Project Report of VE492

Introduction to Artificial Intelligence

Gene Chip Data Analysis by Machine Learning

SUN Shangquan

SJTU-Michigan Joint Institute

Shanghai Jiao Tong University

515370910068

527112517@sjtu.edu.cn

ABSTRACT

In the study of human gene chips, there is a strong and potential interaction between mysterious genes and possibility of having a cancer, since cancer could be probably caused by different orders and appearances of genes. However, it's so difficult for human intelligence to classify the potential relation. Over the past decades, thanks largely to the rapid improvement of machine learning and biological techniques, we could now explore all the genes (more than twenty-two thousands) of more than five thousand persons. Among them, more than two thousand and five hundred people have cancers, while more than one thousand do not, which is pretty adequate for us to develop a model revealing the supposed relation.

Keywords

Machine Learning, Gene Chip, Deep Learning, Cancer, Tumor, Neural Network, CPA.

1. INTRODUCTION

1.1 Related Work

Prof. Yuan Bo[1] has contributed to the exploration of whole human gene chips (more than sixty thousand) in 2001. This great work has provided a solid foundation of background information about human genes for our project. Also, a framework of machine learning from Professor Bo Yuan[2] has helped us establish our own classical machine learning models and deep learning framework.

1.2 Our Contributions

In this project, multiple classical machine learning algorithms and a deep learning algorithm have been utilized. Classical algorithms include Linear Discriminant Analysis, Random Forest, Decision Tree, Logistic Regression, SVM (Support Vector Machine) and KNN (k-Nearest Neighbor). The deep learning algorithm is a full connected Back Propagation Neural Network with three or four layers.

In the part of Dimension Reduction of features, the algorithm of PCA (Principle Component Analysis) is used in this project.

The data[4] included in the project contain information of 22283 genes from 5896 people as well as their physical check data, such as disease state, disease stage and so on, while most of physical data are empty or blank. Therefore, in this project, only filled and available labels are taken into consideration.

The goal of project is to predict whether one could have cancer or not given his or her gene chip data as

features. Therefore, we are going to generate the relation between one's cancer state and his or her gene chip data. Generally, we only consider binary class classification problem in this project, which means that we just predict whether one has cancer or not. Those diseases rather than cancer will be counted as 0 while all cases of cancer or tumor are counted as 1.

2. Data Processing

1.1 Dimension Reduction

Principal Component Analysis (PCA) is the basic method to reducing dimensions of data, especially dimensions of features, in order to have a better performance when we train our model. In the project, we set the threshold of summation of latent for k-dimension 95%. Therefore, we could get data in the shape of 5896 by 2134, which means that the dimension of features has been reduced from 22283 to 2134, which is shown in Figure 1.

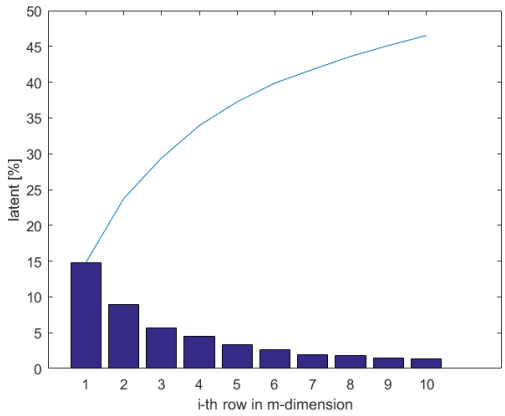

Figure 1. First ten latent for PCA and their accumulation in the line chart

1.2 Normalization and Standardization

Normalization and Standardization are two main ways of mapping data within a large range into data between a narrow bandwidth. Specifically, Normalization is to map data into a range between -1 to 1, while Standardization is to map data into a range between 0 to 1. Two equations are listed as followed.

$$\text{DataNormalized} = \frac{data - data.mean}{data.standard\_deviation}$$

$$\text{DataStandardized} = \frac{\text{data} - \text{data.min}}{data.max - data.min}$$

Before PCA, we also need to do Normalization operation to data. Before training models, we use normalizations in classical machine learning models, while standardization in deep learning model.

1.3 Label Classification

When obtaining labels, we divide all diseases into three categories, one for cancer and malignant tumor, one for uncertain cases and the other for other diseases. After that, we drop all the uncertain cases including unrecorded cases and those remained to be diagnosed. Finally, we note those certain cancers and malignant tumors as 1 and other diseases as 0 or -1.

1.4 F2 Score

F_2 Score is to calculate the goodness of the Binary Classification Model. Comparing to F_1 Score, F_2 Score values more on Recall than Precision, since, in our case, there are 2753 samples in Class 1, while 1238 samples in Class 0. So we pay more attention on the case where actual class is Class 1.

$$\text{precision} = \frac{true\ \ positive}{true\ \ positive + false\ \ positive}$$

$$\text{recall} = \frac{true\ \ positive}{true\ \ positive + false\ \ negative}$$

$$F_m\ \ Score = (1 + m^2) \times \frac{precision \times recall}{m^2 \times precision + recall}$$

3. Classical Machine Learning

After fetching data, we need to firstly shuffle all the samples so that all the samples could be in a random order.

In training and validation, we choose 10-fold validation system, which means to train models with nine-tenth of all data and then test the models with the rest of data. This procedure will repeat for ten runs.

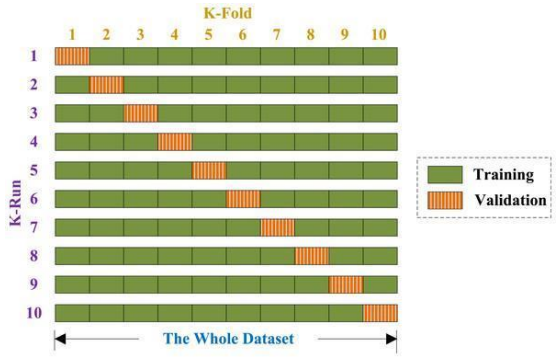

Figure 2. Ten-Fold Cross Validation[3]

1.1 Linear Discriminant Analysis (LDA)

In utilizing Discriminant Analysis function , 'ClassificationDiscriminant', from library function of Matlab, we set the type of discriminant function as 'pseudoLinear' so that we could get a Linear Discriminant Analysis model. After running it for 10 runs, we could evaluate the performance of the model by calculating 10 accuracies for ten-fold Cross Validation.

1.2 Random Forest (RF)

In utilizing Random Forest function, 'TreeBagger', from library function of Matlab, we set the number of ntree as 500 so that it could be complicated enough to match the features. After running it for 10 runs, we could evaluate the performance of the model by calculating 10 accuracies for ten-fold Cross Validation.

1.3 Decision Tree (DT)

In utilizing Decision Tree function, 'classregtree', from library function of Matlab, we just use its default form and parameters. After running it for 10 runs, we could evaluate the performance of the model by calculating 10 accuracies for ten-fold Cross Validation.

1.4 Logistic Regression (LogR)

In utilizing Logistic Regression function, 'glmfit', from library function of Matlab, we add parameters of 'binomial', 'link' and 'logit', to implement Logistic Regression. After running it for 10 runs, we could evaluate the performance of the model by calculating 10 accuracies for ten-fold Cross Validation.

1.5 SVM (Support Vector Machine)

In utilizing Logistic Regression function, 'svmtrain', from library function of Matlab, we just use the default parameters. Thus, there may be some boundary problems to occur. After running it for 10 runs, we could evaluate the performance of the model by calculating 10 accuracies for ten-fold Cross Validation.

1.6 Knn (k-Nearest Neighbor)

In utilizing k-Nearest Neighbor function, 'knnclassify', from library function of Matlab, we just use the default parameters. Thus, there may be some boundary problems to occur. After running it for 10 runs, we could evaluate the performance of the model by calculating 10 accuracies for ten-fold Cross Validation.

1.7 Training and Prediction

After gaining all ten results of accuracy and F2 Score for each model prediction, we could firstly calculate the mean value of accuracies and F2 Scores as well as the standard deviation of accuracies for each model. Then we obtain Table 1 with results of all six models. And based on all accuracies, we could plot a line chart containing the results in Figure 3.

	Accuracy Average[%]	F2 Score Average	Std Dev of Accur
LDA	95.76541	0.961422	0.007609
RF	94.06165	0.98194	0.002354
DT	91.23095	0.938047	0.014765
LogR	85.91767	0.869739	0.031854
SVM	82.58966	0.789738	0.146165
knn	79.07788	0.775097	0.03318

Table 1. Results of all six models

Based on Table 1, we make four key observations:

1. We could find that Discriminant Analysis and Random Forest have the best performance, which means he highest accuracies and the smallest standard deviations. The former one has a slightly

higher accuracy, though the latter one has a better F2 Score and smaller standard deviation.

2. SVM and knn have the worst performance, and the former one has a slightly higher accuracy, though it also has an extremely large standard deviation, which means that SVM's performance is quite unstable. The cause should be some boundary problems, such as those uncertain diseases, that have largely limited their performances.

3. Decision Tree and Logistic Regression have medium accuracies and F2 Scores. Therefore, their performances are just secondary.

4. We consider it reasonable that the performance of Decision Tree is lower than that of Random Forest, since the former one is just a simplified version of the latter one. Random Forest is a combination of multiple Decision Tree.

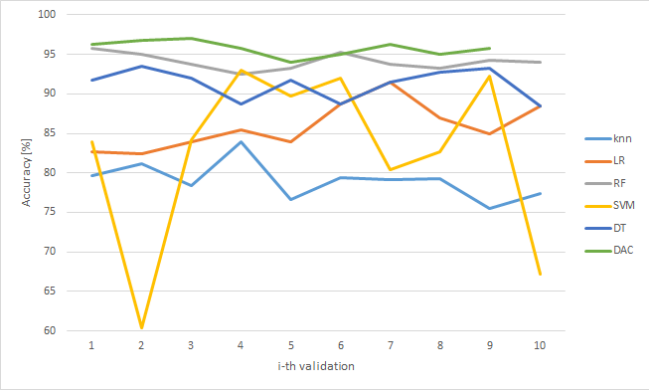

Figure 3. Line chart with all accuracies for 6 models

Based on Figure 3, we could confirm the four observations. The plot of SVM is an unstable line jumping up and down largely. And the performance of six models could be ranked as followed.

$$LDA>RF>DT>LogR>knn \approx SVM$$

4. Deep Learning – Back Propagation Neural Network

4.1 Description

In this method, a fully-connected neural network has been designed. It has only three layers, one input layer, one hidden layer with 500 neurons and one output layer. The diagram of its structures could be shown in Figure 4.

In this section, we have implemented Back Propagation algorithm and set the threshold for an end of training as the model could not increase its performance significantly, which means that the difference between performance of i-th round and that of i+i-th round is smaller than a parameter ε.

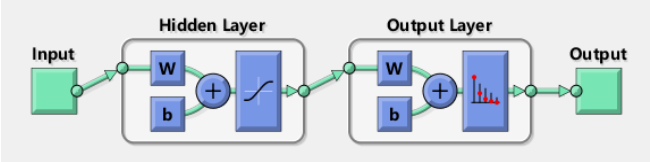

Figure 4. Diagram of BP Neural Network with three layers

4.2 4.2 Training and Prediction

After training the BPNN, we could test it with validation exactly like what we previously have done. We could get an average accuracy of 96.80476%, which is higher than all the classical machine learning models. In addition, the standard deviation of accuracies is 0.002247, slightly smaller than 0.002354 which is the smallest one in classical models. What's more, the F_2 Score is 0.982241, also the best. After that, we could a ROC Curve to further our discussion.

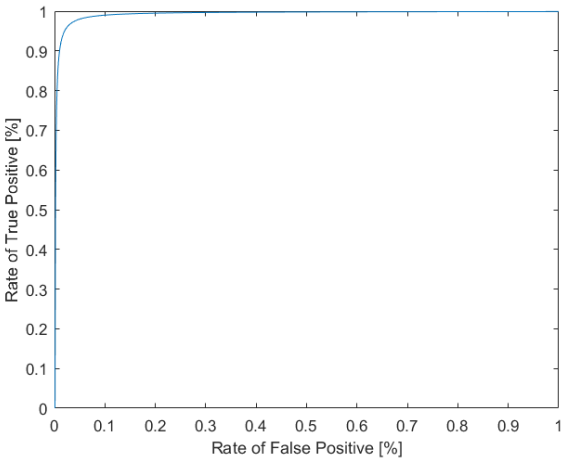

Figure 5. ROC Curve

We could find in ROC Curve, an extremely high rate of true positive is shown comparing to that of false positive. Combining this observation with F_2 Score, it's rather sound to draw a conclusion that the BPNN model has a better performance of prediction.

What's interesting is when we increase the complexity of the BPNN, namely to add another hidden layers, we could not find improved results. Conversely, we sometimes find it worse. We

speculate that there may be a Gradient Disappearance or Gradient Explosion in the case of more complex BPNN, which limits the modification of parameters, vector Θ. Besides, the data do not contain adequate samples and features and thus higher complexity is unnecessary or even trivial. What's more, there will be a higher possibility of overfitting in complex neural network

5. Further Discussion and Conclusion

5.1 Discussion about Classical Machin Learning Models

We have found the ranking of performances for six models in this case is LDA>RF>DT>LogR>knn≈SVM. However, this ranking is not absolute or definite, because we did not choose optimal parameters for each models. We superficially use default values of parameters despite some prompted ones, e.g. 'ntree' and 'logit'. If we could determine optimal parameters by cross validation as well as test set and then apply those parameters to the models, the ranking will definitely change. For example, we could have altered relaxation index C to resolve the Boundary Problem encountered by SVM model. Additionally, altering our goal of prediction and classification will also have a different ranking. The ranking is just for our case.

5.2 Discussion about Deep Learning Model

In this section, we just tried Back Propagation Neural Network. As an alternative method, we could also take Convolution Neural Network into consideration, since the result whether one has a cancer or not may just be determined by part of genes. Using CNN could be more suitable to the case and more time-efficient. However, to choose the part of genes in receptive field is extremely complicated and thus difficult to implement.

What's interesting is when we increase the complexity of the BPNN, namely to add another hidden layers, we could not find improved results. Conversely, we sometimes find it worse. We speculate that there may be a Gradient Disappearance or Gradient Explosion in the case of more complex BPNN, which limits the modification of parameters, vector Θ. Besides, the data do not contain adequate samples and features and thus higher complexity is unnecessary or even trivial. What's more, there will be a higher possibility of overfitting in complex neural network.

5.3 Comparison between Classical Machine Learning and Deep Learning

We could safely conclude that a well-organized Neural Network has a stably better performance than those Classical Machine Learning models in this case of application. Since NN could separate all the combination of high-dimensional features into simplified information in each neuron, NN could generally perform a better regression or classification. Additionally, Back Propagation Algorithm could provide a promising direction for parameters reaching optimal values without being stuck in a local optimum.

However, Classical Machine Learning models also have some advantages. They take much shorter time for training comparing to a Back Propagation Neural Network. Hence, an efficiency could be emphasized since Random Forest and Linear Discriminant Analysis have a slightly lower performance but far shorter time to run.

Reference

[1] Zhuo, Degen, et al. "Physical Mapping and Functional Annotation of 60, 000 Human Genes." Health Psychology : Official Journal of the Division of Health Psychology, American Psychological Association, American Psychological Association Inc., 23 Sept. 2016, mdanderson.influuent.utsystem.edu/en/publications/physical-mapping-and-functional-annotation-of-60-000-human-genes.

[2] "LSBN: A Large-Scale Bayesian Structure Learning Framework for Model Averaging : Yang Lu : Free Download, Borrow, and Streaming." Internet Archive, The Library Shelf, 18 Oct. 2012, archive.org/details/arxiv-1210.5135.

[3] “使用光敏電阻和機器學習來判斷會議室狀態-2.” CH.Tseng, 4 Dec. 2016, chtseng.wordpress.com/2016/09/07/使用光敏電阻和機器學習來判斷會議室狀態-2/.

[4] EMBL-EBI. “ArrayExpress.” AAA ATPase Domain (IPR003593) < InterPro < EMBL-EBI, www.ebi.ac.uk/arrayexpress/experiments/E-TABM-185/.