SAN FRANCISCO
STATE UNIVERSITY

Mushroom Classification

Using Machine Learning

DS 612 Data Mining with

Business Applications

Spring 2024 Final Project

Prepared by Marcus Nogueira

1. Introduction

The objective of this project is to apply machine learning techniques to address a crucial real-world problem: mushroom classification. The goal is to develop predictive models capable of distinguishing between edible and poisonous mushrooms based on a set of 22 features. By leveraging the inherent patterns within the dataset, we aim to build robust classification models that can accurately classify mushrooms, thereby aiding in the prevention of potential health hazards associated with consuming poisonous mushrooms.

Mushroom classification is of significant importance due to the potentially life-threatening consequences of ingesting poisonous varieties. With numerous species exhibiting subtle visual differences, manual identification often proves challenging, if not perilous. Thus, the utilization of data mining methodologies presents a promising avenue for automating this process and enhancing accuracy. Additionally, accurate classification can contribute to identifying specific mushroom species, adding intrinsic value to the product and the technology.

My approach encompasses the utilization of various machine learning algorithms, including Logistic Regression, Support Vector Machines (SVM), K Nearest Neighbors (KNN), and Random Forest. Each algorithm offers unique strengths and characteristics, enabling us to explore diverse modeling techniques and identify the most effective approach for the classification task.

2. Problem Definition

The problem at hand involves the classification of mushrooms into edible and poisonous categories based on a set of 22 features. Given the features describing various attributes of mushrooms, such as cap shape, cap color, odor, etc., the task is to build predictive models that can accurately determine whether a mushroom is safe to consume or not.

Assumptions

We assume that the provided dataset is representative and accurately reflects the characteristics of mushrooms relevant to their edibility. Additionally, we assume that the features included in the dataset are sufficient for building effective classification models and capturing the distinguishing characteristics between edible and poisonous mushrooms.

Relevance

The chosen problem of mushroom classification is highly relevant due to its practical implications for public health and safety. Accidental consumption of poisonous mushrooms can lead to severe health complications, including organ failure and death. Therefore, the ability to accurately identify and classify mushrooms based on their edibility is of paramount importance.

Furthermore, this problem aligns well with the objectives of our machine learning project, offering an opportunity to apply various algorithms and techniques to a real-world scenario. By leveraging the power of machine learning, we aim to develop robust classification models that

can assist individuals in making informed decisions regarding mushroom consumption, ultimately reducing the risk of mushroom-related poisoning incidents.

3. Data Collection and Exploration

The dataset used in this analysis was sourced from the UCI Machine Learning Repository, a reputable platform known for its diverse collection of datasets suitable for machine learning tasks. This dataset specifically pertains to the classification of mushrooms from the Agaricus and Lepiota family and contains information about various attributes of mushrooms.

Dataset Size and Characteristics

The dataset comprises a total of 8124 instances, with each instance representing a unique mushroom observation. Each observation is described by 22 features, providing comprehensive information about the morphological characteristics of the mushrooms. Notably, the dataset exhibits a balanced distribution of classes, with approximately equal numbers of edible and poisonous mushrooms. There are no missing or unknown values.

Descriptive Statistics and Data Exploration

I began by importing the necessary libraries for my analysis, including NumPy, pandas, Matplotlib, Seaborn, and various modules from scikit-learn for preprocessing, model evaluation, and machine learning algorithms. Descriptive statistics were generated to summarize each feature's distribution, and visualizations were used to visualize the relationships between features and the target variable.

Figure 1: Correlation Heatmap of Mushroom Classification Features: Class, Gill Size, Spore Print Color, and Odor illustrates the relationships between key features, highlighting significant correlations that aid in understanding the dataset's structure.

4. Model Development

4.1 Investigating Data

Feature analysis revealed that attributes such as 'odor', 'gill-size', 'gill-color', 'ring-type', 'bruises', and 'spore-print-color' were significant in distinguishing between edible and poisonous mushrooms. We ensured data cleanliness, as there were no missing values or outliers, and converted categorical values into numerical ones using label encoding.

4.2 Description of Different Models Tried

We implemented several machine learning models to identify the most effective approach for mushroom classification:

Logistic Regression: Achieved an overall accuracy of 95%. However, the confusion matrix revealed a notable number of misclassifications, indicating some difficulty in discerning subtle differences between classes. Figure 2: Confusion Matrix for Logistic Regression provides a detailed view of the model's performance.

K Nearest Neighbors (KNN): Showed remarkable precision, recall, and F1-score metrics, but occasional misclassifications were observed. This is further illustrated in Figure 3: Confusion Matrix for K Nearest Neighbors (KNN).

Support Vector Machine (SVM): Exhibited near-perfect precision, recall, and F1-score metrics, with a few instances of misclassification. The performance details are shown in Figure 4: Confusion Matrix for Supported Vector Machines (SVM).

Random Forest: Emerged as the standout performer with flawless precision, recall, and F1-score metrics, and no instances of misclassification. Figure 5: Confusion Matrix for Random Forest corroborates the model's exceptional accuracy..

4.3 Feature Selection/Engineering

Through correlation analysis and feature importance evaluation using the Random Forest model, 'odor', 'gill-size', and 'spore-print-color' were identified as the most influential features. Figure 6: Feature Importance Plot provides insights into the relative significance of different features, aiding in understanding the patterns driving the model's decisions.

4.4 Parameter Tuning

I conducted iterative adjustments to refine the parameters of my models, such as varying the number of neighbors in KNN, adjusting the maximum iterations in logistic regression, and experimenting with different random state values (e.g., 42, 2021, 738) for the Random Forest model. I employed cross-validation to ensure robust model performance. Through these adjustments and iterative evaluations, the Random Forest model notably achieved exceptional

accuracy, reaching 100%. This outcome underscored its superiority over other models tested in the study.

4.5 Training and Testing

The dataset was split into training and testing sets, typically with an 80-20 split, ensuring that the models were trained on a large portion of the data while retaining a separate set for unbiased evaluation. Interestingly, after experimenting with different split ratios, I observed that the model performance did not significantly change with larger training splits. This was an intriguing finding, as it suggests that the models had already reached peak performance with the standard split, demonstrating robustness and consistency in their predictive capabilities.

4.6 Predictive Errors

An analysis of prediction errors revealed distinct variance and bias characteristics for each model. The Logistic Regression model, while proficient in generalizing overall patterns, struggled with capturing subtle nuances in the data. This led to higher variance, as the model tended to overfit to specific aspects of the training set, resulting in less consistent performance on the test set. In contrast, the Random Forest model demonstrated both low variance and low bias, making it the most robust among the models tested. The ensemble nature of Random Forest, which combines multiple decision trees, allowed it to effectively capture complex patterns without overfitting, ensuring stable and accurate predictions across different data splits. Additionally, the SVM model with a fixed kernel type maintained a balance between bias and variance, performing well but not matching the peak performance of the Random Forest. The superior performance of the Random Forest model, which achieved 100% accuracy, underscores its ability to generalize well from the training data to unseen data, making it the

most reliable choice for this classification task. This robustness was further confirmed through various experiments with different random states and training-test splits, highlighting the model's consistency and dependability.

5. Results

To address underfitting and overfitting, our approach encompassed a multifaceted strategy involving cross-validation, regularization, and normalization techniques. Cross-validation allowed us to assess the model's generalization performance by systematically partitioning the dataset into training and validation sets, thus reducing the risk of overfitting. Regularization methods, such as adjusting hyperparameters like the regularization strength in logistic regression or the maximum depth of trees in decision trees, helped control model complexity and combat overfitting. Additionally, we applied normalization techniques, such as scaling features using StandardScaler, to ensure consistent ranges across the dataset, which can aid in model convergence and stability. These combined efforts contributed to optimizing the Random Forest model, which demonstrated a superior balance between bias and variance, effectively mitigating overfitting tendencies. Finally, we evaluated the final model's performance using a comprehensive set of metrics including accuracy, precision, recall, and F1-score, providing a holistic assessment of its predictive capabilities.

Visualizations

Figure 1 illustrates the correlation heatmap of features crucial for mushroom classification, including Class, Gill Size, Spore Print Color, and Odor. Complementing this analysis, Figures 2-5 showcase confusion matrices, offering insights into model performance. Additionally, Figure

6 presents the feature importance plot, highlighting the significance of different features in classification tasks. These visualizations collectively provide a comprehensive understanding of both model performance and the relevance of specific features in the classification process. Notably, the feature importance plot reinforces the efficacy of the Random Forest model by demonstrating its ability to effectively capture and utilize information from various features, further corroborating its status as the top-performing model.

6. Conclusion

In summary, the Random Forest model was identified as the most effective approach for mushroom classification, achieving exceptional accuracy and robustness. The project outcomes highlight the potential of machine learning techniques in addressing real-world problems with significant implications for public health and safety.

The learning experience was invaluable, with challenges such as handling high-dimensional data and optimizing model parameters providing deep insights into machine learning workflows. Future work could involve exploring deep learning techniques, incorporating additional data sources, and developing a mobile application for real-time mushroom classification.

Visualizations

Figure 1: Correlation Heatmap of Mushroom Classification Features: Class, Gill Size, Spore Print Color, and Odor

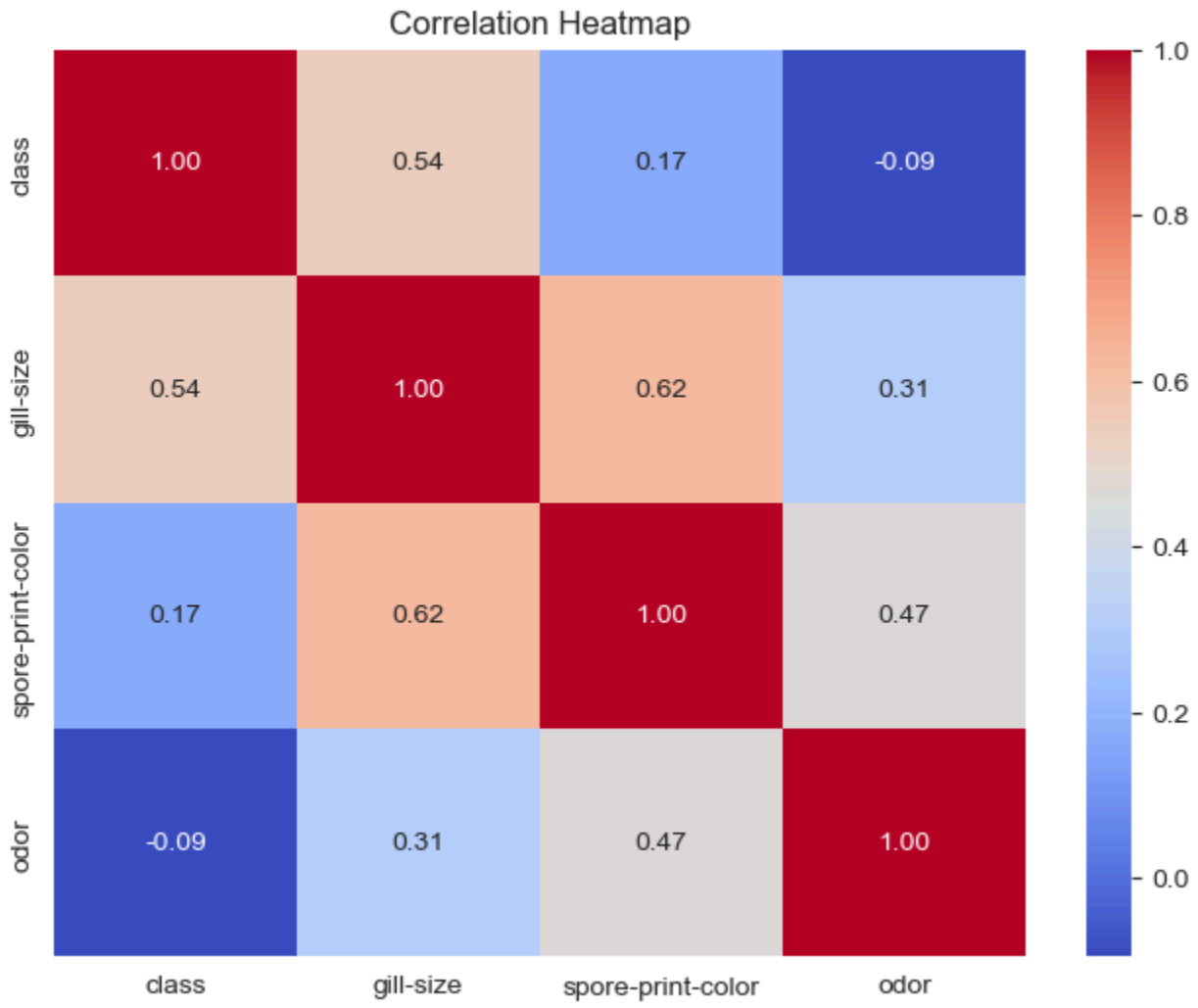

Figure 2: Confusion Matrix for Logistic Regression

```
Logistic Regression Model Evaluation:
Classification Report:
              precision    recall  f1-score   support

           0       0.95      0.95      0.95       843
           1       0.94      0.95      0.95       782

    accuracy                           0.95      1625
   macro avg       0.95      0.95      0.95      1625
weighted avg       0.95      0.95      0.95      1625

Confusion Matrix:
[[799  44]
 [ 41 741]]
```

Figure 3: Confusion Matrix for K Nearest Neighbors (KNN)

```
K Nearest Neighbors Model Evaluation:
Classification Report:
              precision    recall  f1-score   support

           0       1.00      0.99      1.00       843
           1       0.99      1.00      1.00       782

    accuracy                           1.00      1625
   macro avg       1.00      1.00      1.00      1625
weighted avg       1.00      1.00      1.00      1625

Confusion Matrix:
[[837   6]
 [  0 782]]
```

Figure 4: Confusion Matrix for Supported Vector Machines (SVM)

```
Support Vector Machine Model Evaluation:
Classification Report:
              precision    recall  f1-score   support

           0       0.99      1.00      0.99       843
           1       1.00      0.99      0.99       782

    accuracy                           0.99      1625
   macro avg       0.99      0.99      0.99      1625
weighted avg       0.99      0.99      0.99      1625

Confusion Matrix:
[[842   1]
 [ 11 771]]
```

Figure 5: Confusion Matrix for Random Forest

```
Random Forest Model Evaluation:
Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00       843
           1       1.00      1.00      1.00       782

    accuracy                           1.00      1625
   macro avg       1.00      1.00      1.00      1625
weighted avg       1.00      1.00      1.00      1625

Confusion Matrix:
[[843   0]
 [  0 782]]
```

Figure 6: Feature Importance Plot