Assignment: Understanding Classification Error

Goal

In this activity you will practice calculating the ROC curve and computing the confusion matrix.

I. Evaluate an AI-based COVID-19 Diagnosis System

Note: The information provided here is based on real research, however, given the seriousness of COVID19, please assume the information here is hypothetical and may contain errors and should not be used for any purpose beyond this assignment.

The current gold-standard for diagnosis of COVID-19 is real-time polymerase chain reaction (RT-PCR) lab test [Bai 2020]. However, lab resources are expensive, limited and time consuming. A quick, cheaper and non-invasive alternative may be to perform CT imaging and use the features such as peripheral distribution, ground-glass opacity and vascular thickening of the CT images for diagnosis [Bai 2020].

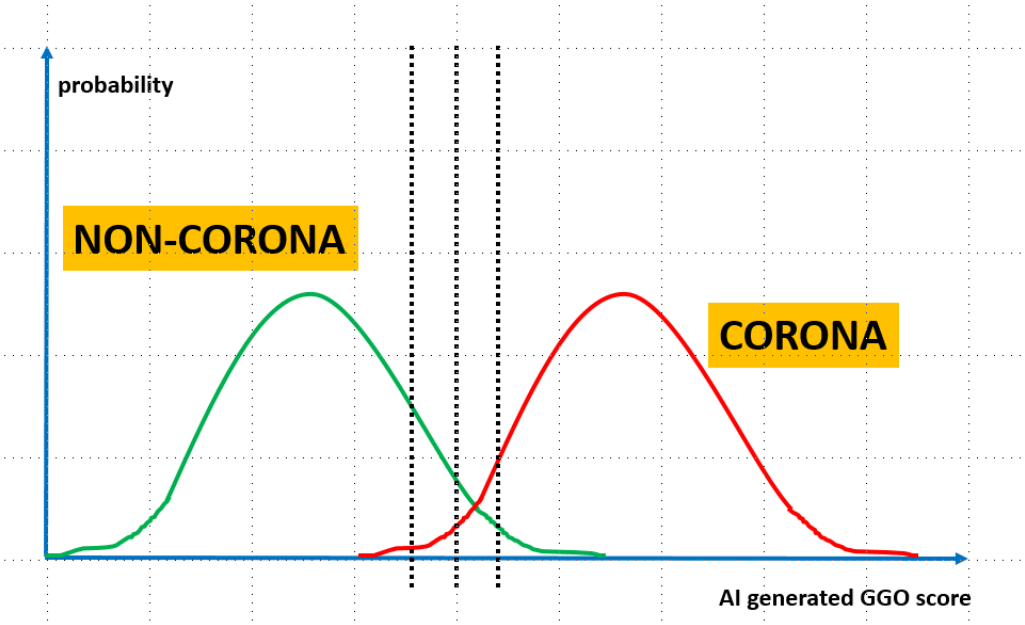
Assume the scientists designed an alternative AI system, which takes in a CT image, recognizes the ground-glass opacity (GGO) feature, and performs the diagnosis in a few seconds. However, there is a trade-off between efficiency and accuracy, so we have to evaluate how much we can trust the system.

(Simulated) Dataset: 100 patients were both tested by RT-PCR and the CT-based AI system: 51 patients were diagnosed by RT-PCR (the gold-standard) as positive (True) while 49 tested negative (False). The raw GGO values were collected from the AI system before making any thresholding.

The data is saved in **data/GGO_value.mat** and **data/diagnosis.mat** respectively.

Question 1

Assume the probability of positive and negative patients follow Gaussian distributions (see the two schematic plots below). Notice there is overlap between the two distributions (which means if we take different thresholds, we'll obtain different prediction results).



(a) Using MATLAB, load the data and find the mean and standard deviation (std) of the Gaussian that models the positive distribution for 51 subjects.
(b) Find the mean and std of the Gaussian that models the negative distribution for 49 subjects by MATLAB.
(c) Plot of the two distributions in MATLAB (using the mean and std values found in parts (a) and (b). Label your axes to obtain a figure similar to the schematic plot shown above. Hint: use MATLAB's **normpdf** function.

Question 1. <u>Your Answers:</u>

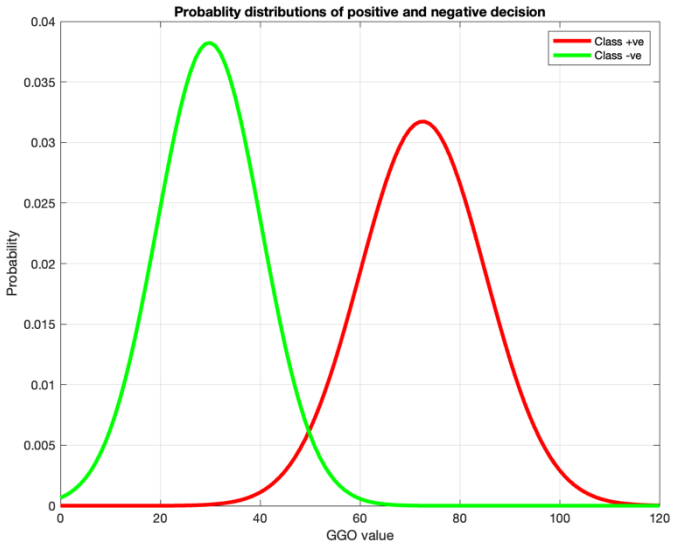
a) Mean of positive subjects: 72.568627450980390

Std of positive subjects: 12.568619497718569

b) Mean of negative subjects: 29.795918367346940

Std of negative subjects: 10.432280814529356

c) Paste plot here:



Paste Code Here:
Loading data:

```
diagnosis_mat = load('diagnosis.mat');
GGO_val_mat = load('GGO_value.mat');
diagnosis_mat_positive = diagnosis_mat.diagnosis ==1;
diagnosis_mat_positive_values =
GGO_val_mat.GGO_values(diagnosis_mat_positive);
diagnosis_mat_negative = diagnosis_mat.diagnosis ==0;
diagnosis_mat_negative_values =
GGO_val_mat.GGO_values(diagnosis_mat_negative);
```

Calculating mean and std:

```
mean_negative_values = mean(diagnosis_mat_negative_values);
mean_positive_values = mean(diagnosis_mat_positive_values);
std_positive_values = std(diagnosis_mat_positive_values);
std_negative_values = std(diagnosis_mat_negative_values);
```

Choose the threshold range and step
Build the distribution function

```
x= 0:1:120;
positive_graph = normpdf(x, mean_positive_values,std_positive_values);
negative_graph = normpdf(x, mean_negative_values,std_negative_values);
```

Plot your figures

```
h1 = plot(x,positive_graph,'r-','LineWidth',3);
hold on;
h2 = plot(x,negative_graph,'g-','LineWidth',3);
legend([h1,h2], "Class +ve", "Class -ve");
xlabel("GGO value");
ylabel("Probability");
title("Probablity distributions of positive and negative decision");
grid on;
```

Question 2

Given the 2 Gaussian distributions in Question 1, the goal is to construct the corresponding ROC curve.

a) Choose your threshold values to construct the ROC curve. Make sure your choice contains at least 10 different values.
b) Plot the ROC curve with true positive rate (TPR) in percentage along the vertical axis, vs. false positive rate (FPR) along the horizontal, using both the erf table and the **normcdf** function. Therefore, you are expected to produce 2 plots.

Note:

- The ROC should show the operating points for equally-separated thresholds.
- Do not use the raw data to calculate the operating points, instead use the Gaussian distributions.
- To calculate needed integrals (i.e., CDF), refer to the lecture slides and make use of the values given in the provided file: **erf_tables.pdf**. Note: for $x < 0$, erf is negative and is equal to -erf(-x) as read from the table.
- Use MATLAB to plot and make sure that the operating points are clearly visible. Double check your answers using MATLAB's built-in function **normcdf** to calculate the integrals over a Gaussian distribution. You can use a finer threshold grid so the ROC curve will look smoother.

Question 2. Your Answers:

a) Choose threshold Values of the ROC Operating points:
```
T=20:5:80
```

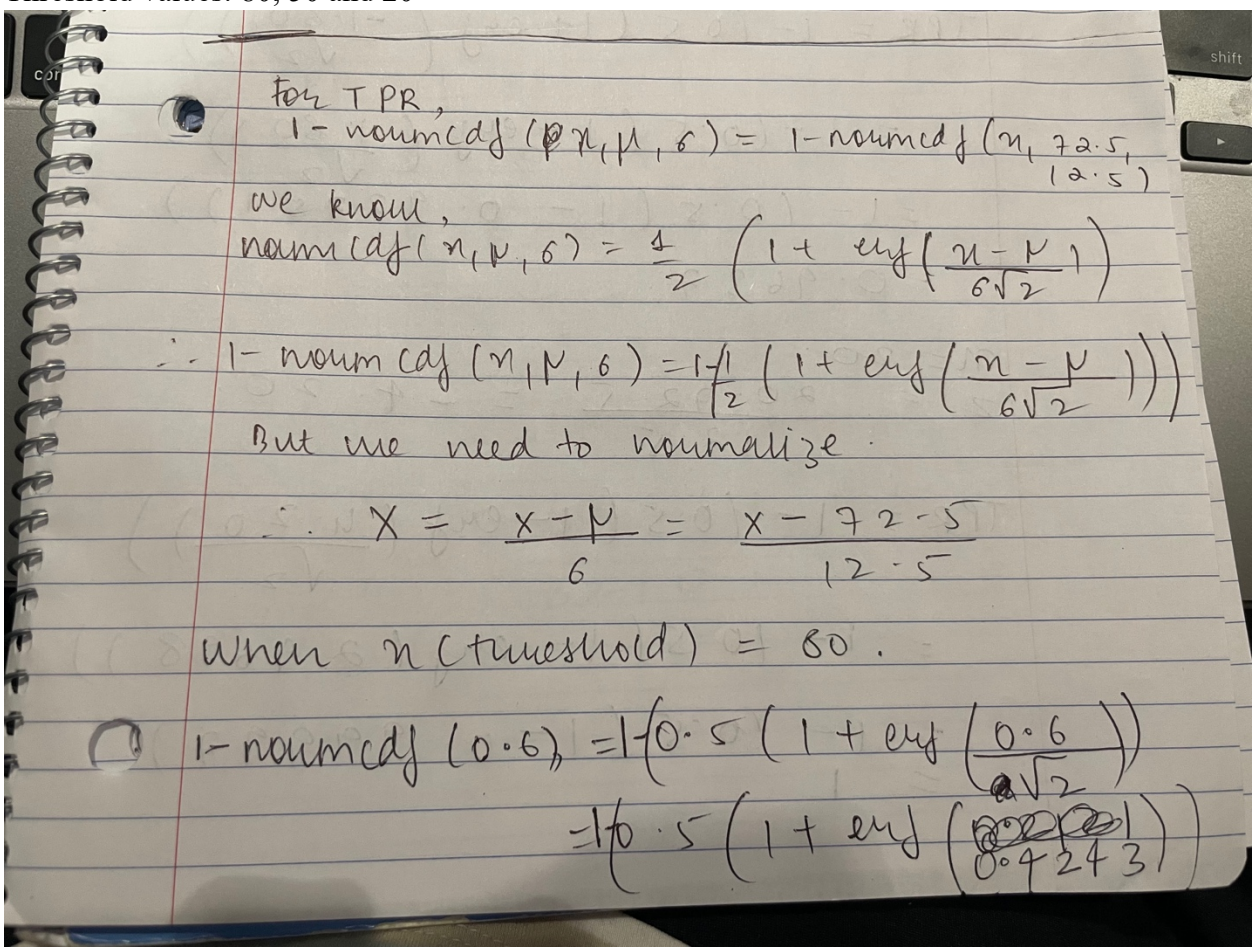
b) Explain how you used erf table for three example thresholds values:
When calculating the erf value you check the table's left column and match your number's One's digit (before decimal point) and the 10^{ths} digit (after decimal point). then you slide to the right to the column matching the 100ths digit in your number.

Error Function Table

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2}\, dt$$

	Hundredths digit of x									
x	0	1	2	3	4	5	6	7	8	9
0.0	0.00000	0.01128	0.02256	0.03384	0.04511	0.05637	0.06762	0.07886	0.09008	0.10128
0.1	0.11246	0.12362	0.13476	0.14587	0.15695	0.16800	0.17901	0.18999	0.20094	0.21184
0.2	0.22270	0.23352	0.24430	0.25502	0.26570	0.27633	0.28690	0.29742	0.30788	0.31828
0.3	0.32863	0.33891	[illegible]	0.35928	0.36936	0.37938	0.38933	0.39921	0.40901	0.41874
0.4	0.42839	0.43797	0.44747	0.45689	0.46623	0.47548	0.48466	0.49375	0.50275	0.51167
0.5	0.52050	0.52924	[illegible]	0.54646	0.55494	[illegible]	0.57162	0.57982	0.58792	0.59594
0.6	0.60386	0.61168	0.61941	0.62705	0.63459	0.64203	0.64938	0.65663	0.66378	0.67084
0.7	0.67780	0.68467	0.69143	0.69810	0.70468	[illegible]	0.71754	0.72382	0.73001	0.73610
0.8	0.74210	0.74800	0.75381	0.75952	0.76514	0.77067	0.77610	0.78144	0.78669	0.79184
0.9	0.79691	0.80188	0.80677	0.81156	0.81627	0.82089	0.82542	0.82987	0.83423	0.83851
1.0	0.84270	0.84681	0.85084	0.85478	0.85865	0.86244	0.86614	0.86977	0.87333	0.87680
1.1	0.88021	0.88353	0.88679	0.88997	0.89308	0.89612	0.89910	[illegible]	0.90484	0.90761
1.2	0.91031	0.91296	0.91553	0.91805	0.92051	0.92290	0.92524	[illegible]	0.92973	0.93190
1.3	0.93401	0.93606	0.93807	0.94002	0.94191	0.94376	0.94556	0.94731	0.94902	0.95067
1.4	0.95229	0.95385	0.95538	0.95686	0.95830	0.95970	0.96105	[illegible]	0.96365	0.96490
1.5	0.96611	0.96728	0.96841	0.96952	0.97059	0.97162	0.97263	0.97360	0.97455	0.97546
1.6	0.97635	0.97721	0.97804	0.97884	0.97962	0.98038	0.98110	0.98181	0.98249	0.98315
1.7	0.98379	0.98441	0.98500	0.98558	0.98613	0.98667	0.98719	0.98769	0.98817	0.98864
1.8	0.98909	0.98952	0.98994	0.99035	0.99074	0.99111	0.99147	0.99182	0.99216	0.99248
1.9	0.99279	0.99309	0.99338	0.99366	0.99392	0.99418	0.99443	0.99466	0.99489	0.99511
2.0	0.99532	0.99552	0.99572	0.99591	0.99609	0.99626	0.99642	0.99658	0.99673	0.99688
2.1	0.99702	0.99715	0.99728	0.99741	0.99753	0.99764	0.99775	0.99785	0.99795	0.99805
2.2	0.99814	0.99822	0.99831	0.99839	0.99846	0.99854	0.99861	0.99867	0.99874	0.99880
2.3	0.99886	0.99891	0.99897	0.99902	0.99906	0.99911	0.99915	0.99920	0.99924	0.99928
2.4	0.99931	0.99935	0.99938	0.99941	0.99944	0.99947	0.99950	0.99952	0.99955	0.99957
2.5	0.99959	0.99961	0.99963	0.99965	0.99967	0.99969	0.99971	0.99972	0.99974	0.99975
2.6	0.99976	0.99978	0.99979	0.99980	0.99981	0.99982	0.99983	0.99984	0.99985	0.99986
2.7	0.99987	0.99987	0.99988	0.99989	0.99989	0.99990	0.99991	0.99991	0.99992	0.99992
2.8	0.99992	0.99993	0.99993	0.99994	0.99994	0.99994	[illegible]	0.99995	0.99995	0.99996
2.9	0.99996	0.99996	0.99996	0.99997	0.99997	0.99997	0.99997	0.99997	0.99997	0.99998
3.0	0.99998	0.99998	0.99998	0.99998	0.99998	0.99998	[illegible]	0.99999	0.99999	0.99999
3.1	0.99999	0.99999	0.99999	0.99999	0.99999	0.99999	0.99999	0.99999	0.99999	[illegible]
3.2	0.99999	0.99999	0.99999	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000

Threshold values: 80, 50 and 20



$= 1 - (0.5(1 + 0.44747))$

$= 0.2763.$

similarly.

$x = 50.$

$z = \frac{50 - 72.5}{12.5} = -1.80$

$TPR = 1 - (0.5(1 + erf(\frac{-1.80}{\sqrt{2}})))$

$= 1 - (0.5(1 - erf(\frac{1.80}{\sqrt{2}})))$

$= 1 - (0.5(1 - 0.92751))$

$= 0.9637.$

$x = 20.$

$z = \frac{20 - 72.5}{12.5} = -4.20.$

$TPR = 1 - (0.5(1 - erf(\frac{4.20}{\sqrt{2}})))$

$= 1 - (0.5(1 - erf(2.9698)))$

$= 1 - (0.5(1 - 0.99997))$

$= 1$

And for FNR.

$x = 80 \Rightarrow z = 80 - 29.7 / 10.43 = 4.8$

$FPR = 1 - (0.5(1 + erf(\frac{4.812}{\sqrt{2}}))$

$= 1 - (0.5(1 + erf(3.4029)))$

$= 1 - (0.5(1 + 1))$

$= 1 - (0.5 \cdot 2)$

$= 1 - 1 = 0.$

when $x = 50 \cdot \Rightarrow z = \frac{50 - 29.7}{10.43} = 1.9463$

$FPR = 1 - (0.5(1 + erf(\frac{1.9463}{\sqrt{2}})))$

$= 1 - (0.5(1 + 0.947 31))$

$= 0.0263.$

when $x = 20 \Rightarrow z = \frac{20 - 29.7}{10.43} = -0.93$

$FPR = 1 - (0.5(1 - erf(\frac{0.93}{\sqrt{2}})))$

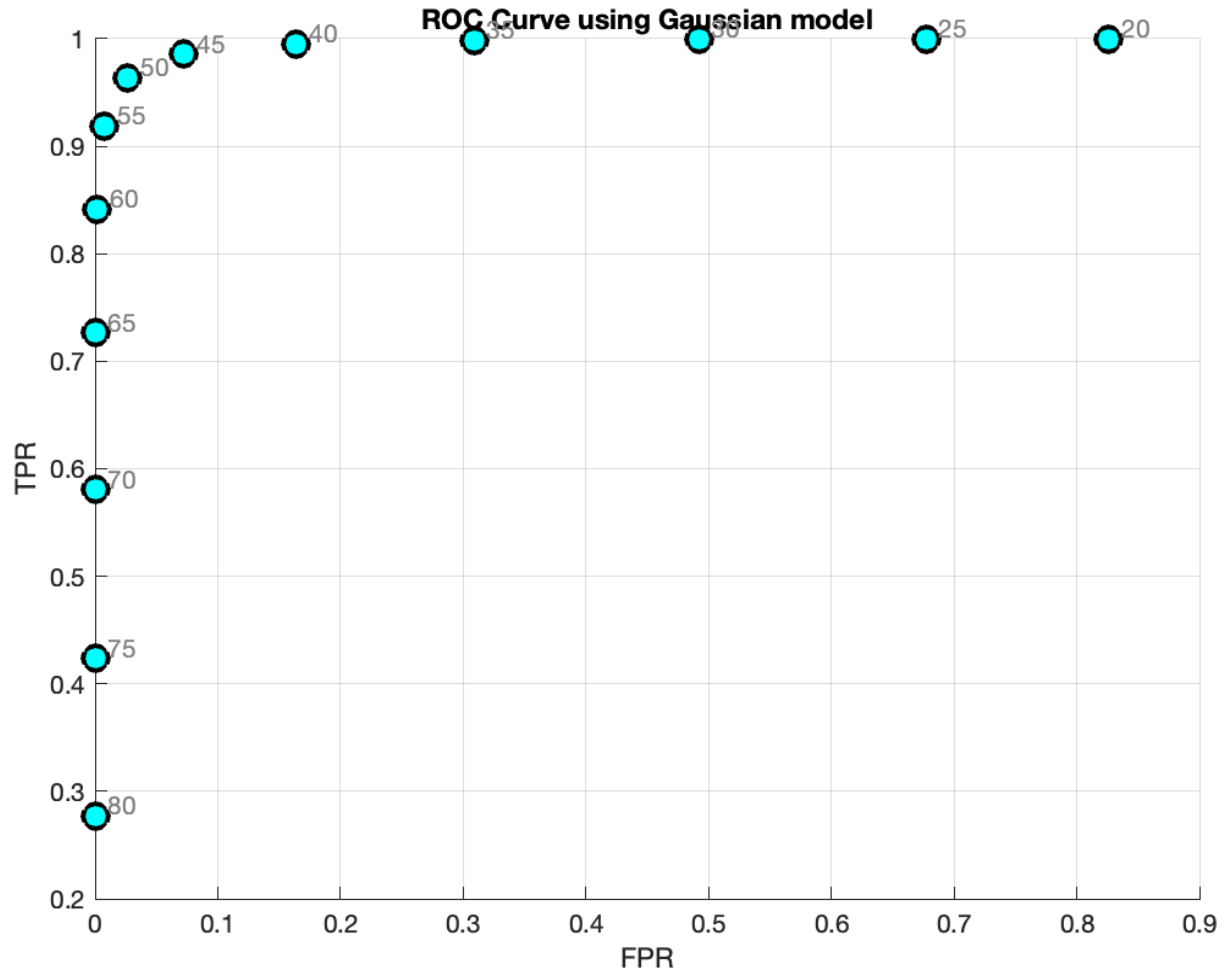
$= 1 - (0.5(1 - 0.64203))$

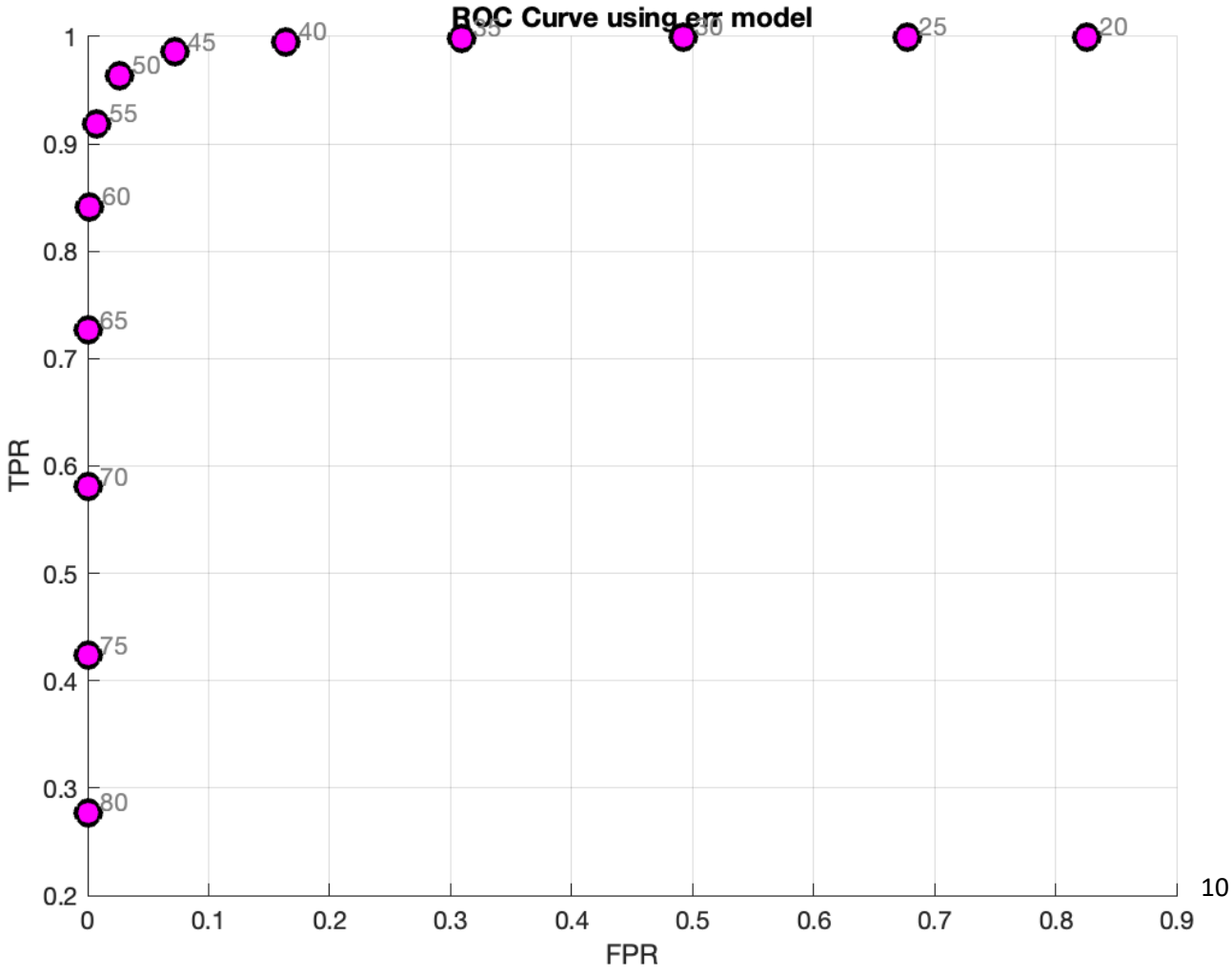
$= 0.8210.$

Paste MATLAB Code for plotting the ROC curve (use scattered points instead of line segments) Here:

```
figure;
grid on;hold on;
xlabel('FPR');
ylabel('TPR');
title('ROC Curve using Gaussian model');
for T=20:5:80
    FPR = 1-normcdf(T,mean_negative_values,std_negative_values);
    TPR = 1-normcdf(T,mean_positive_values,std_positive_values);
    plot(FPR , TPR ,'-ko', 'MarkerSize',10,'MarkerFaceColor',[0 1
1],'LineWidth',2)
    text(FPR+.01 , TPR+0.01, num2str(T),'Color',[0.5 0.5 0.5])

end



figure;
grid on;hold on;
xlabel('FPR');
ylabel('TPR');
title('ROC Curve using err model');

for T=20:5:80
    z1 = (T - mean_positive_values)/(std_positive_values);
    z2 = (T - mean_negative_values)/(std_negative_values);

    FPR = 1-  (0.5 * (1+ (erf(z2/sqrt(2)))));
    TPR = 1-  (0.5 *(1+ (erf(z1/sqrt(2)))));
    plot(FPR , TPR ,'-ko', 'MarkerSize',10,'MarkerFaceColor',[1 0
1],'LineWidth',2)
    text(FPR+.01 , TPR+0.01, num2str(T),'Color',[0.5 0.5 0.5])

end
```

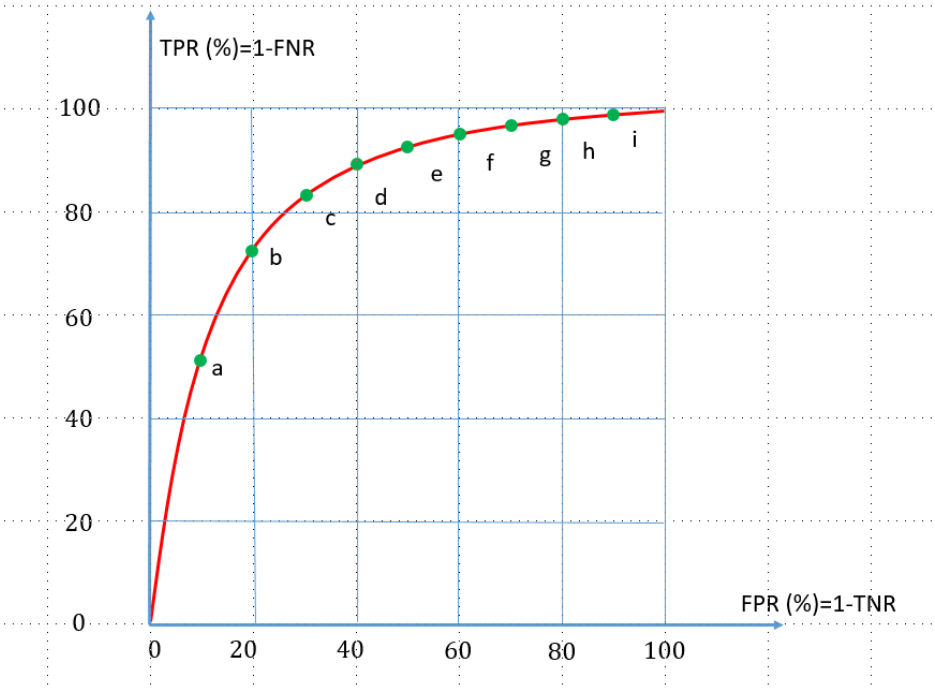
Paste ROC Figure here:


ROC Curve using Gaussian model
TPR
FPR
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
45
40
35
30
25
20
50
55
60
65
70
75
80


ROC Curve using err model
TPR
FPR
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
45
40
35
30
25
20
50
55
60
65
70
75
80

Question 3

Now we look at a dataset collected by a deployed beta-version of the AI system, which resulted in the following ROC curve:



Your task is to control the mis-diagnosis ratio, by reaching a trade-off between TPR and FPR.

1. In particular, you need to tune certain hyper-parameters (e.g., the threshold T) for the decision system so that such that: $FNR \leq 20\%$ with the lowest possible FPR. Among the 9 possible operating points (green dots 'a' through 'i') in the plot above, which one would you choose to satisfy the requirement? Please explain.

2. Assume 20 AI-diagnosed patients had GGO values:
   $V = [70, 60, 30, 80, 40, 20, 50, 90, 85, 45, 75, 65, 55, 15, 35, 45, 45, 65, 65, 75]$.
   Then these 20 patients underwent the more reliable RT-PCR test, which returned, 72 hours later, the following diagnoses, which we regard as "truth":
   C = [P, P, N, P, N, N, N, P, P, N, P, P, P, N, N, N, N, P, P, N].
   where (P: positive, i.e., COVID19; N: negative, i.e., non-COVID19)

   Choose the threshold so that the AI-classification results would have $FNR \leq 10\%$ and $FPR \leq 40\%$? Justify your choice. Note: Calculate FNR using the data points and not a fitted, Gaussian, or any other distribution.

3. Using the threshold, you chose in question 2. Answer the following questions:
   a. How many were misdiagnosed?
   b. How many sick patients were diagnosed as healthy?

c. How many healthy patients were diagnosed as sick?
d. What is the false negative ratio?

4. Calculate entries of 2x2 confusion matrix for the 20 patients using the threshold given in question 2, use the number of patients in the entries, e.g., number of patients that are N but were misdiagnosed as P, etc.

5. Now draw another confusion matrix and enter the percentages instead, i.e., out of 100% negative cases, what percent were correctly classified as N, etc.

Question 3. <u>Your Answers:</u>

1. We need the FNR <= 20 % which makes the TPR > = 80%. This eliminates point a as TNR = ~50%. From the remaining 8 we choose point c to have the lowest possible FPR.

2.

a. List and sort the positive and negative GGO values (you can use MATLAB command sort here)
Positive= [55 60 65 65 65 70 75 80 85 90]
Negative = [15 20 30 35 40 45 45 45 50 75]

b. How to choose a threshold so that FNR<=10%?
For several values in a threshold:
- Make a prediction whether the patient is positive/negative. If <thresh then negative and if >= thresh then positive
- Compare the predictions with the actual status of the patient
- Count all those instances where the model marked the patient as Negative but they actually are Positive then divide that by the total number of Negative patients.
- Add the threshold to a vector if the FNR calculated above <=10%
This was done by the code below:

```
for T=50:5:80
    model_prediction_N = V <= T;
    model_prediction_P = V >= T;

    model_prediction=zeros(size(V));
    model_prediction(model_prediction_N)=0;
    model_prediction(model_prediction_P)=1;

    TP = (model_prediction==1) & (C==1);
    FN = (model_prediction==0) & (C==1);
    FP = (model_prediction==1) & (C==0);
    TN = (model_prediction==0) & (C==0);
    TNR = sum(TN) * 100/nNeg;
    FNR = sum(FN) * 100/nNeg;TPR = sum(TP) * 100/nPos;FPR = sum(FP) * 100/nPos;

    if(FNR <= 10 && FPR <=40)
        tuple = {FNR, FPR, T};
        disp(tuple)

        possibe_solutions = [possibe_solutions, T];
        possibe_FNR = [possibe_FNR, FNR];
        possibe_FPR = [possibe_FPR, FPR];

%choose 55 lowest both
    end

end
```

c. How to choose a threshold so that FPR<=40%?
For several values in a threshold:
   - Make a prediction whether the patient is positive/negative. If <thresh then negative and if >= thresh then positive
   - Compare the predictions with the actual status of the patient
   - Count all those instances where the model marked the patient as Positive but they actually are Negative and then divide that by the total number of Positive patients.
   - Add the threshold to a vector if the FPR calculated above <=40%
   - This was done by the code above.

d. What's your choice of the final threshold?
I chose 55 as the final threshold because it has the lowest possible FNR and FPR of 0% and 10% respectively.

3.

a. Misdiagnosed number = 1

b. Sick diagnosed as healthy = 0

c. Healthy diagnosed as sick = 1

d. FNR = 0

4.

a. Confusion matrix in number

	Model		
Truth		Positive	Negative
	Positive	10	0
	Negative	1	9

5 Confusion matrix %

	Model		
Truth		Positive	Negative
	Positive	100 %	0 %
	Negative	10%	90%

References

[Bai 2020] H. X. Bai et al., "Performance of radiologists in differentiating COVID-19 from viral pneumonia on chest CT", Radiology, 2020. DOI: https://doi.org/10.1148/radiol.2020200823