KNN algorithm of early warning system for applied research course

Zian Li[1,*], Cui Zhao[2]

[1]School of Wuhan Business University, Hubei, China
[2]Adviser, School of Wuhan Business University, Hubei, China

Abstract. This paper designed a small course early warning system based on KNN algorithm, aimed at students haven't finished the course of time can be completed by yourself some predict their courses by chance. In this paper, the basic principle of KNN algorithm is briefly introduced, and the course warning system is modified by Manhattan distance with added weights. This paper briefly describes the basic framework of this model and introduces the application of KNN algorithm in this model. Through a large number of basic experimental data to test the training, using figures to show, finally get the curriculum early warning system model, to achieve the effect of curriculum early warning.

1 Introduction

Before today's college students in the curriculum examination is only to pass a judgment about the course, this will lead to the students for the future study of tension, which affects the enthusiasm of students on this course. This course grades before the early warning system will be in the curriculum examination to a student's course can be through a more accurate prediction, in order to reassure students, make students actively to put energy into complete the requirements of this course.

In this paper, an early warning system of course performance based on KNN algorithm is proposed. Starting with the introduction of KNN algorithm, the realization process of the early warning system of course grades is gradually introduced. Course grade warning system will now past learning of this course students collected data as sample data, by through and not through into two categories, extracted from these two kinds of attributes related to whether through this course, students compare with the target, which can more accurately get the student through the conclusion of this course.

2 KNN algorithm introduction

2.1 KNN algorithm description

KNN algorithm, also called K Nearest Neighbor (K - on his Neighbor) is a kind of typical passive learning, its basic idea is: if a sample in the feature space of the K most similar (i.e., in the feature space adjacent) most of the sample belongs to a certain category, you can determine the sample also belong to this category. Among them, the selected neighbor samples are training samples with known categories.

Algorithm principle: the basic condition is that there is a sample data set (the training sample set), and sample concentration of each data attribute to find a classification. After inputting an unclassified data, extract each characteristic attribute related to the sample data set in the new data, then compare the data in the sample data set one by one, extract k sample data (k nearest neighbors) that are most similar to the feature attributes of the new data, and then classify the new data according to the classification of the extracted sample data that occurs most frequently.

For the convenience of calculation and demonstration, the distances used for k nearest neighbors are extracted using the Manhattan distance. The Manhattan distance is the absolute wheelbase sum of two points on the standard coordinate system. In the plane rectangular coordinate system, for example, $\mathrm{i}(x_1 - y_1)$ and $j(x_2, y_2)$ between the Manhattan distance is:

$$D(\mathrm{i}, j) = |x_1 - x_2| + |y_1 - y_2|$$

The algorithm is a typical negative learning device, that is, the KNN algorithm does not show the training process in advance. This means that the KNN algorithm in import won't training, the training sample just save the training data, the real training process is the input in the target sample calculation and training before, so the training of the KNN algorithm cost is zero.

2.2 KNN algorithm improvement

When the algorithm implementation, you will find the two very serious problem. The first is that KNN

algorithm needs to store all the training samples, which results in that the more accurate the classification is, the larger the sample set that the algorithm needs to store and the more system resources it takes up. The second problem is the KNN algorithm to sample concentration of each sample for distance calculation, the heavy calculation will also reduce the KNN classification efficiency of the algorithm.

Then, according to the second problem, we has carried on the improvement of KNN algorithm, which is grouped near quick search method. The grouping quick search approach is to first decompose the sample set itself into several different groups according to the proximity relation (classification of attributes), then find the position of the center of mass of each group, take the center of mass as the representative point of each group, find the distance between each center of mass and the target sample, and apply the ordinary KNN algorithm to achieve the classification effect. This improved version of the algorithm with no need to sample concentration of each sample data for the calculation of distance, so the group quickly search method to reduce the amount of calculation of KNN algorithm, solving the problem of large amount of calculation of KNN algorithm. Unfortunately, the improved algorithm still stores all of the sample data and does not save memory.

3 Application and realization of KNN algorithm

3.1 Data preprocessing

The passage of a course is determined by many factors. In order to simplify the operation and calculation of the experiment and make it easier for us to explain and understand the whole article, we selected two factors that students can control, namely the rate of class attendance and the degree of homework completion. For the purpose of precise algorithm calculation, we introduce weights. By weighting different data attributes (control factors) in a way that shows the importance correlation between different attributes. For example, in this experiment, the attendance rate accounted for 20% of the final grade of the course, the homework completion rate accounted for 30% of the course, and the final exam score accounted for 50% of the course. Then the weight value of attendance rate is 2, and the weight value of homework completion is 3. Table 1 below is a demonstration of some data samples.

Table 1. Part of the data samples

Student No.	Attendance rate (weight: 2)	Job completion (weight 3)	Whether through
1	90%	90%	Y
2	70%	100%	Y
3	20%	50%	N
...	...	...	...
i	100%	80%	Y

3.2 Application of KNN algorithm

Since this is a two-dimensional data structure, we can represent the sample set in a planar cartesian coordinate system. For example, figure 1 below:

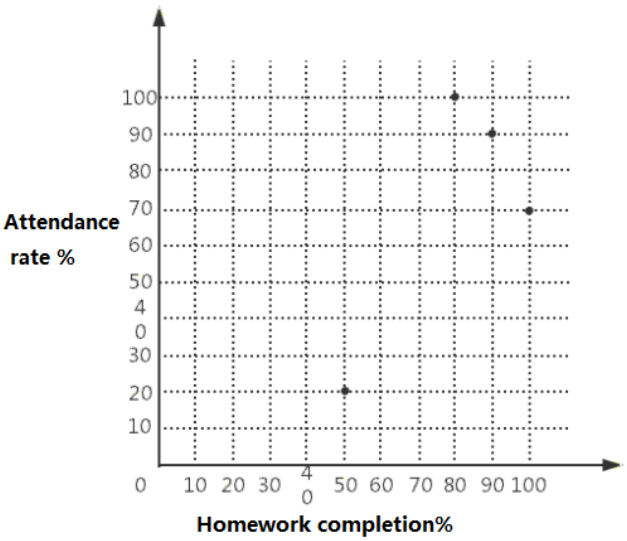

Figure 1. Scatter plot of sample data

After the above data samples are obtained, the improved KNN algorithm is carried out step by step. Input a target sample, calculate the distance between the training samples of the target sample, and calculate through repeated experiments to get a suitable value of k. We believe that within the k value most belong to which a type of classmates, for the classification of the classification of the students. The specific step-by-step algorithm is as follows:

Step 1: divide the training sample set that has been entered in advance into two groups by type: pass and fail.

Step 2: find the location of the center of mass of the two groups of data that have been divided into two categories.

Step 3: since only two groups are set, the value of k is set to 1.

Step 4: calculate the distance between each group and the target students using a modified version of Manhattan distance (Manhattan distance with added weights). The specific formula is:

$$D(i,j) = \omega_1 \cdot |x_1 - x_2| + \omega_2 \cdot |y_1 - y_2|$$

Step 5: determine the category of the nearest grouping when k value is 1. In our opinion, the category passed by the student is the category of the grouping.

Step 6: when the class of the student is "fail", a warning signal is issued to remind the student that "the course result is about to fail" to achieve the effect of course warning. In the following test, a small sample of students was selected

4 Model design and test of curriculum early warning system

4.1 Design of curriculum early warning system model

This paper implements a curriculum early warning system based on KNN algorithm. The system is based on the data and processing of the initial data, the appropriate classifier constructed by KNN algorithm, and then the output of target student types is classified by the classifier, and finally the warning is judged by the type of students. The following is the system diagram of the course warning system.

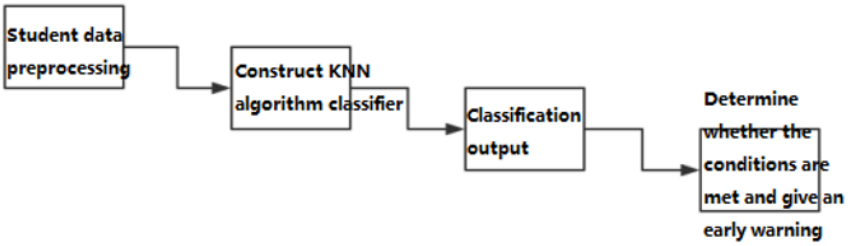

Figure 2. Block diagram of curriculum early warning system

4.2 K value test

In this paper, 100 pieces of student data were collected from previous students and divided into 10 groups. 9 of 10 in each group as the training sample, a target sample for testing. The research shows that in the model of curriculum early warning system designed in this paper, the size of k value has a great influence on the accuracy of curriculum early warning system. Therefore, it is preferred to study the value of k. Theoretically speaking, the larger the value of k is, the nearest neighbor may contain data points that are far away from the target sample, thus increasing the error of the nearest neighbor classifier and even leading to the classification error of the classifier. The smaller the k value, the easier it is to the nearest neighbor classifier was affected by the noise in the training data, resulting in excessive fitting, do more harm than good. After repeated experimental tests we obtained the following data as shown in Table 2.

Table 2 prediction accuracy under different k values

K value	3	5	7
precision	93.45%	92.97%	91.89%

According to the results obtained from this experiment, under the premise of 9 training samples and 1 target sample, the most suitable k value to be used in this model is 3. In other words, when k value is 3, the accuracy of this model is relatively high, reaching 93.45%.

4.3 System function test

After the most suitable k value is obtained through experiments, the system test will be carried out next. The testing process is still divided into steps.

Step 1: select 10 students from the first group from 100 students, input 1-9 students as training samples in advance, and take the 10th student as test samples.

Step 2: confirm that k value is 3, that is, KNN algorithm will select the three nearest neighbors closest to the test sample.

Step 3: initialize a priority queue, which grows from small to large according to the distance from the test sample. The size of the queue is k=3.

Step 4: extract k=3 samples from the training sample, calculate the distance from each sample to the test sample, and put them into the initial priority queue according to the rules of the queue. So the last sample in the queue is the one with the greatest distance between the three samples.

Step 5: traverse the entire sample set in the remaining training sample set, calculating the distance from the sample to the test sample for each sample traversed. Let's call that distance D.

Step 6: every D obtained shall be compared with the largest D(Max) in the queue. When D > and D(Max), the sample shall be discarded. When D < D(Max), the training samples corresponding to D(Max) are deleted

from the priority queue, and the training samples corresponding to D are added

Step 7: repeat steps 5 or 6 until the loop is complete.

Step 8: determine the category of the test sample by the category of the sample in the priority queue. Compare known test sample categories to see if the test was successful.

Step 9: repeat one to eight steps until the end of the test.

To simplify the presentation process, only one set of test data will be listed below, as shown in tables 3 and 4.

Table 3. Sample table of training data

Sample No.	Attendance rate % (weight: 2)	Job completion rate % (weight 3)	Whether through
1	90	90	Y
2	100	90	Y
3	10	20	N
4	80	100	Y
5	70	100	Y
6	90	80	Y
7	50	30	N
8	100	70	Y
9	30	40	N

Table 4. Test data sample table

Sample No.	Attendance rate % (weight: 2)	Job completion rate % (weight 3)	Whether through
10	80	70	Y

In order to facilitate the observation of the operation of KNN algorithm, all sample data are presented in two-dimensional coordinates. As shown in figure 3.

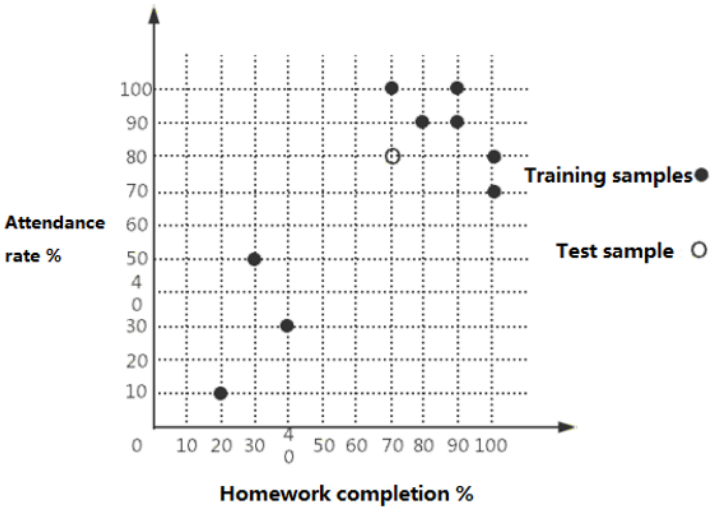

Figure3 scatter diagram of test data

Obviously, when k value is 3, the three data types in the priority queue are all "pass", then we can think that the predicted situation of student no. 10 predicted by this model should also be "pass". Through comparison with the data in table 4, it is known that the real situation is also "passed". Therefore, we can infer that the curriculum early warning model based on KNN algorithm is reasonable.

5 Conclusion

KNN algorithm is a very simple and efficient algorithm and easy to use. The disadvantage of this algorithm is that it needs to store all training samples and carry out heavy distance calculation. However, the course warning system discussed in this paper, with relatively small sample data and relatively monotonous sample attributes, perfectly makes up for its shortcomings.

The above mentioned course warning system solves the problem that college students can only infer whether their courses can pass through by guessing an inaccurate and vague feeling. Enable the students to more attentively to go into the middle course. The model introduced in this paper can be used not only in curriculum warning but also in academic warning. The weighted KNN algorithm used in this model can be used in a wider range, such as the prediction of the stock market, the prediction of the transaction price of an item in the market and so on. When reflected in the transaction price prediction: under the circumstance that the price of a certain item in the market is affected by a series of different factors such as production, demand and quality, the weighted KNN algorithm can be used to calculate the Manhattan distance, so as to obtain a prediction closer to the real situation.

Acknowledgements

Supported by the project of Construction of campus mailbox under Internet Plus background -- Taking Wuhan Business University as an example, the NO. of project is 201811654048

References

1. Research on KNN algorithm for big data classification[J]. Geng Lijuan, Li Xingyi. Computer Application Research. 2014(05)
2. Small sample KNN classification algorithm based on k-nearest neighbor graph [J]. Liu Yingdong, Niu Huimin. Computer Engineering. 2011(09)
3. KNN Model-Based Approach in Classification. Guo,G,Wang,H,Bell,D.et al. Otm Confederated International Conferences. 2003
4. Su Yongjie. Detailed explanation of the principle of KNN nearest neighbor algorithm[Z].2017.
5. Fan Ruizhi. Student loan risk model based on KNN algorithm[J]. Electronic Test, 2019(Z1): 80-81+84.
6. Yang Yanxin. Road construction risk identification based on KNN algorithm[J]. Resources Information and Engineering, 2019, 34(02): 137-138.
7. Yang Yanxin. Road construction risk identification based on KNN algorithm[J]. Resources Information and Engineering, 2019, 34(02): 137-138.
8. Lu Kai, Xu Hua. ML-KNN algorithm based on the nearest neighbor distance weight [J/OL]. Computer Application Research: 1-5 [2019-06-20]. https://doi.org/10.19734/ j.issn.1001-3695.2018.09.0738.