Assignment 2: k-Nearest Neighbour (kNN)

Kaixuan Chen: z5099792

Manqing Dong: z5122290

Arathy Satheesh Babu: z5131176

June 3, 2017

1 Introduction

The goals of this project are to implement two tasks namely classification of radar returns and numeric prediction of the auto price. In our project we have used K nearest neighbor algorithm (k-NN) [1] to implement both classification [11] and numeric prediction [11]. In both cases, the input consists of the k closest training examples in the feature space.

The classification of radar returns from the ionosphere [3] categorize whether the radar return is good or bad. In k-NN classification, the output is a class. An instance is classified by a majority vote of its neighbors, with the instance being assigned to the class most common among its k nearest neighbors. The response variable is categorical with two levels:

- 'g' represents good radar returns.
- 'b' represents bad radar returns.

The target in the radar returns is the free electrons. The classification of radar returns is a significant problem as the free electrons are important in terms of high frequency(HF) radio propagation. HF radio waves bent and are eventually

reflected back to earth because of the free electrons. As proposed in [3], the greater density of the electrons cause, higher frequencies to be reflected.

The second task is a regression problem to predict the price of a car. Predicting the price of used cars in another significant problem because of its very high commercial importance. Residual value of the car needs to be predicted with accuracy. Under-estimation of the residual value will increase the installments to be paid to the financers by the clients. Over-estimation of the residual value will lead to difficulty for the seller to sell the car. Predicting the resale value of a car is not a simple task. The value of used cars depends on a number of factors. In our task, there are 26 attributes which includes 15 continuous attributes, 1 integer attribute and 10 nominal attributes. We have used KNN to do this numeric prediction by giving 14 continuous attributes and 1 integer attribute to predict the price of the car.

2 Related Work

Classification is the core and basic technology in data mining. The most popular classification methods include decision tree, Bayesian classification, k-NN and neural network, etc. Among these methods, k-NN is a simple, effective and non-parametric, which can be applied to diverse fields including text classification, pattern recognition and image classification. k-NN [1] is an improvement of Nearest Neighbor(NN) algorithm [4]. Because k-NN algorithm was proposed early, the shortcomings and weakness of k-NN are gradually revealed with the advancement of other algorithms and techniques. Therefore, a great number of improved k-NN algorithms have been emerging.

The k-NN algorithm stores all the sample data for the training set, which results in significant storage overhead and computational cost. [2] proposed to reduce the data size by delete redundant data which are irrelevant to the classification. Because k-NN needs to calculate all the distances from test points to training samples to find K nearest neighbors, the computational cost is very high when

the data is huge. In order to speed up the k-NN search process, [6] proposed a k-NN Search algorithm based on partial distance calculation in wavelet domain. One of the most obvious drawbacks of traditional k-NN decision rules is that when the sample distribution density is not uniform, only considering the order of K nearest neighbors regardless of their distance will affect the performance of the classification. In fact, in the case of practical design of the classifier, because some categories are easier to obtain than others, it tends to result in an imbalance between the various categories of training samples. The existing improved methods include homogenizing distribution density. [10] used a large number of neighbor sets instead of a single set of k-NNs and obtained the relative support values by accumulating the support of the neighboring data sets for different categories, thus improving the adjacency rules. In addition, basic k-NN algorithm calculates the similarity based on the Euclidean distance, which makes the k-NN algorithm very sensitive to the noise characteristics. In order to avoid the drawbacks in the traditional k-NN algorithm, different weights are assigned to the features in the distance formula of measure similarity. The weights of the features are generally set according to the function of each feature in classification. People have studied the methods to adjust the weights and improved the performance of k-NN classifier such as k-nearest-neighbor with distance weighted (k-NNDW) [9]. Researchers also proposed to combine k-NN with naive Bayes [12, 7, 5, 8], which is more efficient by deploying probabilities in the neighborhood of testing data.

3 Method

In this project, we are proposed to use k-NN method to deal with a classification problem and a regression problem. For the classification problem, the idea of k-NN method is to use the average value of k nearest nodes to be the predicted value; and the idea of k-NN regression is to calculate the average of the numerical target of the k nearest neighbors.

Based on this idea, we put k-NN into our data sets with different k values. And

for evaluating the classification prediction performance, we used accuracy; and for the regression prediction, we used mean average deviation (MAD) and Mean absolute percentage error (MAPE) as the evaluation methods. The project asked us to use leave one out cross validation. For comparing, we also used 5-fold cross validation, which means divide the samples randomly into 5 groups and use four groups' data as the training data set and the other as the test data set.
The details of the experiments are listed in the Experiments and Discussion part.

4 Experiments and Discussion

1. For Classification. Evaluate your system by leave one out cross- validation(LOOCV).

Here for comparison, we conducted two experiments. Firstly, we evaluated the algorithm by leave one out cross-validation as required. Our target is to identify if the data is good or bad, so this is a binary classification problem. The dataset contains 351 samples. Therefore, each time we left only one sample out as test data and trained model with the other 350 data. We conducted the experiment for 351 time to test all the 351 samples and took the average of the percentage of the right results for classification as accuracy. The results for different values of K are shown as Table 1.

Table 1: The Accuracy of LOOCV classification

	k=1	k=5	k=10	k=15	k=20
Accuracy	0.8660969	0.8461538	0.8404558	0.8319088	0.8461538

In addition, we performed a 5-fold cross validation. Firstly, we mixed the 351 objects into a random order, and then we divided the reordered objects into 5 parts, and took each fold as the test dataset successively. The results are shown in Table 2.

To make it clearer, a line chart is presented as Figure 1. It is clear that no

Table 2: The Accuracy of 5-fold cross validation classification

	k=1	k=5	k=10	k=15	k=20
First Fold	0.8428571	0.8142857	0.8142857	0.8285714	0.8000000
Second Fold	0.8857143	0.8857143	0.8714286	0.8571429	0.8428571
Third Fold	0.8714286	0.8285714	0.8142857	0.8285714	0.8000000
Fourth Fold	0.9142857	0.8857143	0.8714286	0.8857143	0.8571429
Fifth Fold	0.8873239	0.8309859	0.7887324	0.7887324	0.7887324
Average	0.8803219	0.8490543	0.8320322	0.8377465	0.8206036

Table 3: The Results of LOOCV Prediction

	k=1	k=5	k=10	k=15	k=20
MAD	1481.723	1611.97	1830.833	1838.964	1782.295
MAPE	0.1224995	0.1314237	0.1436861	0.1406554	0.1338888

matter for LOOCV or 5-fold cross validation, the accuracy reaches the highest point when K=1. It can be inferred that in this dataset, each point is only related to the nearest point to it. That is why we got lower accuracy when we selected more neighbours. Another notable fact is that the line of LOOCV is more stable than that of 5-fold cross validation because for LOOCV, the size of training set is far larger than the size of test set, which reduce the dependency between the results and the value of K.

2. Numeric Prediction. Test your system on dataset autos. The second task is giving us 26 variables, which 15 of them are continuous variables, 1 is integer, and 10 are categorical variables. And in continuous variables, one is the price of the car which is the response variable.

k-NN method is valid in numeric variable. So here we leave out the categorical variables, and use other variables as the independent variables. And actually for the categorical variables, we can make them as the dummy variables first, and

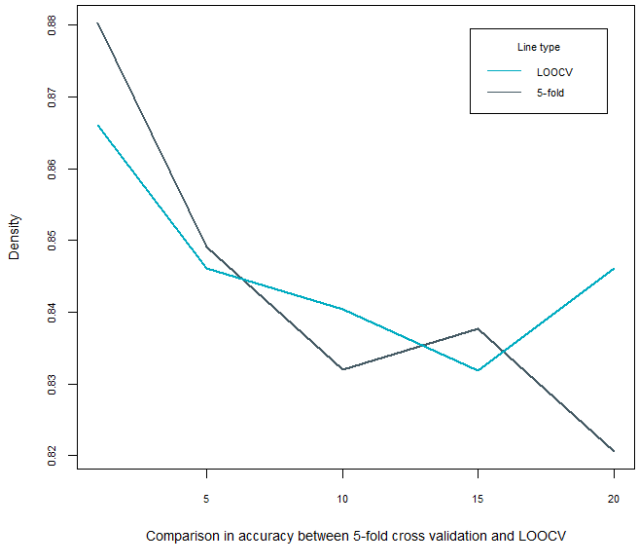


Figure 1: Comparison in Accuracy between 5-fold cross validation and LOOCV

then regard them as the numeric variables. Here we take MAD and MAPE as our measurements for the prediction performance.

The definition of the MAD is:

$$\frac{\sum_{t=1}^{n} \mid y_t - \widehat{y}_t \mid}{n} \tag{1}$$

Where, y_t is the real value, and $\widehat{y}_t$ is the predicted one.

The definition of MAPE is:

$$\frac{\sum_{t=1}^{n} \mid (y_t - \widehat{y}_t)/y_t \mid}{n} (y_t \neq 0) \tag{2}$$

Also, we compared the results between 5-fold cross validation and leave one out cross validation (LOOCV) with different k values (k= 1, 5, 10, 15, 20). The results are shown in Table 3 to Table 5.

Similarly, two line charts are presented in Figure 2.

From the figures we can infer that the prediction showed the good performance with small k value (k=1), but with the increase of the k-value, the prediction performance becomes worse at first and then become better. That might come from the situation with overfitting. But overall the leave-one-out cross validation performs better with small k values.

Table 4: The MAD of 5-fold cross validation Regression

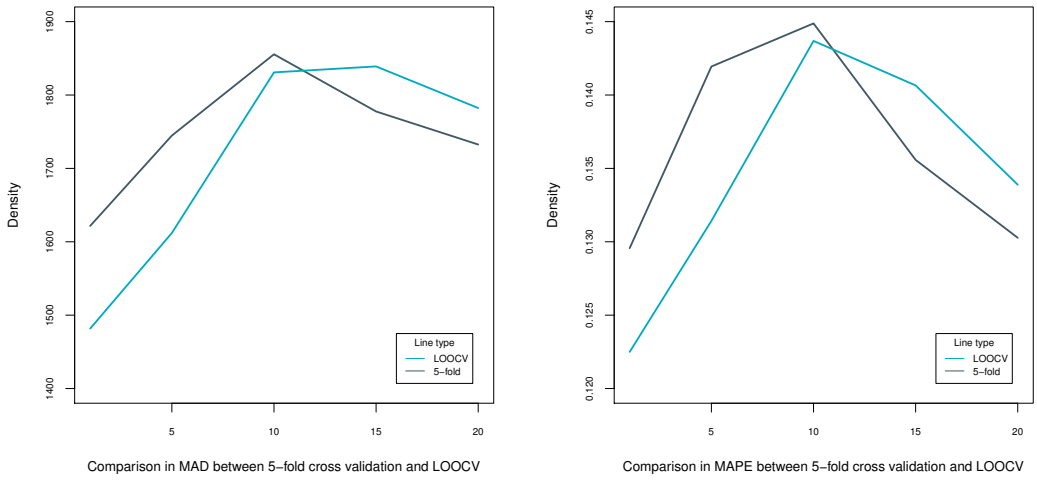
	k=1	k=5	k=10	k=15	k=20
First Fold	1402.5333	1233.9600	1171.6233	1025.3622	972.4233
Second Fold	1723.633	2127.553	2251.130	2101.793	2013.650
Third Fold	2295.300	1992.287	2199.720	2045.156	1957.682
Fourth Fold	1102.833	1424.707	1633.643	1581.029	1555.297
Fifth Fold	1583.795	1945.108	2021.513	2134.513	2163.600
Average	1621.619	1744.723	1855.526	1777.571	1732.530

Table 5: The MAPE of 5-fold cross validation Regression

	k=1	k=5	k=10	k=15	k=20
First Fold	0.11601656	0.10822631	0.10758310	0.09824654	0.09666564
Second Fold	0.1294907	0.1827948	0.1808431	0.1613188	0.1484787
Third Fold	0.1850656	0.1568351	0.1595583	0.1456691	0.1402084
Fourth Fold	0.1036103	0.1099345	0.1327426	0.1249256	0.1170987
Fifth Fold	0.1136571	0.1519185	0.1436685	0.1477322	0.1489077
Average	0.1295680	0.1419418	0.1448791	0.1355785	0.1302718

5 Conclusion

From the results it can be concluded that kNN performs well in both the tasks. The experiment was evaluated by using LOOCV and 5-fold cross validation for both the tasks. In the classification task, the highest predictive accuracy is 0.866 for LOOCV and 0.880 for 5-fold CV when k=1. In the numeric prediction task, the highest predictive accuracy is 0.866 for LOOCV and 0.880 for 5-fold CV when k=1. For numeric prediction both MAD and MAPE have been obtained. In case of LOOCV, the best result is obtained with the lowest value of MAD is 1481.723 and the lowest value of MAPE is 0.1224995. In case of 5-fold CV, the best result



(a) Comparison in MAD (b) Comparison in MAPE

Figure 2: Comparison in MAD and MAPE between 5-fold cross validation and LOOCV

is also obtained for k=1 with the lowest value of MAD is 1621.619 and the lowest value of MAPE is 0.1295680. We have compared the results for 5 k values. The work can be further extended to other k values.

References

[1] David W Aha, Dennis Kibler, and Marc K Albert. Instance-based learning algorithms. *Machine learning*, 6(1):37–66, 1991.

[2] Fabrizio Angiulli. Fast condensed nearest neighbor rule. In *Proceedings of the 22nd international conference on Machine learning*, pages 25–32. ACM, 2005.

[3] Kenneth G Budden. Radio waves in the ionosphere. *Radio Waves in the Ionosphere, by KG Budden, Cambridge, UK: Cambridge University Press, 2009*, 2009.

[4] Thomas Cover and Peter Hart. Nearest neighbor pattern classification. *IEEE transactions on information theory*, 13(1):21–27, 1967.

[5] Eibe Frank, Mark Hall, and Bernhard Pfahringer. Locally weighted naive bayes. In *Proceedings of the Nineteenth conference on Uncertainty in Artificial Intelligence*, pages 249–256. Morgan Kaufmann Publishers Inc., 2002.

[6] Wen-Jyi Hwang and Kuo-Wei Wen. Fast knn classification algorithm based on partial distance search. *Electronics letters*, 34(21):2062–2063, 1998.

[7] Liangxiao Jiang, Harry Zhang, and Zhihua Cai. Dynamic k-nearest-neighbor naive bayes with attribute weighted. *Fuzzy Systems and Knowledge Discovery*, pages 365–368, 2006.

[8] Liangxiao Jiang, Harry Zhang, and Jiang Su. Instance cloning local naive bayes. In *Conference of the Canadian Society for Computational Studies of Intelligence*, pages 280–291. Springer, 2005.

[9] Tom M Mitchell et al. Machine learning, 1997.

[10] Hui Wang. Nearest neighbours without k: A classification formalism based on probability. *Technical Report, Faculty of Informatics*, 2002.

[11] Ian H Witten, Eibe Frank, Leonard E Trigg, Mark A Hall, Geoffrey Holmes, and Sally Jo Cunningham. Weka: Practical machine learning tools and techniques with java implementations. 1999.

[12] Zhipeng Xie, Wynne Hsu, Zongtian Liu, and Mong Li Lee. Snnb: A selective neighborhood based naive bayes for lazy learning. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 104–114. Springer, 2002.