CSE5243 Lab2-KNN Classification

Part 1 Iris data analysis

1.1 Description:

The corroding code for iris data analysis is shown in Iris.R(in this code, I used Euclidean distance). The KNN classification based on Manhattan distance is shown in iris_manhattan.R

a. Duplicated data

In iris training data, there are some duplicated rows without considering duplicate data. However, I didn't remove them, because these even if these data are duplicated, they belong to different class. They are will provide basis for classify test data.

b. Normalization:

I used min_max normalization instead of z-score normalization to transform the train data and test data. Because this method is more robust to outlier comparing to z-score normalization

c. KNN classification:

In this part, I randomly choose k=5 to calculate the 5 nearest point of each test data. THE The euclidean distance is implemented in this part .The following code is KNN classification. The test data point is assigned the majority of class of its nearest point. For example, if the most of the nearest point belong to"Iris-setosa", then the class of the test data point is "Iris-setosa".

Figure1 Iris test data KNN classification

TransactionID	Actual_class	Predicted_cl	Posterior
1	Iris-setosa	Iris-setosa	1
2	Iris-setosa	Iris-setosa	1
3	Iris-setosa	Iris-setosa	1
4	Iris-setosa	Iris-setosa	1
5	Iris-setosa	Iris-setosa	1
6	Iris-setosa	Iris-setosa	1
7	Iris-setosa	Iris-setosa	1
8	Iris-setosa	Iris-setosa	1
9	Iris-setosa	Iris-setosa	1
10	Iris-setosa	Iris-setosa	1
11	Iris-setosa	Iris-setosa	1
12	Iris-setosa	Iris-setosa	1
13	Iris-setosa	Iris-setosa	1
14	Iris-setosa	Iris-setosa	1
15	Iris-setosa	Iris-setosa	1
16	Iris-setosa	Iris-setosa	1
17	Iris-setosa	Iris-setosa	1
18	Iris-setosa	Iris-setosa	1
19	Iris-setosa	Iris-setosa	1
20	Iris-setosa	Iris-setosa	1
21	Iris-versicolor	Iris-versicolc	1
22	Iris-versicolor	Iris-versicolc	0.8
23	Iris-versicolor	Iris-versicolc	1
24	Iris-versicolor	Iris-versicolc	1
25	Iris-versicolor	Iris-versicolc	1
26	Iris-versicolor	Iris-versicolc	0.8
27	Iris-versicolor	Iris-virginica	0.8

1.2 Confusion Matrix (k=5)

1.2.1 Based on Euclidean distance

	Predicated class			
Actual class		Iris-setosa	Iris-versicolor	Iris-virginica
	Iris-setosa	20	0	0
	Iris-versicolor	0	17	13
	Iris-virginica	0	0	20

Error rate=13/70

1.2.1 based on Manhattan distance

	Predicated class			
Actual class		Iris-setosa	Iris-versicolor	Iris-virginica
	Iris-setosa	20	0	0
	Iris-setosa	0	18	12
	Iris-virginica	0	0	20

Error rate=12/70

Compare these two confusion matrix, we can know that the error rate based on different distance are same .

1.3. Compare confusion matrix and error rate based on various k-value

1.3.1 Euclidean distance

K=1

	Predicated class			
Actual class		Iris-setosa	Iris-versicolor	Iris-virginica
	Iris-setosa	20	0	0
	Iris-setosa	0	21	9
	Iris-virginica	0	0	20

Error rate=9/70=12.8%

K=2

	Predicated class		
Actual class		Iris-setosa	Iris-versicolor	Iris-virginica
	Iris-setosa	20	0	0
	Iris-setosa	0	22	8
	Iris-virginica	0	0	20

Error rate=8/70

k=3

	Predicated class			
Actual class		Iris-setosa	Iris-versicolor	Iris-virginica
	Iris-setosa	20	0	0
	Iris-setosa	0	17	13
	Iris-virginica	0	0	20

Error rate=13/70

K=4

	Predicated class			
Actual class		Iris-setosa	Iris-versicolor	Iris-virginica
	Iris-setosa	20	0	0
	Iris-versicolor	0	18	12
	Iris-virginica	0	0	20

Error rate=12/70

K=5

	Predicated class			
Actual class		Iris-setosa	Iris-versicolor	Iris-virginica
	Iris-setosa	20	0	0
	Iris-versicolor	0	17	13
	Iris-virginica	0	0	20

Error rate=13/70

k=6

	Predicated class			
Actual class		Iris-setosa	Iris-versicolor	Iris-virginica
	Iris-setosa	20	0	0
	Iris-versicolor	0	18	12
	Iris-virginica	0	0	20

Error rate=12/70

k=7

	Predicated class			
Actual class		Iris-setosa	Iris-versicolor	Iris-virginica
	Iris-setosa	20	0	0
	Iris-versicolor	0	18	12
	Iris-virginica	0	0	20

Error rate=12/70

k=8

	Predicated class			
Actual class		Iris-setosa	Iris-versicolor	Iris-virginica
	Iris-setosa	20	0	0
	Iris-versicolor	0	18	12
	Iris-virginica	0	0	20

Error rate=12/70

k=9

	Predicated class			
Actual class		Iris-setosa	Iris-versicolor	Iris-virginica
	Iris-setosa	20	0	0
	Iris-versicolor	0	18	12
	Iris-virginica	0	0	20

Error rate=12/70

K=10

	Predicated class			
Actual class		Iris-setosa	Iris-versicolor	Iris-virginica
	Iris-setosa	20	0	0
	Iris-setosa	0	18	12
	Iris-virginica	0	0	20

Error rate=1-(20+18+20)/70=17.14%

1.3.2 Manhattan Distance

K=1

	Predicated class			
Actual class		Iris-setosa	Iris-versicolor	Iris-virginica
	Iris-setosa	20	0	0
	Iris-setosa	0	18	12
	Iris-virginica	0	0	20

Error rate=12/70

K=2

	Predicated class			
Actual class		Iris-setosa	Iris-versicolor	Iris-virginica
	Iris-setosa	20	0	0
	Iris-setosa	0	22	8
	Iris-virginica	0	0	20

Error rate=12/70

k=3

	Predicated class			
Actual class		Iris-setosa	Iris-versicolor	Iris-virginica
	Iris-setosa	20	0	0
	Iris-setosa	0	18	12
	Iris-virginica	0	0	20

Error rate=13/70

K=4

	Predicated class			
Actual class		Iris-setosa	Iris-versicolor	Iris-virginica
	Iris-setosa	20	0	0
	Iris-versicolor	0	19	11
	Iris-virginica	0	0	20

Error rate=12/70

K=5

	Predicated class			
Actual class		Iris-setosa	Iris-versicolor	Iris-virginica
	Iris-setosa	20	0	0
	Iris-versicolor	0	18	12
	Iris-virginica	0	0	20

Error rate=13/70

K=6

	Predicated class			
Actual class		Iris-setosa	Iris-versicolor	Iris-virginica
	Iris-setosa	20	0	0
	Iris-versicolor	0	18	12
	Iris-virginica	0	0	20

Error rate=13/70

k=7

	Predicated class			
Actual class		Iris-setosa	Iris-versicolor	Iris-virginica
	Iris-setosa	20	0	0
	Iris-versicolor	0	18	12
	Iris-virginica	0	0	20

Error rate=13/70

k=8

	Predicated class			
Actual class		Iris-setosa	Iris-versicolor	Iris-virginica
	Iris-setosa	20	0	0
	Iris-versicolor	0	18	12
	Iris-virginica	0	0	20

Error rate=12/70

k=9

	Predicated class			
Actual class		Iris-setosa	Iris-versicolor	Iris-virginica
	Iris-setosa	20	0	0
	Iris-versicolor	0	17	13
	Iris-virginica	0	0	20

Error rate=12/70

K=10

	Predicated class			
Actual class		Iris-setosa	Iris-versicolor	Iris-virginica
	Iris-setosa	20	0	0
	Iris-setosa	0	18	12
	Iris-virginica	0	0	20

Error rate=1-(20+18+20)/70=17.14%

Conclusion:

By comparison, we can know that, no matter based on which distance method ,when k =2, the accuracy of predication is higher than others. Based on above different k values, I recommend k=2, which has the smallest error rate. However, in order to get more convincing result, more different k value need be tested.

Part2 Income data analysis

2.1 Description

The corroding code for income data analysis based on euclidean distance is shown in

Iris_eucli.R. The KNN classification based on Manhattan distance is shown in income_manhattan.R, and the KNN classification based on reduced training data is shown in income_reduced_train.R.

a. Missing data

Through observing the income data, there are some missing data. I decided to remove the missing data object, which is convenient for calculate the distance and KNN classification.

b. Max-min normalization.

Like iris data, I still choose max-min normalization to normalize numeric attribute in income train and test data, to reduce the effect from outlier. Z-score normalization is susceptible to outlier, because the mean is calculated from all data (including outlier)

c. Distance

When calculating the distance between teat data and training data, I used Euclidean distance or Manhattan distance for numerical attribute. For categorical attribute, if the value is different, the distance is 1.

d. KNN classification

Like iris data, I randomly choose k=5 to calculate the 5 nearest point of each test data. The test data point is assigned the majority of class of its nearest point.

Figure2 Income data KNN classification

	TransactionID	Actual_class	Predicted_class	Posterior_Probability
1	1	<=50K	<=50K	1.0
2	2	<=50K	<=50K	0.8
3	3	>50K	<=50K	1.0
4	4	<=50K	>50K	0.4
5	5	<=50K	<=50K	1.0
6	6	<=50K	<=50K	1.0
7	7	<=50K	<=50K	1.0
8	8	<=50K	<=50K	0.6
9	9	>50K	<=50K	0.8
10	10	>50K	>50K	0.0
11	11	<=50K	<=50K	1.0
12	12	>50K	<=50K	0.6

2.2 Confusion matrix K=5

2.2.1 Based on Euclidean distance

Predicted class

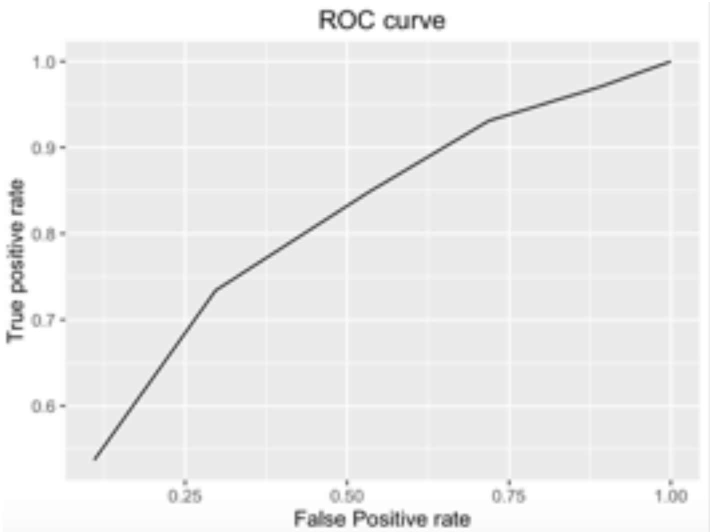
	<=50K	>50K
<=50K	172	31
>50K	34	30

Actual class

Error rate =(31+34)/267=24.34%

TPR	TNR	FPR	FNR	Recall_	Precision	F_measure
0.8472906	0.46875	0.53125	0.1527094	0.8472906	0.8349515	0.8410758

ROC curve:



In this part, I set posterior probability (p_threshold=0.2, 0.4, 0.6, 0.8) several threshold. For example, if p_threshold=0.8 and posterior probability>=0.8, the predicted class is“<=50”,else are“>50”.Base on the classification of p_threshold=0.8, we can calculate corresponding True positive rate and False positive rate. According to each pair of True positive rate and False positive rate on different p__threshold, we can obtain the following ROC curve plot.

According to the curve plot, the curve is above diagonal line. So KNN Classification based on k=5 is better than random model.

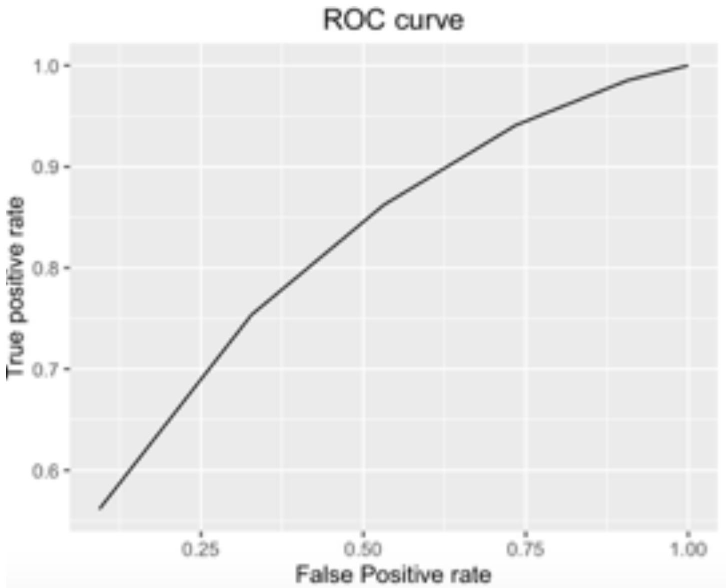
2.2.1 Based on Manhattan distance

Predicted class

		<=50K	>50K
	<=50K	175	28
Actual class	>50K	34	30

Error rate= (28+34)/267=23.22%

TPR	TNR	FPR	FNR	Recall_	Precision	F_measure
0.862069	0.46875	0.53125	0.137931	0.862069	0.8373206	0.8495146



By comparing the confusion matrix of two different distance calculation method, they are slightly different. Compared to Euclidean distance, the KNN classifier based on Manhattan distance has smaller error rate

2.3 Different k values

2.3.1 Based on Euclidean Distance

K=1

Predicted Class

Actual class	<=50K	>50K
<=50K	177	26
>50K	33	31

Error rate= (33+26)/267=22.09%

k=2

Predicted Class

Actual class	<=50K	>50K
<=50K	188	15
>50K	45	19

Error rate= (45+15)/267=22.47%

k=3

Predicted Class

Actual class	<=50K	>50K
<=50K	174	29
>50K	33	31

Error rate=(33+29)/267=23.22%

k=4

Predicted Class

Actual class	<=50K	>50K
<=50K	183	20
>50K	39	25

Error rate=(39+20)/267=22.09%

k=5

		Predicted class	
		<=50K	>50K
Actual class	<=50K	172	31
	>50K	34	30

Error rate =(31+34)/267=24.34%

k=6

		Predicted class	
		<=50K	>50K
Actual class	<=50K	180	23
	>50K	40	24

Error rate =(40+23)/267=23.59%

k=7

		Predicted class	
		<=50K	>50K
Actual class	<=50K	175	28
	>50K	34	30

Error rate =(34+28)/267=23.22%

k=8

		Predicted class	
		<=50K	>50K
Actual class	<=50K	179	24
	>50K	38	26

Error rate =(38+24)/267=23.22%

k=9

		Predicted class	
		<=50K	>50K
Actual class	<=50K	173	30
	>50K	32	32

Error rate =23.22%

K=10

		Predicted Class	
		<=50K	>50K
Actual class	<=50K	178	25
	>50K	34	30

Error rate= (34+25)/267=22.09%

k=15

Predicted Class

Actual class	<=50K	>50K
<=50K	177	26
>50K	29	35

Error rate= 20.59%

k=20

Predicted Class

Actual class	<=50K	>50K
<=50K	176	27
>50K	30	34

Error rate=21.34%

Conclusion:

By comparison, we can know that, different k value result in different confusion matrix. But the error rate of different K-value are slightly various. According to above test, I recommend k=15,since the error rate is lower that others. However, in order to get more convincing result , more different k value need be tested.

2.3.1 Based on Manhattan distance

k=1

Predicted Class

Actual class	<=50K	>50K
<=50K	183	20
>50K	33	31

Error rate=(20+33)/267=19.85%

k=2

Predicted class

Actual class	<=50K	>50K
<=50K	192	11
>50K	49	15

Error rate=60/276=22.47%

k=3

Predicted class

Actual class	<=50K	>50K
<=50K	177	26
>50K	35	29

Error rate=61/276=22.84%

k=4

Predicted class

Actual class	<=50K	>50K
<=50K	185	18
>50K	42	22

Error rate=60/276=22.47%

k=5

Predicted Class

Actual class	<=50K	>50K
<=50K	184	19
>50K	47	17

Error rate=(47+19)/267=24.7%

k=6

Predicted Class

Actual class	<=50K	>50K
<=50K	182	21
>50K	42	22

Error rate=(42+21)/267=23.59%

K=7

Predicted class

Actual class	<=50K	>50K
<=50K	173	30
>50K	33	31

Error rate=(33+30)/267=23.59%

k=8

Predicted class

Actual class	<=50K	>50K
<=50K	182	21
>50K	38	26

Error rate=(38+21)/267=22.09%

k=9

Predicted class

Actual class	<=50K	>50K
<=50K	174	29
>50K	32	32

Error rate=(32+29)/267=22.84%

k=10

Predicted class

Actual class	<=50K	>50K
<=50K	181	22
>50K	33	31

Error rate=(32+22)/267=20.59%

k=15

Predicted class

Actual class	<=50K	>50K
<=50K	178	25
>50K	28	36

Error rate=(28+25)/267=19.85%

k=20

Predicted class

Actual class	<=50K	>50K
<=50K	178	25
>50K	32	32

Error rate=(32+25)/267=21.34%

Conclusion:

By comparison, we can know that, when k=1 and k=15,the error rate is less than other. I recommend k=15 or k=1. However, in order to get more convincing result , more different k value need be tested.

2.4 Reduced training data set

The training data set is reduced from 487 into 244(k=5), which is half of the original training data set. From following confusion matrix, we can know that it changed slightly. Compare the error rate with original training data set; the error rate is almost same. So, in this case (k=5), reducing the training data set won't increase accuracy. We can try to set different k value to see affect from reducing training data set.

Predicted Class

Actual class	<=50K	>50K
<=50K	180	23
>50K	44	20

Error rate= (44+23)/267=26.2%