Chatgot version

Section 2. Why is the data interesting: Data description

**Data 1 : Wine quality**

A group of blue and white bars

Description automatically generated with medium confidence A chart of different types of substances

Description automatically generated with medium confidence

The Wine Quality dataset, sourced from Kaggle [1], provides psychochemical and sensor data for Portuguese "Vinho Verde" wine. After cleaning, it consists of 6,463 records and 12 features. Among these features, 11 are continuous, covering aspects like acidity, sugar content, and more, while 1 (the "type") is categorical with two categories.

The target variable, "quality," is ordinal, ranging from 3 to 9, with higher values indicating better wine quality. Notably, the dataset's quality distribution is imbalanced, with the majority falling into categories 5, 6, and 7 (32%, 46%, and 16% respectively), while the remaining four categories each represent less than 3% of the total distribution. To mitigate prediction errors on both extremes, we introduced a derived target feature called "quality\_cat," indicating "average or better" quality (quality > 5), resulting in two categories with a distribution of 36.7% and 63.3% for low and average-or-better quality, respectively.

Cleaning the data involved removing rows with missing values, which accounted for only 0.5% of the total data. This step did not compromise the generalization of the dataset.

To simplify analysis, both the "type" and "quality\_cat" features were converted into Boolean variables, where "white" and "average or above" were assigned values of 1.

Upon examining the univariate distribution of all features and the target variable (Figure 1), it becomes apparent that 8 out of the 11 numeric features exhibit significant skewness, except for "fixed acidity," "pH," and "alcohol." Different machine learning models may respond differently to these distributions, making it an interesting aspect to explore.

Figure 2 reveals that there are no strong correlations in the data, except for "total sulfur dioxide" with "free sulfur dioxide" and the "type" feature, showing correlations of -0.72 and 0.7 respectively. However, these correlations were not addressed in this study.

Lastly, the data was divided into an 80:20 ratio for the training and test sets. The training data was further used to create a cross-validation set, distinct from the actual training set.

**Data 2 : Gender Prediction**

A group of blue and white graphs

Description automatically generated A graph of height and height

Description automatically generated with medium confidence

The Gender Classification dataset, sourced from Kaggle, contains synthetic data representing various facial attributes used for predicting gender.

- \*\*Data Overview\*\*: This dataset comprises 5,001 records and 7 features. Notably, 5 of these features are categorical, while the remaining 2 ('forehead\_width\_cm' and 'forehead\_height\_cm') are continuous. This dataset was chosen to explore how different models handle two distinct types of features.

- \*\*Target Variable\*\*: The target variable is categorical, with two possible values: Male and Female, evenly split at 50% each. This balanced distribution differs from the previous dataset, which had slight skewness.

- \*\*Data Characteristics\*\*: With the exception of the 'long\_hair' categorical feature, the distributions for other features are approximately symmetric. Additionally, two variables exhibit balanced data in the lower 75% of values, with a slight skew toward the higher end. These distributions are not entirely uniform, highlighting a more balanced dataset and reducing potential model biases caused by skewness.

- \*\*Correlation Analysis\*\*: Figure 2 illustrates varying degrees of correlation between features, with no clear inverse relationships. Notably, there is a significant positive correlation among features related to the nose, lips, and gender.

- \*\*Data Splitting\*\*: Similar to the previous dataset, we divided this data into training, testing, and validation sets for improved model evaluation and comprehensibility.

This dataset offers an opportunity to examine how different machine learning models handle both categorical and continuous features. The balanced nature of the target variable and the variety of feature correlations make it an interesting case for model exploration and analysis.

**Evaluation Metric: Negative Log Loss**

In our study, we employ Negative Log Loss as our evaluation metric. Log Loss quantifies the disparity between predicted probabilities and actual class labels. Minimizing Log Loss equates to maximizing the likelihood of the true labels given the predicted probabilities.

Log Loss places significant emphasis on the accuracy of individual predictions. It penalizes strongly for confidently incorrect predictions. In simpler terms, when the model is highly confident in an incorrect prediction, Log Loss increases significantly. This sensitivity encourages the model to provide well-calibrated probability estimates, effectively addressing class imbalances within the data.

**Section 3: Modeling and Analysis Methodology:**

Five algorithms were modeled over each dataset, where the nature of underfitting and overfitting was studied, along with the preference and inductive bias of the model across different parameters. The best set of parameters was found using iterative validation (grid search), within a limited scope of parameter sets and ranges. The learning curve of each model for the best-found parameters and a suboptimal variant was further studied.

The analysis in the following sections is segmented into three segments:

1. Analysis of each algorithm across different datasets.
2. Comparative analysis across algorithms for each dataset.
3. Summary analysis.

**Section 4: Analysis of Each Algorithm Across Different Datasets**

This section provides a comprehensive analysis of the implementation of the five different algorithms mentioned in Section 1. These algorithms are applied to the datasets discussed in Section 3.

**4.1 Decision Tree**

A graph of a graph

Description automatically generated with medium confidenceA screenshot of a graph

Description automatically generatedThe decision tree was trained and optimized through pre-pruning using the "Max\_depth" parameter and post-pruning using "Ccp-alpha" (cost complexity parameter) on both datasets, as detailed below.

Parameter 1: Cost Complexity Parameter (ccp\_alpha)

- Dataset 1:

- Smaller values of ccp\_alpha result in a very high negative log loss (NLL) for the training data and a very low NLL for the validation set, indicating low classification accuracy and significant overfitting. Increasing the parameter value brings both the train and validation NLL closer, reaching a peak NLL of approximately -0.533 for cross-validation (CV) at around 0.00265, representing the highest generalization performance. Further increases in the parameter cause both train and CV NLL to gradually decrease together, indicating a move toward underfitting.

- ccp\_alpha controls the trade-off between model complexity and accuracy for Decision Trees (DT). Lower ccp\_alpha values result in limited pruning, leading to deeper and more complex trees that fit training instances more accurately. However, due to the preference bias of DT, lower nodes contribute less information gain compared to the root node, resulting in models fitting specific intricacies of the training data but losing generalization over unseen instances, as observed by low NLL for CV and high NLL for training data. Continuous features in Dataset 1 provide more opportunities for creating defined rules over the features, increasing the chance of overfitting.

- Conversely, higher ccp-alpha values prune more nodes closer to the root, corresponding to higher information gain. These nodes may represent the minimum specificity required for representing part of the target function, leading to more general rules and decreasing both training and CV NLL.

- Dataset 2:

- Similar to Dataset 1, the decision tree model overfits at lower values of ccp\_alpha, reaches its peak NLL of approximately -0.1066 at around 0.0008, and then starts underfitting as the parameter value increases. Compared to Dataset 1, Dataset 2 exhibits significantly better classification performance at the peak. This is true for both the lowest and highest complexity of the model for Dataset 2 as well. Additionally, the peak performance is achieved at a much lower parameter value for this dataset.

- The reason for this improvement is that Dataset 2 has fewer parameters, resulting in a lower max-depth. Moreover, most features are categorical with only two possible values, further limiting the possible cuts that can be made. Additionally, as the data is synthetically created, more features have higher correlation with the **A graph with orange and blue dots

Description automatically generated with medium confidence**target feature compared to Dataset 1, which could result in a significant increase in information gain.

- A common observation for both datasets is that after a certain value of ccp\_alpha, the descent toward underfitting is much more gradual, suggesting that these parameter values were not sufficient to remove any highly significant nodes yet, and these nodes would be much closer to the root or could be the root node itself.

Parameter 2: Max Depth

- Dataset 1:

- Higher max-depth represents a more complex tree, and similar to ccp\_alpha at lower values, it shows underfitting, with both CV and train NLL being low. The peak NLL of approximately -0.5319 for CV occurs at depth 4. As postulated for ccp\_alpha, the performance difference between depth 1 and the peak performance depth 4 is not substantial. This suggests that the information gain attained by the root node is significantly higher compared to subsequent feature rules, supporting the slow descent to underfitting for higher values of ccp\_alpha for Dataset 1. This showcases the preference bias of DT. The peak performance for both parameters is comparable, indicating that both pruning mechanisms are equally effective for the current classification problem.

- At higher max-depth values, the model descends into overfitting.

- Dataset 2:

- DT max-depth behaves similarly to Dataset 1, with underfitting at low values and peak performance at an NLL of approximately -0.1116 for CV at depth 4, followed by a descent into overfitting. Compared to Dataset 1, the root node for Dataset 2 itself shows higher performance, and subsequent 3 tree depths add significant information gain. Unlike Dataset 1, more features in Dataset 2 were correlated with the target, but they were less correlated with each other. This results in a better peak NLL, supporting the idea that while more features in Dataset 2 were correlated with the target, they exhibited less correlation among themselves, adding more individual value to classification.

**Learning curve**

The performance of DT with increase in the number training instances was observed, comparing Best found DT parameters for the data and a suboptimal DT based on the final CV NLL.

Dataset 1:

As shown above, the performance of the suboptimal Decision Tree (DT) with ccp\_alpha: 0.00001 and depth: 6, representing a more overfit model, was compared to the best-found model with 'ccp\_alpha': 0.0002 and 'max\_depth': 4, achieving a maximum negative log loss (NLL) of -0.5319. In both cases, increasing the size of the training data reduces the gap between CV and training data performance, indicating that more data leads to better generalization.

However, throughout different training sizes, the gap in NLL between CV and training data for the suboptimal model remains higher due to significant overfitting. Additionally, since the suboptimal model has more decision nodes with low ccp\_alpha and high tree depth, the training time for the suboptimal model is also higher, approximately 0.035 compared to approximately 0.020 for the best model with the entire dataset.

[1] Wine Quality dataset: <https://www.kaggle.com/datasets/rajyellow46/wine-quality>