Diabetes Analysis Based on Naive Bayes Model

Linyue Zhang*

Department of Mathematics, Nankai University, Tianjin, China

*Corresponding author: 2110588@mail.nankai.edu.cn

Abstract. As a matter of fact, diabetes has been one of the most disturbing diseases in human beings' daily life in recent years. With this in mind, the method to predict diabetes has been conducted through artificial intelligence technology including machine learning as well as big data analysis. On this basis, this study is going to discuss the possibility of predicting diabetes via one of the widely used machine learning scenarios, i.e., Naive Bayes models. According to the analysis, the result shows that although the accuracy is improved after parameter changing, the value is 75.61%, still below 80%. In other words, it indicates that the features offered nowadays are still abundant enough to realize high accuracy predictions. Despite the low accuracy, the result still can be used and applied in many fields, such as the pre-processing of another model's building. Overall, these results shed light on guiding further exploration of diabetes analysis.

Keywords: Naive Bayes; machine learning; diabetes prediction; artificial intelligence.

1. Introduction

Diabetes is a long-term condition. It occurs when the body is unable to adequately adjust the insulin produced by the pancreas or when insulin synthesis is insufficient. Blood sugar is controlled by the hormone insulin. Hyperglycemia, or high blood sugar levels, is a widespread side effect of uncontrolled diabetes that over time can gravely affect a number of body systems, including neurons and blood vessels [1]. Despite the reality that diabetes is a fatal condition and has been acknowledged in ancient documents, it did not appear that doctors or therapists had frequently worked with it in the past [2]. However, with the development of economics and scientific technology, people are much more easily to enjoy a feast, which leads to the increasing number of diabetes patients. Adults over the age of 18 had diabetes in 8.5% of cases in 2014. In account, 1.5 million deaths were directly linked to diabetes in 2019, of which 48% deaths happen before the age of 70 [3]. Between 2000 and 2019, deaths from diabetes increased by 3%. Diabetes mortality in low- and middle-income countries has risen by 13%.

According to the research, before a patient is diagnosed being a diabetes, there is a stage called prediabetes. During this period, the patients are not really a diabetes yet, while still have a great risk to become one. In other word, they are backups. Though with a high risk, this kind of patient have chance to become a perfectly healthy person as long as they take their body seriously. Therefore, it is important to predict whether a patient would have diabetes so that he or she could pay more attention on it. Many biologists have attempted to develop techniques and systems that will aid in the prediction of diabetes using the data gathered by the diabetes association. Utilizing several machines learning methods, including classification and association algorithms, they have created prediction models. The most popular algorithms were Decision Trees, Support Vector Machine (SVM), and Linear Regression [4]. This has made scholars from both biological and other fields like computer science focus on increasing the accuracy either by creating models or combining different classifiers. The public Pima Indian Dataset collected from the UCI repository was often used in these kind of research [5]. Researchers use different type of method to deal with the same problem. Although most of them have accomplished several goals and made achievement, the final accuracy produced by these methods varied. Former research shows that, using models KNN, Random forests and SVM has the accuracy rates of average 72%, 71.74% and73.34% [6]. It's easy to be seen that although all of the method ends up a higher than 70% accuracy, they are not accurate enough to be a real diagnose model.

This study is going to use a new model called Naive Bayes model to classify the data and predict whether or not a patient would get diabetes. With the use of the public Pima Indian datasets to train and test the model, the calculation of the accuracy would be conducted and the best fit parameter would be found. It is also an attempt to improve the existing diagnose and treatment of diabetes. Hoping this could help more patient stop their course of disease in the stage of prediabetes. The rest of the paper is organized as follows. Section 2 provides the information of the data and shows the method this study uses. Section 3 describes the Results and Discussion which contains feature engineering, model performance, explanation. Limitations and Future outlooks are discussed in Section 4. Finally the paper is concluded in Section 5.

2. Data and Method

2.1. Data

The National Institute of Diabetes and Digestive and Kidney Disease provided the dataset in its original form. The dataset's purpose is to use specified forensic statistics included in the collection to diagnostically predicted whether a patient, who is a female of at least 21 years old of Pima Indian ancestry, has diabetes. These particular instances were chosen under a number of constraints from a larger database, which means this dataset is already pre-processed. This dataset contains 9 columns and 768 rows. There are 8 columns useful for the model training, because the last column is the "Outcome", indicating whether the patient has diabetes.

2.2. Model

Since the goal is to predict the outcome which is 0 or 1, there is a method to predict it from the classification point of view. Bayes naive model is a common classification model in sklearn. To make the samples classified, it is necessary to calculate the conditional probability. P(a|b) means the probability of result A occurring under condition B. The appraoch to calculate conditional probability is called the Bayesian formula. The naive Bayes techniques are a classic supervised learning algorithms that use Bayes theory with the "naive" assumption that each pair of features is conditionally independent. The Bayes theorem establishes the following relationship given the class variable y and the dependent feature vector, $x_1\ to\ x_n$:

$$\mathrm{P}(\mathrm{y}|\mathrm{x}_1,\mathrm{x}_2,\ldots,\mathrm{x}_\mathrm{n}) = \frac{\mathrm{P}(\mathrm{y})\mathrm{P}(\mathrm{x}_1,\mathrm{x}_2,\ldots,\mathrm{x}_\mathrm{n})}{\mathrm{P}(\mathrm{x}_1,\mathrm{x}_2,\ldots,\mathrm{x}_\mathrm{n})} \quad (1)$$

Given $\mathrm{P}(\mathrm{x}_1,\mathrm{x}_2,\ldots,\mathrm{x}_\mathrm{n})$ is constant, the classification rule described below can be applied:

$$\mathrm{P}(\mathrm{y}|\mathrm{x}_1,\mathrm{x}_2,\ldots,\mathrm{x}_\mathrm{n}) \text{ is proportional to } \mathrm{P}(\mathrm{y})\prod_{\mathrm{i}=1}^{\mathrm{n}}\mathrm{P}(\mathrm{x}_\mathrm{i}|\mathrm{y}) \quad (2)$$

Therefore,

$$\hat{y} = arg\max_{y}\mathrm{P}(\mathrm{y})\prod_{\mathrm{i}=1}^{\mathrm{n}}\mathrm{P}(\mathrm{x}_\mathrm{i}|\mathrm{y}) \quad (3)$$

There are five types of naive bayes algorithm in sklearn. If look at its source code, the following source code can be seen at the beginning of the module file: 'BernoulliNB', 'GaussianNB', 'MultinomialNB', 'ComplementNB' and 'CategoricalNB'. Different naive Bayes classifiers differ from one another based on the distributional assumptions they make. Despite what seem to be its overly simplistic assumptions, naive Bayes classifiers have excelled in a variety of real-world situations.

To test the accuracy of the model, the common method is that divide the data into two-part, one part is for training the model, another part is for testing the model. Compare the test set's real outcome with the prediction made by the model used the test set's data, the accuracy can be estimated. Nevertheless, this dataset cannot be test like this. In machine learning field, the sample size is at least a few hundred. If it is a classification problem, like in this case, then it is best to have thousands of samples. However, as the first part of this section said, there are only 768 rows in the dataset. This quantity is just above the qualified line, but not good enough. Therefore, cross validation should be

considered. In order to conduct cross validation, the whole collection must first be divided into n parts with the same size. Afterwards, one goes through each of these n subsets in turn, using the current subset as the validation set and the rest of the data points as the training set for the modeling process. The final accuracy is determined using the average of the accuracy evaluations from the n-time process. In practical experiments, k is usually taken as 10. However, in this case, since the data is not large enough, the k is taken as 5 (seen from Fig. 1). More specifically, when using the"cross_val_score" from sklearn, the parameter "cv" is set to 5.

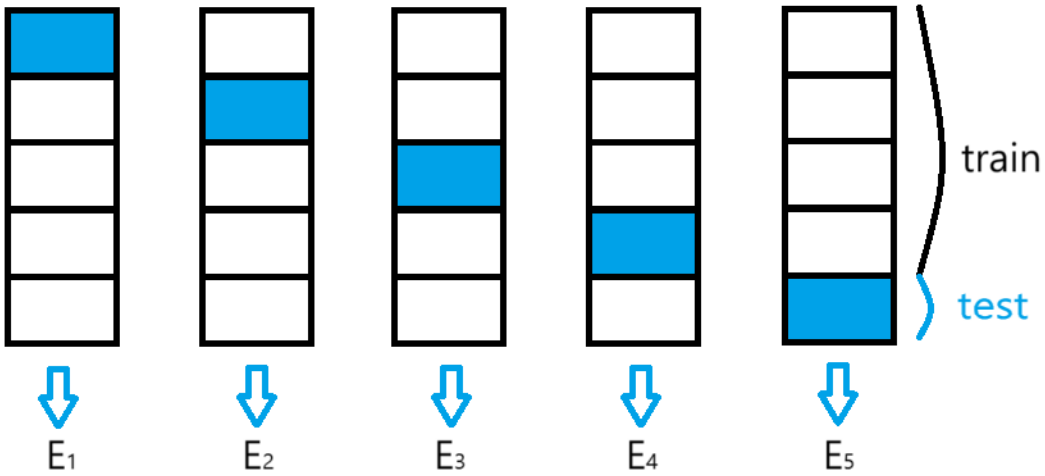


Fig. 1 The cross validation procedure chart when cv equals 5. The final score is the average of each outcome of iterations.

3. Result and discussion

3.1. Featuring engineering

The very first step of the modeling process is to see the feature and the correlation of the data. After seeing though the columns, it is possible to discover that some of the indexes are correlated like skin thickness and age. Whereas, most of them seems independent. Besides, there is only one decisive index which is the "Outcome". In this column, if outcome equals 1, the patient has diabetes; if outcome equals 0, then the patient does not have diabetes. To visualize the correlation between each variable, a heatmap via maplotlib library in python is made (seen from Fig. 2). The lighter the color is, the greater the positive correlation is. The darker the color is, the greater the negative correlation is.

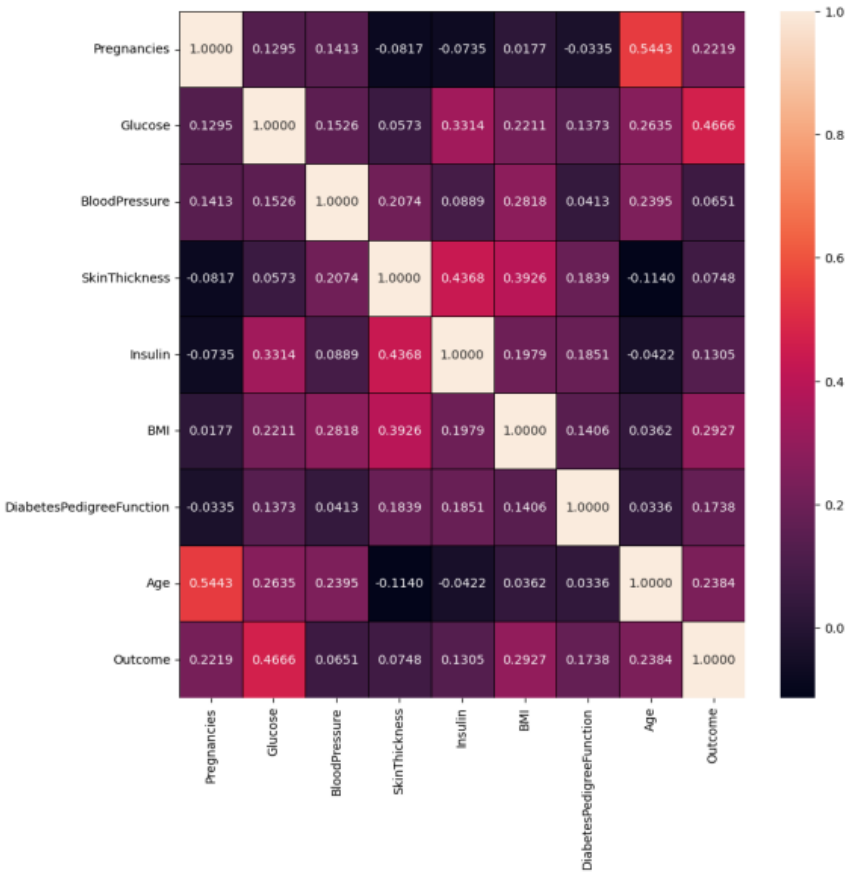


Fig. 2 The correlation heatmap of the dataset

Looking through the figure, the overall color is dark, despite the diagonal grids meaning the correlation of the index and itself which is obviously equals one hundred percent. Therefore, the index of dataset conforms to the "naive" assumption that each feature is independent. The following modeling process is reasonable. Then, one sets a hue value equals "Outcome". According to the distribution of the scatter plot and the diagonal line chart (not shown here), it is easy to find that diabetic and non-diabetic patients differ slightly or greatly in different type of the indexes. For example, the Glucose， it is a very important indicator for the diagnosis of diabetes.

3.2. Model performance

The first step is to find out which model should use in this project. The method is that Look at the souce codes to understand their application. It shows that there are 2 algorithm might match the problem. The first one is the GaussianNB which is suitable for analyzing the continious data that obey Gaussian distribution, another one is the CategoricalNB which is suitable for classification with discrete features of classification distribution, Then, one tests each algorithm and use the same deviation of the data and calculate its accuracy via "cross_val_score" to maintain uniformity. The results are given in Table 1. Accordingly, in this case, the best matched model is GaussianNB model. The second step is to tune the parameters. The only parameter can be changed in GaussianNB model is its "var_smoothing" parameter. Set a loop to let "var_smoothing" equals 1, 0.1, 0.01, 0.001, ..., 1e-09 and record each accuracy. After that, draw a learning curve map to show its accuracy changes.

Seen from Fig. 3 and Fig. 4, because the evaluation index is the cross validation score, more emphasis should be put on the red line. When var_smoothing is around 1e-6, the model has best accuracy. Hence, set a loop again and let "var_smoothing" equals 1e-6, 2e-6,..., 9e-6, 1e-5 recording each accuracy. After that, draw a learning curve map to show its accuracy changes. When var_smoothing equals 4e-6, the model has the best accuracy score 0.756. After determining the parameter to use, add a input module to let the users to input their data and the model would output

the outcome which only has 2 options, 0 or 1. Zero means that the inputter would not have diabetes and one means opposite like what is recorded in the original dataset.

Table 1. The accuracy of each Naive Bayes model

	BernoulliNB	GaussianNB	MultinomialNB	ComplementNB	CategoricalNB
accuracy	0.6510	0.7417	0.6510	0.6042	0.7256

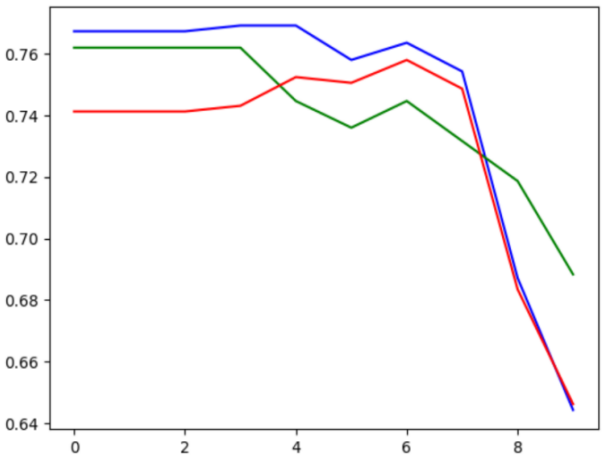


Fig. 3 First learning curve map. Blue line is the accuracy of train set. Green line is the accuracy of test set. Red line is the cross validation score. Blue line is the accuracy of train set. Green line is the accuracy of test set. Red line is the cross validation score.

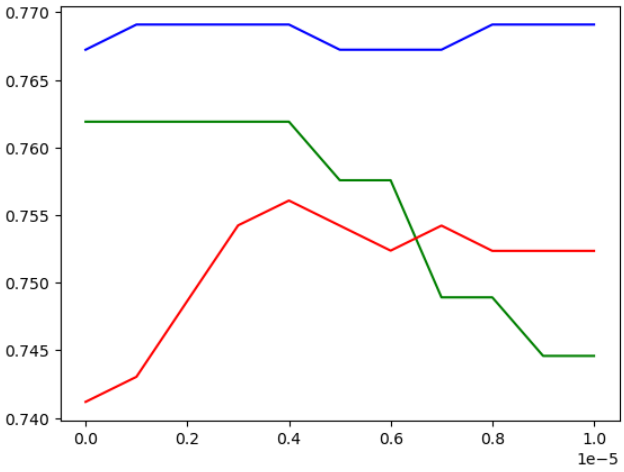


Fig. 4 Second learning curve map. Blue line is the accuracy of train set. Green line is the accuracy of test set. Red line is the cross validation score.

3.3. Explanation

After all, the Gaussian Naive Bayes model with var_smoothing equals 4e-6 is established. The users can predict whether a patient would suffer from diabetes by inputting his or her data of each required index. It can be used both medical diagnosis and disease prevention. The accuracy of the final model is 0.7561. Compared to the original model when the parameter “smooth_var” is not set whose accuracy score is 0.7417, the accuracy has improved.

4. Limitation and Future Outlooks

There are still some inadequacies and this study would ameliorate following method to improve it. First of all, although the dataset has been pre-processing, there are many noises in it and it would influence the model training and test. This study would improve it with a more complex pre-process

and try to delete the noises. Secondly, either the first accuracy or the last accuracy is less than 0.8, therefore it is necessary to consider again what model should be used. Compared to more sophisticated methods, naive Bayes learners and classifiers are known as a decent classifier, but also known to be a bad estimator [7]. Besides, like what has shown in the featuring engineering part, some of the indexes are dependent slightly which does not exactly obey naive assumption [8-10]. For all this, the model still has its useful aspect. It is a good classification. The result can be used in many further machine learning models. It can improve other models' accuracies.

5. Conclusion

To sum up, users can input their number of pregnancies, glucose, blood pressure, skin thickness, insulin, BMI, diabetes pedigree function, age to know wherher they will get diabetes. The accuracy of original gaussian naive bayes model is around0.7417 and after setting a parameter var_smoothing = 4e-6, the model's accuracy is around 0.7561. The inadequacies are that there are still some noises in the dataset and the naive bayes model is not a good predictor. The result would greatly improve if doing something about these aspects. However, this research still has some practical value. Although naive bayes model is not a good estimator, it is a good classifier. The research does improve its classification accuracy. These results can be used in further study, such as the logistic regression's model building.

References

[1] World Health Organization, World Health Organization. Definition and diagnosis of diabetes mellitus and intermediate hyperglycemia: report of a WHO/IDF consultation. Geneva: World Health Organization; 2006: 1–50.

[2] Roglic G. WHO Global report on diabetes: A summary. International Journal of Noncommunicable Diseases 2016, 1(1): 3-8.

[3] Bansal N. Prediabetes diagnosis and treatment: A review. World journal of diabetes 2015, 6(2): 296.

[4] Larabi-Marie-Sainte S, Aburahmah L, Almohaini R, Saba T. Current Techniques for Diabetes Prediction: Review and Case Study. Applied Sciences, 2019, 9(21): 4604.

[5] Varma K V, Rao A A, Lakshmi T S M, Rao P N. A computational intelligence approach for a better diagnosis of diabetic patients. Comput. Electr. Eng. 2014, 40, 1758–1765.

[6] Kandhasamy J P, Balamurali S. Performance Analysis of Classifier Models to Predict Diabetes Mellitus. Procedia Comput. Sci. 2015, 47, 45–51.

[7] Das A, Verma R. Automated email generation for targeted attacks using natural language. arXiv preprint 2019, arXiv:1908.06893.

[8] Wang D. A comparison of machine learning methods to predict hospital readmission of diabetic patient. Studies of Applied Economics, 2021, 39: 4.

[9] Wang X Z, He Y L, Wang D D. Non-naive Bayesian classifiers for classification problems with continuous attributes. IEEE transactions on cybernetics, 2013, 44(1): 21-39.

[10] Ji S, Carin L. Cost-sensitive feature acquisition and classification. Pattern Recognition, 2007, 40(5): 1474-1485.