**Homework 2**

Due 10/30/2023

## Part 1. Reflections on Homework 1

From the feedback you received (either from the instructor/TA or peers), what are the takeaways/lessons learned you could apply to future analysis?

## Part 2. Create a model card

The model card contains information about the properties of models. This is one way of organizing the knowledge about the model, which becomes handy in data science problem solving. Prepare a table summarizing the properties of each base model we learned so far (Decision tree, Naive Bayes, K-nearest neighbour, logistic regression, SVM) along the following properties:

1. parametric or non-parametric
2. Input (continuous or discrete or both or mixed)
3. Output (continuous or discrete or both)
4. Can the model handle missing value
5. Model representation
6. Model Parameters
7. How to make the model more complex
8. How to make the model less complex
9. Is the model interpretable or transparent

Your table can be organized into rows and columns, rows are the properties and column are the models. This table can be expanded in the future when you learn additional new models

## Part 3. Wine-Tasting Machine

In this task, we will practice building supervised machine learning with Logistic Regression (LR), Naïve Bayes (NB), Support Vector Machine (SVM), and Decision tree (DT), Random Forest (RF) classifiers, as compared with simple/baseline methods. The data for this exercise comes from the wine industry. Each record represents a sample of a specific wine product, the input attributes include its organoleptic characteristics, and the output denotes the quality class of each wine: {high, low}. The labels have been assigned by human wine-tasting experts, and we can treat that information as "ground truth" in this exercise. Your job is to build the best model to predict wine quality from its characteristics so that the winery can replace the costly services of professional sommeliers with your automated alternative to enable quick and effective quality tracking of their wines at production facilities. They need to know whether such change is feasible and what extent inaccuracies may be involved in using your tool.

You will be asked to run experiments in Python.

You are given two datasets red-wine.csv and white-wine.csv: Dataset folder

**Python Tasks (50 points)**

1. Read **red-wine.csv** into Python as a data frame, use a pandas profiling tool (https://github.com/pandas-profiling/pandas-profiling) to create an HTML file, and paste a screenshot of the HTML file here (10 points)

2. Fit a model using each of the following methods and report the performance metrics of 10-fold cross-validation using **red-wine.csv** as the training set (25 points).
   *Note:*
   - *You are not required to tune the parameter for this homework assignment.*
   - *You can use the default parameter for each model.*
   - *Baseline model accuracy is the accuracy when predicting the majority class; Baseline model AUC is the random classifier AUC*

| Model | Baseline | Logistic Regression | Naive Bayes | Decision Tree | SVM-Linear | SVM-RBF | Random Forest |
|---|---|---|---|---|---|---|---|
| AUC | | | | | | | |
| Accuracy | | | | | | | |

3. Plot the ROC curve of the Random Forest classifier from the Python package, and paste a screenshot of your ROC curve here (10 points)
4. Using the best model obtained above in Q2 (according to AUC), running the model on **white-wine.csv,** and reporting the AUC score, comment on the performance. (5 points)
5. Suppose all the models have comparable performance. Which model would you prefer if the wine-tasting experts would like to gain some insights into the model? Note: there could be multiple model types fitting this criterion. (5 points)

GPT policy:

You are allowed to use GPT to complete this assignment, if you did, please make sure you summarize your GPT usage (GPT statement), and share a link to the chat history.

Deliverable:

- An editable link to Google Doc with answers to Parts 1, 2, and 3 (i.e. anyone with the link can edit)
- A link to the Python Notebook uploaded to GitHub
- If you use GPT, the GPT statement and link to the chat history

## Reference (Python)

- Run K-fold cross-validation experiment
    - https://www.askpython.com/python/examples/k-fold-cross-validation

- Fitting model and compute AUC/ROC
    https://www.youtube.com/watch?v=uVJXPPrWRJ0

- Baseline model OneR and ZeroR - you may refer to https://scikit-learn.org/stable/modules/generated/sklearn.dummy.DummyClassifier.html

  Or you may implement your own version

- Model fitting
    - You should be able to find all the following model fitting functions from the sklearn package

https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html

https://scikit-learn.org/stable/modules/naive_bayes.html

https://scikit-learn.org/stable/modules/svm.html#svm-classification

https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

https://scikit-learn.org/stable/modules/tree.html