Name:

|  |  |  |
| --- | --- | --- |
| Problem | Points | Score |
| 0 | 50 | Simply show up… |
| 1 | 60 |  |
| 2 | 70 |  |
| 3 | 80 |  |
| 4 | 90 |  |
| 5 | 100 |  |
| Extra Credit | 10 |  |
| Total | 110 |  |

Notes:

1. This exam must be completed in Python 3. Place your solution in:

/data/courses/engr\_2011/current/exams/exam\_04/lastname\_firstname

1. You are allowed to web surf with Google and use other tools, like ChatGPT, but the code you turn in must be your own and must demonstrate mastery of the material. Simply copying ChatGPT code, or using code that you do not understand, is unacceptable.
2. Your code and results should be placed in directories p01, p02, …, p05. Create one MS Word document, MySolutions.docx, and place in the root directory of your workspace (/data/courses/engr\_2011/current/exams/exam\_04/lastname\_firstname). Put all your explanations in this one document. Use a heading “p01” for the first solution, “p02” for the second, etc. Start a new page for each solution.
3. You must work the problems in order – you cannot skip a problem. For example, you cannot complete p05 until you have finished p04.

This exam is structured in a tiered manner. Start with problem no. 1. When done, copy your code to the next problem and continue editing it. Only turn in code that is completely working. I will grade the highest level you submit. If that level doesn’t pass my test cases, you will get a 50 for this exam (we call this the “you didn’t debug your code and wasted my time” penalty). Therefore, what you submit must work and meet the stated requirements.

If you write good clean code for problem no. 1, you can reuse that code for the remaining problems and quickly solve those problems.

**In this exam, we will work with this data set:**

/data/courses/engr\_2011/current/exams/ex\_04/picone\_joseph/data\_00.csv

You can copy this to your account.

**Problem No.** $1$**:** Write a Python program, myprog.py, that generates an $xy$ scatter plot of this data on a $2D$ grid. Plot the data labeled “turkey” in black, which we will refer to as class no. $1$, and the data labeled “pumpkin\_pie”, which we will refer to as class no. $2$, in orange. Verify your plot is correct by comparing it to a scatter plot generated in Excel. Your program should produce a plot when I run it. Grab a screenshot of the plot you generate, and the plot generated in Excel, and paste it into your MS Word document as your solution for this problem. Comment on any discrepancies.

**Problem No.** $2$**:** Compute the mean of class no. $1$, which we will denote as a vector $m\_{1}$. Do the same for the data in class no. $2$, and denote this mean as a vector $m\_{2}$. For each data point in class no. $1$, compute the distance between $m\_{1}$ and $m\_{2}$. If the data point is closer to $m\_{2}$, this means it was classified as an error. Count the number of errors. Repeat this for class no. $2$ (e.g., if a data point in class no. $2$ is closer to $m\_{1}$**,** this is an error).

Generate what is called a confusion matrix:

|  |  |  |
| --- | --- | --- |
| Ref/Hyp | (1) turkey | (2) pumpkin\_pie |
| (1) turkey | 0 | 0 |
| (2) pumpkin\_pie | 0 | 0 |

For example, if the label was class no. $1$ (“turkey”) and the closest vector was $m\_{1}$, then matrix element $[0,0]$ gets incremented by $1$. If the label was class no. $1$ (“turkey”) and the closest vector was $m\_{2}$, then matrix element $[0,1]$ gets incremented by $1$.

Tabulate a confusion matrix for the data and include it in your Word document as your solution to problem no. $2$. Compare your results to the plot generated in problem no. $1$, and discuss why the error rates are not zero and why your numerical results make sense.

**Problem No. 3:** We are going to repeat problem no. $2$ using probabilities ☺. A multivariate Gaussian model for this data can be written as:

$$p\_{i}\left(x\right)=\frac{1}{\left(2π^{d/2}\right)\left|Σ\_{i}\right|^{1/2}}exp\left(-\frac{1}{2}\left(x-μ\_{i}\right)^{t}Σ\_{i}^{-1}\left(x-μ\_{i}\right)\right)$$

where $Σ\_{i}$is the covariance matrix of the ith class, $μ\_{i}$ is the mean vector, $d$ is the dimension of the data ($2 $in this case) and $x$ is your input vector.

For each class, compute $p\_{i}\left(x\right)$, and choose the class that has the largest probability. Generate a confusion matrix. Compare it to the matrix generated in problem no. $2$. Comment on any similarities or differences.

**Problem No.** $4$**:** Perform a principal component analysis on the covariance matrix for each class and plot the eigenvectors extending from the mean of each class. You should be able to reuse your code from the labs to do this. Include your plot in your MS Word document as the solution for this problem. Comment on whether your results make sense. If they do not, discuss why.

**Problem No.** $5$**:** Using the whitening transformation we discussed in class (see lecture $39$), for class no. $1$, subtract the mean and plot the data after transformation by the whitening transformation. Comment on the shape of the scatter plot and how this relates to the covariance of the transformed data. Did everything work as expected?

**Extra Credit:** Discuss what you think is the single most important concept you learned in this class and why that concept is significant. How do you plan to use this concept in your future engineering career?