**Assignment 02 – Generative Methods**

There are two main sections in Generative Methods: 1) Sequential Ensembles Methods and 2) Parallel Ensembles. This assignment will use two datasets, one for classification and the other for boosting regression (analogous to regression but not same), and utilize python’s scikit-learn library to evaluate and predict the class or the value of test datasets. As usual, this is a group assignment. Please submit the—**ipynb and pdf both** through the canvas assignment portal. Bold points in parentheses show the earned points for the correct answer. Total points = 75 +5.

You will use the [online shopping](https://archive-beta.ics.uci.edu/ml/datasets/Online%20Shoppers%20Purchasing%20Intention%20Dataset) dataset for classification and [Capital Bike-sharing data](https://archive-beta.ics.uci.edu/ml/datasets/Bike%20Sharing%20Dataset) for regression, available on UC Irvine Machine Learning Repository. Please start with a random.seed(last four digits of your GUIDg).

**I have intentionally misspelled some explanations in assignments. When you copy the text, make sure to spell check it. I will drop points for incorrect spelling and grammar (-5 points). Please also provide explanations/ reasoning for utilizing a command and describe the output. Failure to describe/ explain will result in lower points. Also, make sure to add a g#groupnumber (for example, g01) suffix to each variable, so we will know you are not just copy-pasting the given example code.**

The following needs to be done only once per data set. Do this at the beginning of the assignment.

# Initialization and Data Explortion (total 5 Points)

1. Load the dataset. **(0.5 x 2)**
2. Show first 6 data points using head(). **(0.5 x 2)**
3. Describe pandas Dataframe by using describe. **(0.5 x 2)**
4. Show correlation heat plot of the entire dataset using matplotlib and sns, choose any color pallet (except blue) you like (experiment). **(0.5 x 2)**
5. Show the distribution plots of each variable using hist function from pandas+matplotlib. Also, experiment with visual aspects of the image (not a lot, but an excellent visual will always leave a better impression. you can change color, thickness, font, font size, font color, etc.). Explain the plot distributions as much as you can. For example, you can describe the attributes of the distributions like “*From the distribution plot of variable x we can see that the mean is xx with std dev of yy and the variable seems to be skewed towards left.”* **(0.5 x 2)**

# Intermediate Steps (Essential, no points granted)

The following steps are core to any machine learning analysis. You will need to do them to perform subsequent steps. **So even though this step has no points, not performing them will lose 15 points each from the next step.**

1. Split the dataset into the Training set and Test set. Choose your preferred split and justify the rationale.
2. If you like, you can scale the dataframe features. Scaling does not alter training or prediction of and from the model. If you do choose to scale the dataset, please explain the rationale behind it.

# Classification (total 48)

1. Perform classification routine by using Generative ensemble method for 1) AdaBoost, 2) Gradient Boost, 3) XG Boost, and 4) Bagging  **(total 48, 12 each)** 
   1. Import appropriate algorithm from scikit-learn and explain what you did. **(1.5)**
   2. Create the appropriate classifier and describe what the syntax represents and what parameters you choose **(1.5).**
   3. Train classifier on train data and explain what you did. **(1.5)**
   4. Test/fit classifier test data and explain what you did. **(1.5)**
   5. Calculate accuracy and explain what you did. **(1.5)**
   6. Show both text and visual confusion Matrices using scikit learn and matplotlib and explain what the graph tells you and what you did. **(2.5)**
   7. Repeat the same with a different parameter set and compare the result with **(2)**

# Regression (22 Points)

1. Perform Regression routine by using Generative ensemble method for 1) Gradient Boost, 2) XG Boost, and 4) Bagging  **(total 22, 5.5 each)** 
   1. Import appropriate algorithm from scikit-learn and explain what you did. **(0.75)**
   2. Create the appropriate classifier and describe what the syntax represents and what parameters you choose **(0.75).**
   3. Train classifier on train data and explain what you did. **(0.75)**
   4. Test/fit classifier test data and explain what you did. **(0.75)**
   5. Calculate accuracy and explain what you did. **(0.75)**
   6. Show both text and visual confusion Matrices using scikit learn and matplotlib and explain what the graph tells you and what you did. **(1.25)**
   7. Repeat the same with a different parameter set, and compare the result with **(1)**

# Bonus Question (5)

For all the given classifiers (Q3), evaluate the different parameter sets including (njobs, learning rate, etc).

1. For boosting and bagging compare the tradeoff between njobs and learning rate. Plot the graph of different learning rates vs number of jobs(label the plot correctly. It should show title, x and y tik labels, and x and y axis labels). **(1)**
2. Explain the graph in details, specifically describe the trade off between thelearning rate and n jobs. Also comment on the eolution of error for each combination(1 paragraph at least, **1.5**).
3. For bagging compare the tradeoff between the bootstrap features and max samples. Plot the graph of different combination of bootstrap features and max samples (label the plot correctly. It should show title, x and y tik labels, and x and y axis labels). **(1)**
4. Explain the graph in details, specifically describe the trade off between bootstrap features and max samples (1 paragraph at least, **1.5**)