ARYAN JAGANI

In-class activities/Lab (IS 733)

Task 1: Understanding Naive Bayes and K-nearest neighbors

1a: Manually calculate prediction using **the Naive Bayes Model and K nearest neighbor, K=2; Euclidean Distance** for the test example for the following example:

- Use any random combination to test/report your probability

ID	Contains Link	Contains Money Words	Length	Class
1	Yes	Yes	Long	Spam
2	No	No	Short	Ham
3	Yes	No	Long	Spam
4	No	Yes	Short	Spam
5	Yes	Yes	Short	Spam
6	No	No	Long	Ham
7	Yes	No	Short	Ham
8	No	Yes	Long	Spam
9	Yes	Yes	Long	Spam
10	No	No	Short	Ham

~~→ k nearest neighbour(k=2) calculations
Euclidean distance
√. √~~

Aryan Jagani
classwork.

Q1 a] Manually calculate prediction using the Naive Bayes model & k nearest neighbour k=2. Euclidean distance for the test example for following example.

Soln—

1] Test example
Contains link = Yes (CL)
Contains Money coords = No (CM)
length = Long. (L)

2] Naives Bayes Calcolations.

a] calculate prior probablities

$$P(spam) = \frac{\text{No of spam Msg}}{\text{Total Msg}} = \frac{6}{10} = 0.6$$

$$P(ham) = \frac{\text{No of Ham Msg}}{\text{Total Msg}} = \frac{4}{10} = 0.4$$

b] Calcolate condition probablity.

i]. P(CL = yes | Spam).

$$P(yes|spam) = \frac{\text{Spam with yes}}{\text{Total spam Msg}} = \frac{4}{6} = 0.67$$

ii] P(CL = yes | ham)

$$P(yes|ham) = \frac{\text{ham with yes}}{\text{Total ham Msg}} = \frac{1}{4} = 0.25$$

- P(CM=No|spam)

  $P(\text{No}|\text{spam}) = \frac{\text{Spam with No}}{\text{Total spam msg}} = \frac{2}{6} = 0.33$.

- P(CM=No|ham)

  $P(\text{No}|\text{ham}) = \frac{\text{Ham with No}}{\text{Total ham msg}} = \frac{3}{4} = 0.75$.

- P(Length=long|spam)

  $P(\text{long}|\text{spam}) = \frac{\text{Spam Msg with long}}{\text{Total spam Msg}} = \frac{3}{6} = 0.5$.

- P(Length=long|ham)

  $P(\text{long}|\text{ham}) = \frac{\text{Ham with long}}{\text{Total ham}} = \frac{2}{4} = 0.5$

c] Apply Bayes Theorem

$P(\text{spam}|X) = P(\text{spam}) \times P(X|\text{spam})$

$P(\text{ham}|X) = P(\text{ham}) \times P(X|\text{ham})$.

$P(\text{spam}|X) = 0.6 \times 0.67 \times 0.33 \times 0.5 = 0.666$

$P(\text{ham}|X) = 0.4 \times 0.25 \times 0.75 \times 0.5 = 0.0375$.

Since P(spam|X) > P(ham|X), the test example is classified as spam.

3] K-Nearest Neighbors (K=2) using Euclidean distance

→ Each feature is represented as

Contains link (yes=1, No=0)

Contains Money word (yes=1, No=0)

Length (Long=1, Short=0).

So, the test example is (1,0,1).

i] Computing Euclidean distance.

$d = \sqrt{(x_1-x_2)^2+(y_1-y_2)^2+(z_1-z_2)^2}$.

calculating the distance from the test example (1,0,1) to all other poins

ID	CL	Cm	L	Class	Distance
1	1	1	1	Spam	$\sqrt{(1-1)^2+(0-1)^2+(1-1)^2} = 1$
2	0	0	0	ham	1.41
3	1	0	1	Spam	0
4	0	1	0	Spam	1.73
5	1	1	0	Spam	1.41
6	0	0	1	ham	1
7	1	0	0	ham	1
8	0	1	1	spam	1
9	1	1	1	spam	1
10	0	0	0	ham	1.41

ii] Find the nearest neighbors

two closest Points

1 ID 3 (distance = 0.0, Spam),

ID1 (distance = 1.0, Spam)

Since Both are spam, test example is classified as Spam.

Final classification

Naive Bayes = Spam

Knearest neighbour = Spam.

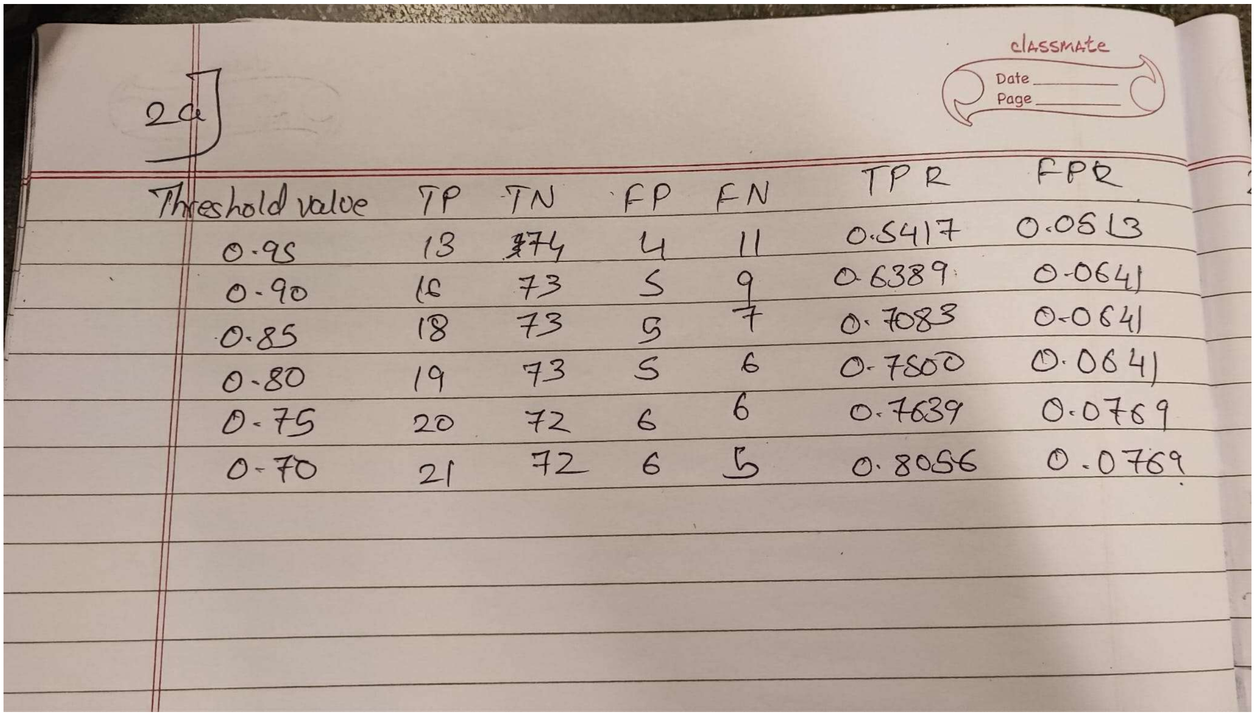
Thus, test example is Spam.

1b: write code (with AI assistant) to build a naive Bayes and KNN classifier. You can use the hamspam.csv to test it out.

https://github.com/AryanJ09/IS733_Class/blob/main/01272025_CW/CW-03-03-2025/CLASSWORK_1B.ipynb

Task2: Understanding ROC and AUC

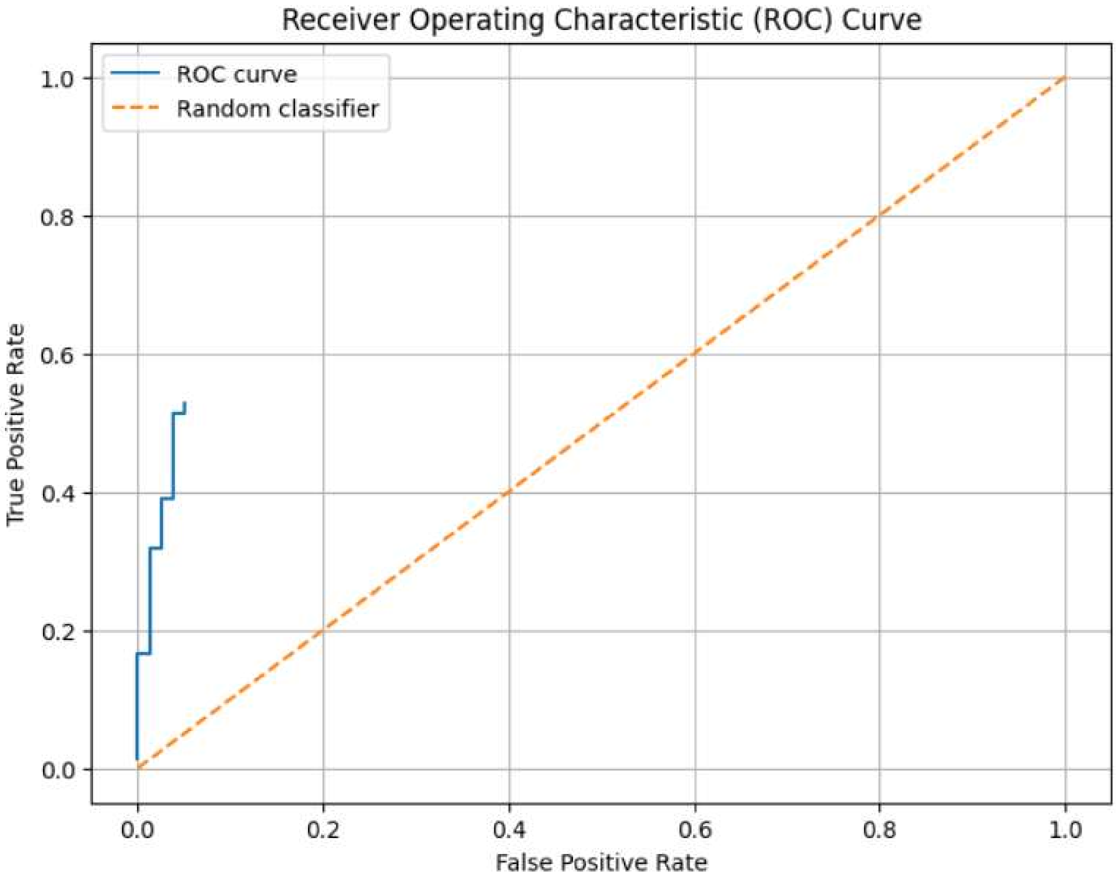
2a: Create a ROC (with AI assistant/Excel) **(Refer to** roc_data.csv**)**
Step1: Given the threshold (0.95,0.90,0.85,0.80,0.75,0.70), derive True Positive and False Positive
Step2: Calculate the True Positive Rate (TPR) and False Positive Rate (FPR), enter the values into the sheet
Step3: plot the set points (FRP, TPR) on the ROC diagram



2a

Threshold value	TP	TN	FP	FN	TPR	FPR
0.95	13	74	4	11	0.5417	0.0513
0.90	16	73	5	9	0.6389	0.0641
0.85	18	73	5	7	0.7083	0.0641
0.80	19	73	5	6	0.7500	0.0641
0.75	20	72	6	6	0.7639	0.0769
0.70	21	72	6	5	0.8056	0.0769

https://github.com/AryanJ09/IS733_Class/blob/main/01272025_CW/CW-03-03-2025/733_classwork_graph_2a.ipynb



https://github.com/AryanJ09/IS733_Class/blob/main/01272025_CW/CW-03-03-2025/roc_dataaryan_q2a.csv

2b. Write code (with AI assistant) to fit the model using your favorite classifier (NB, KNN, or Decision tree); using the hamspam.csv, ask to output an ROC curve and AUC score. (Hint: if you fit a decision tree, you might want to reduce max_depth)

https://github.com/AryanJ09/IS733_Class/blob/main/01272025_CW/CW-03-03-2025/CLASSWORK_2B.ipynb

Submission to blackboard

1a and 2a: photocopy of your manual calculation

The rest of the task (1b, 2b): Python Notebook uploaded to GitHub and submit a link