BioData Mining

METHODOLOGY

Open Access

Single_cell_GRN: gene regulatory network identification based on supervised learning method and Single-cell RNA-seq data

Bin Yang[1], Wenzheng Bao[2*], Baitong Chen[3] and Dan Song[1*]

*Correspondence:
baowz55555@126.com;
songdan@uzz.edu.cn

[1] School of Information Science and Engineering, Zaozhuang University, Zaozhuang 277160, China
[2] School of Information and Electrical Engineering, Xuzhou University of Technology, Xuzhou 221018, China
[3] Xuzhou First People's Hospital, Xuzhou 221000, China

Abstract

Single-cell RNA-seq overcomes the shortcomings of conventional transcriptome sequencing technology and could provide a powerful tool for distinguishing the transcriptome characteristics of various cell types in biological tissues, and comprehensively revealing the heterogeneity of gene expression between cells. Many Intelligent Computing methods have been presented to infer gene regulatory network (GRN) with single-cell RNA-seq data. In this paper, we investigate the performances of seven classifiers including support vector machine (SVM), random forest (RF), Naive Bayesian (NB), GBDT, logical regression (LR), decision tree (DT) and K-Nearest Neighbor (KNN) for solving the binary classification problems of GRN inference with single-cell RNA-seq data (Single_cell_GRN). In SVM, three different kernel functions (linear, polynomial and radial basis function) are utilized, respectively. Three real single-cell RNA-seq datasets from mouse and human are utilized. The experiment results prove that in most cases supervised learning methods (SVM, RF, NB, GBDT, LR, DT and KNN) perform better than unsupervised learning method (GENIE3) in terms of AUC. SVM, RF and KNN have the better performances than other four classifiers. In SVM, linear and polynomial kernels are more fit to model single-cell RNA-seq data.

Keywords: Single-cell, RAN-seq, Gene regulatory network, Supervised learning, Classification

Introduction

Human diseases, especially polygenetic genetic diseases, mainly including heart disease, hypertension, diabetes, asthma and cancer, are caused by the interaction of multiple gene loci and environmental factors [1–4]. Therefore, to construct gene regulatory network (GRN) and analyze regulatory mechanism have contributed to finding out the key network nodes, which could make an importance role in formulating new treatment plans and drug targets [5–8].

For gene regulatory network modeling, the existing learning methods could be divided into two categories: supervised learning and unsupervised learning [9]. Supervised learning methods could simulate problem of gene regulatory network recognition as

classification problem. For a certain transcription factor (TF), genes could be divided into either TF-regulating genes or non-TF-regulating genes. The known regulation relationships are utilized to train the classifier and predict the unknown regulation relations. Due to the guidance with prior knowledge, supervised learning methods have been presented to infer GRN in the past decade. Wang et al. proposed a novel supervised inference method of GRN based on linear programming to infer the potential known transcription regulators [10]. Mordelet and Vert presented support vector machine (SVM) algorithm to solve binary classification problem of GRN [11]. Cerulo et al. solved the problem of unreasonable selection of negative samples [12]. Gillani investigated the performances of the different kernel functions of SVM for GRN inference and given the guidance about the research on supervised learning in the future [13]. Brouard et al. proposed a Markov Logic network to infer GRN and asymmetric bagging was utilized to handle the unbalanced training data [14]. Many neural network models have been also utilized to infer GRN [15–17].

The gene expression data used in supervised learning algorithms are all obtained by traditional sequencing technology, such as DNA microarray. However, biological tissue is composed of a variety of heterogeneous cells, and the differences between single cells may have a profound impact on the functions of multicellular organisms. In recent years, single-cell RNA-seq technology has been developed, which can be used for unbiased, repeatable, high-resolution and high-throughput transcription analysis of single cells [18–20]. Compared with the traditional transcriptome analysis of colony cells, single-cell RNA-seq technology can obtain the expression information of nearly 3000 genes in a single cell, which provides a powerful tool for distinguishing the transcriptome characteristics of various cell types in biological tissues, and comprehensively revealing the heterogeneity of gene expression between cells and the regulatory relationships between genes [21–24]. However, single-cell RNA-seq data has many shortcomings, such as high noise, many missing values, etc., so it is still challenging to reconstruct GRN using single-cell RNA-seq. Chan et al. proposed an information theory algorithm based on multivariate information measures to infer GRN according to single-cell data [25]. Nan et al. created time-stamped cross-sectional expression data and utilized regularized linear regression to identify GRN [26]. Matsumoto et al. proposed a novel GRN inference based on ordinary differential equation from single-cell RNA-seq [27].

In order to investigate the performances of supervised learning methods for GRN inference with single-cell RNA-seq data, we proposed a hybrid supervised learning method (Single_cell_GRN), which utilizes SVM, random forest, Naive Bayesian (NB), GBDT, logical regression (LR), decision tree (DT) and K-Nearest Neighbor (KNN) to infer gene regulatory network separately. For SVM, linear kernel, polynomial kernel and radial basis function are utilized and investigated. Three real single-cell RNA-seq datasets from mouse and human are utilized to test the supervised learning methods.

Methods

Supervised learning methods

Support vector machine

Support vector machine is a kind of machine learning method based on statistical theory, which was proposed by Vapnik [28]. It is mainly utilized to solve two-class classification

problems. The main idea is to map the samples to the high-dimensional feature space (kernel space), in which the linear classifier is constructed in order to obtain the largest interval [29]. Due to its advantages in solving small samples, and nonlinear and high-dimensional pattern recognition, SVM has been widely applied to text classification [30], bioinformatics [31, 32], financial data prediction [33], signal processing [34] and image processing [35].

The mechanism of SVM is to search the optimal hyperplane to meet the classification requirements. Two restricted conditions need to be considered, such as classification accuracy and maximizing the blank area on both sides of the hyperplane. So the learning process of SVM is an optimization problem.

Give the training dataset (x_i, y_i), $i = 1,2,\ldots,N$, N is the number of data, x_i is feature vector, and y_i is classification label $(+1, -1)$. Hyperplane is labeled as $(w{\cdot}x) + b = 0)$ w and b are coefficients and deviation term). The optimal hyperplane problem is constructed as follows.

$$\begin{aligned} &\min_{\alpha} \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j (x_i \cdot x_j) - \sum_{i=1}^{N}\alpha_i \\ &s.t. \quad \sum_{i=1}^{N}\alpha_i y_i = 0 \\ &\quad \alpha_i \geq 0 \end{aligned} \tag{1}$$

By solving the optimal problem, the optimal solution $\alpha^* = (\alpha_1^*, \alpha_2^*, \ldots, \alpha_N^*)^T$ is obtained. The optimal classification function could be also obtained as follows.

$$f(x) = \mathrm{sgn}\{\sum_{i=1}^{N}\alpha_i^* y_i (x_i \cdot x) + b^*\}. \tag{2}$$

Linear SVM utilizes hyperplane to divide two kinds of data. If the data itself is nonlinear, it is not suitable to use hyperplane as decision boundary. By kernel function, SVM can be applied to solve nonlinear classification problems. Kernel function is utilized to replace the inner product between two instances after a nonlinear transformation. The common kernel functions contain linear kernel, polynomial kernel and radial basis function (rbf), which are defined as followed.

$$K_{linear}(x, y) = x \cdot y. \tag{3}$$

$$K_{polynomial}(x, y) = ((x \cdot y) + 1)^d. \tag{4}$$

$$K_{rbf}(x, y) = \exp(-\gamma \left\|x - y\right\|^2). \tag{5}$$

Random forest

Random forest (RF) is a flexible and easy-to-use machine learning algorithm [36]. Compared with SVM, the selection of super parameters has less effect on the performance of RF, which is commonly utilized to solve classification and regression problems [37–40]. RF

was proposed based on ensemble learning method and decision tree. Its basic unit is decision tree, which is also a classifier. For an input sample, N trees could create N classification results. RF integrates all the classification results by voting method and specifies the category with the most voting times as the final output. The principle of RF is given as follows.

(1) Firstly M samples are randomly selected from the sample set by bootstrap algorithm. For each sample, K features are selected randomly from all attributes. According to the selected K features, a decision tree is established.
(2) Repeat step (1) N times in order to obtain N decision trees.
(3) Input variables are given to each decision tree, which could output a result. N decision trees could get N classification results.
(4) Calculate the number of votes of all classes and select the classification result with the highest number of votes as the final category.

Naive Bayesian

Naive Bayesian is built on Bayes' theorem and is a typical generative learning method [41]. The main idea is to adopt the attribute conditional independence assumption. It assumes that all attributes are independent of each other, and the impact of different attributes on the classification results is irrelevant. The algorithm can not only simplify the calculation and be easy to implement, but also has good robustness. It is commonly used in statistical decision-making fields such as text document classification [42] and medical diagnosis [43].

Bayesian theorem is expressed as follows:

$$P(c|x) = \frac{P(c)P(x|c)}{P(x)}. \tag{6}$$

where c denotes class, $P(c)$ represents a priori probability, $P(c \mid x)$ indicates a posteriori probability, $P(x \mid c)$ denotes the class conditional probability and $P(x)$ is the edge probability of x.

Based on the assumption of attribute conditional independence, Eqs. (6) can be rewritten as:

$$P(x|c) = \frac{P(c)}{P(x)} \prod_{i=1}^{d} P(x_i|c). \tag{7}$$

where d is the number of attributes and x_i is the value on the i-th attribute.

Because the denominator in Eqs. 7 is the same for all categories, it has no impact on the result. Therefore, the simplified formula of Naive Bayes is defined as follows.

$$c(x) = \underset{c \in C}{argmax}\, P(c) \prod_{i=1}^{d} P(x_i|c). \tag{8}$$

GBDT

GBDT is a type of ensemble learning method [44] and an algorithm with strong generalization ability [45, 46]. The main idea is to use the negative gradient of the loss

function to simulate the residual, and take the residual of the previous tree as the input of the next tree. In each iteration, the loss decreases rapidly along the negative gradient direction, and finally accumulates the prediction results of all trees as the final result of the model.

The training dataset is $T= \{(x_1,y_1),(x_2,y_2),\cdots, (x_N,y_N)\}$, and the loss function is $L(y, f(x))$, where xi represents the feature vector and y is the label. The main flowchart of the algorithm is indicated as follows.

(1) Initialize weak learner.

$$f_0(x) = \arg\min_c \sum_{i=1}^{N} L(y_i, c). \tag{9}$$

(2) For $m= 1,2,\cdots,M$, where M is the number of iterations.

(a) Calculate the negative gradient of the loss function in the current tree, the residual is written as:

$$r_{mi} = -[\frac{\partial L(y_i, f(x_i))}{\partial f(x_i)}]_{f(x)=f_{m-1}(x)}. \tag{10}$$

(b) Fit a regression tree to the target r_{mi} and compute the leaf node region R_{mj} $(j = 1, 2, \cdots, J)$ of the regression tree.

(c) For $j = 1, 2, \cdots, J$, the optimal coefficient of leaf node region is calculated.

$$c_{mj} = \underset{c}{argmin} \sum_{x_i \in R_{mj}} L(y_i, f_{m-1}(x_i) + c). \tag{11}$$

(d) The strong learner in this iteration is obtained.

$$f_m(x) = f_{m-1}(x) + \sum_{J}^{j=1} c_{mj} I(x \in R_{mj}). \tag{12}$$

(3) After all the iterations, the strong learner is obtained.

$$f_M(x) = \sum_{m=1}^{M} \sum_{j=1}^{J} c_{mj} I(x \in R_{mj}). \tag{13}$$

Logical regression

Logistic regression [47] is an important statistical model in machine learning and has been widely used in biology, epidemiology and other fields [48, 49]. Logical regression consists of linear regression and Sigmoid function. The continuous values of the regression results are allocated between 0 and 1 in order to solve the classification problems. The specific process is as follows.

(1) Firstly, assuming that x is the input vector, θ is the parameters to be solved, and y represents the prediction result of linear regression, the linear regression model is given as follows.

$$y = \theta_0 + \theta_1 x_1 + ... + \theta_n x_n = \sum_{i=1}^{n} \theta_i x_i = \theta^T x. \tag{14}$$

(2) In LR, the logic function is Sigmoid function, which is defined as follows.

$$g(z) = \frac{1}{1 + e^{-z}}. \tag{15}$$

According to Sigmoid function, the output of model is constructed as follows.

$$h_\theta(x) = \frac{1}{1 + e^{-\theta^T x}} \tag{16}$$

The parameter θ can be estimated by the maximum likelihood method, and the final model can be obtained by continuously optimizing the parameters through inputting the test samples.

Decision tree

Decision tree is a machine learning method used to solve classification problems [50]. It is a tree structure that divides the data by making a series of decisions. A decision tree contains root node, internal nodes and leaf nodes. The decision-making process of the decision tree starts from the root node. By testing the corresponding characteristic attributes of the items to be classified, the output branches are selected according to the results. The generation of decision tree is a recursive process. Each step will pick up the optimal selection of the current state until the leaf node has been selected. Finally take the category stored in the leaf node as the final decision result.

In the generation process of decision tree, the key step is the measurement of feature selection. The feature selection is based on the principle that the samples contained in the branch nodes belong to the same category as much as possible. At present, there are three main algorithms for the construction of a decision tree, namely ID3, C4.5 and CART. In this paper, we select CART algorithm [51, 52].

CART algorithm utilizes binary recursive segment method to divide the sample set into two sub sample sets, which contains feature selection and tree pruning. *Gini* index is utilized to select the features, determine the optimal partition points and measure the purity of dataset D, which is defined as follows.

$$Gini(D) = \sum_{i=1}^{n} p(x_i)(1 - p(x_i)) = 1 - \sum_{i=1}^{n} p(x_i)^2. \tag{17}$$

where $p(x_i)$ is the probability of category x_i , and n is the number of categories in D. *Gini* (D) reflects the probability that the category labels of two samples are inconsistent, which are randomly selected from dataset D. Therefore, the smaller the *Gini* (D) is, the higher the purity of the dataset D is.

K-Nearest Neighbor

K-Nearest Neighbor is a commonly used machine learning algorithm [53, 54]. It can be utilized to solve classification and regression problems, and is widely used in data mining and pattern recognition. The algorithm idea is to identify the K training samples of the known categories that are most similar to the test sample based on some distance measures in the sample space. Then judge the category of the sample based on the information of K neighbors. The main algorithm flowchart of KNN is given as follows.

(1) Build the training sample set and calculate the distances between the test sample and the training samples based on the distance measurement.
(2) Sort the training samples in ascending order according to the distances.
(3) Select the K training samples closest to the test one as the K neighbors of the test sample.
(4) Count the category frequencies of K neighbors, and select the category with the highest frequency as the category of the test sample.

GRN inference with single-cell RNA-seq data and supervised learning method

For the inference of gene regulatory networks, the complex regulatory relationships among genes are identified, which could be evolved to two-class problems. Single-cell RNA-seq data and the corresponding regulatory relationships between genes are collected from the public databases. Count up the number of TF as N_{TF}. With the regulatory networks verified by biology experiments from the well-known databases, for each regulatory factor i, all the target genes set can be divided into two categories. The target genes regulated by the regulator factor i are marked as positive gene set, while the target genes not regulated by the regulator factor i are marked as negative gene set. The single-cell RNA-seq data of two kinds of gene sets are constructed. K-fold cross validation method is utilized to divide the training and testing datasets in order infer the regulatory relationships between all target genes and regulatory factors. For each classification problem, different classification methods are selected. If the number of positive samples is zero, the sample is classified as negative, which reveal that there are no regulatory relationships between the regulator factor i and the targets. Otherwise SVM, RF, NB, GBDT, LR, DT and KNN are utilized, respectively. When the regulations of all regulatory factors have been inferred, the algorithm stop; otherwise repeat the above process. The regulations of all TFs are integrated in order to obtain the overall GRN. The flowchart is depicted in Fig. 1.

Experiments and discussions

Three real single-cell RNA-seq datasets are utilized to test our methods. The first dataset is derived from primitive endoderm cells differentiated from mouse embryonic stem cells, which includes 456 cells (Data1) [55]. The second dataset was derived from scRNA-Seq data obtained to examine direct reprogramming from mouse

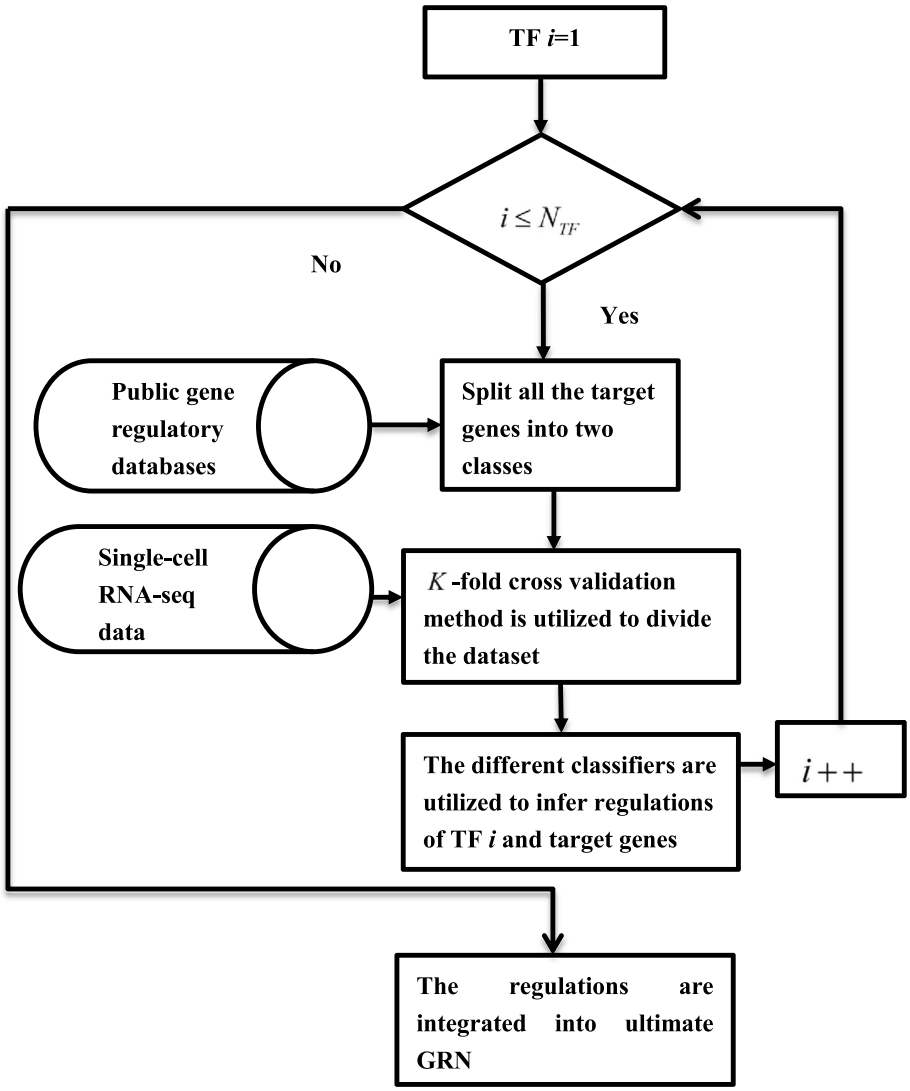

Fig. 1 The flowchart of gene regulatory network inference with single-cell RNA-seq data and supervised learning method

embryonic fibroblast (MEF) cells to myocytes, which includes 405 cells (Data2) [56]. The third dataset was derived from definitive endoderm (DE) cells differentiated from human ES cells, which includes 758 cells (Data3) [57]. Three extracted sub networks and the validation regulatory relationships are from the previous study [27].

Receiver Operating Characteristic (ROC) curve considers true positive rate (TPR) and false positive rate (FPR), and could accurately reflect the relationship between TPR and FPR of a learner, which is a comprehensive manner to evaluate model sensitivity and specificity. TPR denotes the proportion of the inferred real regulatory relationships in all real regulations. FPR represents the proportion of the inferred false-positive regulatory relationships in all the true non-regulations. Area Under ROC Curve (AUC) is the area covered by ROC curve, which could reflect the performance of the learner more intuitively. In this part, ROC curves and AUC are utilized to evaluate our methods.

Results

In this part SVM with different kernel functions (linear kernel (SVM + linear), polynomial kernel (SVM + poly) and radial basis function (SVM + rbf)), RF, NB, GBDT, LR, DT and KNN are utilized to infer GRN with three real single-cell RNA-seq data, respectively. Leave-One-Out Cross Validation (LOOCV) is utilized to classify the unknown regulatory relationships. To better evaluate the performance of supervised learning algorithms, the famous unsupervised learning method (GENIE3) is also utilized to infer the same GRNs, which has the highest performance in the DREAM3 Challenge. The ROC curves and the corresponding AUC values of ten methods are depicted in Figs. 2, 3 and 4, respectively. For Data1, SVM with polynomial kernel has the highest AUC value, which is 0.5% higher than SVM + rbf, 0.7% higher than SVM + linear, 13.3% higher than RF, 14% higher than DT, 14.6% higher than GBDT, 9.4% higher than KNN, 9.5% higher than LR, 10.5% higher than NB and 11.8% higher than GENIE3. From the results of ROC and AUC, SVM methods with three different kernel functions perform better than RF, DT, GBDT, KNN, LR and NB. GENIE3 performs better than RF, DT and GBDT, worse than other six classifiers. For Data2, in terms of ROC curve, KNN and RF have the similar performances, which are better than other eight methods. In terms of AUC, KNN has the best performance, which is 2.9% higher than SVM + poly, 7.8% higher than SVM + rbf, 2.7% higher than SVM + linear, 2.1% higher than RF, 17.3% higher than DT, 10.5% higher than GBDT, 30.3% higher than LR, 7.2% higher than NB and 23.4% higher than GENIE3. Unsupervised learning method (GENIE3) and LR have lower AUC value than other eight supervised learning methods, which are less than 0.5. For Data3, RF has the highest AUC value, which is 5.3% higher than SVM + poly, 5.1% higher than SVM + linear, 11.3% higher than SVM + rbf, 17.1% higher than DT, 11.9% higher than GBDT, 2.9% higher than KNN, 17.7% higher than LR, 13.1% higher than NB, and 22.1% higher than GENIE3. GENIE3 has the worst performance. Combined with ROC curves,

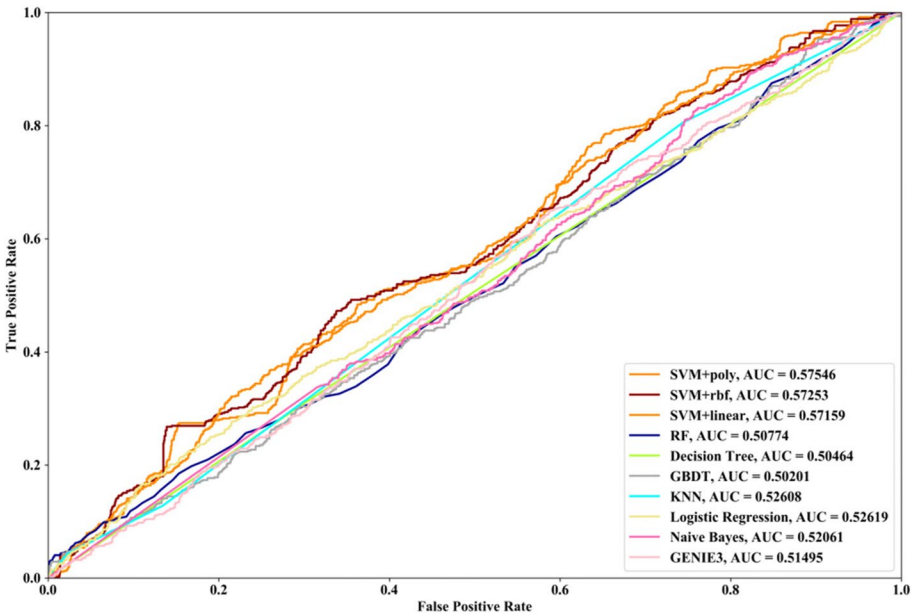

Fig. 2 AUC and ROC performances of ten methods by LOOCV with Data1 for GRN inference

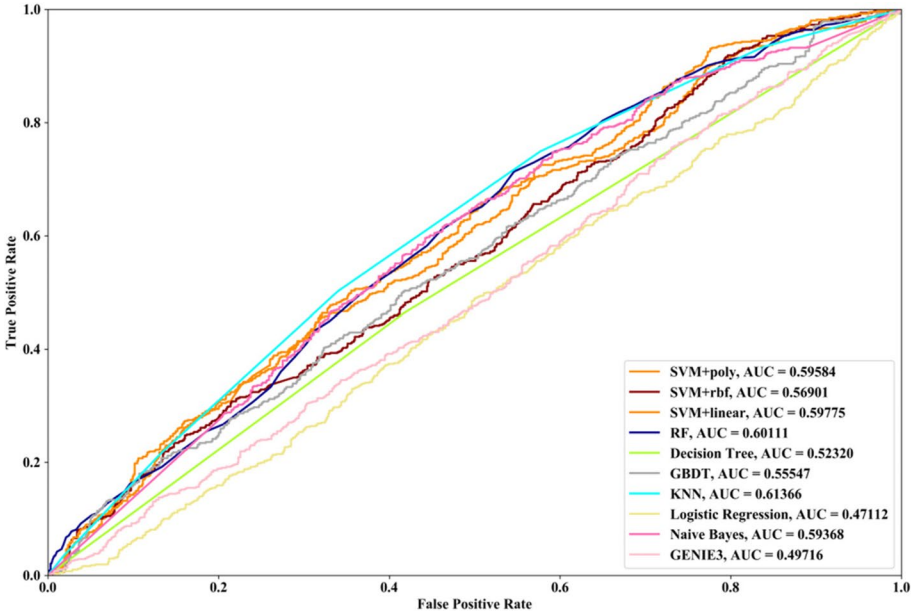

Fig. 3 AUC and ROC performances of ten methods by LOOCV with Data2 for GRN inference

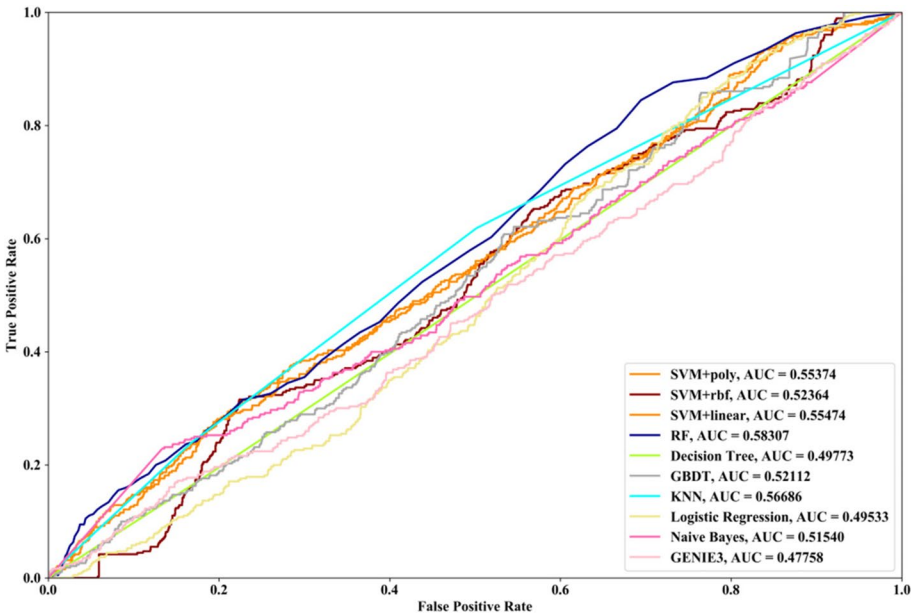

Fig. 4 AUC and ROC performances of ten methods by LOOCV with Data3 for GRN inference

we can see that in most cases the performance GENIE3 is less than random and supervised learning methods perform better than unsupervised learning method.

Discussions

Compared with the transcriptome data by traditional sequencing technologies, sing-cell RNA-seq data has its own internal characteristics. In this part, we compare

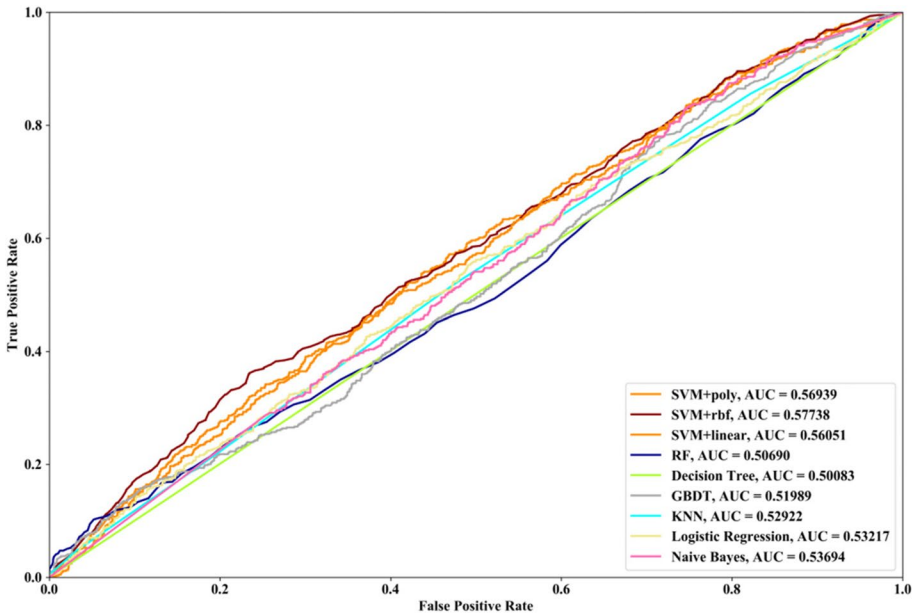

Fig. 5 AUC and ROC performances of nine classifiers with Data1 and 3-cross validation method for GRN inference

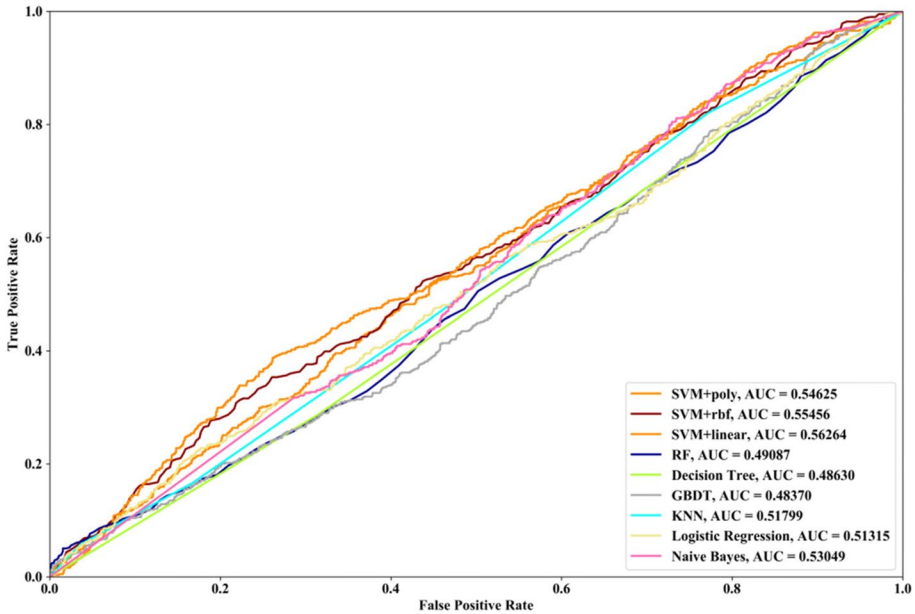

Fig. 6 AUC and ROC performances of nine classifiers with Data1 and 5-cross validation method for GRN inference

the performances of SVM methods with different kernel functions in our proposed method. We also compare SVM with RF, NB, GBDT, DT, LR and KNN. threefold cross validation, fivefold cross validation and tenfold cross validation are utilized and the AUC results and ROC curves of nine methods with three datasets are depicted in Figs. 5, 6, 7, 8, 9, 10, 11, 12 and 13, respectively. For threefold cross validation results,

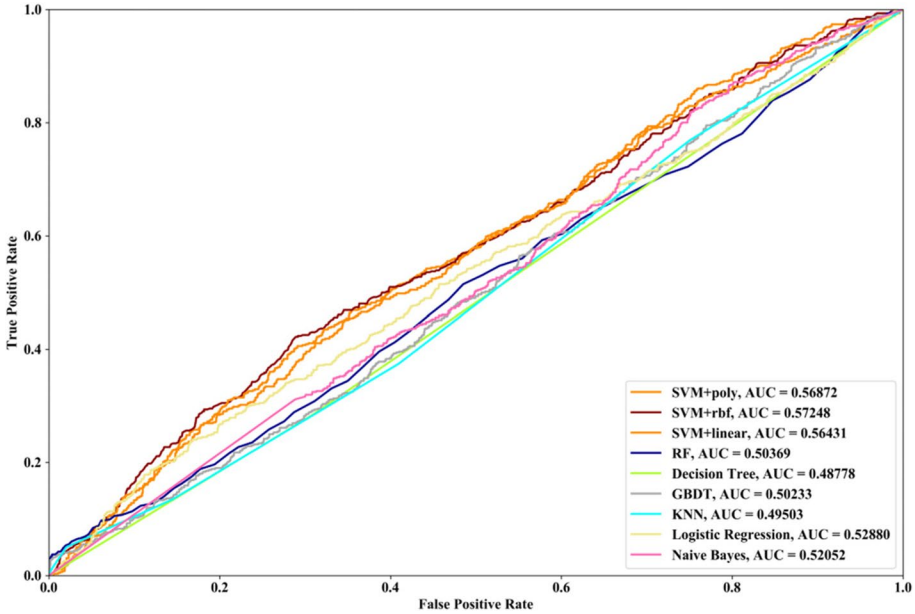

Fig. 7 AUC and ROC performances of nine classifiers with Data1 and 10-cross validation method for GRN inference

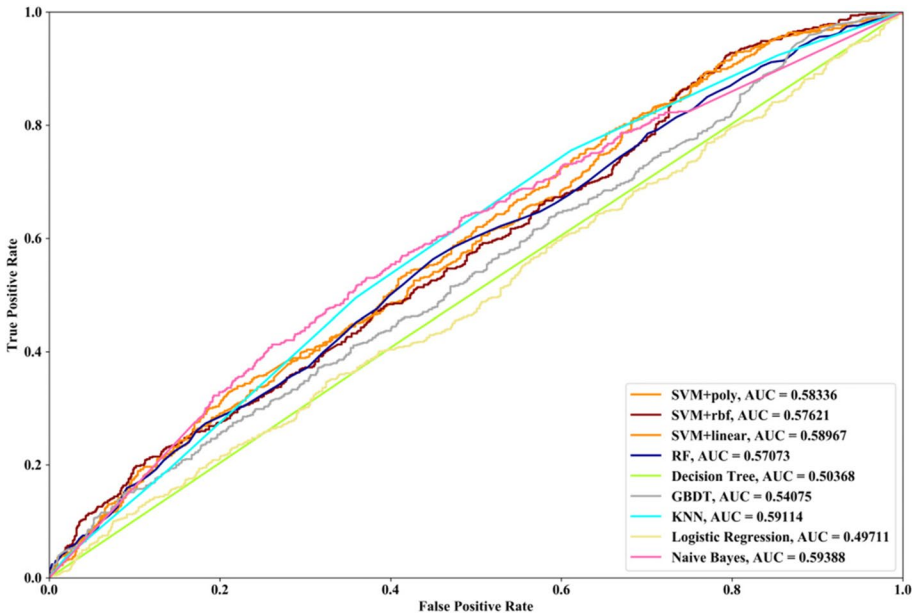

Fig. 8 AUC and ROC performances of nine classifiers with Data2 and 3-cross validation method for GRN inference

SVM + rbf, NB and RF have the highest AUC values with Data1, Data2 and Data3, respectively. For fivefold cross validation results, with Data1 SVM + linear is 1.45% higher than SVM + rbf, 3% higher than SVM + poly, 14.6% higher than RF, 15.7% higher than DT, 16.3% higher than GBDT, 8.6% higher than KNN, 9.6% higher than LR and 6.1% higher than NB. With Data2, KNN has the highest AUC value, which is 0.63703. With Data3, RF also has the highest AUC value, which is 7% higher than

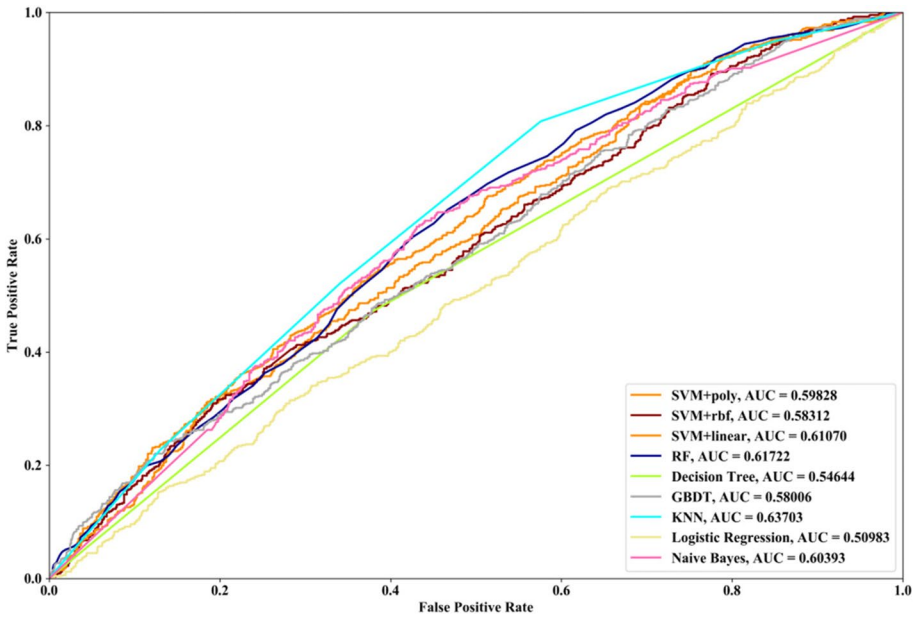

Fig. 9 AUC and ROC performances of nine classifiers with Data2 and 5-cross validation method for GRN inference

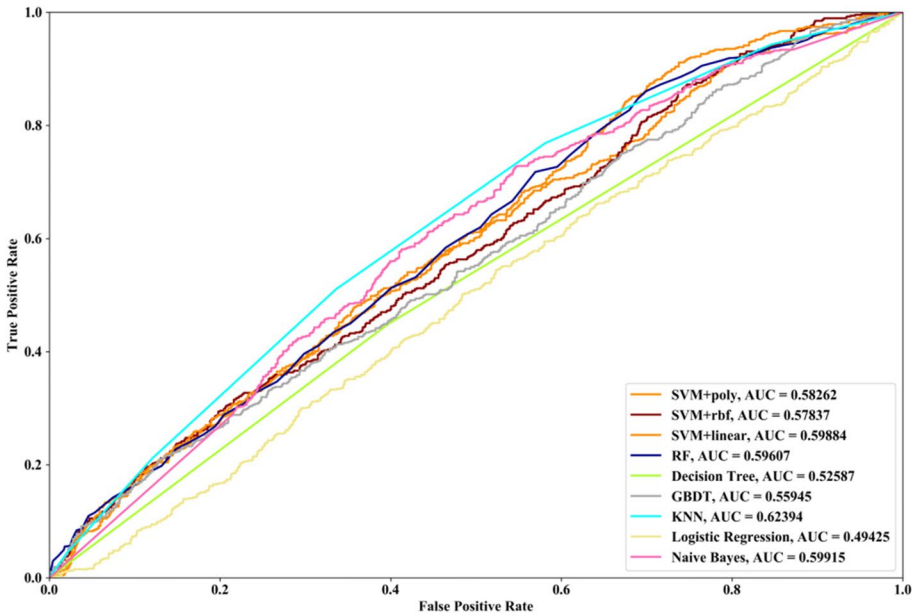

Fig. 10 AUC and ROC performances of nine classifiers with Data2 and 10-cross validation method for GRN inference

SVM + rbf, 6.7% higher than SVM + linear, 5.4% higher than SVM + poly, 16.8% higher than DT, 6.6% higher than GBDT, 3.8% higher than KNN, 26.9% higher than LR and 15.4% higher than NB. From tenfold cross validation results, it could be seen that SVM + rbf is 0.66% higher than SVM + poly, 1.4% higher than SVM + linear, 13.7% higher than RF, 17.4% higher than DT, 14% higher than GBDT, 15.6% higher than KNN, 8.3% higher than LR and 1.0% higher than NB with Data1. With Data2 and

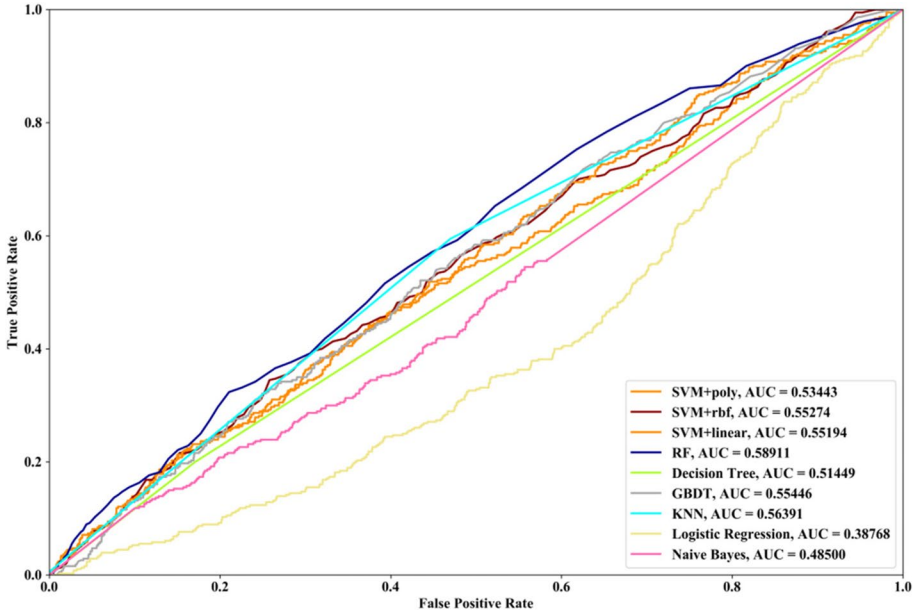

Fig. 11 AUC and ROC performances of nine classifiers with Data3 and 3-cross validation method. for GRN inference

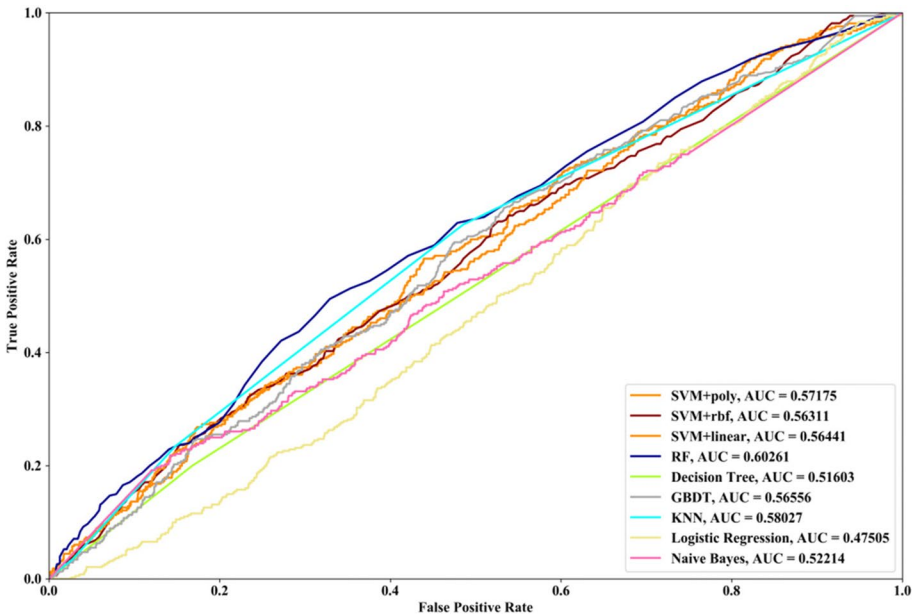

Fig. 12 AUC and ROC performances of nine classifiers with Data3 and 5-cross validation method for GRN inference

Data3, KNN and RF have the higher AUC performances than other eight methods, respectively.

In order to compare the performances of different supervised learning methods for GRN inference obviously, we rank these nine methods according to the performances of LOOCV (Figs. 2, 3 and 4), threefold cross validation (Figs. 5, 8 and 11), fivefold

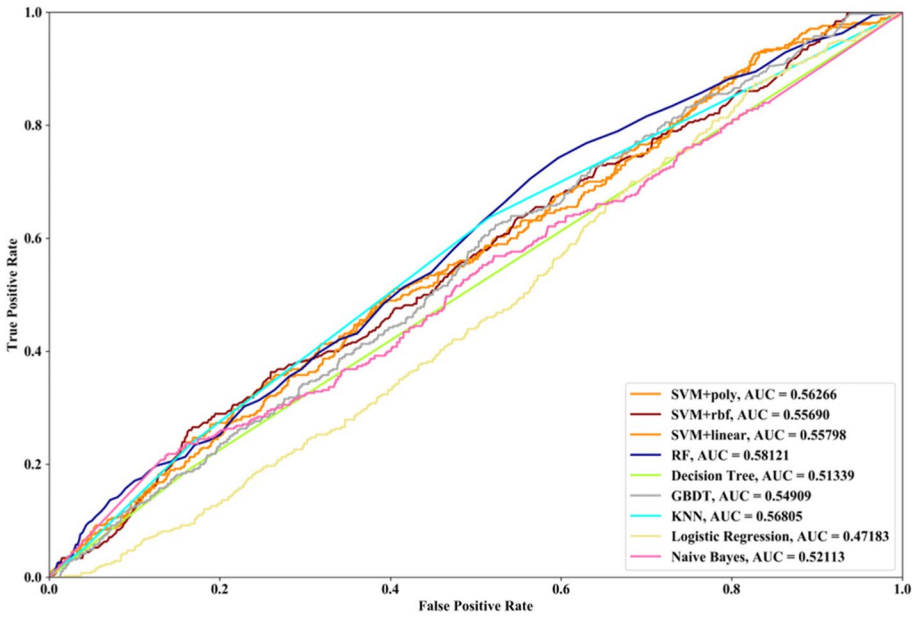

Fig. 13 AUC and ROC performances of nine classifiers with Data3 and 10-cross validation method for GRN inference

Table 1 Ranking performances of nine methods with three datasets

		SVM + poly	SVM + rbf	SVM + linear	RF	DT	GBDT	KNN	LR	NB
LOOCV	Data1	1	2	3	7	8	9	5	4	6
	Data2	4	6	3	2	8	7	1	9	5
	Data3	4	5	3	1	8	6	2	9	7
threefold cross validation	Data1	2	1	3	9	8	7	6	5	4
	Data2	4	5	3	6	8	7	2	9	1
	Data3	6	4	5	1	7	3	2	9	8
fivefold cross validation	Data1	3	2	1	7	8	9	5	6	4
	Data2	5	6	3	2	8	7	1	9	4
	Data3	3	6	5	1	8	4	2	9	7
tenfold cross validation	Data1	2	1	3	6	9	7	8	4	5
	Data2	5	6	3	4	8	7	1	9	2
	Data3	3	5	4	1	8	6	2	9	7
Average ranking		3.5	4.1	3.25	3.9	8	6.6	3.1	7.6	5

cross validation (Figs. 6, 9 and 12) and tenfold cross validation (Figs. 7, 10 and 13) with three datasets. The ranking results are listed in Table 1. From Table 1, it could be clearly seen that in most cases SVM, RF and KNN methods have the highest ranking performances among nine classifiers, which show that these three methods could infer gene regulatory network more accurately. DT and LR have worse performances than other seven methods for gene regulatory network inference. Among SVM methods with three kernel functions, SVM methods with linear kernel and polynomial kernel have the higher ranking performances than SVM with rbf kernel, which prove that linear and polynomial functions are fitter to model single-cell RNA-seq data.

Conclusions

In this paper, a hybrid supervised learning method based on SVM, RF, NB, GBDT, LR, DT and KNN is utilized to solve the binary classification problem of gene regulatory network inference. In SVM, three different kernel functions (linear, polynomial and radial basis function) are also utilized. Three real single-cell RNA-seq datasets from mouse and human are utilized to test these supervised learning methods. Nine supervised learning methods and one unsupervised learning method are utilized. With Data1, Data2 and Data3, in terms of AUC, SVM, KNN and RF are 0.5%-14%, 2.1%-30.3% and 2.9%-22.1% higher than other nine methods, respectively. The inference results prove that in most cases supervised learning methods (SVM, RF, NB, GBDT, LR, DT and KNN) have the better ROC and AUC performances than unsupervised learning method (GENIE3).

We also compare the performances of SVM methods with different kernel functions, RF, NB, GBDT, LR, DT and KNN further. threefold cross validation, fivefold cross validation and tenfold cross validation are utilized. The results show that in most cases SVM, RF and KNN methods have the best performances among nine classifiers. Among SVM methods with three kernel functions, SVM methods with linear kernel and polynomial kernel have the better performance than SVM with rbf kernel, which prove that linear and polynomial functions are fitter to model single-cell RNA-seq data than rbf kernel.

Acknowledgements

This work was supported by the talent project of "Qingtan Scholar" of Zaozhuang University,the Natural Science Foundation of China (No. 61902337), the fundamental Research Funds for the Central Universities, 2020QN89, Xuzhou science and technology plan project, KC19142, KC21047. Shandong Provincial Natural Science Foundation, China (No. ZR2015PF007), Jiangsu Provincial Natural Science Foundation (No. SBK2019040953), Natural Science Fund for Colleges and Universities in Jiangsu Province (No. 19KJB520016) and Young talents of science and technology in Jiangsu.

Authors' contributions

W.B. conceived the method. B.Y designed the method. D.S. designed the website of this algorithm. B.Y. conducted the experiments and W.B. and B.C. wrote the main manuscript text. All authors reviewed the manuscript. The author(s) read and approved the final manuscript.

Availability of data and materials

The data used to support the findings of this study are available from the corresponding author upon request.

Declarations

Competing interests

The authors declare no competing interests.

Received: 9 February 2022 Accepted: 22 May 2022
Published online: 11 June 2022

References

1. Unwin N, Samuels TA, Rose AM, Hennis AJ. Cardiovascular and Vascular Disease in the Tropics Including Stroke, Hypertension and Ischaemic Heart Disease. In: Mansons Tropical Infectious Diseases. 2014. p. 854–72.
2. Chun JN, Lim JM, Kang Y, Kim EH, Shin YC, Kim HG, Jang D, Kwon D, Shin SY, So I, Jeon JH. A network perspective on unraveling the role of TRP channels in biology and disease. Pfluegers Arch. 2014;466(2):173–82.
3. Wang J, Sen S. MicroRNA functional network in pancreatic cancer: from biology to biomarkers of disease. J Biosci. 2011;36(3):481–91.
4. Hu JX, Thomas CE, Brunak S. Network biology concepts in complex disease comorbidities. Nat Rev Genet. 2016;17(10):615–29.
5. Fazilaty H, Rago L, Youssef KK, et al. A gene regulatory network to control EMT programs in development and disease. Nat Commun. 2019;10(1):5115.
6. Sumantra C, Aravinda C. A gene regulatory network explains RET–EDNRB epistasis in Hirschsprung disease. Hum Mol Genet. 2019;28(18):3137–47.
7. Crespo I, Roomp K, Jurkowski W, et al. Gene regulatory network analysis supports inflammation as a key neurodegeneration process in prion disease. BMC Syst Biol. 2012;6:132.

8. Zickenrott S, Angarica VE, Upadhyaya BB, et al. Prediction of disease–gene–drug relationships following a differential network analysis. Cell Death Dis. 2016;7(1):e2040.
9. Maetschke SR, Madhamshettiwar PB, Davis MJ, et al. Supervised, semi-supervised and unsupervised inference of gene regulatory networks. Brief Bioinform. 2014;15(2):195–211.
10. Wang Y, Joshi T, Xu D, Zhang XS, Chen L. Supervised Inference of Gene Regulatory Networks by Linear Programming. In: Huang DS, Li K, Irwin GW, editors. Computational Intelligence and Bioinformatics. Berlin, Heidelberg: Springer; 2006. ICIC 2006. Lecture Notes in Computer Science, vol 4115.
11. Mordelet F, Vert JP. SIRENE: supervised inference of regulatory networks. Bioinformatics. 2008;24:i76-82.
12. Cerulo L, Elkan C, Ceccarelli M. Learning gene regulatory networks from only positive and unlabeled data. BMC Bioinformatics. 2010;11:228.
13. Gillani Z, Akash MSH, Rahaman M, et al. CompareSVM: supervised, Support Vector Machine (SVM) inference of gene regularity networks. BMC Bioinformatics. 2014;15(1):395.
14. Brouard C, Vrain C, Dubois J, et al. Learning a Markov Logic network for supervised gene regulatory network inference. BMC Bioinformatics. 2013;14(1):273.
15. Yang B. A new supervised learning for gene regulatory network inference with novel filtering method. Int J Perform Eng. 2018;14(5):945–54.
16. Yang B, Zhang W. Supervised Learning for Gene Regulatory Network Based on Flexible Neural Tree Model. In: Communications in Computer and Information Science. 2017. p. 293–301.
17. Liu S, Yang B, Wang H. Inference of gene regulatory network based on radial basis function neural network. LNCS. 2016;10122:442–50.
18. Zeisel A, Muñoz-Manchado AB, Codeluppi S, et al. Brain structure. Cell types in the mouse cortex and hippocampus revealed by single-cell RNA-seq. Science. 2015;347(6226):1138–42.
19. Treutlein B, Brownfield DG, Wu AR, et al. Reconstructing lineage hierarchies of the distal lung epithelium using single-cell RNA-seq. Nature. 2014;509(7500):371–5.
20. Islam S, Zeisel A, Joost S, et al. Quantitative single-cell RNA-seq with unique molecular identifiers. Nat Methods. 2013;11(2):163–6.
21. Sebé-Pedrós A, Baptiste S, Elad C, et al. Cnidarian cell type diversity and regulation revealed by whole-organism single-cell RNA-Seq. Cell. 2018;173(6):1520–34.
22. Hook PW, Mcclymont SA, Cannon GH, et al. Single-cell RNA-seq of mouse dopaminergic neurons informs candidate gene selection for sporadic Parkinson disease. Am J Hum Genet. 2018;102(3):427–46.
23. Aibar S, González-Blas CB, Moerman T, et al. SCENIC: single-cell regulatory network inference and clustering. Nat Methods. 2017;14(11):1083–6.
24. Karbalayghareh A, Braga-Neto U, Dougherty ER. Intrinsically Bayesian robust classifier for single-cell gene expression trajectories in gene regulatory networks. BMC Syst Biol. 2018;12(Suppl 3):23.
25. Chan TE, Stumpf MPH, Babtie AC. Gene regulatory network inference from single-cell data using multivariate information measures. Cell Syst. 2017;5(3):251–67.
26. Nan PG, Minhaz UDSM, Olivier G, et al. SINCERITIES: inferring gene regulatory networks from time-stamped single cell transcriptional expression profiles. Bioinformatics. 2018;34(2):258–66.
27. Matsumoto H, Kiryu H, Furusawa C, et al. SCODE: an efficient regulatory network inference algorithm from single-cell RNA-Seq during differentiation. Bioinformatics. 2017;33(15):2314–21.
28. Guyon I, Weston J, Barnhill S, et al. Gene selection for cancer classification using support vector machines. Mach Learn. 2002;46:389–422.
29. Joachims T. Support Vector Machines. In: Learning to Classify Text Using Support Vector Machines. The Springer International Series in Engineering and Computer Science. Boston: Springer; 2002. vol 668.
30. Tong S, Koller D. Support vector machine active learning with applications to text classification. J Mach Learn Res. 2002;2(1):999–1006.
31. Magnin B, Mesrob L, Kinkingnéhun S, et al. Support vector machine-based classification of Alzheimer's disease from whole-brain anatomical MRI. Neuroradiology. 2009;51(2):73–83.
32. Orrù G, Pettersson-Yeo W, Marquand AF, et al. Using Support vector machine to identify imaging biomarkers of neurological and psychiatric disease: a critical review. Neurosci Biobehav Rev. 2012;36(4):1140–52.
33. Huang W, Nakamori Y, Wang SY. Forecasting stock market movement direction with support vector machine. Comput Oper Res. 2005;32(10):2513–22.
34. Chen S, Samingan AK, Hanzo L. Support vector machine multiuser receiver for DS-CDMA signals in multipath channels. IEEE Trans Neural Netw. 2001;12(3):604–11.
35. Gomez-Perez G, Camps-Valls G, Gutierrez J, et al. Perceptual adaptive insensitivity for support vector machine image coding. IEEE Trans Neural Netw. 2005;16(6):1574–81.
36. Breiman L. Random forest. Mach Learn. 2001;45:5–32.
37. Ham J, Chen Y, Crawford MM, et al. Investigation of the random forest framework for classification of hyperspectral data. IEEE Trans Geosci Remote Sens. 2005;43(3):492–501.
38. Rodriguez-Galiano VF, Ghimire B, Rogan J, et al. An assessment of the effectiveness of a random forest classifier for land-cover classification. ISPRS J Photogramm Remote Sens. 2012;67:93–104.
39. Xia X, Togneri R, Sohel F, et al. Random forest classification based acoustic event detection utilizing contextual-information and bottleneck features. Pattern Recogn. 2018;81:1–13.
40. Balachandran M, Shin TH, Kim MO, et al. AIPpred: sequence-based prediction of anti-inflammatory peptides using random forest. Front Pharmacol. 2018;9:276.
41. Leung KM. Naive bayesian classifier. Polytechnic University Department of Computer Science/Finance and Risk Engineering. 2007;2007:123–56.
42. Singh G, et al. "Comparison between multinomial and Bernoulli naïve Bayes for text classification." 2019 International Conference on Automation, Computational and Technology Management (ICACTM). Piscataway: IEEE; 2019.
43. Choudhury A, Gupta D. A survey on medical diagnosis of diabetes using machine learning techniques. In: Recent developments in machine learning and data analytics. Singapore: Springer; 2019. p. 67–78.

44. Dietterich TG. Ensemble learning. In: The handbook of brain theory and neural networks 2.1. 2002. p. 110–25.
45. Ke G, et al. DeepGBM: A deep learning framework distilled by GBDT for online prediction tasks. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2019.
46. Liang W, et al. Predicting hard rock pillar stability using GBDT, XGBoost, and LightGBM algorithms. Mathematics. 2020;8:765–5.
47. Wright RE. Logistic regression. 1995.
48. Hosmer Jr DW, Lemeshow S, Sturdivant RX. Applied logistic regression. Vol. 398. New York: Wiley; 2013.
49. Pal S, Talukdar S. Application of frequency ratio and logistic regression models for assessing physical wetland vulnerability in Punarbhaba river basin of Indo-Bangladesh. Hum Ecol Risk Assess Int J. 2018;24(5):1291–311.
50. Safavian SR, Landgrebe D. A survey of decision tree classifier methodology. IEEE Trans Syst Man Cybern. 1991;21(3):660–74.
51. Hasanpoor D, et al. Applying Rough Developed theoretical Models (ERST), Interpretation-Structural Analysis (ISM) and Decision Tree (CART) for Help Auditors to Identify Fraud in the Financial Statements of Companies Listed on the Stock Exchange of Iran. J Investment Knowledge. 2020;9(33):179–208.
52. Li M. "Application of CART decision tree combined with PCA algorithm in intrusion detection." 2017 8th IEEE international conference on software engineering and service science (ICSESS). Piscataway: IEEE; 2017.
53. Gazalba I, Reza NGI. "Comparative analysis of k-nearest neighbor and modified k-nearest neighbor algorithm for data classification." 2017 2nd International conferences on Information Technology, Information Systems and Electrical Engineering (ICITISEE). Piscataway: IEEE; 2017.
54. Gou J, et al. A generalized mean distance-based k-nearest neighbor classifier. Expert Syst Appl. 2019;115:356–72.
55. Shimosato D, Shiki M, Niwa H. Extra-embryonic endoderm cells derived from ES cells induced by GATA factors acquire the character of XEN cells. BMC Dev Biol. 2007;7:80.
56. Treutlein B, Lee QY, Camp JG. Dissecting direct reprogramming from fibroblast to neuron using single-cell RNA-seq. Nature. 2015;534(7607):391–5.
57. Chu LF, Leng N, Zhang J. Single-cell RNA-seq reveals novel regulators of human embryonic stem cell differentiation to definitive endoderm. Genome Biol. 2016;17(1):173.

Publisher's Note