**IS-733 HOMEWORK 2**

**Part 1. Reflections on Homework 1**

**1.** From the feedback you received, what are the takeaways/lessons learned you could apply to future analysis?

**1 Ans: -** The Homework 1 was understanding the data, creating a dataset profile using the data profiling library to clearly understand its attributes and variables, creating temporal plots based on the data, using data mining techniques to solve any machine learning problems the data may encounter, and finally creating a dashboard for users to explore the data were all covered in Homework 1. I was able to learn more about the meaning of category and numerical values, the amount of mean, median, and standard deviation values, and the missing values in the data by using data profiling on datasets pertaining to water quality measures. I was able to learn more about data, its applications, how to separate it, and how to use it for analysis thanks to this.

I was able to comprehend the data patterns from the temporal plots, such as the day with the highest readings and the variations over the course of the year and the months. I learned how to categorize, sort, and organize data using data mining techniques from the machine learning problem. I also learned how to train a model to comprehend the data and provide the necessary analysis output that can be applied to similar data in the future.

I have been able to identify the possible uses of machine learning models with the use of distribution plots.I could examine the frequency of readings for different qualities more easily thanks to the distribution plots, which provided me with a better look at different water quality measurements in the form of barplots and histograms. I was able to gather the necessary data from these and discovered how to use heatmaps and scatter plots to comprehend the relationship between parameters and apply them to any future use cases.

Determining whether water quality is good or bad was the machine learning problem I identified from my dataset. To do this, I used the random forest model to test on 20% of the dataset after training it on 80% of it. In this process I have learnt how to split, and train the dataset, how to choose the optimal model (i.e. which one to use whether it is classification, regression, or Naive bayes), and how to train the model on this dataset. To confirm whether the model is operating correctly, later check to see if the accuracy, precision, recall, and F-1 score match expectations. I wasn't sure how to pick the optimal modeling technique for the data at first, but I've since confirmed which model should be selected based on accuracy metrics and AUC score. These assisted me in relearning and refining the data mining approaches.

I have a great understanding of how to use features like rolling mean, scaling over plots, and data interpretation while building the dashboard with Python libraries like dash. This has aided me in developing a dashboard that allows me to download the plot, zoom in and out, and choose a specific moment on the plot to search for readings. These have aided me in honing my ability to be precise and pay close attention to plot nuances in order to conduct a more thorough analysis of the story.

Overall, I've learned from Homework 1 that, in addition to data analysis, it's critical to interpret the data and give consumers visual representations of the findings. My understanding of data visualization tools and data mining techniques has improved, and I was able to change the data modeling once more. I can now use these for future analysis and data mining problems.

**Part 2. Create a model card**

The model card contains information about the properties of the 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 neighbor, logistic regression, SVM) with respect to 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

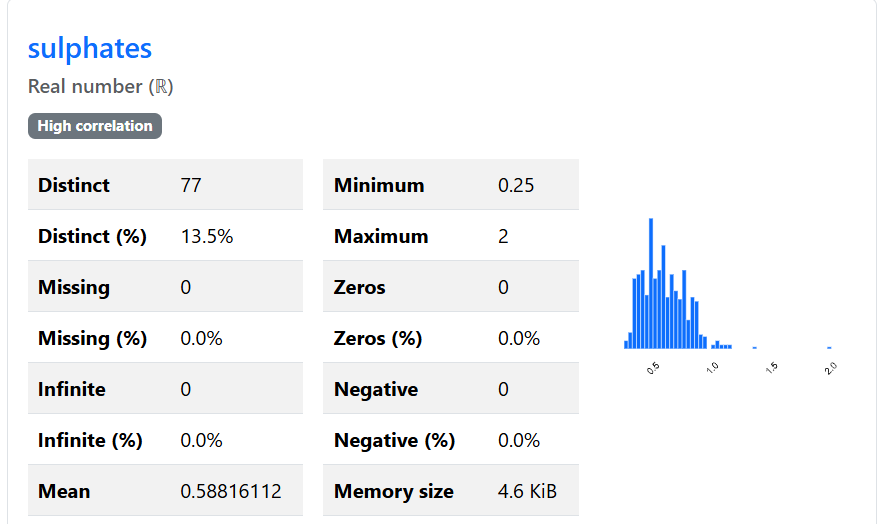
**2 Ans: -** Please find below the model card containing the information related to the properties of the models, this is a summarized table consisting of properties of each base model;

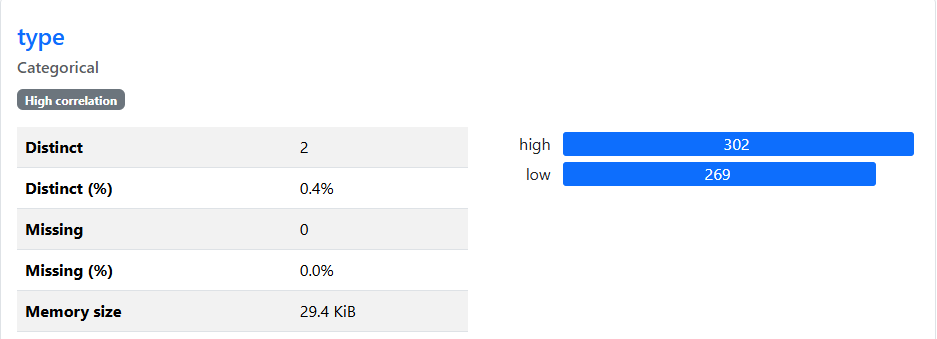
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Property | Decision Tree | Naive Bayes | K-Nearest Neighbor (KNN) | Logistic Regression | Support Vector Machine (SVM) |
| Parametric or Non-Parametric | Non-Parametric | Parametric | Non-Parametric | Parametric | Non-Parametric |
| Input Type | Both | Both | Both | Both | Both |
| Output Type | Discrete (Classification) | Discrete (Classification) | Discrete (Classification) | Discrete (Classification) | Discrete (Classification) |
| Can Handle Missing Values | Yes (with strategies) | Yes (with assumptions) | No (needs preprocessing) | No (needs preprocessing) | No (needs preprocessing) |
| Model Representation | Tree structure | Probabilistic model (Bayes) | Memory-based (data points) | Linear equation | Hyperplane in feature space |
| Model Parameters | Depth, split criteria | Prior and likelihood | Value of k, distance metric | Weights, bias | C (penalty), kernel, margin |
| How to Increase Complexity | Increase depth | Add more features | Increase k, complex metrics | Add interaction terms | Use nonlinear kernels |
| How to Reduce Complexity | Prune tree | Reduce features | Decrease k | Regularization (L1/L2) | Use linear kernel, adjust C |
| Interpretable or Transparent | Yes (highly interpretable) | Partially (with assumptions) | No (hard to interpret) | Yes (moderately interpretable) | No (especially with kernels) |

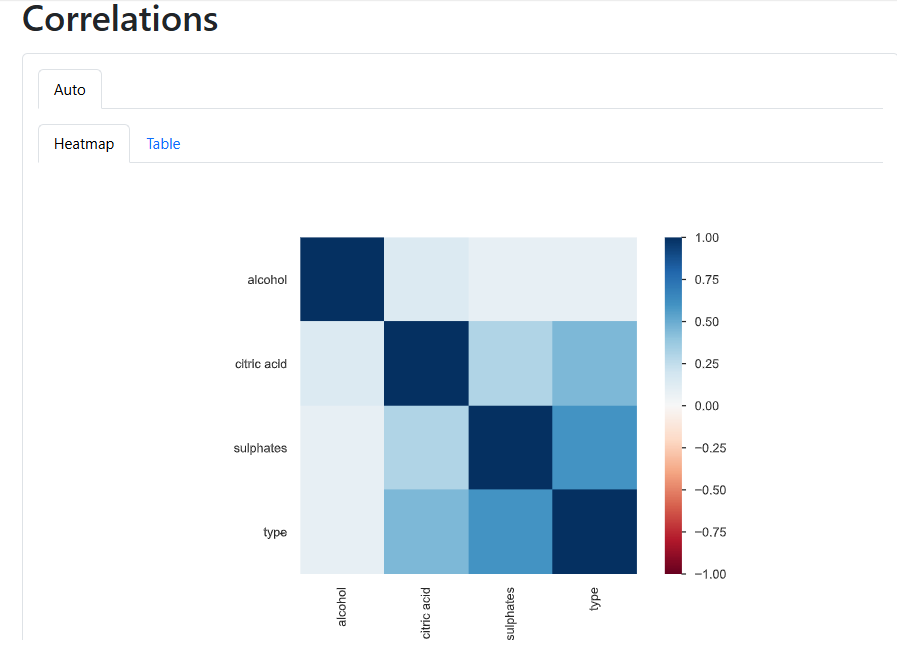
**Part 3. Wine-Tasting Machine**

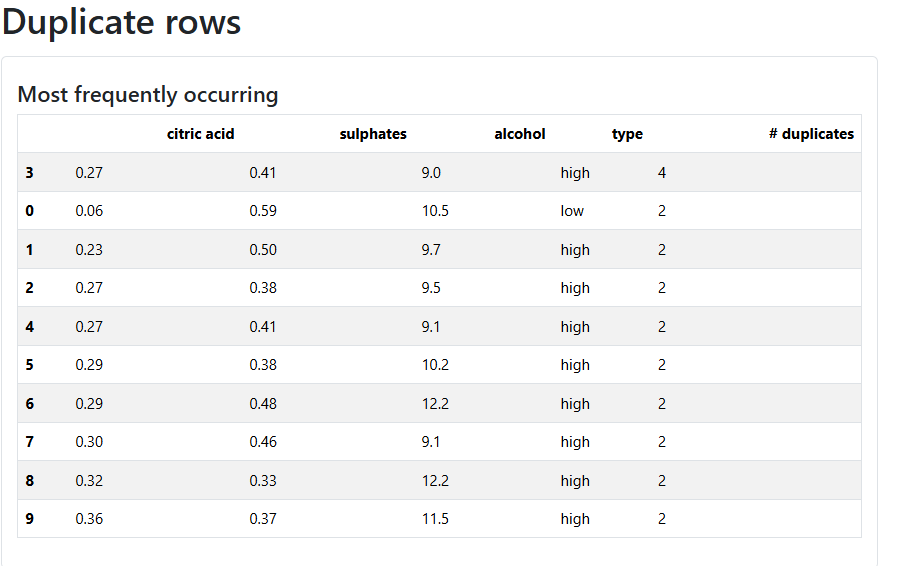
**1.** Read red-wine.csv into Python as a data frame, use a pandas profiling tool (<https://github.com/pandas-profiling/pandas-profiling>) to create an HTML file.

**Part 3 1 Ans: -**



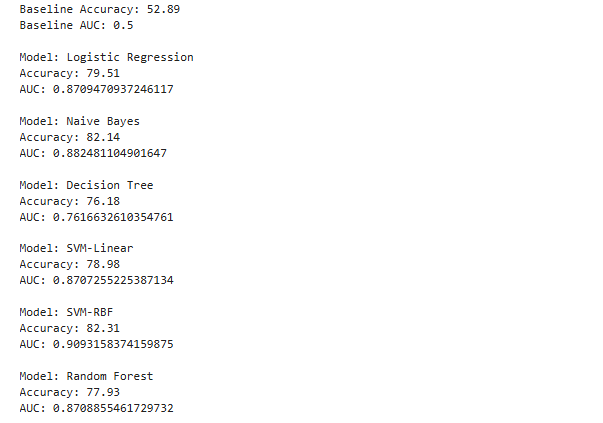






**2.** Fit a model using each of the following methods and report the performance metrics of 10-fold cross-validation using red-wine.csv as the training set.

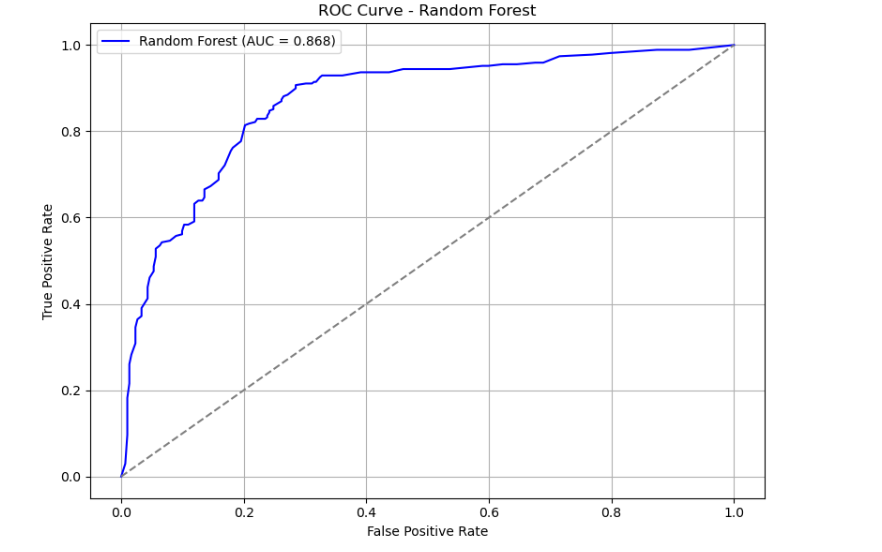
**Part 3 2 Ans: -** Below is the table showing the values of Accuracy and AUC of various models with 10-fold cross-validation;



|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Model | Baseline | Logistic Regression | Naive Bayes | Decision Tree | SVM - Linear | SVM - RBF | Random Forest |
| AUC | 0.50 | 0.87 | 0.88248 | 0.76021 | 0.87130 | 0.91059 | 0.86945 |
| Accuracy | 52.89% | 79.51% | 82.14% | 75.83% | 78.98% | 82.31% | 80.04% |

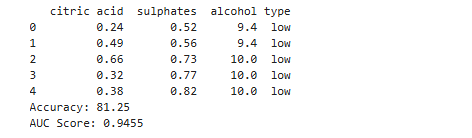
**3**. Plot the ROC curve of the Random Forest classifier from the Python package, and paste a screenshot of your ROC.

**Part 3 3 Ans: -**



**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.

**Part 3 4 Ans**:



**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?

**Part 3 5 Ans: -** I would advise utilizing a model based on either logistic regression or decision trees.

These models are far simpler to comprehend and describe. For instance:

Wine specialists can better understand how the algorithm is making decisions by using a Decision Tree, which displays a clear set of "if-then" rules (such as "if alcohol > 10%, then likely red").

Experts can determine precisely which chemical characteristics—such as alcohol or sulphates—are most affecting the forecast by using logistic regression, which assigns straightforward weights to each feature.

In certain situations, models such as SVM or Random Forest may be more accurate, but they function more like "black boxes"—they make decisions without providing much context.

Therefore, simpler and more transparent models like Decision Trees and Logistic Regression are the better option for wine-tasting experts who wish to comprehend the logic behind predictions.