**About Dataset**

**Context**

The two datasets are related to red and white variants of the Portuguese "Vinho Verde" wine. For more details, consult the reference [Cortez et al., 2009]. Due to privacy and logistic issues, only physicochemical (inputs) and sensory (the output) variables are available (e.g. there is no data about grape types, wine brand, wine selling price, etc.).

These datasets can be viewed as classification or regression tasks. The classes are ordered and not balanced (e.g. there are much more normal wines than excellent or poor ones).

This dataset is also available from the UCI machine learning repository, [*https://archive.ics.uci.edu/ml/datasets/wine+quality*](https://archive.ics.uci.edu/ml/datasets/wine+quality) , I just shared it to kaggle for convenience. (If I am mistaken and the public license type disallowed me from doing so, I will take this down if requested.)

### Content

For more information, read [Cortez et al., 2009].  
  
Input variables (based on physicochemical tests):  
  
1 - fixed acidity  
  
2 - volatile acidity  
  
3 - citric acid  
  
4 - residual sugar  
  
5 - chlorides  
  
6 - free sulfur dioxide  
  
7 - total sulfur dioxide  
  
8 - density  
  
9 - pH  
  
10 - sulphates  
  
11 - alcohol  
  
Output variable (based on sensory data):  
  
12 - quality (score between 0 and 10)

### Tips

What might be an interesting thing to do, is aside from using regression modelling, is to set an arbitrary cutoff for your dependent variable (wine quality) at e.g. 7 or higher getting classified as 'good/1' and the remainder as 'not good/0'.  
This allows you to practice with hyper parameter tuning on e.g. decision tree algorithms looking at the ROC curve and the AUC value.  
Without doing any kind of feature engineering or overfitting you should be able to get an AUC of .88 (without even using random forest algorithm)

**KNIME** is a great tool (GUI) that can be used for this.  
  
1 - File Reader (for csv) to linear correlation node and to interactive histogram for basic EDA.  
  
2- File Reader to 'Rule Engine Node' to turn the 10 point scale to dichtome variable (good wine and rest), the code to put in the rule engine is something like this:

* **$quality$ > 6.5 => "good"**
* **TRUE => "bad"**  
    
  3- Rule Engine Node output to input of Column Filter node to filter out your original 10point feature (this prevent leaking)  
    
  4- Column Filter Node output to input of Partitioning Node (your standard train/tes split, e.g. 75%/25%, choose 'random' or 'stratified')  
    
  5- Partitioning Node train data split output to input of Train data split to input Decision Tree Learner node and  
    
  6- Partitioning Node test data split output to input Decision Tree predictor Node  
    
  7- Decision Tree learner Node output to input Decision Tree Node input  
    
  8- Decision Tree output to input ROC Node.. (here you can evaluate your model base on AUC value)

### Inspiration

Use machine learning to determine which physiochemical properties make a wine 'good'!

### Acknowledgements

This dataset is also available from the UCI machine learning repository, <https://archive.ics.uci.edu/ml/datasets/wine+quality> , I just shared it to kaggle for convenience. (I am mistaken and the public license type disallowed me from doing so, I will take this down at first request. I am not the owner of this dataset.

**Please include this citation if you plan to use this database: P. Cortez, A. Cerdeira, F. Almeida, T. Matos and J. Reis. Modeling wine preferences by data mining from physicochemical properties. In Decision Support Systems, Elsevier, 47(4):547-553, 2009.**

### Relevant publication

P. Cortez, A. Cerdeira, F. Almeida, T. Matos and J. Reis. Modeling wine preferences by data mining from physicochemical properties.  
In Decision Support Systems, Elsevier, 47(4):547-553, 2009.

Link : https://www.kaggle.com/datasets/uciml/red-wine-quality-cortez-et-al-2009