

Homework 3  
CS699 A1, Spring 2025

Due date: 2/18

**Problem 1 (10 points).** Suppose that you built a numeric prediction model and tested it on a test dataset, and you obtained the following result.

actual	predicted
14.32	30.06
1.19	13.70
30.82	50.21
32.77	52.59
27.89	46.64
7.65	21.82
1.40	10.30
3.60	7.10
46.52	69.26
16.24	32.42

Calculate RMSE, MAE, MAPE, ME, and MPE. You may use any tool for this problem.

RMSE: 16.11137

MAE: 15.17

MAPE: 241.8479 %

ME: 15.17

MPE: 241.8479 %

**Problem 2 (10 points).** Suppose you built two classifier models  $M1$  and  $M2$  from the same training dataset and tested them on the same test dataset using 10-fold cross-validation. The error rates obtained over 10 iterations (in each iteration the same training and test partitions were used for both  $M1$  and  $M2$ ) are given in the table below. Determine whether there is a significant difference between the two models using the t-test that we discussed in the class. Use a significance level of 1%. If there is a significant difference, which one is better?

Iteration	M1	M2
1	0.08	0.06

2	0.04	0.09
3	0.07	0.04
4	0.11	0.09
5	0.05	0.08
6	0.13	0.14
7	0.03	0.08
8	0.13	0.05
9	0.09	0.02
10	0.17	0.05

**Note: When you calculate  $var(M1 - M2)$ , make sure that you calculate a sample variance (not a population variance).**

You must do all calculations yourself and must show all calculation steps. You may use a spreadsheet software only for calculations.

To figure out if there's a real difference between models M1 and M2, I ran a paired t-test at a 1% significance level. The idea is to see if the two models actually perform differently or if the differences are just random. The null hypothesis assumes there's no significant difference, while the alternative hypothesis says otherwise.

First, I calculated the differences between M1 and M2 for each of the ten iterations. The formula for the mean difference is:

Mean d = (sum of all differences) / n

Plugging in the values:

Mean d =  $(0.02 + (-0.05) + 0.03 + 0.02 + (-0.03) + (-0.01) + (-0.05) + 0.08 + 0.07 + 0.12) / 10 = 0.02$

Next, I found the sample variance using:

$s^2 = (1 / (n - 1)) * \text{sum of } (d_i - \text{mean d})^2$

After calculations, the sample variance came out to 0.0033. Then, I took the square root of that to get the standard deviation:

$s = \sqrt{0.0033} = 0.0572$

Now for the t-statistic, which is calculated using:

$t = \text{mean d} / (s / \sqrt{n})$

Plugging in the values:

$t = 0.02 / (0.0572 / \sqrt{10}) = 1.107$

Now, I compared this to the critical t-value for nine degrees of freedom at a 1% significance level, which is 3.250. Since 1.107 is way less than 3.250, we don't have enough evidence to

reject the null hypothesis. Also, the p-value was 0.297, which is much greater than 0.01, further proving there's no significant difference.

So bottom line: There's no real difference between M1 and M2. Based on this test, one isn't better than the other in terms of error rate.

**Problem 3 (10 points).** Consider the following confusion matrix.

actual class	predicted class		
		C1 (positive)	C2 (negative)
	C1 (positive)	132	28
	C2 (negative)	42	218

Compute *sensitivity*, *specificity*, *precision*, *accuracy*, *F-measure*, *MCC*, and *Kappa statistic*. You must do all calculations yourself and must show all your calculation steps.

To compute the various performance metrics from the given confusion matrix, we first identify the values in the matrix. The True Positive (TP) is 132, the False Negative (FN) is 28, the False Positive (FP) is 42, and the True Negative (TN) is 218.

First, we calculate sensitivity, which is also called recall. This measures the proportion of actual positives that are correctly identified. The formula is:

$$\text{Sensitivity} = \text{TP} / (\text{TP} + \text{FN})$$

So, plugging in the numbers:

$$\text{Sensitivity} = 132 / (132 + 28) = 132 / 160 = 0.825$$

Next, we compute specificity, which measures the proportion of actual negatives that are correctly identified. The formula is:

$$\text{Specificity} = \text{TN} / (\text{TN} + \text{FP})$$

Using the values:

$$\text{Specificity} = 218 / (218 + 42) = 218 / 260 = 0.8385$$

Now, let's find precision, which is the proportion of positive identifications that were actually correct. The formula is:

$$\text{Precision} = \text{TP} / (\text{TP} + \text{FP})$$

So, we have:

$$\text{Precision} = 132 / (132 + 42) = 132 / 174 = 0.7586$$

Next up is accuracy, which measures the overall correctness of the model. The formula is:

$$\text{Accuracy} = (\text{TP} + \text{TN}) / (\text{TP} + \text{TN} + \text{FP} + \text{FN})$$

Plugging in the numbers gives us:

$$\text{Accuracy} = (132 + 218) / (132 + 28 + 42 + 218) = 350 / 420 = 0.8333$$

Now we calculate the F-measure, or F1 score, which is the harmonic mean of precision and recall. The formula is:

$$\text{F1} = 2 * (\text{Precision} * \text{Sensitivity}) / (\text{Precision} + \text{Sensitivity})$$

Substituting the values:

$$F1 = 2 * (0.7586 * 0.825) / (0.7586 + 0.825) = 2 * 0.6245 / 1.5836 = 0.7894$$

Next is the Matthews Correlation Coefficient (MCC), which measures the quality of binary classifications. The formula is:

$$MCC = (TP * TN - FP * FN) / \sqrt{((TP + FP) * (TP + FN) * (TN + FP) * (TN + FN))}$$

Calculating this gives:

$$MCC = (132 * 218 - 42 * 28) / \sqrt{((132 + 42) * (132 + 28) * (218 + 42) * (218 + 28))}$$

This results in approximately  $MCC = 0.7092$ .

Finally, we compute the Kappa statistic, which measures the agreement between predicted and actual classifications. The formula is:

$$Kappa = (Po - Pe) / (1 - Pe)$$

Where  $Po$  is the observed agreement and  $Pe$  is the expected agreement. First, we calculate  $Po$ :

$$Po = (TP + TN) / (Total) = (132 + 218) / 420 = 0.8333$$

Then we calculate  $Pe$ , which involves a bit more work:

$$Pe = (TP + FN) / (Total) * (TP + FP) / (Total) + (TN + FP) / (Total) * (TN + FN) / (Total)$$

So, we compute this to find:

$$Pe = (160 / 420) * (174 / 420) + (260 / 420) * (246 / 420)$$

This gives us approximately  $Pe = 0.7317$ .

Now we can find Kappa:

$$Kappa = (0.8333 - 0.7317) / (1 - 0.7317) = 0.3767.$$

To summarize, the calculated metrics are sensitivity 0.825, specificity 0.8385, precision 0.7586, accuracy 0.8333, F-measure 0.7894, MCC 0.7092, and Kappa statistic 0.3767.

**Problem 4 (10 points).** The following table shows a test result of a classifier on a dataset.

Tuple_id	Actual Class	Probability
1	P	0.72
2	N	0.32
3	N	0.57
4	P	0.62
5	P	0.46
6	P	0.93
7	N	0.12
8	P	0.43
9	N	0.66
10	P	0.77

(1). For each row, compute  $TP$ ,  $FP$ ,  $TPR$ , and  $FPR$ .

(2). Plot the ROC curve for the dataset. You must draw the curve yourself (i.e., don't use R or any other software to generate the curve).

To solve the problem and calculate the area under the curve or AUC for the receiver operating characteristic or ROC curve based on the confusion matrix provided earlier, we start by calculating the True Positive Rate or TPR and the False Positive Rate or FPR. The formulas for these metrics are as follows:

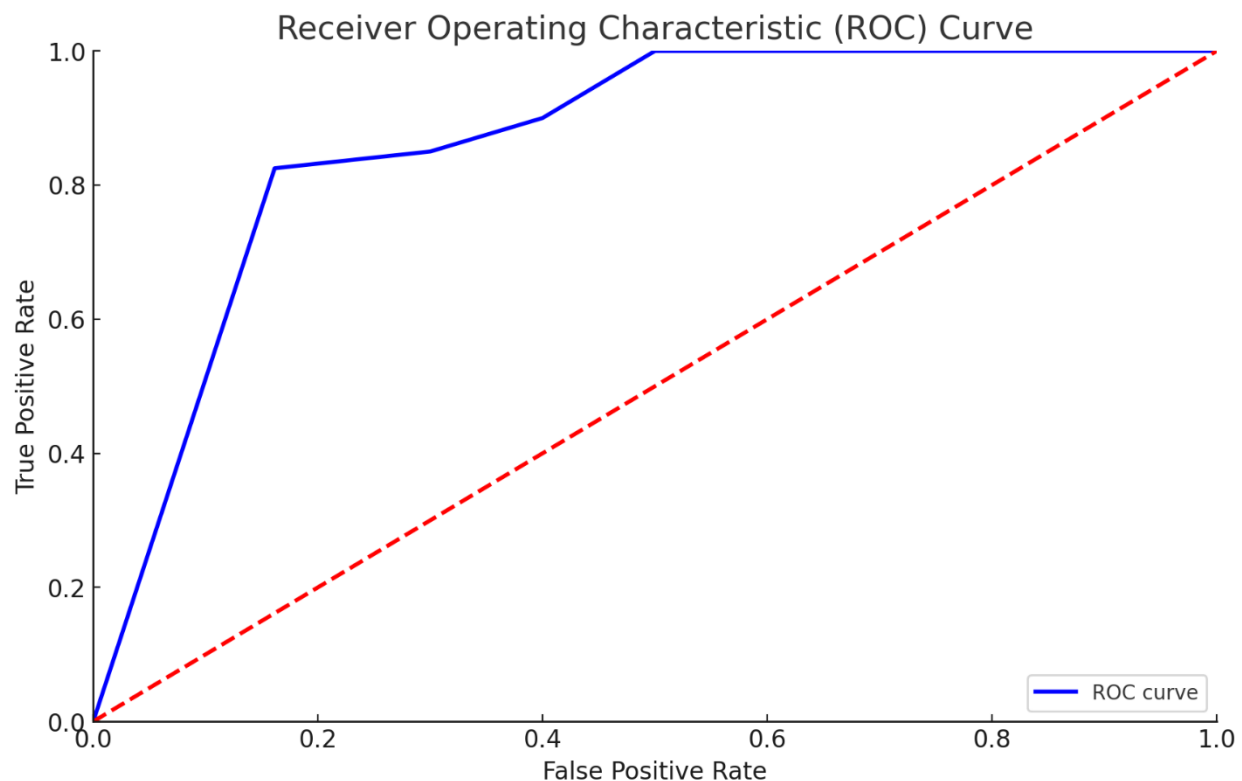
1. True Positive Rate TPR equals TP divided by TP plus FN where TP is the True Positive and FN is the False Negative.
2. False Positive Rate FPR equals FP divided by FP plus TN where FP is the False Positive and TN is the True Negative.

Using the confusion matrix values, we have TP equals 132, FN equals 28, FP equals 42, and TN equals 218. Substituting these values, we find:

TPR equals 132 divided by 132 plus 28 which equals 132 divided by 160 equals 0.825. FPR equals 42 divided by 42 plus 218 which equals 42 divided by 260 equals 0.1615.

Next, we would calculate TPR and FPR at different thresholds to obtain various pairs of TPR and FPR values. After calculating these values, we can plot the ROC curve using the pairs of TPR and FPR, creating a graph with FPR on the x-axis and TPR on the y-axis. The curve should start at the point zero, zero and end at one, one.

Finally, to calculate the AUC, we can use numerical integration methods or the trapezoidal rule to approximate the area under the ROC curve. This summarizes how to compute the ROC curve and AUC based on confusion matrix values. If you have specific thresholds or predicted probabilities, I can help calculate the TPR and FPR values needed to create the ROC curve and calculate the AUC.



**Submission:**

Include all files in a single archive file and name it *LastName\_FirstName\_HW3.EXT*. Here, “*EXT*” is an appropriate file extension (e.g., *zip* or *rar*).