**CPSC 483 FINAL EXAM**

**DUE DATE**: **Monday, Dec 23, 11:59 PM**

**Points to be noted:**

* **THIS IS AN INDIVIDUAL EXAM. ABSOLUTELY NO GROUP WORK AND NO COLLABORATION OF ANY KIND IS ALLOWED.**
* Work should be you own. If you refer to any source on internet or anywhere else, please add a reference to that source. Remember that sources are for reference only, the work is your own.
* Helper packages like numpy, math modules or any other such package or module for data storage and manipulation, standard deviation, mean calculation, plotting data, etc. **ARE TOTALLY ALLOWED.**
* But you should normalize the data, calculate covariance, etc. yourself. Only major change from previous projects is that you can use **linalg** or such function for pca and svd. If you have a question about which package can be used, please email me. The idea is to use as less packages as you can.
* You can use the PCA starter code that I uploaded on Titanium.
* Missing anything from the “things to do” below will result in losing points.
* Absolutely no extensions are allowed. Late Submissions of full or missing parts beyond the deadline will not be accepted.
* Finally, coding is an art. 20% of the grade is for how perfect your code is, not just that is working and doing everything! Tips for perfecting your code is to vectorize (use matrices instead of loops) where ever possible, make it more efficient where ever possible, having perfectly labeled plots, etc. While every student cannot have everything of these, but if you strive to achieve a majority of this, you would earn this 20%.

**Things to do:**

1. Perform normalization, the necessary preprocessing step.
2. Use PCA and SVD on the dataset given below. Reduce the dimensionality to 2 dimensions. Display the principal components from both PCA and SVD. Are they the same ? You can answer this in the output. Project the original data instances (points) onto the principal components resulting from both SVD and PCA. Are they the same ? You can answer this in the output. Plot the two principal components from SVD ( PC1, labeled as such, capturing the maximum variance from the data, and PC2, labeled as such, the next one) and the projections onto them of the original data.
3. Project the features (dimensions) onto the principal components and print out the resulting array. Interpret these projections as a short answer, explaining what features’ essence is captured by PC1 and PC2, and what are they signifying. Remember, when you reduce the dimensions, the new dimensions carry the weight or essence of the more than one original dimension or feature.
4. A working program to do all the above, preferably commenting, broadly indicating which part, does what. Don’t waste time commenting each little bit of code. The program output also should have preceding comments like “Printing original data”, and then print data; “After SVD, Printing original data projected onto principal components” and then print...

**Dataset:**

I have uploaded the **cars dataset** both as .xlsx and .mat. **You should use the .mat version in your code**. The .xlsx version has extra instances or data points, and those are missing some features, as you can see, marked with ‘\*’. The .dat version has eliminated those missing feature instances. The .xlsx is for looking at and understanding what the data is about. You should be able to retrieve the data exactly in the format as .xlsx in your numpy array. You should note “X = points['X']” in PCA starter code only retrieves the numerical data and not the name of the cars. But for plotting the data, you would need the name of the cars. Also, the columns 2 through 9 in the data are just boolean values. You can see that in the .xlsx. You **should not** consider those columns in PCA and SVD calculations. Only begin from column 10 onwards for the features to be incorporated in the original dataset.

**Deliverables:**

1. **All items in things to do above, such as (perfect!) code, Plot figures (Output from running the code), Output printed from running the code, answer to question 3 above, etc.**

Submit item 1) above, in Titanium by **Monday, Dec 23, 11:59 PM**