iSuc-ChiDT: A Computational Method for Identifying Succinylation Sites using Statistical Difference Table Encoding and the Chi-Square Decision Table Classifier

Ying Zeng
Hunan Institute of Engineering https://orcid.org/0000-0002-9358-8047
Yuan Chen
Hunan Agricultural University
Zheming Yuan (zhmyuan@sina.com)
Hunan Agricultural University

Research

Keywords: succinylation site, chi-square statistical difference table, ChiDT, imbalanced dataset, feature selection

Posted Date: June 10th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-590597/v1

Version of Record: A version of this preprint was published at BioData Mining on February 10th, 2022.
See the published version at https://doi.org/10.1186/s13040-022-00290-1.

iSuc-ChiDT: a computational method for identifying succinylation sites using statistical difference table encoding and the chi-square decision table classifier

Ying Zeng[1], Yuan Chen[2], Zheming Yuan[2*]

1. College of Computer and Communication, Hunan Institute of Engineering, Xiangtan 411104, China;

2. Hunan Engineering & Technology Research Center for Agricultural Big Data Analysis & Decision-making, Hunan Agricultural University, Changsha 410128, China;

***Corresponding author:** Zheming Yuan (zhmyuan@sina.com)

Abstract

Background: Lysine succinylation is a type of protein post-translational modification which is widely involved in cell differentiation, cell metabolism and other important physiological activities. To study the molecular mechanism of succinylation in depth, succinylation sites need to be accurately identified, and because experimental approaches are costly and time-consuming, there is a great demand for reliable computational methods. Feature extraction is a key step in building succinylation site prediction models, and the development of effective new features improves predictive accuracy. Because the number of false succinylation sites far exceeds that of true sites, traditional classifiers perform poorly, and designing a classifier to effectively handle highly imbalanced datasets has always been a challenge.

Results: We propose a new computational method, iSuc-ChiDT, to identify succinylation sites in proteins. In iSuc-ChiDT, chi-square statistical difference table encoding is developed to extract positional features, and has the highest predictive accuracy and fewest features compared to binary encoding and physicochemical property encoding. The chi-square decision table (ChiDT) classifier is designed to implement imbalanced pattern classification. With a training set of 4748:50,551(true: false sites), independent tests showed that ChiDT significantly outperformed traditional classifiers (including random forest, artificial neural network and relaxed variable kernel density estimator) in predictive accuracy and only taking 17s. Using an independent testing set of experimentally identified succinylation sites, iSuc-ChiDT achieved sensitivity of 70.47%, specificity of 66.27%, Matthews correlation coefficient of 0.205, and a global accuracy index Q^9 of 0.683, showing a significant improvement in sensitivity and overall accuracy compared to PSuccE, Success, SuccinSite and other existing succinylation site predictors.

Conclusions: iSuc-ChiDT shows great promise in predicting succinylation sites and is expected to facilitate further experimental investigation of protein succinylation.

Keywords

succinylation site, chi-square statistical difference table, ChiDT, imbalanced dataset, feature selection

Background

Protein post-translational modifications (PTMs) regulate cellular physiology and significantly increase protein diversity and complexity. Lysine succinylation is an evolutionarily conserved PTM present in both prokaryotic and eukaryotic cells where a succinyl group is covalently bonded

to specific lysine residues by enzymatic or non-enzymatic processes [1, 2]. Succinylation can promote remarkable changes in protein structure and function, and may play a role in many diseases, such as tuberculosis [3], allergic dermatitis [4], and inflammation [5]. Therefore, elucidating the molecular mechanism of succinylation will provide valuable information for both biomedical research and drug development.

Accurate identification of succinylation sites is critical to succinylation research, and because experimental methods are costly and time-consuming, and have been unable to keep up with the exponential growth of the number of sequenced proteins, efficient *in silico* methods are in great demand. To date, many predictors for identifying succinylation sites have been developed, such as SucPred [6], SuccinSite [7], pSuc-Lys [8], PSuccE [9], and so on, but with their limited overall accuracy and poor sensitivity, numerous true succinylation sites remain undetected. Actually, what interested us more is the information on true succinylation sites. Therefore, it is necessary to further improve predictive accuracy, especially sensitivity. Two key components, feature extraction and classifier construction, can greatly affect the accuracy of a computational method.

Commonly used features include positional features [7, 9-11], sequence composition [7-11], evolutionary information [12-14], and protein secondary structure [13-15]. Positional information of amino acids is basic but important to a protein sequence. While binary encoding [7, 9] is the most intuitive method to extract positional features, the feature matrix is very sparse. The binary encodings are the same for the same residue at different positions, and so it cannot reflect positional differences. Physicochemical property encoding [7, 9, 11] is another position-based amino acid encoding scheme that is frequently used. The AAindex [16] database records 531 physicochemical properties of 20 standard amino acids. Since it is not known in advance which

physicochemical properties are related to classification, physicochemical property encoding means each position needs to be represented by 531 physicochemical properties, resulting in many irrelevant and redundant features.

Traditional classifiers including support vector machine (SVM) [6, 9-11, 13], random forest (RF) [7, 8] and decision tree [12, 15] have been applied in succinylation site prediction. The number of false succinylation sites (non-succinylated lysine residues) far exceeds that of true sites, for example, the dataset from Hasan *et al.* [7] contains 5004/53524 true/false succinylation sites (a ratio of positive to negative samples of about 1:10). Training any traditional classifier with such highly imbalanced datasets could strongly bias classification results [17], and the large number of training samples would make the training time of some classifiers (*e.g.* SVM) unbearable. To address this, some methods (*e.g.* SucPred, SuccinSite) balanced the class distribution by under-sampling the negative samples, but this might lead to the loss of some potential classification information due to the mass discarding of negative samples; some methods (*e.g.* pSuc-Lys, PSuccE) designed classifier ensemble algorithms, however, they were still integrated results of several individual classifiers trained with a balanced subset where positive samples were repeatedly used.

With a highly imbalanced dataset, we developed an efficient approach called iSuc-ChiDT for predicting succinylation sites. We used chi-square statistical difference table encoding to extract positional features, then incorporated amino acid composition (AAC) and undirected pair-coupled amino acid composition (undirected-PCAAC) features. After feature selection with the Chi-MIC-share [18] algorithm, the ChiDT classifier was designed to resolve the imbalanced classification problem. The flow chart of our method is shown in Fig. 1.

Methods

Datasets

From Uni-ProtKB/Swiss-Prot [19] database and NCBI protein sequence database [20], Ning *et al*. [9] obtained 2322 succinylated proteins with 5009 experimentally verified lysine succinylation sites, by applying a 30% homology-reducing screening procedure with CD-HIT [21], and then randomly singled out 124 succinylated proteins to build an independent testing set, and used the remaining 2198 succinylated proteins as a training set. We used the same training and independent testing dataset as in Ning *et al*., which were freely available via the web link [22]. Our training set, namely Tr_data, contains 4748/50,551 true/false succinylation sites; and our testing set, namely Te_data, contains 254/2977 true/false succinylation sites.

Each true/false succinylation site was represented by a sequence fragment with an initial length of 51 amino acid residues, where the candidate site (lysine residue) was at the central position 0, and the upstream positions were successively labeled as -1, -2, …, -25, and the downstream positions labeled 1, 2, …, 25. If the number of up- or downstream residues of the candidate site was less than 25, amino acids were created through mirror extension to make up the difference [8]. For example, the original sequence of the succinylated protein "SP-P0ABS8" is "ML**K**NLAKLDQTEMDKVNVDLAAAGVAFKE…". The first lysine (*K*) is the candidate site and therefore the sequence fragment generated by mirror extension is "KFAVGAAALDVNVKDMETQDLKAML**K**NLAKLDQTEMDKVNVDLAAAGVAFK". All sequence samples contain only the 20 standard amino acids.

Compressing the 2×20 contingency table of each position with chi-square tests

The maximal information coefficient (MIC) [23] is a novel measure proposed to capture

dependences between paired variables. The MIC score ranges from 0 to 1, and only approaches 0 if two variables are statistically independent. To calculate the MIC score of the paired variables x and y, the ApproxMaxMI [23] algorithm sets the $n_x \times n_y < B(n)$, where $B(n) = n^{0.6}$ is the maximal grid size restriction, and n is the sample size, and n_x, n_y are the number of partition bins on x and y, respectively. The MIC score for two independent variables calculated by ApproxMaxMI depends on the ratio between $B(n)$ and n [24], and it is close to 0 only when n approaches infinity. For two independent variables under finite samples (especially for small sample sizes), ApproxMaxMI leads to a large deviation between the calculated MIC score and 0, meaning that the MIC will capture false associations. To address this drawback, Chen *et al.* [25] proposed an improved algorithm, ChiMIC [25], which uses chi-square test to determinate optimal bin size for the calculating of MIC score. For two independent variables with 100 sample points, ApproxMaxMI tends to fall into the maximal grid size ($100^{0.6} \approx 16$), and the corresponding grid partition is a 2×8 grid, and the MIC score is 0.24. With ChiMIC, the MIC score is only 0.06, and the corresponding grid partition is a 2×2 or 2×3 grid. This shows that the grid partition searched by ChiMIC is more reasonable and that compressing a 2×8 grid into a 2×2 or 2×3 grid is wise.

Similarly, for each position in succinylation site-containing sequences, we can construct a 2×20 contingency table by respectively counting the occurrence frequencies of the 20 standard amino acids in the positive and negative samples. For instance, Fig. 2 gives the 2×20 table of position -10 in Tr_data. What we need to investigate is whether the 2×20 table (2×20 grid) is reasonable, and could it be compressed into a 2×10, or even a 2×2 table? A similar attempt was made in donor splice site prediction. For each position in donor site-containing sequences, a 2×4 contingency table can be built by counting the frequencies of 4 bases in the positive and negative

samples. Following on from ChiMIC, Zeng *et al.* [26] compressed the 2×4 table of each position into a 2×l (2≤l≤4) table using chi-square tests, and developed a high-performance approach to predict donor splice sites based on this compression strategy.

Encouraged by the successful application of the compression strategy on nucleotide sequences, we applied it to protein sequences. For the 2×20 contingency table for each position in succinylation site-containing sequences, the compression procedure is described below.

Step 1: Set the initial value of r (r is an integer) to 20.

Step 2: The 2×r contingency table is compressed by merging two columns corresponding to two different residues, and some 2×(r-1) contingency tables are obtained, then select a 2×(r-1) contingency table with the maximum chi-square value, denoted as $max_{2\times(r\text{-}1)}$.

Step 3: A local 2×2 contingency table is constructed based on the merged residues in $max_{2\times(r\text{-}1)}$ and perform a chi-square test. If the p-value is lower than a given threshold, $max_{2\times(r\text{-}1)}$ is unreasonable and will be backtracked to the 2×r contingency table and the compression procedure is terminated. If the p-value is greater than a given threshold, $max_{2\times(r\text{-}1)}$ is reasonable, and a further compression of $max_{2\times(r\text{-}1)}$ is attempted following these steps: 1) set r=r-1; 2) if r≥3, repeat Step 2~3; otherwise, terminate compression.

Taking position -10 in Tr_data as an example, its 2×20 contingency table was finally compressed into a 2×3 table (Fig. 2). The 20 original status values of position -10 were therefore turned into 3 status values, *i.e.*, “ARDGTV”, “NQEHILKM” and “CFPSWY”, where, “ARDGTV” indicated that A, R, D, G, T, V at position -10 were regarded as the same status value, and the others were similar.

Key positions selection and window size determination

For each position in the sequences with 51 residues, a $2\times r$ ($2\leq r\leq 20$) contingency table can be obtained after compression based on the training set. A chi-square test was then performed on the $2\times r$ contingency table and the corresponding chi-square value was calculated. Higher chi-square values indicate that the corresponding positions are more important for discriminating positives from negatives. Fig. 3 shows the chi-square values of 50 positions (-25~+25, excluding position 0) in Tr_data, and the chi-square tests of all the positions are significant. We calculate the average of the chi-square values of all the positions, denoted as χ^2_{ave}, then set χ^2_{ave} as the threshold to select key positions. The chi-square values of positions -8, -4~-1, 1, 2, 5, 7 are above $\chi^2_{ave} = 92.797$ (see the red line in Fig. 3), therefore these 9 positions are regarded as the key positions. Furthermore, the contiguous 16 residues (positions -8~+7) are determined as the window size.

Positional feature extraction

For the 9 key positions in each sequence sample, we extracted 9 positional features, denoted as P_{-8}, P_{-4}, P_{-3}, P_{-2}, P_{-1}, P_1, P_2, P_5 and P_7 respectively, where, P_{-8} represents the positional feature of position -8, P_{-4} represents the positional feature of position -4, and so forth. The detailed process is described as follows.

In the training set, the occurrence frequencies of the 20 standard amino acids were counted at the i^{th} (i=1, 2, …, 9) position in the positive and negative samples, and then a 2×20 contingency table was built (Table 1).

Table 1 Frequency distribution of amino acids at the i^{th} position

Sample	Amino acid residue						Total
	1	2	…	j	…	20	
Positive	$f_{i,1}^{+}$	$f_{i,2}^{+}$	…	$f_{i,j}^{+}$	…	$f_{i,20}^{+}$	f_i^{+}
Negative	$f_{i,1}^{-}$	$f_{i,2}^{-}$	…	$f_{i,j}^{-}$	…	$f_{i,20}^{-}$	f_i^{-}
Total	$f_{i,1}$	$f_{i,2}$	…	$f_{i,j}$	…	$f_{i,20}$	N

In Table 1, $f_{i,j}^{+}$ represents the frequency of the j^{th} (j=1,2,…,20) residue at the i^{th} position in the positive samples, $f_{i,j}^{-}$ represents the corresponding frequencies in the negative samples, f_i^{+} and f_i^{-} represent the total number of positive and negative samples, and N represents the total number of samples. The chi-square value corresponding to the i^{th} position is calculated by:

$$\chi^2 = \frac{N^2}{f_i^{+} \times f_i^{-}} \left[\sum_{j=1}^{20} \frac{f_{i,j}^{+2}}{f_{i,j}} - \frac{f_i^{+2}}{N} \right] \quad (1)$$

If a new training sample is added, and the j^{th} residue appears at the i^{th} position, first assume this new training sample is positive, replace $f_{i,j}^{+}$ with $f_{i,j}^{+}+1$, and calculate a chi-square value $\chi_{i,j}^{2+}$ using formula (1); then assume this new training sample is negative, replace $f_{i,j}^{-}$ with $f_{i,j}^{-}+1$, and calculate a chi-square value $\chi_{i,j}^{2-}$ using formula (1). The score for the chi-square statistical difference table with the j^{th} residue at the i^{th} position is defined as:

$$\Delta\chi_{i,j}^2 = \chi_{i,j}^{2+} - \chi_{i,j}^{2-} \quad (2)$$

Next, build a 20×9 chi-square statistical difference table (Table 2). Table 2 gives the scores of the various amino acid residues at each position. If the j^{th} residue appears at the i^{th} position, the i^{th} positional feature will be assigned a value of $\Delta\chi_{i,j}^2$. Table S1 (Additional file 1) shows the 20×9 chi-square statistical difference table constructed based on 9 key positions in Tr_data.

Table 2 20×9 chi-square statistical difference table

Amino acid residue	Position				
	1	…	*i*	…	9
1	$\Delta\chi_{1,1}^2$	…	$\Delta\chi_{i,1}^2$	…	$\Delta\chi_{9,1}^2$
…	…	…	…	…	…
j	$\Delta\chi_{1,j}^2$	…	$\Delta\chi_{i,j}^2$	…	$\Delta\chi_{9,j}^2$
…	...	…	…	…	…
20	$\Delta\chi_{1,20}^2$	…	$\Delta\chi_{i,20}^2$	…	$\Delta\chi_{9,20}^2$

Compositional feature extraction

For each sequence sample with a window size of 16 residues, 230 compositional features are extracted, including 20 AAC features and 210 undirected-PCAAC features.

The AAC features are defined as the occurrence frequencies of the 20 standard amino acids in the sequence, respectively denoted as $f_A, f_R, \ldots, f_V$, where, f_A represents the frequency of alanine (A), f_R represents the frequency of arginine (R), and so forth.

The individual amino acid components are independent of each other, so the AAC features cannot reflect any correlation between amino acids. The pair-coupled amino acid composition [27] (PCAAC) features are composed of the occurrence frequencies of pairwise coupling between two adjacent residues, which can reflect both sequence components and the most preliminary association effect. To reduce feature dimension and solve the sparse problem of feature matrix, we assume that the pairwise coupling has no direction. For example, A-R coupling is treated the same as R-A coupling, and the corresponding pair-coupled component will be expressed by either f_{AR} or f_{RA}, where $f_{AR}(f_{RA})$ is the sum of AR pair occurrence frequency and RA pair occurrence frequency found in a sequence.

Feature selection based on Chi-MIC-share

Not all input features are equally important. Some may not be relevant to prediction, or there may be redundancies, so feature selection is a necessary step in constructing a reliable model. Minimum redundancy maximum relevance (mRMR) [28] is a popular feature selection method. However, relevance measure and redundancy measure in mRMR are not comparable, mRMR only gives the order of feature introduction and it is time-consuming to perform cross-validation in training sets to get the optimal feature subset. To address this, Li *et al.* [18] used ChiMIC as the unified measure of relevance and redundancy, and designed a redundancy sharing (rather than

redundancy removing) strategy to propose a novel feature selection method, Chi-MIC-share. We used the Chi-MIC-share algorithm in this study for feature selection.

Given an original feature set $\Omega = \{X_1, X_2, \ldots, X_i, \ldots, X_n\}$, $|\Omega|$ is the number of elements in Ω, and $|\Omega|=n$. If the introduced feature set is represented by S, the complement of S is represented as $\Omega_S=\Omega-S$. Denoting the response variable as Y, the Chi-MIC-share algorithm is described as follows.

For an introduced feature X_i in S, the score after redundancy sharing is calculated by:

$$\text{Chi-MIC-share}\left(X_i\right) = \sum_{X_j \in S} \frac{\text{Chi-MIC}\left(X_i;Y\right)}{\text{Chi-MIC}\left(X_i;X_j\right)} \tag{3}$$

The total score of all features in S after redundancy sharing is:

$$\text{Chi-MIC-share}\left(S\right) = \sum_{X_i \in S} \frac{\text{Chi-MIC}\left(X_i;Y\right)}{\sum_{X_j \in S} \text{Chi-MIC}\left(X_i;X_j\right)} \tag{4}$$

If the next introduced feature is X_{next}, set $E=S+\{X_{next}\}$, then $|E|=|S|+1$. The criterion for introducing the next optimal feature is:

$$\max_{X_{next} \in \Omega_S} \left[\text{Chi-MIC-share}\left(E\right)\right] = \sum_{X_i \in E} \frac{\text{Chi-MIC}\left(X_i;Y\right)}{\sum_{X_j \in E} \text{Chi-MIC}\left(X_i;X_j\right)} \tag{5}$$

If a new introduced feature no longer makes the total Chi-MIC-share score increase, this feature will be discarded and feature selection will be automatically terminated. Thus, the criterion for terminating feature introduction is:

$$\text{Chi-MIC-share}\left(E\right) \le \text{Chi-MIC-share}\left(S\right) \tag{6}$$

Furthermore, feature introduction can be forced to terminate according to the following criterion:

$$\frac{\text{Chi-MIC-share}\left(E\right) - \text{Chi-MIC-share}\left(S\right)}{\text{Chi-MIC-share}\left(S\right)} \le 0.01 \tag{7}$$

Classifier construction

To efficiently realize the imbalanced pattern classification, a new classifier, ChiDT, was designed as follows.

(1) Compress the 2×*m* contingency table of each retained feature

For each feature retained by the Chi-MIC-share feature selection, its 2×*m* contingency table (*m* is the number of original status values of the feature) was compressed according to the previously described procedure to obtain a 2×*r* contingency table (*r* is the number of new status values of the feature, 2≤*r*≤*m*). During the compression process, since the status values of each retained feature are continuous, only adjacent status values could be merged together.

(2) Introduce the retained features one by one

Supposing the proportion of the k^{th} class samples in sample set D is p_k (k=1, 2), the information entropy of D is defined as:

$$H(D) = -\sum_{k=1}^{2} p_k \log_2 p_k \tag{8}$$

Given a Chi-MIC-share retained feature X_i, supposing it has r new status values as $\{s_1, s_2, \ldots, s_j, \ldots, s_r\}$ after compressing, then the information gain that X_i brings for D can be calculated by:

$$Gain(D, X_i) = H(D) - \sum_{j=1}^{r} \frac{|D^j|}{|D|} H(D^j) \tag{9}$$

where D^j represents the samples in D whose X_i takes the status value as s_j ($1 \le j \le r$), while $H(D^j)$ is the information entropy of D^j.

From the features whose information gains are above the average, pick out the one with the highest gain ratio to be the first introduced feature. Here, the gain ratio of X_i is defined as:

$$GainRatio(D, X_i) = \frac{Gain(D, X_i)}{IV(X_i)} \tag{10}$$

where

$$IV(X_i) = -\sum_{j=1}^{r} \frac{|D^j|}{|D|} \log_2 \frac{|D^j|}{|D|} \tag{11}$$

and $IV(X_i)$ is the intrinsic value of X_i.

Next, the remaining features are introduced one by one with the following steps.

Step 1: Under the conditions in which the introduced features have existed, the $2\times r$ contingency table of each remaining feature is further compressed. If the r columns of the $2\times r$ contingency table are compressed into one column, the remaining feature cannot be introduced; if the r columns are not compressed into one column, the remaining feature will be considered as a candidate feature to be introduced.

Step 2: Calculate the information gain of every candidate feature. From the candidate features whose information gains are above the average, the one with the highest gain ratio is selected to be the next introduced feature.

Step 3: Repeat Step 1~2 until no further features can be introduced.

After this, the introduced features with their status values generate various rules. Taking Tr_data as an example, 10 Chi-MIC-share retained features were finally introduced and 137 rules were generated (see Additional file 1: Table S2).

(3) Construct a balanced decision table for making decisions

We counted the number of positive and negative training samples conforming to each rule then constructed a 2×137 imbalanced decision table (Table 3).

Table 3 Imbalanced decision table

Sample	Rule*			Total
	(P_{-1}=-2.028) ∧ (-0.907≤P_2≤0.501) ∧ (-0.715≤P_{-8}≤0.066)	…	(P_{-1}=1.839) ∧ (P_1=2.060)	
Positive	23	…	32	4748
Negative	2907	…	83	50551

*For instance, "(P_{-1}=-2.028)∧(-0.907≤P_2≤0.501)∧(-0.715≤P_{-8}≤0.066)" represents P_{-1} taking a value of -2.028 and P_2 ranging from -0.907 to 0.501 and P_{-8} ranging from -0.715 to 0.066, where, "∧" denotes the logical conjunction.

The number of negative samples far exceeds the positives. To resolve the imbalanced classification problem, based on cost-sensitive learning [29], we adjust the decision weight of negative samples in each column of the imbalanced decision table, by multiplying the number of negative samples in each column of Table 3 by θ, where θ is defined as the ratio of the total number of positive and negative training samples, here, θ=4748/50551. Then, a 2×137 balanced decision table is obtained (Table 4).

Table 4 Balanced decision table

Sample	Rule*			Total
	(P_{-1}=-2.028) ∧ (-0.907≤P_2≤0.501) ∧ (-0.715≤P_{-8}≤0.066)	…	(P_{-1}=1.839) ∧ (P_1=2.060)	
Positive	23	…	32	4748
Negative	273.04	…	7.80	4748

We can then use the balanced decision table for making decisions. Suppose that a testing sample meets the rule "(P_{-1}=-2.028) ∧ (-0.907≤P_2≤0.501) ∧ (-0.715≤P_{-8}≤0.066)". First, we assume that it is positive and replace 23 with 23+1, then calculate the corresponding chi-square value χ_{i+}^2. We then assume that it is negative and replace 273.04 with 273.04+1, then calculate the corresponding chi-square value χ_{i-}^2. If $\chi_{i+}^2 > \chi_{i-}^2$, the testing sample is predicted to be positive, if not, it is negative.

Performance evaluation

Sensitivity (SN), specificity (SP) and Matthews correlation coefficient (MCC) as the common indexes for evaluating binary classification are defined as follows:

$$SN = \frac{TP}{TP + FN} \tag{12}$$

$$SP = \frac{TN}{TN + FP} \tag{13}$$

$$MCC = \frac{TP \times TN - FN \times FP}{\sqrt{(TP + FN) \times (TP + FP) \times (TN + FP) \times (TN + FN)}} \tag{14}$$

Here, TP, FP, TN, and FN denote the numbers of true positives, false positives, true negatives, and

false negatives. MCC is a balanced statistical index that considers SN and SP, but it is sensitive to class distribution in a testing set. As shown in Table 5, when a prediction model has an SN of 93% and SP of 95%, as the imbalance degree of the testing set grows, the MCC value declines. This shows that a low MCC value does not always indicate poor prediction performance as it may be caused by a highly imbalanced testing set.

Table 5 Various evaluation indexes on different ratios of positives to negatives

Positives/Negatives*	SN (%)	SP (%)	MCC	Q^9
100/100	93.00	95.00	0.880	0.939
100/1000	93.00	95.00	0.752	0.939
100/10000	93.00	95.00	0.371	0.939

* positive testing sample size/negative testing sample size

The content-balancing accuracy index Q^9 [30] is independent of the class distribution of the dataset and has been widely used to evaluate performance of many prediction programs including gene-finding, splice site prediction and protein secondary structure prediction [31-33]. As Table 5 shown, the value of Q^9 remains unchanged across different ratios of positives to negatives. In this study, we introduced Q^9 as the measure of global accuracy to evaluate the prediction performance of models in case of an imbalanced testing set. Q^9 is defined as:

$$Q^9 = (1 + q^9)/2 \tag{15}$$

where

$$q^9 = \begin{cases} \text{(TN-FP)/(TN+FP)}, & \text{if TP+FN=0} \\ \text{(TP-FN)/(TP+FN)}, & \text{if TN+FP=0} \\ 1-\sqrt{2}\sqrt{\left[FN/(TP+FN)\right]^2+\left[FP/(TN+FP)\right]^2}, & \text{if TP+FN} \neq 0 \text{ and TN+FP} \neq 0 \end{cases}$$

The value of Q^9 ranges from 0 to 1, and the larger the Q^9 value, the better the prediction performance.

Results and discussion

Features retained by Chi-MIC-share

Based on Tr_data, the Chi-MIC-share feature selection was performed on 239 original input features (9 positional features and 230 compositional features). As shown in Fig. 4, when the 37th feature was introduced, the Chi-MIC-share score peaked (0.12544), after which is began to decline and feature selection was automatically terminated. To improve computational efficiency, forced termination criteria were adopted and 10 features were retained (see the red line in Fig. 4). Table 6 describes the retained features in detail. It can be seen that positional features contribute 80% of all the retained features, indicating that positional features have an important contribution to succinylation site prediction.

Table 6 Features retained by Chi-MIC-share

No.	Retained features	Type	*No.*	Retained features	Type
1	P_{-1}	position	6	P_{7}	position
2	P_{1}	position	7	f_R	AAC
3	P_{-2}	position	8	f_K	AAC
4	P_{5}	position	9	P_{2}	position
5	P_{-8}	position	10	P_{-3}	position

No. denotes the order of feature introduction

Comparison of different classifiers

Based on the same input features (10 features retained by Chi-MIC-share), the ChiDT classifier was compared to traditional classifiers including random forest (RF), artificial neural network (ANN) and relaxed variable kernel density estimator (RVKDE) [34]. We choose RVKDE for the comparison because it delivers the same level of accuracy as SVM when the number of training samples exceeds 10,000, with a significantly lower average time complexity of $O(n\log n)$ [34]. RF and ANN classifiers were built with Weka 3.8.1 and the neural network toolbox in Matlab R2015a, respectively, and all parameters took the default values. The independent tests based on Tr_data and Te_data were employed for comparisons (Table 7).

Table 7 Independent test accuracy based on different classifiers

Classifier	SN (%)	SP (%)	MCC	Q^9
RF	2.75	99.83	0.115	0.312
ANN	0	99.90	-0.009	0.293
RVKDE	9.84	97.25	0.106	0.362
ChiDT	**70.47**	**66.27**	**0.205**	**0.683**

The results show that: 1) ChiDT achieved a significantly higher predictive accuracy and effectively realized imbalanced pattern classification. When the training set was imbalanced (4748 positives/50,551 negatives), the prediction results of RF, ANN and RVKDE were biased to negative samples, resulting in poor sensitivities (SNs<10%). With ANN, while specificity was up to 99.9%, sensitivity was equal to 0. This meant that all positive samples were predicted to be negative and thus the global accuracy of ANN was the lowest (Q^9=0.293). ChiDT built a balanced decision table through weighted correction strategy to perform imbalanced classification and obtained the highest accuracy (Q^9=0.683). 2) ChiDT has a satisfactory calculation speed and can be applied to large samples. All simulations were run on an Intel Core i5-3320M 2.6 GHz/8 GB RAM system, and the elapsed time of ChiDT and RVKDE were 17 seconds and 18 minutes, respectively. ChiDT's high speed is achieved because there is no need for parameter optimization.

Comparison of different position-based encoding schemes

Based on the 9 key positions in Tr_data (here using 4748 positive samples and 4748 negative samples), 5-fold cross validation was applied to evaluate binary encoding, physicochemical property encoding (including 531 physicochemical properties [7] and 10 physicochemical properties [9] for encoding) and chi-square statistical difference table encoding, respectively. The results showed that chi-square statistical difference table encoding achieved the highest predictive accuracy and the fewest features (Table 8).

Table 8 5-fold cross accuracy based on different encoding schemes

Encoding scheme	Feature dimension	SN (%)	SP (%)	MCC	Q^9
Binary	180	63.20	62.41	0.258	0.623
Physicochemical properties(531)	4779	58.86	60.39	0.188	0.593
Physicochemical properties(10)	90	59.77	62.59	0.225	0.607
Chi-square statistical difference table	**9**	**65.91**	**62.91**	**0.289**	**0.641**

Binary encoding means that each position is represented by 20 0/1-features and the corresponding feature matrix is therefore very sparse. When using binary encoding scheme, the encodings of the same residue at different positions are the same, which does not reflect positional difference, and for different residues at the same position, it does not reflect the degree of difference between residues. For example, the amino acid polarity indexes of residue S, T and R at the same position are 1.67, 1.66 and 52, respectively, indicating that the polarity difference between S and T is small, and between T and R is large, but the hamming distances of both S-T and S-R are equal to 2 when using binary encoding. As for physicochemical property encoding, when 531 amino acid indices in AAindex were all considered for sequence characterization, the number of features reached 531×9=4779 (Table 8), and a lot of irrelevant and redundant features would be seen. Ning *et al* [9] ranked 531 physicochemical properties according to their abilities to distinguish between true and false succinylation sites, then chose the top 10 physicochemical properties for sequence encoding, so that the feature dimension was greatly reduced. However, as shown in Table 8, no matter whether 531 or 10 physicochemical properties are used, the predictive accuracy is always lower than that of chi-square statistical difference table encoding.

Chi-square statistical difference table encoding reflects the difference of the same residue at different positions, as well as the degree of difference between different residues at the same position, thus, it could differentiate between the highly similar positive and negative samples. Another benefit of chi-square statistical difference table encoding is that it has a low feature

dimension, low redundancy, and a non-sparse feature matrix.

Comparison of different window sizes

Based on Tr_data and Te_data, independent tests were performed to compare the prediction performance of the determined window size (-8~+7) with longer (*e.g.* -25~+25, -15~+15) and shorter window sizes (*e.g.* -5~+5). The results (Table 9) show that the proposed model with a window size of 16 residues (-8~+7) can achieve higher independent test accuracy compared to other window sizes. This indicates that an overly long window size may introduce some irrelevant information, while too short a window may lead to insufficient information collection, both of which reduce predictive accuracy. This confirms that our window size determination is reliable.

Table 9 Independent test accuracy based on different window sizes

Window size	SN (%)	SP (%)	MCC	Q^9
51(-25~+25)	68.50	61.10	0.162	0.646
31(-15~+15)	64.57	66.48	0.174	0.655
16(-8~+7)	**70.47**	**66.27**	**0.205**	**0.683**
11(-5~+5)	62.20	65.03	0.152	0.636

Necessity of Chi-MIC-share feature selection

Based on Tr_data and Te_data, the independent test results with or without Chi-MIC-share feature selection are shown in Table 10. They show that feature selection based on Chi-MIC-share can: 1) improve predictive accuracy, with the Q^9 value improving from 0.663 to 0.683, and 2) reduce feature dimension and save computational time. After feature selection, the number of original input features was reduced from 239 to 10, and the elapse time of ChiDT was reduced by 95%. Therefore, it is necessary and beneficial to perform a Chi-MIC-share feature selection.

Table 10 Independent test accuracy with and without Chi-MIC-share

Feature selection	Feature dimension	SN (%)	SP (%)	MCC	Q^9	Time(mm:ss)
No feature selection	239	70.08	62.95	0.182	0.663	06:14
Chi-MIC-share	10	70.47	66.27	0.205	0.683	00:17

Comparison with existing methods

To further evaluate the performance of our method (iSuc-ChiDT), we compared it with existing succinylation site predictors, SucPred, iSuc-PseAAC [10], SuccFind [11], SuccinSite, iSuc-PseOpt [35], pSuc-Lys, Success [13] and PSuccE, using the same independent testing set (Te_data). The results show that iSuc-ChiDT had a superior overall accuracy (Q^9=0.683) and sensitivity (70.47% *vs*. 12.20%~37.50%) (Table 11).

Table 11 Independent test accuracy for different methods

Method	SN (%)	SP (%)	MCC	Q^9
SucPred	27.20	67.30	-0.030	0.436
iSuc-PseAAC	12.20	88.70	0.013	0.374
SuccFind	25.20	79.20	0.029	0.451
SuccinSite	37.10	88.20	0.199	0.548
iSuc-PseOpt	30.30	75.80	0.038	0.478
pSuc-Lys	22.40	82.60	0.036	0.436
Success	14.20	86.80	0.007	0.386
PSuccE	37.50	88.60	0.204	0.551
iSuc-ChiDT	**70.47**	**66.27**	**0.205**	**0.683**

Positional information of amino acids is valuable for succinylation site prediction. Most compared methods used binary encoding or physicochemical property encoding to extract positional features. iSuc-ChiDT used chi-square statistical difference table encoding and our experiments showed that it was superior to these two encoding schemes (see Table 8). Moreover, iSuc-ChiDT combined positional features and compositional features to characterize samples. Employing the independent tests with Tr_data and Te_data, the MCC values of 9 positional features, 230 compositional features and 239 combinational features-based models were 0.153, 0.099 and 0.182, respectively, confirming that feature fusion improved predictive accuracy. The ChiDT classifier outperformed traditional classifiers when dealing with imbalanced datasets (see Table 7), further supporting the observation that iSuc-ChiDT could achieve better prediction performance.

Conclusion

In this study, a computational method called iSuc-ChiDT is proposed to identify protein succinylation sites, which incorporates chi-square statistical difference table encoding and the ChiDT classifier. Chi-square statistical difference table encoding can differentiate between highly similar positive and negative sequences, and its advantages include a reduced feature dimension and non-sparse feature matrix, and the ChiDT classifier efficiently resolves the imbalanced dataset problem, both of which contribute to the accurate prediction of succinylation sites. iSuc-ChiDT greatly improved sensitivity and overall accuracy compared to previous predictors, and it will serve as an useful complementary tool for detecting potential succinylation sites in proteins. In future studies, we aim to explore more valuable features (*e.g.* evolutionary information, structural information) for characterizing succinylation sites, in pursuit of better prediction performance.

Additional file

Additional file 1: Table S1 (.xlsx 14KB). This table shows the 20×9 chi-square statistical difference table constructed based on 9 key positions in Tr_data.

Additional file 2: Table S2 (.xlsx 26KB). This table shows 137 rules generated based on Tr_data, and lists the number of positive and negative training samples conforming to each rule.

Abbreviations

ChiDT: chi-square decision table; PTM: post-translational modification; SVM: support vector machine; RF: random forest; ANN: artificial neural network; RVKDE: relaxed variable kernel density estimator; AAC: amino acid composition; PCAAC: pair-coupled amino acid composition; undirected-PCAAC: undirected pair-coupled amino acid composition; SN: sensitivity; SP: specificity; TP: true positive; FP: false positive; TN: true negative; FN: false negative; MCC: Matthews correlation coefficient; MIC: maximal information coefficient; mRMR: minimum redundancy maximum relevance.

Declarations

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Availability of data and material

All data generated or analyzed during this study are included in this published article and its supplementary information files.

Competing interests

The authors declare that they have no competing interests.

Funding

This research was supported by the Doctoral Science Foundation of Hunan Institute of Engineering and the Youth Key Research Project of Hunan Institute of Engineering (No. XJ2002).

Authors' contributions

YZ, ZMY conceived and designed the experiments. YZ performed the experiments and drafted the manuscript. ZMY revised the manuscript. YC contributed software coding. All authors read and approved the final manuscript.

Acknowledgments

We would like to thank all the anonymous reviewers for their constructive advices.

Authors' information

[1]College of Computer and Communication, Hunan Institute of Engineering, Xiangtan, Hunan, China. [2]Hunan Engineering & Technology Research Center for Agricultural Big Data Analysis & Decision-making, Hunan Agricultural University, Changsha, Hunan, China.

References

1. Zhang ZH, Tan MJ, Xie ZY, Dai LZ, Chen Y, Zhao TM. Identification of lysine succinylation as a new post-translational modification. Nature Chemical Biology. 2011(1);7:58-63.
2. Papanicolaou KN, O'Rourke B, Foster DB. Metabolism leaves its mark on the powerhouse: recent progress in post-translational modifications of lysine in mitochondria. Frontiers in Physiology. 2014;5:301.
3. Xu XY, Liu T, Yang J, Chen LH, Liu B, Wei CD, *et al*. The first succinylome profile of Trichophyton rubrum reveals lysine succinylation on proteins involved in various key cellular processes. BMC Genomics. 2017;18:577.
4. Shershakova N, Bashkatova E, Babakhin A, Andreev S, Nikonova A, Shilovsky L, *et al*. Allergen-specific immunotherapy with monomeric allergoid in a mouse model of atopic

dermatitis. PloS One. 2015;10(8):e0135070.
5. Tannahill GM, Curtis AM, Adamik J, Palsson-McDermott EM, McGettrick AF, Goel G, et al. Succinate is an inflammatory signal that induces IL-1β through HIF-1α. Nature. 2013;496:238-42.
6. Zhao XW, Ning Q, Chai HT, Ma ZQ. Accurate in silico identification of protein succinylation sites using an iterative semi-supervised learning technique. Journal of Theoretical Biology. 2015;374:60-5.
7. Hasan MM, Yang SP, Zhou Y, Mollah MN. SuccinSite: a computational tool for the prediction of protein succinylation sites by exploiting the amino acid patterns and properties. Molecular BioSystems. 2016;12:786-95.
8. Jia JH, Liu Z, Xiao X, Liu BX, Chou KC. pSuc-Lys: Predict lysine succinylation sites in proteins with PseAAC and ensemble random forest approach. Journal of Theoretical Biology. 2016;394:223-30.
9. Ning Q, Zhao XS, Bao LL, Ma ZQ, Zhao XW. Detecting succinylation sites from protein sequences using ensemble support vector machine. BMC Bioinformatics. 2018; 19(1):237-46.
10. Xu Y, Ding YX, Ding J, Lei YH, Wu LY, Deng NY. iSuc-PseAAC: predicting lysine succinylation in proteins by incorporating peptide position-specific propensity. Scientific Reports. 2015;5(1):10184.
11. Xu HD, Shi SP, Wen PP, Qiu JD. SuccFind: a novel succinylation sites online prediction tool via enhanced characteristic strategy. Bioinformatics. 2015;31(23): 3748-50.
12. Dehzangi A, López Y, Lal SP, Taherzadeh G, Michaelson J, Sattar A, *et al*. PSSM-Suc: Accurately predicting succinylation using position specific scoring matrix into bigram for feature extraction. Journal of Theoretical Biology. 2017;425:97-102.
13. López Y, Sharma A, Dehzangi A, Lal SP, Taherzadeh G, Sattar A, *et al*. Success: evolutionary and structural properties of amino acids prove effective for succinylation site prediction. BMC Genomics. 2018;19 Suppl 1:923-27.
14. Dehzangi A, López Y, Lal SP, Taherzadeh G, Sattar A, Tsunoda T, *et al*. Improving succinylation prediction accuracy by incorporating the secondary structure via helix, strand and coil, and evolutionary information from profile bigrams. PLoS ONE. 2018;13(2):e0191900.
15. López Y, Dehzangi A, Lal SP, Taherzadeh G, Michaelson J, Sattar A, *et al*. SucStruct: Prediction of succinylated lysine residues by using structural properties of amino acids. Analytical Biochemistry. 2017;527:24-32.
16. Kawashima S, Ogata H, Kanehisa M. AAindex: amino acid index database. Nucleic Acids Research. 1999;27(1): 368-69.
17. Weiss GM, Provost F. The effect of class distribution on classifier learning: An empirical study. Technical Report ML-TR-44. Department of Computer Science, Rutgers University, 2001.
18. Li YT, Dai ZJ, Cao D, Luo F, Chen Y, Yuan ZM. Chi-MIC-share: a new feature selection algorithm for quantitative structure-activity relationship models. RSC Advances. 2020;10:19852-60.
19. UniProt Consortium. Ongoing and future developments at the Universal Protein Resource. Nucleic Acids Research. 2011;39(Database issue):214-9.

20. NCBI protein sequence database. https://www.ncbi.nlm.nih.gov/protein/. Accessed 21 May 2021.
21. Li WZ, Godzik A. Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. Bioinformatics. 2006;22(13):1658-9.
22. PSuccE. https://github.com/ningq669/PSuccE. Accessed 17 April 2021.
23. Reshef DN, Reshef YA, Finucane HK, Grossman SR, McVean G, Turnbaugh PJ, *et al*. Detecting novel associations in large data sets. Science. 2011;334:1518-24.
24. Reshef DN, Reshef YA, Finucane HK, Grossman SR, McVean G, Turnbaugh PJ, *et al*. Supporting Online Material for Detecting Novel Associations in Large Data Sets. Science. 2011; 334:1518-24.
25. Chen Y, Zeng Y, Luo F, Yuan ZM. A new algorithm to optimize maximal information coefficient. PLoS One. 2016;11(6):e0157567.
26. Zeng Y, Yuan HJ, Yuan ZM, Chen Y. A high-performance approach for predicting donor splice sites based on short window size and imbalanced large samples. Biology Direct. 2019;14:6.
27. Chou KC. Using pair-coupled amino acid composition to predict protein secondary structure content. Journal of Protein Chemistry. 1999;18(4):473-80.
28. Peng H, Long F, Ding C. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. IEEE Transactions on Pattern Analysis & Machine Intelligence. 2005;27(8), 1226-38.
29. Sun YM, Kamel MS, Wong AKC, Wang Y. Cost-sensitive boosting for classification of imbalanced data. Pattern Recognition, 2007;40(12):3358-78.
30. Zhang CT, Zhang R. Evaluation of gene-finding algorithms by a content-balancing accuracy index. Journal of Biomolecular Structure and Dynamics. 2002;19(6):1045-52.
31. Zhang CT, Ren Z. Q9, a content-balancing accuracy index to evaluate algorithms of protein secondary structure prediction. International Journal of Biochemistry and Cell Biology. 2003;35:1256-62.
32. Zhang QW, Peng QK, Zhang Q, Yan YH, Li KK, Li J. Splice sites prediction of human genome using length-variable Markov model and feature selection. Expert Systems with Applications. 2010;37:2771-82.
33. Wei D, Zhang HL, Wei YJ, Jiang QS. A Novel Splice Site Prediction Method using Support Vector Machine. Journal of Computational Information Systems. 2013; 20:8053-60.
34. Oyang YJ, Hwang SC, Ou YY, Chen CY, Chen ZW. Data classification with radial basis function networks based on a novel kernel density estimation algorithm. IEEE Transactions on Neural Networks. 2005;16(1):225-36.
35. Jia JH, Liu Z, Xiao X, Liu BX, Chou KC. iSuc-PseOpt: Identifying lysine succinylation sites in proteins by incorporating sequence-coupling effects into pseudo components and optimizing imbalanced training dataset. Analytical Biochemistry. 2016;497:48-56.

Fig. 1 Flow chart of iSuc-ChiDT

Fig. 2 Illustration of compression procedure (position -10 in Tr_data)

Fig. 3 Chi-square values for different positions in Tr_data

Fig. 4 Chi-MIC-share scores after introduction of each feature. The red line represents the forced termination of feature introduction.

Figures

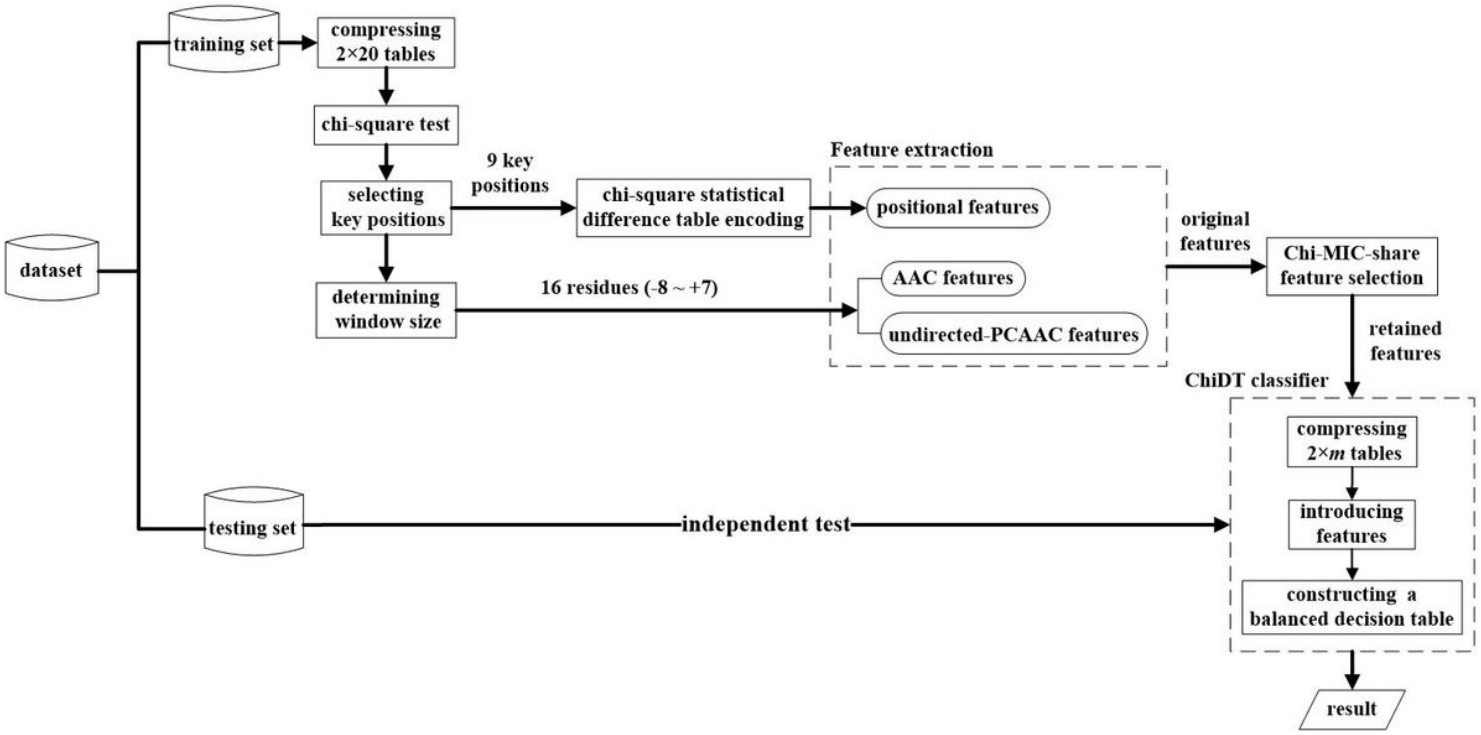

Figure 1

Flow chart of iSuc-ChiDT

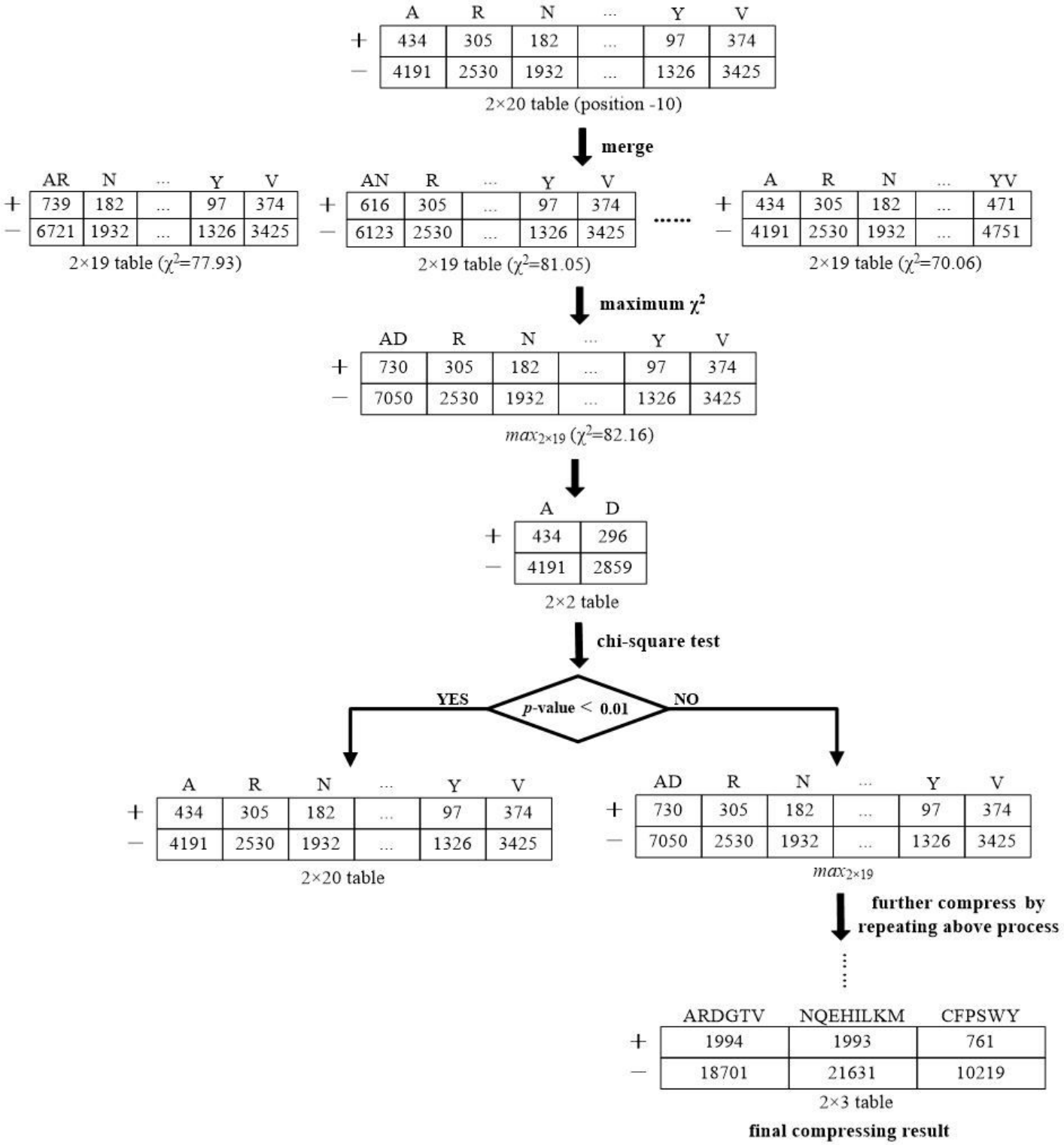

Figure 2

Illustration of compression procedure (position -10 in Tr_data)

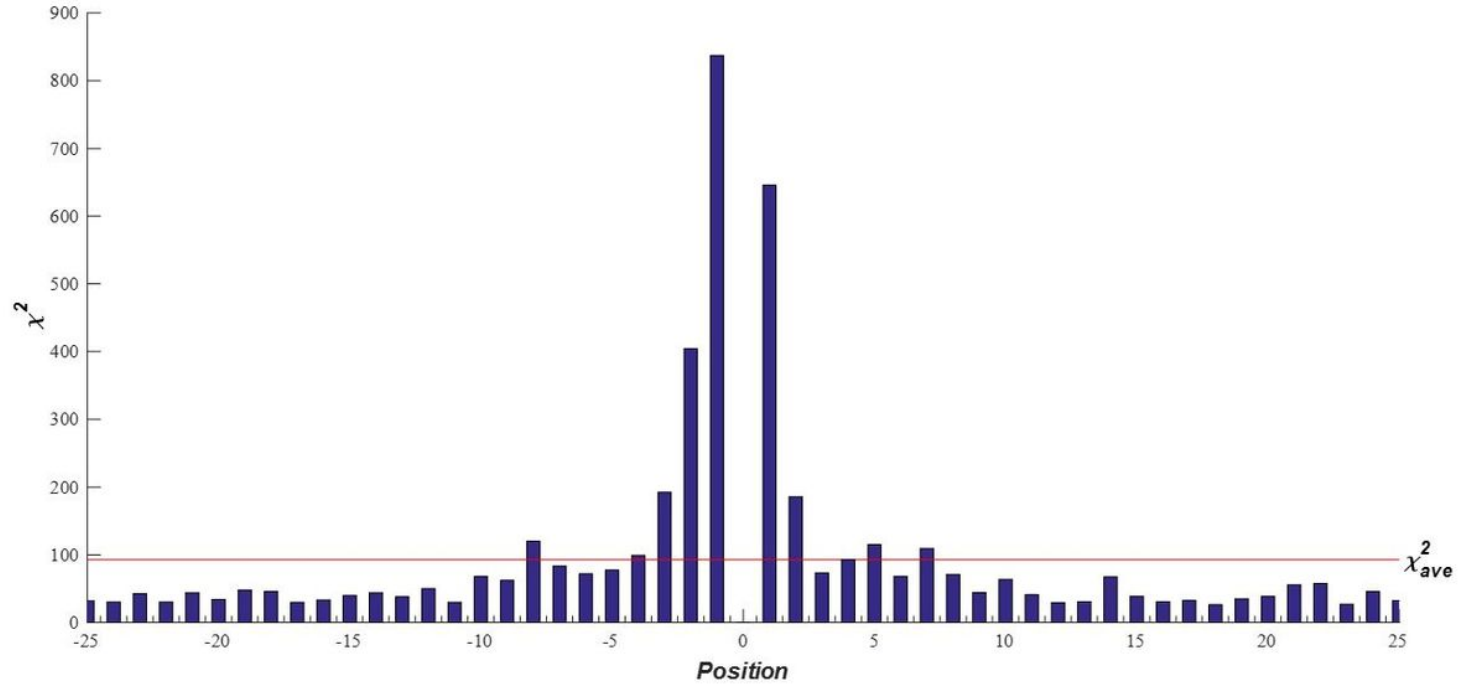

Figure 3

Chi-square values for different positions in Tr_data

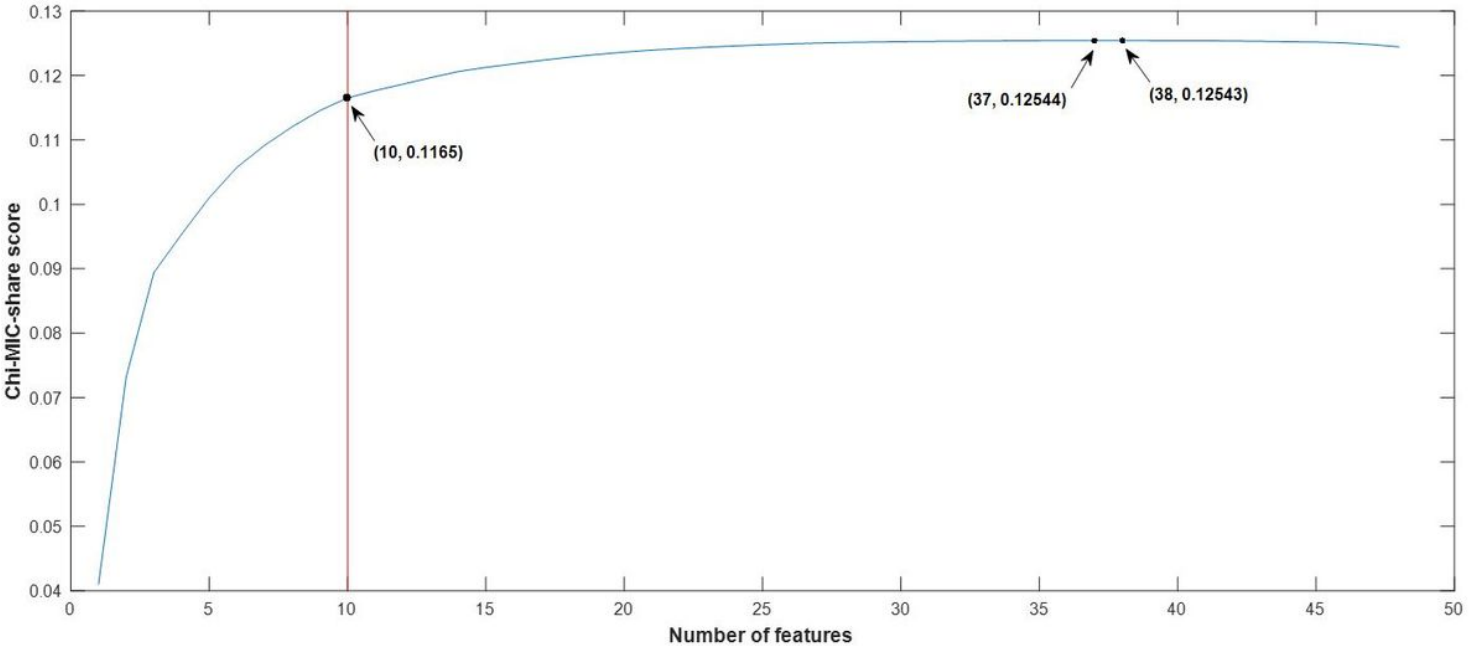

Figure 4

Chi-MIC-share scores after introduction of each feature. The red line represents the forced termination of feature introduction.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- TableS1.xlsx
- TableS2.xlsx