Artificial Intelligence Project Report

Xinyi Yu

Abstract—Artificial intelligence is a great breakthrough in human technology. With its quickly development, aitificial intelligence is playing a more and more significant part in many fields. Technologies based on artificial intelligence is also becoming more general. In this project, we are given the data set of gene chip and our goal is to classify them. First, classical approach is used to train the classifier. For data pre-processing, we make use of Principal Component Analysis(PCA) to reduce dimension. Then, Support Vector Machine(SVM), Logistic Regression and Decision Tree are used to train the classifier. Besides, we compare these methods in accuracy, training time and other aspects. Second, using deep learning, we construct a neural network to train the classifier. The accuracy and loss are also discussed in the paper.

Index Terms—Artificial Intelligence, Principal Component Analysis, Support Vector Machine, Logistic Regression, Decision Tree

1 Introduction

NOWADAYS, artificial intelligence is developing quickly and playing a more and more important part in many fields. As biology develops rapidly, effective methods of processing and analyzing the large set of human gene data in order to get more biological insight is in urgent need. Classical machine learning methods turn out to be a good solution, such as Principal Component Analysis(PCA), Support Vector Machine(SVM), Logistic Regression and Decision Tree. Moreover, as a burgeoning method, deep learning can also give an excellent result.

Classical machine learning methods, such as Principal Component Analysis(PCA), Support Vector Machine(SVM), Logistic Regression and Decision Tree, is relatively robust and leads to a great result in classification tasks whose goal is classifying the large data set into several categories. PCA is in widely use in data preprocessing, which is used to reduce the dimension of data set. We use Support Vector Machine, Logistic Regression and Decision Tree to train the classifier respectively and analyse their classification abilities with their performance, including accuracy and training time, on different sizes of data set. Also, we compare them with each other and find out their differences, in order to pick out the best solution.

As a powerful learning method using neural networks based on representation learning, deep learning combines low-level features to form more abstract high-level presentation attribute categories or features to discover distributed feature representations of data. Given the large data set of gene chips, we construct a neural network to train a classifier using deep learning method. The project result is fully discussed in the paper to give a rough but intuitive interpretation of deep learning.

In the project, we use both both classical methods and deep learning to train the classifiers and evaluate their performance. We fully discuss their advantages and disadvantages respectively in the paper according to their performance. Our code can be downloaded at https://github.com/Renjie-Woo/AI/

• *Email: yuxinyi0524@sjtu.edu.cn*

2 Method

The whole project is divided into three steps: Data preprocessing, Model Construction and Performance Evaluation. First, in the data preprocessing part, PCA(Principal Component Analysis) is used to reduce dimensions and preprocess the large data set of gene chips. Then, both classical methods and deep learning are used in Model Construction step. Finally, we apply cross validation as the main method to evaluate the model performance according to their accuracy and training time.

2.1 Data Preprocessing

Given the large data set of gene chip, we first normalize the features and then use Principal Component Analysis(PCA) to reduce dimensions.

2.1.1 Normalization

Data normalization is a basic work of machine learning. Different features tend to have different dimensions and dimensional units, which will affect the result of data analysis. In order to eliminate the dimension influence between indicators and solve the comparability among the data, data standardization is in urgent need. After data standardization, all features are in the same order of magnitude, which is suitable for comprehensive comparative evaluation. Also, normalization can accelerate the solving process improve the accuracy. Here are two common normalization methods: Z-score Normalization and Min-Max Normalization. We use the former one in this project.

Standard scores, or z score, of the samples are calculated as follow:

$$z = \frac{x_i - \mu}{\sigma} \tag{1}$$

μ and σ are the mean and standard deviation of corresponding features respectively.

After Z-score normalization, the processed data conform to the standard normal distribution, whose mean value is 0 and standard deviation is 1. The normalized features will be distributed in the interval of $[-1, 1]$.

2.1.2 Principal Component Analysis(PCA)

PCA(Principal Component Analysis) is the most widely used data dimensionality reduction algorithm. The main idea of PCA is to map the n-dimensional feature to the k-dimensional feature, which is a new orthogonal feature also known as the principal component and a k-dimensional feature reconstructed on the basis of the original n-dimensional feature. PCA's job is to find a set of mutually orthogonal coordinate axes sequentially from the original space, and the choice of new coordinate axes is closely related to the data itself. Among them, the selection of the first new coordinate axis is the direction in which the variance of the original data is the largest, the selection of the second new coordinate axis is the direction in which the variance is the largest to the first one, and the selection of the third axis is the direction in which the variance is the largest to the first and second axes. Repeat this process, we get n of these axes.Thus, most of the variance is contained in the previous k axes and the variance in the later axes is almost zero, so we can just consider the previous k axes and ignore the rest of the axes, which realizes dimension reduction of data features.

Fig.1 shows PCA of a multivariate Gaussian distribution centered at (1,3) with a standard deviation of 3 in roughly the (0.866, 0.5) direction and of 1 in the orthogonal direction. The vectors shown are the eigenvectors of the covariance matrix scaled by the square root of the corresponding eigenvalue, and shifted so their tails are at the mean.

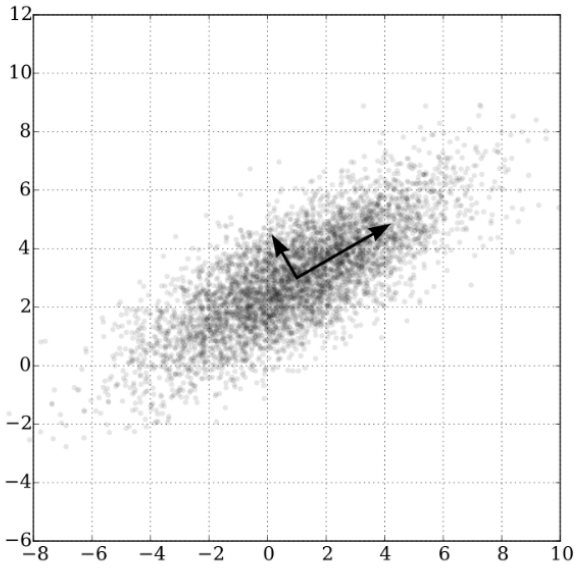

Fig. 1: PCA of a multivariate Gaussian distribution

2.2 Model Construction

Both classical methods and deep learning are used to train a classifier in model construction part.

2.2.1 Classical Methods

We use three classical methods: Logistic Regression, Support Vector Machine(SVM) and Decision Tree to do the classification task.

Logistic Regression

Logistic regression is a supervised statistical learning method mainly used to classify samples. The hypothesis function is:

$$h_\theta(x) = g(\theta^T x) \tag{2}$$

$$g(z) = \frac{1}{1+e^{-z}} \tag{3}$$

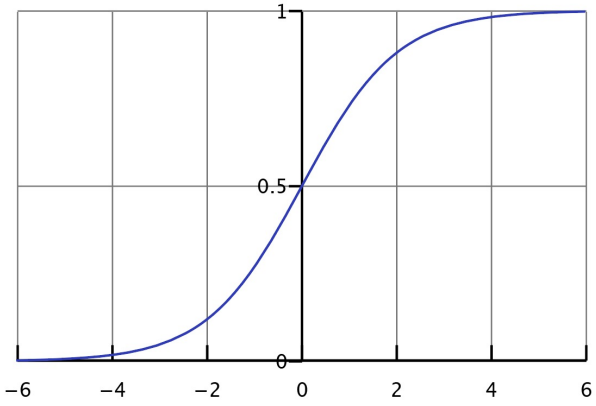

Fig. 2: Sigmoid function

Support Vector Machine(SVM)

Support vector machines(SVM) is a classification model. Its purpose is to find a hyperplane to segment samples. The principle of segmentation is to maximize the margin, and finally it is converted into a convex quadratic programming problem to solve. Fig.3 simply indicates the principle of SVM.

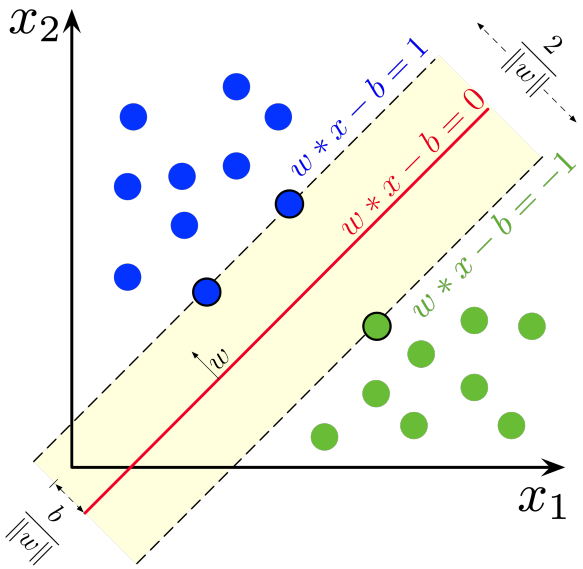

Fig. 3: Margin and Support Vector

Kernel function is applied in SVM, aiming to map a nonlinear problem into a higher dimensional feature space to make the problem linearly separable. Linear kernel is used in this project, which is calculated as follow:

$$K(x_i, x) = \phi(x_i)^T \phi(x) \tag{4}$$

And thus the target function can be written in the form below:

$$y = w^T\phi(x) + b = \Sigma_{i=1}^{N}\alpha_i y_i \phi(x_i)^T\phi(x) + b \quad (5)$$

SVM is usually applied in binary classification since its primal form classifies data into two classes: positive and negative class. However, SVM can also be used in multi-class classification. By combining several binary SVM classifiers together, a multi-class classifier is constructed.

Decision Tree

A decision tree is a flowchart-like structure in which each internal node represents a test on an attribute, each branch represents the outcome of the test, and each leaf node represents a class label. The paths from root to leaf represent classification rules.

2.2.2 Deep Learning

Deep learning is also used in the model construction part. We construct a neural network to train the classifier.

First, we use random function to initialize weights, in order to avoid the bad local solution caused by simple all zero initialization.

As for parameter sparsification and model simplification, regularization serves as an excellent approach. Models with high complexity always cause over-fitting problem, which can also be prevented by regularization. L2 regularization is used rather than L1 since L2 is smoother while L1 is more aggressive, which will result in some useful features being eliminated.

Batch normalization also plays an important part in neural network, which speeds up the convergence process and selects the proper learning rate. Moreover, generalization ability and training accuracy can also be improved through batch normalization.

The loss function takes the difference between predicted labels and associated ground truth into consideration. Minimizing the loss function will lead to a more accuracy and robust model.

Then, we apply the gradient descent method to update parameters. The optimization idea of gradient descent method is to use the negative gradient direction of the current position as the search direction, which is the fastest descending direction of the current position.

2.3 Performance Evaluation

Cross Validation is used as the evaluation method in the performance evaluation part. The basic idea is to divide the data set into two parts, one of which serves as the train set and the other serves as the validation set. First, train the classifier with the training set. Then test the model obtained from the training result with the validation set, which is used as performance evaluation of the classifier.

3 RESULT

3.1 Data Preprocessing

In order to improve training accuracy and speed, we exclude samples without labels and the classes with too few samples, which is not suitable for training. Finally, 3613 samples with 92 labels are picked out for training. In binary classification, labels related to cancer or tumor are denoted as class "1" while non-cancer labels are denoted as class "0". In multi-class classification, labels are denoted by 93 numerical numbers.

After normalization, PCA is used to reduce dimensions. Also, we set different percentage of preserved variance and apply normalization and PCA respectively to figure out the best size of dataset for later training. Table 1 shows the left number of dimensions with different variance percentage. It can be seen in Table 1 that the dimension number varies as the variance percentage changes.

As we all know, more dimensions always lead to the less loss of information. However, the model complexity and computational cost will increase as more dimensions are selected, which will reduce the accuracy and slower down the training process. Thus, we tend to choose a data set of medium size in order to obtain the suitable model. In this project, we choose the data sets with both 441 and 1063 dimensions in order to get more general results.

TABLE 1: PCA variance preserved

Variance Percentage	Dimension Number
80%	89
85%	186
90%	441
95%	1063

3.2 Classical Methods

We use several classical methods, including Logistic Regression, SVM and Decision Tree, to do both binary and multi-class classification.

3.2.1 Kernel Function

Kernel function is a core issue when using Support Vector Machine(SVM), which has a lot to do with the performance of SVM. In order to choose a better kernel function in SVM, hoping to obtain a more accuracy model, we use three kinds of kernel functions and compare them with each other according to their accuracy in both binary and multi-class classification.

We test linear, polinomial and gaussian kernels and Table 2 shows the accuracy of these different kernels. We can find that linear kernel beats the other two kernels, taking an overall consideration of both binary and multi-class classification. It is also indicated that our data set is linearly separable and thus we choose linear kernel as the kernel function used in SVM for later training.

TABLE 2: Accuracy of different kernels

Kernel	Accuracy(binary)	Accuracy(multi-class)
Linear	99.03%	85.64%
Polinomial	99.34%	82.01%
Gaussian	98.73%	78.83%

3.2.2 Binary Classification

In binary classification, Logistic Regression, SVM and Decision Tree are applied. We use MatLab to implement them and annalyse their results. Their performance on different data set can be seen in Table 3 and Table 4.

To be more visualized, Fig.4 and Fig.5 respectively show the distribution of the first two and three principal components of the dataset, which are the classification results of different data sets.

TABLE 3: Binary classification for 441-dimension dataset

Method	Accuracy	Training Time(sec)
Logistic Refression	99.22%	63.9870
SVM	99.00%	10.3919
Decision Tree	95.65%	10.4227

TABLE 4: Binary classification for 1063-dimension dataset

Method	Accuracy	Training Time(sec)
Logistic Refression	98.92%	301.3798
SVM	99.09%	36.1320
Decision Tree	94.85%	27.4953

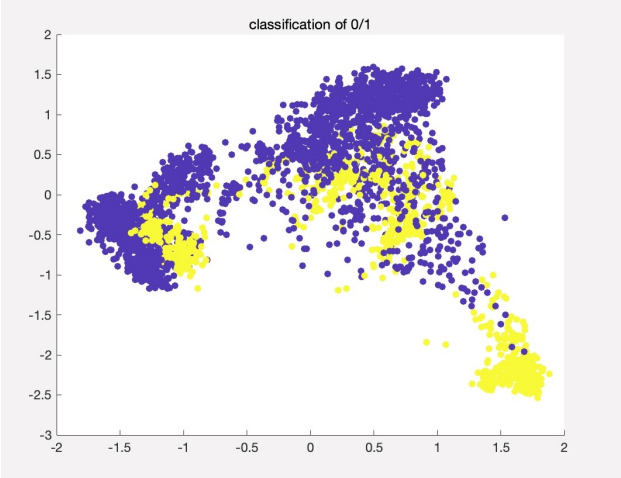

Fig. 4: 2-D Data Distribution of Binary Classification

Fig. 5: 3-D Data Distribution of Binary Classification

3.2.3 *Multi-class Classification*

SVM and Decision Tree are used to train the classifier in multi-class classification. Their performance on different data sets and comparison are shown in Table 5 and Table 6.

Also, we give the classification results in Fig.5 and Fig.6, in the form of distribution of the first two and three principal components of the dataset.

TABLE 5: Multi-class classification for 441-dimension dataset

Method	Accuracy	Training Time(sec)
SVM	85.83%	498.7569
Decision Tree	61.25%	46.9884

TABLE 6: Multi-class classification for 1063-dimension dataset

Method	Accuracy	Training Time(sec)
SVM	72.29%	953.0567
Decision Tree	60.86%	109.6956

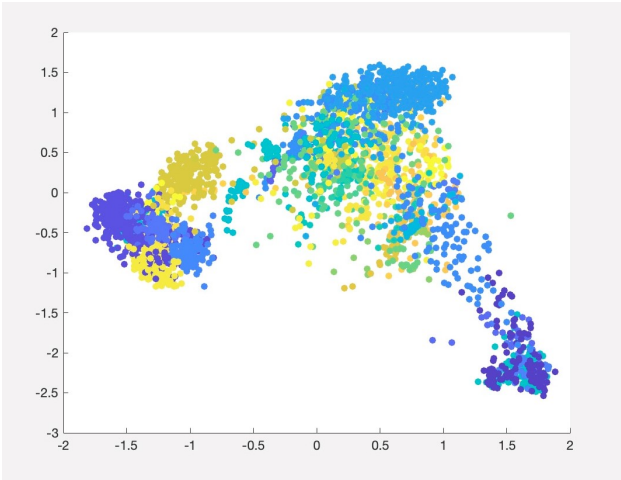

Fig. 6: 2-D Data Distribution of Multi-class Classification

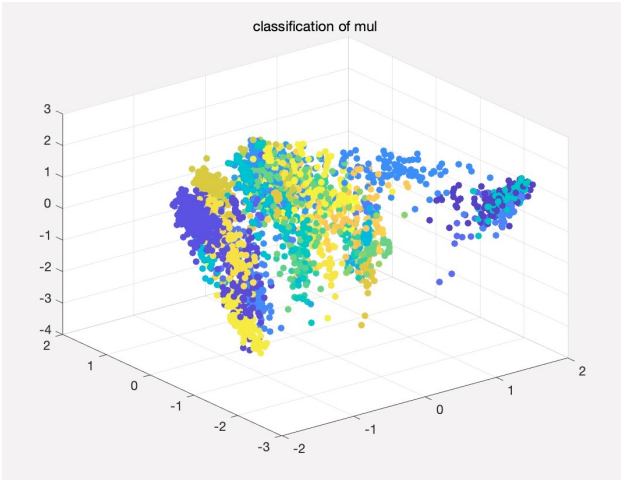

Fig. 7: 3-D Data Distribution of Multi-class Classification

3.3 Deep learning

In deep learning part, we construct a neural network using the framework of Keras. Keras is a highly integrated machine learning framework. In total, 5 hidden layers are constructed for training, with an additional layer for L2

regularization. There are 93 neural units in output layer, representing the classification labels. In order to get balanced data distribution, we apply under-sampling to most of the data and over-sampling to the rest of the data. Batch normalization is also used in every layer to improve the performance.

Cross validation is used as the evaluation method. We set 80% of the input data as training set and the rest as testing set. Through this method, the model can be tested well and avoid the high variance which leads to the over-fitting problem.

We use TensorBoard to visualize the results of deep learning and the results are shown in Fig.8 and Fig.9. The training accuracy is relatively high, which indicates that the network is compatible with the data and the training process works well. Testing accuracy is also high, but lower than training accuracy expectedly, which indicates that the network is robust with low bias and low variance.

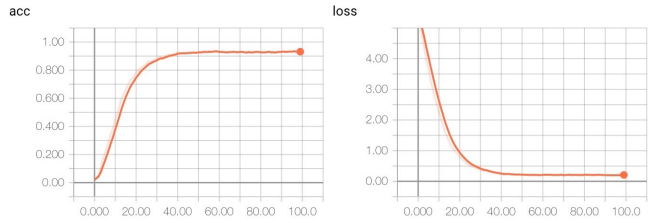

Fig. 8: Training accuracy and loss

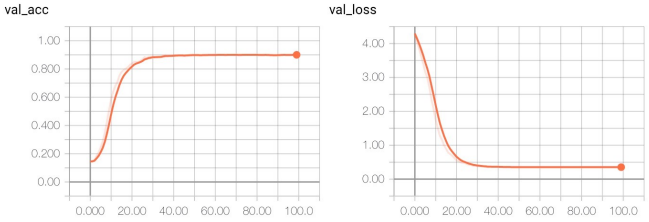

Fig. 9: Testing accuracy and loss

4 Discussion

We fulfill two tasks in the project: binary classification and multi-class classification. For potential improvement in the future, there is a need to annalyse them and attempt to figure out the optimal solution for each task. Using several methods in the project, we evaluate their performance and compare them with each other to annalyse them, which makes contribution to our deep understanding of these defferent methods.

4.1 Binary Classifier and Multi-class Classification

We can see from Table 2-Table 5 that binary classfier achieves higher accuracy with less training time, which indicates that binary classifier is more robust and performs better than multi-class classfier. Since binary classfication just has two kinds of labels while multi-class classification has 92, the model of binary classfication is obviously easier and thus the better performance is reasonable.

4.2 Logistic Regression and Support Vector Machine

From their performance in binary classification, SVM leads to higher accuracy with less training time. Since SVM uses kenel function to map the non-linear problem into a higher dimensional feature space, only the inner product should be calculated and there is no need to take the detailed computational process of complex high-dimensional mapping into consideration. Thanks to the kernel trick, SVM can get better results using less time.

Moreover, SVM exports the concept of Support Vector(SV), which is greatly beneficial to the performance. Since SVM just consider the data points near the segment hyperplane, only the support vectors count. But Logistic Regression takes all the data points into consideration. Directly depending on the data distribution, each sample data point will affect the result. If there is a serious imbalance between different categories of training data, it is generally necessary to balance the data to make the samples of different categories as balanced as possible.

4.3 Classical Methods and Deep Learning

As shown in the results, classical methods such as SVM achieve a better performance than deep learning when the data set is not large. Deep learning aims at learning the topological structure of the given data and the relation between the features instead of the feature itself, which requires a lot of data for training because of the difficulty in overall structure learning. But classical methods focus on the features rather than the structure, thus results in a more robust and accurate model trained with a small size of data.

Since classical methods have been developed for several years, their performance has also passed many tests, making them robust and practical for most tasks. As a newly developed method, deep learning is playing a more and more significant role in many fields and also has a large application potential.

Acknowledgments

Thank our teacher Bo Yuan and the teaching assistant for their patient guidance. Thank my partner Renjie Wu for his help and coorperation.

References

[1] Baldi,PierreandKurtHornik *Neuralnetworks and principal component analysis: Learning from examples without local minima*
[2] Fung, Glenn M and Olvi L Mangasarian *Multicategory proximal support vector machine classifiers*

Xinyi Yu has been studying in Shanghai Jiao Tong University since 2016. She is an undergraduate of School of Electronic Information and Electrical Engineering, majoring in Computer Science.