

# PREDICTING RED WINE QUALITY: A MACHINE LEARNING APPROACH

**Abstract:** - The project aims to build a machine learning model for classifying red wine quality as good or not good based on physiochemical features. Various classification algorithms and techniques are employed to achieve the highest model accuracy.

**Problem Statement:** - ML modelling with different classification algorithm to build model with highest accuracy which in turns lead to predicting quality of wine in term of good or not good. With help of EDA to determine which features are the most indicative of a good quality wine. The goal is to predict wine quality using the Red Wine Quality Data Set from Kaggle's UCI machine learning repository. This involves creating a binary classification task where wines with a quality score of 7 or higher are classified as good, while others are classified as not good.

**Data Set Used:** - The Red Wine Quality Data Set contains 12 variables recorded for 1,599 observations. These variables include fixed acidity, volatile acidity, citric acid, residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide, density, pH, sulphates, alcohol, and quality (output variable).

### Tools Used: -

1. **Python Programming Language:** - Python is a high-level, versatile programming language known for its simplicity and readability. It's widely used in various fields, including data science, web development, automation, and scientific computing.
2. **Google Colab:** - Google Colab is a cloud-based platform that allows users to write and execute Python code in a collaborative environment.

## MACHINE LEARNING PROJECT

3. **Pandas**: - Pandas is a Python library used for data manipulation and analysis. It provides data structures like DataFrame and Series, along with functions for reading and writing data, handling missing values, grouping, merging, and more.
4. **NumPy**: - NumPy is a fundamental library for numerical operations in Python. It introduces the ndarray (N-dimensional array) data structure, which allows efficient handling of large datasets and provides mathematical functions for array operations.
5. **Scikit-learn**: - Scikit-learn is a machine learning library in Python that provides a wide range of algorithms for classification, regression, clustering, dimensionality reduction, and more. It also offers tools for model evaluation, preprocessing, and hyperparameter tuning.
6. **Matplotlib and Seaborn**: - Matplotlib is a plotting library in Python used for creating static, interactive, and publication-quality visualizations. Seaborn is built on top of Matplotlib and provides a higher-level interface for creating statistical graphics, such as heatmaps, violin plots, and pair plots, with less code.

### Algorithms Used:

1. **Logistic Regression**: - Logistic Regression is a classification algorithm used for binary and multiclass classification tasks. It models the probability of a binary outcome based on one or more predictor variables using a logistic function.  
**Accuracy score = 0.9128630705394191**
2. **K-Nearest Neighbors (KNN)**: - K-Nearest Neighbors is a simple and intuitive classification algorithm. It assigns a class label to a data point based on the majority class among its k nearest neighbors in the feature space.  
**Accuracy score = 0.9107883817427386**
3. **Decision Tree**: - Decision Tree is a tree-based algorithm that recursively splits the data into subsets based on the most significant feature at each node. It's easy to interpret and can handle both classification and regression tasks.  
**Accuracy score= 0.8755186721991701**

## MACHINE LEARNING PROJECT

4. **Random Forest:** - Random Forest is an ensemble learning method that constructs multiple decision trees during training and outputs the class that is the mode of the classes predicted by individual trees. It reduces overfitting and improves accuracy.  
**Accuracy score** = 0.9253112033195021
5. **Support Vector Machine (SVM):** - Support Vector Machine is a powerful classification algorithm that finds the optimal hyperplane to separate classes in the feature space. It works well for both linearly separable and non-linearly separable data using kernel functions.  
**Accuracy score** = 0.9128630705394191
6. **Naive Bayes:** - Naive Bayes is a probabilistic classifier based on Bayes' theorem with an assumption of independence among features. Despite its simplicity, it performs well in many real-world applications and is particularly effective for text classification tasks.  
**Accuracy score** = 0.9087136929460581
7. **AdaBoost:** - AdaBoost (Adaptive Boosting) is an ensemble learning technique that combines multiple weak classifiers into a strong classifier. It assigns higher weights to misclassified instances in each iteration, focusing on difficult-to-classify samples.  
**Accuracy score** = 0.8900414937759336
8. **Gradient Boosting:** - Gradient Boosting is another ensemble learning method that builds a strong model by sequentially adding weak learners (typically decision trees) and correcting errors made by previous models. It's known for its high predictive accuracy.  
**Accuracy score** = 0.9004149377593361
9. **Bagging (Ensemble Learning):** - Bagging (Bootstrap Aggregating) is an ensemble learning technique that combines predictions from multiple models trained on bootstrap samples of the training data. It reduces variance and improves stability by averaging predictions.  
**Accuracy score** = 0.8921161825726142

### Methodology:

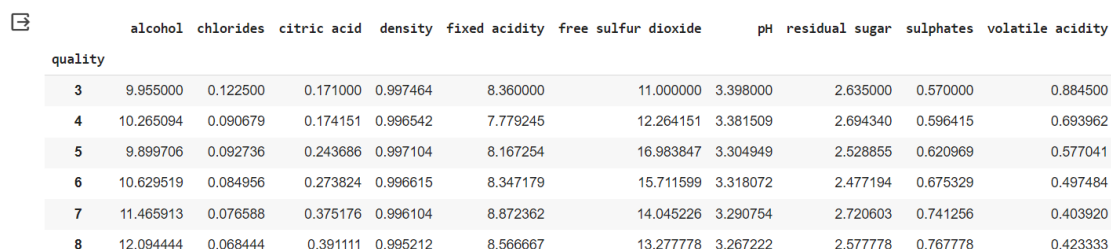
# MACHINE LEARNING PROJECT

1. Exploratory Data Analysis (EDA)
2. Feature Selection
3. Outliers Detection using Interquartile Range (IQR)
4. Outliers Removal using Z-score Method
5. Skewness Detection and Transformation
6. Correlation Analysis
7. Standard Scaling for normalization
8. Principal Component Analysis (PCA) for dimensionality reduction
9. Machine Learning Model Building
10. Cross-Validation for model evaluation
11. Hyperparameter Tuning using GridSearchCV

**Results:** - The best-performing model achieved an accuracy score of 0.917, indicating a high level of predictive accuracy in classifying wine quality.

**Screenshots: -**

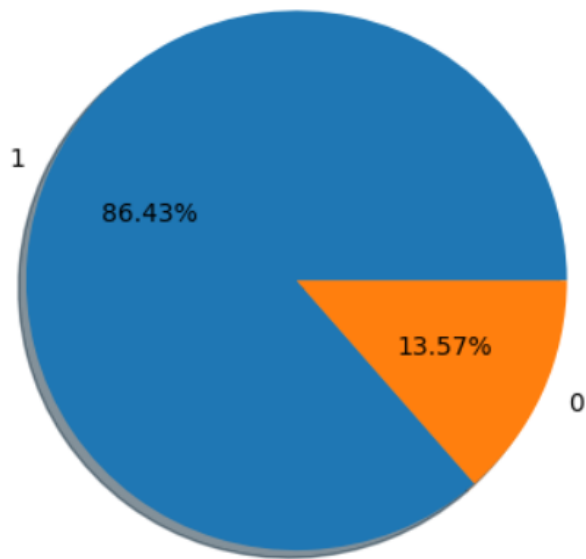
**Dataset –**



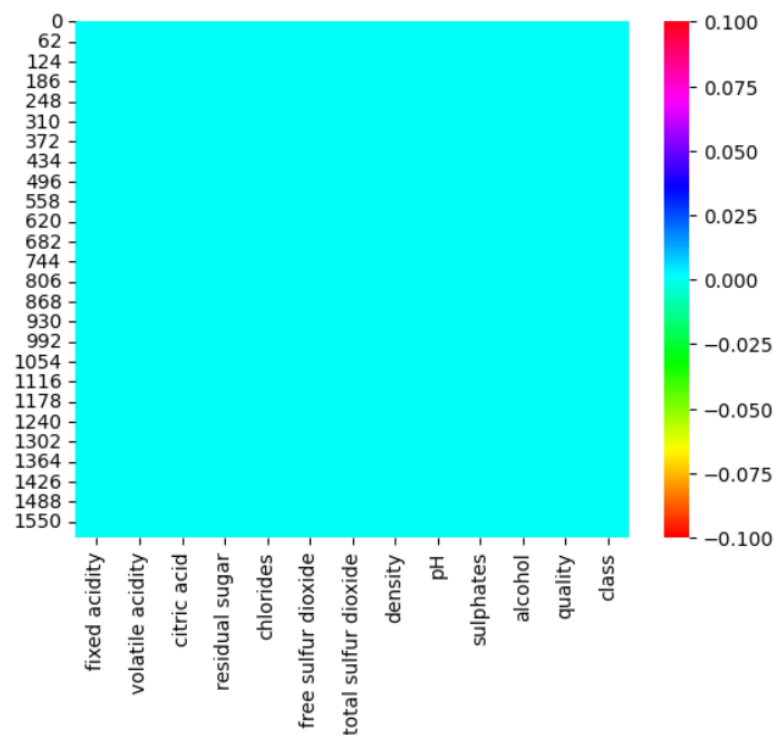
	alcohol	chlorides	citric acid	density	fixed acidity	free sulfur dioxide	pH	residual sugar	sulphates	volatile acidity
quality										
3	9.955000	0.122500	0.171000	0.997464	8.360000	11.000000	3.398000	2.635000	0.570000	0.884500
4	10.265094	0.090679	0.174151	0.996542	7.779245	12.264151	3.381509	2.694340	0.596415	0.693962
5	9.899706	0.092736	0.243686	0.997104	8.167254	16.983847	3.304949	2.528855	0.620969	0.577041
6	10.629519	0.084956	0.273824	0.996615	8.347179	15.711599	3.318072	2.477194	0.675329	0.497484
7	11.465913	0.076588	0.375176	0.996104	8.872362	14.045226	3.290754	2.720603	0.741256	0.403920
8	12.094444	0.068444	0.391111	0.995212	8.566667	13.277778	3.267222	2.577778	0.767778	0.423333

**Mean feature values based on class**

## MACHINE LEARNING PROJECT

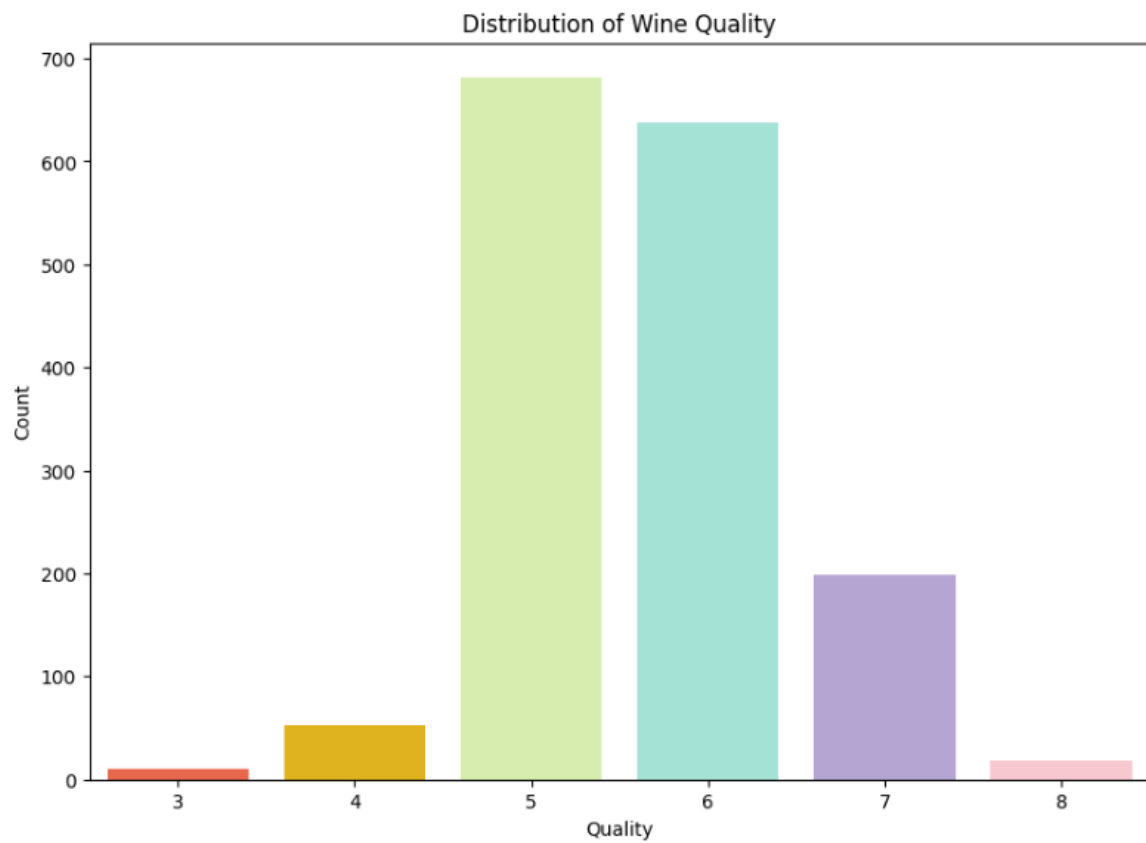


### Checking null value or missing data

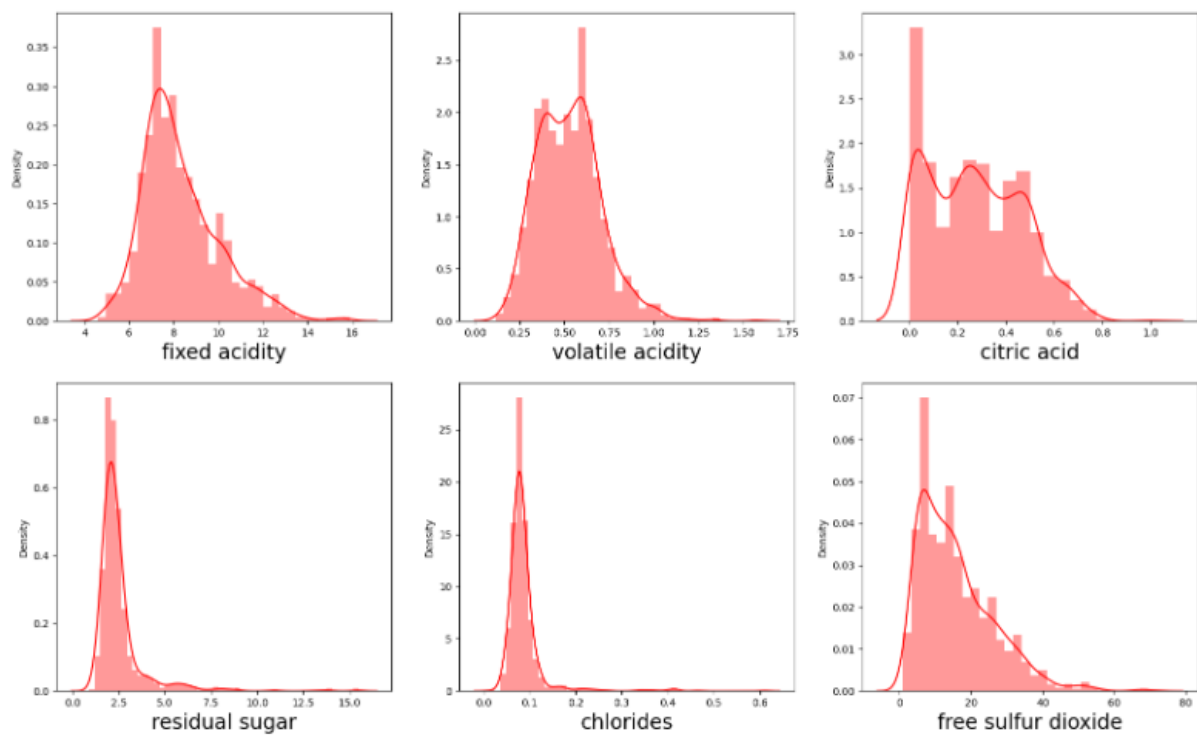


### Exploratory Data Analysis

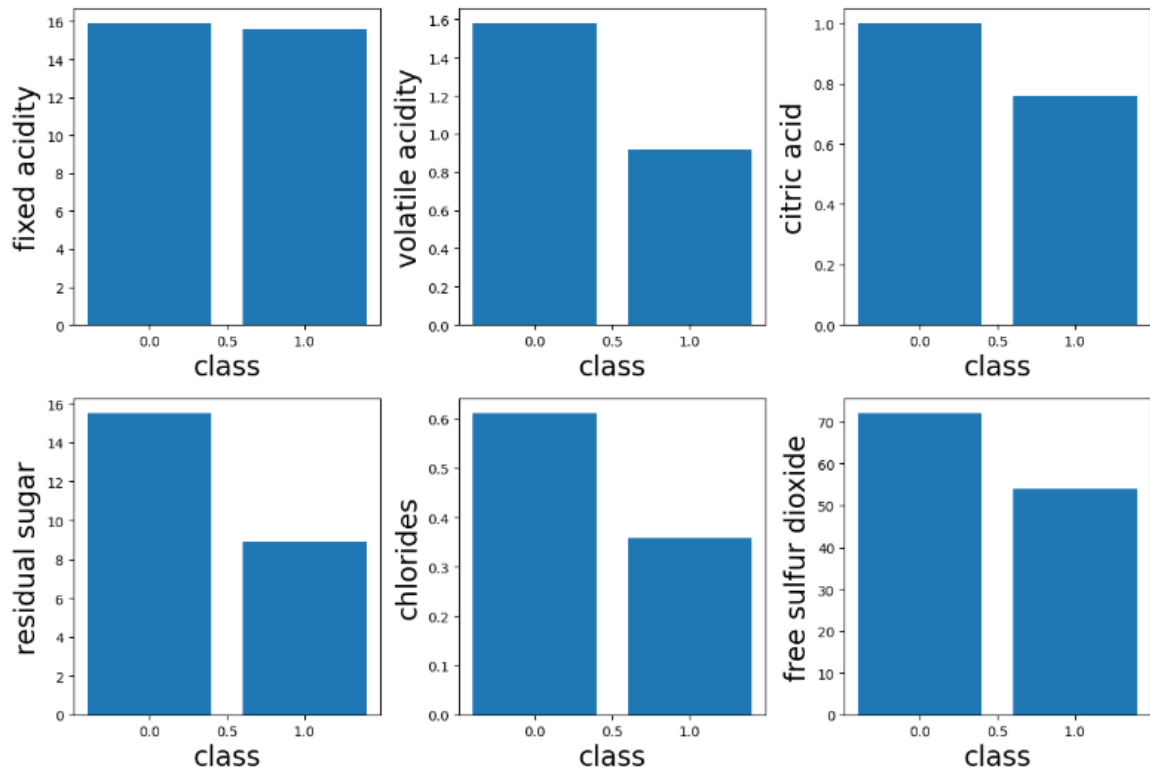
## MACHINE LEARNING PROJECT



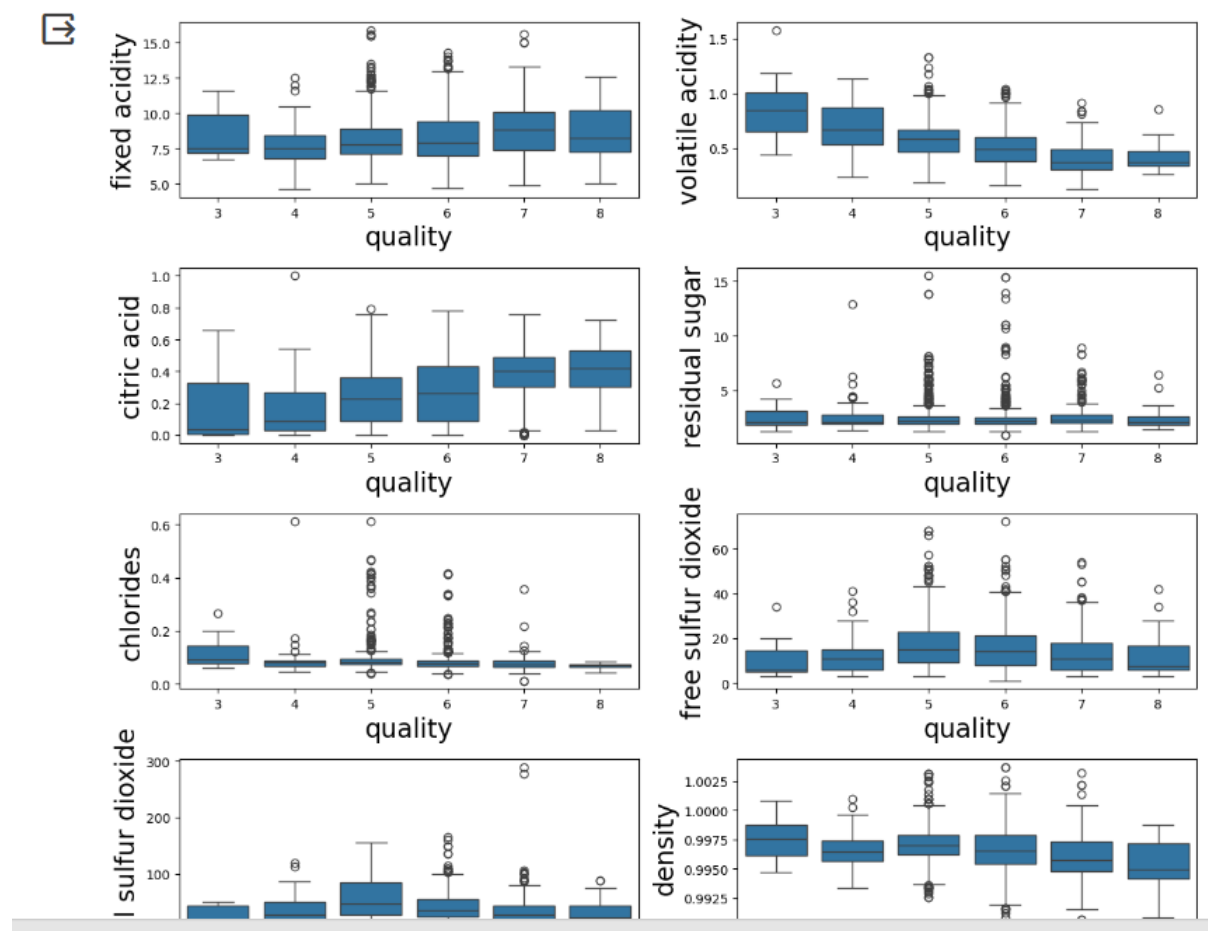
## Distribution of features :



## MACHINE LEARNING PROJECT

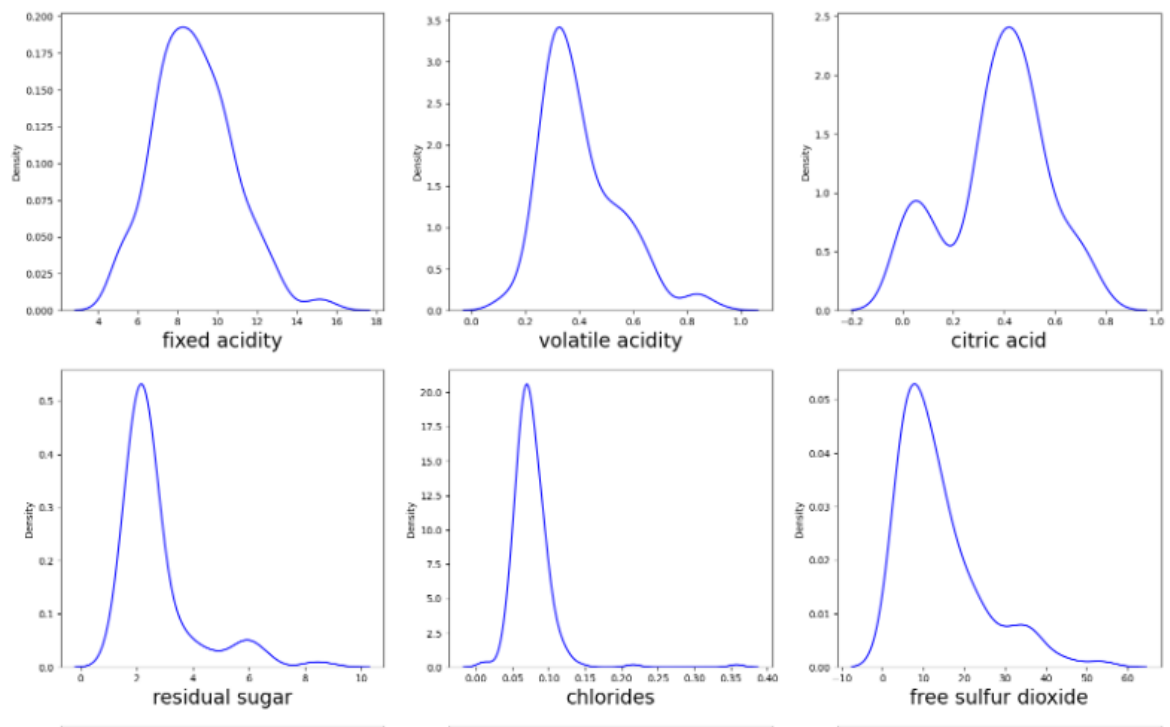


## BOXPLOT –



# MACHINE LEARNING PROJECT

## KDEPLOT-



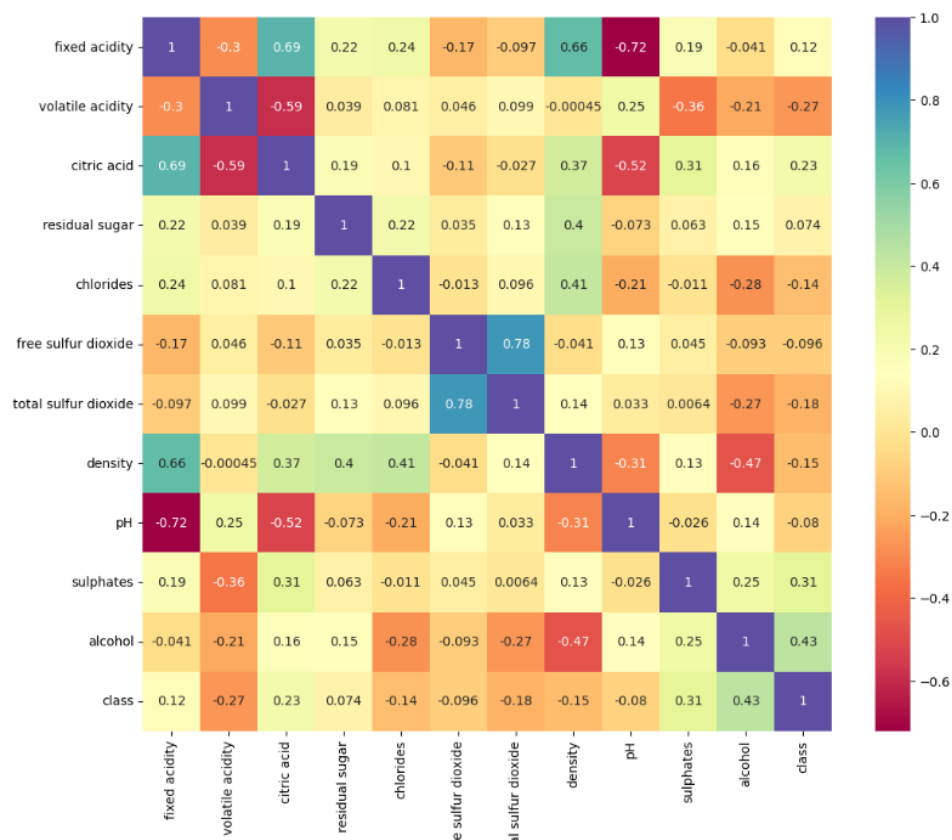
## PAIRPLOT –





# MACHINE LEARNING PROJECT

## HEATMAP –

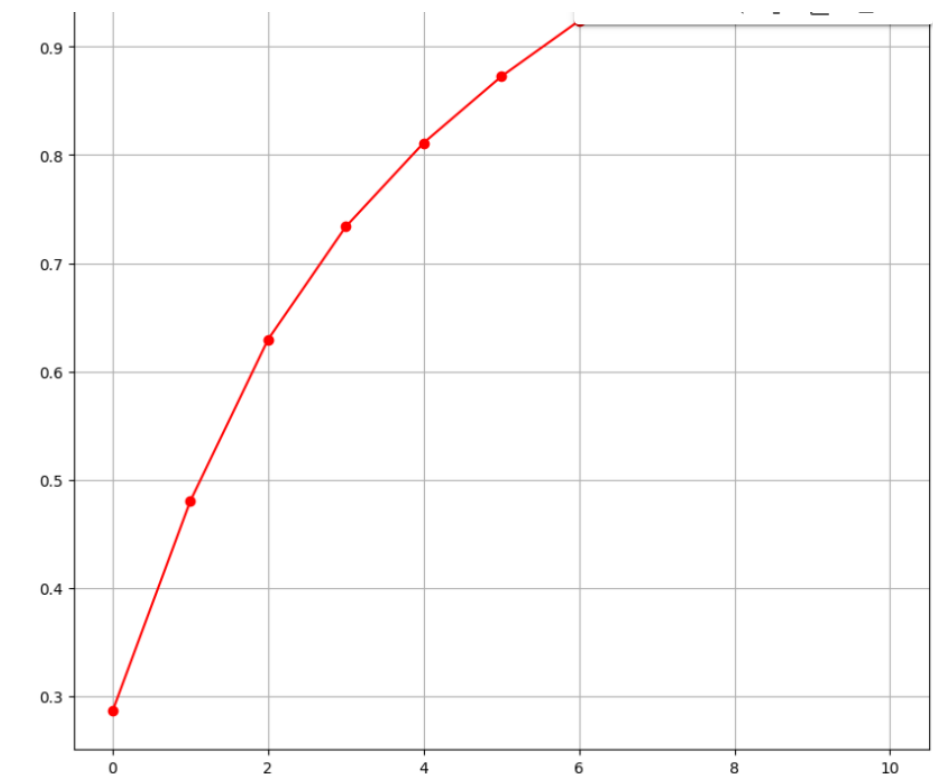


## Visualizing correlation of feature columns with label column

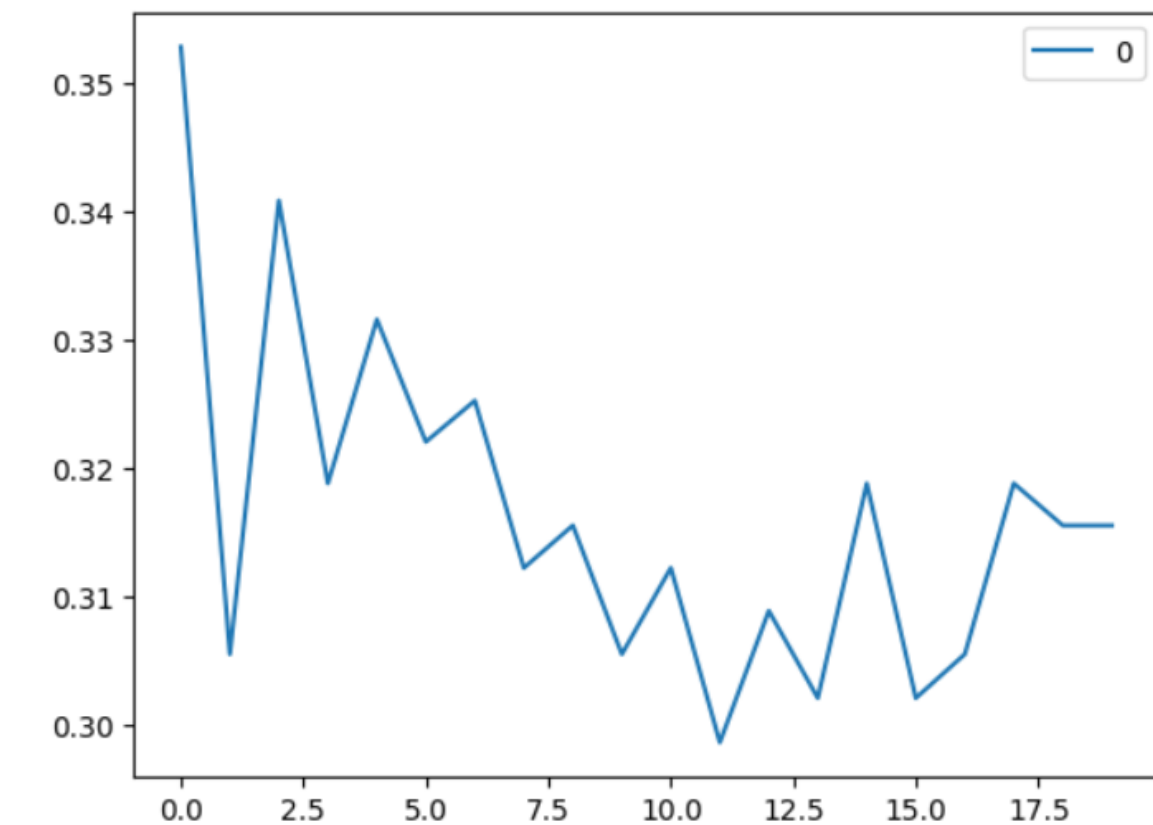


## MACHINE LEARNING PROJECT

### PCA CURVE

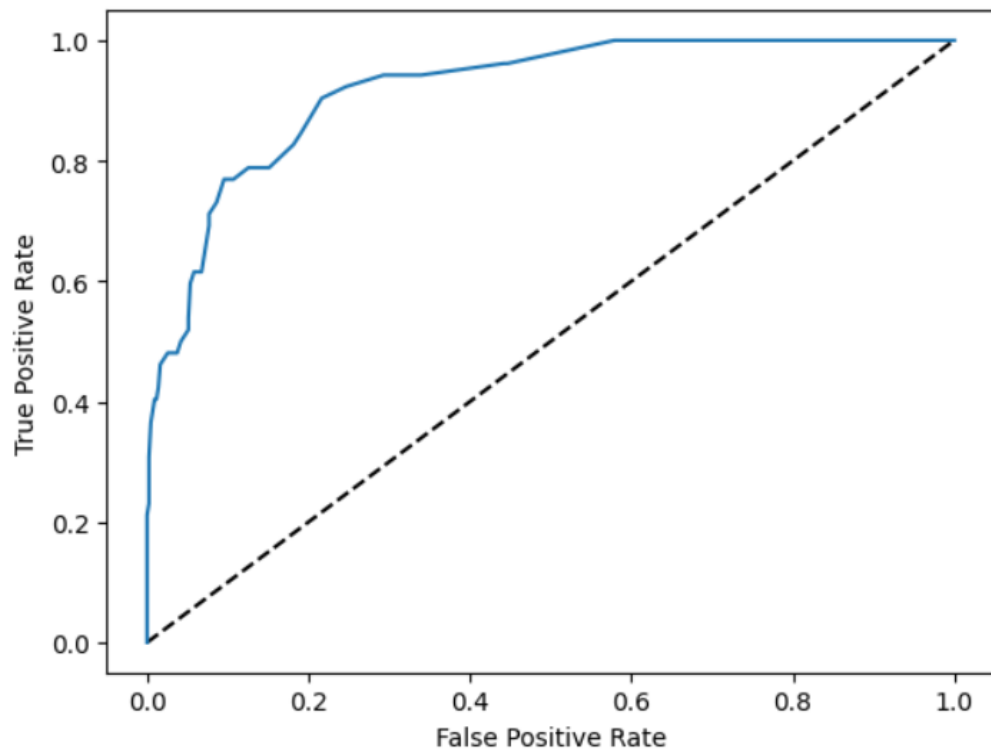


plotting the rmse value for the k value



## MACHINE LEARNING PROJECT

### ROC CURVE



Auc score :  
0.7045617173524151