Wine Quality Prediction Documentation

1 Introduction

This project predicts wine quality using machine learning based on physicochemical properties like acidity alcohol pH density and sulphates. The objective is to identify which features affect wine quality the most and to build a reliable classifier.

2 Libraries Used

These are the exact libraries used in the notebook:

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

from sklearn.model_selection import train_test_split

from sklearn.ensemble import RandomForestClassifier

from sklearn.metrics import accuracy_score

3 Dataset Overview

Contains only numerical features.

Target variable is quality.

Quality ranges from 0 to 10.

No categorical encoding required.

4 Data Preprocessing

Checked for missing values.

Verified all columns numeric.

Identified quality as target.

Cleaned and prepared features.

5 Exploratory Data Analysis

Alcohol shows strong positive influence.

Volatile acidity decreases quality.

Sulphates have mild positive effect.

Citric acid contributes positively.

Density has weak negative effect.

6 Correlation Analysis

Alcohol highest positive correlation.

Volatile acidity strong negative correlation.

Sulphates and citric acid moderate positive.

Density slight negative impact.

pH and chlorides minimal correlation.

7 Feature Selection

alcohol

volatile acidity

sulphates

citric acid

density

8 Model Used

RandomForestClassifier used due to high accuracy and ability to handle nonlinear patterns.

9 Model Training

Data split using train_test_split.

Model trained on training set.

Predictions made on test set.

10 Model Evaluation

Accuracy_score used for evaluation.

Random Forest performed best.

11 Key Results

Alcohol strongest indicator.

Volatile acidity reduces quality.

Sulphates and citric acid improve quality.

Random Forest gave highest performance.

12 Conclusion

Machine learning successfully predicts wine quality.

Random Forest effective and aligned with wine chemistry.

13 Future Improvements

Try Gradient Boosting or XGBoost.

Use cross validation.

Convert scores into categorical classes.

Tune hyperparameters.

Deploy using Flask or Streamlit.

14 Author

Satyam Gajjar