Instructions

- This homework assignment is worth 75 points.
- Please submit a **.ipynb** file to Blackboard.
- **Please strive for clarity and organization.**
- **Due Date: February 10, 2023 by 11:59 pm.**

For this homework assignment, we will consider the `train.csv` data file. This data represents the results of a large product testing study. For each `product_code` you are given a number of product attributes (fixed for the code) as well as a number of measurement values for each individual product, representing various lab testing methods. Each product is used in a simulated real-world environment experiment, and and absorbs a certain amount of fluid (loading) to see whether or not it fails. The task is to use the data to predict individual product failures. Notice the target variable is `failure`. Also note that there are missing values in this data. **In Python**, answer the following:

Exercise 1

(5 points) Upload the `train.csv` data file to your S3 bucket, and using the pandas library, read the `train.csv` data file and create a data-frame called `train_data`.

Exercise 2

(5 points) Report the number of observations For each `product_code`.

Exercise 3

(15 points) Create two visualizations that may show interesting relationships between the input variables and the target variable. Make sure to describe the visualizations.

Exercise 4

(50 points) Split the `train_data` into two data-frames (taking into account the proportion of 0s and 1s in `failure`): `train` (80%) and `test` (20%). Then, do the following:

(i) Fill the missing values in the `train` using the k-NN imputation strategy (5 neighbors). Then, use that k-NN imputer to fill any missing values in the `test` dataset. After that, build a classification model in the `train` dataset (use all the numerical variables as input variables and `failure` as the target variable), and use this model to predict the likelihood of `failure` in the `test` dataset. Evaluate the performance of the model by computing the area under the ROC curve between the predicted probability and the observed target

variable. Note that you can use the `roc_auc_score` function from scikit-learn to compute the area under the ROC curve.

(ii) Fill the missing values in the `train` using the k-NN imputation strategy (5 neighbors). In this case, fill the missing values based on the `product_code`. That is, for example, fill any missing values for observations in `product_code = A` only using observations in `product_code = A`. Then, use that k-NN imputer to fill any missing values in the `test` dataset. After that, build a classification model (same model from part (i)) in the `train` dataset (use all the numerical variables as input variables and `failure` as the target variable), and use this model to predict the likelihood of `failure` in the `test` dataset. Evaluate the performance of the model by computing the `area under the ROC curve` between the predicted probability and the observed target variable. Note that you can use the `roc_auc_score` function from scikit-learn to compute the area under the ROC curve.

(iii) Based on your results from parts (i) and (ii), what strategy would you use to predict `failure`? Be specific.