National University of Computer & Emerging Sciences, Karachi

Spring 2018 CS-Department

Final - Solution

17th May 2017, 9:00 am – 12:00 am

Course Code: CS481	**Course Name:** Data Science
Instructor Name: Dr Muhammad Atif Tahir	
Student Roll No:	**Section No:**

Instructions:

- Return the question paper.
- Read each question completely before answering it. There are **6 questions and 4 pages**
- In case of any ambiguity, you may make assumption. But your assumption should not contradict any statement in the question paper.
- All relevant formulas are provided in Appendix.
- In each question, Show all steps clearly.

Time: 180 minutes. **Total Marks**: 50 points

Question 1: Briefly answer the following questions. Each questions should be answered in maximum *20* words including articles. Otherwise, answer will not be checked. [10 Points]

a) As a programmer, list two possible tools that can be used for ETL process
Ans: wget, curl, Beautiful Soup, lxml, trifaca wrangler, kmine or any other possibility

b) Why it is important to scale attributes in kNN classifier?
Ans: Attributes may have to be scaled to prevent distance measures from being dominated by one of the attributes

c) What are the two main advantages of Decision Tree based Classifiers?
Ans: Inexpensive to construct
Extremely fast at classifying unknown records
Easy to interpret for small-sized trees
Accuracy is comparable to other classification techniques for many simple data sets

d) What is Hyperplane?
An hyperplane is a generalization of a plane, in one dimension, an hyperplane is called a point in two dimensions, it is a line, in three dimensions, it is a plane, in more dimensions you can call it an hyperplane

e) Briefly explain the difference between Homogenous and Heterogeneous Classifiers.
Ans: Homogenous Classifiers (Same Classifiers but different training data) e.g. Bagging, Boosting etc
Heterogeneous Classifiers (Different Classifiers but same training data) e.g. Majority Voting, Mean etc)

f) Why the base classifiers should be unstable for bagging?
Ans: due to high variance

g) List two solutions to Initial Centroids Problem in kmeans?
Multiple runs – Helps, but probability is not on your side, Sample and use hierarchical clustering to determine initial centroids, Select more than k initial centroids and then select among these initial centroids – Select most widely separated

h) Can you think of a real-world application in which stop words would be useful for text classification?
One example: Forensic or artistic situation.

i) Explain the difference between feature selection and feature extraction

Feature selection is to find subset of features from original features that can either reduce the cost of features or reduce the cost of extracting features. Feature extraction will transform the features into new domain

j) Why plotting 2D arrays are useful?
For images, 2 -Dimensional data etc.

Question 2: **[5 Points]**

Use the k-means algorithm and Euclidean distance to cluster the following 8 examples into 3 clusters: A1=(2,10), A2=(2,5), A3=(8,4), A4=(5,8), A5=(7,5), A6=(6,4), A7=(1,2), A8=(4,9).
The distance matrix based on the Euclidean distance is given below:

	A1	A2	A3	A4	A5	A6	A7	A8
A1	0	$\sqrt{25}$	$\sqrt{36}$	$\sqrt{13}$	$\sqrt{50}$	$\sqrt{52}$	$\sqrt{65}$	$\sqrt{5}$
A2		0	$\sqrt{37}$	$\sqrt{18}$	$\sqrt{25}$	$\sqrt{17}$	$\sqrt{10}$	$\sqrt{20}$
A3			0	$\sqrt{25}$	$\sqrt{2}$	$\sqrt{2}$	$\sqrt{53}$	$\sqrt{41}$
A4				0	$\sqrt{13}$	$\sqrt{17}$	$\sqrt{52}$	$\sqrt{2}$
A5					0	$\sqrt{2}$	$\sqrt{45}$	$\sqrt{25}$
A6						0	$\sqrt{29}$	$\sqrt{29}$
A7							0	$\sqrt{58}$
A8								0

Suppose that the initial seeds (centers of each cluster) are A1, A4 and A7. Run the k-means algorithm for 1 epoch only. At the end of this epoch show:
a) The new clusters (i.e. the examples belonging to each cluster)
b) The centers of the new clusters
c) Draw a 10 by 10 space with all the 8 points and show the clusters after the first epoch and the new centroids.
d) How many more iterations are needed to converge? Draw the result for each epoch.

Note: Epoch means iteration.

Solution:

Solution:
a)
d(a,b) denotes the Eucledian distance between a and b. It is obtained directly from the distance matrix or calculated as follows: $d(a,b)=sqrt((x_b-x_a)^2+(y_b-y_a)^2))$
seed1=A1=(2,10), seed2=A4=(5,8), seed3=A7=(1,2)

epoch1 – start:

A1:
d(A1, seed1)=0 as A1 is seed1
d(A1, seed2)= $\sqrt{13}$ >0
d(A1, seed3)= $\sqrt{65}$ >0
→A1 ∈ cluster1

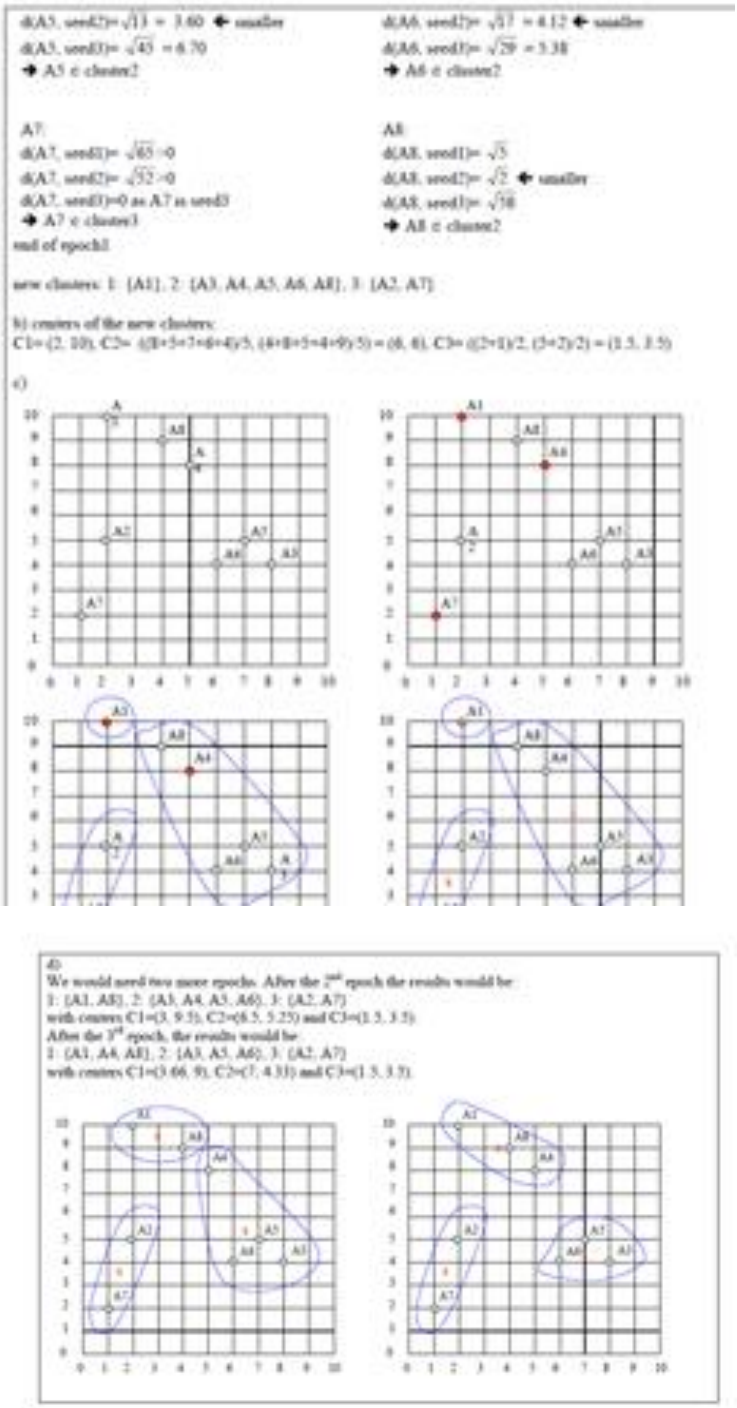
A2:
d(A2,seed1)= $\sqrt{25}$ = 5
d(A2, seed2)= $\sqrt{18}$ = 4.24
d(A2, seed3)= $\sqrt{10}$ = 3.16 ← smaller
→ A2 ∈ cluster3

A3:
d(A3, seed1)= $\sqrt{36}$ = 6
d(A3, seed2)= $\sqrt{25}$ = 5 ← smaller
d(A3, seed3)= $\sqrt{53}$ = 7.28
→ A3 ∈ cluster2

A4:
d(A4, seed1)= $\sqrt{13}$
d(A4, seed2)=0 as A4 is seed2
d(A4, seed3)= $\sqrt{52}$ >0
→ A4 ∈ cluster2

A5:
d(A5, seed1)= $\sqrt{50}$ = 7.07

A6:
d(A6, seed1)= $\sqrt{52}$ = 7.21



Question 3: **[5 Points]**

Consider a Database D (Table 2) consists of 5 transactions. Let min_sup= 60% and min_conf = 80%.

(a) Find frequent Itemsets using Apriori Algorithm [3 Points]

(b) List all the strong association rules (with support s and confidence c) matching with the following metarule, where X is a variable representing customers and $item_i$ denotes variables representing items (e.g. "A", "B", etc)

$$\forall x \in transaction,\ buys(X, item_1) \wedge buys(X, item_2) \Rightarrow buys(X, item_3) \quad [s, c]$$

TID	*items_bought*
T100	{M, O, N, K, E, Y}
T200	{D, O, N, K, E, Y }
T300	{M, A, K, E}
T400	{M, U, C, K, Y}
T500	{C, O, O, K, I ,E}

Table2: Database D for Transactions.

Solution:

C1 =

m	3
o	3
n	2
k	5
e	4
y	3
d	1
a	1
u	1
c	2
i	1

L1 =

m	3
o	3
k	5
e	4
y	3

C2 =

mo	1
mk	3
me	2
my	2
ok	3
oe	3
oy	2
ke	4
ky	3
ey	2

L2 =

mk	3
ok	3
oe	3
ke	4
ky	3

C3=

oke	3
key	2

L3=

oke	3

$$k,o \rightarrow e \ [0.6,1]$$
$$e,o \rightarrow k \ [0.6,1]$$

Question 4: **[8 Points]**

(a) **[4 Points]** Consider a document containing 100 words wherein the word student appears 10 times and word university appears 50 times. Let's also assume that there 10,000 documents and the word student appears in one thousands of these while word university appears in 100 of these. Calculate TFIDF weight for student and university

Answer:

TFIDF (student) = tf * idf = 100/10 * log2(10000/1000) = 10 * log2(10) = 33.2
TFIDF (university) = tf * idf = 100/50 * log2(10000/100) = 2 * log2(100) = 13.2

(b) **[4 Points]** Consider the following data matrix (M) consists of 2 features and 4 instances. Labels are also given

Feature 1	Feature 2
2	3
3	2
1	4
4	1

(a) Use Principal Component Analysis to find the first principal component of the above data matrix (M). In other words, reduce the number of dimensions to 1 (3 Points)

(b) Let's assume that instead of starting with M^TM, you would like to examine eigenvalues using MM^T. What do you will be the eigenvalues using MM^T? Are they going to be same or different (1 Point)?

Ans (a) [30 20; 20, 30; eigen values are 50,10

Ans (b) 50,10 plus 2 zeros

Question 5 [4 Points]

Lets assume the weak learner produces hypotheses of the form: $x < v$, or $x > v$ for a 2-class problem [1,-1]. The threshold v is determined to minimize the probability of error over the entire data (No sampling). Lets assume the following function is obtained after training AdaBoost Classifier

$f(x) = 0.2I(x < 2.5) + 0.7I(x < 8.5) + 0.8I(x > 5)$

Determine the label either 1 or -1 for test data (i) x = 3 and (ii) x = 6. Show all steps

Solution:

x = 3: f(x) = 0.2(-1) + 0.7(1) + 0.8(-1) = -0.3 i.e. -1

x = 6: f(x) = 0.2(-1) + 0.7(1) + 0.8(1) = 1.3 i.e. 1

Question 6: Consider the data set shown below [4 Points]

(a) Estimate the conditional probabilities for P(A|+), P(B|+), P(C|+) , P(A|-), P(B|-), P(C|-)

(b) Use the estimate of conditional probabilities given in the previous question to predict the class label for a test sample (A=0,B=1,C=0) using the Naïve Bayes approach.

Record	A	B	C	Class
1	0	0	0	+
2	0	0	1	−
3	0	1	1	−
4	0	1	1	−
5	0	0	1	+
6	1	0	1	+
7	1	0	1	−
8	1	0	1	−
9	1	1	1	+
10	1	0	1	+

(a) Estimate the conditional probabilities for $P(A|+)$, $P(B|+)$, $P(C|+)$, $P(A|-)$, $P(B|-)$, and $P(C|-)$.

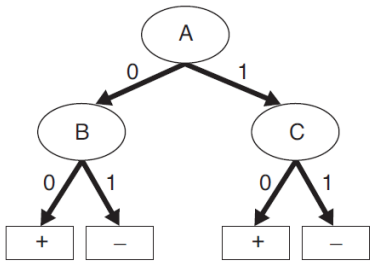
Answer:

$P(A = 1|-) = 2/5 = 0.4$, $P(B = 1|-) = 2/5 = 0.4$,
$P(C = 1|-) = 1$, $P(A = 0|-) = 3/5 = 0.6$,
$P(B = 0|-) = 3/5 = 0.6$, $P(C = 0|-) = 0$; $P(A = 1|+) = 3/5 = 0.6$,
$P(B = 1|+) = 1/5 = 0.2$, $P(C = 1|+) = 2/5 = 0.4$,
$P(A = 0|+) = 2/5 = 0.4$, $P(B = 0|+) = 4/5 = 0.8$,
$P(C = 0|+) = 3/5 = 0.6$.

Let $P(A = 0, B = 1, C = 0) = K$.

$$
\begin{aligned}
& P(+|A = 0, B = 1, C = 0) \\
= & \frac{P(A = 0, B = 1, C = 0|+) \times P(+)}{P(A = 0, B = 1, C = 0)} \\
= & \frac{P(A = 0|+)P(B = 1|+)P(C = 0|+) \times P(+)}{K} \\
= & 0.4 \times 0.2 \times 0.6 \times 0.5/K \\
= & 0.024/K.
\end{aligned}
$$

Question 7: Consider decision tree as shown below **[3 Points]**



Training:

Instance	A	B	C	Class
1	0	0	0	+
2	0	0	1	+
3	0	1	0	+
4	0	1	1	–
5	1	0	0	+
6	1	0	0	+
7	1	1	0	–
8	1	0	1	+
9	1	1	0	–
10	1	1	0	–

Validation:

Instance	A	B	C	Class
11	0	0	0	+
12	0	1	1	+
13	1	1	0	+
14	1	0	1	–
15	1	0	0	+

(a) [1.5 Points] Compute the generalization error rate of the tree using the training data.

(b) [1.5 Points] Compute the generalization error rate of the tree using the validation data.

Solution (a) 5/10 = 50% (b) 1/5 = 20%

Question 8: **[10 Points]**

(a) [4 Points] Write down in the cells below seaborn and matlabplot functions according to their correct definitions: **matplotlib.axes.Axes.hexbin, seaborn swarmplot, seaborn regplot, seaborn jointplot, matlabplot.pyplot.xlim, seaborn heatmap, matplotlib.axes.Axes.hist2d, seaborn stripplot, seaborn violin**

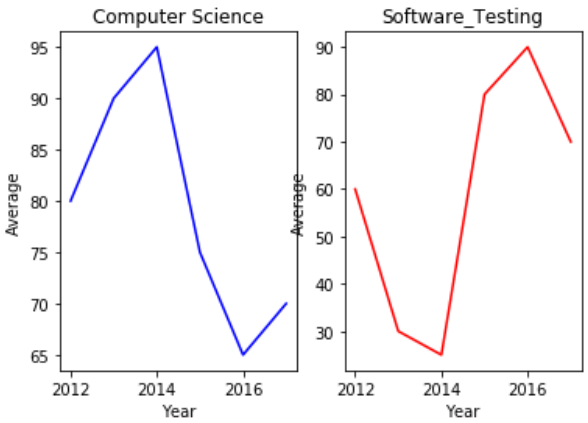
For example; seaborn tsplot	Plot one or more timeseries with flexible representation of uncertainty
seaborn stripplot	Draw a scatterplot where one variable is categorical
matplotlib.axes.Axes.hexbin	Make a hexagonal binning plot
Seaborn violin	Draw a combination of boxplot and kernel density estimate.
seaborn heatmap	Plot rectangular data as a color-encoded matrix
seaborn jointplot	Draw a plot of two variables with bivariate and univariate graphs
seaborn regplot	Plot data and a linear regression model fit
matplotlib.axes.Axes.hist2d	Make a 2D histogram plot
Seaborn swarmplot	Draw a categorical scatterplot with non-overlapping points
matlabplot.pyplot.xlim	Set the x-axis range

(b) [3 Points] Complete the following program to generate the graphs below. Some documentation is added in the appendix but it is not necessary that these methods are needed.

```
# Import matplotlib.pyplot
import matplotlib.pyplot as plt

year = [2012,2013,2014,2015,2016,2017]

average_datascience = [80,90,95,75,65,70]
average_softwaretesting = [60,30,25,80,90,70]
```



Answer: plt.subplot(1,2,1)

```
plt.plot(year,average_datascience, color='blue')
plt.title('Computer Science')
plt.xlabel('Year')
plt.ylabel('Average')

plt.subplot(1,2,2)

plt.plot(year,average_softwaretesting, color='red')
plt.title('Software_Testing')
plt.xlabel('Year')
plt.ylabel('Average')

plt.show()
```

(c) [3 Points] Seaborn is a Python visualization library based on matplotlib. It provides a high-level interface for drawing attractive statistical graphics. One of the simplest things one can do using seaborn is to fit and visualize a simple linear regression between two variables using seaborn lmplot**.** One difference between seaborn and regular matplotlib plotting is that you can pass pandas DataFrames directly to the plot and refer to each column by name. For example, if you were to plot the column 'price' vs the column 'area' from a DataFrame df, you could call lmplot(x='area', y='price', data=df) from seaborn.

Residuals on the other hand visualizie how far datapoints diverge from the regression line. Seaborn residplot function will regress y on x. Below is the syntax of that function
seaborn.residplot(x, y, data=None, lowess=False, x_partial=None, y_partial=None, order=1, robust=False, dropna=True, label=None, color=None, scatter_kws=None, line_kws=None, ax=None)

Based on the above information and following instructions, write down the program regression.py; that will do the following

- Import matplotlib.pyplot and seaborn using the standard names plt and sns respectively.
- Plot a linear regression between the 'weight' column (on the x-axis) and the 'hp' column (on the y-axis) from the DataFrame auto.
- Generate a green residual plot of the regression between 'hp' (on the x-axis) and 'mpg' (on the y-axis). Ignore observations with missing data

```
# Import plotting modules
```

```
import matplotlib.pyplot as plt

import seaborn as sns


# Plot a linear regression between 'weight' and 'hp'

sns.lmplot(x = 'weight', y = 'hp', data=auto)

sns.resigplot(x='hp', y = 'mpg', dropna = True)

# Display the plot

plt.show()
```

Appendix: TDIDF (Help Notes Only)

A more complex way of calculating the weights is called TFIDF, which stands for Term Frequency Inverse Document Frequency. This combines term frequency with a measure of the rarity of a term in the complete set of documents. It has been reported as leading to improved performance over the other methods.

The TFIDF value of a weight X_{ij} is calculated as the product of two values, which correspond to the term frequency and the inverse document frequency, respectively.

The first value is simply the frequency of the jth term, i.e. t_j , in document i. Using this value tends to make terms that are frequent in the given (single) document more important than others.

We measure the value of inverse document frequency by $log2(n/nj)$ where n_j is the number of documents containing term t_j and n is the total number of documents. Using this value tends to make terms that are rare across the collection of documents more important than others.

Appendix:

matplotlib.pyplot.subplot

matplotlib.pyplot.subplot(*args, **kwargs)

Return a subplot axes at the given grid position.

Call signature:

```
subplot(nrows, ncols, index, **kwargs)
```

Axes class ¶

class matplotlib.axes.Axes(*fig*, *rect*, *facecolor=None*, *frameon=True*, *sharex=None*, *sharey=None*, *label=''*, *xscale=None*, *yscale=None*, ***kwargs*)

The Axes contains most of the figure elements: Axis, Tick, Line2D, Text, Polygon, etc., and sets the coordinate system.

Appendix (Formulas)

Bayes Classifier	$$P(C \mid A) = \frac{P(A \mid C)P(C)}{P(A)}$$ **Or** $$P(C \mid A) = \frac{P(A \mid C)P(C)}{P(A \mid C)P(C) + P(A \mid \sim C)P(\sim C)}$$ where ~ = NOT
Conditional Probability	$$P(C \mid A) = \frac{P(A, C)}{P(A)}$$ $$P(A \mid C) = \frac{P(A, C)}{P(C)}$$

BEST OF LUCK!