=80%, WS=21 and ratio=1:2; and Mosharaf *et al.* (2020) [24] used CHT=40%, WS=27 and ratio=1:1 to develop their prediction models. Thus, different authors used different CHTs, WSs and ratios of positive and negative windows to develop their prediction models. However, none of them discussed how they selected these tuning parameters. In this case, we tried to discuss this issue slightly. At first, we considered three CHTs 30%, 40% and 50% to select the more appropriate one of them for removing identical sequences from the protein dataset-1 to solve the overprediction problem. The protein dataset-1 was reduced to three datasets of sizes 550, 665 and 824 based on 3 values of CHT, respectively. We observed that all predictors showed better performance with the protein dataset of size 550 corresponding to CHT=30%. Thereafter, we considered the dataset-1 of size p_{11}=550 corresponding to CHT=30% to develop the predictor. The dataset-1 was partitioned into a training dataset and independent test dataset-1, where the test dataset-1 was constructed by taking p_{12}=50 proteins from the protein dataset-1 that were also used in the previously published prediction model as the independent test dataset [14, 16-18] for a fair comparison with our proposed model. Rest of the 500 proteins were used to construct the training dataset. There were 1195 positive and 16842 negative window sites within all 500 proteins of the training dataset. On the other hand, there were 54 positive and 2004 negative window sites within all 50 proteins of the independent test dataset-1. We also constructed independent test dataset-2 by taking p_{21}=202 proteins randomly from the protein dataset-2 for a more fair investigation on the performance of the prediction model. There were 463 positive and 6742 negative window sites within all 202 proteins of the independent test dataset-2.

The training dataset of size p_{11}=500 was used to train different candidate predictors as described below. The performance of prediction models depends on the window size as mentioned previously [15-18, 24-26]. Therefore, the training dataset was partitioned into positive and negative window samples corresponding to each of the window sizes 19, 25**,** 31, 37 and 41, respectively to select one of them as a more appropriate window size. Each site was defined as a peptide segment of 2w+1 length with lysine (K) in the center, where 2w+1 is the window size. We observed that the predictor was optimized corresponding to the window size 2w+1=25. Thus, we considered positive and negative window samples corresponding to the window size 25 to develop a better predictor. Obviously, positive window samples (n_1=1195) and negative window samples (n_2=16842) in the training dataset were unbalanced. To construct a comparatively balanced dataset, the training datasets were constructed at three ratios 1:1, 1:2, and 1:3 of positive and negative window samples, respectively by randomly taking the negative windows out of n_2=16842 for each ratio case. The training dataset of ratio 1:1 was constructed by taking all 1195 positive windows with the randomly selected 1195 negative windows out of 16842. The training dataset of ratio 1:2 was constructed by taking all 1195 positive windows with the randomly selected 1195x2=2390 negative windows out of 16842. Similarly, the training dataset of ratio 1:3 was constructed by taking all 1195 positive windows with the randomly selected 1195x3=3585 negative windows out of 16842. We developed the prediction model with each training dataset and investigated their performance by using 20-fold cross-validation (CV) and two independent test datasets as introduced previously. We observed that the predictor was optimized corresponding to the 1:2 ratio of positive and negative window samples.

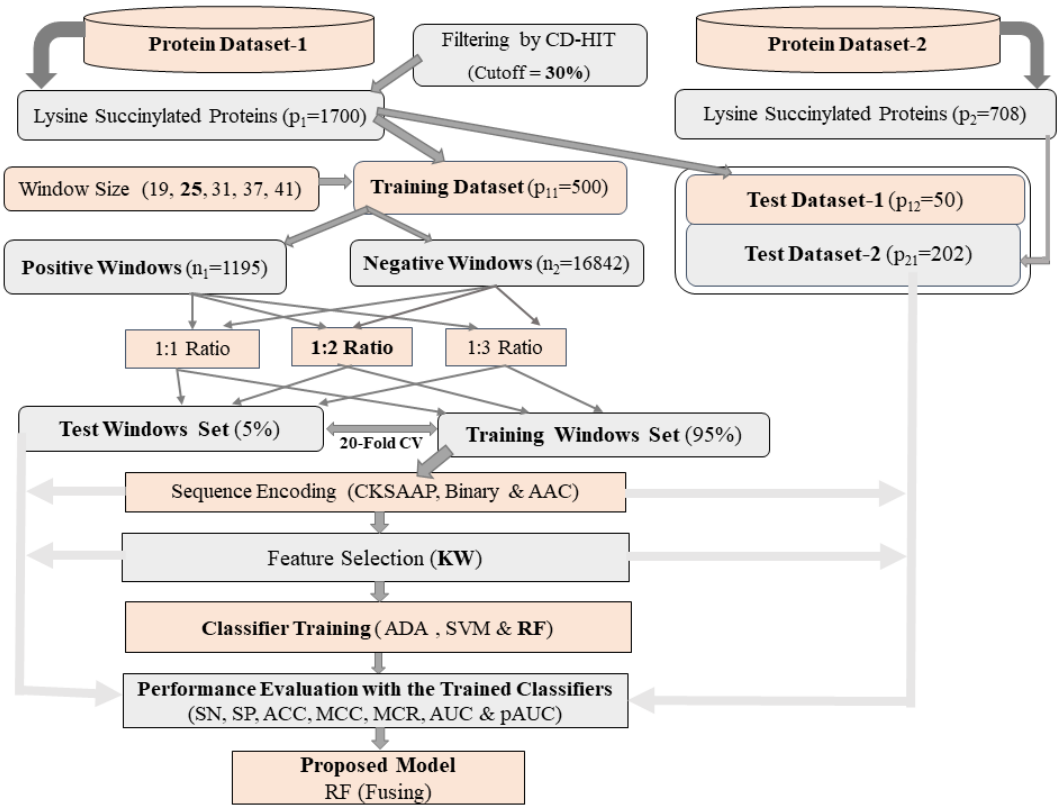

Fig. (1). Overview on the development of the proposed predictor.

To convert the protein window sequence data to numeric data, we considered three popular encoding schemes *i.e.* CKSAAP, binary and AAC (see section 2.4). To remove the unimportant encoded features from the dataset for reducing the computational load of the predictor, we considered a non-parametric feature selection approach known as Kruskal-Wallis (KW) [22] to select a better one of them. We trained 3 different classifiers ADA [23], SVM [27] and RF [28] (see section 2.6) separately based on the selected features of each encoding system. Then we improved the prediction model by fusing three encoding schemes (see section 2.7). We observed that the proposed RF-based fusion model is optimized corresponding to the 1:2 ratio of positive and negative window samples. Thus, we built an improved computational prediction model as displayed in Fig. (**1**) (see the results section for further more discussion).

2.3. Two Sample Logo (TSL) Analyses

The TSL analysis of protein sequences is used to visualize significant differences between the amino acid samples of positive and negative window groups. It determines the statistically significant residues (amino acids) around the protein PTM site for finding the differences between the two window groups. Statistical evidence is computed for each amino acid at every position between two window groups under the null hypothesis (Ho) that the residue samples follow the same distribution in both positive and negative window groups.

Let *A* and *B* be two groups of protein sequences based on the positive and negative windows. Let |*A*| and |*B*| be the number of sequences in these groups. Let *N* denote the length of each window in both groups. Let A_i be the *i*th sequence in group *A* and let $A_{i,j}$ be the *j*th position in A_i. Let $X_A^{j,r} = 1, \text{if } A_{i,j} = r, \text{otherwise } X_A^{j,r} = 0$, where r is the symbol of a residue. The vector $X_B^{j,p}$is formed conversely. Then we calculated the *p*-value of H_0 that both vectors $X_A^{j,p}$ and $X_B^{j,p}$ follow the same distribution.

To test H_0, one of the two types of statistical tests (i) two-sample t-test and (ii) binomial test, is used usually. It should be noted here that the t-test is less accurate but significantly faster, while the binomial test is more accurate, but significantly slower [29]. The TSL exhibits two kinds of graphical image: (i) significant amino acid symbols are plotted using the size of the symbol that is proportional to the difference between the two amino acid samples; (ii) significant amino acids are plotted based on the same size for each amino acid symbol. Amino acids are divided into two groups: (i) enriched in the positive window samples, and (ii) depleted in the positive window samples.

2.4. Data Encoding Scheme

To construct a robust prediction model, a numeric feature vector is essential to train the classifiers. There are several encoding approaches in the compositions to covert the sequence data into numeric data. Here, we considered three encoding approaches as discussed below:

2.4.1. CKSAAP Encoding

In various PTMs site prediction, there is a powerful feature encoding scheme named the composition of *k*-spaced

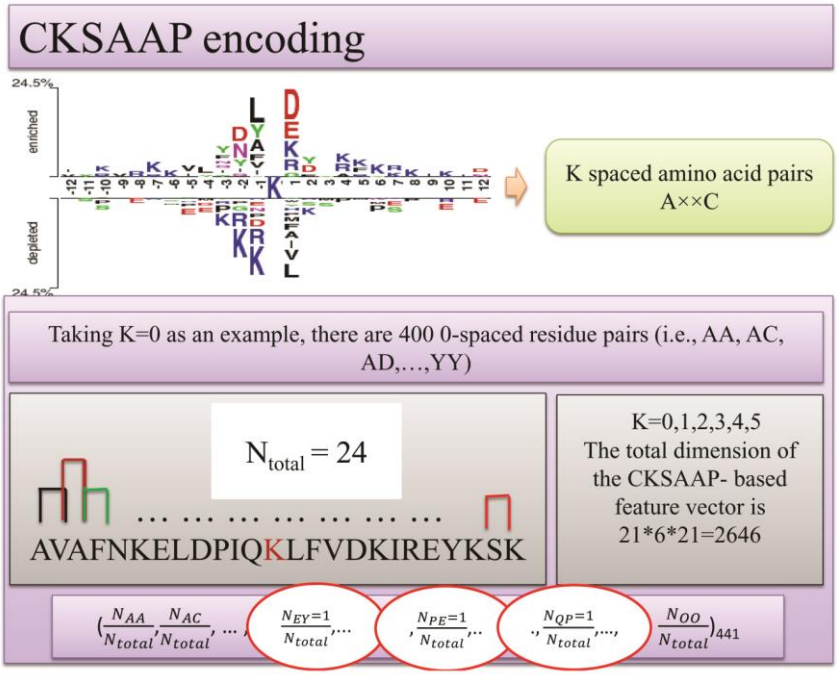

Fig. (2). A schematic diagram of CKSAAP encoding. *(A higher resolution/colour version of this figure is available in the electronic copy of the article).*

amino acid pairs (CKSAAP) [19]. It has been predominantly used in the numerous bioinformatics work [30-34]. In this paper, a sequence fragment of 25 amino acids is detected from the Succinylation or non- Succinylation site. For every single *k* (*k* denotes the gap between two amino acids), it may create (21× 21) = 441 (21 means 21 kinds of amino acids with the gap (O)) types of amino acid pairs (*i.e.* AA, AC, AD, . . . , OO), if window size of a fragment is 2*r* + 1. For each sequence, there is 21× (kmax + 1) × 21 = 2646 distinct amino acid combination developed for the highest *k* taking *k* max =5. Then, the feature vectors are calculated using the following equation:

$$\left(\frac{N_{AA}}{N_{total}}, \frac{N_{AC}}{N_{total}}, \frac{N_{AD}}{N_{total}}, \ldots, \frac{N_{OO}}{N_{total}}\right)_{441,} \quad (1)$$

where, N_{total} denotes the total composition residues length. N_{AA},N_{AC},..., N_{OO} are fragments' frequency of the amino acid pair. More details are available somewhere [14, 34]. A schematic diagram of CKSAAP encoding was depicted in Fig. (**2**).

2.4.2. Binary Encoding

In the binary encoding scheme, the number of residues may be less than 25 for the sites placed in k-terminal. In this binary encoding approach, 21 amino acids (including gap (O)) are reshaped to numeric vectors. Thus, 21 different amino acids (such as, ACDEFGHIKLMNPQRSTVWYO) are organized in this encoding scheme. Each amino acid is displayed in the query proteins by a 21-dimensional binary vector, *e.g.* A: 100000000000000000000, C: 010000000000 000000 000, …, O: 000000000000000000001, *etc.* In each window of succinylation site, the central location is always K, which is unnecessary to be taken into account. The whole dimension of this encoding scheme is (21X(25-1)) = 504 when we select a window of size 25. Details are described in previous studies [15, 16].

2.4.3. AAC Encoding

Amino acid composition (AAC) is a typical attribute used to examine substrate site motifs. AAC determines the probability of amino acids occurring in the flanking region of PTM sites [35, 36]. It can create protein arrangements data by recreating amino acid event frequencies. In this research, AAC was determined dependent on amino acid event frequencies in the arrangement sections encompassing the succinylation and non-succinylation sites (the site itself is not tallied). For every grouping section, 20 frequencies were determined for 20 types of amino acids. Given a divided arrangement x with a 25-mer string length, $n_x(m)$ is the quantity of a particular amino acid, m, occurring in the section, where m indicates the 20 amino acids. Thus, the probability $P_x(m)$ of particular amino acid m is [37]

$$P_x(m) = \frac{n_x(m)}{\sum_{m=1}^{20} n_x(m)} ; k=1,\ldots, 20$$

Then, the creation of the 20 amino acids can be changed to a 20-dimensional numeric vector Vx for the divided sequence x:

$$V_x = [P_x(1), P_x(2), \ldots, P_x(20)]$$

2.5. Feature Selection from the Encoded Data

A high dimensional feature vector is created by encoding each of the succinylated and non-succinylated fragments as discussed in section 2.4. However, some features are not so important to develop the prediction model. These unimportant features are removed from the dataset to reduce the computation burden from the prediction model. An encoded feature is said to be unimportant if its mean difference between succinylated and non-succinylated groups is statisti-

cally insignificant. In this study, we considered the top 1500 and 400 most important features based on CKSAAP and binary encoding schemes respectively to develop the prediction model. To remove the unimportant features from the prediction model, we considered the non-parametric Kruskal-Wallis (KS) test procedure [22], since the distribution of feature components is unknown.

2.6. Learning Classifiers

To build a better predictor for protein Succinylation site prediction, we considered 3 popular classifiers namely, Random Forest (RF), AdaBoost (ADA), & Support Vector Machine (SVM), for a comparison based on the encoded protein sequences. Let us consider a dataset consisting of *n* training data $(\boldsymbol{x}_1,y_1)$, $(\boldsymbol{x}_2,y_2)$... $(\boldsymbol{x}_n,y_n)$, where $\boldsymbol{x}_i$ is an input vector in space $X\subseteq R^m$ and y_i is the response variable that takes value +1 (succinylated site) and -1 (non-succinylated site). The objective is to classify a new window sample $\boldsymbol{x}$ into one of two classes (+1, -1). For convenience of the readers, let us introduce together those classifiers as follows:

2.6.1. Random Forest (RF)

The random forest (RF) classifier is a statistical learning algorithm. It is widelyhttps://www.thesaurus.com/browse/ magnificently used in protein bioinformatics [19, 24, 28, 30-32, 36-43]. Generally, it has two stages: one is random forest creation, and the other is to make a prediction from the random forest classifier created in the first stage. For the convenience of presentation, let (X, Y) ={$(\boldsymbol{x}_1,y_1)$, $(\boldsymbol{x}_2,y_2)$... $(\boldsymbol{x}_n,y_n)$}. Then B (b = 1, … , B) times selects a random sample (X_b, Y_b) with replacement from the given dataset (X,Y) and trains a regression tree $f_b$on (X_b, Y_b) to fit trees to these samples. After training, predictions for new samples $\boldsymbol{x}'$ can be written as,

$$\hat{f} = \frac{1}{B}\sum_{b=1}^{B} f_b(\boldsymbol{x}') \quad (2)$$

In this study, we have applied the RF classifier through the 'randomForest' R package [44].

2.6.2. AdaBoost

AdaBoost is an adaptive boosting machine learning meta-algorithm [23]. It is adaptive in the sense that subsequent weak learners are tweaked in favor of those instances misclassified by previous classifiers. Its technical description is as follows:

Feature vector: $\boldsymbol{x} = (x_1, x_2,\ldots,x_p)$

Training dataset: { ($\mathbf{x}_i$,y_i); i=1,2,…,n }

Suppose there are T weak classifiers defined by $f_t(\boldsymbol{x}); t = 1, 2, 3, \ldots, T)$ satisfying,

$$y_t = \text{sign}\big(f_t(\boldsymbol{x})\big) = \pm 1;\ t = 1, 2, 3, \ldots, T)$$

Then the AdaBoost classifier is defined by:

$$F_T(\boldsymbol{x}) = \sum_{t=1}^{T} \alpha_t f_t(\boldsymbol{x}),$$

where $\alpha_t = \frac{1}{2}\log\frac{1-\varepsilon_t(f_t)}{\varepsilon_t(f_t)}$, $\varepsilon_t(f_t) = \min_{f\in F} \varepsilon_t(f_t)$, $f_t = \operatorname{argmin}_{f\in F} \varepsilon_t(f)$, $\varepsilon_t(f) = \sum_{i=1}^{n} I(y_i \neq f_t(x_i))w_t(i)/\sum_{j=1}^{n} w_t(j)$, $w_{t+1}(i) = w_t(i)\exp\{-\alpha_t f_t(\boldsymbol{x_i})y_i\}$

Then the classification rule is defined as: $f_T(\boldsymbol{x}) = \text{sign}(F_T(\boldsymbol{x})) = \pm 1$,

In this paper, R package 'ada' was used to execute the AdaBoost algorithm [45].

2.6.3. Support Vector Machine (SVM)

The main objective of SVM is to find a hyperplane in an *m*-dimensional space that clearly classifies the data points [17, 19, 27, 46, 47]. Let us consider the data points consist of *n* training data $(\boldsymbol{x}_1,y_1)$, $(\boldsymbol{x}_2,y_2)$... $(\boldsymbol{x}_n,y_n)$, where $\boldsymbol{x}_i$ is an input vector in space $X\subseteq R^m$ and y_i is the output variable that takes value 1 (succinylated site) and -1 (non-succinylated site). The SVM approach constructs a hyperplane in high dimensional space, which can be used in both classification and regression. The hyperplane can be written as follows:

$$W^T X + b = 0 \quad (4)$$

Where b is scalar and W is an *m*-dimensional normalized vector and perpendicular to the separating hyperplane. If the data are linearly separable, the two classes are as follows: $W^T X + b > 0\ if\ y_i = 1$ and $W^T X + b < 0\ if\ y_i = -1$. If the data is not linearly separable, then SVM uses kernel functions to transform the original data space with high dimension space that can easily separate the classes as succinylated site and non-succinylated site. In such a case, the hyperplane can be written as:

$$f(\boldsymbol{x}) = \sum_{i=1}^{m} \alpha_i y_i\, K(\boldsymbol{x}_i, \boldsymbol{x}) + b \quad (5)$$

where, α_n is Lagrange multiplier, y_i is the class label that belongs to (-1, 1), and $K(\boldsymbol{x}_i, \boldsymbol{x})$ is the Kernel function between $\boldsymbol{x}_i$ and x. In this study, we have adopted kernel as a radial basis function (RBF). In this paper, R package 'e1071' was used to execute the SVM algorithm [48].

2.7. Fusion Model (Proposed)

Several authors used fusion technique to improve the performance of their prediction models [33, 40, 41].

In this article, we also attempted to improve the performance of our prediction model by fusing 3 encoding schemes Binary, CKSAAP and AAC with RF classifier as follows:

$$\text{RF(CKSAAP, Binary, AAC)} = w_1\times\text{RF(CKSAAP)} + w_2\times\text{RF(Binary)} + w_3\times\text{RF(AAC)},\ldots\ldots\ldots \quad (6)$$

where RF(Binary), RF(CKSAAP) and RF(AAC) represent the RF classification scores estimated with binary, CKSAAP and AAC encoding schemes, respectively. The values of w_1, w_2 and w_3 were selected based on the ratio of individual prediction performance of RF(Binary), RF(CKSAAP) and RF(AAC) satisfying $w_1 + w_2 + w_3 = 1$. Similarly, we improved the prediction performance of ADA and SVM classifiers by fusing binary, CKSAAP and AAC encoding schemes to compare with the proposed RF-based prediction model.

2.8. Performance Evaluation Measures

In order to compare the performance of different candidate prediction models, we considered some popular widely used measurements including true positive rate (TPR), which is also known as sensitivity (SN), true negative rate (TNR),

which is also known as specificity (SP), false-positive rate (FPR), false-negative rate (FNR), accuracy (ACC), misclassification rate (MCR) and Mathew correlation coefficient (MCC), receiving operating characteristics (ROC) curve, area under the ROC curve (AUC) and partial AUC (pAUC). These measurements are defined as follows:

$$\text{TPR} = \frac{n(TP)}{n(TP)+n(FN)};\ 0 \le TPR \le 1 \quad (7)$$

$$\text{FPR} = \frac{n(FP)}{n(TN)+n(FP)};\ 0 \le FPR \le 1 \quad (8)$$

$$\text{TNR} = \frac{n(TN)}{n(TN)+n(FP)};\ 0 \le TNR \le 1 \quad (9)$$

$$\text{FNR} = \frac{n(FN)}{n(TP)+n(FN)};\ 0 \le FNR \le 1 \quad (10)$$

$$ACC = \frac{n(TP)+n(TN)}{n(TP)+n(FP)+n(TN)+n(FN)};\ 0 \le ACC \le 1 \quad (11)$$

$$MCR = \frac{n(FP)+n(FN)}{n(TP)+n(FP)+n(TN)+n(FN)};\ 0 \le MCR \le 1 \quad (12)$$

$$MCC = \frac{(n(TP)\times n(TN))-(n(FP)\times n(FN))}{\sqrt{(n(TP)+n(FN))\times(n(TN)+n(FP))\times(n(TP)+n(FP))\times(n(TN)+n(FN))}};\ -1 \le MCC \le 1 \quad (13)$$

where, n(TN): number of true negative, n(TP): number of true positive, n(FN): number of false negative, n(FP): number of false positive. The ROC curve is created by plotting TPR against FPR. Obviously TPR+FNR=1, TNR+FPR=1, (FPR, FNR)→(0,0) implies MCR→0 and (TPR, TNR, ACC, MCC, AUC)→(1,1,1,1,1), conversely (FPR, FNR)→(1,1) implies MCR→1 and (TPR, TNR, ACC, MCC, AUC)→(0,0,0,0,0,-1). Therefore, the larger scores of TPR, TNR, ACC, MCC, AUC and conversely the smaller scores of FPR, FNR and MCR indicate better prediction performance. The more detailed descriptions about these evaluation measures were given in Supplementary file (**S1**).

2.8.1. K-fold Cross-Validation (CV)

In order to perform *K*-fold CV, the training dataset 'D' was randomly partitioned into *k* =20 mutually exclusives subsets (D_1, D_2,…,D_K) of almost equal sizes. Among *K=20* subsets, the (*K*– 1)=19 subsets were used as training data to train the prediction model, and the remaining one subset was used as the test subset to evaluate the performance of the prediction models. This procedure was then repeated *K*=20 times by replacing one subset from the training dataset with the test subset such that each subset is used only one time as the test dataset. Then, the average value of *K* =20 scores was used as a single score for each of the performance measures (TPR/SN, TNR/SP, FPR, FNR, MCR, ACC, MCC & AUC).

3. RESULTS

As mentioned previously in section 2.2, the prediction performance is not only influenced by the estimation of the model parameters but also depends on the selection of tuning parameters like CD-HIT threshold (CHT), window size (WS) and ratio of positive and negative windows. To select comparatively better combination of CHT, WS and ratio; we considered three CHTs (30%, 40%, 50%), five WSs (19, 25, 31, 37, 41) and three ratios (1:1, 1:2, 1:3). We developed seven-candidate prediction models based on three encoding schemes (Binary, CKSSAP and AAC) with each of three classifiers (ADA, SVM and RF) for each of the 45 combinations of CHT, WS and ratio. By the trial and performance checking strategy, we observed that the combination of CHT=30%, WS=25 and ratio=1:2 with each prediction model optimizes their performance scores (TPR/SN, TNR/SP, FPR, FNR, MCR, ACC, MCC & AUC) out of different 45 (=3×5×3) combinations. That is, the prediction performance scores (SN, SP, ACC, MCC & AUC) were maximized and conversely, the alternative performance scores (FPR, FNR & MCR) were minimized at CHT=30%, WS=25 and ratio=1:2. Thereafter, we only discussed the performance scores of 7×3=21 different prediction models based on three classifiers (ADA, SVM and RF) corresponding to the combination of tuning parameters at CHT=30%, WS=25 and ratio=1:2 to select the better prediction model. To examine the adequacy of the training dataset for the window size 25, two sample logo (TSL) analyses were performed in section 3.1. To select the better prediction model in a comparison of the others, first, we assessed the training performance scores in section 3.2. Then we investigated the prediction performance by using 20-fold CV and discussed the results in section 3.3. The performance scores based on two independent test datasets were discussed in section 3.4. Then we compared the performance of the proposed prediction model with four existing models [14, 16-18] in section 4.

3.1. The TSL Analysis

We performed two-sample logo (TSL) analyses to investigate the adequacy of the dataset. By TSL software, the neighboring succinylation and non-Succinylation sites are shown for the training dataset in Fig. (**3**) [49]. The positive or negative samples represented residues at each location above and under the X-axis plotted respectively, in proportion to the percentage of over represented (if positive) or underrepresented samples (if negative) shown by the height of the letter denoting the resultant residue. Y-axis indicates the cumulative percentage of these positive / negative residues. TSL logos represent the amino acid occurrences between positive and negative samples of succinylation protein. Only residues that were significantly enriched or depleted (*t*-test, $P<0.05$) flanking the centered succinylation sites are shown in Fig. (**3**). Significant differences were observed between positive and negative samples. We found that at specific points, some amino acids are over / under represented, which shows that the positional amino acid encoding is essential to identify the succinylation sites.

The binary encoding is encoded position-wise. Thus, the efficiency of binary encoding (Table **1**) is sufficient to exactly identify the succinylation sites. In the following calculation and operation, we selected 25-mer (-12, +12) window size and Fig. (**3**) shows the position-specific difference of amino acid compositions between succinylation sites and non-succinylation sites. We also provided the two sample logos for other window sizes (19, 31, 37 & 41) in the Supplementary file (Figs. **S1**-**S4**) for a visual representation of patterns of amino acid conserved-ness around the lysine succinylation sites.

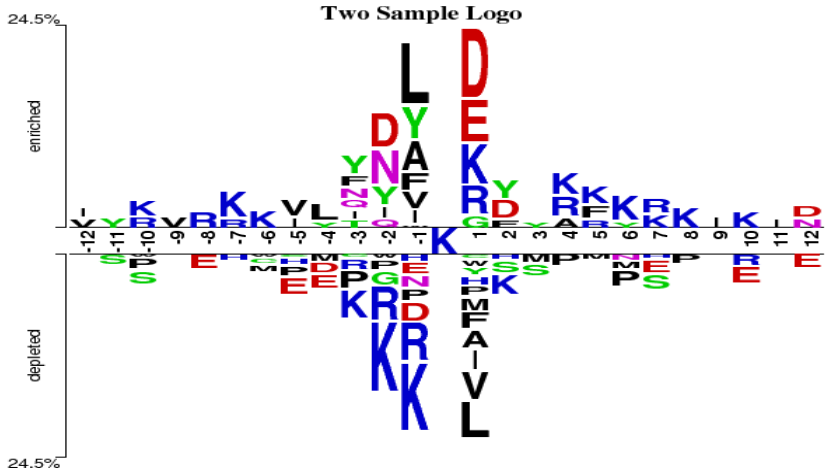

Fig. (3). The occurrences of amino acid propensities of surrounding positive windows (succinylation site) and negative windows (non-succinylation sites) of size 25 are presented by the Two-Sample Logos software (Vacie *et al.*, 2006) [29]. *(A higher resolution/colour version of this figure is available in the electronic copy of the article).*

3.2. Assessment of Prediction Performance with the Training Dataset

To investigate the prediction performance of the proposed model in comparison of the other candidate models, all prediction models were trained by the training dataset corresponding to the ratio 1:2 of 1195 positive and 2390 negative windows as discussed in section 2.2. Then we computed prediction performance scores SN, SP, ACC, MCC and AUC as well as the alternative prediction performance scores FPR, FNR and MCR to investigate the training performance of the proposed fusion prediction model RF(Binary, CKSAAP, AAC) in a comparison of the other 26 candidate models ADA(Binary), ADA(CKSAAP), ADA(AAC), ADA(CKSAAP, Binary), ADA(CKSAAP, AAC), ADA(Binary, AAC), ADA(CKSAAP, Binary, AAC), SVM (Binary), SVM(CKSAAP), SVM(AAC), SVM (CKSAAP, Binary), SVM(CKSAAP, AAC), SVM(Binary, AAC), SVM(CKSAAP, Binary, AAC), RF(Binary), RF(CKSAAP), RF(AAC), RF(CKSAAP, Binary), RF(CKSAAP, AAC) and RF(Binary, AAC). Table **1** shows the summary of different training performance scores (SN, SP, FNR, ACC, MCR, MCC, ROC, AUC and pAUC) with all prediction models. At first, we observed that binary encoding produced slightly better results with all three classifiers (ADA, SVM and RF) than the CKSAAP and AAC encodings. Then we observed that the fusion model RF(CKSAAP, Binary, AAC) produced the highest prediction performance scores of TPR (0.869), TNR (0.902), ACC (0.937), MCC (0.781), pAUC (0.198) at FPR=0.10 and AUC (0.959), and the smallest alternative prediction performance score of FNR (0.131) and MCR (0.100) at FPR=0.10 against any of the other prediction models. Similar performance trend was also observed at FPR=0.20 (See Table **S1** in the Supplementary file). Thus, the proposed RF-based fusion model shows better prediction performance than the other candidate prediction models with the training dataset.

3.3. Assessment of Prediction Performance based on 20-fold Cross-validation (CV)

To investigate the test performance of the proposed RF-based fusion prediction model in a comparison of the other candidate prediction models by 20-fold CV, the training dataset corresponding to 1:2 ratio of 1195 positive and 2390 negative window samples was partitioned into 20 mutually exclusive subgroups (G1, G2, G3,…, G20) of almost equal sizes such that each subgroup contains 1:2 ratio of positive and negative samples. Obviously, each subgroup consisted of around 59 (5% of 1195) positive window samples and around 118 (95% of 2390) negative window samples. The 20-fold CV was completed within 20 steps. At step 1, nineteen (19) groups (G2, G3, G4,…, G20), those contained around 95% window samples satisfying 1:2 ratio of positive and negative window samples, were used to train all prediction models ADA(Binary), ADA(CKSAAP), ADA(AAC), ADA(CKSAAP, Binary), ADA(CKSAAP, AAC), ADA(Binary, AAC), ADA(CKSAAP, Binary, AAC), SVM(Binary), SVM(CKSAAP), SVM(AAC), SVM(CKSAAP, Binary), SVM(CKSAAP, AAC), SVM(Binary, AAC), SVM(CKSAAP, Binary, AAC), RF(Binary), RF(CKSAAP), RF(AAC), RF(CKSAAP, Binary), RF(CKSAAP, AAC), RF(Binary, AAC), and proposed RF(CKSAAP, Binary, AAC). The other group G1 was used to evaluate the trained models by using different performance measures (SN, SP, FNR, ACC, MCR, MCC, ROC, AUC and pAUC). At step 2, nineteen (19) groups (G1, G3, G4,…, G20), those also contained around 95% window samples were used to train all the prediction models as before. The other group G2 was used to evaluate the trained models by using different performance measures (SN, SP, FNR, ACC, MCR, MCC, ROC, AUC and pAUC) as before. In this step, G1-G2 subgroups interchanged their positions between the training and test sets only. Similarly, at other eighteen (18) steps/loops, subgroup pairs G2-G3, G3-G4, G4-G5, …, G19-G20 interchanged their positions between the training and test sets, respectively. Then we calculated the average performance scores of SN, SP, FNR, ACC, MCR, MCC, ROC, AUC and pAUC. Table **2** shows the summary of average performance scores of different candidate predictors based on 20-fold CV with a 1:2 ratio of positive and negative window samples in the training dataset with a cutoff point at FPR=0.10. We also calculated the standard error (SE) for each performance measure to observe the stability of the

Table 1. Summary performance at FPR=0.10 for different candidate predictors based on the training dataset corresponding to 1:2 ratio of positive and negative windows.

Predictors Classifier(Encodings)	TPR (SN)	TNR (SP)	FNR	ACC	MCC	MCR	AUC	pAUC
ADA(Binary)	0.752	0.901	0.248	0.867	0.652	0.187	0.881	0.131
ADA(CKSAAP)	0.747	0.900	0.253	0.853	0.637	0.221	0.865	0.110
ADA(AAC)	0.750	0.902	0.250	0.862	0.643	0.198	0.876	0.115
ADA(CKSAAP, Binary)	0.763	0.901	0.237	0.871	0.687	0.178	0.902	0.142
ADA(CKSAAP, AAC)	0.757	0.901	0.243	0.868	0.656	0.189	0.887	0.133
ADA(Binary, AAC)	0.761	0.900	0.239	0.869	0.658	0.186	0.899	0.139
ADA(CKSAAP,Binary,AAC)	0.769	0.901	0.231	0.873	0.690	0.171	0.908	0.144
SVM(Binary)	0.643	0.902	0.357	0.756	0.543	0.266	0.822	0.069
SVM(CKSAAP)	0.632	0.903	0.368	0.734	0.532	0.286	0.812	0.073
SVM(AAC)	0.638	0.901	0.362	0.737	0.541	0.268	0.820	0.079
SVM(CKSAAP, Binary)	0.665	0.901	0.335	0.777	0.573	0.250	0.841	0.069
SVM(CKSAAP, AAC)	0.668	0.902	0.332	0.779	0.575	0.243	0.848	0.071
SVM(Binary, AAC)	0.675	0.901	0.325	0.781	0.578	0.241	0.850	0.072
SVM(CKSAAP,Binary,AAC)	0.678	0.902	0.322	0.788	0.581	0.234	0.852	0.076
RF(Binary)	0.833	0.902	0.167	0.912	0.858	0.121	0.940	0.195
RF(CKSAAP)	0.801	0.903	0.199	0.903	0.703	0.134	0.904	0.181
RF(AAC)	0.761	0.902	0.239	0.905	0.627	0.159	0.913	0.129
RF(CKSAAP, Binary)	0.854	0.900	0.146	0.930	0.776	0.118	0.940	0.156
RF(CKSAAP, AAC)	0.857	0.901	0.143	0.932	0.778	0.113	0.942	0.157
RF(Binary, AAC)	0.859	0.901	0.141	0.934	0.779	0.110	0.947	0.161
RF(CKSAAP,Binary,AAC)	**0.869**	**0.902**	**0.131**	**0.937**	**0.781**	**0.100**	**0.965**	**0.198**

estimates. The values within the first bracket in Table **S2.1** indicate the SE. A prediction model with the smallest SE of performance scores is known as a more stable predictor. We observed that binary encoding produces slightly better results with all three classifiers (ADA, SVM and RF) than the AAC and CKSAAP encoding as before. Then we observed that the fusion model RF(CKSAAP, Binary, AAC) produces the highest average prediction performance scores of TPR (0.800), TNR (0.902), ACC (0.919), MCC (0.766), AUC (0.958) and pAUC (0.163) at FPR=0.10, and the smallest average of alternative prediction performance scores of MCR (0.141) in a comparison of the other prediction models. Similar performance trend was also observed at FPR=0.20 (See Table **S2.2** in the supplementary file). The ROC curves given in the Supplementary Figs. **(S5-S7)** also supported these results. The SE values for each performance are measured corresponding to the RF(CKSAAP, Binary, AAC) prediction model as much smaller compared to any of the other twenty (20) prediction models. So RF (CKSAAP, Binary, AAC) prediction model would be more efficient than the other prediction model. Thus, the RF-based fusion model shows better prediction performance compared to other candidate prediction models by cross-validation.

3.4. Assessment of Prediction Performance with the Independent Test Datasets

The prediction performance of the proposed model was compared with the other candidate prediction models by using two independent test datasets as discussed below:

3.4.1. Performance with the Independent Test Dataset-1

All candidate prediction models including the proposed RF(CKSAAP, Binary, AAC) were trained by the training dataset corresponding to a 1:2 ratio of 1195 positive and 2390 negative windows as discussed previously in section 3.2. We considered independent test dataset-1 that was introduced in section 2.2 to demonstrate the performance of the proposed RF(CKSAAP, Binary, AAC) prediction model in a comparison of the other candidate twenty (20) prediction models ((ADA(Binary), ADA(CKSAAP), ADA(AAC), ADA(CKSAAP, Binary), ADA(CKSAAP, AAC), ADA(Binary, AAC), ADA(CKSAAP, Binary, AAC), SVM

Table 2. Summary of average performance at FPR=0.10 for different candidate predictors based on 20-fold CV with 1:2 ratio of positive and negative window samples in the training dataset.

Predictors Classifier(Encodings)	TPR	TNR	FNR	ACC	MCC	MCR	AUC	pAUC
ADA(Binary)	0.378	0.902	0.621	0.593	0.205	0.407	0.734	0.048
ADA(CKSAAP)	0.364	0.903	0.635	0.589	0.199	0.410	0.726	0.050
ADA(AAC)	0.344	0.901	0.655	0.607	0.251	0.392	0.702	0.030
ADA(CKSAAP, Binary)	0.557	0.899	0.443	0.683	0.376	0.318	0.783	0.069
ADA(CKSAAP, AAC)	0.478	0.901	0.521	0.670	0.368	0.329	0.774	0.059
ADA(Binary, AAC)	0.559	0.902	0.441	0.685	0.377	0.386	0.788	0.078
ADA(CKSAAP,Binary,AAC)	0.612	0.901	0.487	0.721	0.456	0.302	0.826	0.110
SVM(Binary)	0.456	0.900	0.515	0.650	0.317	0.350	0.745	0.058
SVM(CKSAAP)	0.343	0.901	0.657	0.578	0.178	0.421	0.719	0.044
SVM(AAC)	0.281	0.901	0.716	0.574	0.182	0.425	0.708	0.034
SVM(CKSAAP, Binary)	0.557	0.903	0.442	0.685	0.384	0.314	0.779	0.068
SVM(CKSAAP, AAC)	0.543	0.902	0.456	0.667	0.376	0.356	0.766	0.054
SVM(Binary, AAC)	0.567	0.901	0.432	0.684	0.382	0.324	0.771	0.069
SVM(CKSAAP,Binary,AAC)	0.598	0.902	0.401	0.700	0.422	0.312	0.814	0.100
RF(Binary)	0.725	0.902	0.275	0.871	0.688	0.175	0.895	0.132
RF(CKSAAP)	0.691	0.901	0.308	0.786	0.584	0.183	0.869	0.123
RF(AAC)	0.681	0.899	0.319	0.859	0.672	0.192	0.877	0.124
RF(CKSAAP, Binary)	0.789	0.902	0.211	0.917	0.761	0.153	0.948	0.153
RF(CKSAAP, AAC)	0.711	0.899	0.249	0.869	0.682	0.172	0.891	0.134
RF(Binary, AAC)	0.725	0.902	0.275	0.871	0.688	0.175	0.895	0.132
RF(CKSAAP,Binary,AAC)	0.800	0.902	0.200	0.919	0.766	0.141	0.958	0.163

(Binary), SVM(CKSAAP), SVM(AAC), SVM(CKSAAP, Binary), SVM(CKSAAP, AAC), SVM(Binary, AAC), SVM(CKSAAP, Binary, AAC), RF(Binary), RF(CKSAAP), RF(AAC), RF(CKSAAP, Binary), RF(CKSAAP, AAC), RF(Binary, AAC)) along with two existing predictors [14, 18].

The independent test dataset-1 was consisted based on 50 proteins with 54 positive windows and 2004 negative windows as mentioned in section 2.2. It was used to evaluate the trained models by using different performance measures (SN, SP, FNR, ACC, MCR, MCC, ROC, AUC and pAUC) as before. Table **3** shows different performance scores (SN, SP, FNR, ACC, MCR, MCC, ROC, AUC and pAUC) of different candidate prediction models with the independent test dataset-1. It is clearly observed that the proposed RF(CKSAAP, Binary, AAC) prediction model gives the highest correct prediction performance scores of TPR (0.798), TNR (0.902), ACC (0.891), MCC (0.629), AUC (0.921) and pAUC (0.139) at FPR=0.10 and the smallest incorrect prediction performance scores of FNR (0.201) and MCR (0.145) against the other prediction models. Fig. (**4**) also supports the results of Table **3**. A similar performance trend was also observed at FPR=0.20 (Table **S3** in the Supplementary file). Thus, the proposed RF(CKSAAP, Binary, AAC) prediction model shows better performance than the other candidate prediction models with the independent test dataset-1 also. A comparative discussion between the proposed prediction model and two existing models has been given in section 4.

3.4.2. Performance with the Independent Test Dataset-2

To investigate the performance of the proposed RF(CKSAAP, Binary, AAC) prediction model in a comparison of the others more fairly, we considered the independent test dataset-2 that was introduced in section 2.2. For the convenience of presentation, again it should be mentioned here that the training dataset and the independent test dataset-1 were generated from the same data source, while the independent test dataset-2 was collected from the other independent data source. This independent test dataset-2 was based on 202 lysine succinylated proteins, which contained 463 positive and 6742 negative window sites.

Table 3. Summary performance at FPR=0.10 for different candidate predictors based on independent test dataset-1.

Predictors Classifier(Encodings)	TPR (SN)	TNR (SP)	FNR	ACC	MCC	MCR	AUC	pAUC
ADA(Binary)	0.645	0.901	0.345	0.772	0.563	0.227	0.841	0.101
ADA(CKSAAP)	0.637	0.902	0.363	0.769	0.558	0.231	0.812	0.100
ADA(AAC)	0.698	0.902	0.301	0.761	0.526	0.238	0.820	0.102
ADA(CKSAAP,Binary)	0.703	0.902	0.297	0.811	0.666	0.208	0.866	0.122
ADA(CKSSAP,AAC)	0.701	0.901	0.298	0.812	0.667	0.209	0.856	0.121
ADA(Binary,AAC)	0.732	0.902	0.267	0.791	0.587	0.208	0.881	0.123
ADA(CKSSAP,Binary,AAC)	0.739	0.901	0.260	0.799	0.602	0.200	0.889	0.138
SVM(Binary)	0.393	0.901	0.607	0.646	0.339	0.353	0.752	0.069
SVM(CKSAAP)	0.388	0.901	0.611	0.644	0.335	0.355	0.742	0.073
SVM(AAC)	0.389	0.901	0.610	0.645	0.336	0.353	0.747	0.075
SVM(CKSAAP,Binary)	0.425	0.901	0.575	0.669	0.382	0.310	0.762	0.069
SVM(CKSSAP,AAC)	0.389	0.901	0.610	0.645	0.336	0.353	0.748	0.075
SVM(Binary,AAC)	0.479	0.902	0.520	0.677	0.387	0.322	0.793	0.098
SVM(CKSSAP,Binary,AAC)	0.482	0.901	0.518	0.679	0.397	0.311	0.796	0.103
RF(Binary)	0.742	0.902	0.267	0.851	0.603	0.189	0.898	0.125
RF(CKSAAP)	0.701	0.901	0.299	0.810	0.593	0.220	0.864	0.121
RF(AAC)	0.732	0.902	0.267	0.771	0.544	0.228	0.868	0.124
RF(CKSAAP,Binary)	0.761	0.902	0.239	0.860	0.627	0.159	0.913	0.129
RF(CKSSAP,AAC)	0.742	0.902	0.257	0.781	0.554	0.208	0.878	0.128
RF(Binary,AAC)	0.761	0.902	0.239	0.860	0.627	0.159	0.913	0.129
RF(CKSSAP,Binary,AAC)	0.798	0.902	0.201	0.891	0.629	0.145	0.921	0.139

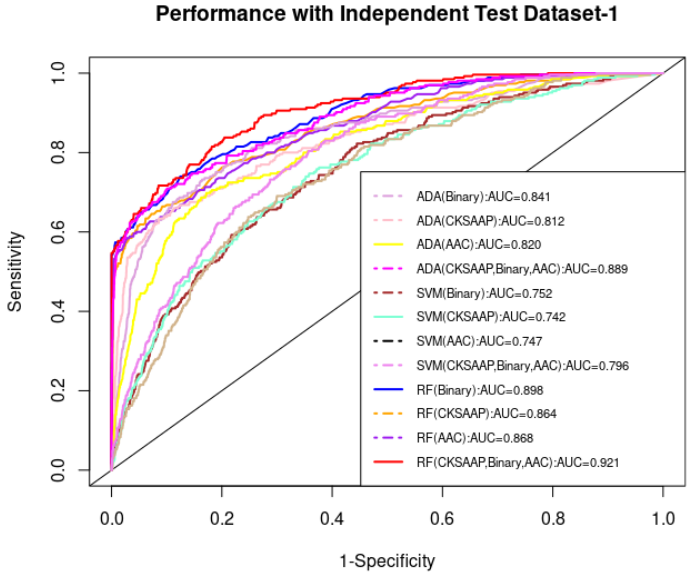

Fig. (4). ROC curve to display the performance of different candidate predictors based on independent test dataset-1. *(A higher resolution/colour version of this figure is available in the electronic copy of the article).*

Table 4. Summary performance at FPR=0.10 for different candidate predictors based on independent test dataset-2.

Predictors Classifier(Encoding)	TPR (SN)	TNR (SP)	FNR	ACC	MCC	MCR	AUC	pAUC
ADA(Binary)	0.627	0.901	0.373	0.763	0.547	0.236	0.862	0.110
ADA(CKSAAP)	0.626	0.902	0.374	0.763	0.547	0.236	0.793	0.093
ADA(AAC)	0.702	0.901	0.297	0.756	0.515	0.243	0.816	0.100
ADA(CKSAAP,Binary)	0.643	0.903	0.357	0.812	0.602	0.189	0.874	0.113
ADA(CKSSAP,AAC)	0.631	0.901	0.368	0.783	0.592	0.217	0.846	0.109
ADA(Binary,AAC)	0.698	0.902	0.301	0.815	0.654	0.182	0.893	0.129
ADA(CKSSAP,Binary,AAC)	0.700	0.902	0.299	0.816	0.656	0.181	0.895	0.130
SVM(Binary)	0.604	0.902	0.396	0.704	0.558	0.302	0.775	0.077
SVM(CKSAAP)	0.453	0.903	0.652	0.613	0.324	0.543	O.689	0.05
SVM(AAC)	0.581	0.901	0.418	0.672	0.351	0.327	0.740	0.069
SVM(CKSAAP,Binary)	0.604	0.901	0.396	0.704	0.558	0.302	0.776	0.077
SVM(CKSSAP,AAC)	0.581	0.901	0.418	0.672	0.351	0.327	0.740	0.069
SVM(Binary,AAC)	0.667	0.902	0.332	0.727	0.458	0.272	0.807	0.089
SVM(CKSSAP,Binary,AAC)	0.678	0.901	0.321	0.731	0.466	0.268	0.810	0.099
RF(Binary)	0.724	0.902	0.276	0.845	0.680	0.204	0.902	0.125
RF(CKSAAP)	0.693	0.902	0.307	0.804	0.666	0.225	0.847	0.114
RF(AAC)	0.615	0.901	0.384	0.801	0.651	0.198	0.877	0.121
RF(CKSAAP,Binary)	0.745	0.901	0.255	0.881	0.601	0.148	0.910	0.138
RF(CKSSAP,AAC)	0.702	0.902	0.298	0.797	0.608	0.202	0.878	0.122
RF(Binary,AAC)	0.749	0.902	0.250	0.884	0.669	0.169	0.920	0.140
RF(CKSSAP,Binary,AAC)	**0.772**	**0.901**	**0.227**	**0.886**	**0.677**	**0.163**	**0.923**	**0.141**

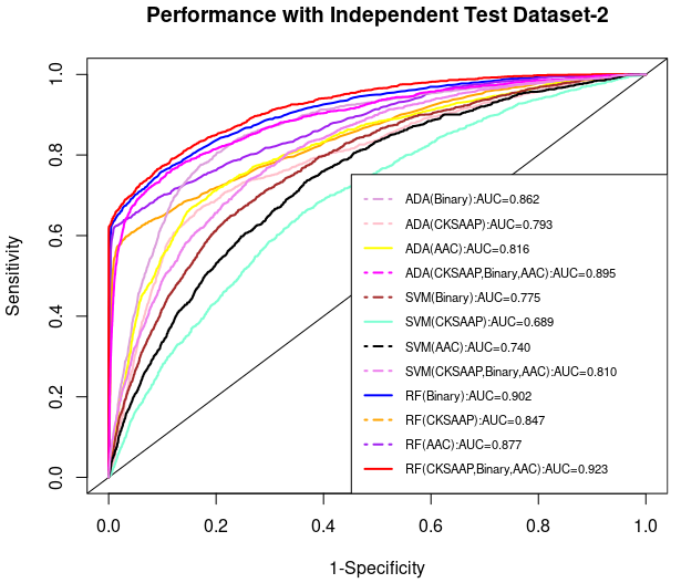

Fig. (5). ROC curve to display the performance of different candidate predictors based on independent test dataset-2. *(A higher resolution/colour version of this figure is available in the electronic copy of the article).*

Table 5. Performance comparison of the proposed predictor with other existing Predictors.

Prediction methods	Independent Test Dataset	TPR (SN)	TNR (SP)	ACC	MCC	AUC
Succinsite2.0	Dataset-1	0.632	0.872	0.866	0.241	0.845
GPSuc	Dataset-1	0.693	0.877	0.872	0.279	0.885
HybridSucc	Dataset-1	0.822	0.855	0.859	0.562	0.891
CNN-SuccSite	Dataset-1	0.716	0.844	0.842	0.443	0.839
Proposed Predictor	Dataset-1	0.798	0.902	0.891	0.629	0.921
	Dataset-2	0.772	0.901	0.886	0.677	0.923

Table **4** shows the summary of different prediction performance scores (SN, SP, FNR, ACC, MCR, MCC, ROC, AUC and pAUC) of different candidate predictors with the independent test-2 dataset. It is clearly observed that proposed RF(CKSAAP, Binary, AAC) prediction model gives the highest correct prediction performance scores of TPR (0.772), TNR (0.901), ACC (0.886), MCC (0.677), AUC (0.923) and pAUC (0.141) at FPR=0.10 and the smallest incorrect prediction performance score of FNR (0.227) and MCR (0.163) against to other twenty prediction models (ADA (Binary), ADA(CKSAAP), ADA(AAC), ADA (CKSAAP, Binary), ADA(CKSAAP, AAC), ADA(Binary, AAC), ADA(CKSAAP, Binary, AAC), SVM(Binary), SVM (CKSAAP), SVM(AAC), SVM(CKSAAP, Binary), SVM (CKSAAP, AAC), SVM (Binary, AAC), SVM (CKSAAP, Binary, AAC), RF (Binary), RF (CKSAAP), RF (AAC), RF (CKSAAP, Binary), RF (CKSAAP, AAC), RF (Binary, AAC)) as like as 20-fold CV results. Fig. (**5**) also supports the results of Table **4**. A similar performance trend was also observed at FPR=0.20 (Table **S4** in the Supplementary file). Thus, the proposed RF(CKSAAP, Binary, AAC) prediction model shows better performance than the other candidate prediction models also with the independent test dataset-2.

4. DISCUSSION

We developed an improved computational prediction model by maximizing the prediction performance scores (SN, SP, ACC, MCC & AUC) and conversely minimizing the alternative prediction performance scores (FPR, FNR & MCR) with respect to different model parameters and some tuning parameters like a cut-off value of CD-HIT at 30%, window size at 25, a ratio of positive and negative windows at 1:2 based on the combined model of three encoding schemes (binary, CKSAAP and AAC) with the random forest (RF) classifier to predict lysine succinylation sites mapping on *Homo sapiens*. We observed that all candidate prediction models show much better performance with both 1:2 and 1:3 ratios of positive and negative window samples than 1:1 ratio. We also observed that their performance with 1:2 and 1:3 ratios of positive and negative window samples was not so different significantly. Therefore, we considered the ratio 1:2 of positive and negative window samples to develop the prediction model by reducing the computational cost. The performance of the proposed predictor in a comparison of the other candidate predictors was investigated by using the prediction performance scores with the training dataset and independent test performance scores based on 20-fold cross-validation and two independent test datasets that were collected from two different sources. It should be mentioned here that the training dataset and the independent test dataset-1 were collected from the same data source, while the independent test dataset-2 was collected from the other data source.

We observed that binary encoding produces slightly better performance scores with all three classifiers (ADA, SVM and RF) than the CKSAAP and AAC encoding in all cases (Tables **1**-**4**, **S1**-**S4**, Figs. **4** and **5**). So, we imposed more weight on the binary encoding than the CKSAAP and AAC encoding during the construction of the fusion model. Then we observed from Table **1** and **S1** as discussed in the previous section 3.2 that the proposed RF(CKSAAP, Binary, AAC) model shows much better prediction performance compared to other candidate prediction models (ADA(Binary), ADA(CKSAAP), ADA(AAC), ADA(CKSAAP, Binary), ADA(CKSAAP, AAC), ADA(Binary, AAC), ADA(CKSAAP, Binary, AAC), SVM(Binary), SVM(CKSAAP), SVM(AAC), SVM(CKSAAP, Binary), SVM(CKSAAP, AAC), SVM(Binary, AAC), SVM(CKSAAP, Binary, AAC), RF(Binary), RF(CKSAAP), RF(AAC), RF(CKSAAP, Binary), RF(CKSAAP, AAC), RF(Binary, AAC)). Similarly, Tables **2**, **S2.1** and **S2.2** as discussed in the previous section 3.3 show that the proposed RF-based fusion prediction model outperforms the other candidate prediction models in the case of 20-fold CV. Tables **3**, **4**, **S3**-**S4** and Figs. (**4** and **5**) showed that the proposed RF(CKSAAP, Binary, AAC) fusion prediction model outperforms the other candidate prediction models also with both independent test datasets-1&2.

Furthermore, we considered the test dataset-1 to investigate the performance of the proposed RF(CKSAAP, Binary, AAC) model with four existing species-wise prediction models known as SuccinSite2.0 [14], CNN-SuccSite [16]), HybridSucc [17] and GPSuc [18] for the prediction of lysine succinylation sites using the same independent test dataset as mentioned previously in section 2.2. We considered five important performance measures (SN, SP, ACC, MCC and AUC) to compare the proposed method with these four existing methods. Table **5** shows 5 performance scores with the same independent dataset-1 for the mentioned four existing methods and the proposed method. We observed that the proposed method greatly improves the performance over the

existing four methods with respect to all 5 measures. Especially, MCC values increased and the AUC value was 4-8% higher than the existing prediction models. Thus, our proposed prediction model outperformed the existing prediction models. Noted that, until now, only four predictors are publicly available. All of the other existing predictors are not publicly available. Therefore, to make a fair comparison, we compared these four existing prediction models. In addition, to examine our prediction model robustly, we considered the independent test dataset-2 which was already introduced in section 2.2. It consisted of 202 human pathogen *Histoplasma capsulatum* succinylated proteins which contained 463 modification sites. We found from Tables **4** and **5**, **S4** and Fig. (**5**) that our prediction model can be applied for *Histoplasma capsulatum* succinylation site prediction.

CONCLUSION

We developed an improved predictor based on the sequence information to predict lysine succinylation sites mapping on *Homo sapiens* by fusing three encoding schemes (binary, CKSAAP and AAC) with the random forest machine learning framework. We performed a comparative study on the prediction of succinylation sites by using the empirically developed succinylated protein sequences of *Homo sapiens*. The investigational results by 20-fold CV and two independent test-sets show that the proposed method can identify succinylated sites more accurately than the other candidate prediction models. Moreover, the benchmarking experiments demonstrated that our proposed predictor gives a competitive performance compared to the existing four predictors. The proposed method may be a useful and encouraging computational resource for lysine succinylation site prediction in the case of Human PTMs. To implement the proposed method, the computational codes and necessary instructions can be downloaded from http://www.ru.ac.bd/biorgru/software/succinsitefusing-zip/. To further improve the prediction performances, we may use newly proposed encoding schemes and integrated approaches [26, 50-54].

ETHICS APPROVAL AND CONSENT TO PARTICIPATE

Not applicable.

HUMAN AND ANIMAL RIGHTS

No animals/humans were used for studies that are the basis of this research.

CONSENT FOR PUBLICATION

Not applicable.

AVAILABILITY OF DATA AND MATERIALS

The data of this study and the source code (R & Perl) are openly accessible at http://www.ru.ac.bd/biorgru/software/succinsitefusing-zip/.

FUNDING

None.

CONFLICT OF INTEREST

The authors declare no conflict of interest, financial or otherwise.

ACKNOWLEDGEMENTS

Declared none.

SUPPLEMENTARY MATERIAL

Supplementary material is available on the publisher's website along with the published article.

REFERENCES

[1] Weinert, B.T.; Schölz, C.; Wagner, S.A.; Iesmantavicius, V.; Su, D.; Daniel, J.A.; Choudhary, C. Lysine succinylation is a frequently occurring modification in prokaryotes and eukaryotes and extensively overlaps with acetylation. *Cell Rep.*, **2013**, *4*(4), 842-851. http://dx.doi.org/10.1016/j.celrep.2013.07.024 PMID: 23954790

[2] Xie, Z.; Dai, J.; Dai, L.; Tan, M.; Cheng, Z.; Wu, Y.; Boeke, J.D.; Zhao, Y. Lysine succinylation and lysine malonylation in histones. *Mol. Cell. Proteomics*, **2012**, *11*(5), 100-107. http://dx.doi.org/10.1074/mcp.M111.015875 PMID: 22389435

[3] Tan, M.; Peng, C.; Anderson, K.A.; Chhoy, P.; Xie, Z.; Dai, L.; Park, J.; Chen, Y.; Huang, H.; Zhang, Y.; Ro, J.; Wagner, G.R.; Green, M.F.; Madsen, A.S.; Schmiesing, J.; Peterson, B.S.; Xu, G.; Ilkayeva, O.R.; Muehlbauer, M.J.; Braulke, T.; Mühlhausen, C.; Backos, D.S.; Olsen, C.A.; McGuire, P.J.; Pletcher, S.D.; Lombard, D.B.; Hirschey, M.D.; Zhao, Y. Lysine glutarylation is a protein posttranslational modification regulated by SIRT5. *Cell Metab.*, **2014**, *19*(4), 605-617. http://dx.doi.org/10.1016/j.cmet.2014.03.014 PMID: 24703693

[4] Zhang, Z.; Tan, M.; Xie, Z.; Dai, L.; Chen, Y.; Zhao, Y. Identification of lysine succinylation as a new post-translational modification. *Nat. Chem. Biol.*, **2011**, *7*(1), 58-63. http://dx.doi.org/10.1038/nchembio.495 PMID: 21151122

[5] Rosen, R.; Becher, D.; Büttner, K.; Biran, D.; Hecker, M.; Ron, E.Z. Probing the active site of homoserine trans-succinylase. *FEBS Lett.*, **2004**, *577*(3), 386-392. http://dx.doi.org/10.1016/j.febslet.2004.10.037 PMID: 15556615

[6] Machida, Y.; Chiba, T.; Takayanagi, A.; Tanaka, Y.; Asanuma, M.; Ogawa, N.; Koyama, A.; Iwatsubo, T.; Ito, S.; Jansen, P.H.; Shimizu, N.; Tanaka, K.; Mizuno, Y.; Hattori, N. Common anti-apoptotic roles of parkin and α-synuclein in human dopaminergic cells [J]. *Biochem. Biophys. Res. Commun.*, **2005**, *332*(1), 233-240. http://dx.doi.org/10.1016/j.bbrc.2005.04.124 PMID: 15896322

[7] Lind, C.; Gerdes, R.; Hamnell, Y.; Schuppe-Koistinen, I.; von Löwenhielm, H.B.; Holmgren, A.; Cotgreave, I.A. Identification of S-glutathionylated cellular proteins during oxidative stress and constitutive metabolism by affinity purification and proteomic analysis [J]. *Arch. Biochem. Biophys.*, **2002**, *406*(2), 229-240. http://dx.doi.org/10.1016/S0003-9861(02)00468-X PMID: 12361711

[8] Park, J.; Chen, Y.; Tishkoff, D.X.; Peng, C.; Tan, M.; Dai, L.; Xie, Z.; Zhang, Y.; Zwaans, B.M.; Skinner, M.E.; Lombard, D.B.; Zhao, Y. SIRT5-mediated lysine desuccinylation impacts diverse metabolic pathways. *Mol. Cell*, **2013**, *50*(6), 919-930. http://dx.doi.org/10.1016/j.molcel.2013.06.001 PMID: 23806337

[9] Colak, G.; Xie, Z.; Zhu, A.Y.; Dai, L.; Lu, Z.; Zhang, Y.; Wan, X.; Chen, Y.; Cha, Y.H.; Lin, H.; Zhao, Y.; Tan, M. Identification of lysine succinylation substrates and the succinylation regulatory enzyme CobB in *Escherichia coli*. *Mol. Cell. Proteomics*, **2013**, *12*(12), 3509-3520. http://dx.doi.org/10.1074/mcp.M113.031567 PMID: 24176774

[10] Li, X.; Hu, X.; Wan, Y.; Xie, G.; Li, X.; Chen, D.; Cheng, Z.; Yi, X.; Liang, S.; Tan, F. Systematic identification of the lysine succinylation in the protozoan parasite *Toxoplasma gondii*. *J. Proteome Res.*, **2014**, *13*(12), 6087-6095. http://dx.doi.org/10.1021/pr500992r PMID: 25377623

[11] Yang, M.; Wang, Y.; Chen, Y.; Cheng, Z.; Gu, J.; Deng, J.; Bi, L.; Chen, C.; Mo, R.; Wang, X.; Ge, F. Succinylome analysis reveals the involvement of lysine succinylation in metabolism in pathogen-

ic *Mycobacterium tuberculosis. Mol. Cell. Proteomics*, **2015**, *14*(4), 796-811.
http://dx.doi.org/10.1074/mcp.M114.045922 PMID: 25605462

[12] Jin, W.; Wu, F. Proteome-wide identification of lysine succinylation in the proteins of tomato (*Solanum lycopersicum*). *PLoS One*, **2016**, *11*(2), e0147586.
http://dx.doi.org/10.1371/journal.pone.0147586 PMID: 26828863

[13] Xie, L.; Li, J.; Deng, W.; Yu, Z.; Fang, W.; Chen, M.; Liao, W.; Xie, J.; Pan, W. Proteomic analysis of lysine succinylation of the human pathogen *Histoplasma capsulatum. J. Proteomics*, **2017**, *154*, 109-117.
http://dx.doi.org/10.1016/j.jprot.2016.12.020 PMID: 28063982

[14] Hasan, M.M.; Khatun, M.S.; Mollah, M.N.H.; Yong, C.; Guo, D. A systematic identification of species-specific protein succinylation sites using joint element features information. *Int. J. Nanomedicine*, **2017**, *12*, 6303-6315.
http://dx.doi.org/10.2147/IJN.S140875 PMID: 28894368

[15] Hasan, M.M.; Yang, S.; Zhou, Y.; Mollah, M.N. SuccinSite: a computational tool for the prediction of protein succinylation sites by exploiting the amino acid patterns and properties. *Mol. Biosyst.*, **2016**, *12*(3), 786-795.
http://dx.doi.org/10.1039/C5MB00853K PMID: 26739209

[16] Huang, K.Y.; Hsu, J.B.; Lee, T.Y. Characterization and identification of lysine succinylation sites based on deep learning method. *Sci. Rep.*, **2019**, *9*(1), 16175.
http://dx.doi.org/10.1038/s41598-019-52552-4 PMID: 31700141

[17] Ning, W.; Xu, H.; Jiang, P.; Cheng, H.; Deng, W.; Guo, Y.; Xue, Y. HybridSucc: A hybrid-learning architecture for general and species-specific succinylation site prediction. *Genomics Proteomics Bioinformatics*, **2020**, *18*(2), 194-207.
http://dx.doi.org/10.1016/j.gpb.2019.11.010 PMID: 32861878

[18] Hasan, M.M.; Kurata, H. GPSuc: Global prediction of generic and species-specific succinylation sites by aggregating multiple sequence features. *PLoS One*, **2018**, *13*(10), e0200283.
http://dx.doi.org/10.1371/journal.pone.0200283 PMID: 30312302

[19] Shoombuatong, W.; Hongjaisee, S.; Barin, F.; Chaijaruwanich, J.; Samleerat, T. HIV-1 CRF01_AE coreceptor usage prediction using kernel methods based logistic model trees. *Comput. Biol. Med.*, **2012**, *42*(9), 885-889.
http://dx.doi.org/10.1016/j.compbiomed.2012.06.011 PMID: 22824642

[20] Rashid, M.M.; Shatabda, S.; Hasan, M.M.; Kurata, H. Recent development of machine learning methods in microbial phosphorylation sites. *Curr. Genomics*, **2020**, *21*(3), 194-203.
http://dx.doi.org/10.2174/1389202921666200427210833 PMID: 33071613

[21] Manavalan, B.; Govindaraj, R.G.; Shin, T.H.; Kim, M.O.; Lee, G. iBCE-EL: A new ensemble learning framework for improved linear B-cell epitope prediction. *Front. Immunol.*, **2018**, *9*, 1695.
http://dx.doi.org/10.3389/fimmu.2018.01695 PMID: 30100904

[22] Huang, Y.; Niu, B.; Gao, Y.; Fu, L.; Li, W. CD-HIT Suite: a web server for clustering and comparing biological sequences. *Bioinformatics*, **2010**, *26*(5), 680-682.
http://dx.doi.org/10.1093/bioinformatics/btq003 PMID: 20053844

[23] Eva, O.; Oskar, O.; Jozef, K. Methodology and Application of the Kruskal-Wallis Test. *Appl. Mech. Mater.*, **2014**, *611* Available at: www.scientific.net/AMM.611.11

[24] Rahman, M.M.; Mollah, M.N.H. Robustification of gaussian bayes Classifier by the minimum β-divergence method. *J. Classif.*, **2019**, *36*, 113-139.
http://dx.doi.org/10.1007/s00357-019-9306-1

[25] *Boosting Algorithms: AdaBoost, Gradient Boosting and XGBoost*, **2018**. Available at: hackernoon.com, May 5, 2018. Retrieved 2020-01-04.

[26] Cortes, C.; Vapnik, V.N. Support-vector networks. *Mach. Learn.*, **1995**, *20*(3), 273-297.
http://dx.doi.org/10.1007/BF00994018

[27] Breiman, L. Random forests. *Mach. Learn.*, **2001**, *45*, 5-32.
http://dx.doi.org/10.1023/A:1010933404324

[28] Chen, Z.; Chen, Y.-Z.; Wang, X.-F.; Wang, C.; Yan, R.-X.; Zhang, Z. Prediction of ubiquitination sites by using the composition of k-spaced amino acid pairs. *PLoS One*, **2011**, *6*(7), e22930.
http://dx.doi.org/10.1371/journal.pone.0022930 PMID: 21829559

[29] Hasan, M.M.; Zhou, Y.; Lu, X.; Li, Z.; Song, J.; Zhang, Z. Computational Identification of protein pupylation sites by using profile-based composition of k-spaced amino acid pairs. *PLoS One*, **2015**, e0129635.
http://dx.doi.org/10.1371/journal.pone.0129635

[30] Hasan, M.M.; Schaduangrat, N.; Lee, G.; Shoombuatong, W.; Manavalan, B. HLPpred-Fuse: Improved and robust prediction of hemolytic peptide and its activity by fusing multiple feature representation. *Bioinformatics*, **2020**, *36*(11), 3350-3356.
http://dx.doi.org/10.1093/bioinformatics/btaa160

[31] Charoenkwan, P.; Nantasenamat, C.; Hasan, M.M.; Shoombuatong, W. 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**.
http://dx.doi.org/10.1007/s10822-020-00323

[32] Khatun, M.S.; Hasan, M.M.; Kurata, H. PreAIP: computational prediction of anti-inflammatory peptides by integrating multiple complementary features. *Front. Genet.*, **2019**, *10*, 129.
http://dx.doi.org/10.3389/fgene.2019.00129 PMID: 30891059

[33] Islam, M.M.; Alam, M.J.; Ahmed, F.F.; Hasan, M.M.; Mollah, M.N.H. Improved prediction of protein-protein interaction mapping on homo sapiens by using amino acid sequence features in a supervised learning framework. *Protein Pept. Lett.*, **2020**, *28*(1), 74-83.
http://dx.doi.org/10.2174/0929866527666200610141258 PMID: 32520672

[34] Saidijam, M.; Azizpour, S.; Patching, S.G. Amino acid composition analysis of human secondary transport proteins and implications for reliable membrane topology prediction. *J. Biomol. Struct. Dyn.*, **2017**, *35*(5), 929-949.
http://dx.doi.org/10.1080/07391102.2016.1167622 PMID: 27159787

[35] Sahu, S.S.; Panda, G. A novel feature representation method based on Chou's pseudo amino acid composition for protein structural class prediction. *Comput. Biol. Chem.*, **2010**, *34*(5-6), 320-327.
http://dx.doi.org/10.1016/j.compbiolchem.2010.09.002 PMID: 21106461

[36] Breiman, L. SNP-based analysis of genetic substructure in the German population. *Mach. Learn.*, **2001**, *45*, 5-32.
http://dx.doi.org/10.1023/A:1010933404324

[37] Mosharaf, M.P.; Hassan, M.M.; Ahmed, F.F.; Shamima, K.M.; Moni, M. Mollah, M. N. H. Computational Prediction of Protein Ubiquitination Sites Mapping on *Arabidopsis Thaliana. Comput. Biol. Chem.*, **2020**, *85*, 107238.
http://dx.doi.org/10.1016/j.compbiolchem.2020.107238

[38] Charoenkwan, P.; Nantasenamat, C.; Hasan, M.M.; Shoombuatong, W. iTTCA-Hybrid: Improved and robust identification of tumor T cell antigens by utilizing hybrid feature representation. *Anal. Biochem.*, **2020**, *599*, 113747.
http://dx.doi.org/10.1016/j.ab.2020.113747 PMID: 32333902

[39] Hasan, M.M.; Manavalan, B.; Shoombuatong, W.; Khatun, M.S.; Kurata, H. i4mC-Mouse: Improved identification of DNA N4-methylcytosine sites in the mouse genome using multiple encoding schemes. *Comput. Struct. Biotechnol. J.*, **2020**, *18*, 906-912.
http://dx.doi.org/10.1016/j.csbj.2020.04.001 PMID: 32322372

[40] Charoenkwan, P.; Yana, J.; Schaduangrat, N.; Nantasenamat, C.; Hasan, M.M.; Shoombuatong, W. iBitter-SCM: Identification and characterization of bitter peptides using a scoring card method with propensity scores of dipeptides. *Genomics*, **2020**, *112*(4), 2813-2822.
http://dx.doi.org/10.1016/j.ygeno.2020.03.019 PMID: 32234434

[41] Hasan, M.M.; Khatun, M.S.; Kurata, H. iLBE for computational identification of linear B-cell epitopes by integrating sequence and evolutionary features. *Genomics Proteomics Bioinformatics*, **2020**, *S1672-0229*(18), 30274-2.

[42] Khatun, M.S.; Hasan, M.M.; Shoombuatong, W.; Kurata, H. ProIn-Fuse: improved and robust prediction of proinflammatory peptides by fusing of multiple feature representations. *J. Comput. Aided Mol. Des.*, **2020**, *34*(12), 1229-1236.
http://dx.doi.org/10.1007/s10822-020-00343-9 PMID: 32964284

[43] Basith Mail, S.; Manavalan, B.; Shin, T.H.; Lee, D.; Lee, G. Evolution of machine learning algorithms in the prediction and design of anticancer peptides. *Curr. Protein Pept. Sci.*, **2020**, *21*(12), 1242-1250.
http://dx.doi.org/10.2174/1389203721666200117171403 PMID: 31957610

[44] Andy, L.; Matthew, W. Classification and regression based on a forest of trees using random inputs. *R Package*, **2018**.

[45] Chatterjee, S. *Implements Adaboost based on C^{++} backend code*, **2016**. Available from: https://github.com/souravc83/fastAdaboost

[46] David, M.; Evgenia, D.; Kurt, H.; Andreas, W.; Friedrich, L.; Chih-Chung, C.; Chih-Chen, L. Functions for latent class analysis, short time Fourier transform, fuzzy clustering, support vector machines, shortest path computation, bagged clustering, naive Bayes classifier. **2019**. Available from: https://anaconda.org/bioconda/r-e1071/files?version=

[47] Manavalan, B.; Basith, S.; Shin, T.H.; Wei, L.; Lee, G. Meta-4mCpred: a sequence-based meta-predictor for accurate DNA 4mC site prediction using effective feature representation. *Mol. Ther. Nucleic Acids*, **2019**, *16*, 733-744. http://dx.doi.org/10.1016/j.omtn.2019.04.019 PMID: 31146255

[48] Vacic, V.; Iakoucheva, L.M.; Radivojac, P. Two Sample Logo: a graphical representation of the differences between two sets of sequence alignments. *Bioinformatics*, **2006**, *22*(12), 1536-1537. http://dx.doi.org/10.1093/bioinformatics/btl151 PMID: 16632492

[49] Manavalan, B.; Hasan, M.M.; Basith, S.; Gosu, V.; Shin, T.H.; Lee, G. Empirical comparison and analysis of web-based DNA N^4-methylcytosine site prediction tools. *Mol. Ther. Nucleic Acids*, **2020**, *22*, 406-420. http://dx.doi.org/10.1016/j.omtn.2020.09.010 PMID: 33230445

[50] Hasan, M.M.; Manavalan, B.; Khatun, M.S.; Kurata, H. i4mC-ROSE, a bioinformatics tool for the identification of DNA N4-methylcytosine sites in the Rosaceae genome. *Int. J. Biol. Macromol.*, **2020**, *157*, 752-758. http://dx.doi.org/10.1016/j.ijbiomac.2019.12.009 PMID: 31805335

[51] Charoenkwan, P.; Yana, J.; Nantasenamat, C.; Hasan, M.M.; Shoombuatong, W. iUmami-SCM: A novel sequence-based predictor for prediction and analysis of umami peptides using a scoring card method with propensity scores of dipeptides. *J. Chem. Inf. Model.*, **2020**, *60*(12), 6666-6678. http://dx.doi.org/10.1021/acs.jcim.0c00707 PMID: 33094610

[52] Hasan, M.M.; Basith, S.; Khatun, M.S.; Lee, G.; Manavalan, B.; Kurata, H. Meta-i6mA: an interspecies predictor for identifying DNA N6-methyladenine sites of plant genomes by exploiting informative features in an integrative machine-learning framework. *Brief. Bioinform.*, **2020**, bbaa202.

[53] Basith, S.; Manavalan, B.; Hwan Shin, T.; Lee, G. Machine intelligence in peptide therapeutics: A next-generation tool for rapid disease screening. *Med. Res. Rev.*, **2020**, *40*(4), 1276-1314. http://dx.doi.org/10.1002/med.21658 PMID: 31922268

[54] Chen, J.; Zhao, J.; Yang, S.; Chen, Z.; Zhang, Z. Prediction of protein ubiquitination sites in *Arabidopsis thaliana*. *Curr. Bioinform.*, **2019**, *14*(7), 614-620. http://dx.doi.org/10.2174/1574893614666190311141647