Tutorial 3

DSA1101
Introduction to Data Science

September 14, 2018

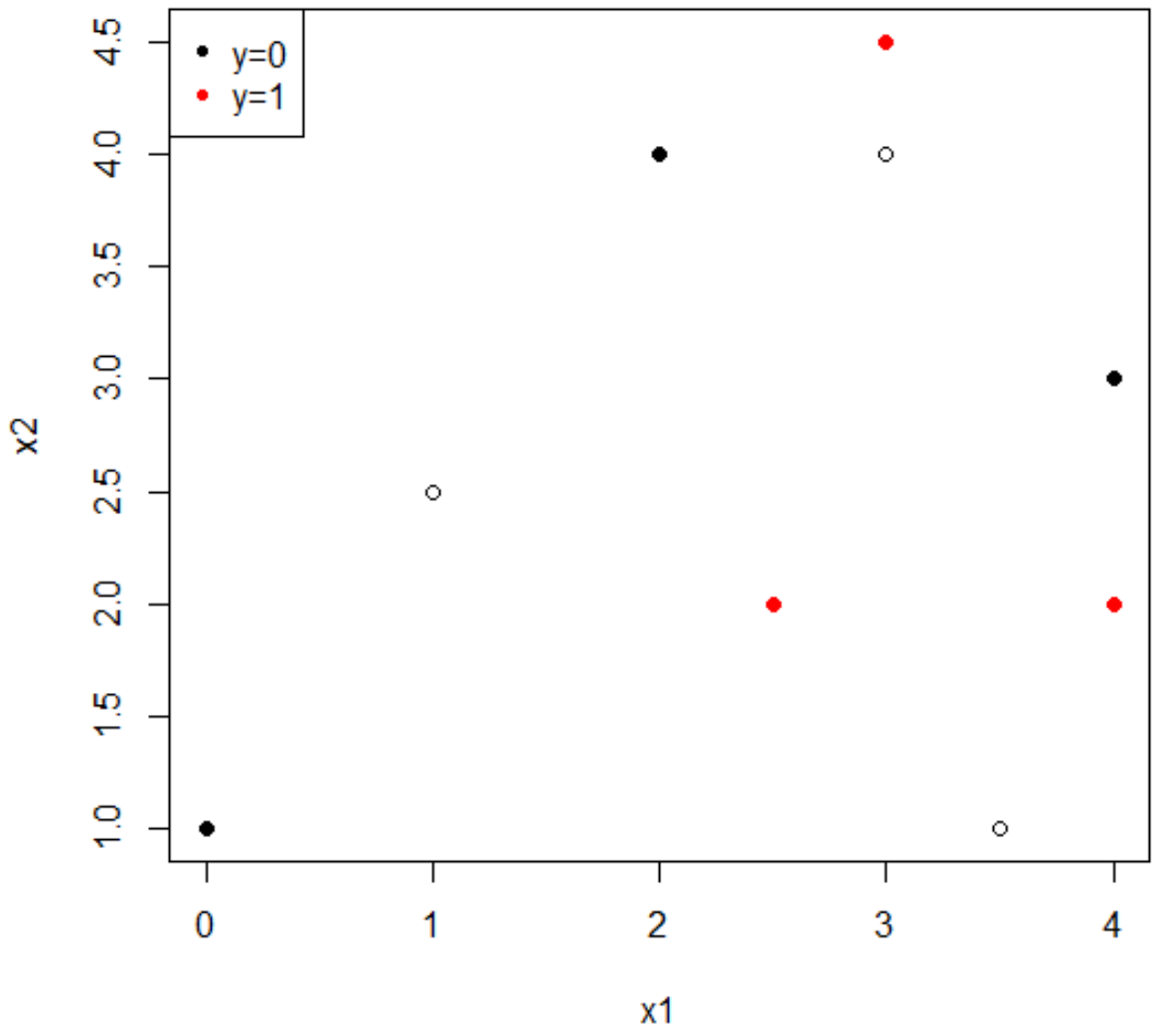
Exercise 1. *The k-nearest neighbor classifier*

Suppose we have a training set of six data points in two features $x(1) = c(0, 2, 2.5, 3, 4, 4)$ and $x(2) = c(1, 4, 2, 4.5, 2, 3)$, as well as the corresponding binary label values $y = c(0, 0, 1, 1, 1, 0)$

(a) Compute the k-nearest neighbor fit $\hat{y}(x)$ for the following test objects based on the two features $x = [x(1), x(2)]$ and $k = 3$. Here $\hat{y}(x) = \frac{1}{k}\sum_{x_i \in N_k(x)} y_i$ where $N_k(x)$ is the neighborhood of x defined by the k closest points x_i (in terms of Euclidean distance) in the training set.

test object	x_1	x_2
A	1	2.5
B	3	4
C	3.5	1

$$\hat{y}_A = \frac{0+0+1}{3} = \frac{1}{3}$$
$$\hat{y}_B = \frac{0+0+1}{3} = \frac{1}{3}$$
$$\hat{y}_C = \frac{1+1+0}{3} = \frac{2}{3}$$


y=0
y=1
x2
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
0
1
2
3
4
x1

(b) Predict the label values of y for the test objects A,B and C, based on the majority rule.

$$\hat{y}_A = \frac{1}{3} < \frac{1}{2} \rightarrow \text{ predicted } y \text{ for A=0}$$

$$\hat{y}_B = \frac{1}{3} < \frac{1}{2} \rightarrow \text{ predicted } y \text{ for B=0}$$

$$\hat{y}_C = \frac{2}{3} > \frac{1}{2} \rightarrow \text{ predicted } y \text{ for C=1}$$

(b) Suppose the actual label values for the test objects are as follows: $y_A = 0$, $y_B = 1$ and $y_C = 0$. Compute the *accuracy*, *true positive rate*, *false positive rate*, *false negative rate* and *precision* of the classifier based on the actual and predicted values of y for the three test objects. The definitions are as follows:

		Predicted y	
		1	0
Actual y	1	True Positives (TP)	False Negatives (FN)
	0	False Positives (FP)	True Negatives (TN)

answer:

		Predicted y	
		1	0
Actual y	1	0	1
	0	1	1

$$accuracy = \frac{0+1}{0+1+1+1} = \frac{1}{3}$$

$$TPR = \frac{0}{0+1} = 0$$
$$FPR = \frac{1}{1+1} = \frac{1}{2}$$
$$FNR = \frac{1}{0+1} = 1$$
$$Precision = \frac{0}{0+1} = 0$$

Exercise 2. Loan managers often need to take into account an applicant's demographic and socio-economic profiles in deciding whether to approve a loan to the applicant, to minimize losses due to defaults. In this exercise we will build and evaluate a classifier based on the German Credit Data to predict whether an applicant is considered as having good or bad credit risk. The features or predictors include (1) loan duration (in months), (2) credit amount, (3) Installment rate in percentage of disposable income and (4) age in years.

(a) Read and explore the data from the file `German_credit.csv`

```
banktrain <- read.table("German_credit.csv",header=TRUE,sep=",")
summary(banktrain)
dim(banktrain)
```

(b) Randomly select 500 customer records to form the training data, and the remaining 500 records will be the test data

```
set.seed(1)
train =sample(1:1000, 500);
banktrain[,2:5] <- lapply(banktrain[,2:5], scale)
train.data = banktrain[train,]
test.data  = banktrain[-train ,]

train.x = train.data[,2:5]
test.x = test.data[,2:5]
train.y = train.data[,1]
test.y = test.data[,1]
```

(c) Use k-nearest neighbor classifier with $k = 1$ to predict if a loan applicant is credible, and compute the *accuracy* of the classifier.

```
library(class)

knn.pred = knn(train.x,test.x,train.y,k=1)
confusion.matrix=table(knn.pred, test.y)
confusion.matrix
```

(d) Use k-nearest neighbor classifier with $k = 5$ to predict if a loan applicant is credible, and compute the *accuracy* of the classifier.

```
knn.pred = knn(train.x,test.x,train.y,k=5)
confusion.matrix=table(knn.pred, test.y)
confusion.matrix
```

(e) Use k-nearest neighbor classifier with $k = 10$ to predict if a loan applicant is credible, and compute the *accuracy* of the classifier.

```
knn.pred = knn(train.x,test.x,train.y,k=10)
confusion.matrix=table(knn.pred, test.y)
confusion.matrix
```