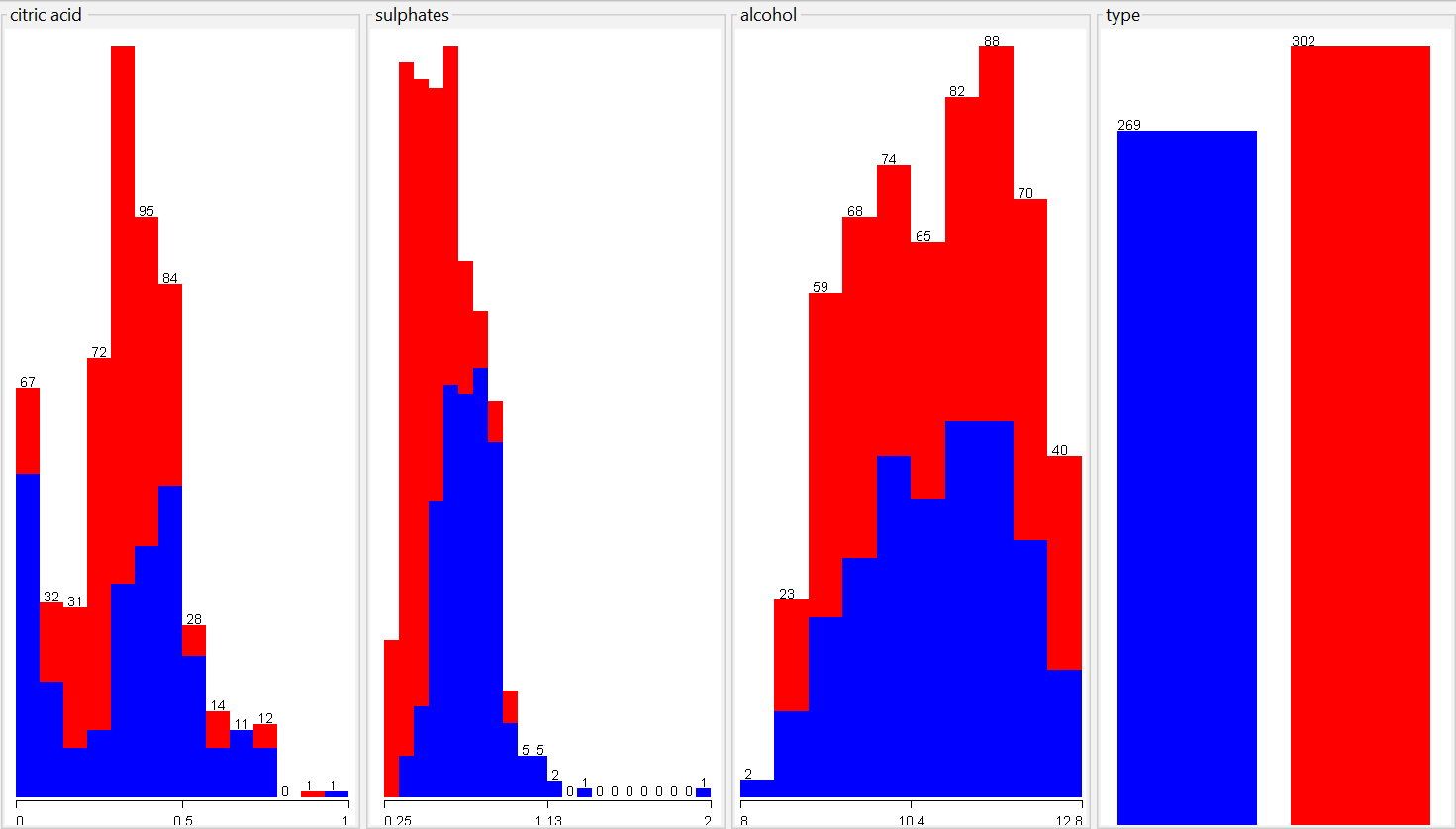
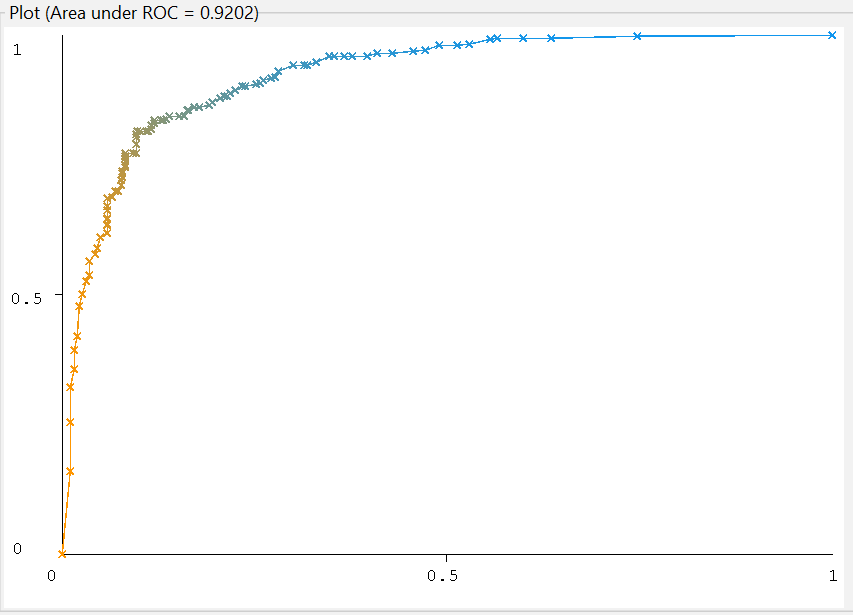
**WEKA TASK:**

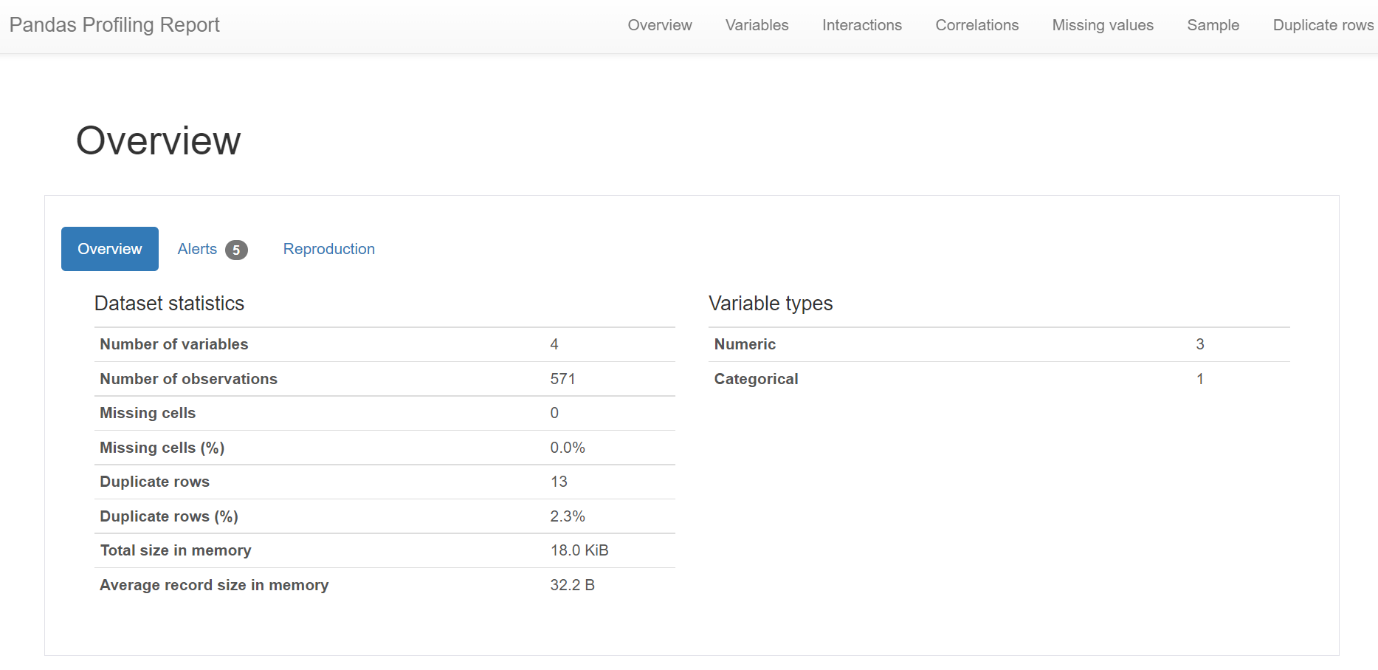
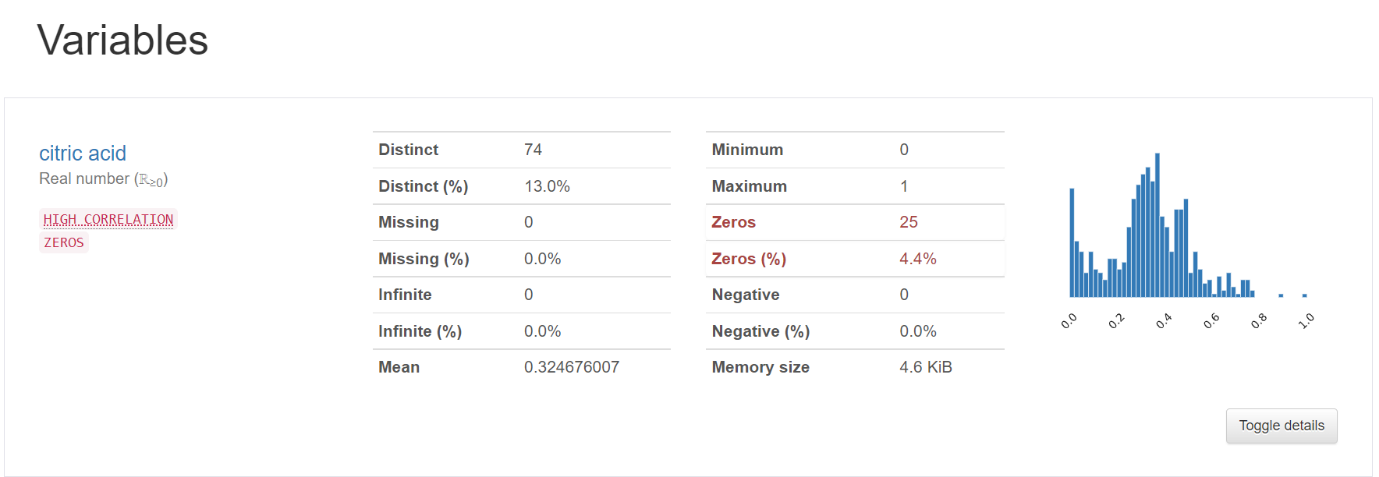
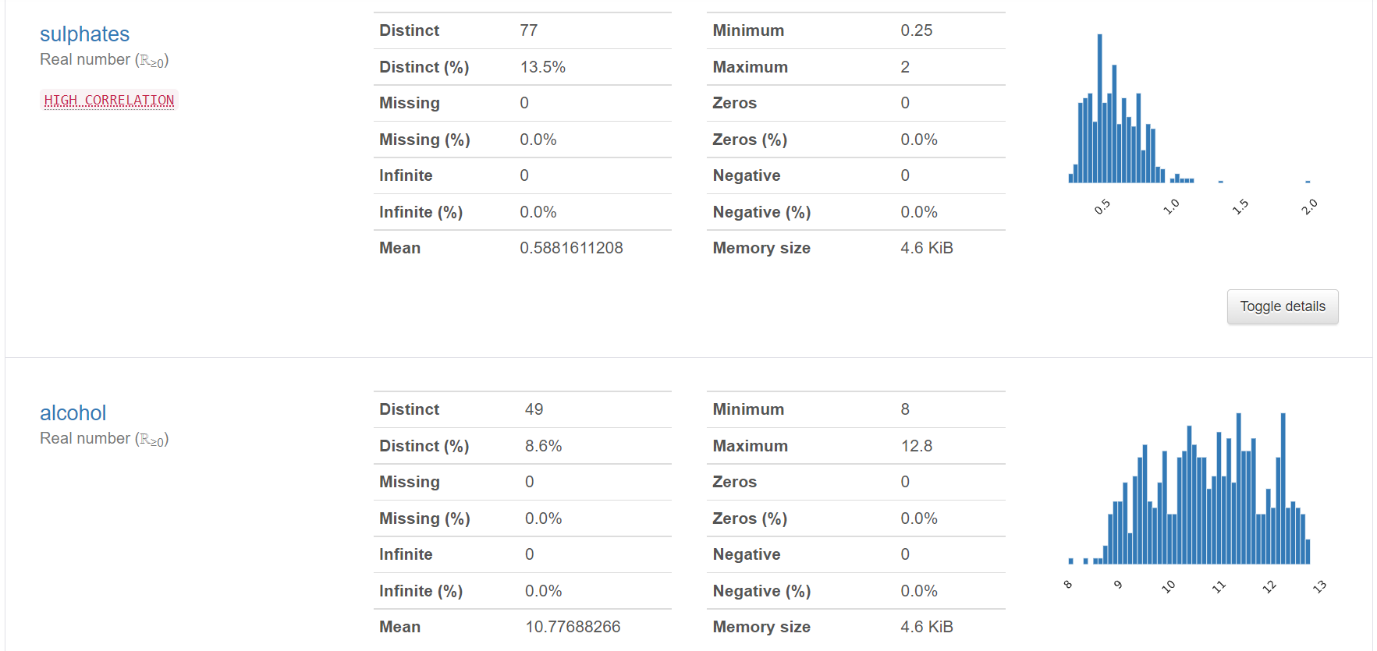
* 1. Conditional distribution of all input variables with respect to output variable.
  2. Sulphates would be more predictable of wine quality because from the distribution, the lower the value of sulphates, the greater the quality of wine. While, for alcohol which is being almost normally distributed, the higher value or lower value have almost same concentration of wine.
  3. My answer is backed by the model. The sulphates have higher predictive power of wine quality than alcohol as AUC-ROC for both of them are 0.867 and 0.436 respectively.

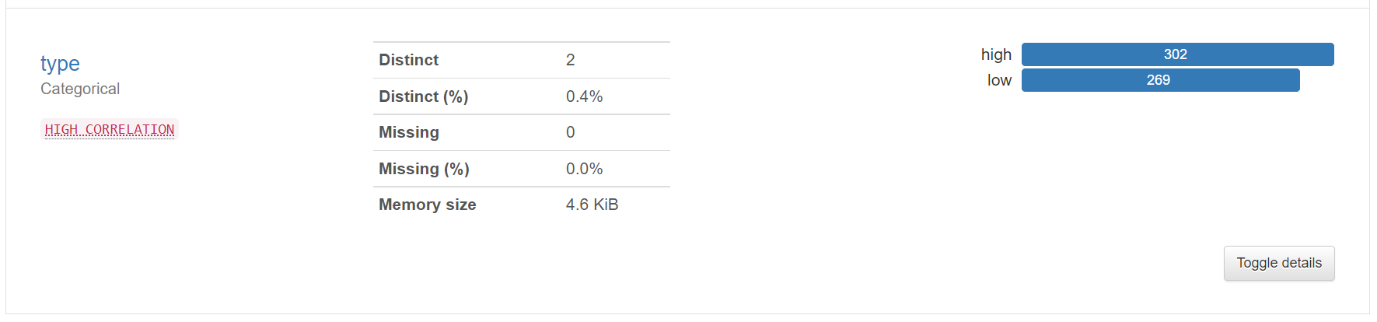
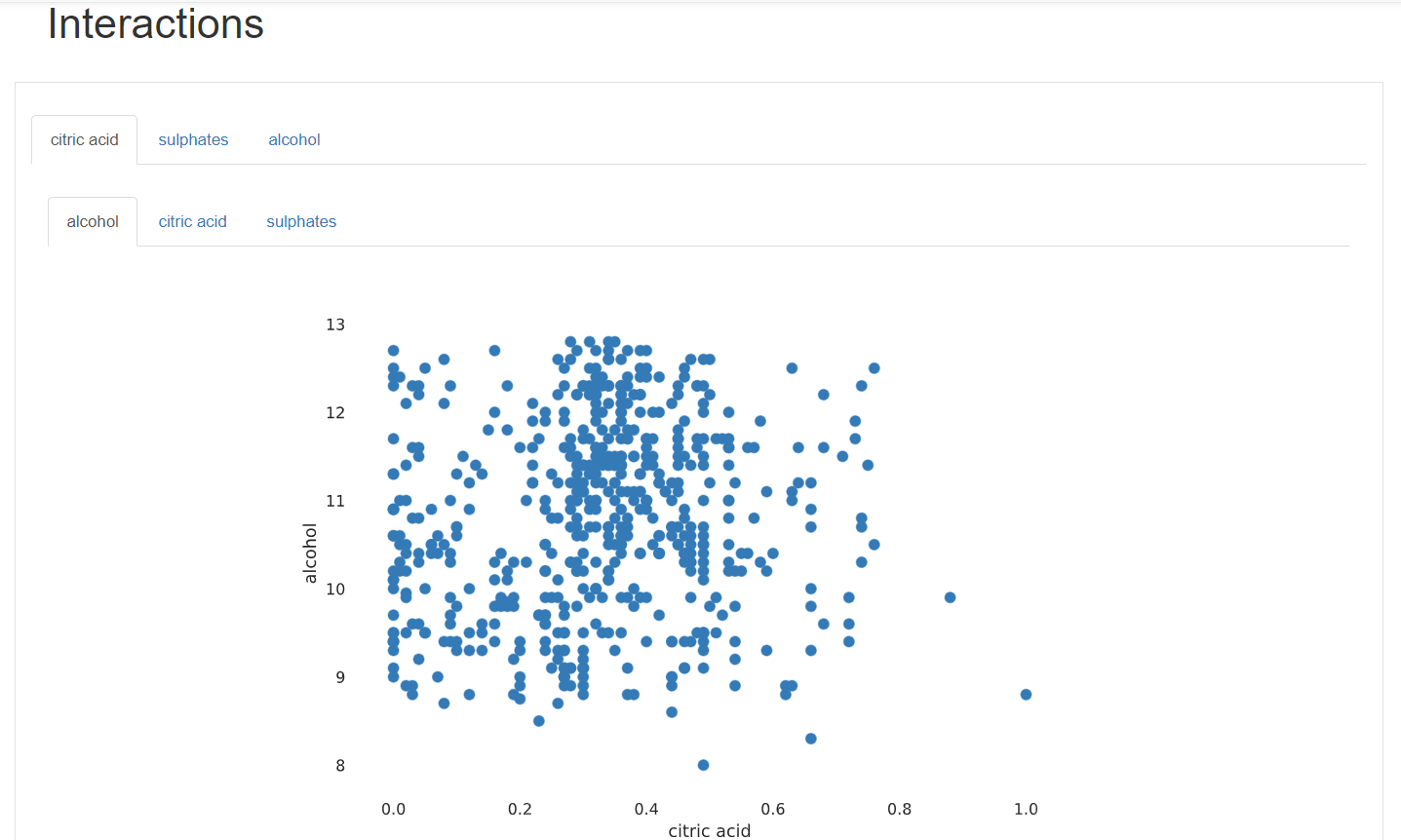
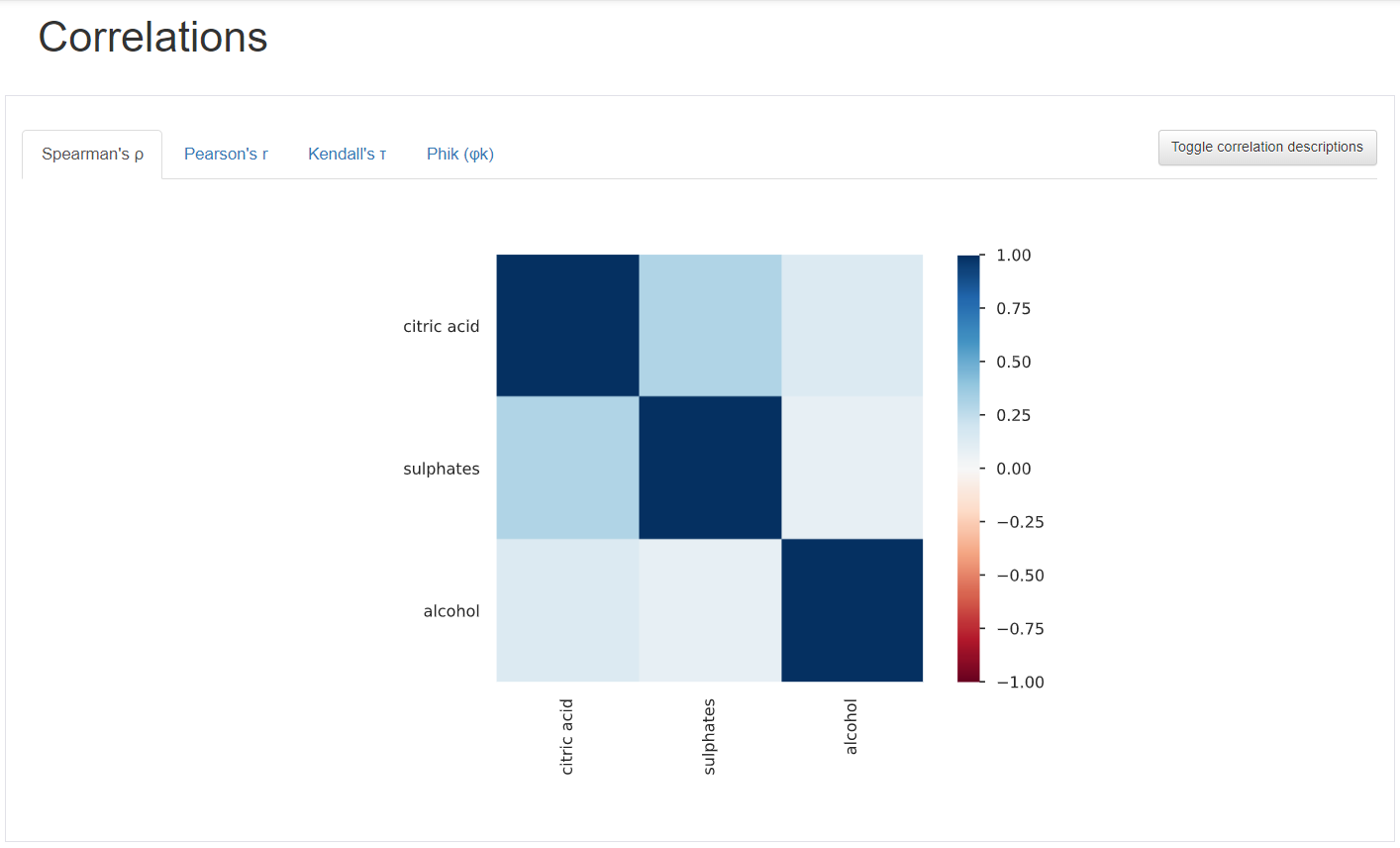
1. Performance metrics for 10-fold CV using red-wine.csv as training set.

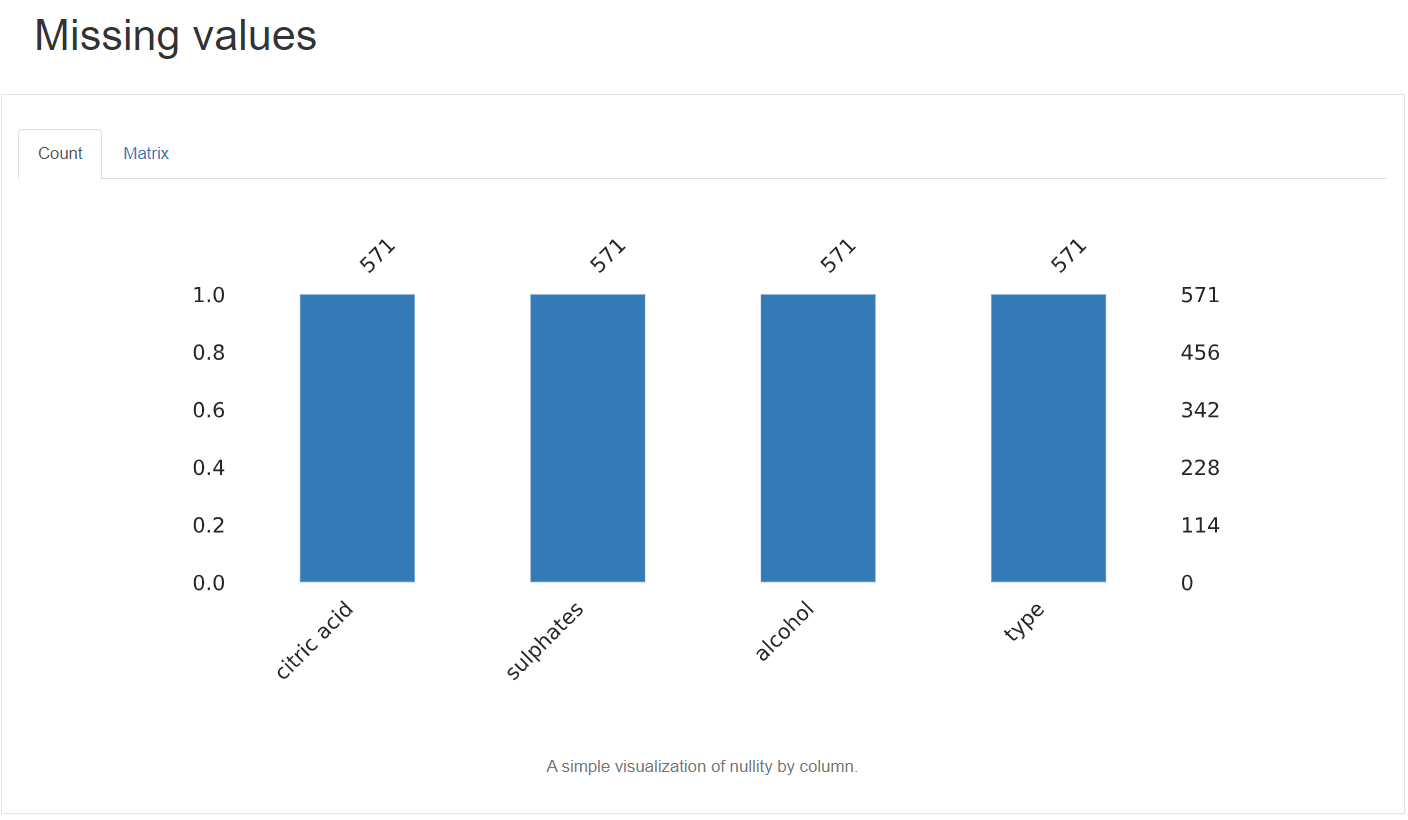
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Model | ZeroR | OneR | LR | NB | DT | SVM | RF |
| AUC | 0.496 | 0.785 | 0.873 | 0.89 | 0.877 | 0.781 | 0.92 |
| Accuracy | 0.529 | 0.785 | 0.793 | 0.825 | 0.842 | 0.783 | 0.853 |

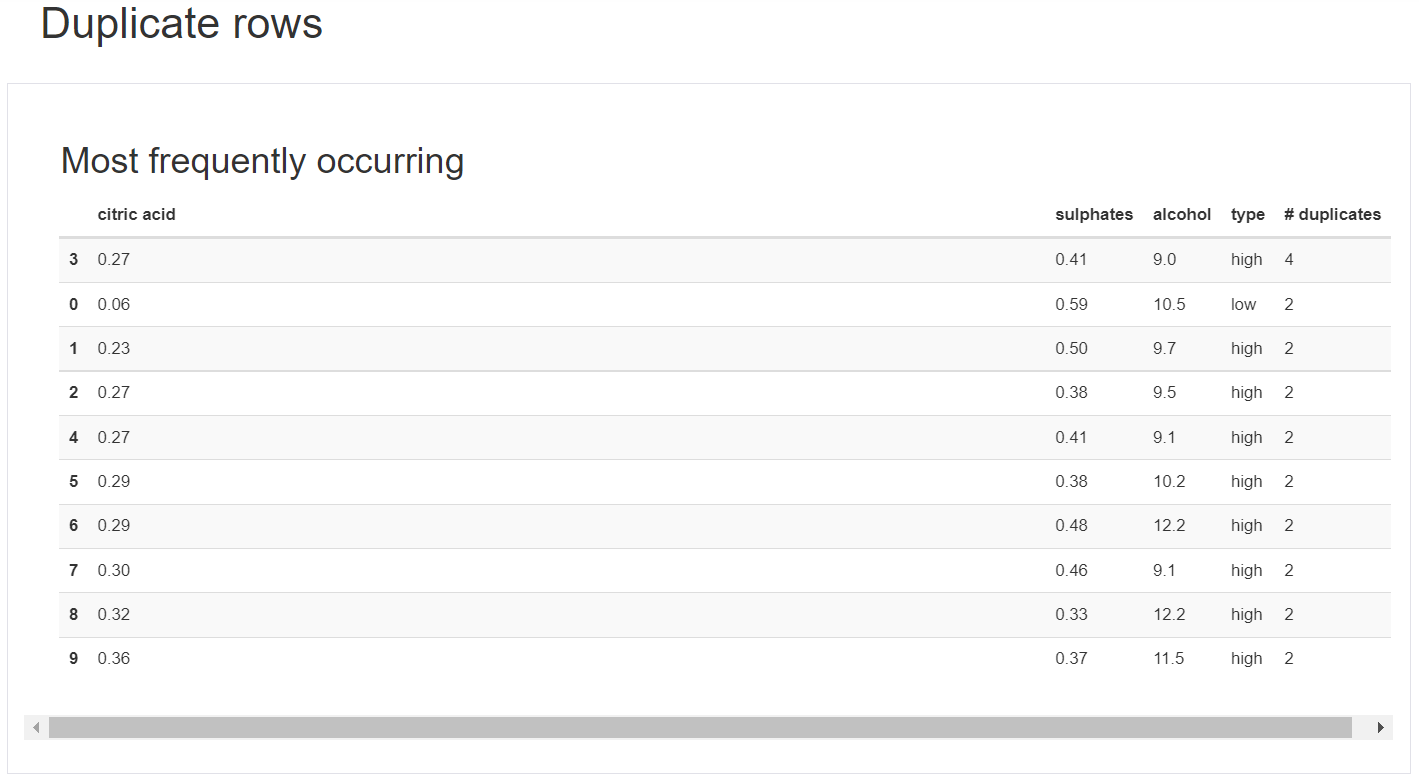
1. ROC curve for best performing model (RF) in terms of AUC score. The AUC-ROC value is **0.92** meaning the model is performing very good and doing a good job in predicting the low-quality wine as low-quality and high-quality wine as high-quality.
2. AUC score for the above RF model when run on white-wine.csv is **0.956**. The performances of the model when trained on red-wine dataset is comparable with the performance of the same model when tested in white-wine dataset meaning the model have learnt the very close approximation to the underlying function that maps the input to the output.

**PYTHON TASK:**

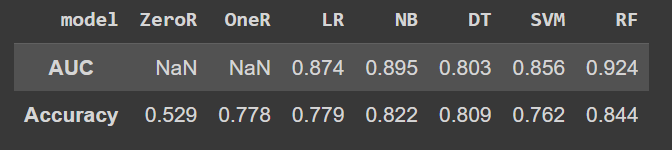
1. Profiling of red-wine.csv file.

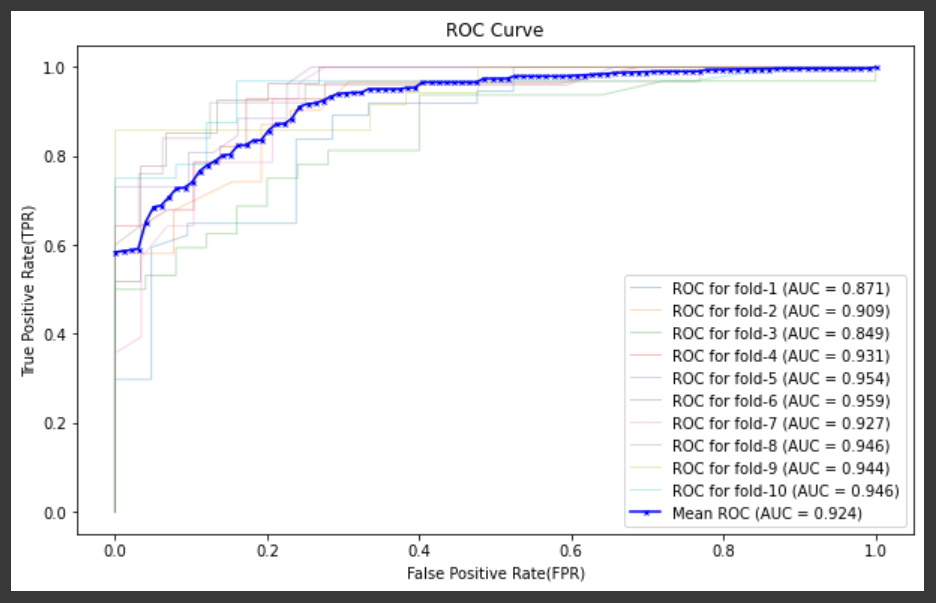






1. Performance metrics for 10-fold CV using red-wine.csv as training set and in python.

****

1. ROC Curve for RandomForest Model.
2. The Best performing model is RandomForest (RF) and the AUC-ROC score obtained when tested on white-wine is **0.978.** The train and test scores are comparable so the model almost learnt the necessary parameters properly.
3. If all models have comparable performance, then I would choose the model with least complexity like OneR, Logistic because, those models are easiest to explain to non-technical people like wine-tasting experts.

References:

1. <https://github.com/rasbt/mlxtend/blob/master/mlxtend/classifier/oner.py#L19>
2. <http://rasbt.github.io/mlxtend/user_guide/classifier/OneRClassifier/>
3. <https://scikit-learn.org/stable/auto_examples/model_selection/plot_roc_crossval.html>