**Wine Quality Classification Project**

**Objective**:  
The main objective of this project is to classify red wine into two quality categories: 'good' and 'bad', based on a set of physicochemical properties.

**Dataset**:  
The dataset used is the "Red Wine Quality" dataset. Each entry in the dataset represents a specific wine sample and has various physicochemical attributes like fixed acidity, volatile acidity, citric acid, etc. There is also a 'quality' attribute that rates the wine on a scale of 0 to 10.

**Methodology**:

1. **Data Pre-processing**:
   * The 'quality' attribute, originally a score from 0 to 10, is transformed into a binary classification task. Wines with a quality score of 7 or above are label as 'good' (represented by 1) and the rest are labelled as 'bad' (represented by 0).
   * The dataset is split into training and testing sets, ensuring that the model can be evaluated on unseen data.
2. **Model Selection and Pipeline Creation**:
   * A Random Forest Classifier, an ensemble learning method, is chosen due to its ability to handle complex datasets and give importance scores for features.
   * A pipeline is constructed to streamline the modelling process. This pipeline first scales the features (using **Standard Scalar**) to ensure that all of them are on a similar scale, and then applies the Random Forest Classifier.
3. **Hyper parameter Tuning**:
   * Grid Search CV is employed to exhaustively search over a specified parameter grid. This ensures that the model uses the best hyper parameters for training and can potentially improve its predictive capabilities.
   * Parameters tuned include the number of trees (**n\_ estimators**), the maximum depth of the trees (**max\_ depth**), the minimum number of samples required to split (**min\_ samples\_ split**), and the minimum number of samples required at the leaf level (**min\_ samples\_ leaf**).
4. **Model Evaluation**:
   * Cross-validation is used to evaluate the model's performance on the training data. This technique divides the training data into 'folds' and ensures that the model is tested on different subsets, providing a more robust evaluation metric.
   * The model's performance on the test set is then evaluated to get an understanding of how it performs on completely unseen data.
5. **Results**:
   * The optimal hyper parameters for the Random Forest Classifier are displayed.
   * The average accuracy from cross-validation provides an insight into the model's expected performance.
   * A classification report on the test set displays key metrics such as precision, recall, and f1-score for both the 'good' and 'bad' wine classes.

**Conclusion**:  
The Random Forest Classifier, combined with pre-processing steps and hyper parameter optimization, provides a robust method to classify wine quality based on physicochemical properties. Future steps could involve trying different machine learning models, feature engineering, or gathering more data to improve predictive accuracy.