**Instructions:**

**You can use Word, Excel, Power Point, R and/or Python to answer the questions in this exam. There are a total of seven (7) multi-part questions, with point values noted for each question. You must use Excel if it is specified in the problem statement.**

**Please show your calculations, or the details of your program(s) for each problem. You must supply the R/Python programs, and the programs should be commented so that each step is clearly explained.**

**Combine all of your answers/files into a single zipped file and post the zipped file to CANVAS.**

**#1** (10 Points)

**Measure the distance between (0, 0, 0) and (1, 1, 1) using the following distance formula. Is the following function a proper distance function? Why? Explain your answer.**

A proper distance function should satisfy the following properties:

Non-negativity: d(x, y) ≥ 0 for all x, y

Identity of indiscernibles: d(x, y) = 0 if and only if x = y

Symmetry: d(x, y) = d(y, x) for all x, y

Triangle inequality: d(x, z) ≤ d(x, y) + d(y, z) for all x, y, z

Using the given function, we need to calculate d(x,y) where x = (0, 0, 0) and y = (1, 1, 1):

d(x, y) = Σ((x\_i - y\_i)³)

d((0, 0, 0), (1, 1, 1)) = (0 - 1)³ + (0 - 1)³ + (0 - 1)³ = (-1)³ + (-1)³ + (-1)³ = -1 -1 -1 = -3

Checking validity of the distance function properties on the distance values calculated using given distance function.

1. d(x, y) ≥ 0 for all x, y: The given function does not satisfy the non-negativity property, as we can see that the distance between (0, 0, 0) and (1, 1, 1) is negative (-3).
2. d(x, y) = 0 if and only if x = y: d((0,1,0),(1,0,0)) = (0 - 1)³+ (1 - 0)³ + (0 - 0)³ = 0. However, (0,1,0) is not equal to (1,0,0).
3. d(x, y) = d(y, x) for all x, y: if we swap the points and calculate d((1, 1, 1), (0, 0, 0)): d((1, 1, 1), (0, 0, 0)) = (1 - 0)³ + (1 - 0)³ + (1 - 0)³ = 1³ + 1³ + 1³ = 1 + 1 + 1 = 3 As d(x, y) ≠ d(y, x), it is not satisfied either.

In conclusion, the given function is not a proper distance function because it violates the non-negativity, identity of indiscernibles, and symmetry properties.

**#2** (15 Points)

**Load the “CS513\_targeting\_num.csv” dataset, from the raw\_data module in CANVAS, into R/Python. This is a fictional customer targeting dataset. Perform the EDA analysis by:**

1. **Summarizing each numerical column (e.g., min, max, mean)**

Table

Description automatically generated

1. **Identifying missing values**

**Table

Description automatically generated**

1. **Replacing the numerical missing values with the “median” of the corresponding columns**Median = 6653
2. **Displaying the scatter plot of “Age”, and “Income”**

**Chart, scatter chart

Description automatically generated**

1. **Show the box plots for columns: “Age” and “Income”**

**Chart, box and whisker chart

Description automatically generatedChart, box and whisker chart

Description automatically generated**

**#3** (15 Points)

**Use EXCEL and the "CS513\_targeting\_num\_sml2.csv" file, which contains a different version of fictional customer targeting data for this problem. Employ the weighted k-nearest neighbors (k-NN) algorithm with k=3 in Excel to impute the missing "Income" values for customers whose monthly income ranges between 0 and 25,000 and whose age is between 20 and 70 years old.**

**#4** (15 Points)

**Load the CANVAS “CS513\_targeting\_cat\_full.CSV” dataset into R/Python. Remove the missing values. Construct a Naïve Bayes model to classify “Purchase” (Purchase =’yes’). Predict purchase for a random sample (30%) of the data (test dataset). Measure the following: accuracy, precision, specificity, recall and F1.**

Accuracy: 77.77778 %

Precision: 70.27027 %

Recall: 83.87097 %

Specificity: 73.17073 %

F1 Score: 76.47059 %

**#5** (15 Points)

**Use EXCEL and the "CS513\_targeting\_cat\_sml2.csv" file, which features another version of fictional customer targeting data for this question. Apply the Bayesian and Naïve Bayes algorithms to predict/classify whether a customer in the north who is presented with an offer for product “B” will buy the product or not.**

**Graphical user interface, application, table, Excel

Description automatically generated**

**#6 (**15 Points)

**Load the CANVAS “CS513\_targeting\_num.CSV” dataset into R/Python. Remove the missing values. Construct an unweighted knn (k=3) model to classify “Purchase” (Purchase =’Yes’). Predict the purchase for a random sample (30%) of the data (test dataset). Measure the following metrics: accuracy, precision, specificity, recall and F1.**

Accuracy: 50 %

Precision: 69.23077 %

Specificity: 27.27273 %

Recall: 52.94118 %

F1 Score: 60 %

**#7** (15 Points)

**A COVID test was administered to 1,000,000 individuals. The test correctly identified 95% of those who were sick (P[positive/sick] = 0.95) but also produced a positive result for 10% of those who were not sick (P[positive/not sick] = 0.10). If the prevalence of COVID in this population is 20%, what is the probability that an individual who tested positive is actually sick? What is the probability that an individual who tested negative is actually sick?**

**(Show your calculations)**

P[sick] = 0.20

P[not sick] = 0.80

P[positive/sick] = 0.95

P[positive/not sick] = 0.10

For: P[A|B] = (P[B|A] \* P[A]) / P[B]

P[positive] = P[positive/sick] \* P[sick] + P[positive/not sick] \* P[not sick] = 0.95 \* 0.20 + 0.10 \* 0.80 = 0.19 + 0.08 = 0.27

P[negative] = 1 - P[positive] = 0.73

P[sick/positive] = (P[positive/sick] \* P[sick]) / P[positive] = (0.95 \* 0.20) / 0.27 ≈ 0.7037

P[negative/sick] = 1 - P[positive/sick] = 0.05

P[sick/negative] = (P[negative/sick] \* P[sick]) / P[negative] = (0.05 \* 0.20) / 0.73 ≈ 0.0137

**Datasets: CS513\_targeting\_num.csv, CS513\_targeting\_num\_sml2.csv, CS513\_targeting\_cat\_full.CSV, CS513\_targeting\_cat\_sml2**