**CMPE-255**

**Jaswanth Karangula**

[**jaswanth.karangula@sjsu.edu**](mailto:jaswanth.karangula@sjsu.edu)

**Question :**

**Option 1: Accelerate regression or classification via Intel or NVIDIA optimizations**

* You can re-use and modify our sample code (regression or classification) based on SKlearn/XGBoost. You can select one of our used dataset in our sample code and only do the modifications for this selected section. You also can choose your own dataset.
* Evaluate and compare the performance of the algorithm before and after the acceleration, draw the comparison graph
* You can select one of the following three acceleration options
  + You can use Intel oneAPI AI Analytics Toolkit to accelerate the algorithm (if you did a similar thing in your homework1, you can try a different example)
  + You can use NVIDIA RAPIDS (cuML, cuDF, XGBOOST,...) to accelerate regression or classification
  + Visualization acceleration (e.g., <https://developer.nvidia.com/blog/making-a-plotly-dash-census-viz-powered-by-rapids/>
  + [Links to an external site.](https://developer.nvidia.com/blog/making-a-plotly-dash-census-viz-powered-by-rapids/)
  + )

**Dataset:**

For this assignment i have selected the Cardiovascular Disease prediction data set from Kaggle

Data set link : <https://www.kaggle.com/datasets/sulianova/cardiovascular-disease-dataset?resource=download>

I have uploaded this dataset using the google cloud console UI

The uploading procedure has been recorded and can be found here

<https://drive.google.com/drive/folders/1DeQNMGLwiik1cMNw6yJ3FDYJGANChWpj?usp=share_link>

**Code:**

Colab Notebook Link :

<https://colab.research.google.com/drive/1ki_4bL3WJeq5xYv-uZMAITJUvjTWUWPm#scrollTo=fqdO21LBvwhC>

**Introduction**

I used NVIDIA RAPIDS – cuML and cuDF to create the regression model and compared the performance of the models run in cpu vs gpu

For this assignment I have performed the ” KNN SVM and Random Forest” and compared the each models performance in CPU vs GPU

**Acceleration solution**

Select the **KNeighborsClassifier, SVM , RandomForestClassifier from cuML**

And train on the train data and test the model on test split and measured the elapsed time

**classifiers = { "KNN": cuml.neighbors.KNeighborsClassifier, "SVM": cuml.svm.SVC, "RandomForest": cuml.ensemble.RandomForestClassifier }**

**start = time.time()**

**clf.fit(X\_train, y\_train)**

**elapsed\_sum += time.time() - start**

**Performance comparison**

The average time taken by the each algorithm for classification on a CPU is

{'KNN': 0.38519883155822754,

'SVM': 526.2775363922119,

'RandomForest': 17.87536895275116}

Similarly The average time taken by the each algorithm for classification on a GPU is

{'KNN': 1.8432966470718384,

'SVM': 14.757538557052612,

'RandomForest': 2.5625743865966797}

