Briefings in Bioinformatics, 22(5), 2021, 1–11

https://doi.org/10.1093/bib/bbaa393
Problem Solving Protocol

Identification of haploinsufficient genes from epigenomic data using deep forest

Yuning Yang, Shaochuan Li, Yunhe Wang, Zhiqiang Ma, Ka-Chun Wong and Xiangtao Li

Corresponding author: Xiangtao Li, School of Artificial Intelligence, Jilin University, Changchun, Jilin, China. Tel: 0431-85168703. Email: lixt314@jlu.edu.cn

Abstract

Haploinsufficiency, wherein a single allele is not enough to maintain normal functions, can lead to many diseases including cancers and neurodevelopmental disorders. Recently, computational methods for identifying haploinsufficiency have been developed. However, most of those computational methods suffer from study bias, experimental noise and instability, resulting in unsatisfactory identification of haploinsufficient genes. To address those challenges, we propose a deep forest model, called HaForest, to identify haploinsufficient genes. The multiscale scanning is proposed to extract local contextual representations from input features under Linear Discriminant Analysis. After that, the cascade forest structure is applied to obtain the concatenated features directly by integrating decision-tree-based forests. Meanwhile, to exploit the complex dependency structure among haploinsufficient genes, the LightGBM library is embedded into HaForest to reveal the highly expressive features. To validate the effectiveness of our method, we compared it to several computational methods and four deep learning algorithms on five epigenomic data sets. The results reveal that HaForest achieves superior performance over the other algorithms, demonstrating its unique and complementary performance in identifying haploinsufficient genes. The standalone tool is available at https://github.com/yangyn533/HaForest.

Key words: deep forest; epigenomic data; haploinsufficiency; HaForest

Introduction

The most obvious mechanism for loss-of-function tolerance (LoFT) in heterozygosity is haploinsufficiency (HIS) [1]. The abnormal phenotypes due to the reduction in alleles or transcriptional activity are in normal diploid genes [2]. For some genes, a single functional copy (i.e. allele) is not sufficient to maintain regular human function and may induce death at the early stage of development, thereby leading to haplolethality and developmental diseases [3]. However, in whole-genome

Yuning Yang is with the School of Artificial Intelligence, Jilin University and School of Information Science and Technology, Northeast Normal University. His research interests include bioinformatics, machine learning and computational intelligence.

Shaochuan Li is a Master student in School of Information Science and Technology, Northeast Normal University. His research interests include bioinformatics, machine learning and computational intelligence.

Yunhe Wang is a PhD student in School of Information Science and Technology, Northeast Normal University. Her research interests include intelligent computation and machine learning.

Zhiqiang Ma is a Professor in the School of Information Science and Technology, Northeast Normal University. He is the vice president of the research association of Computer Education in the Normal Universities of China and the executive director of the Jilin Computer Federation. His interests include bioinformatics, software engineering, molecular biology, and data mining.

Ka-Chun Wong assumed his duty as an assistant professor at City University of Hong Kong in 2015. His research interests include bioinformatics, computational biology, evolutionary computation, data mining, machine learning, and interdisciplinary research. He is merited as the associate editor of BioData Mining. In addition, he is on the editorial board of Applied Soft Computing.

Xiangtao Li is a Professor in the School of Artificial Intelligence, Jilin University. He has published more than 60 research papers. His research interests include bioinformatics, computational biology, and evolutionary data mining. E-mail: lixt314@jlu.edu.cn. (corresponding author)

Submitted: 19 September 2020; **Received (in revised form):** 29 November 2020

sequencing data, individual can harbor nonsense and missense mutations [4]; this provides inadequate statistical information to distinguish haploinsufficient (HIS) genes from those with random mutations. Therefore, accurate computational methods for identifying HIS genes in genome data are crucial.

Over the years, a series of machine-learning approaches have been developed for identifying HIS genes based on high-throughput data. The data are combined with extra features of the genetic, evolutionary and functional properties to estimate HIS probabilities of genes with the LDA classifier [5]. However, such methodology was demonstrated in [6] to be strongly biased toward well-studied genes but performing poorly on less-studied annotated genes [6]. Indeed, Steinberg *et al.* used similar input information as [5] to construct a coexpression network with an unbiased genome-wide haploinsufficiency score (GHIS) using a support vector machine (SVM) model, thus solving the influence of study bias. Although both methods achieved good performance in distinguishing obvious characteristic differences between HIS and haplosufficient (HS) genes, other potentially informative sources of chemical modification of epigenetic marks and functional annotation include the NIH Roadmap Epigenomics [7] as well as the Encyclopedia of DNA Elements) projects [8]. The HIPred [4] has developed genomic and evolutionary features based on ensemble learning. Recently, Episcore, a computational method [9] published in *Nature Communications*, was strongly associated with epigenomic patterns, whereas the genomic data could enhance the discriminating ability and without bias under well-studied genes. However, Episcore still has limitations in the mission of using biological data for HIS genes identification. On the one hand, the characteristics of inherently small sample sizes and the high-dimensionality of genetic data heighten the risk of overfitting in the training processes, which leads to a weak generalization ability of models. On the other hand, the large prioritizing bias of biological data would weaken the capacity of model estimation. Therefore, alternative more accurate and robust methods need to be developed to determine HIS genes, which can contribute to interpretation of pathological mechanisms of sequenced mutations.

To address those problems, in this study, we propose a method inspired by deep forest [10], an alternative to the deep learning framework, called HaForest. The algorithm consists of two components, a multiscale scanning module and a cascade forest structure. Compared to deep neural networks, HaForest combines machine learning algorithms and deep learning ideas of multilayer learning, originating from its representation learning ability. Moreover, we compare our model with various machine learning algorithms and four deep learning algorithms to demonstrate the stability and identification performance of our architecture. Further, the LightGBM library is employed to find the highly expressive features from epigenomic data. Finally, we benchmark our model using several datasets consisting of known and candidate disease genes to demonstrate the generalizability of HaForest.

Materials and methods

Datasets

The datasets were collected from reference [9]. Among them, the positive dataset (curated HIS genes) came from known haploinsufficient genes of previous studies and the human-curated ClinGen dosage sensitivity map [1, 5]. For the negative dataset (curated HS genes), the genes were selected from a copy-number variation study of 2026 normal individuals [11]. After that, we retained 287 HIS genes and 574 HS genes. Figure. 1 summarizes the enrichment analysis of the HIS gene set. Figure. 1a and b represent the enrichment of gene ontology (GO) terms [12] and the Kyoto Encyclopedia of Genes and Genomes pathways [13], which show that these genes are closely related to many human diseases and biological processes, especially cancers. Indeed, the phenomenon of HIS may lead to tumorigenesis because the expression level of anticancer genes is very important [1]. To further capture the relationships between the terms, the network of enriched terms and protein–protein interaction enrichment analysis are plotted by Metascape [14], and are depicted in Figure 1c and d. In particular, the network in Figure 1c constructed by functional relevance and similarity includes HIS and HS genes. The redundant terms within a cluster are naturally due to the high intracluster similarities, and clusters are occasionally bridged, reflecting the relatedness of two separate processes; this indicates that this type of data identification is extremely difficult due to the correlation of different genes[14].

To compare the generalizability of our method, we constructed experiments on four test sets to evaluate performance. The nonredundant testing data were sourced from HIPred [4], including a set of genes to be associated with human disease, i.e. 156 related genes in OMIM HI and 92 related genes in OMIM HI *de novo* from the paper of Petrovskiet *al.* [15]. There are two independent sets of *de novo* LoF genes related to autism: ASD1 [16] with 102 genes; and ASD2 [4] with 113 genes. For the sake of fairness, we assess the performance based on 10-fold cross-validation in training and testing processes.

Epigenomic features

Haploinsufficient gene-specific epigenomic patterns are strictly controlled by a combination of transcription factors and epigenomic modifications, which can effectively identify HIS and HS genes [17, 18]. The epigenomic data can be constructed from the following feature groups [9]:

- *Promoter Features*: including active (H3K4me3, H3K9ac and H2A.Z) and repressive (H3K27me3) promoter modifications.
- *Enhancer Features*: marks associated with enhancers (H3K4me1, H3K27ac, DNase I hypersensitivity sites).

For promoter features, 'GappedPeaks' based on broad domains of the ChIP-seq signal is used. After that, the number of interacting promoters and enhancers within predefined topologically associated domains is counted as the enhancer features. The promoter features and enhancer features are consolidated into a matrix including H2A.Z, H3K27me3, H3K4me3, H3K9ac and the number of interacting enhancers with the dimensionality of 23, 127, 127, 62 and 19 respectively, which is from reference[9].

Model Architecture

In this section, the HaForest algorithm is proposed for identifying HIS genes. The model architecture as shown in Figure 2 consists of three components, the LightGBM library, a multiscale scanning module and a cascade forest structure. In HaForest, the epigenomic data are first processed by the LightGBM library to find highly expressive features. After that, a multiscale scanning with LDA is proposed to inform the useful contextual or structural features since the probability distribution features generated by the standard multigrained scanning module are not capable of enhancing the results. Finally, the cascade forest

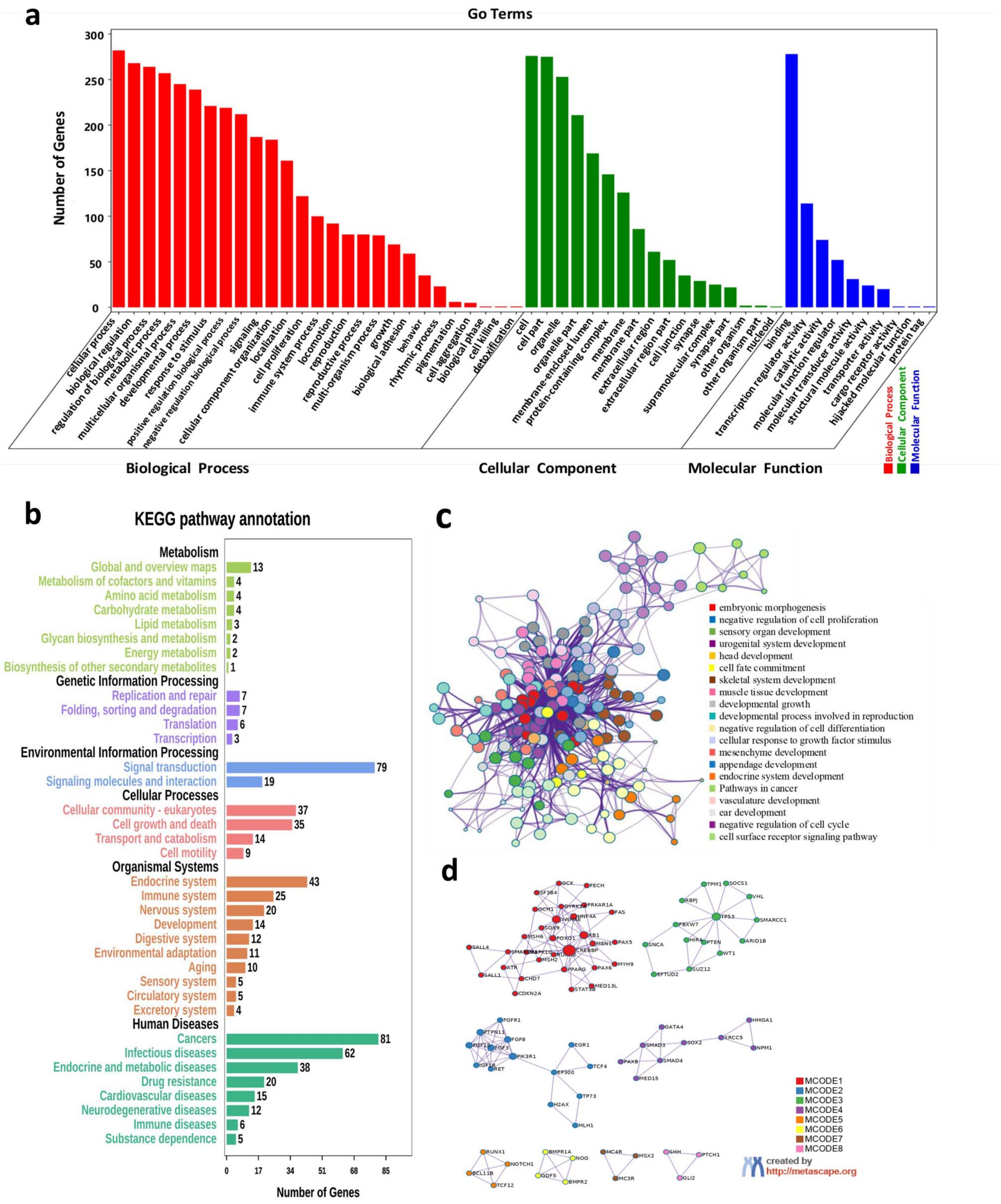


Figure 1. The enrichment analysis of 287 HIS genes. (a) GO annotations at different levels, including biological process, cellular component and molecular function. (b) KEGG pathway enrichment analysis of HIS genes, which links genes to various biochemical reactions so that can contribute to comprehending the relationship between genes and diseases. (c) The network of enriched terms, colored by cluster ID from the pathway and process enrichment analysis. (d) Protein–protein interaction enrichment analysis of HIS genes, the pathway and process enrichment analysis was applied to each MCODE component independently.

structure is employed to manage the concatenate features to generate a more precise class distribution probability of the features by integrating three decision tree-based forests [a random forest (RT), an extra tree (ET) and a Extreme Gradient Boosting (XGBoost)] under the Layer-Wise Training. Next, we introduce those components successively.

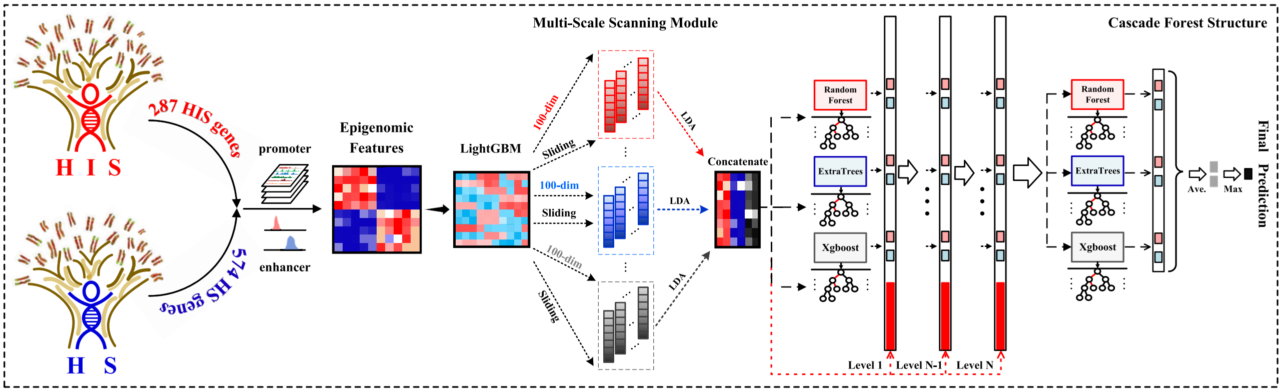


Figure 2. Overall procedure of HaForest. First, the positive training set of haploinsufficiency selected from ClinGen and the literature includes 287 genes, the negative training set 574 genes from healthy individuals, recurrently detected. The epigenomic feature matrix consists of promoter and enhancer features. Then, a lightGBM library is used to eliminate feature information with low importance. The epigenomic data are inputted into a multiscale scanning module and transformed into multi-instance feature vectors, which can be used to do feature combinations to strengthen the representation learning ability for structured data identification. In the cascade forest structure, a level-by-level tree structure, makes the method applicable to small-scale datasets. Finally, the end output of the cascade level is averaged to get a probability distribution, and a maximum value as classification result.

The LightGBM Library

In reality, particular features have diverse effects on identification performance. An important attribute can help in creating the classification model, whereas an irrelevant quality will not benefit identification performance. In the worst case, they may even affect algorithms adversely by blurring the classifier boundaries. Therefore, research is still necessary to develop feature analysis methods for identifying haploinsufficient genes.

To further enhance the classification performance, the illustration in Figure 2 highlights the gradient boosting machine (GBM) of the LightGBM library [19], which is first used to calculate the importance of epigenomic features. The LightGBM algorithm employs a leaf-wise growth strategy with depth constraints, finding leaves with the highest branching gain, and then going through the branching cycle. However, this leaf growth orientation results in deep decision trees that lead to overfitting. Therefore, LightGBM adds a limitation of maximum depth to the top leaf to prevent overfitting while ensuring high efficiency.

Given the supervised training dataset $X = \{(x_i, y_i)\}_{i=1}^{m}$, the purpose of LightGBM is to find an approximation $\hat{f}(x)$ to the function $f^*(x)$ which can minimize the expected value of the loss function $L(y, f(x))$ as follows:

$$\hat{f} = \arg\min_{f} E_{y,X} L(y, f(x)) \tag{1}$$

The LightGBM algorithm integrates multiple regression trees $\sum_{t=1}^{T} f_t(X)$ to approximate the ultimate model, which is defined as follows:

$$f_T(X) = \sum_{t=1}^{T} f_t(X) \tag{2}$$

The regression trees are described as $w_{q(x)}, q \in \{1, 2, \ldots, N\}$, where w is a leaf node sample weight vector, q denotes the decision rule of trees and N represents the number of leaves. Hence, in step t, the algorithm is trained with an additive form as follows:

$$\Gamma_t = \sum_{i=1}^{n} L\left(y_i, F_{t-1}(x_i) + f_t(x_i)\right) \tag{3}$$

Using the Newton–Raphson method, the objective function of LightGBM is rapidly approximated as follows:

$$\Gamma_t \cong \sum_{i=1}^{n} \left(g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i)\right) \tag{4}$$

where g_i and h_i represent the 1st and 2*nd*-order gradient statistics of the loss function, respectively. When the sample set of leaf j is defined by I_j, Equation (4) can be further transformed as follows:

$$\Gamma_t = \sum_{j=1}^{J} \left(\left(\sum_{i\in I_j} g_i\right) w_j + \frac{1}{2}\left(\sum_{i\in I_j} h_i + \lambda\right) w_j^2\right) \tag{5}$$

For a specific tree structure $q(x)$, w_j^* denotes the optimum leaf weight values of each leaf node and the extreme value, i.e. Γ_K is converted to the following expressions:

$$w_j^* = -\frac{\sum_{i\in I_j} g_i}{\sum_{i\in I_j} h_i + \lambda} \tag{6}$$

$$\Gamma_T^* = -\frac{1}{2} \sum_{j=1}^{J} \frac{\left(\sum_{i\in I_j} g_i\right)^2}{\sum_{i\in I_j} h_i + \lambda} \tag{7}$$

where Γ_T^* denotes the scoring function that measures the quality of tree structure $q(x)$. The final objective function after gathering the split is derived as follows:

$$G = \frac{1}{2}\left(\frac{\left(\sum_{i\in I_L} g_i\right)^2}{\sum_{i\in I_L} h_i + \lambda} + \frac{\left(\sum_{i\in I_R} g_i\right)^2}{\sum_{i\in I_R} h_i + \lambda} - \frac{\left(\sum_{i\in I} g_i\right)^2}{\sum_{i\in I} h_i + \lambda}\right) \tag{8}$$

where the parameters of I_L and I_R are the left and right branch of sample sets, respectively. After that, the importance values are calculated by the LightGBM and sorted by normalized importance where the total sums to 1.

Multiscale scanning

The multiscale sliding window operations are adopted to scan the epigenomic data processed by the LightGBM from high-dimensionality to multi-instance feature vectors, which can strengthen the representation learning ability of the model. The module is employed in the Figure 2. Considering a haploinsufficient data $\mathbf{x}\left(\in \mathbb{R}^{m\times n}\right)$, the window size is ws with the stride of 1; thus, the total feature vectors can be generated as follows:

$$\mathbf{x}\left(\in \mathbb{R}^{m\times n}\right) \xrightarrow[Stride=1]{Windows=ws} \left\{\tilde{x}_i \in \mathbb{R}^{m\times ws}\right\}_{i=1}^{n\text{-}ws+1} \tag{9}$$

Then, each feature vector $\tilde{x}_i$ is processed by LDA technique; the details are described in the next section. After concatenating, an output of $n - ws + 1$ dimension c_i is defined as follows:

$$\left\{\tilde{x}_i \in \mathbb{R}^{m\times ws}\right\}_{i=1}^{n\text{-}ws+1} \xrightarrow{LDA} c_i\left(\in \mathbb{R}^{m\times(n\text{-}ws+1)}\right) \tag{10}$$

Finally, the output of multiscale window size k will be concatenated and transmitted to the cascade forest structure as follows:

$$\left\{c_i \in \mathbb{R}^{m\times(n\text{-}ws+1)}\right\}_{i=1}^{k} \xrightarrow{Concatenate} \mathbf{c} = \left(\tilde{y}_1, \ldots, \tilde{y}_k\right) \tag{11}$$

Linear Discriminant Analysis

LDA is good for handling within-class variation, which extracts the most discriminating features from the dataset and is commonly uesd in HISidentification [5, 20, 21]. The basic idea of LDA is to project the dataset $D = \left\{(x_i, y_i)\right\}_{i=1}^{m}$, where $x_i \in \mathbb{R}^n$, $o_i \in \{0,1\}$ onto a straight line; the projection points of similar samples should be as close as possible. Considering w_1, w_2 are two classes of HIS and HS, and $N_l, N_2, \ldots, N_L$ represent the number of genes in each class. Let M_1, M_2 and M be the means of the classes and the total mean, respectively. Then, the between-class scatter matrix S_b and within-class scatter matrix S_w are defined as:

$$S_b = \sum_{i=1}^{L} P\left(w_i\right)\left(M_i - M\right)\left(M_i - M\right)^T \tag{12}$$

and

$$\begin{aligned} S_w &= \sum_{i=1}^{L} P\left(w_i\right)\Sigma_i \\ &= \sum_{i=1}^{L} P\left(w_i\right) E\left\{\left[X - M_i\right]\left[X - M_i\right]^T |w\right\} \end{aligned} \tag{13}$$

where $P\left(w_i\right)$ and Σ_i represent the priori probability and covariance matrix of the class sample w_i, respectively. The LDA derives a projection matrix ψ, which can maximize the ratio $J(\psi)$ as the following equation:

$$\arg\max_{\psi} J(\psi) = \frac{\psi^T S_b \psi}{\psi^T S_w \psi} \tag{14}$$

The ratio $J(\psi)$ is maximized when ψ composes of the eigenvectors with the $S_w^{-1}S_b$ matrix. It involves the simultaneous diagonalization of the between-class scatter matrix and the within-class scatter matrix, respectively.

In conclusion, as long as the priori probability and covariance matrix of the original data are calculated, the best projection ratio $J(\psi)$ can be determined. The LDA induces the nonorthogonal projection axes, which can distinguish within and between class scatters. The relevant characteristics in the epigenomic features are first preselected and then removed, replaced or added iteratively to the model. Thus, the most useful advanced features are integrated and transmitted to the cascade forest structure to improve the identification performance.

Cascade forest structure

Instead of learning hidden contextual features with the complex forward and backward propagation frameworks of deep neural networks, the cascade forests can achieve representation learning according to multigrained scanning at low expense. Multiple decision tree forests are assembled in a cascade forest to learn identification distribution according to cascade layers. The structure attempts to obtain a more precise class distribution probability of input features by integrating directly multiple decision tree-based forests in a layer-wise supervised learning, which can be trained easily and performs well on learning discriminative representations.

Our procedure of cascade forest structure is depicted in Figure 2, illustrating that the concatenated features $\mathbf{c}\left(\in \mathbb{R}^{m\times h}\right)$ processed by the multiscale scanning are transmitted to three forests (a RF, an ET and a XGBoost [22]) on the same level. The concatenated features are considered as candidates to choose the one with the best *gini* value for split; the ground true label information is needed to annotate leaf nodes. Each level in the cascade forest will receive these probabilistic values counted by its categorical level, and the original input, then outputs its concatenated result $(h + 3 \times 2)$ to the next level, whereas the ground true labels are still regarded as labels of the new training features. Then, the structure of the cascade forest can appropriately deal with the various scales of data training. After calculating the last cascade layer N, the average of each probabilistic class value is produced by the three types of forests, and the highest probability of the probability distribution is set as the final identification label.

Parameter settings

In this study, we compare our model to multiple computational methods, including traditional machine-learning algorithms and other deep learning algorithms. The parameters of each method are set as follows:

- *HaForest Algorithm*:

We add three types of basic decision tree models at each level, each forest containing 200 trees to enhance its diversity, which is crucial for ensemble construction [23]. To avoid overfitting, we use 3-fold cross-validation when class vectors are generated from each instance. The expanding process can be automatically determined by an evaluation criterion. If the performance on the validation set does not improve after adding a new cascade level, the growth of levels should be terminated.

- *Deep learning algorithms*:

The DNN architecture consists of two *denselayer*; each layer contains 317 filters, and the activation of '*relu*' is followed by the layer to increase the nonlinear effect. For the CNN framework, we add two *conv1Dlayer* with 256 filters and '*relu*' activation to make convolution calculations. The convolution kernel scale in one and the *flattenlayer* is used to integrate the convolution results. The RNN includes two *LSTMlayer* with the parameter '*return sequences*' as true. Each layer also has 256 filters and uses the '*tanh*' activation. In addition, the *flattenlayer* is also applied to transmit the output results. The SAE architecture uses four *denselayer* with different filters, including 317, 159, 106 and 79 filters to make a dimension reduction operation. The activation of '*relu*' is also followed by each layer and the additional *denselayer* with '*sigmoid*' activation to identify the category.

The implementation of all the above network architectures uses the TensorFlow [24] and Keras library [25]. The dropout and batch normalization modules in the hidden layers are applied to improve the generalizability of the models. Meanwhile, the Adam optimizer [26] is used to train all the layers in the deep network simultaneously and the loss function of '*binarycrossentropy*'.

- *Machine learning algorithms*:

The traditional machine-learning algorithms are compiled with scikit-learn package in Python [27]. The parameter settings of RF and SVM are the same as Episcore, including *nestimators*: 2,000, *cost*: 10 and gamma: 10, respectively. The K-Nearest Neighbor (KNN) algorithm with parameters of *nneighbors*: 10, *weights*: 'distance' and *p*: 1. For the Logistic Ridge Regression (LRR) method, the *penalty* is 'l2' and the *solver* is lbfgs. The Gaussian Naive Bayesian (GaussianNB) model is used as the default setting.

Results and discussion

Performance comparison to different achine learning algorithms

We compare our HaForest method to multiple machine-learning algorithms, including KNN, SVM, GaussianNB, LRR and Episcore. The epigenomic features are processed by LightGBM library. Meanwhile, to execute a fair comparison, we conform to the rules of Episcore, which assess the performance of all algorithms based on 100 runs of 10-fold cross-validation. Ultimately, we calculate the average area under curve (AUC) values (area under the ROC curves) and the mean precision (area under the precision-recall curves) as the final results. The comparison results are summarized in Figure 3a and b, respectively.

Our method achieves a 96% AUC value, which is higher than the 90% value of Episcore in *Nature Communications* article and also higher than all the other algorithms. Meanwhile, the mean precision value of HaForest is 93%, which is also superior to other algorithms. To further verify the stability of HaForest, we compare our method to other machine learning algorithms in 30 independent runs, and the comparison results are summarized in Figure 3d. Notably, our algorithm has a minimal standard deviation (SD). The comparison results demonstrate the effectiveness of our proposed model for identifying HIS genes.

Performance comparisons to other deep learning algorithms

Besides the above conventional machine learning algorithms, a variety of deep learning methods are also compared to HaForest, including DNN, RNN, CNN and SAE. Although deep learning algorithms have made great progress in several applications, especially image and speech recognition [29, 30], there are a few deficiencies when dealing with small-scale network structured data, such as the HIS data. For biological data, the contextualization information between the epigenomic features is not as obvious as adjacent pixels in visuals. Furthermore, it is well known that the high performance of deep learning requires a huge amount of training data to adjust the hyperparameters of networks. However, the reliable data of HIS genes is not large enough at present, which can easily cause it to fall into overfitting and local optimization.

The AUC value of each method is stochastically run for 30 times and demonstrates that HaForest is more practical than deep learning algorithms. As shown in Figure. 3c, the average AUC values sorted according to each method are 77.31%, 87.22%, 86.35%, 84.56%, 89.78% and 96.27%, respectively. RNN performs the worst because the epigenomic data does not contain temporal information as natural languages do. Due to the amount of data, other deep learning methods can hardly be applied to tasks of small-scale training data, leading to a wide fluctuation in results as shown in Fig. 3d. However, HaForest achieves the highest AUC value of 96.27% and has the minimum SD. Therefore, our method achieves not only the best performance but shows stability of the model, thus working well on small-scale data.

Feature importance nalysis

To further advance the identification performance, and illustrated in Figure 2, we use the feature selection method based on LightGBM to deal with the raw feature matrix, and to connect to the HaForest algorithm. The original features with experimental noise may have the following problems, features with a high percentage of missing values, collinear (highly correlated) features, features with zero importance in a tree-based model and features with low importance. Therefore, we use LightGBM to analyze our feature vectors and calculate the importance of features.

We consider the feature correlation between all positive and negative genes in the datasets, as shown in Figure 4a, which is a heatmap showing the Spearman correlation between epigenomic features clearly indicating whether they are positively or negatively correlated. Then, we use LightGBM to calculate the importance values of the epigenomic features. The feature importances are averaged over 10 training runs of the gradient boosting machine so as to reduce variance. Figure 4b and c illustrate the efficiency of the 4b denotes the incorrect classification results of samples in linearly discriminated regions, and the symbols in Figure 4c are the P-values, where *** denotes 0.001; ** denotes 0.01; * denotes 0.05. In addition, the violin illustrations are plotted in Figure 4d and e represent the distribution of epigenomic molecular entities. The H3K4me3 group in the figures showed a higher importance than the others, which as mentioned for the Episcore [9], group members have been shown to be predominantly haploinsufficient based on somatic mutation patterns and their critical dosage sensitivity. It is particularly important that we find that the repressive promoter features H3K27me3 accounted for the majority of the removed features, also verified for Episcore by RandomForest method, thus demonstrating the reliability of scoring. The t-distributed stochastic neighbor embedding is applied to visualize the classification effect of 358-dimensional and 317-dimensional features on the same dataset, respectively. The results in Figure 4f and g conclude that the 317-dimensional remaining features have

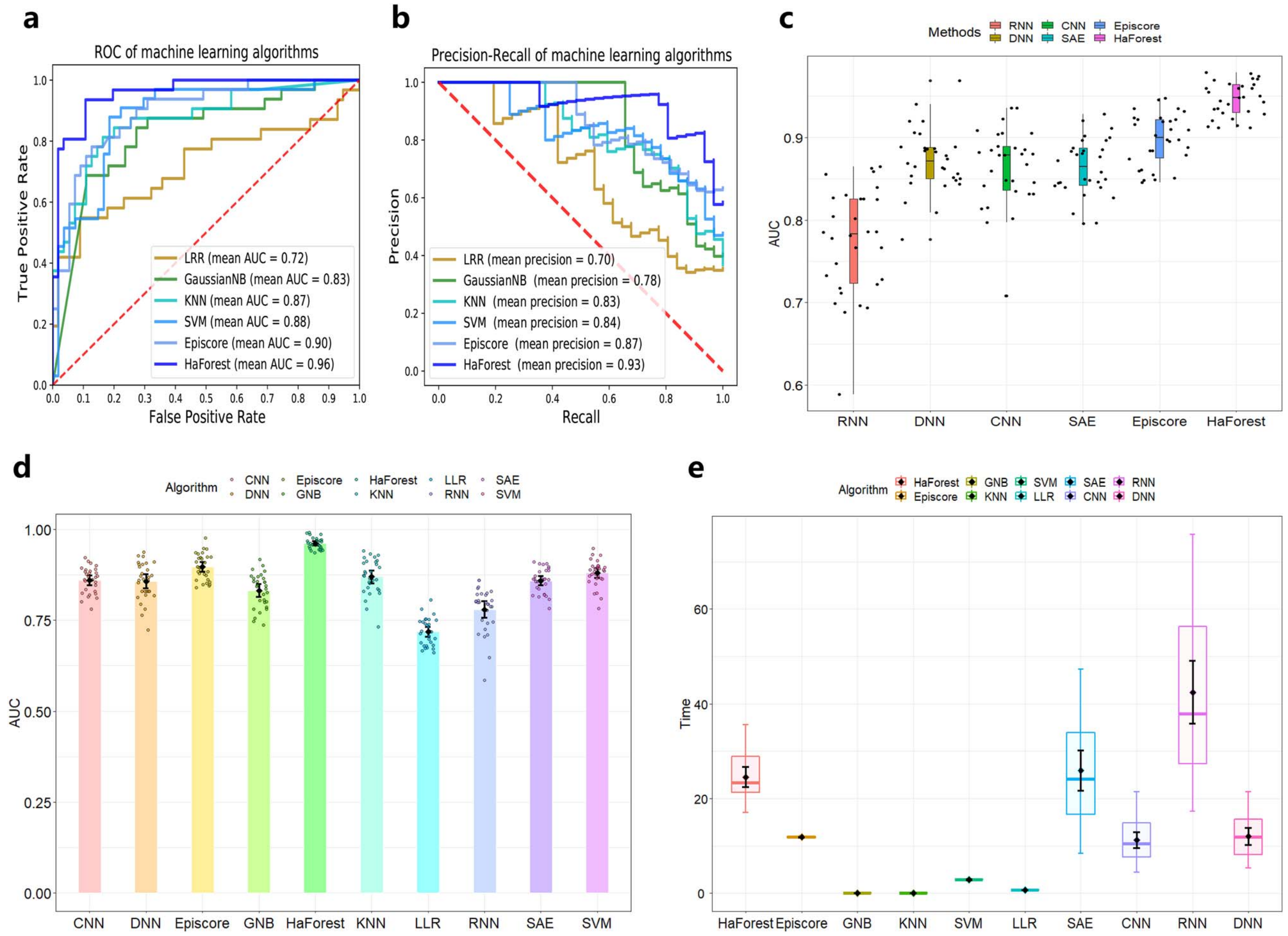


Figure 3. Performance comparisons (a) and (b). The AUC value and mean precision based on 100 runs comparing HaForest with multiple machine learning algorithms. (c). HaForest compared with multiple deep learning algorithms. The average AUC value of 96.27% in 30 randomly runs is superior. (d). Performance comparison of the stability between HaForest and multiple machine learning or deep learning algorithms based on 30 independent runs. (e). Running time comparisons between HaForest and multiple machine learning or deep learning algorithms based on 30 independent runs.

better discriminative ability. In addition, we compare the HaForest with several network-based regularized variable selection models [28], including minimax concave penalty, elastic net and least absolute shrinkage and selection operator. To conduct a fair comparison, we adopt 10-fold cross-validation in network constrained regularization (NCR) method to avoid that well-studied features are over-represented in the networks [31]. With different network-based regularization, the average AUCs of NCR and HaForest are 88.69%, 96.12%, 89.11% and 87.15%, respectively. From the results, we observe that our proposed model perform better than the network-based regularized variable selection models. The reason is attributed to the observation that NCR method is very suitable for high dimensional data [28] whereas the moderate size of epigenomic features influences the performance of NCR.

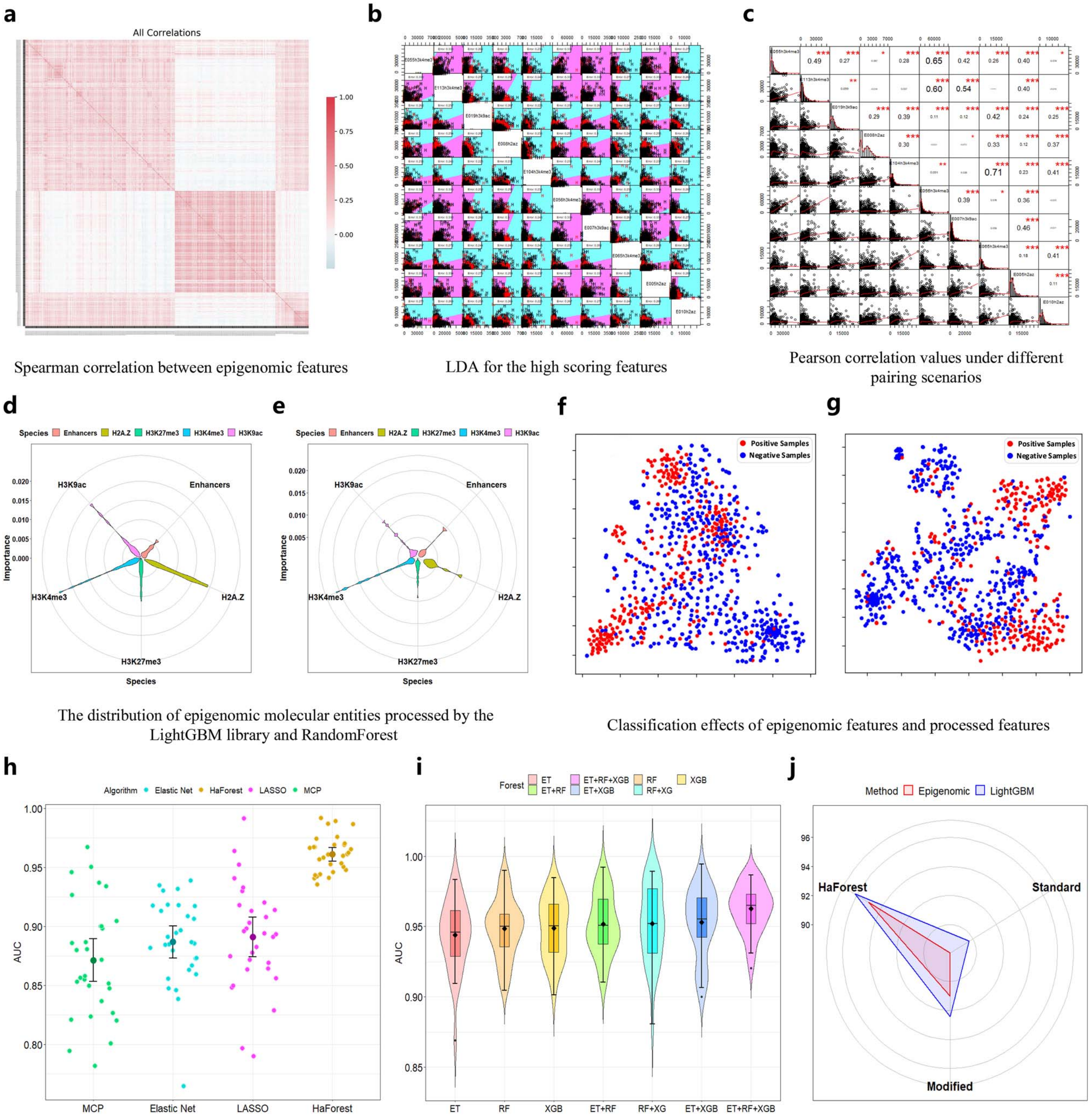
Model architecture analysis

In this section, the epigenomic features and the features processed by LightGBM are used to authenticate the effects of different model architectures, respectively, including the standard deep forest, modified deep forest, and HaForest with a window size of 300, 200 and 100. The modified deep forest structure means the traditional multi-grained scanning module without exercising a RF and an ET. For the standard deep forest, the class distribution vectors are generated by each forest and integrated as transformed features with dimensions 1036-dim (2×259×2 dim), 636-dim (2×159×2 dim) and 236-dim (2×59×2 dim). The 1908-dimensional concatenate features are transmitted to the cascade forest. Similarly, the 1,416-dimensional features are generated by 317-dim data. For the modified structure, the input dimensions received by the cascade forest are 75 400-dim and 50 800-dim, respectively. Our HaForest method generated the 477-dim and 354-dim feature matrices, using the multiscale scanning module. The comparison results of all model architectures are shown in Figure 4j.

The standard deep forest is 88.05% with 1908-dim data, and 89.67% with 1416-dim data. The AUC values for the input dimensions of 75400 and 50800 using modified structure are 91.03% and 92.43%, respectively. The proposed HaForest achieves the highest AUC values of 95.02% and 96.27% for each dataset. The comparsion results demonstrate the effectiveness of the proposed architecture.

Effect of ifferent orest algorithm subsets

To construct a good cascade forest structure, individual learners should be accurate and diverse [10]. Therefore, in our study, we



Spearman correlation between epigenomic features

LDA for the high scoring features

Pearson correlation values under different pairing scenarios

The distribution of epigenomic molecular entities processed by the LightGBM library and RandomForest

Classification effects of epigenomic features and processed features

Comparison results on multiple highly expressive extraction algorithms

The average AUC values of different forest combination subsets

Effect of multiple model architectures based on Deep Forest algorithm

Figure 4. Contribution of epigenomic features to HIS identification using Haforest. **(a)** Spearman correlation between promoter and enhancer features on all training genes. **(b)** LDA for the high scoring features, the red font denotes the wrongly classified genes. **(c)** Feature correlation matrix related to (b), the Pearson correlation values in the different pairing scenarios are tabulated in the upper triangle. **(d)** The importance values of epigenomic features supplied by the LightGBM library. **(e)** The feature contributions dealing with the RandomForest method in Episcore. **(f)** t-SNE visualization of the epigenomic data. **(g)** Identification results of processed features using t-SNE. **(h)** Performance comparison of our proposed HaForest with several network-based regularized variable selection models [28] including minimax concave penalty (MCP), elastic net, and least absolute shrinkage and selection operator (LASSO). **(i)** The average AUC values among several combination schemes as visualized in violin plots. **(j)** The comparsion results with multiple model architectures. The HaForest algorithm with features processed by LightGBM achieves the highest AUC value of 96.27%.

choose different types of forests including RF, ET, and XGBoost to encourage diversity because it is crucial for cascade forest structure construction. To demonstrate the effect of different forest algorithm subsets, we conduct an experiment to compare our proposed model under six different combinations of forest algorithms. The experimental results are summarized in Figure 4i. Specifically, the scheme of three-type-forests achieves the highest average AUC value of 96.27% and the minimum SD based on 30 independent runs. According to each pairwise forest combination, the average AUC value sorted are

Table 1. Performance of HaForest and Episcore in identifying HIS genes related to known human diseases.

Method	Acc	Sen	Spe	Pre	F1	AUC
ASD 1						
LRR	0.5297	0.4919	0.5660	0.5200	0.5050	0.5048
GaussianNB	0.4844	0.7281	0.2553	0.4820	0.5802	0.4693
KNN	0.5431	0.7207	0.3730	0.5243	0.6068	0.5410
SVM	0.5109	0.5689	0.4553	0.4998	0.5320	0.5215
RNN	0.5123	0.5726	0.4546	0.5030	0.5339	0.5099
DNN	0.4993	0.5844	0.4178	0.4895	0.5281	0.4908
SAE	0.5065	0.6474	0.3716	0.4953	0.5571	0.5100
CNN	0.5083	0.5904	0.4298	0.4997	0.5368	0.5084
Episcore	0.5274	0.5726	0.4665	0.5058	0.5336	0.5787
HaForest*	**0.7710**	**0.8556**	**0.6901**	**0.7144**	**0.7791**	**0.8540**
ASD 2						
LRR	0.5080	0.5147	0.5013	0.5080	0.5111	0.5152
GaussianNB	0.5540	0.8073	0.3007	0.5358	0.6439	0.5653
KNN	0.5853	0.6687	0.5020	0.5733	0.6167	0.6186
SVM	0.5140	0.5360	0.4830	0.5131	0.5241	0.5008
RNN	0.5150	0.5353	0.4947	0.5143	0.5231	0.5247
DNN	0.5493	0.6133	0.4853	0.5462	0.5726	0.5641
SAE	0.5397	0.6693	0.4100	0.5335	0.5880	0.5501
CNN	0.5313	0.5967	0.4660	0.5295	0.5560	0.5521
Episcore	0.5230	0.5593	0.4867	0.5212	0.5394	0.5414
HaForest*	**0.7434**	**0.8080**	**0.6443**	**0.6708**	**0.7417**	**0.8023**
OMIM HI						
LRR	0.5878	0.5020	0.7076	0.7056	0.5862	0.6341
GaussianNB	0.6159	0.8109	0.3437	0.6331	0.7111	0.6180
KNN	0.5452	0.6184	0.4430	0.6078	0.6128	0.5375
SVM	0.5932	0.6388	0.5458	0.6324	0.6205	0.5887
RNN	0.5159	0.5144	0.5181	0.5972	0.5510	0.5006
DNN	0.5858	0.6413	0.5083	0.6499	0.6395	0.5891
SAE	0.5872	0.7054	0.4222	0.6337	0.6622	0.5708
CNN	0.5864	0.6383	0.5139	0.6499	0.6394	0.5821
Episcore	0.6047	0.6537	0.5361	0.6628	0.6583	0.6187
HaForest*	**0.8304**	**0.8522**	**0.7958**	**0.8275**	**0.8413**	**0.8819**
OMIM HI denovo						
LRR	0.5423	0.4467	0.7087	0.7279	0.5530	0.6017
GaussianNB	0.6836	0.8317	0.4261	0.7159	0.7692	0.6418
KNN	0.5497	0.6425	0.3884	0.6459	0.6440	0.4844
SVM	0.5410	0.5350	0.4870	0.5829	0.5580	0.5420
RNN	0.4915	0.4958	0.4841	0.6240	0.5501	0.4709
DNN	0.5534	0.6133	0.4493	0.6616	0.6312	0.5346
SAE	0.5693	0.6725	0.3899	0.6573	0.6602	0.5295
CNN	0.5529	0.6058	0.4609	0.6635	0.6293	0.5413
Episcore	0.5857	0.6533	0.4681	0.6811	0.6668	0.5949
HaForest*	**0.8206**	**0.8667**	**0.7405**	**0.8135**	**0.8393**	**0.8629**

*Experimental results calculated by us. Bold data represent the best experimental results.

95.17%, 95.22%, and 95.32%, respectively. The average AUC values for single forest classifiers are 94.39%, 94.85%, and 94.88%, respectively. From the results, we can observe that the performance of our proposed model can be arisen from the diversity of deep-cascade structure, and the ensemble schemes usually obtained good performance compared with the single learners [32].

Comparison of HaForest and existing methods on runtimes

In this section, we conduct runtime comparisons between HaForest and other computational methods based on 30 independent runs. The entire architecture is executed on a desktop computer system with the following hardware and software configurations: CPU: 3.60 GHz Intel Core i9, RAM: 32.0 GB, OS: Windows (64-bit, Version 10), and Python 3.7. The runtime results on the datasets are summarized in Figure 3e. Indeed, HaForest performs better than SAE and RNN although it costs long running times compared to the other machine learning and deep learning algorithms. Such phenomenon could stem from the fact that HaForest employs multigrained scanning to retrieve its transformed feature representations, and then continues through the cascade until the last level.

Generalization analysis

Existing methods are affected by study bias, i.e. a good performance on well-studied genes but may not generalize well to less studied genes. To verify the generalizability of our unbiased model, we use four kinds of benchmark datasets [4] from known human disease-associated genes to contrast the performances alongside the five machine learning algorithms and four deep learning algorithms. As the potential overlaps between the benchmark datasets and the training set can lead to overestimation of performance, we remove the genes present in our training data. After processing, we are left with 92 genes in ASD1 and 100 genes in ASD2, 115 in OMIM HI, and 63 in OMIM HI *de novo*. The comparison results based on 30 random runs under 10-fold cross-validation are listed in Table 1, which demonstrates the generalizability and consistency of HaForest on other benchmark datasets.

It can be seen from Table 1 that our HaForest method shows much higher identification achievement than the other algorithms in six measurements, including accuracy, sensitivity, specificity, precision, F1score and AUC value. For ASD1 and ASD2, the accuracy values of the deep learning algorithms are in the range [48.44–52.97] and [50.80–58.53], respectively; the machine learning algorithms in the range [49.93–51.23] and [51.50–54.93], respectively. Which makes it almost impossible to distinguish between HIS and HS genes. However, HaForest on ASD1 and ASD2 achieves higher accuracy values of 85.40% and 80.23%. In addition, in the other five measurements, HaForest is superior to the other methods. Moreover, for OMIM HI and OMIM HI *de novo*, although experimental performance improved, it is still not as good as HaForest. The weak generalizability of comparison methods makes it hard to recognize HIS genes. The experimental results of multiple datasets also reveals that the performance of traditional machine learning methods is better than that of standard deep learning algorithms for HIS identification, caused by the limited data of known HIS genes. Concisely, our method demonstrates not only an outstanding learning ability on the training set but also robustness toward unknown gene data, demonstrating its general applicability.

Conclusion

In this study, we propose a computational method, HaForest, based on the deep forest algorithm for classifying haploinsufficient genes from epigenomic data. The algorithm consists of two parts, a multiscale scanning module and a cascade forest structure. To exploit the complex pathological mechanisms of haploinsufficient genes, the LightGBM library is used to extract the highly expressive features. Furthermore, to illustrate the effectiveness of our method, we conduct experiments comparing HaForest with several machine learning algorithms, showing our model achieves superior experimental results. After that, several state-of-the-art deep learning algorithms are utilized to contrast with HaForest to estimate the robustness on biological data with fewer hyper-parameters and more accurate identification ability. Then, we compare a variety of model architectures to verify our overall design. Finally, we benchmark our method on several datasets with known human disease genes. The comparison results at different levels demonstrate the capability of our approach and reflect the potential of the proposed model for identifying haploinsufficient genes. We believe that our study provides a stimulating view on the use of deep frameworks for identifying haploinsufficient genes in epigenomic data, enabling downstream studies of related genomic problems. On the other hand, as suggested by an anonymous reviewer, it would be interesting to extend the NCR to the epigenomic features subject to data availability. The growth in epigenomic data will be another computational challenge in the future.

Key points

- At present, the existing methods for haploinsufficiency identification suffer from study bias, experimental noise and instability. Significant improvements are still required.
- We propose a novel method called HaForest, a multiscale scanning module and a cascade forest structure for haploinsufficient gene identification. Notably, the LightGBM library is embedded into HaForest to reveal highly expressive features in epigenomic data.
- Multiple experiments are conducted on five epigenomic data sets, including known and candidate disease genes. The experimental results indicate that the HaForest framework has unique and complementary performance for identifying haploinsufficient genes.

Funding

National Natural Science Foundation of China under Grant No. 62076109, and funded by the Natural Science Foundation of Jilin Province under Grant No. 20190103006JH. Health and Medical Research Fund, the Food and Health Bureau, the Government of the Hong Kong Special Administrative Region [07181426], and the funding from Hong Kong Institute for Data Science (HKIDS) at City University of Hong Kong. The work described in this paper was partially supported by two grants from City University of Hong Kong (CityU 11202219, CityU 11203520). National Natural Science Foundation of China under Grant No. 32000464.

References

1. Dang VT, Kassahn KS, Ragan MA, *et al*. Identification of human haploinsufficient genes and their genomic proximity to segmental duplications. *Eur J Hum Genet* 2008; **16**(11): 1350.
2. Seidman J, Seidman C. Transcription factor haploinsufficiency: when half a loaf is not enough. *J Clin Invest* 2002; **109**(4): 451–5.
3. Veitia RA. Exploring the etiology of haploinsufficiency. *Bioessays* 2002; **24**(2): 175–84.
4. Shihab HA, Rogers MF, Gaunt TR, *et al*. HIPred: an integrative approach to predicting haploinsufficient genes. *Bioinformatics* 2017; **33**(12): 1751–7.
5. Huang N, Lee I, Hurles ME, *et al*. Characterising and predicting haploinsufficiency in the human genome. *PLoS Genet* 2010; **6**(10):e1001154.
6. Steinberg J, Honti F, Webber C, *et al*. Haploinsufficiency predictions without study bias. *Nucleic Acids Res* 2015; **43**(15): e101–1.
7. Kundaje A, Meuleman W, Ziller MJ, *et al*. Integrative analysis of 111 reference human epigenomes. *Nature* 2015; **518**(7539): 317.
8. Consortium EP, *et al*. An integrated encyclopedia of DNA elements in the human genome. *Nature* 2012; **489**(7414): 57.
9. Han X, Chen S, Shen Y, *et al*. Distinct epigenomic patterns are associated with haploinsufficiency and predict risk genes of developmental disorders. *Nat Commun* 2018; **9**(1): 2138.
10. Zhou Z-H, Feng J. Deep Forest. *arXiv preprint arXiv:170208835* 2017.

11. Shaikh TH, Gai X, D'arcy M, *et al.* High-resolution mapping and analysis of copy number variations in the human genome: a data resource for clinical and research applications. *Genome Res* 2009; **19**(9): 1682–90.
12. Ashburner M, Ball CA, Eppig JT, *et al.* Gene ontology: tool for the unification of biology. *Nat Genet* 2000; **25**(1): 25.
13. Kanehisa M, Goto S, Tanabe M, *et al.* KEGG for integration and interpretation of large-scale molecular data sets. *Nucleic Acids Res* 2011; **40**(D1): D109–14.
14. Zhou Y, Zhou B, Chanda SK, *et al.* Metascape provides a biologist-oriented resource for the analysis of systems-level datasets. *Nat Commun* 2019; **10**(1):1523.
15. Petrovski S, Wang Q, Goldstein DB, *et al.* Genic intolerance to functional variation and the interpretation of personal genomes. *PLoS Genet* 2013; **9**(8): e1003709.
16. Iossifov I, Ronemus M, Leotta A, *et al.* De novo gene disruptions in children on the autistic spectrum. *Neuron* 2012; **74**(2): 285–99.
17. Davoli T, Xu AW, Elledge SJ, *et al.* Cumulative haploinsufficiency and triplosensitivity drive aneuploidy patterns and shape the cancer genome. *Cell* 2013; **155**(4): 948–62.
18. Benayoun BA, Pollina EA, Mancini E, *et al.* H3K4me3 breadth is linked to cell identity and transcriptional consistency. *Cell* 2014; **158**(3): 673–88.
19. Ke G, Meng Q, Liu T-Y, *et al.* Lightgbm: a highly efficient gradient boosting decision tree. In: *Advances in Neural Information Processing Systems*, 2017, 3146–54.
20. Norris M, Lovell S, Delneri D. Characterization and prediction of haploinsufficiency using systems-level gene properties in yeast. *G3* 2013; **3**(11): 1965–77.
21. Quinodoz M, Royer-Bertrand B, Rivolta C, *et al.* DOMINO: using machine learning to predict genes associated with dominant disorders. *Am J Hum Genet* 2017; **101**(4): 623–9.
22. Chen T, Guestrin C. Xgboost: A scalable tree boosting system. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, 785–94.
23. Zhou Z-H. *Ensemble Methods: Foundations and Algorithms*. CRC press, 2012.
24. Abadi M, Isard M, *et al.* Tensorflow: a system for large-scale machine learning. In: *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, 2016, 265–83.
25. F. Chollet *et al.*, "Keras," 2015.
26. Kingma DP, Ba J. Adam: a method for stochastic optimization. *arXiv preprint arXiv:14126980* 2014.
27. Pedregosa F, Varoquaux G, Dubourg V, *et al.* Scikit-learn: machine learning in python. *J Mach Learning Res* 2011; 2825–30.
28. Ren J, He T, Wu C, *et al.* Network-based regularization for high dimensional SNP data in the case-control study of type 2 diabetes. *BMC Genet* 2017; **18**(1): 44.
29. Hinton G, Deng L, Kingsbury B, *et al.* Deep neural networks for acoustic modeling in speech recognition. *IEEE Signal Process Mag* 2012;29.
30. Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. In: *Advances in Neural Information Processing Systems*, 2012, 1097–105.
31. Wu C, Ma S. A selective review of robust variable selection with applications in bioinformatics. *Brief Bioinform* 2015; **16**(5): 873–83.
32. Kuncheva LI, Whitaker CJ. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Mach Learn* 2003; **51**(2): 181–207.