Homework 2

Part 1. Reflections on Homework 1

Feedback from instructor/TA :

Based on the feedback from Homework 1, I'll focus on improving clarity, data cleaning, visualization, statistical explanations, and citation. Peer feedback and collaboration will remain essential for enhancing the quality of my work.

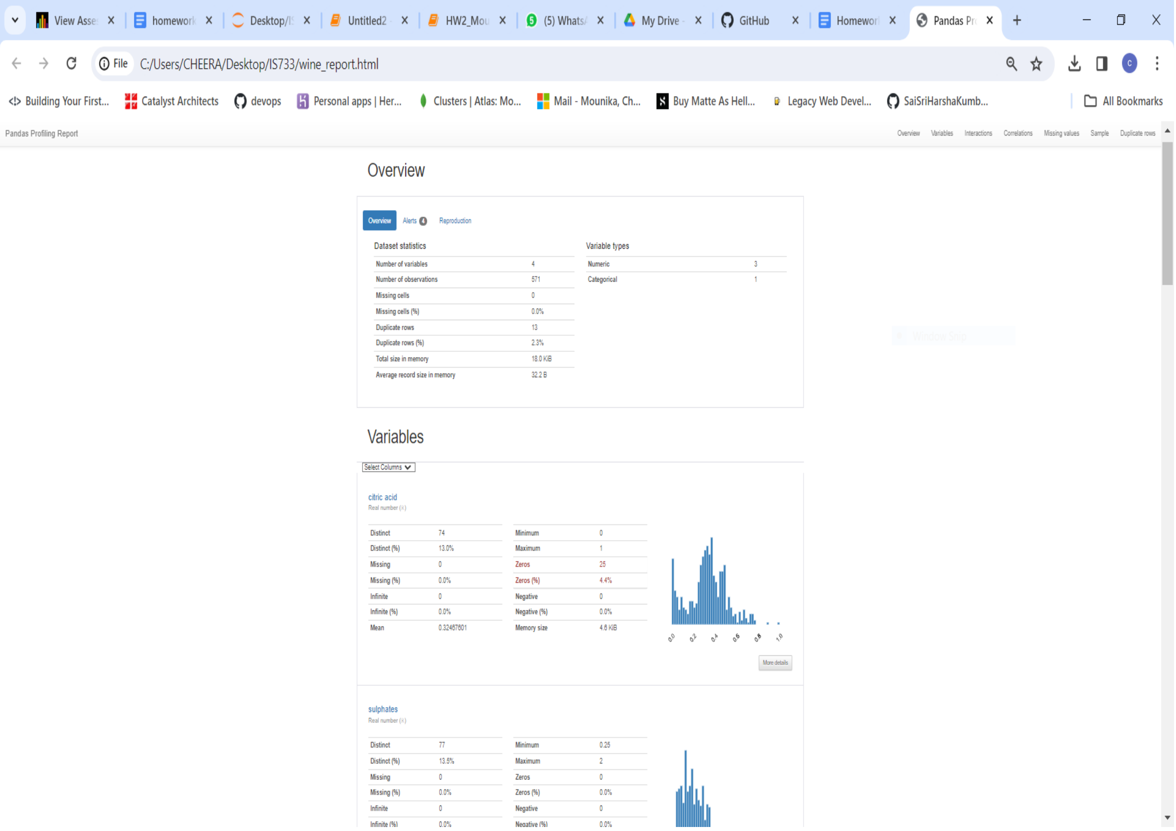
Part 2. Create a model card

Properties	**Decision Tree**	**Naive Bayes**	**K-Nearest Neighbors**	**Logistic Regression**	**Support Vector Machine**
Parametric or Non-parametric	Non-parametric	Parametric	Non-parametric	Parametric	Non-parametric
Input	Both	Both	Both	Both	Both
Output	Both	Discrete	Discrete	Discrete	Discrete
Handles Missing Values	Yes	Yes	Yes	Yes	Yes
Model Representation	Tree structure	Probabilistic	Proximity-based	Linear equation	Hyperplane

Model Parameters	Depth, Impurity Metric	Prior probabilities, Conditional probabilities	Number of Neighbors (K)	Weights and Bias	Kernel Parameters
Make Model More Complex	Increase depth or allow more splits	Incorporate more features, Fine-tune probabilities	Increase K (Number of Neighbors), Use a distance weighting	Add more features, Use higher-order terms	Use a more complex kernel, Increase regularization parameter (C)
Make Model Less Complex	Decrease depth, Limit the number of splits	Simplify feature assumptions, Reduce features	Decrease K, Use a simpler distance metric	Reduce features, Use simpler terms, Regularize coefficients	Use a simpler kernel, Decrease regularization parameter (C)
Interpretable or Transparent	Can be interpretable, Depending on depth and features	Interpretable, Relatively transparent	Less interpretable, Proximity-based	Can be interpretable, Depending on features	Less interpretable, Depending on kernel

Part 3. Wine-Tasting Machine

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)**



Overview

Dataset statistics

Number of variables	4
Number of observations	571
Missing cells	0
Missing cells (%)	0.0%
Duplicate rows	13
Duplicate rows (%)	2.3%
Total size in memory	18.0 KiB
Average record size in memory	32.2 B

Variable types

Numeric	3
Categorical	1

Variables

citric acid
Real number (ℝ)

Distinct	74	Minimum	0
Distinct (%)	13.0%	Maximum	1
Missing	0	Zeros	25
Missing (%)	0.0%	Zeros (%)	4.4%
Infinite	0	Negative	0
Infinite (%)	0.0%	Negative (%)	0.0%
Mean	0.32467601	Memory size	4.6 KiB

sulphates
Real number (ℝ)

Distinct	77	Minimum	0.25
Distinct (%)	13.5%	Maximum	2
Missing	0	Zeros	0
Missing (%)	0.0%	Zeros (%)	0.0%
Infinite	0	Negative	0
Infinite (%)	0.0%	Negative (%)	0.0%

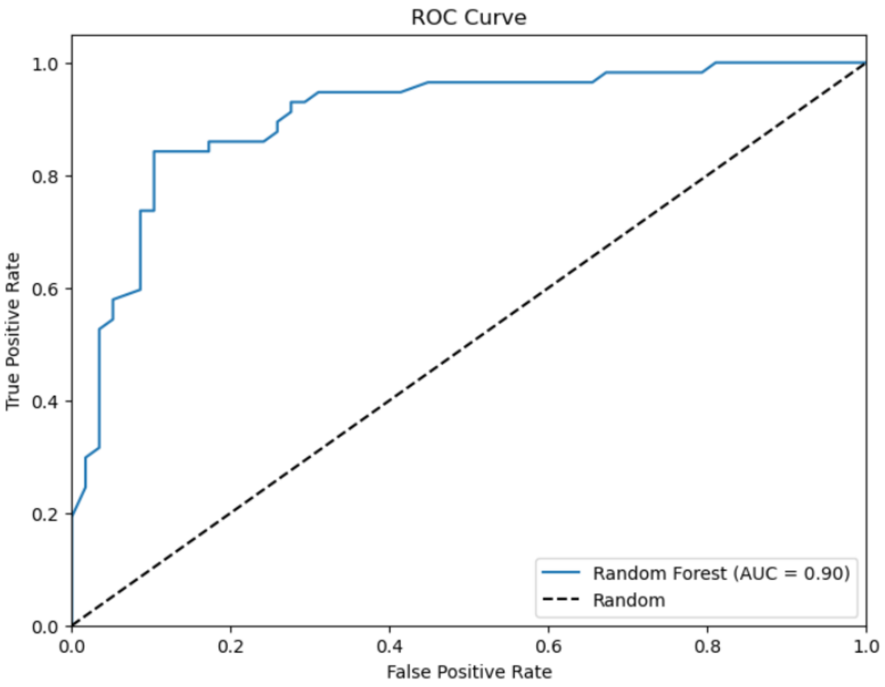
2.Fit a model using each of the following methods and report the performance metrics of 10-fold cross-validation using red-wine.csv as the training set (25 points).

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	AUC	Accuracy
Baseline	0.50	0.53
Logistic Regression	0.88	0.79
Naive Bayes	0.90	0.82
Decision Tree	0.82	0.81
SVM(Linear)	0.88	0.79
SVM(RBF)	0.86	0.54
Random Forest	0.93	0.85

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)

Mean AUC for the Random Forest model on white-wine.csv: 0.9566239316
The Random Forest model has excellent performance on the white-wine dataset.

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)

Based on the results we've obtained, the "Random Forest" model stands out as the best performer. It achieved the highest AUC of 0.93 and an accuracy of 0.85, indicating strong predictive capabilities.
However, it's worth noting that the choice of the "better" model depends on our specific goals. If our main objective is to achieve the highest predictive performance, then Random Forest is the clear winner based on the accuracy and AUC.
On the other hand, if we prioritize model interpretability and the ability to provide insights to wine-tasting experts, we might consider other models such as Logistic Regression, Naive Bayes, or Decision Trees. While their performance metrics are slightly lower, they offer greater interpretability and can help experts understand the factors driving wine quality predictions.

In summary, if performance is our primary concern, Random Forest is the better choice. If interpretability and insights for experts matter more, then models like Logistic Regression, Naive Bayes, or Decision Trees are worth considering.

GPT Usage:

In our conversation, I received valuable assistance on various aspects of data science. I began by exploring Python code to handle CSV files and generate pandas profiling reports, gaining practical insights for data analysis. Additionally, I delved into the characteristics of base models like decision trees, Naive Bayes, K-Nearest Neighbors, logistic regression, and SVM, which provided a solid foundation in model understanding.

I also sought and received Python code to calculate the AUC score for a Random Forest Classifier, a task that will undoubtedly be invaluable in future analyses. Finally, I was provided with code to evaluate a trained model's performance on a white-wine dataset, which will be immensely helpful in practical applications.