Machine Learning for Wine Cultivars

Classifying Italian Wines by Chemical Profiles

Jingyi Xu
jx15@ualberta.ca
University of Alberta
Edmonton, Alberta, Canada

ABSTRACT

This project explores the application and evaluation of machine learning classifiers on the Wine Dataset, which comprises chemical analysis results from wines grown in a specific region in Italy, originating from three different cultivars. The dataset features 13 chemical attributes, including Alcohol, Malic Acid, Ash, among others, to serve as input for the classifiers. Our objective is to accurately categorize these wines into their respective cultivars, labelled as classes 0, 1, and 2, based on their chemical profiles. To achieve this, we employ three distinct approaches: K-Nearest Neighbors (KNN), Support Vector Machine (SVM), and Softmax, alongside a random prediction model as our baseline for comparison. The effectiveness of these models will be assessed through metrics such as accuracy, precision, and recall. This study aims to highlight the potential of machine learning in distinguishing between wine cultivars based on chemical properties, potentially offering insights into their quality, characteristics, and market segmentation.

1 INTRODUCTION

Wine-making is an art and a science that has been perfected over thousands of years, leading to a rich and diverse industry steeped in tradition. The chemical composition of wine is a critical factor that influences its distinct characteristics, including aroma, taste, potential for aging, and overall quality. This project harnesses the power of machine learning to decode the intricate relationship between a wine's chemical makeup and its variety.

This research focuses on classifying Italian wines according to their chemical fingerprints. The challenge lies in analyzing the Wine Dataset. Despite their common geographic origin, these wines hail from three different cultivars, each with its unique signature. Thirteen chemical markers, such as Alcohol content, Malic Acid, and Ash, are considered, offering a comprehensive profile for each wine sample.

The motivation for this study is dual: not only does it seek to precisely categorize wines by their cultivars, enhancing our understanding of quality preservation and trait amplification—critical for meeting consumer demands and industry benchmarks—but it also aims to demonstrate the capabilities of machine learning models in complex classification scenarios. The project involves the application and comparative evaluation of several models, including K-Nearest Neighbors (KNN), Support Vector Machine (SVM), and Softmax Regression, using a random prediction approach as a baseline measure of effectiveness.

2 PROBLEM SETUP AND FORMULATION

The main goal of this research is to devise a machine learning strategy capable of classifying wines according to their cultivars using chemical attributes as the basis for differentiation. This task is framed as a multinomial classification problem, where the input is a set of chemical features of each wine, and the output is the categorization of each wine into one of the varieties, identified as classes 0, 1, and 2.

Let us consider a dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$, where x_i denotes a feature vector containing 13 chemical attributes associated with the i^{th} wine sample, and y_i represents the corresponding cultivar label. The challenge lies in constructing a predictive function $f : X \rightarrow Y$ that accurately assigns a class label y to a given input feature vector x. This function f, represented by a machine learning model, aims to minimize the prediction error across the entirety of the dataset, effectively mapping chemical profiles to their respective wine cultivars.

2.1 Datasets and Samples

The study utilizes the Wine Dataset. The dataset comprises 178 samples, and each is defined by 13 measured chemical properties. These properties encompass a range of attributes, namely: Alcohol, Malic Acid, Ash, Alcalinity of Ash, Magnesium, Total Phenols, Flavanoids, Nonflavanoid Phenols, Proanthocyanins, Color Intensity, Hue, OD280/OD315 of diluted wines, and Proline.

The classification within the dataset is tripartite, reflecting the three cultivars. Specifically, there are 59 instances of Class 0, 71 instances of Class 1, and 48 instances of Class 2. This distribution highlights the balanced nature of the dataset, which is beneficial for training machine learning models.

2.2 Configuration

This section outlines the systematic setup for the machine learning models, their evaluation criteria, and the experimental framework.

- **Model Configuration:** The study deploys three machine learning algorithms: K-Nearest Neighbors (KNN), Support Vector Machine (SVM), and Softmax Regression. A meticulous hyperparameter tuning process is conducted using cross-validation techniques, specifically GridSearchCV, to

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM
https://doi.org/10.1145/nnnnnnn.nnnnnnn

determine the most effective model parameters for the task of classifying wines into their respective cultivars.
- **Evaluation Metrics:** The efficacy of the models is quantified by accuracy, precision, recall, and F1-score. These metrics are pivotal in understanding the models' ability to correctly classify the wine samples and provide a comparative measure against a baseline model that utilizes random predictions, thus establishing a benchmark for model assessment.
- **Experimental Setup:** In adherence to common practice, the Wine Dataset is partitioned into training, validation, and testing subsets, constituting 60%, 20%, and 20% of the data, respectively. The models are initially trained using the training subset, refined with the validation subset, and subsequently evaluated on the test subset to validate their predictive generalization capabilities.

3 METHODOLOGY

Models and Baseline:

- **Baseline Model**: We established our baseline with a model that predicts wine classes based on the distribution of classes in the training set. This random prediction model provided us with a basic accuracy range around 24%- 50%, against which the performance of more sophisticated models could be measured.
- **K-Nearest Neighbors (KNN)**: We applied the KNN algorithm with a range of hyperparameters, utilizing GridSearchCV to find the optimal configuration. The model demonstrated significant improvement over the baseline, showcasing the efficacy of machine learning in this classification task.
- **Support Vector Machine (SVM)**: The SVM model was configured with various kernels and regularization parameters, again using GridSearchCV for optimization. This model's performance highlighted the advantages of SVM in handling non-linear classification boundaries.
- **Softmax Regression**: As a generalized logistic regression applicable for multi-class classification, Softmax Regression was fine-tuned with different regularization strengths and solvers, aiming to maximize the classification accuracy.

3.1 Comparison

Each model underwent evaluation on the validation set for hyperparameter tuning and was finally assessed on the test set. The evaluation criteria included accuracy, precision, recall, and F1-score, providing a comprehensive view of each model's performance.

The methodology not only involved the application of these models but also emphasized the importance of comparing their outcomes against a simple baseline. This comparative analysis allowed us to discern the value added by each algorithm and its suitability for the wine classification problem.

The SVM and Softmax models, in particular, achieved notable accuracy, significantly surpassing the baseline and demonstrating the potential of machine learning for predictive modelling in enology.

4 EVALUATION

We use 4 metrics to evaluate the success of machine learning models in classifying Italian wines based on their chemical profiles, which are accuracy, precision, recall, and F1-score.

4.1 Reason for Metrics Choosen

- **Accuracy**: the most intuitive performance measure and it is simply a ratio of correctly predicted observations to the total observations. Accuracy is a great measure when the target classes are well-balanced. Given the relatively balanced nature of the wine dataset's classes, accuracy is a reasonable initial indicator of model performance.
- **Precision (Positive Predictive Value)**: Precision measures the accuracy of positive predictions. It is crucial for scenarios where the cost of a false positive is high. In wine classification, precision helps us understand how reliable the model is when it predicts a specific cultivar.
- **Recall**: Recall calculates the ratio of correctly predicted positive observations to all observations in the actual class. This metric is important when the cost of a false negative is high. For wine classification, high recall means the model effectively captures most wines from a particular cultivar.
- **F1-Score**: Since precision and recall are often a trade-off, F1-Score becomes important as it balances the two metrics. It is particularly useful when we have an uneven class distribution, as it accounts for both false positives and false negatives.

4.2 Why Are These Metrics Reasonable?

These metrics are reasonable for our evaluation for several reasons:

- Comprehensive Evaluation: Together, they provide a rounded assessment of model performance, capturing not just the overall correctness (accuracy) but also how well the model distinguishes between different classes (precision and recall) and balances these aspects (F1-score).
- Balanced vs. Imbalanced Classes: While our dataset is relatively balanced, precision, recall, and F1-score ensure that we are well-equipped to handle and accurately assess performance even if the dataset has imbalanced classes.
- Decision-Making: Understanding these metrics allows stakeholders to make informed decisions. For instance, if a particular cultivar is more valuable or requires greater classification accuracy, prioritizing models with higher precision for that class could be beneficial.

5 RESULT

Classifier	Accuracy	Precision	Recall	F1-Score
Baseline	0.56	0.56	0.56	0.56
KNN	0.72	0.76	0.72	0.73
SVM	0.89	0.89	0.89	0.89
Softmax	0.94	0.95	0.94	0.94

Table 1: Summary of Classifier Performance

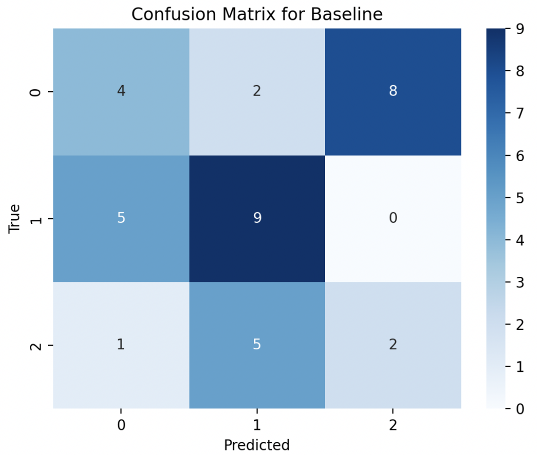

Figure 1: Confusion Matrix for Baseline classifier

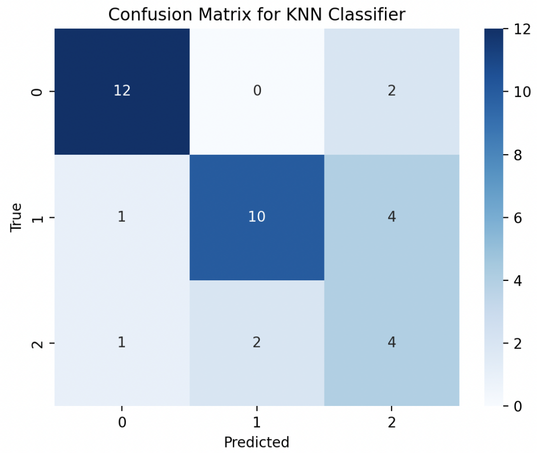

Figure 2: Confusion Matrix for KNN classifier

The results from the classification reports and confusion matrices provide insights into the performance of the baseline, KNN, SVM, and Softmax classifiers on the wine classification task.

5.1 Output Interpretation

According to the summary of classifier performance and confusion matrixes above:

(1) **Baseline Results**:
- Accuracy: 56%(each time is different but within the range 24%-50%)
- The classification report indicates moderate effectiveness in classification with precision, recall, and F1-score of around 0.50 for two classes, suggesting room for significant improvement.
- The confusion matrix reveals that the baseline model has a relatively even spread of predictions across the classes, but with a tendency to misclassify certain classes, as seen by off-diagonal numbers.

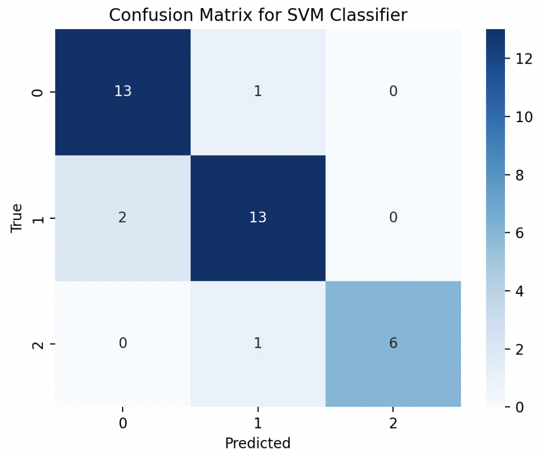

Figure 3: Confusion Matrix for SVM classifier

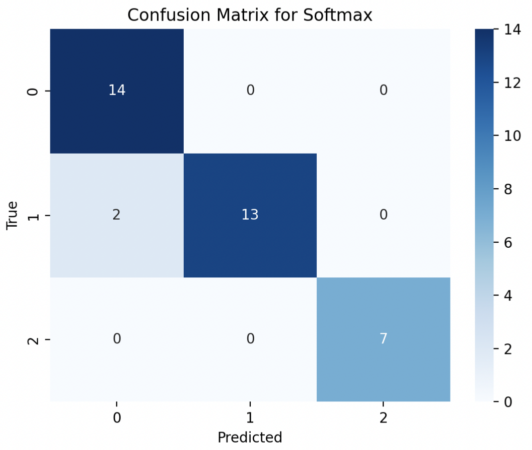

Figure 4: Confusion Matrix for Softmax classifier

(2) **KNN Classifier**:
- Accuracy: 72%
- Best Hyperparameters:

```
{ 'metric': 'manhattan',
  'n_neighbors': 7,
  'weights': 'distance' }
```

- KNN shows an improvement over the baseline, particularly for class_0, which has a precision and recall of 0.86.
- However, the F1-score for class_2 is just 0.47, which indicates a weakness of the KNN model in accurately classifying this particular class.
- The confusion matrix displays some misclassifications between the classes, especially class_2, which seems to be confused with both other classes.

(3) **SVM Classifier**:
- Accuracy: 89%
- Best Hyperparameters:

```
{'C': 0.1,
'degree': 3,
'gamma': 'auto',
'kernel': 'poly'}
```

- The SVM classifier performed well, especially for class_2, which has a perfect precision of 1.00, indicating no false positives for this class.
- However, recall for class_2 is 0.86, suggesting that the SVM classifier missed some true class_2 instances.
- The confusion matrix presents a few misclassifications, particularly between class_1 and class_0, but overall shows strong diagonal values, signifying a high rate of correct predictions.

(4) **Softmax Regression**:
- Accuracy: 94%
- Best Hyperparameters:

```
{'C': 1,
'solver': 'newton-cg'}
```

- The Softmax classifier achieved high precision and recall scores across all classes, with class_1 and class_2 showing perfect precision.
- This suggests that the model is highly capable of distinguishing between the classes with minimal false positives and false negatives.
- The confusion matrix supports this, showing strong diagonal values (correct classifications) and very few off-diagonal entries (misclassifications).

From the results, it is clear that both the SVM and Softmax classifiers outperformed the baseline and KNN models significantly. The Softmax Regression, in particular, showed exceptional performance with the highest accuracy and F1 scores, which implies it is highly effective at correctly classifying the wine samples in this dataset. This could be due to the ability of the Softmax Regression to model the probabilities of different classes well, providing an advantage in multi-class classification tasks.

The strong performance of the SVM model, particularly in terms of precision, suggests that it is quite effective at making correct positive predictions (when it predicts a class, that class is usually correct). However, the slight trade-off in recall indicates there is still some room for improvement in capturing all positive instances.

The KNN classifier's lower performance for class_2 could result from the class's complexity or an indication that the local similarity-based approach of KNN is not as effective for the given feature space as the global methods like SVM and Softmax.

5.2 Differences in Model Performance

The differences in performance between the models can be attributed to how they approach the classification task:

- **K-Nearest Neighbors (KNN)**: This model leverages the proximity of data points, relying heavily on the local distribution of data. Its sensitivity to noise and irrelevant features can affect performance, particularly if the dataset contains outliers or is not well-prepped for such locality-based assessments.
- **Support Vector Machine (SVM)**: SVM's strength lies in its ability to find an optimal hyperplane that maximizes the margin between classes in a high-dimensional space. Its efficacy is further enhanced with an appropriate kernel function, enabling it to effectively manage non-linear class separations.
- **Softmax Regression**: Ideal for multinomial classification, Softmax Regression operates under the premise that classes are linearly separable within the feature space. The better performance of this model suggests that such a linear assumption is valid for the given dataset.

5.3 Implications

The superior performance exhibited by the SVM and Softmax models suggests that the chemical properties of the wines have distinct patterns that these models can learn and generalize well. Particularly, the efficacy of Softmax Regression underscores the dataset's amenability to linear classification approaches, suggesting a potential linear relationship between the chemical attributes and the wine cultivars.

On the other hand, the lower performance of the K-Nearest Neighbors (KNN) model in accurately classifying wines from class 2 hints at complexities or similarities in the chemical profiles of this category that challenge proximity-based classification methods, but employing linear separations or operating in higher-dimensional feature spaces can handle this challenge.

6 ETHICAL IMPLICATIONS

- **Dual Use**: While wine classifiers can verify wine components, they can also potentially be used to create counterfeit wines by replicating a classifier's associated chemical profile of high-quality or specific cultivars of wines.
- **Overgeneralization**: The differences that contribute to a wine's profile may be oversimplified by a model, leading to broad categorization that fails to capture what makes a particular wine unique.

7 ADDITIONAL INFORMATION

The project repository is https://github.com/xuclaire0234/ML_Wine

REFERENCES

- Wine recognition dataset in scikit-learn (link)
- Multiclass using SKlearn's LogisticRegression (link)
- LogisticRegression in sklearn (link)
- Inspiration for this project (link)
- Inspiration for adding cross-validation to optimize the code (link)