CS 5525
Solutions to Homework Assignment 5

Meghendra Singh

December 2, 2016

[10] 1.

Given;

$$P(Undergraduatestudent) = P(U) = \frac{4}{5} = 0.8$$

$$P(Graduatestudent) = P(G) = \frac{1}{5} = 0.2$$

$$P(Smokes|Undergraduatestudent) = P(S|U) = 0.15$$

$$P(Smokes|Graduatestudent) = P(S|G) = 0.23$$

(a) The probability that a student who smokes is a graduate student is computed as:

$$\begin{aligned} P(Graduatestudent|Smokes) &= P(G|S) \\ &= \frac{P(S|G)P(G)}{P(S)} \\ &= \frac{0.23 * 0.2}{0.15 * 0.8 + 0.23 * 0.2} \\ &= \frac{0.046}{0.166} \end{aligned}$$

$$P(G|S) = 0.277 \tag{1}$$

(b) A randomly chosen smoker is more likely to be an **undergraduate student**, since the probability of a student who smokes being an undergraduate student is more than that of a graduate student. These probabilities can be computed as follows:

$$\begin{aligned} P(Undergraduatestudent|Smokes) &= P(U|S) \\ &= \frac{P(S|U)P(U)}{P(S)} \\ &= \frac{0.15 * 0.8}{0.15 * 0.8 + 0.23 * 0.2} \\ &= \frac{0.12}{0.166} \end{aligned}$$

$$P(U|S) = 0.722 \tag{2}$$

From (1) and (2), $P(U|S) > P(G|S)$
Hence, a randomly chosen smoker is more likely to be an **undergraduate student**.

(c) A student who smokes and lives in the dorm, is more likely to be a **graduate student**. Assuming independence between students who live in a dorm and those who smoke, this can be concluded based on the following: Given,

$$\begin{aligned} P(Livesindorm|Graduatestudent) &= P(D|G) \\ &= 0.3 \end{aligned}$$

$$\begin{aligned} P(Livesindorm|Undergraduatestudent) &= P(D|U) \\ &= 0.1 \end{aligned}$$

$$\begin{aligned} P(Graduatestudent|Smokes, Livesindorm) &= P(G|S, D) \\ &= \frac{P(G)P(S, D|G)}{P(S, D)} \\ &= \frac{0.2 * 0.3 * 0.23}{P(S, D)} \end{aligned}$$

$$P(G|S, D) = \frac{0.0138}{P(S, D)} \tag{3}$$

$$\begin{aligned} P(Undergraduatestudent|Smokes, Livesindorm) &= P(U|S, D) \\ &= \frac{P(U)P(S, D|U)}{P(S, D)} \\ &= \frac{0.8 * 0.1 * 0.15}{P(S, D)} \end{aligned}$$

$$P(U|S, D) = \frac{0.012}{P(S, D)} \tag{4}$$

Ignoring the common denominator in (3) and (4), we have $P(G|S, D) > P(U|S, D)$. Hence, if a student smokes and lives in the dorm, he or she is more likely to be a **graduate student**.

---

[25] 2.

PART I.
Given one-dimensional data points: {0.1, 0.2, 0.4, 0.5, 0.6, 0.8, 0.9}

Data Point	Cluster Assignment
0.1	A
0.2	A
0.4	B
0.5	B
0.6	B
0.8	C
0.9	C

(a) Upon applying k-means clustering to obtain three clusters, A, B, and C for initial centroids at $0, 0.25, 0.6$, for A,B and C respectively, we get the following cluster assignments after 3 iterations: The location of centroids after the first three iterations is $\{0.15, 0.5, 0.85\}$ for clusters A, B and C, respectively.
The SSE of the k-means solution after 3 iterations is given by:

$$\begin{aligned} SSE &= \sum_{i=1}^{k} \sum_{x \in C_i} dist^2(m_i, x) \\ &= 0.0025 + 0.0025 + 0.01 + 0.0 + 0.01 + 0.0025 + 0.0025 \\ &= 0.03 \end{aligned}$$

(b) Upon applying k-means to the data points with k=2 using initial centroids located at {0.1,0.9} we get the following cluster assignments: K-means converges after 1 iteration and

Data Point	Cluster Assignment
0.1	A
0.2	A
0.4	A
0.5	A
0.6	B
0.8	B
0.9	B

the location of centroids are $\{0.3, 0.766\}$ for clusters A and B respectively and the SSE for the two clusters are $\{0.1, 0.0505\}$ respectively. Cluster A has to be split into two clusters A and C as it has the higher SSE, we get the following cluster assignments after applying k-means to split cluster A: Once again, k-means converges after 1 iteration and we get

Data Point	Cluster Assignment
0.1	A
0.2	A
0.4	C
0.5	C

the location of centroids as $\{0.15, 0.45\}$. The final clustering solution after the applying bisecting k-means with k=3 is as follows:

Data Point	Cluster Assignment
0.1	A
0.2	A
0.4	C
0.5	C
0.6	B
0.8	B
0.9	B

PART II.

(a) A total of 17 instances were clustered incorrectly, below is the confusion matrix:

	Cluster Assignments			
Actual Class		Cluster 0	Cluster 1	Cluster 2
	Iris-setosa	0	50	0
	Iris-versicolor	47	0	3
	Iris-virginica	14	0	36

(b) Cluster 2 has a total of 39 instances out of these, 36 belong to the class **Iris-virginica** and 3 belong to the class **Iris-versicolor**. Hence, 3 instances of class **Iris-versicolor** were incorrectly clustered into cluster 2 and these should belong to cluster 0, which has the majority of instances (47 instances) belonging to class **Iris-versicolor**.

(c) **Iris-setosa** has all instances (50 instances) clustered correctly into cluster 1.

---

[15] 3.

(a) *zoo.arff* has 18 attributes out of which only the **legs** attribute is numerical. I used the following *weka.filters.unsupervised.attribute.Discretize* filter to discretize it into 6 bins:

Bin No.	Interval	Count of instances in bin
1	(− inf to 1.333333]	23
2	(1.333333 to 2.666667]	27
3	(2.666667 to 4]	38
4	(4 to 5.333333]	1
5	(5.333333 to 6.666667]	10
6	(6.666667 to inf)	2

(b) The following are 4 interesting rules:

Rule	**Confidence**	**Lift**
venomous=false and tail=true 71 $\Longrightarrow$ backbone=true 71	1.0	1.22
tail=true 75 $\Longrightarrow$ backbone=true 74	0.99	1.2
backbone=true and tail=true 74 $\Longrightarrow$ venomous=false 71	0.96	1.04
backbone=true 83 $\Longrightarrow$ venomous=false 79	0.95	1.03

(c) The venomous=false and tail=true 71 $\Longrightarrow$ backbone=true 71 rule would always be true according to the *Apriori algorithm*, as the confidence for this rule is **1.0**. This implies that for all item-sets where *venomous* attribute is $false$ and *tail* attribute is *true* (71 item-sets) the *backbone* attribute is always *true*.

---

[10] 4.

(a) Nearest-neighbor based (or Distance based) anomaly detection has the following problems:

(i) **Sensitive to the value of k (Low value of k):** For a low value of k, nearest-neighbour based anomaly detection fails to detect a group of closely spaced outliers, if the number of outliers in this group is greater than k. For E.g. if there are two closely spaced outliers for *k=1*, the distance to 1-NN for both the outliers would be very small (both of them will be nearest neighbours of each other), hence these will not be detected, while normal points with 1-NN farther than the distance between the two outliers, will be detected as outliers.

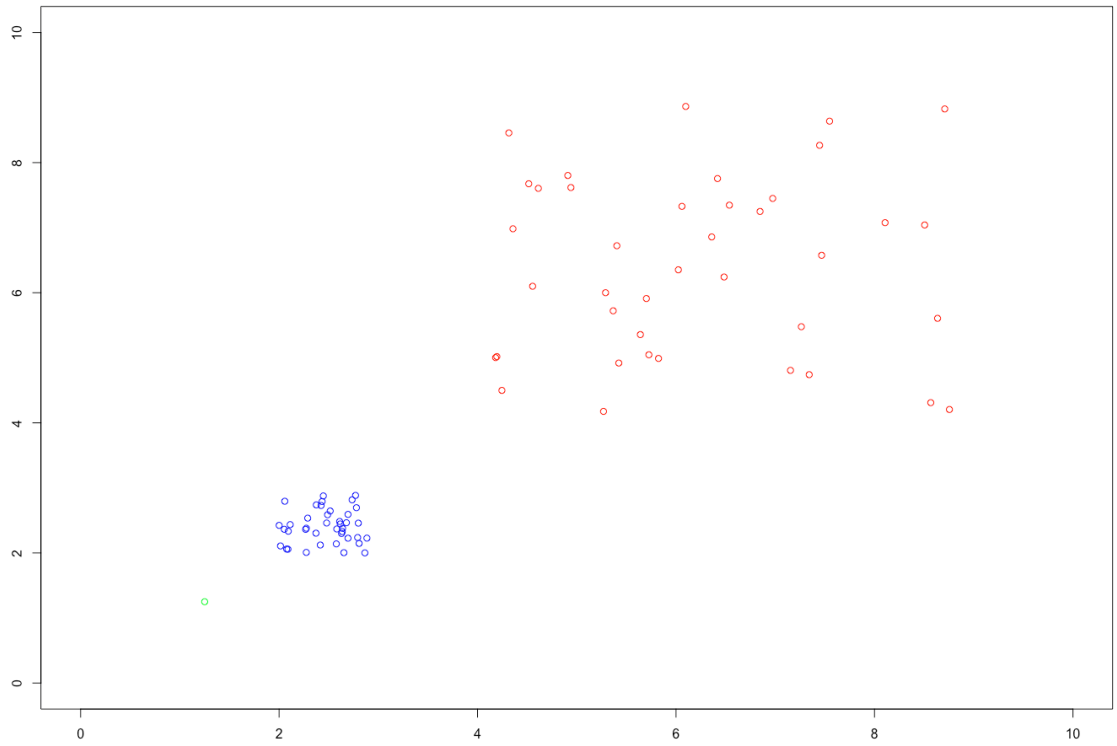
(ii) **Sensitive to the value of k (High value of k):** Conversely, for a high value of k, nearest-neighbour based anomaly detection will also detect any small cluster with number of members less than or equal to k as outliers. For E.g. if there is one small cluster with 5 members (data points) and one more data point at a great distance from this small cluster, for *k=5* the distance to the 5^{th} nearest neighbor for each of the points in the small cluster would be very large, which would cause all of the members of the small cluster to be detected as outliers.

(iii) **Sensitive to the density of clusters:** Nearest neighbour based anomaly detection fails to detect outliers near high density regions (clusters with large number of data points in close proximity). Whereas, normal points belonging to low density regions (clusters with small number of data points spread out in space) are detected as outliers. In both of these case the distance to the k^{th} nearest neighbor is not a meaningful metric for anomaly detection, because an outlier for a low density region is different from an outlier for a high density region.

(iv) The concept of distance itself becomes less meaningful in **high-dimensional space**.

(v) It is **computationally expensive**, i.e. the algorithms execution time complexity is: $O(n^2)$, which makes it infeasible to execute for very-large inputs.

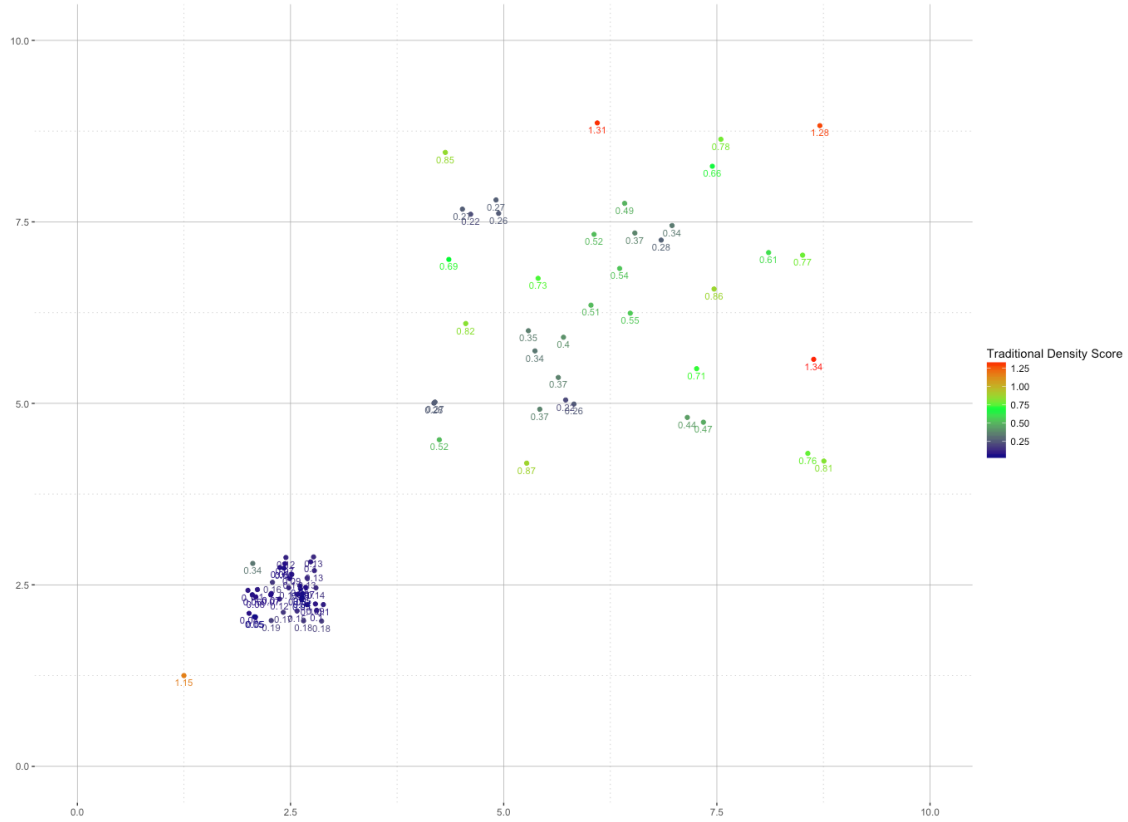
(b) An anomaly or outlier for a low density region is different from an outlier for a high density region. Therefore, a point that is located at a smaller distance from a high density region should be considered as an outlier, while a point located at a similar distance from a low density region might not be considered as an outlier. This implies that the concept of an outlier is relative to the density of other points in a region. If we use the traditional density based approach for anomaly detection, we assume that the density of data points is uniform across the space being considered, which might not be true. Consider the following example:



Here, the green colored data point is an anomaly, and the red and blue colored data points belong to two cluster of data points with low and high densities respectively. When we apply the traditional density based approach of assigning a score to each point using the following scoring function:

$$density(x,k) = \left(\frac{\sum_{y \in N(x,k)} distance(x,y)}{|N(x,k)|} \right)^{-1}$$

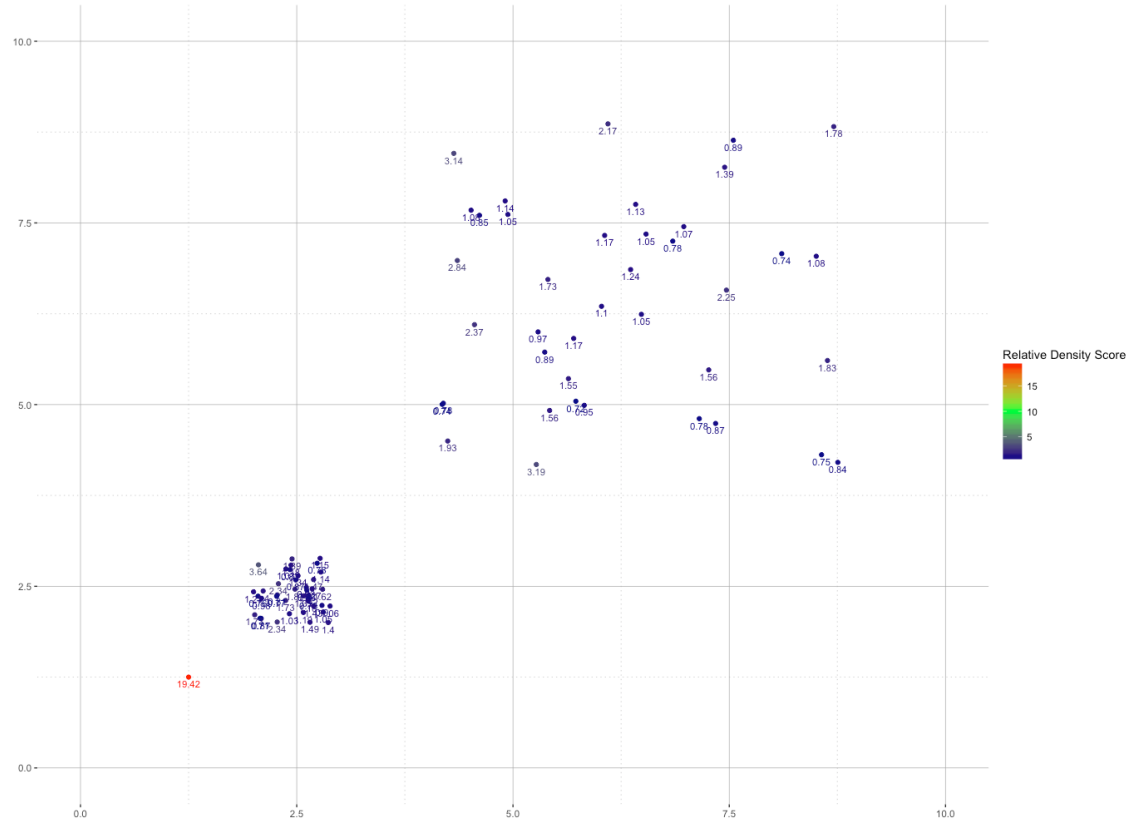
for $k = 2$ we get the following result:



We observe that the anomalous data point is given a lower score (1.15) as compared to three normal points of the low density cluster (1.31, 1.28 and 1.34). So if we were to remove the top 3 highest scoring data points as anomalous, we would end up removing these normal points from our data, while the actual anomaly will not be removed. Now, if we apply the average relative density based approach (LOF approach) for calculating the score of the data points to detect anomalies, using $k = 2$ and the following scoring function:

$$average\_relative\_density(x,k) = \left[\frac{density(x,k)}{\left(\frac{\sum_{y \in N(x,k)} distance(y,k)}{|N(x,k)|} \right)} \right]$$

We get the following result:



We observe that the anomalous data point is given the highest score of 19.42. So if we were to remove the top n highest scoring data points, the anomalous data point would be removed first.

---

[15] 5.

After round 1, we get the following values for ϵ_1 and α_1 for the three weak classifiers $\{H1, H2, H3\}$, using initial uniform weight of 0.1 for each instance:

Weak Classifier	ϵ_1	α_1
H1	$0.1 * 2 = 0.2$	$\frac{1}{2} * \ln(\frac{1-0.2}{0.2}) = 0.693$
H2	$0.1 * 4 = 0.4$	$\frac{1}{2} * \ln(\frac{1-0.4}{0.4}) = 0.202$
H3	$0.1 * 1 = 0.1$	$\frac{1}{2} * \ln(\frac{1-0.1}{0.1}) = 1.09$

Using these values of α_i we can compute the weights ($D(i)$'s) for all the instance as:

$$D_{t+1}(i) = \frac{D_t(i) * \exp(-\alpha_t y_i h_t(x_i))}{Z_t}$$

For this problem we are concerned with weights after one round without normalization, therefore;

$$D_1(i) = D_0(i) \exp(-\alpha_0 y_i h_0(x_i))$$

Here $D_0(i)$ is the initial uniform weight of 0.1 assigned to each instance, and $D_1(i)$ is the weight after 1 round of Boosting. The following table gives the weights for each of the instances after the first round of boosting for the three classifiers:

		H1		**H2**		**H3**	
X	**Y**	H1(X)	$D_1(X)$	H2(X)	$D_1(X)$	H3(X)	$D_1(X)$
0.1	1	1	0.050	-1	0.122	1	0.033
0.2	1	1	0.050	-1	0.122	1	0.033
0.3	1	1	0.050	-1	0.122	1	0.033
0.4	-1	-1	0.050	-1	0.081	-1	0.033
0.5	-1	-1	0.050	-1	0.081	-1	0.033
0.6	-1	-1	0.050	-1	0.081	-1	0.033
0.7	-1	-1	0.050	-1	0.081	-1	0.033
0.8	-1	-1	0.050	1	0.122	-1	0.033
0.9	1	-1	0.199	1	0.081	-1	0.29
1	1	-1	0.199	1	0.081	1	0.033

---

[25] 6.

(a,c) The following table gives the fitted linear model at each step of backward elimination method for feature subset selection along with the AIC values:

Step	**Fitted Model**	**AIC Value**
1	Y = -4.562370 + 1.879336 * X1 + 0.003068 * X2 - 0.000258 * X3 + 0.311844 * X4 + 0.003646 * X5 + 0.210679 * X6 - 0.009511 * X7	-2922
2	Y = -4.549609 + 1.879336 * X1 + 0.003068 * X2 - 0.000258 * X3 + 0.311844 * X4 + 0.210679 * X6 - 0.009511 * X7	-2923
3	Y = -4.05502 + 1.65139 * X1 + 0.00256 * X2 + 0.29967 * X4 + 0.21068 * X6 - -0.009511 * X7	-2924

Using 5-fold cross validation on the model obtained after step 3, which considers features **X1**, **X2**, **X4**, **X6** and **X7** for predicting **Y**, and after converting the predicted values into binary using a threshold of **0.5**, we get the following confusion matrix and accuracy values:

	Predicted Class		
Actual Class		0	1
	0	365	0
	1	19	384
Accuracy	0.975		

(b,c) The fitted logistic regression model is:

$$\log \frac{Y}{1-Y} = 3242.438 - 1922.991X1 - 3.150X2 + 1.580X3 - 25.747X4 + 0.205X5 + 13.490X6 - 0.510X7$$

We get the following confusion matrix and accuracy values using 5-fold cross validation:

	Predicted Class		
Actual Class		0	1
	0	362	3
	1	19	384
Accuracy	0.971		

Have implemented the regression models in R, the source code is as follows:

```
library(xlsx)
library(DAAG)
library(boot)
library(caret)

#Read the data file into R
setwd("~/DAFall16/HW5/")
enb2012 <- read.xlsx("ENB2012_data.xlsx",sheetIndex = 1)
enb2012_data <- enb2012[1:768,]

#Part (a)
#Make a linear regression model object
linearModel <- lm(Y~X1+X2+X3+X4+X5+X6+X7, data=enb2012_data)

#Perform backward elimination using AIC
selectedVars <- step(linearModel, direction = "backward",trace = 1)
```

```
#Print coefficients for each step of elimination
#based on output of backward elimination
lm(Y~X1+X2+X3+X4+X5+X6+X7, data=enb2012_data)
lm(Y~X1+X2+X3+X4+X6+X7, data=enb2012_data)
lm(Y~X1+X2+X4+X6+X7, data=enb2012_data)

#Make predictions the model using 5-fold cross validation
fitLinearRegression <- cv.lm(data = enb2012_data, selectedVars, m = 5)

#Convert model output to binary for creating confusion matrix
#and computing accuracy.
fitLinearPredicted <- fitLinearRegression$cvpred > 0.5

#Part (b)
#Make a logistic regression model object
logisticModel <- glm(Y~X1+X2+X3+X4+X5+X6+X7, data=enb2012_data,
                     family='binomial')

#Print coefficients for the logistic regression model
logisticModel$coefficients

#Make predictions using 5-fold cross validation
fitLogisticRegression <- cv.binary(logisticModel, nfolds = 5)

#Convert model output to binary for creating confusion matrix
#and computing accuracy.
fitLogisticPredicted <- fitLogisticRegression$cvhat >= 0.5

#Part (c)
#Print confusion matrices for the linear and logistic
#regression predictions
d<-table(enb2012_data$Y, fitLinearPredicted)
d
d1<-table(enb2012_data$Y, fitLogisticPredicted)
d1

#Print accuracies for the linear and logistic
#regression predictions
sum(diag(d))/sum(d)
sum(diag(d1))/sum(d1)
```