Fast**FAST- National University of Computer & Emerging Sciences, Karachi.  
FAST School of Computing,  
Mid-1, Spring 2022  
9th March, 2022, 10:00 am – 11:00 am**

|  |  |
| --- | --- |
| **Course Code:** CS4048 | **Course Name:** Data Science |
| **Instructor Names:** Dr. Muhammad Nouman Durrani, Waqas Sheikh, Muhammad Sohail Afzal | |
| **Student Roll No:** | **Section No:** |

**Instructions:**

* Return the question paper along with the answer script.
* Read each question completely before answering it. There are 4 questions on 2 pages
* In case of any ambiguity, you may make an assumption. But your assumption should not contradict any statement in the question paper.

**Time Allowed**: 60 minutes  **Maximum Points**: 22 Points

===============================================================================

**Question 1: Briefly answer the following questions. [ 1 x 10 = 10 Points ]**

1. How Data Science could powers business in 2022?

The following ways a data scientist can add value to business:

* Empowering management and officers to make better decision
* Directing actions based on trends—which in turn help to define goals
* Challenging the staff to adopt best practices and focus on issues that matter Identifying opportunities
* Decision making with quantifiable, data-driven evidence
* Testing these decisions
* Identification and refining of target audiences
* Recruiting the right talent for the organization

1. What is Big Data and why is it important?

Big Data” is data whose scale, diversity, and complexity require new architecture, techniques,

algorithms, and analytics to manage it and extract value and hidden knowledge from it.

It is used for Aggregation and Statistics, Indexing, Searching, Querying and Knowledge discovery

1. How Data Science is different from what statisticians have been doing for years?

The answer lies in the difference between explaining and predicting.

* A statisticians usually explains what is going on by processing history of the data.
* Data Scientist not only does the exploratory analysis to discover insights from it, but also uses various advanced machine learning algorithms to identify the occurrence of a particular event in the future.

1. The first step of a data science process is setting the research goals. What does it explain?

* A project starts by the WHAT, the Why and the HOW? First of all in this phase we understand buiseness goals and context.
* Here we also discuss about the project plan with well-defined milestones.
* We also set a list of deliverables. Here we involves Senior personnel

1. How do you deal with outliers if you find them in your dataset?

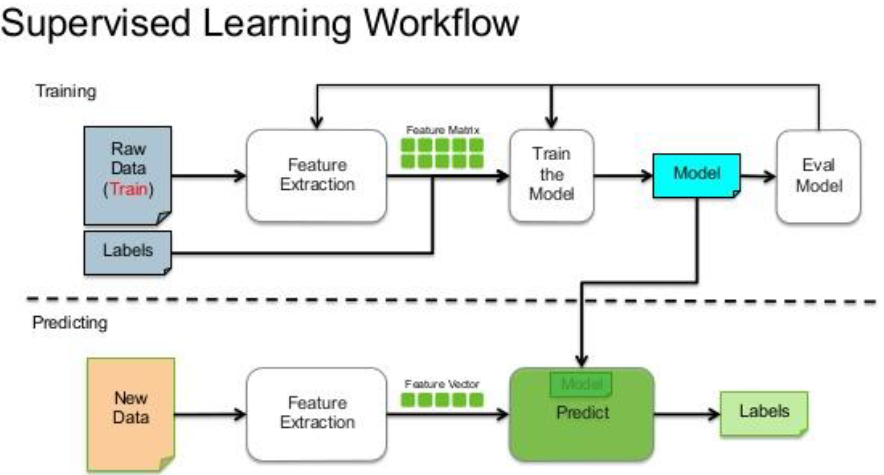
* Trimming/removing the outlier
* Quantile based flooring and capping
* Mean/Median imputation

1. What is the purpose of exploratory data analysis? Which graphical techniques would you employ in this phase?

Exploratory data analysis is an approach for analyzing data sets to summarize their main characteristics, often with visual methods.

EDA usually employs graphical techniques to get better understanding of data, such as Histograms, Line charts, Box plots etc.

1. Draw and discuss the Supervised Learning workflow discussed in the class.



1. You have given the following 2 statements, find which of these option(s) is/are true in case of k-NN?

1. In case of a very large value of k, we may include points from other classes into the neighborhood.

2. In case of a too small value of k, the algorithm is very sensitive to noise

A) 1 B) 2 C) 1 and 2 D) None of these

**C**

1. Suppose we increased the size of the training set. Would this likely improve or deteriorate the performance of the model on new data? Why?

Increasing the size of the training set is likely to improve the model’s performance on new data. Increasing the size of the training set tends to add more variability to the data. More variability tends to make it easier to detect which features are truly correlated with the target class and which features are not. If we had only five training instances, for example, it would be nearly impossible to determine which features are truly correlated.

1. When you select precision over recall as an evaluation metric in any machine learning model?

Precision is a useful metric in cases where False Positive is a higher concern than False Negatives.

Precision is important in music or video recommendation systems, ecommerce websites, etc. Wrong results could lead to customer churn and be harmful to the business.

Recall is a useful metric in cases where False Negative trumps False Positive. Recall is important in medical cases where it doesn’t matter whether we raise a false alarm but the actual positive cases should not go undetected!

**Question 2:**

Suppose we train a model to predict whether an email is Spam or Not Spam. After training the model, we apply it to a test set of 410 new emails (also labeled) and the model produces the following confusion matrix.

|  |  |  |  |
| --- | --- | --- | --- |
|  | | True Class | |
| Spam | Not Spam |
| Predicted Class | Spam | 145 | 18 |
| Not Spam | 37 | 210 |

1. Compute the precision and recall of this model with respect to the Spam class. **[ 2 Points ]**

Precision with respect to SPAM = # correctly predicted as SPAM / # predicted as SPAM

= 145 / (145 + 18) = 145 / 263 = 88.95%

Recall with respect to SPAM = # correctly predicted as SPAM / # true SPAM

= 145 / (145 + 37) = 145 / 182 = 79.67%

1. Suppose we have two users (Sohail and Nouman) with the following preferences.

Sohail hates seeing spam emails in her inbox! However, she doesn’t mind periodically checking the “Junk” directory for genuine emails incorrectly marked as spam.

Nouman doesn’t even know where the “Junk” directory is. He would much prefer to see spam emails in his inbox than to miss genuine emails without knowing!

Which user is more likely to be satisfied with this classifier? Why? **[ 2 Points ]**

**High-precision and low recall with respect to SPAM**: whatever the model classifies as SPAM is probably SPAM. However, many emails that are truly SPAM are misclassified as NOT SPAM. The user is likely to see some SPAM messages in his/her inbox, but will never have to go to the “junk” directory to look for genuine messages incorrectly marked as SPAM.

**High recall and low precision with respect to SPAM:** the model filters all the SPAM emails, but also incorrectly classifies some genuine emails as SPAM. The user will never see SPAM emails in his/her inbox, but will have to periodically check the “junk” directory for genuine emails incorrectly marked as SPAM.

Because the classifier achieves higher precision than recall, Nouman is more likely to be satisfied with the classifier

**Question 3:**

Suppose, you have the following training examples: {((1, 1), −1), ((1, 7), +1), ((3, 3), +1), ((5, 4), −1), ((2, 5), −1)}, as ((*xi*, *yi*), *ci*), where *xi* and *yi* are the two attribute values (positive integers) and *ci* is the binary class label.

Classify a test example at coordinates (3, 6) using a *k-NN classifier* with *k* = 3 and Manhattan distance defined by:

*d*((*u*, *v*), (*p*, *q*)) = |*u* − *p*| + |*v* − *q*|. **[ 3 Points ]**

+1 because the Manhattan distances from (3,6) to each training example are:

(1,1) is distance 2+5=7; **(1,7) is distance 2+1=3;**  **(3,3) is distance 0+3=3;**

(5,4) is distance 2+2=4; and **(2,5) is distance 1+1=2.**

So, points ((2,5), -1), ((1,7), +1) and ((3,3), +1) are the 3 nearest neighbors and their majority class is +1,

so classify (3,6) as class +1

**Question 4:** Consider the data from a dataset named Hepatitus.csv. Answer all the questions using Python code:

**[ 5 Points ]**

1. Import the dataset Hepatitus.csv using the pandas library and convert them into a DataFrame df.

df = pd.read\_csv (Hepatitus.csv)

1. Count the number of missing values in each column using isna(), and remove columns having 30% or more null values using dropna (you can also use your own logic).

df.isna().sum() # count missing values in each column

df.isna().sum()/len(df)\*100

df.dropna(thresh**=**0.3**\***len(df),axis**=**1,inplace**=True**)

1. Fill the NaN values of numeric columns with the average value.

Firstly, we select numeric columns.

**import** numpy **as** np  
numeric **=** df.select\_dtypes(include**=**np.number)  
numeric\_columns **=** numeric.columns

Then, we fill the NaN values of numeric columns with the average value, given by the df.mean() function.

df[numeric\_columns] **=** df[numeric\_columns].fillna(df.mean())

1. Sort the DataFrame by ‘Age’ in ascending order.

df.sort\_values(by=‘Age, ascending=False)

1. Change the column header from “class” to “status”.

df = df.rename(columns={"class":"status"})

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| age | antivirals | fatigue | liver\_big | spleen\_palpable | bilirubin | alk\_phosphate | sgot | albumin | protime | class |
| 30 | FALSE | FALSE | FALSE | FALSE | 1 | 85 | 18 | 4 |  | live |
| 50 | FALSE | TRUE | FALSE | FALSE | 0.9 | 135 | 42 | 3.5 |  | live |
| 78 | FALSE | TRUE | TRUE | FALSE | 0.7 | 96 | 32 | 4 |  | live |
| 31 | TRUE | FALSE | TRUE | FALSE | 0.7 | 46 | 52 | 4 | 80 | live |
| 34 | FALSE | FALSE | TRUE | FALSE | 1 |  | 200 | 4 |  | live |
| 34 | FALSE | FALSE | TRUE | FALSE | 0.9 | 95 | 28 | 4 | 75 | live |
| 51 | FALSE | TRUE | TRUE | TRUE |  |  |  |  |  | die |
| 23 | FALSE | FALSE | TRUE | FALSE | 1 |  |  |  |  | live |
| 39 | FALSE | TRUE | TRUE | FALSE | 0.7 |  | 48 | 4.4 |  | live |
| 30 | FALSE | FALSE | TRUE | FALSE | 1 |  | 120 | 3.9 |  | live |
| 39 | TRUE | FALSE | FALSE | FALSE | 1.3 | 78 | 30 | 4.4 | 85 | live |
| 32 | TRUE | TRUE | TRUE | FALSE | 1 | 59 | 249 | 3.7 | 54 | live |