Genomics xxx (xxxx) xxx–xxx

Contents lists available at ScienceDirect

Genomics

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

Original Article

iRSpot-DTS: Predict recombination spots by incorporating the dinucleotide-based spare-cross covariance information into Chou's pseudo components

Shengli Zhang[a,*], Kaiwen Yang[b], Yuqing Lei[c,1], Kang Song[c,1]

[a] School of Mathematics and Statistics, Xidian University, Xi'an, 710071, PR China
[b] School of Electronic Engineering, Xidian University, Xi'an, 710071, PR China
[c] School of Computer Science, Xidian University, Xi'an, 710071, PR China

ARTICLE INFO

Keywords:
Recombination spots
Spatial autocorrelation
Cross correlation
T-SNE
SAE softmax classifier

ABSTRACT

Meiotic recombination plays an important role in the process of genetic evolution. Previous researches have shown that the recombination rates provide important information about the mechanism of recombination study. However, at present, most methods ignore the hidden correlation and spatial autocorrelation of the DNA sequence. In this study, we proposed a predictor called iRSpot-DTS to identify hot/cold spots based on the benchmark datasets. We proposed a feature extraction method called dinucleotide-based spatial autocorrelation (DSA) which can incorporate the original DNA properties and spatial information of DNA sequence. Then it used t-SNE method to remove the noise which outperformed PCA. Finally, we used SAE softmax classifier to do classification which is based on networks and can get more hidden information of DNA sequence, our iRSpot-DTS achieved remarkable performance. Jackknife cross validation tests were done on two benchmark datasets. We achieved state-of-the-art results with 96.61% overall accuracy(OA), 93.16% Matthews correlation coefficient (MCC) and over 95% in Sn and Sp which are the best in this state.

1. Introduction

Meiotic recombination plays an important role in the process of genetic evolution. It maintenances the sequence diversity in human genomes and provides chance for natural exchanges of genetic material [1]. Therefore, recombination is considered a main driven force in these variations. How to estimate the gene recombination rate becomes a geneticist's concern [2]. Through researches, it has been demonstrated that the gene recombination rates between different chromosomes of the same species and even different regions of the same chromosome are different [3]. In general, regions of chromosomes with high recombination rate are called hot spots and regions with low recombination rates are called cold spots [4].

With the rapid development of bioinformatics, the number of gene sequences has increased explosively. However, traditional methods such as biology experiments and comparative genomics methods to identify the recombination are time-consuming and labor-intensive [5,35,36]. Therefore, a stable, strong and effective method is in need.

Recently, several computational methods have been proposed to predict recombination hotspots and cold spots. Most of them are based on sequence content information and are developed by some machine learning methods because various sequence information is easy to incorporate. RF-DYMHC [35] was proposed by random forest model using gapped dinucleotide composition features and iRSpot-GAEnsC [37] was developed based on nucleotide dinucleotide and trinucleotide content by Genetic algorithm. IDQD [38] was developed with k-mer frequencies along with DNA sequences. However, these methods ignored some sequence order information which shows the discriminative power in many previous studies [39] and was proved useful in identifying recombination spots. To address this problem, some methods based on DNA property matrix which can reflect the sequence order information are proposed, such as iRSpot-PseDNC [41], iRSpot-DACC [42], iRSpot-DACC-PCA [42], iRSpot-EL [43], iRSpot-TNCPseAAC [44], and iRSpot-SF [40].

Driven by previous studies, a predictor which aimed to further improve the quality of identifying recombination spots was proposed. Our work was focused on how to extract discriminatory information from the dinucleotide pairs at different positions along with the given sequence, how to select the features and how to use the most effective classifier to achieve great performance. We have defined a feature extraction method called dinucleotide-based spatial autocorrelation (DSA), which is able to reflect the comprehensive sequence-order

* Corresponding author.
E-mail address: zhangsl@xidian.edu.cn (S. Zhang).
[1] Contributed equally.

https://doi.org/10.1016/j.ygeno.2018.11.031
Received 21 August 2018; Received in revised form 29 November 2018; Accepted 30 November 2018

Table 1
Values of original dinucleotide properties.

	AA/TT	AC/GT	AG/CT	AT	CA/TG	CC/GG	CG	GA/TC	GC	TA
F-roll	0.04	0.06	0.04	0.05	0.04	0.04	0.04	0.05	0.05	0.03
F-tilt	0.08	0.07	0.06	0.1	0.06	0.06	0.06	0.07	0.07	0.07
F-twist	0.07	0.06	0.05	0.07	0.05	0.06	0.05	0.06	0.06	0.05
F-slide	6.69	6.80	3.47	9.61	2.00	2.99	2.71	4.27	4.21	1.85
F-shift	6.24	2.91	2.80	4.66	2.88	2.67	3.02	3.58	2.66	4.11
F-rise	21.34	21.98	17.48	24.79	14.51	14.25	14.66	18.41	17.31	14.24
roll	1.05	2.01	3.60	0.61	5.60	4.68	6.02	2.44	1.70	3.50
tilt	−1.26	0.33	−1.66	0.00	0.14	−0.77	0.00	1.44	0.00	0.00
twist	35.02	31.53	32.29	30.72	35.43	33.54	33.67	35.67	34.07	36.94
slide	−0.18	−0.59	−0.22	−0.68	0.48	−0.17	0.44	−0.05	−0.19	0.04
shift	0.01	−0.02	−0.02	0.00	0.01	0.03	0.00	−0.01	0.00	0.00
rise	3.25	3.24	3.32	3.21	3.37	3.36	3.29	3.30	3.27	3.39
energy	−1.00	−1.44	−1.28	−0.88	−1.45	−1.84	−2.17	−1.30	−2.24	−0.58
enthalpy	−7.60	−8.40	−7.80	−7.20	−8.50	−8.00	−10.60	−8.20	−9.80	−7.20
entropy	−21.30	−22.40	−21.00	−20.40	−22.70	−19.90	−27.20	−22.20	−24.40	−21.30

effects in the DNA sequences. As to data reduction, we used t-distributed stochastic neighbor embedding (t-SNE) method. It is a nonlinear dimensionality reduction technique well-suited for embedding high-dimensional data for a low-dimensional space, which far outperforms PCA in this study. Finally, combined with SAE softmax classifier which performs better than SVM in classification, a predictor called iRSpot-DTS is proposed. Cross-validation tests on two benchmark datasets demonstrated that our method can well identify the recombination points and achieve the best performance in this project.

As shown in previous publications [63,64], in order to develop a useful sequence-based statistical predictor for a biological system, the following guidelines according to the Chou's 5-step rule [45–54] should be very valuable: (i) construct or select a valid benchmark dataset to train and test the predictor; (ii) formulate the biological sequence samples with an effective mathematical expression that can truly reflect their intrinsic correlation with the target to be predicted; (iii) introduce or develop a powerful algorithm (or engine) to operate the prediction; (iv) properly perform cross-validation tests to objectively evaluate the anticipated accuracy of the predictor; (v) establish a user-friendly webserver for the predictor that is accessible to the public.

2. Materials and methods

2.1. Datasets

Two datasets are used in this study to validate our method and directly compare it with other methods. For convenience, the two datasets are denoted as S1 and S2 which come from Jiang et al. [55] and Liu et al. [43]. The S1 dataset contains 490 hotspots and 591 cold spots, which can be expressed as follows:

$$S_1 = S_1^+ \cup S_1^- \tag{1}$$

where $S_1{}^+$ is the set of recombination hotspots [5], $S_1{}^-$ is the set of recombination cold spots [5] and ∪ is a mathematical operator representing union. The dataset S2 contains 478 hotspot and 572 cold spots samples.

2.2. Feature extraction

With the explosive growth of biological sequences in the post-genomic era, one of the most important but also most difficult problems in computational biology is how to express a biological sequence with a discrete model or a vector, yet still keep considerable sequence-order information or key pattern characteristic. This is because all the existing machine-learning algorithms can only handle vector but not sequence samples, as elucidated in a comprehensive review [8]. However, a vector defined in a discrete model may completely lose all the sequence-pattern information. To avoid completely losing the sequence-pattern information for proteins, the pseudo amino acid composition [9] or PseAAC [10] was proposed. Ever since the concept of Chou's PseAAC was proposed, it has been widely used in nearly all the areas of computational proteomics (see, e.g., [11–25] as well as a long list of references cited in [26]). Because it has been widely and increasingly used, recently three powerful open access soft-wares, called'PseAAC-Builder', 'propy', and'PseAAC-General', were established: the former two are for generating various modes of Chou's special PseAAC [27]; while the 3rd one for those of Chou's general PseAAC [28], including not only all the special modes of feature vectors for proteins but also the higher level feature vectors such as "Functional Domain" mode, "Gene Ontology" mode, and "Sequential Evolution" or "PSSM" mode. Encouraged by the successes of using PseAAC to deal with protein/peptide sequences, the idea of PseAAC was extended to PseKNC (Pseudo K-tuple Nucleotide Composition) to generate various feature vectors for DNA/RNA sequences that have proved very successful as well [29–34,51]. Particularly, recently a very powerful webserver called'Pse-in-One' [6] and its updated version'Pse-in-One2.0′ [7] have been established that can be used to generate any desired feature vectors for protein/peptide and DNA/RNA sequences according to the users' need or their own definition.

2.2.1. DNA property matrix

Several DNA properties have been used in recombination spot identification research, these properties includes the physical and thermodynamic parameters of dinucleotide [41,43,56], which are listed in Table 1. In this study, these properties will be normalized with the following formula:

$$\frac{X - X_{min}}{X_{max}} \tag{2}$$

where X is the original property value while X_{min} and X_{max} are the minimum and the maximum property values, respectively. In this mode, the dinucleotide sequence can be converted into a matrix $Pro = p_{(i,j)_{15\times(L-1)}}$ where L is the length of the sequence, and 15 is the total number of attributes $p_{(i,j)}$ denotes the jth property value of the ith dinucleotide pair consisting of two adjacent nucleotides in the sequence.

2.2.2. Dinucleotide-based spatial autocorrelation (DSA)

As mentioned above, global sequence order information shows strong discriminating ability to identify complex hot/cold spots. Therefore, it is essential to incorporate global sequence-order information into our model. To solve this problem, a feature extraction method called dinucleotide-based spatial autocorrelation(DSA) is proposed. DSA incorporates global sequence-order information by computing spatial autocorrelation along DNA sequences. Next, we will

introduce DSA in details. Suppose we get a DNA sequence R:

$$R = D_1D_2D_3D_4...D_L \tag{3}$$

where L is the length of DNA sequence and $D_i(i = 1, 2, \ldots, L)$ means the nucleic acid residue at different position through the sequence. Then, we use Geary's C which is a measure of spatial autocorrelation to represent the global sequence-order information in dinucleotide. It can be formulated as:

$$G_{s,t}^{lag} = \frac{1/2(L-g)(p_{i,s} - p_{i+g,s})(p_{i,t} - p_{i+g,t})}{1/(L-1)\sum_{i=1}^{L}(p_{i,s} - \overline{p}_s)(p_{i,t} - \overline{p}_t)} \tag{4}$$

where s and t is the index of dinucleotide local property. When s = t, it represents the autocorrelation of the sequence. When s ≠ t, it represents the cross-correlation of the sequence. L represents the length of DNA sequence, $p_{i,\ s}$ is the numerical value of the dinucleotide (D_iD_{i+1}) at position i for the property index s(t) and $\overline{p}_s$ ($\overline{p}_t$) is the average value of which the property value is s(t) through the sequence.

In this way, the dimension of feature vectors is N*N*g, where N is the number of dinucleotide properties used in this paper and g is the lag we chosen which reaches the peak overall accuracy of prediction accuracies on dataset. The process of generating the feature vectors of DSA is presented in Fig. 1 respectively. In this study, we used 15 DNA properties which are listed in Table 1.

2.3. Feature selection method

As g grows, the dimensions increase dramatically. If g = 100, the features would be 22,500 dimension. However, the predicted accuracy does not increase as g increases. So we experiment with the accuracy by setting different g. As shown in the Fig. 2, when g = 11, the overall accuracy reaches its peak. So we selected g = 11, thus a 2475 dimensional feature vector can be formulated as

$$FV_{2475} = [\varphi_1, \varphi_2, \ldots, \varphi_m, \ldots, \varphi_{2475}]^T \tag{5}$$

Because the number of features is huge, it is necessary to use feature selections algorithm to remove redundancy and noises to improve accuracy. Here, we used t-SNE to select the vector and compare it with PCA (Principal Component Analysis) algorithm to prove that using t-SNE is a better choice to our work.

2.3.1. T-distributed stochastic neighbor embedding (t-SNE)

T-distributed stochastic neighbor embedding (t-SNE) is a kind of nonlinear descending algorithm well-suited for embedding high-dimensional data for visualization in a low-dimensional space of two or three dimensions [57].

In recent studies, researchers explore the applicability of t-SNE to human genetic data and make these observations: (i) similar to previously used dimension reduction techniques such as principal component analysis (PCA), t-SNE is able to separate samples from different continents; (ii) unlike PCA, t-SNE is more robust with respect to the presence of outliers; (iii) t-SNE is able to display both continental and sub-continental patterns in a single plot [65]. So, we conclude that the ability for t-SNE to reveal population stratification at different scales could be useful for our study.

The basic principle of t-SNE is to map data points to probability distributions through affinity transformations, including two steps:

Step 1. Construct a probability distribution in high dimensional space where similar objects have higher probability to be selected and the dissimilar objects have lower probability to be chosen.
Step 2. Construct such a new probability distribution in low-dimensional space that it approximates the previous probability distribution.

To sum up, firstly, given an n-dimensional data $C_1, \ldots, C_N$, t-SNE computes probabilities p_{ij} that are proportional to the similarity of objects C_i and C_j:

$$p_{j|i} = \frac{exp(-\|C_i - C_j\|^2/(2\sigma^2))}{\sum_{k\neq i} exp(-\|C_i - C_j\|^2/(2\sigma^2))} \tag{6}$$

Then, in order to optimize the calculation, we use the joint probability to replace the conditional probability. The formula is as follows:

$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2N} \tag{7}$$

In low d-dimension space $y_1, \ldots, y_n$ (with $y_i \epsilon R^d$), we should use the T distribution with more emphasis on the long tail distribution and convert the distance into probability distribution. After that the low distance in the high dimension will get a larger distance after mapping. A very similar approach is used to measure similarities between two points in the map:

$$q_{ij} = \frac{(1 + \|y_i - y_j\|^2)^{-1}}{\sum_{k\neq i}(1 + \|y_i - y_j\|^2)^{-1}} \tag{8}$$

In order to achieve excellent dimensionality reduction effect, local

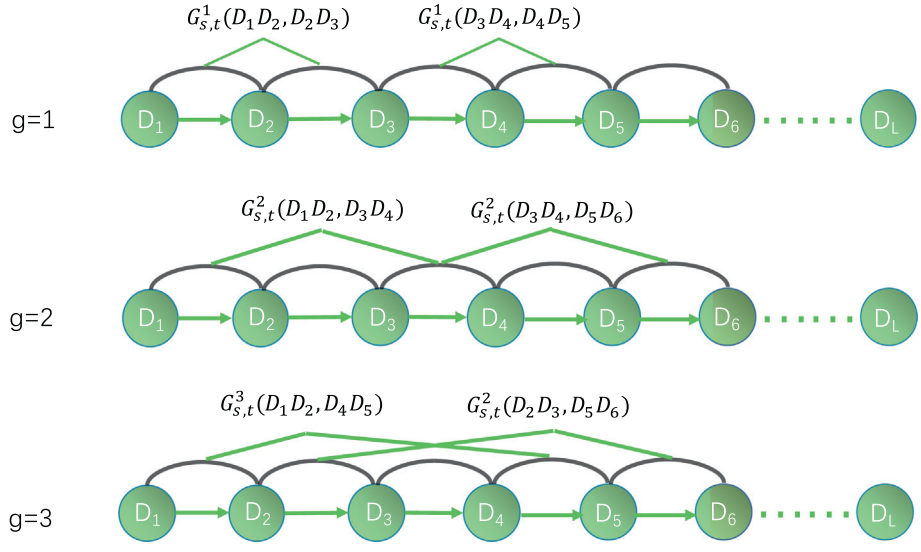

Fig. 1. A schematic illustration to show the process of generating DSA feature vector.

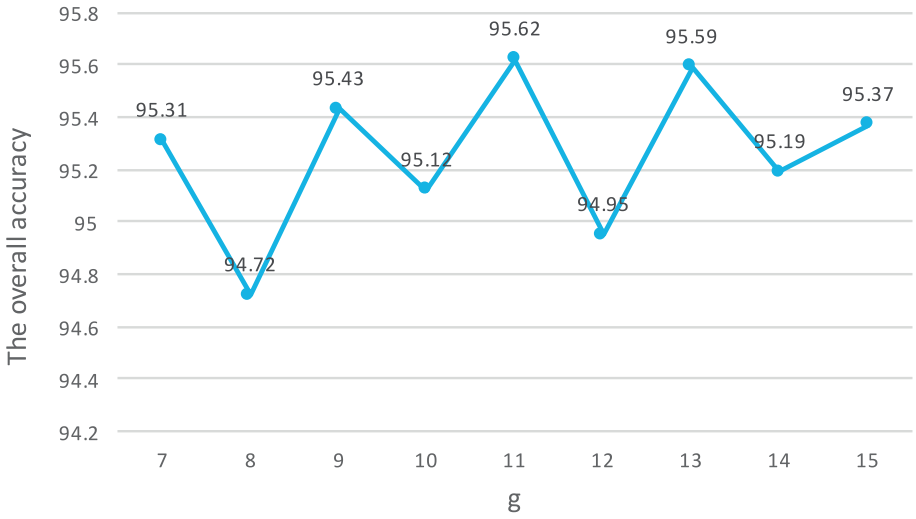

Fig. 2. The line chart shows that the overall accuracy(OA) with different g value from 7 to 15 experimented in S1 dataset.

features need to be preserved intact. So we use the Kullback-Leibler divergence as our loss function between the two distributions that is:

$$C = \sum_i KL(P_i \| Q_i) = \sum_i \sum_j p_{i,j} \log \frac{p_{ij}}{q_{ij}} \tag{9}$$

where C represents the loss function. Then we use gradient descent to minimize the loss function of the points y_i, the gradient function is formulated as:

$$\frac{\partial C}{\partial y_i} = 4 \sum_i (p_{ij} - q_{ij})(y_i - y_j)(1 + \|y_i - y_j\|^2)^{-1} \tag{10}$$

It is a map of the optimization result which reflects the similarities between the high-dimensional inputs well. We will use t-SNE in our work and test it on the dataset in section 3.

As we can see in Fig. 3, the overall accuracy achieves the highest value when the features increase to 140-dimension, the peak overall accuracy is 95.7%. Thus, the selected 140 features by the t-SNE can be formulated as:

$$FV_{140} = [\varphi_{w_1}, \varphi_{w_2}, \ldots, \varphi_{w_m}, \ldots, \varphi_{w_{140}}]^T \tag{11}$$

where $w_i (i = 1, 2, \ldots 140)$ is the $w_i th$ element of feature vector FV_{2475}.

2.4. SAE softmax classifier

The SAE softmax classifier is consist of two parts: a sparse auto encoder(SAE) and a softmax classifier. SAE is a type of artificial neural network used to learn efficient data codings in an unsupervised manner. Recently, SAE concept has become more widely used for learning generative models of data. Some of the most powerful AI in the 2010s have involved sparse auto encoders stacked inside of deep neural networks. Although the feature of the DNA sequence has been reduced using t-SNE, there are still some correlations existed in these features. SAE is introduced to excavate these relationships. It can automatically compress features from unlabeled data and can get a better feature description than the original data. So, in practice, the features found by the auto encoder can replace the original data. What's more, networks can handle a large number of data while guarantee good performance but traditional classification algorithms cannot. Using graphic approaches to study biological and medical systems can provide an intuitive vision and useful insights for helping analyze complicated relations therein, as indicated by many previous studies on a series of important biological topics, (see, e.g., [66–76]). An auto encoder is shown in Fig. 4.

The goal of the SAE network is to learn a function:

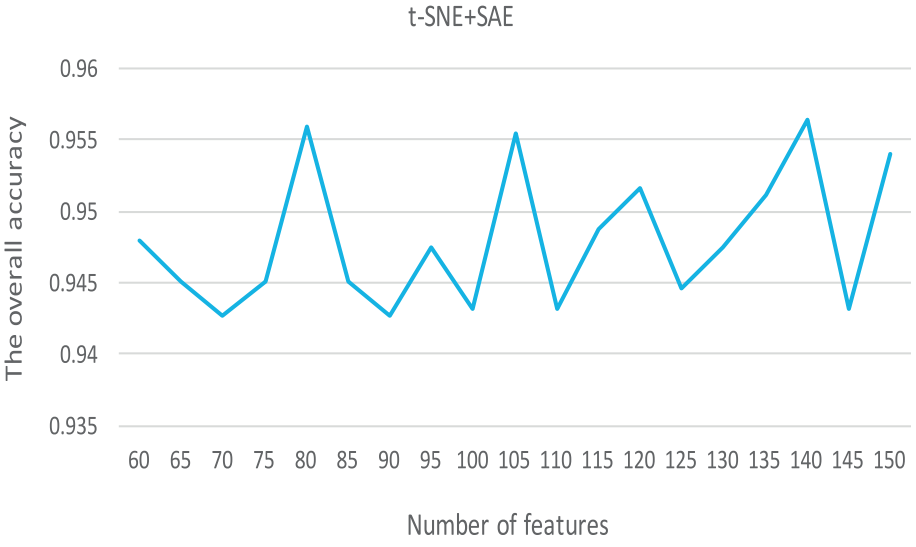

Fig. 3. The overall accuracy(OA) of t-SNE with adding different feature vectors one by five on S1 dataset.

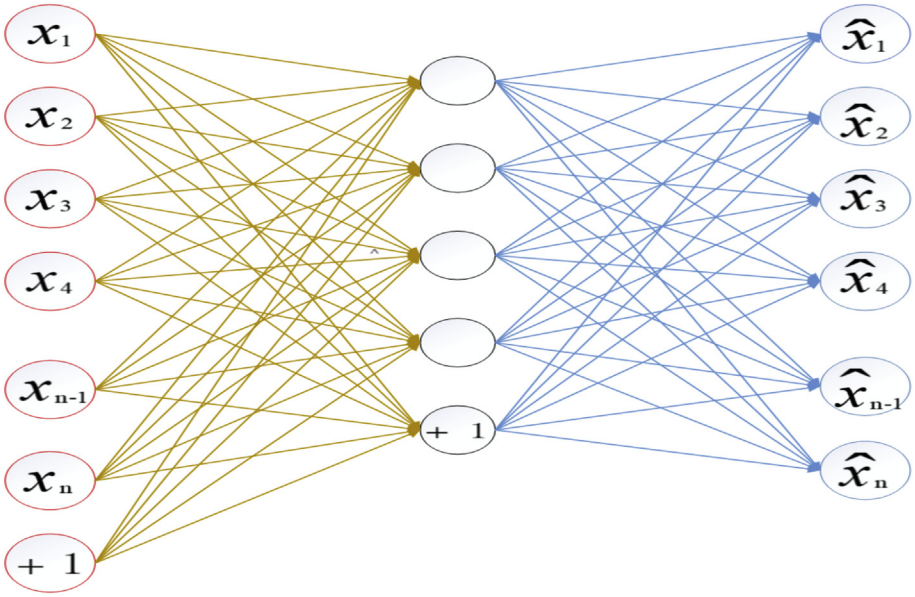

Fig. 4. Structure of the sparse auto encoder.

$$h_{W,b}(x) \approx x \tag{12}$$

In other words, it is trying to learn an approximation to the identify function, to make the output $\hat{x}$ close to the input SAE can be seen as a neural network with three layer whose inputs equals to outputs. So its feedforward is the same as neural networks, We define n_l as the number of layers, where $n_l = 3$ and s_l represents the number of neurons of layer l. SAE network contains parameters:

$$(W, b) = (W^1, b^1, W^2, b^2) \tag{13}$$

where W_{ij}^l represents the weight between the ith neurons in layer l and the jth neuron in layer l + 1. The feedforward process is calculated by the following formula:

$$\begin{aligned} z^{(2)} &= W^{(1)}x + b^{(1)} \\ a(2) &= f(z^{(2)}) \\ z^{(3)} &= W^{(2)}x + b^{(2)} \\ h_{w,b}(x) &= a^{(3)} = f(z^{(3)}) \end{aligned} \tag{14}$$

where x is the inputs, $h_{w,\ b}(x)$ is the outputs and f() represents the sigmoid function

$$f(z) = \frac{1}{1 + \exp(-z)}. \tag{15}$$

As to the back propagation, if the input is x, the output of the sparse auto encoder should be the same. So the corresponding loss function is

$$J(W, b; x) = \frac{1}{2} \|h_{W,b}(x) - x\|^2. \tag{16}$$

Suppose there are m samples, the loss function can be written as:

$$\begin{aligned} J(W, b) &= \left[\frac{1}{m}\sum_{i=1}^{m} J(W, b; x^{(i)})\right] + \frac{\lambda}{2}\sum_{l=1}^{n_l-1}\sum_{l=1}^{s_l}\sum_{j=1}^{s_l+1}(W^l{}_{ji})^2 \\ &= \left[\frac{1}{m}\sum_{i=1}^{m}\left(\frac{1}{2}\|h_{W,b}(x^i) - x^{(i)}\|^2\right)\right] + \frac{\lambda}{2}\sum_{l=1}^{n_l-1}\sum_{l=1}^{s_l}\sum_{j=1}^{s_l+1}(W^l{}_{ji})^2 \end{aligned} \tag{17}$$

Then, the gradient descent can be used to update parameters weights W and bias b. We calculate the partial derivative of W and b and update them according to the following formula:

$$\begin{aligned} \frac{\partial}{\partial W_{ij}^l}J(W, b) &= \left[\frac{1}{m}\sum_{i=1}^{m}\frac{\partial}{\partial W_{ij}^l}J(W, b; x^l)\right] + \lambda W_{ij}^l \\ \frac{\partial}{\partial b_i^l}J(W, b) &= \frac{1}{m}\sum_{i=1}^{m}\frac{\partial}{\partial b_i^l}J(W, b; x^l) \\ W_{ij}^{(l)} &= W_{ij}^l - \alpha\frac{\partial}{\partial W_{ij}^l}J(W, b) \\ b_i^{(l)} &= b_i^l - \alpha\frac{\partial}{\partial b_i^l}J(W, b) \end{aligned} \tag{18}$$

Through the back propagation algorithm, we can get parameters W^1 and b^1. With them, we can extract more meaningful features. Then, the output layer is moved out and replaced by a softmax classifier. In other words, the neuron of the hidden are used as features for the softmax classifier. In this way, we get the sparse sparse auto encoder softmax classifier. The following Fig. 5 shows the structure of the total classifier.

Here, the softmax has 2 outputs, indicating the result of a DNA sequence. We use L-BFGS to train this part. In the classifier, the number of the neurons in the hidden layer is an important parameter. We use grid search to search from 20 to 60, with step length equals to 1. And find that, when we set 57 neurons in the hidden layer, it reaches the highest accuracy over 95% which shows in Fig. 3.

2.5. Prediction assessment

One of the important procedures in developing a useful statistical predictor [60] is to objectively evaluate its performance or anticipated success rate. To examine a predictor for its effectiveness in practical application, we often use the following three methods: independent dataset test, subsampling test, and cross-validation. Among the three methods, cross-validation has been increasingly adopted or recognized by investigators to test the power of various prediction methods, cause in cross-validation, almost all samples in each round are used to train the model, so the distribution is closest to the maternal sample and estimated to be less generalized error. Accordingly, the jackknife test and 5-fold cross validation are employed to directly compare our results with the state-of-the-art methods found in the literature. To provide a more intuitive and easier-to-understand method to measure the prediction quality, the criteria proposed in [61] was adopted here. According to that criteria, the rates of correct predictions for the recombination hotspots(in data set S^+)and the recombination cold spots (in data set S^-) are respectively defined by:

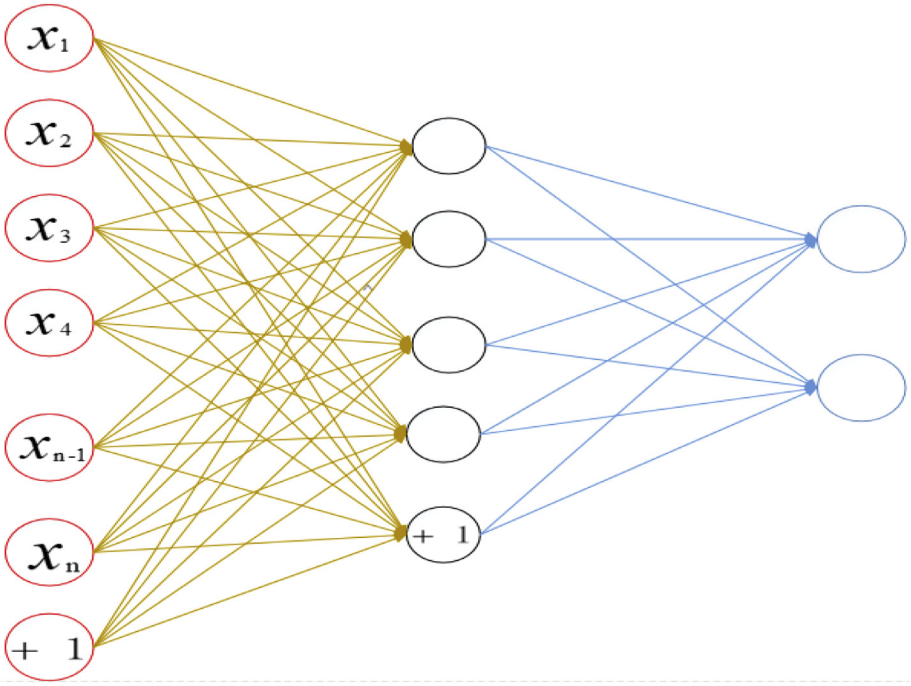

Fig. 5. Structure of total classifier.

$$\begin{cases} \Lambda^{+} = \dfrac{N^{+} - N_{-}^{+}}{N^{+}}, \text{for the recombination hotspots} \\ \Lambda^{-} = \dfrac{N^{-} - N_{+}^{-}}{N^{-}}, \text{for the recombination coldspots} \end{cases} \tag{19}$$

where N_{+} is the total number of the recombination hotspots investigated, whereas N_{-}^{+} is the number of the recombination hotspots incorrectly predicted as the cold spots; N_{-} the total number of the recombination cold spots investigated, whereas N_{+}^{-} is the number of the recombination cold spots incorrectly predicted as the hotspots. The overall success prediction rate is given by:

$$\Lambda = \frac{\Lambda^{+}N^{+} + \Lambda^{-}N^{-}}{N_{+} + N_{-}} = 1 - \frac{N_{-}^{+} + N_{+}^{-}}{N^{+} + N^{-}} \tag{20}$$

It is obvious from Equations and that, if and only if none of the recombination hotspots and the recombination cold spots are mis-predicted, i.e. $N_{-}^{+} = N_{+}^{-} = 0$ and $\Lambda^{+} = \Lambda^{-} = 1$, we have the overall success rate $\Lambda^{+} = 1$. Otherwise, the overall success rate would be smaller than 1. On the other hand, it is instructive to point out that the following equation set is often used in literatures for examining the performance quality of a predictor:

$$\begin{cases} Sn = \dfrac{TP}{TP + FN} \\ Sp = \dfrac{TN}{TN + FP} \\ Precision = \dfrac{TP}{TP + FP} \\ F = 2 \times \dfrac{Precision \times Sn}{Precision + Sn} \\ G - mean = \sqrt{Sn^{*}Sp} \\ MCC = \dfrac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \\ Acc = \dfrac{TP + TN}{TP + TN + FP + FN} \end{cases} \tag{21}$$

where *TP* represents the true positive; *TN*, the true negative; *FP*, the false positive; *FN*, the false negative; *Sn*, the sensitivity; *Sp*, the specificity; *Acc*, the accuracy; *MCC*, the Mathews correlation coefficient. The F-measure is a more robust metric avoiding overestimating the performance of some metrics, which is the harmonic mean of recall and precision. The G-mean is a measure to avoid over-fitting, which can be said to be the geometric average of correct prediction of positive and negative classes. The relations between the symbols in Eq. (17) and those in Eq. (18) are given by

$$\begin{cases} TP = N^{+} - N_{-}^{+} \\ TN = N^{-} - N_{+}^{-} \\ FP = N_{+}^{-} \\ FN = N_{-}^{+} \end{cases} \tag{22}$$

Substituting Eq. (19) into Eq. (18) and also considering Eq. (17), we obtain

$$\begin{cases} Sn = 1 - \dfrac{N_{-}^{+}}{N^{+}} \\ Sp = 1 - \dfrac{N_{+}^{-}}{N^{-}} \\ Acc = \Lambda = 1 - \dfrac{N_{+}^{-} + N_{-}^{+}}{N^{-} + N^{+}} \\ MCC = \dfrac{1 - \left(\frac{N_{-}^{+}}{N^{+}} + \frac{N_{+}^{-}}{N^{-}}\right)}{\sqrt{\left(1 + \frac{N^{+}}{N_{-}^{+} - N_{+}^{-}}\right)\left(1 + \frac{N^{-}}{N_{+}^{-} - N_{-}^{+}}\right)}} \end{cases} \tag{23}$$

We can see from the above equation that when $N_{-}^{+} = 0$, meaning none of the recombination hotspots was mis-predicted to be a cold spots, we have the sensitivity when $Sn = 1$, where $N_{-}^{+} = N^{+}$, meaning that all the recombination hotspots were mis-predicted to be cold spots, we have sensitive $Sn = 0$. Likewise, when $N_{+}^{-} = 0$, meaning none of the recombination cold spots was mis-predicted, we have the specificity $Sp = 1$, when $N_{+}^{-} = N^{-}$, meaning that all the recombination cold spots were incorrectly predicted as recombination hotspots, we have the specificity $Sp = 0$. When $N_{+}^{-} = N_{-}^{+} = 0$, meaning that none of the recombination hotspots in the S_{+} and none of the recombination cold spots in S_{-} was incorrectly predicted, we have the overall accuracy which is $Acc = \Lambda = 1$. When $N_{-}^{+} = N^{+}$ and $N_{+}^{-} = N^{-}$, meaning that all recombination hotspots in data set S^{+} + and all the recombination cold spots in S^{-} were mis-predicted, we

have the overall accuracy $Acc = \Lambda = 1$ The MCC correlation coefficient is usually used for measuring the quality of binary (two-class) classifications. When $N_{-}^{+} = N_{+}^{-} = 0$, meaning that none of the recombination hotspots in the data set S^{+} and none of the recombination cold spots in $S^{-}$was mis-predicted, we have $Mcc = 1$; When $N_{-}^{+} = N_{+}/2$ and $N_{+}^{-} = N_{-}/2$, we have $MCC = 0$ means that no better than random prediction; When $N_{-}^{+} = N^{+}$ and $N_{+}^{-} = N^{-}$ we have $MCC = -1$ meaning total disagreement between prediction and observation. As we can see from the above discussion, it is more easier to understand when using Eq. (21) to examine a predictor for its sensitivity, specificity, overall accuracy and Mathews correlation coefficient.

Although the metrics (Eq. (21)) copied from math books were often used in literature to measure the prediction quality of a prediction method, they are no longer good because of lacking intuitiveness and not easy-to-understand for most biologists. Particularly the MCC (the Matthews correlation coefficient), which is a very important metrics used for reflecting the stability of a prediction method. Fortunately, based on the Chou's symbols introduced for studying protein signal peptides [78], a set of four intuitive metrics were derived [77], as given in Eq. (23). The set of intuitive metrics have been concurred and applauded by a series of recent publications (see, e.g., [33,34,48,51,77,79–86]). However, it is instructive to point out that the metrics (Eq. (21) and Eq. (23)) are valid only for single label systems; for the multi-label systems (where a sample may simultaneously belong to several classes), whose existence has become more frequent in system biology [87–95], system medicine [96,97] and biomedicine [98], a completely different set of metrics as defined in [99] is absolutely needed.

3. Results and discussion

3.1. Comparison with other methods

In order to prove that our method has better performance than others, we compare t-SNE with PCA(Principal Component Analysis). And we compare parse auto-encoder softmax classifier with SVM (Support vector machine) and Decision Tree. The performance of PCA is shown in Fig. 6.

Fig. 6 shows that the overall accuracy achieve the peak value 93.7%. We can obviously see that the perform of t-SNE is better than PCA. Then, we compare the sparse auto-encoder softmax classifier with the SVM classifier and use them to compute the features processed by t-SNE. The results can be seen in Fig. 7.

Fig. 7 shows that the sparse auto-encoder softmax classifier's performance far outweighs SVM's performance. Given features from 60-dimensional to 140-dimensional, the overall accuracy of SVM ranges from 84.71% to 86.50%. While SAE softmax classifier has overall accuracy from 94.27% to 95.63%. SAE softmax classifier improves the accuracy by nearly 10% than SVM.

Also, decision tree classifier was used in order to compare with SAE softmax classifier. Fig. 8 illustrates that SAE softmax classifier is slightly better than decision tree whose overall accuracy ranges from 93.47% to 94.83%. What's more, it can be seen that decision tree outperforms SVM too.

It means that SAE softmax classifier can find hidden correlation of the DNA sequence and provide more useful features for the classifier. What's more, for that we put datasets S1 and S2 together, the training data become larger, this allows deep networks to take its advantage that it can handle large amounts of data. With the training data becoming bigger, SAE softmax classifier has a huge increase in accuracy. Then, we compare the sparse auto-encoder softmax classifier with Decision Tree classifier, the results can be seen in Fig. 8.

As we can see from the Tables 2 and 3, iRSpot-DTS achieved remarkable performance. It achieved highest result using jackknife test method in four criteria: OA, MCC, Sn and Sp and was the first method which achieved over 95% accuracy in this project. From Table 2 we can see that compared to the state-of-the-art method iRSpot-DACC-PCA and iRSpot-ADPM [62] based on dataset S1, our method iRspot-DTS improved over 10% in accuracy and even nearly 30% in MCC. For that we use S1 and S2 together as the dataset, we reproduced iRspot-ADPM method based on S1 and S2 too in order to compare with our iRSpot-DTS. We found that when adding S2 to the dataset, iRSpot-ADPM also had some improvements in the four criteria. However, it was still not comparable to our iRSpot-DTS, especially in MCC. The MCC of iRSpot-DTS is 13.62% higher than that of iRSpot-ADPM. And the overall accuracy of iRSpot-DTS is 7.49% higher than that of iRSpot-ADPM. This denotes that the increase of date set can improve the performance relatively. However, our method can adapt well to the increase in data set (Table 4).

Our iRSpot-DTS used a unified Geary's spatial autocorrelation formula which makes the data dimension more consistent than iRSpot-DACC which use two formula respectively for autocorrelation and cross-correlation. And Geary's spatial autocorrelation formula cam extract more space information and DNA's related information than iRspot-ADPM which used simple correlation formula. Also, iRSpot-DTS used classification algorithm based on network which can powerfully handle large amounts of data. While other methods like iRSpot-ADPM

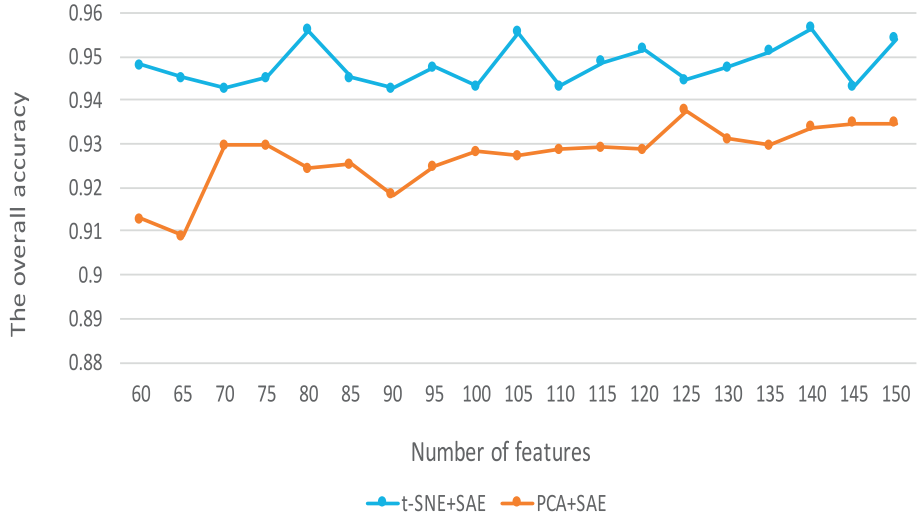

Fig. 6. Performance comparison between t-SNE and PCA on S1 + S2 datasets.

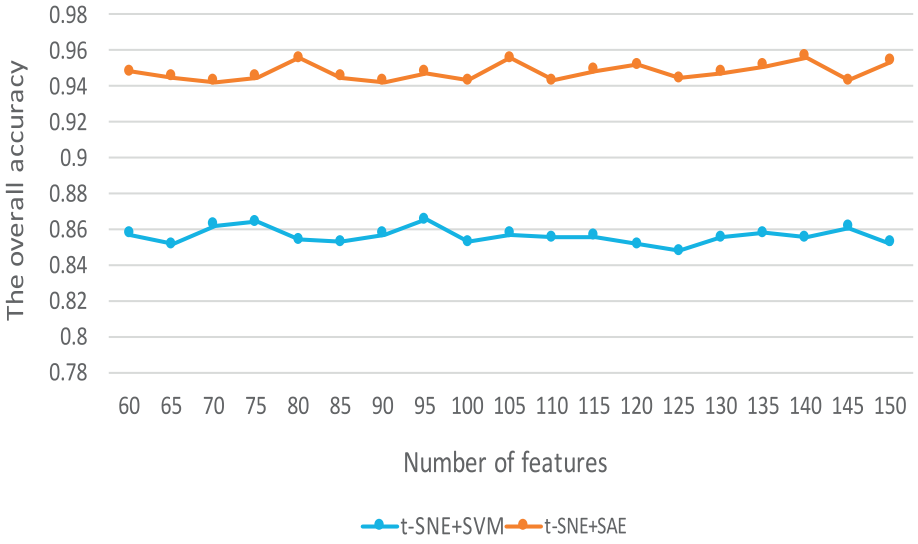

Fig. 7. Performance comparison between SAE Softmax classifier and SVM on S1 + S2 datasets.

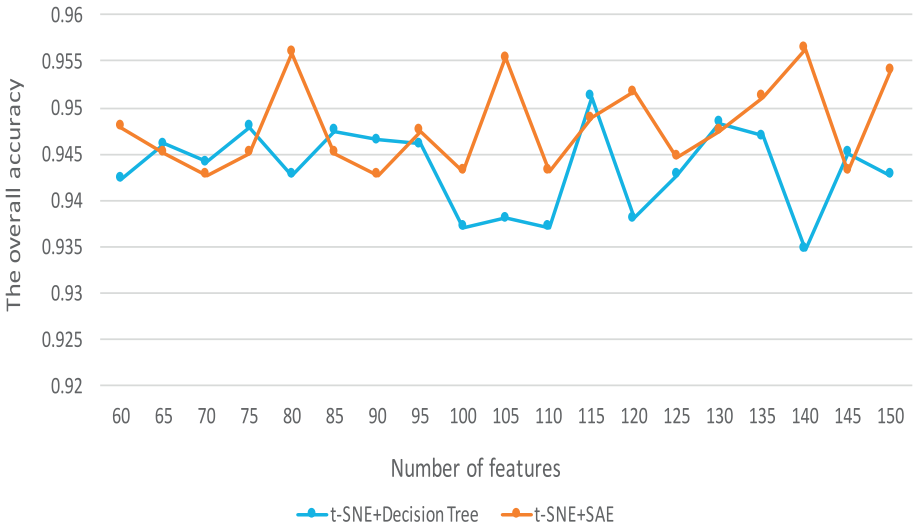

Fig. 8. Performance comparison between SAE Softmax classifier and Decision Tree on S1 + S2 datasets.

Table 2
Performance comparison of different methods on datasets by jackknife test.

Predictor	Dataset	Test method	Sn(%)	Sp(%)	MCC	Acc(%)
iRSpot-PseDNC	S1	jackknife	73.06	89.49	63.8	82.04
iRSpot-DACC-PCA	S1	jackknife	76.33	87.99	65.1	82.7
iRSpot-GAEnsC	S1	jackknife	73.77	79.92	54.0	83.44
iRSpot-ADPM	S2	jackknife	75.51	90.52	67.3	83.72
iRSpot-ADPM	S1 + S2	jackknife	81.34	93.78	79.54	89.12
iRSpot-DTS	S1 + S2	jackknife	96.06	97.06	93.16	96.61

Table 3
Performance comparison of different methods on datasets by 5-fold test.

Predictor	Dataset	Test method	Sn(%)	Sp(%)	MCC	Acc(%)
IDQD	S2	5-fold	79.52	81.82	61.6	80.77
iRSpot-EL	S2	5-fold	75.29	88.81	65.1	82.65
iRSpot-TNCPseAAC	S2	5-fold	76.56	70.99	47.37	73.52
iRSpot-PseDNC	S2	5-fold	71.75	85.84	58.3	79.33
iRSpot-ADPM	S2	5-fold	77.19	90.73	69.05	84.57
iRSpot-ADPM	S1 + S2	5-fold	81.53	94.1	81.94	90.68
iRSpot-DTS	S1 + S2	5-fold	95.15	97.58	92.92	96.48

Table 4
The overfitting test of iRSpot-DTS.

Predictor	F-measure(%)	G-mean(%)
iRSpot-DTS	96.9141	97.1402

used traditional classification methods SVM which is insensitive to data volume.

To further prove the effectiveness of iRSpot-DTS, we tested it using 5-fold method and compared it to state-of-the-art method IDQD and iRSpot-ADPM. Also, it achieved the best results in Sp, Sn, MCC and overall accuracy. Our iRSpot-DTS improved over 10% in accuracy and nearly 30% in MCC than the 2 methods. What's more, iRSpot-DTS completely surpassed iRSpot-ADPM in four criteria, improving accuracy and MCC for 5.8% and 10.98% respectively. By comparing iRSpot-DTS using jackknife and 5-fold, we can see that testing it using jackknife was slightly better using 5-fold in Sp, MCC and accuracy. This illustrated again that our approach is sensitive to data amount for that jackknife has more data during training procedure process than 5-fold.

In order to prove that our method is reliable, we tested the method with F-measure and G-measure, and obtained F-measure as 96.91%, G-

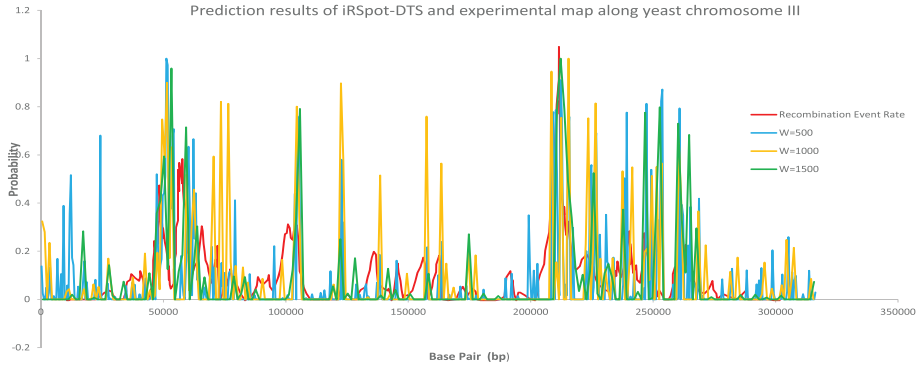

Fig. 9. Comparison between prediction results of iRSpot-DTS and experimental map along yeast chromosome III. The red line represents the recombination event rate determined experimentally by Mancera et al. [107]. The other curves represent the probability values calculated by iRSpot-DTS with different window sizes. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

measure as 97.14%. It can be seen that our method has no over-fitting, the results are valid and reliable.

3.2. Performance on analysis of the whole genome

To further demonstrate its practical application, the genome-wide analysis by iRSpot-DTS was performed on the yeast chromosome III. Chromosome III is 316,620 bp long. For investigation into the effects of different parameters on the predictive performance, the genome-wide prediction was conducted with different sliding windows. We have used three different window sizes, W = 500, W = 1000 and W = 1500 and we have found predictions for respective points as probability by our predictor iRSpot-DTS. Fig. 9 shows plot of the probabilities along with the recombination count determined experimentally by Mancera et al. [107]. The effectiveness of iRSpot-DTS can be noted from the similarity of the predictions with that of the laboratory methods. Interestingly, we have also observed that the cases with lager sliding window sizes tend to show better results. The reason is that larger window sizes can incorporate more global sequence information, which is critical for improving the performance [31,43,108].

4. Conclusions

Considering that the amount of data is very important in many areas today, we put S1 and S2 together as our data sets in order to increase data amount and we constructed models which suited well to it. First, in feature extraction, we used a unified geary spatial autocorrelation formula which makes the data dimension more consistent and reflects rich spatial information. Then, we chose t-SNE to do data reduction and compared it to PCA. We found that t-SNE is more suitable to our model and could achieve better results than PCA. Finally, we use spatial sparse auto encoder and SAE softmax classifier based on networks which has been proved to be effective in many project with large amounts of data. What's more, although the feature has been reduced through t-SNE, there are still some correlations which can be extracted by sparse sparse auto encoder. As pointed out in [100], user-friendly and publicly accessible web-servers represent the future direction for developing practically more useful predictors or any computational tools. Actually, user-friendly web-servers as given in a series of recent publications [31,81–83,101–106] will significantly enhance the impacts of theoretical work because they can attract the broad experimental scientists [8], driving the medical science into an unprecedented revolution [26]. So, we shall make efforts in our future work to provide a web-server for the prediction of DNA recombination spots.

Acknowledgements

The authors thank the anonymous reviewers of the manuscript for their helpful comments and suggestions. This work was supported by the National Natural Science Foundation of China (No. 11601407), the Key Project for the Teaching Reform and Research of Xidian University, and the Natural Science Basic Research Plan in Shaanxi Province of China (No. 2018JM1037).

Author contributions

Conceived and designed the experiments: SZ KY KS. Performed the experiments: KS KY. Analyzed the data: KY KS SZ. Contributed materials/analysis tools: KY KS YL. Wrote the paper: YL SZ. All authors read and approved the final manuscript.

References

[1] P. Paul, D. Nag, S. Chakraborty, Recombination hotspots: models and tools for detection, DNA Repair 40 (2016) 47–56.
[2] M.J. Lercher, L.D. Hurst, Human SNP variability and mutation rate are higher in regions of high recombination, Trends Ingenet. 18 (2002) 337–340.
[3] M.I. Jensen-Seaman, T.S. Furey, B.A. Payseur, Y. Lu, K.M. Roskin, C.F. Chen, M.A. Thomas, D. Haussler, H.J. Jacob, Comparative recombination rates in the rat, mouse, and human genomes, Genome Res. 14 (2004) 528–538.
[4] E. Mancera, R. Bourgon, A. Brozzi, W. Huber, L.M. Steinmetz, High-resolution mapping of meiotic crossovers and non-crossovers in yeast, Nature 454 (2008) 479–485.
[5] J.L. Gerton, J. Derisi, R. Shroff, M. Lichten, P.O. Brown, T.D. Petes, Global mapping of meiotic recombination hotspots and coldspots in the yeast Saccha- romyces cerevisiae, Proc. Natl. Acad. Sci. U. S. A. 97 (2000) 11383–11390.
[6] B. Liu, F. Liu, X. Wang, J. Chen, L. Fang, Pse-in-one: a web server for generating various modes of pseudo components of DNA, RNA, and protein sequences, Nucleic Acids Res. 43 (2015) W65–W71.
[7] B. Liu, H. Wu, Pse-in-one 2.0: an improved package of web servers for generating various modes of pseudo components of DNA, RNA, and protein sequences, Nat. Sci. 9 (2017) 67–91.
[8] K.C. Chou, Impacts of bioinformatics to medicinal chemistry, Med. Chem. 11 (2015) 218–234.
[9] K.C. Chou, Prediction of protein cellular attributes using pseudo amino acid composition, Proteins: Struct. Funct. Genet. 43 (2001) 246–255 Erratum: ibid., 2001, Vol. 44, 60.
[10] K.C. Chou, Using amphiphilic pseudo amino acid composition to predict enzyme subfamily classes, Bioinformatics 21 (2005) 10–19.
[11] S. Akbar, M. Hayat, iMethyl-STTNC: Identification of N(6)-methyladenosine sites by extending the Idea of SAAC into Chou's PseAAC to formulate RNA sequences, J. Theor. Biol. 455 (2018) 205–211.
[12] M. Arif, M. Hayat, Z. Jan, iMem-2LSAAC: a two-level model for discrimination of membrane proteins and their types by extending the notion of SAAC into Chou's pseudo amino acid composition, J. Theor. Biol. 442 (2018) 11–21.
[13] E. Contreras-Torres, Predicting structural classes of proteins by incorporating their global and local physicochemical and conformational properties into general Chou's PseAAC, J. Theor. Biol. 454 (2018) 139–145.
[14] M.S. Krishnan, Using Chou's general PseAAC to analyze the evolutionary relationship of receptor associated proteins (RAP) with various folding patterns of protein domains, J. Theor. Biol. 445 (2018) 62–74.
[15] Y. Liang, S. Zhang, Identify Gram-negative bacterial secreted protein types by incorporating different modes of PSSM into Chou's general PseAAC via Kullback-Leibler divergence, J. Theor. Biol. 454 (2018) 22–29.
[16] J. Mei, Y. Fu, J. Zhao, Analysis and prediction of ion channel inhibitors by using feature selection and Chou's general pseudo amino acid composition, J. Theor. Biol. 456 (2018) 41–48.
[17] J. Mei, J. Zhao, Analysis and prediction of presynaptic and postsynaptic neurotoxins by Chou's general pseudo amino acid composition and motif features, J. Theor. Biol. 427 (2018) 147–153.
[18] W. Qiu, S. Li, X. Cui, Z. Yu, M. Wang, J. Du, Y. Peng, B. Yu, Predicting protein submitochondrial locations by incorporating the pseudo-position specific scoring matrix into the general Chou's pseudo-amino acid composition, J. Theor. Biol. 450

(2018) 86–103.
[19] S.M. Rahman, S. Shatabda, S. Saha, M. Kaykobad, M. Sohel Rahman, DPP-PseAAC: a DNA-binding protein prediction model using Chou's general PseAAC, J. Theor. Biol. 452 (2018) 22–34.
[20] M.F. Sabooh, N. Iqbal, M. Khan, M. Khan, H.F. Maqbool, Identifying 5-methylcytosine sites in RNA sequence using composite encoding feature into Chou's PseKNC, J. Theor. Biol. 452 (2018) 1–9.
[21] E.S. Sankari, D.D. Manimegalai, Predicting membrane protein types by incorporating a novel feature set into Chou's general PseAAC, J. Theor. Biol. 455 (2018) 319–328.
[22] A. Srivastava, R. Kumar, M. Kumar, BlaPred: predicting and classifying beta-lactamase using a 3-tier prediction system via Chou's general PseAAC, J. Theor. Biol. (2018), https://doi.org/10.1016/j.jtbi.2018.08.030.
[23] L. Zhang, L. Kong, iRSpot-ADPM: Identify recombination spots by incorporating the associated dinucleotide product model into Chou's pseudo components, J. Theor. Biol. 441 (2018) 1–8.
[24] S. Zhang, X. Duan, Prediction of protein subcellular localization with oversampling approach and Chou's general PseAAC, J. Theor. Biol. 437 (2018) 239–250.
[25] J. Mei, J. Zhao, Prediction of HIV-1 and HIV-2 proteins by using Chou's pseudo amino acid compositions and different classifiers, Sci. Rep. 8 (2018) 2359.
[26] K.C. Chou, An unprecedented revolution in medicinal chemistry driven by the progress of biological science, Curr. Top. Med. Chem. 17 (2017) 2337–2358.
[27] K.C. Chou, Pseudo amino acid composition and its applications in bioinformatics, proteomics and system biology, Curr. Proteom. 6 (2009) 262–274.
[28] K.C. Chou, Some remarks on protein attribute prediction and pseudo amino acid composition, 50th anniversary Year Review, J. Theor. Biol. 273 (2011) 236–247.
[29] W. Chen, H. Lin, Pseudo nucleotide composition or PseKNC: an effective formulation for analyzing genomic sequences, Mol. BioSyst. 11 (2015) 2620–2634.
[30] W. Chen, H. Tang, J. Ye, H. Lin, iRNA-PseU: identifying RNA pseudouridine sites, Mol. Ther. – Nucl. Acid. 5 (2016) e332.
[31] B. Liu, L. Fang, R. Long, X. Lan, iEnhancer-2L: a two-layer predictor for identifying enhancers and their strength by pseudo k-tuple nucleotide composition, Bioinformatics 32 (2016) 362–369.
[32] B. Liu, R. Long, iDHS-EL: Identifying DNase I hypersensi-tivesites by fusing three different modes of pseudo nucleotide composition into an ensemble learning framework, Bioinformatics 32 (2016) 2411–2418.
[33] B. Liu, S. Wang, R. Long, iRSpot-EL: identify recombination spots with an ensemble learning approach, Bioinformatics 33 (2017) 35–41.
[34] B. Liu, F. Yang, 2L-piRNA: a two-layer ensemble classifier for identifying piwi-interacting RNAs and their function, Mol. Ther. – Nucl. Acids 7 (2017) 267–277.
[35] P. Jiang, H. Wu, J. Wei, F. Sang, X. Sun, Z. Lu, RF-DYMHC: detecting the yeast meiotic recombination hotspots and coldspots by random forest model using gapped dinucleotide composition features, Nucleic Acids Res. 35 (2007) W47–W51.
[36] T. Zhou, J. Weng, X. Sun, Z. Lu, Support vector machine for classi cation of meiotic recombination hotspots and coldspots inSaccharomyces cerevisiae based on codon composition, BMC Bioinfo. 7 (223) (2006).
[37] M. Kabir, M. Hayat, iRSpot-GAEnsc: identifying recombination spots via ensemble classifier and extending the concept of Chou's PseAAC to formulate DNA samples, Mol. Gen. Genomics. 291 (2016) 285–296.
[38] G. Liu, J. Liu, X. Cui, L. Cai, Sequence-dependent prediction of recombination hotspots in Saccharomyces cerevisiae, J. Theor. Biol. 293 (2012) 49–54.
[39] W.R. Qiu, X. Xiao, K.C. Chou, iRSpot-TNCPseAAC: identify recombination spots with trinucleotide composition and pseudo amino acid components, Int. J. Mol. Sci. 15 (2) (2014) 1746–1766.
[40] A.A. Maruf, S. Shatabda, iRSpot-SF: prediction of recombination hotspots by incorporating sequence based features into Chou's pseudo components, Genomics (2018), https://doi.org/10.1016/j.ygeno.2018.06.003.
[41] W. Chen, P.M. Feng, H. Lin, K.C. Chou, iRSpot-PseDNC: identify recombination spots with pseudo dinucleotide composition, Nucleic Acids Res. 41 (6) (2013) e68.
[42] B. Liu, Y. Liu, X. Jin, X. Wang, B. Liu, iRSpot-DACC: a computational predictor for recombination hot/cold spots identification based on dinucleotide-based auto-cross covariance, Sci. Rep. 6 (2016) 33483.
[43] B. Liu, S. Wang, R. Long, K.C. Chou, iRSpot-EL: identify recombination spots with an ensemble learning approach, Bioinformatics 33 (1) (2017) 35–41.
[44] W.R. Qiu, X. Xiao, K.C. Chou, iRSpot-TNCPseAAC: identify recombination spots with trinucleotide composition and pseudo amino acid components, Int. J. Mol. Sci. 15 (2) (2014) 1746–1766.
[45] K.C. Chou, Some remarks on protein attribute prediction and pseudo amino acid composition, 50th anniversary Year Review, J. Theor. Biol. 273 (2011) 236–247.
[46] P.M. Feng, W. Chen, H. Lin, iHSP-PseRAAAC: Identifying the heat shock protein families using pseudo reduced amino acid alphabet composition, Anal. Biochem. 442 (2013) 118–125.
[47] W. Chen, P.M. Feng, H. Lin, iSS-PseDNC: identifying splicing sites using pseudo dinucleotide composition, Biomed. Res. Int. (BMRI) (2014) (2014) 623149.
[48] H. Lin, E.Z. Deng, H. Ding, W. Chen, iPro54-PseKNC: a sequence-based predictor for identifying sigma-54 promoters in prokaryote with pseudo k-tuple nucleotide composition, Nucleic Acids Res. 42 (2014) 12961–12972.
[49] J. Jia, Z. Liu, X. Xiao, iPPI-Esml: an ensemble classifier for identifying the interactions of proteins by incorporating their physicochemical properties and wavelet transforms into PseAAC, J. Theor. Biol. 377 (2015) 47–56.
[50] J. Jia, Z. Liu, X. Xiao, B. Liu, iSuc-PseOpt: identifying lysine succinylation sites in proteins by incorporating sequence-coupling effects into pseudo components and optimizing imbalanced training dataset, Anal. Biochem. 497 (2016) 48–56.
[51] P. Feng, H. Ding, H. Yang, W. Chen, H. Lin, iRNA-PseColl: Identifying the occurrence sites of different RNA modifications by incorporating collective effects of nucleotides into PseKNC, Mol. Ther. – Nucl. Acids 7 (2017) 155–163.
[52] L. Cai, T. Huang, J. Su, X. Zhang, W. Chen, F. Zhang, L. He, Implications of newly identified brain eQTL genes and their interactors in Schizophrenia, Mol. Ther. – Nucl. Acids 12 (2018) 433–442.
[53] H. Yang, W.R. Qiu, G. Liu, F.B. Guo, W. Chen, H. Lin, iRSpot-Pse6NC: Identifying recombination spots in Saccharomyces cerevisiae by incorporating hexamer composition into general PseKNC, Int. J. Biol. Sci. 14 (2018) 883–891.
[54] W.R. Qiu, B.Q. Sun, X. Xiao, Z.C. Xu, J.H. Jia, iKcr-PseEns: Identify lysine crotonylation sites in histone proteins with pseudo components and ensemble classifier, Genomics 110 (2018) 239–246.
[55] P. Jiang, H. Wu, J. Wei, F. Sang, X. Sun, Z. Lu, RF-DYMHC: detecting the yeast meiotic recombination hotspots and coldspots by random Forest model using gapped dinucleotide composition features, Nucleic Acids Res. 35 (2007) W47–W51.
[56] W. Chen, H. Lin, P.M. Feng, C. Ding, Y.C. Zuo, K.C. Chou, iNuc-PhysChem: a sequence-based predictor for identifying nucleosomes via physicochemical properties, PLoS One 7 (2012) e47843.
[57] Laurens van der Maaten, Geoffrey Hinton, Visualizing Data using t-SNE, J. Mach. Learn. Res. 9 (2008) 2579–2605.
[60] K.C. Chou, Some remarks on protein attribute prediction and pseudo amino acid composition, 50th anniversary Year Review, J. Theor. Biol. 273 (2011) 236–247.
[61] K.C. Chou, Using subsite coupling to predict signal peptides, Protein Eng. 14 (2001) 75–79.
[62] L.C. Zhang, iRSpot-ADPM: identify recombination spots by incorporating the associated dinucleotide product model into Chou's pseudo components, J. Theor. Biol. 441 (2018) (2017) 1–8.
[63] X. Cheng, X. Xiao, pLoc-mGneg: predict subcellular localization of gram- negative bacterial proteins by deep gene ontology learning via general PseAAC, Genomics (2017), https://doi.org/10.1016/j.ygeno.2017.10.002.
[64] X. Cheng, X. Xiao, pLoc-mEuk: predict subcellular localization of multi-label eukaryotic proteins by extracting the key GO information into general PseAAC, Genomics (2017), https://doi.org/10.1016/j.ygeno.2017.08.005.
[65] W. Li, J.E. Cerise, Y. Yang, et al., Application of t-SNE to human genetic data[J], J. Bioinforma. Comput. Biol. 15 (04) (2017) 1750017.
[66] S.P. Jiang, W.M. Liu, C.H. Fee, Graph theory of enzyme kinetics: 1. Steady-state reaction system, Sci. Sinica 22 (1979) 341–358.
[67] S. Forsen, Graphical rules for enzyme-catalyzed rate laws, Biochem. J. 187 (1980) 829–835.
[68] G.P. Zhou, M.H. Deng, An extension of Chou's graphic rules for deriving enzyme kinetic equations to systems involving parallel reaction pathways, Biochem. J. 222 (1984) 169–176.
[69] K.C. Chou, Graphic rules in steady and non-steady enzyme kinetics, J. Biol. Chem. 264 (1989) 12074–12079.
[70] I.W. Althaus, J.J. Chou, A.J. Gonzales, M.R. Diebel, F.J. Kezdy, D.L. Romero, P.A. Aristoff, W.G. Tarpley, F. Reusser, Steady-state kinetic studies with the non-nucleoside HIV-1 reverse transcriptase inhibitor U-87201E, J. Biol. Chem. 268 (1993) 6119–6124.
[71] K.C. Chou, Review: applications of graph theory to enzyme kinetics and protein folding kinetics. Steady and non-steady state systems, Biophys. Chem. 35 (1990) 1–24.
[72] I.W. Althaus, A.J. Gonzales, J.J. Chou, M.R. Diebel, F.J. Kezdy, D.L. Romero, P.A. Aristoff, W.G. Tarpley, F. Reusser, The quinoline U-78036 is a potent inhibitor of HIV-1 reverse transcriptase, J. Biol. Chem. 268 (1993) 14875–14880.
[73] K.C. Chou, Graphic rule for drug metabolism systems, Curr. Drug Metab. 11 (2010) 369–378.
[74] G.P. Zhou, The disposition of the LZCC protein residues in Wenxiang diagram provides new insights into the protein-protein interaction mechanism, J. Theor. Biol. 284 (2011) 142–148.
[75] I.W. Althaus, J.J. Chou, A.J. Gonzales, M.R. Diebel, F.J. Kezdy, D.L. Romero, P.A. Aristoff, W.G. Tarpley, F. Reusser, Kinetic studies with the nonnucleoside HIV-1 reverse transcriptase inhibitor U-88204E, Biochemistry 32 (1993) 6548–6554.
[76] W.Z. Lin, X. Xiao, Wenxiang: a web-server for drawing Wenxiang diagrams, Nat. Sci. 3 (2011) 862–865.
[77] W. Chen, P.M. Feng, H. Lin, iRSpot-PseDNC: identify recombination spots with pseudo dinucleotide composition, Nucleic Acids Res. 41 (2013) e68.
[78] K.C. Chou, Prediction of signal peptides using scaled window, Peptides 22 (2001) 1973–1979.
[79] Y. Xu, X. Wen, L.S. Wen, L.Y. Wu, N.Y. Deng, iNitro-Tyr: Prediction of nitrotyrosine sites in proteins with general pseudo amino acid composition, PLoS One 9 (2014) e105018.
[80] J. Jia, Z. Liu, X. Xiao, B. Liu, pSuc-Lys: predict lysine succinylation sites in proteins with PseAAC and ensemble random forest approach, J. Theor. Biol. 394 (2016) 223–230.
[81] C.J. Zhang, H. Tang, W.C. Li, H. Lin, W. Chen, iOri-human: identify human origin of replication by incorporating dinucleotide physicochemical properties into pseudo nucleotide composition, Oncotarget 7 (2016) 69783–69793.
[82] W. Chen, H. Ding, P. Feng, H. Lin, iACP: a sequence-based tool for identifying anticancer peptides, Oncotarget 7 (2016) 16895–16909.
[83] W. Chen, P. Feng, H. Yang, H. Ding, H. Lin, iRNA-AI: identifying the adenosine to inosine editing sites in RNA sequences, Oncotarget 8 (2017) 4208–4217.
[84] B. Liu, F. Yang, D.S. Huang, iPromoter-2L: a two-layer predictor for identifying promoters and their types by multi-window-based PseKNC, Bioinformatics 34 (2018) 33–40.
[85] A. Ehsan, K. Mahmood, Y.D. Khan, S.A. Khan, A novel modeling in mathematical

biology for classification of signal peptides, Sci. Rep. 8 (2018) 1039.
[86] P. Feng, H. Yang, H. Ding, H. Lin, W. Chen, iDNA6mA-PseKNC: Identifying DNA N6-methyladenosine sites by incorporating nucleotide physicochemical properties into PseKNC, Genomics (2018), https://doi.org/10.1016/j.ygeno.2018.01.005.
[87] X. Cheng, X. Xiao, pLoc-mPlant: predict subcellular localization of multi-location plant proteins via incorporating the optimal GO information into general PseAAC, Mol. BioSyst. 13 (2017) 1722–1727.
[88] X. Xuao, X. Cheng, G. Chen, Q. Mao, pLoc-bal-mGpos: predict subcellular localization of Gram-positive bacterial proteins by quasi-balancing training dataset and PseAAC, Genomics (2018), https://doi.org/10.1016/j.ygeno.2018.05.017.
[89] X. Cheng, X. Xiao, pLoc-mVirus: predict subcellular localization of multi-location virus proteins via incorporating the optimal GO information into general PseAAC, Gene 628 (2017) 315–321 Erratum: ibid., 2018, Vol. 644, 156–156.
[90] X. Cheng, S.G. Zhao, W.Z. Lin, X. Xiao, pLoc-mAnimal: predict subcellular localization of animal proteins with both single and multiple sites, Bioinformatics 33 (2017) 3524–3531.
[91] X. Xiao, X. Cheng, S. Su, Q. Nao, pLoc-mGpos: Incorporate key gene ontology information into general PseAAC for predicting subcellular localization of gram-positive bacterial proteins, Nat. Sci. 9 (2017) 331–349.
[92] X. Cheng, X. Xiao, pLoc-mGneg: predict subcellular localization of gram-negative bacterial proteins by deep gene ontology learning via general PseAAC, Genomics 110 (2018) 231–239.
[93] X. Cheng, X. Xiao, pLoc-mEuk: Predict subcellular localization of multi-label eukaryotic proteins by extracting the key GO information into general PseAAC, Genomics 110 (2018) 50–58.
[94] X. Cheng, W.Z. Lin, X. Xiao, pLoc-bal-mAnimal: predict subcellular localization of animal proteins by balancing training dataset and PseAAC, Bioinformatics (2018), https://doi.org/10.1093/bioinformatics/bty628.
[95] K.C. Chou, X. Cheng, X. Xiao, pLoc-bal-mHum: predict subcellular localization of human proteins by PseAAC and quasi-balancing training dataset, Genomics (2018), https://doi.org/10.1016/j.ygeno.2018.08.007.
[96] X. Cheng, S.G. Zhao, X. Xiao, iATC-mISF: a multi-label classifier for predicting the classes of anatomical therapeutic chemicals, Bioinformatics 33 (2017) 341–346 Corrigendum, ibid., 2017, Vol.33, 2610.
[97] X. Cheng, S.G. Zhao, X. Xiao, iATC-mHyb: a hybrid multi-label classifier for predicting the classification of anatomical therapeutic chemicals, Oncotarget 8 (2017) 58494–58503.
[98] W.R. Qiu, B.Q. Sun, X. Xiao, Z.C. Xu, iPTM-mLys: identifying multiple lysine PTM sites and their different types, Bioinformatics 32 (2016) 3116–3123.
[99] K.C. Chou, Some remarks on predicting multi-label attributes in molecular biosystems, Mol. Biosyst. 9 (2013) 1092–1100.
[100] H.B. Shen, Recent advances in developing web-servers for predicting protein attributes, Nat. Sci. 1 (2009) 63–92.
[101] B. Liu, L. Fang, F. Liu, X. Wang, J. Chen, Identification of real microRNA precursors with a pseudo structure status composition approach, PLoS One 10 (2015) e0121501.
[102] J. Jia, Z. Liu, X. Xiao, B. Liu, iCar-PseCp: identify carbonylation sites in proteins by Monto Carlo sampling and incorporating sequence coupled effects into general PseAAC, Oncotarget 7 (2016) 34558–34570.
[103] B. Liu, H. Wu, D. Zhang, X. Wang, Pse-Analysis: a python package for DNA/RNA and protein/peptide sequence analysis based on pseudo components and kernel methods, Oncotarget 8 (2017) 13338–13343.
[104] W.R. Qiu, B.Q. Sun, X. Xiao, Z.C. Xu, iHyd-PseCp: identify hydroxyproline and hydroxylysine in proteins by incorporating sequence-coupled effects into general PseAAC, Oncotarget 7 (2016) 44310–44321.
[105] W.R. Qiu, X. Xiao, Z.C. Xu, iPhos-PseEn: identifying phosphorylation sites in proteins by fusing different pseudo components into an ensemble classifier, Oncotarget 7 (2016) 51270–51283.
[106] X. Xiao, H.X. Ye, Z. Liu, J.H. Jia, iROS-gPseKNC: predicting replication origin sites in DNA by incorporating dinucleotide position-specific propensity into general pseudo nucleotide composition, Oncotarget 7 (2016) 34180–34189.
[107] E. Mancera, et al., High-resolution mapping of meiotic crossovers and non-crossovers in yeast, Nature 454 (2008) 479–485.
[108] M.R. Jani, M.T.K. Mozlish, S. Ahmed, N.S. Tahniat, D.M. Farid, S. Shatabda, iRecSpot-EF: effective sequence based features for recombination hotspot prediction, Comput. Biol. Med. 103 (2018) 17–23.