CS545 PS 2023 – PROBLEM SET 3

This problem set is due on October 18th. Submit your answers for parts I & II as a rendered python notebook, and part III as a PDF. Try to do all the problems by yourself and write your own code for classifiers, feature extraction, etc; do not make use of toolkits like sklearn, etc.

You should jointly write the project proposal in part III with your project partners and submit it as a team (please coordinate with our TA). Make sure that all the partner names are mentioned in it. If you need more time for this part, the project proposal can be submitted up to a week later (Oct 25th). Feel free to discuss your project ideas with the class staff before you commit to anything.

PART I – IN THEORY

Problem 1. Discriminant functions

For the four following cases that involve two Gaussian distributed classes with the given means and covariances, plot the discriminant function that optimally separates them. Make sure that you plot the two class Gaussians as well.

a) $\mathbf{m}_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ $\mathbf{C}_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ $\mathbf{m}_2 = \begin{bmatrix} 4 \\ 0 \end{bmatrix}$ $\mathbf{C}_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

b) $\mathbf{m}_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ $\mathbf{C}_1 = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}$ $\mathbf{m}_2 = \begin{bmatrix} 4 \\ 3 \end{bmatrix}$ $\mathbf{C}_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

c) $\mathbf{m}_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ $\mathbf{C}_1 = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}$ $\mathbf{m}_2 = \begin{bmatrix} \frac{1}{2} \\ 0 \end{bmatrix}$ $\mathbf{C}_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

d) $\mathbf{m}_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ $\mathbf{C}_1 = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}$ $\mathbf{m}_2 = \begin{bmatrix} 4 \\ 0 \end{bmatrix}$ $\mathbf{C}_2 = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}$

PART II – IN PRACTICE

Problem 2. Handwritten digit recognition

Get the file digits-labels.npz. It contains a set of digit images and their corresponding labels. Split the data so that you keep 100 training digits for each class and use the remaining data as testing samples. Using the training digits, construct a classifier that performs PCA to reduce the dimensionality and learn a Gaussian for each digit class. Evaluate the predicted classification from the test-

ing digits. How many dimensions do you have to use to obtain performance in the 90% range? Make a note of any interesting behavior that you encounter while making your classifier.

Just for fun (and to get extra credit): Can you implement a different classification scheme that can achieve better performance? Can you do so while keeping fewer dimensions in the dimensionality reduction step? Feel free to implement a different classifier, but also a different dimensionality reduction or preprocessing method.

This is a subset of the MNIST digit data that is a standard benchmark for classification tests. You can find the full data set and state of the art results here. The data in the mat file is only the test set data from the original set. As before, you need to write your own code for this problem.

Problem 3. Speech / No Speech classifier

I hate DJs. They talk too much as opposed to playing music. One way to automatically deal with that is to construct a classifier that can distinguish between speech and music. In order to make such a classifier use the data in `SpeechMusic` from the problem set archive. When you unzip it you will find two directories, one full of speech sound files and one full of music sound files.

Perform a spectrogram analysis on each file, take its absolute value and then its log. This is a very common representation for audio. Using this representation, each sound file would provide you with lots of training spectral vectors (i.e. each column of this representation) for both speech and music.

Design a classifier that can distinguish between the two classes. The only design requirement is that given a second of audio data the classifier has to return a decision on whether the input is speech or music. Feel free to use any techniques you want. Also feel free to use an alternative representation as opposed to the one I suggested above. To evaluate your results put aside 10% of the training files to use as test data. Rerun the training a few times with a different set of test files each time. What are your recognition rates for each second of test data?

Now record your voice and find a piece of music that you like. Make sure you use the same sampling rate and put it through the classifier. Does it work? Just as before, write your own code (you can use an existing STFT implementation if you like).

Problem 4. A pool detector

You have to make a swimming pool detector. Use the image `ekalismall.png` to train a classifier, and evaluate its performance on `ekalismall2.png`. The TAs or I will not answer any questions on this problem, you are on your own. Have fun!

PART III – PROJECT PROPOSAL

Propose a project you would like to do for this class. Make sure that this project is related to the material we covered. It is ok if your project involves research you are already working on, but if so it will have to meet higher standards and properly engage all your teammates. If you are an undergrad, pick something doable that will make your resume look good for job/grad school interviews. Either way, do not try to solve the world's greatest problems in a month.

Find a partner or two for your project, we do not have enough time to present/grade all projects otherwise. Submit your proposal as a PDF document via one group member, make sure you mention your partner's names. Don't get lazy and ask to go solo because you don't know how to incorporate a partner to your planned project. A big part of research is compromising while collaborating, another big part is managing partners.

I suggest that your proposal has three parts:

1) A problem that you will definitely be able to solve by the end of the semester, so at worst case you will have something to show.
2) A further elaboration on that problem that is more technically challenging and more in line with a good final project.
3) Yet another elaboration that if resolved will be so good that it will be worthy of publication.

For example, a few years ago we had a student that proposed: 1) A neural network that will denoise voice (something known to be doable), 2) a further extension using a specific RNN architecture which should work better (it did), 3) Using that network to separate voice from real music recordings (ended up being a paper with a couple hundred citations so far). Don't just send in a proposal to use a humongous neural net to solve all of world's problems, it will likely not work well, and does not exhibit a good understanding of how things should be done. What you need to demonstrate though your project is that you understand what you do and why you do it. If any of your explanations is "*I dunno, it just works*" that's bad; "*It doesn't work, but I know exactly why*" is a lot better. Show me that you learned something and that you can think in a principled way.

You will eventually be expected to write a technical paper describing any prior work, what you have done (methodology, results, etc), and, if you are so inclined, what the future steps might be. We will have a poster session at the end of the semester. Past students got conference papers, and started off in their thesis research using projects from this class. I suggest you think along the same lines. Don't waste all your effort for just a class project; make something more of it.

Good luck!