A Comparative Analysis of Three Supervised Learning Algorithms in Stock Selection

Zhen Sun\textsuperscript{1,a}, Shangmei Zhao\textsuperscript{1,b,*}

\textsuperscript{1}School of Economics and Management, Beihang University, Beijing, China
\textsuperscript{a}tomallen@foxmail.com
\textsuperscript{b}randomvague@163.com
\textsuperscript{*}Corresponding author

Abstract: In this paper, our goal is to judge which algorithm is the best, through comparing the classification accuracy of the three supervised machine learning algorithms, using the data of four financial factors which can reflect the intrinsic value of corporate stock. Our empirical results show that Support Vector Machine got the extremely high classification accuracy in the test both inside and outside the samples; Random Forest achieved the highest classification accuracy in the test within the samples, but it’s accuracy was not as good as Support Vector Machine in the test outside the samples, it means Random Forest was prone to over-fitting; The classification accuracy of Naive Bayes was very low in the test both inside and outside the samples. According to the Bayesian hypothesis, we can see that financial factors are not independent of each other. As a result, the optimal algorithm is Support Vector Machine, followed by the Random Forest, and it is not advisable to use the Naive Bayes, when we selecting the stocks using financial factors data.

1. Introduction

The quality of the financial data of a listed company determines the intrinsic value of its stocks and influences its stock price in the market. Therefore, it is the focus of attention among investors to accurately select companies with excellent financial statistics. Selection of stocks is in essence a classification problem, and one prevailing solution to this problem is using the method of machine learning.

Machine learning\textsuperscript{[1;2]} is a collective term for a type of algorithm that aims to dig out the hidden law from large amounts of historical data to make forecast and classification. From the perspective of mathematics, machine learning can be described as a mapping relation which, to be specific, converts the original input $x$ into the ideal output $\hat{y}$. Machine learning includes supervised learning, semi-supervised learning, unsupervised learning and reinforcement learning. Supervised learning\textsuperscript{[3;4]} minimizes the difference between the ideal output $\hat{y}$ and the actual $y$, and estimate the model parameters. In supervised learning, the trained data need to be labelled, but in unsupervised learning\textsuperscript{[5;6]}, the data do not need to be labelled, so unsupervised learning can categorize the data without any extra information. When the training data label is hard to obtain, semi-supervised learning can use a few labelled training data to improve the learning accuracy of supervised learning or unsupervised learning. Reinforcement learning is an algorithm that studies how the intelligent agent matches its behavior with the environment to obtain the maximum cumulative reward\textsuperscript{[7]}.

A company’s financial status needs financial professionals to judge, guide and supervise the machine
to learn. This kind of machine learning belongs to supervised learning. In this study, we classify 3498 stocks in China’s A stock market using three widely-used supervised learning algorithms (Random Forest, SVM and Naive Bayes) and the financial factors data, and identify which algorithm performs the best.

Random Forest\cite{8,9} is a very effective non-parameter classification algorithm. It extracts a group of training sample set from the original data and generates a group of individual classifiers which produce the classifying result through voting. It assembles the voting result of a group of individual classifiers, so the result this algorithm achieves is high in comparison with others’. The SVM algorithm\cite{11} can solve some problems effectively in the classification procedures, including the limited size of samples, nonlinearity and the high dimension of data. This algorithm replaces the dot product between vectors with the kernel function to maximize the gap of support vector\cite{12,13} so that the problem of input space can be converted into a linear separable problem of the characteristic space and the classification result can reach high accuracy. The Naive Bayes\cite{14} is an algorithm based on the Bayes theorem and Bayes hypothesis, and the parameter estimation is usually maximum likelihood estimation. According to the Bayes hypothesis, if the data features is of poor independence, the classification performance of Naive Bayes is awful\cite{15}.

2. Three Supervised Learning Algorithms

2.1. Random Forest

The Random Forest uses the Bootstrapping algorithm to build \( N \) training sets from the original sample data, trains the data by the individual classification algorithm to obtain \( N \) individual classifiers which is used to produce the final classification result. The Random Forest algorithm uses random uniform sampling, the weights of individual classifier are equals and each individual classifier can be generated in parallel.

The individual classification algorithm is Classification And Regression Tree (CART) in this paper. CART\cite{10} uses the Gini coefficient to identify the optimal split features. For set \( Z \), we assumes it can be divided into \( N \) classes. \( N_i \) represents the set of \( i \)-th class in \( Z \), \( |N_i| \) is the size of \( N_i \), \( |Z| \) is the size of \( Z \). The Gini coefficient of set \( Z \) is:

\[
Gini(Z) = \sum_{i=1}^{N} \frac{|N_i|}{|Z|} \left( 1 - \frac{|N_i|}{|Z|} \right) = 1 - \sum_{i=1}^{N} \left( \frac{|N_i|}{|Z|} \right)^2
\]

When a node of CART split, the Gini coefficients of all features in this node set \( D \) are calculated according to the equation \( Gini(D, f_i) = \frac{|D_1|}{|D|} Gini(D_1) + \frac{|D_2|}{|D|} Gini(D_2) \). The coefficients are ordered and the feature with the minimum Gini coefficient is taken as the split feature. In the process, the CART only realizes binary split of the features. \( D_1 \) and \( D_2 \) are two subsets of \( D \) divided by the feature \( f_i \).

The loss function of the CART is \( C_g(T) = \sum_{t=1}^{T} N_t Gini(t) + \alpha |T| \).

In the function, \( C_g(T) \) is the total loss of the node \( T \), \( N_t \) is the number of samples at the \( t \)-th sub-node of \( T \). \( Gini(t) \) is the Gini coefficient of the \( t \)-th sub-node of \( T \), \( \alpha \geq 0 \) is the penalty factor and \( |T| \) is the number of sub-nodes of \( T \).

When \( C_g(T) \leq Gini(T) \), node \( T \) turns into leaves and the model can be simplified.

2.2. SVM

It is known that the training set is \( D = \{(x_1, y_1), (x_2, y_2), \cdots, (x_m, y_m)\} \), where \( y_i \in \{0, +1\} \). The hyperplane is \( w^T x + b = 0 \), where \( w \) and \( b \) represent the weight and bias of the hyperplane. The gap of the SVM (i.e. the total distance of two support vectors from the hyperplane), which can be represented as \( \text{margin} = \frac{2}{\|w\|} \). To realize more accurate classification, the maximum gap of the SVM needs to be found, and the target function is
Because to maximize $\frac{1}{||w||}$ is equivalent to minimize $||w||^2$, the target function can be converted to:

$$
\begin{align*}
\max_{w,b} & \quad \frac{2}{||w||} \\
\text{s.t.} & \quad y_i(w^T x_i + b) \geq 1, i = 1,2,\cdots, m
\end{align*}
$$

By adding the Lagrange multiplier $\alpha_i \geq 0$ to each constraint, the Lagrange function can be built as:

$$
L(w, b, \alpha) = \frac{1}{2}||w||^2 + \sum_{i=1}^{m} \alpha_i (1 - y_i (w^T x_i + b))
$$

where $\alpha = (\alpha_1, \alpha_2, \cdots, \alpha_m)$. According to Lagrange duality, the target function can be converted to:

$$
\begin{align*}
\min_{\alpha} & \quad \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j (x_i^T \cdot x_j) - \sum_{i=1}^{m} \alpha_i \\
\text{s.t.} & \quad \sum_{i=1}^{m} \alpha_i y_i = 0, \\
& \quad \alpha_i \geq 0, \quad i = 1,2,\cdots, m
\end{align*}
$$

The previous target function is established under the condition that the training samples are linear and separable. In actual conditions, however, there may be no hyperplane which can correctly divide two classes of samples in the space of the original samples. Therefore, a high-dimensional linear separable feature space is needed to replace the original space of samples. To this end, $\emptyset(x)$ is introduced into the target function to replace $x$. $\emptyset(x)$ represents the feature vector obtained by mapping $x$ in the original space of samples onto the high-dimensional feature space. The target function is then converted into:

$$
\begin{align*}
\min_{\alpha} & \quad \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j \emptyset(x_i)^T \emptyset(x_j) - \sum_{i=1}^{m} \alpha_i \\
\text{s.t.} & \quad \sum_{i=1}^{m} \alpha_i y_i = 0, \\
& \quad \alpha_i \geq 0, \quad i = 1,2,\cdots, m
\end{align*}
$$

As the dimension of the feature space is high or even infinite, it is very difficult to calculate $\emptyset(x_i)^T \emptyset(x_j)$. Thus, we used the kernel function to replace $\emptyset(x_i)^T \emptyset(x_j)$, and the function $k(x_i, x_j) = \langle \emptyset(x_i), \emptyset(x_j) \rangle = \emptyset(x_i)^T \emptyset(x_j)$ is achieved. The target function is:

$$
\begin{align*}
\min_{\alpha} & \quad \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j k(x_i, x_j) - \sum_{i=1}^{m} \alpha_i \\
\text{s.t.} & \quad \sum_{i=1}^{m} \alpha_i y_i = 0, \\
& \quad \alpha_i \geq 0, \quad i = 1,2,\cdots, m
\end{align*}
$$

This target function is the expression formula of SVM. The kernel function used in this study is

$$
k(x_i, x_j) = \exp \left( -\frac{||x_i^T x_j||^2}{2\sigma^2} \right), \quad \text{where} \quad \sigma > 0 \quad \text{is the width of the Gaussian function}.
$$

2.3. Naive Bayes

It is assumed that the training set $D = \{(x_1, y_1), (x_2, y_2), \cdots, (x_i, y_i), \cdots, (x_N, y_N)\}$, where $N$ is the
number of training samples, $x_i$ is the $i$-th input sample. $x_i = \{x_{i1}, x_{i2}, \cdots, x_{im}, \cdots, x_{in}\}$, where $x_{im}$ is the $m$-th feature value of $x_i$ and $M$ is the number of features. $y_i$ is the label value of $x_i$, $y_i \in \{c_1, c_2, \cdots, c_k, \cdots, c_K\}$, and $K$ is the number of classes.

It is known that the Naive Bayes has an assumption, that is, to assume that the features of data are independent of each other for the known classes. According to the Bayes theorem, the following equation is achieved.

$$P(y_i = c_k|x_i) = \frac{p(x_i|y_i = c_k)p(y_i = c_k)}{p(x_i)} = \frac{p(y_i = c_k)}{p(x_i)} \prod_{m=1}^{M} p(x_{im}|y_i = c_k)$$

As it is the same for every class $P(x_i)$, the Naive Bayes equation can be achieved:

$$\hat{y}_i(x_i) = \arg\max_{c_k} p(y_i = c_k) \prod_{m=1}^{M} p(x_{im}|y_i = c_k)$$

In Bayes learning, it is necessary to estimate the prior probability $P(y_j)$ and the conditional probability $P(x_i|y_j)$. The most common estimation method is maximum likelihood estimation:

$$\hat{p}(y_i = c_k) = \frac{\sum I(y_i = c_k) + \lambda}{N + k\lambda}$$

where $I$ is the indicator function. The function value $I$ equals 1, when the condition $(y_i = c_k)$ is satisfied; otherwise the function equals 0. $\lambda$ is the parameter and $k$ is the number of values that can be assigned to $y_i$.

$$\hat{p}(x_{im}|y_i = c_k) = \frac{\sum I(x_{im}, y_i = c_k) + \lambda}{\sum I(y_i = c_k) + u\lambda}$$

Where, $u$ is the number of values that can be assigned to the $m$-th feature of $x_i$. In actual calculation, it is often assumed $\lambda = 1$, and this situation is called Laplacian smoothing.

3. Financial Factors

In this paper, the financial factors includes the price-to-earnings ratio (P/E ratio), price-to-book ratio (P/B ratio), return on equity (ROE) and price/earnings to growth ratio (PEG ratio). The calculation equations are as follows.

1. $PE = stock\ \text{price\ per\ share}/net\ \text{earnings\ per\ share}$. The smaller PE is, the higher the investment value the stock has.

2. $PB = stock\ \text{price\ per\ share}/net\ \text{asset\ per\ share}$. The smaller PB is, the higher the investment value the stock has.

3. $ROE = net\ \text{earnings\ per\ share}/net\ \text{assets\ per\ share}$. It indicates how much profit a stock holder will get by investing one unit of currency. The higher the ROE is, the more investment value the stock has.

4. $PEG = PE/net\ \text{profit\ growth\ ratio}$. $PEG = 1$ is the threshold point. When $PEG < 1$, it means the earnings of company increase fast and faster than the growth rate of the stock price. In this case, though the PE ratio is high, there is still room for price growth. The smaller the PEG is, the larger the stock’s investment value is.

4. Data description

The dataset is built with the financial factors data of 3498 stocks on the A stock market. The selected time is from the first season of 2014 to the third season of 2018.

4.1. Data pretreatment

1. Converting the original factors by the $Value$ equation.

$$Value = (V_{now} - V_{low})/(V_{high} - V_{low})$$

where $V_{now}$ represents the present value of the factor, $V_{low}$ represents the minimum value of the factor, and $V_{high}$ represents the maximum value of the factor. $Value$ is a measure to identify the position of the current value of the factor in historical data. When
the four financial factors are put into the equation Value, the EV Value, PB Value, ROE Value and PEG Value can be achieved.

(2) Setting the rules for design of labels.

Label rules: to calculate the composite financial factor of each stock (Multi)

\[ Multi = mean(PE\text{Value}, PB\text{Value}, (1 - ROE\text{Value}), PEG\text{Value}) \]

As Multi is the mean of four financial factors and it is between 0 and 1. Among all data samples, the maximum of Multi is 1, and the minimum is 0.0052, and the mean is 0.395. In selection of stocks, the stock with a smaller value of Multi is preferred, but it would be meaningless if the value of Multi is too small. So, the range of Multi is set between the mean and 0.1, so the range is between 0.1 and 0.3. If the value of Multi is below 0.3 or above 0.1, this stock is considered suitable for investment and is labelled 1; otherwise, the label is 0.

4.2. Training data set: \( D_{\text{train}} = (X_{\text{train}}, Y_{\text{train}}) \)

\( X_{\text{train}} = [x_{i1}, x_{i2}, x_{i3}, x_{i4}] \), where \( i \) represents the \( i \)-th stock, \( x_{i1} \) represents the value \( PE\text{Value} \) of the \( i \)-th stock, \( x_{i2} \) represents the value \( PB\text{Value} \) of the \( i \)-th stock, \( x_{i3} \) represents the value \( (1 - ROE\text{Value}) \) of the \( i \)-th stock, and \( x_{i4} \) is the value \( PEG\text{Value} \) of the \( i \)-th stock.

\( Y_{\text{train}} = [y_{1}, y_{2}, \ldots, y_{j}, \ldots] \), where \( y_{j} \) is obtained based on the labelling rule.

To make comprehensive analysis of the classification effect of the three algorithms, the training dataset can be divided into two classes. One is the training dataset \( D_{\text{train1}} \) that contains the test data and can be used to analyze the classification effect of the classification algorithms within the set of samples; the other is the training set \( D_{\text{train2}} \) which does not contain the test data and can be used to compare the classification effect of the algorithms outside the set of samples.

(1) The training set \( D_{\text{train1}} \) that contains the test data includes the financial factors data of 3498 stocks on the A stock market, and the time is from the first season of 2014 to the third season of 2018.

(2) The training set \( D_{\text{train2}} \) that does not contain the test data includes the financial factors data of 3264 stocks except those on the MSCI China Index, and the time is from the first season of 2014 to the third season of 2018.

Before the missing training data are filtered, \( 1 \leq i \leq 3498 \) in the training dataset \( D_{\text{train1}} \), and \( 1 \leq i \leq 3264 \) in the dataset \( D_{\text{train2}} \).

After the missing training data are filtered, \( 1 \leq i \leq 2695 \) in the training dataset \( D_{\text{train1}} \), and \( 1 \leq i \leq 2481 \) in the training dataset \( D_{\text{train2}} \).

4.3. Test dataset: \( D_{\text{test}} = (X_{\text{test}}, Y_{\text{test}}) \)

\( D_{\text{test}} \) includes the financial factors data of 234 stocks on the MSCI China Index, and the time is from the first season of 2014 to the third season of 2018.

\( X_{\text{test}} = [x_{j1}, x_{j2}, x_{j3}, x_{j4}] \), where \( j \) represents the \( j \)-th stock, \( x_{j1} \) represents the value \( PE\text{Value} \) of the \( j \)-th stock, \( x_{j2} \) represents the value \( PB\text{Value} \) of the \( j \)-th stock, \( x_{j3} \) represents the value \( (1 - ROE\text{Value}) \) of the \( j \)-th stock, and \( x_{j4} \) represents the value \( PEG\text{Value} \) of the \( j \)-th stock.

\( Y_{\text{test}} = [y_{1}, y_{2}, \ldots, y_{j}, \ldots] \), where \( y_{j} \) is obtained based on the labelling rule.

Before the missing test data are filtered, \( 1 \leq j \leq 234 \); after the missing tests data are filtered, \( 1 \leq j \leq 188 \).

5. Comparison of classification algorithms

5.1. Training model

Through training three classifiers (Random Forest, SVM and Naive Bayes) respectively by the training set \( D_{\text{train}} \), we can obtain the stable parameters of three models. The detailed steps are as follows.

Firstly, we process the original training data \( X_{\text{train}} \) by the classifier, and obtain the prediction value \( \hat{Y}_{\text{train}} \).

Secondly, we compare the prediction value \( \hat{Y}_{\text{train}} \) with the label value \( Y_{\text{train}} \) and update the
parameters through repetitive iteration.

5.2. Assessment model
With the test set \( D_{test} \) and the assessment indicators \( N_{diff} \) and \( accRate \), the classification effect of three classification algorithms is analyzed to judge which model is best to classify those stocks. The assessment rule is as follows.

The classification effect is measured by two indicators. The first is the number of error classification samples \( N_{diff} \), and the less the error is, the better the classification effect is. The second indicator is the classification accuracy rate \( accRate \), and the higher the accuracy rate is, the better the classification effect is.

(1) The number of errors \( N_{diff} \) equals the number of differences between \( \hat{y}_j \) and \( y_j \), where \( \hat{y}_j \) represents the predication value of the \( j \)-th stock in the test set, and \( y_j \) is the label value of the \( j \)-th stock in the test set;

(2) Classification accuracy rate: \( accRate = \frac{N_{diff}}{N} \), in which \( N \) is the number of samples in the test set.

6. Classification result of classification algorithms

6.1. Classification result of the test within the samples
The test within the samples means the test data in the training set, that is, the training samples that include the test data. In this case, there are 2695 training samples and the number of test data is 188.

Table 1. Classification result of the test within the samples

| Classification result (The test within the samples) | The number of errors | Accuracy rate |
|-----------------------------------------------------|----------------------|---------------|
| Random Forest                                       | 0                    | 100%          |
| SVM                                                 | 5                    | 97%           |
| Naive Bayes                                         | 101                  | 46%           |

The result shows that the accuracy rate of the Random Forest is 100% and there is 0 error; the accuracy rate of SVM is 97% and there are five errors; the accuracy rate of the Naive Bayes is 46% and there are 101 errors.

6.2. Classification result of the test outside the samples
The test outside the samples means the test data is not included in the training sample. In this case, there are 2481 training samples and 188 test data.

Table 2. Classification result of the test outside the samples

| Classification result (The test outside the samples) | The number of errors | Accuracy rate |
|-----------------------------------------------------|----------------------|---------------|
| Random Forest                                       | 14                   | 93%           |
| SVM                                                 | 10                   | 95%           |
| Naive Bayes                                         | 101                  | 46%           |

The result shows that the accuracy rate of the Random Forest is 93% and there are 14 errors; the accuracy rate of SVM is 95% and there are 10 errors, and the accuracy rate of the Naive Bayes is 46% and there are 101 errors.

6.3. Classification result analysis
As is known that a higher classification accuracy rate of the algorithm within the samples means it has a better classification performance for the known data, and a higher classification accuracy rate of the algorithm outside the samples indicates it has a better classification performance for the unknown data and it has a better prediction effect.

According to the classification results, the classification accuracy rate of the SVM is quite high in
the test both inside and outside the samples, and the prediction effect is excellent. The Random Forest has the best classification effect in the test within the samples, but its accuracy rate is not as good as SVM in the test outside the samples, so the Random Forest is likely to be subject to overfitting. The Naive Bayes has awful classification performance in the test both inside and outside the samples, it means financial factors are not independent of each other.

7. Conclusion

Three supervised learning algorithms (Random Forest, SVM and Naive Bayes) are used in this study to select stocks that have excellent financial data. The classification results of the three algorithms show that the SVM has a better classification effect in the test both inside and outside the samples, and the accuracy rates are 97% and 95%, so the comprehensive performance of this algorithm is the best. Since the Random Forest is likely to be subject to overfitting, the classification accuracy rate of Random Forest reaches 100% in the test inside the samples, but in the test outside the samples this rate is 93% and is not as good as SVM, so the comprehensive performance of the Random Forest ranks the second place. As financial factors are not independent of each other, the classification effect of the Naive Bayes is not ideal. The classification accuracy rates that the Naive Bayes achieves in the test both inside and outside the samples are both 46%, and the comprehensive performance of this algorithm is the worst among the three. Therefore, in selection of stocks based on financial factors, the best algorithm is SVM, followed by the Random Forest, and it is not advisable to use the Naive Bayes.

Acknowledgements

This research was financially supported by National Natural Science Foundation of China (Grant No. 71373017) and National Social Science Foundation of China (Grant No. 18BJY159)

References

[1] Kadhim A I. Survey on supervised machine learning techniques for automatic text classification [J]. Artificial Intelligence Review, 2019(3).
[2] Zheng W, O'Boyle M. Machine Learning in Compiler Optimization [J]. Proceedings of the IEEE, 2018, PP (99):1-23.
[3] Zhihua Z. A brief introduction to weakly supervised learning [J]. National Science Review, 2018.
[4] Zheng X, Wang Y, Wang G, et al. Fast and robust segmentation of white blood cell images by self-supervised learning [J]. Micron, 2018, 107:55-71.
[5] Tse K C, Chiu H C, Tsang M Y, et al. Unsupervised learning on scientific ocean drilling datasets from the South China Sea[J]. Frontiers of Earth Science, 2019, 13(1):1-11.
[6] Goldsmith J. Unsupervised learning of the morphology of a natural language [J]. Computational Linguistics, 2001, 27(27): 153-198.
[7] Wang Y, Ye Z, Wan P, et al. A survey of dynamic spectrum allocation based on reinforcement learning algorithms in cognitive radio networks [J]. Artificial Intelligence Review, 2019(2):1-14.
[8] Jin Y, Liu X, Chen Y, et al. Land-cover mapping using Random Forest classification and incorporating NDVI time-series and texture: a case study of central Shandong[J]. International Journal of Remote Sensing, 2018, 39(23):1-21.
[9] Everitt B S. Classification and regression trees [J]. Monterey, CA: Wadsworth & Brooks/Cole Advanced Books & Software, 1984.
[10] Bing Z, Wei Z, et al. An Empirical study on Predicting Blood Pressure using Classification and Regression Trees [J]. IEEE Access, 2018, PP (99):1-1.
[11] Rezvani S, Wang X, Pourpanah F. Intuitionistic Fuzzy Twin Support Vector Machines [J]. IEEE Transactions on Fuzzy Systems, 2019, PP (99):1-1.
[12] Aizerman M A, Braverman, et al. Theoretical foundations of the potential function method in pattern recognition learning [J]. Automation and Remote Control, 1964, 25: 821–837.
[13] Boser B E, Guyon I M, et al. A training algorithm for optimal margin classifiers [J]. Proceedings of the fifth annual workshop on Computational learning theory – COLT ’92, 1992.
[14] Jiang L, Zhang L, Li C, et al. A Correlation-Based Feature Weighting Filter for Naive Bayes [J]. 2019, 31(2):201-213.

[15] Caruana R, Niculescumizil A. An empirical comparison of supervised learning algorithms [J]. International Conference on Machine Learning, 2006.