**CS 4824 / ECE 4424 HW 4 Programming Portion**

**General Instructions:**

* For this programming assignment, we have provided a starter code. Your primary task will be to complete the code from scratch to implement K-Means Clustering. You cannot use implementations of Sklearn or other libraries for this assignment.
* We are providing you with two iPython notebooks: *Clustering.ipynb* and *SklearnKMeans.ipynb*. Both iPython notebooks are almost similar to each other with few differences. *SklearnKMeans.ipynb* uses Sklearn implementation of KMeans Clustering and serves the purpose of sample outputs for each cell in the notebook. *Clustering.ipynb* will use your implementation of KMeans Clustering imported from *cluster\_kmeans.py*.
* You need to upload *cluster\_kmeans.py*, that you will be modifying for this assignment. You also need to upload a ReadMe file for this assignment. We will be testing your code while keeping all other files the same as provided in the original folder. Please make sure that your code works with our provided iPython notebook *Clustering.ipynb*. At the top of your Python submission file and ReadMe file, please include your name and VT email id as a comment, so that it is easy for us to manage your submissions during the grading process.
* We are not providing any test cases file for this assignment. We will test your code with the provided iPython notebook *Clustering.ipynb*.
* You will use customer credit card dataset for this problem, provided in *creditcards.csv*. *SklearnKMeans.ipynb* can also help you in exploring the data.
* Before starting all the tasks, examine the entire codebase. Follow the code from both of the iPython notebooks to see which methods they call. Make sure you understand what all of the code does.

**Programming Tasks:**

**Task 1:** (65 points) Implement the KMeans Clustering from scratch using Class *CustomKMeans* in *custom\_kmeans.py.* Implement the *fit* method and run it using *Clustering.ipynb*. You will also need to maintain these attributes: “*inertia\_*” and “*labels\_”* to run *Clustering.ipynb*. Inertia is the sum of sample distances from their centroid. Labels represent the cluster number for each sample. Hint: You can make two helper functions. One is to assign data points to centroids and other is to update the centroids.

**Task 2:** (20 Points) Choose the best K for above task and defend your choice in the ReadMe file. K represents the number of clusters. Hint: one method could be Elbow Method. Do explore other techniques as well.

**Task 3:** (15 Points) Interpret the clusters to tell, what do each of the clusters represent? Mention that in the ReadMe file.

**Note:** You can also use *plot\_3d.py* to check out the labels along the first three principal components by running `*python plot\_3d.py*` in the command line.