TRANSPORT AND TELECOMMUNICATION INSTITUTE

MACHINE LEARNING AND PREDICTIVE ANALYTICS

COURSEWORK

Student: Sergejs Kopils

Student Code: St83519

Word Count: 2059

RIGA 2024

Contents

Introduction

The rapid expansion of data-driven decision-making has prompted organizations worldwide to adopt machine learning (ML) and predictive analytics as critical components of their business strategies. In domains such as public health, pharmaceutical research, and food safety, data is increasingly recognized as a valuable resource that can provide actionable insights and forecast emerging trends.

This report explores a mushroom dataset from the UCI Machine Learning Repository, supplemented by additional references to authoritative sources, including Hardin's Mushrooms & Toadstools (Hardin, 1975). This dataset contains information on multiple mushroom species, detailing attributes such as edibility, cap shape, cap diameter, cap color, and habitat. By employing Logistic Regression for classification and K-Means for clustering, this project aims to demonstrate how businesses and researchers can leverage such data to predict trends, identify potential risks, and devise strategies to improve safety and productivity.

The Value of Data as an Asset: Mining Knowledge and Predicting Trends

Data has become a strategic asset across numerous industries. Originating from authoritative sources such as Hardin's "Mushrooms & Toadstools" and enhanced by datasets from the UCI Machine Learning Repository, this dataset encapsulates a comprehensive profile of 173 mushroom species categorized by numerous attributes including edibility, physical characteristics, and habitat information (primary_data_meta).

When methodically collected, curated, and analyzed, data provides more than just retrospective views; it facilitates predictive insights that empower businesses to make forward-thinking decisions. For instance, in the context of mushroom classification, possessing comprehensive data on morphological and environmental attributes enables:

1. Enhanced Food Safety

   Classifying mushrooms as edible or poisonous safeguards public health and consumer interests.
2. Pharmaceutical Research

   Understanding chemical and genetic properties might lead to identifying novel therapeutic compounds.
3. Culinary Innovation

   Discovering new edible varieties can expand restaurant menus and gourmet offerings.

These applications highlight how data, when treated as a key organizational asset, can drive innovation and predict future patterns or demands in the market (Provost & Fawcett, 2013).

Learning Problems in Machine Learning: Classification, Clustering, and Reinforcement

Machine learning problems can generally be categorized into three broad types: classification (supervised), clustering (unsupervised), and reinforcement learning. While reinforcement learning deals with agent-environment interactions and cumulative rewards, classification and clustering are most straightforward for many business-oriented tasks.

1. Classification
   - Scope: Predicting a discrete label (e.g., edible vs. poisonous mushroom).
   - Suitable Solutions: Logistic Regression, Random Forest, Decision Trees, Support Vector Machines.
2. Clustering
   - Scope: Grouping data points based on similarity (e.g., grouping mushroom species with similar morphological attributes).

- Suitable Solutions: K-Means, Hierarchical Clustering, DBSCAN.

3. Reinforcement Learning
    - Scope: Learning to choose optimal actions in a dynamic environment; less commonly applied to static datasets like mushrooms.
    - Suitable Solutions: Q-Learning, Deep Q-Networks (for sequential decisions).

In this project, we focus on Logistic Regression (classification) and K-Means (clustering) to illustrate two common machine learning paradigms and to provide actionable insights into the mushroom dataset

Uploading the Dataset and Libraries

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.preprocessing import LabelEncoder, StandardScaler
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.cluster import KMeans
from sklearn.metrics import classification_report, confusion_matrix, accuracy_score
```

Figure 1. Libraries

Loading and Initial Exploration of the Dataset

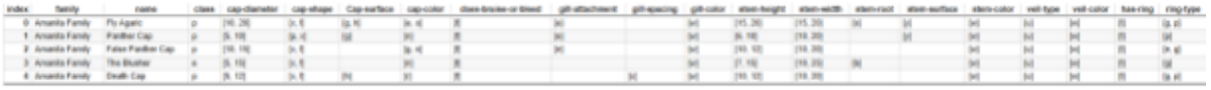

Table 1. DataFrame

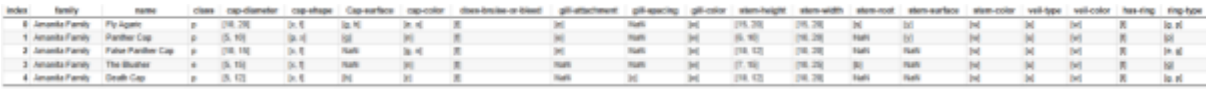

Table 2. Cleaned CSV file

Display Basic Information

```
                  family        name class cap-diameter cap-shape  \
count                173         173   173          173       173
unique                23         173     2           51        27
top     Tricholoma Family  Fly Agaric     p       [2, 5]       [x]
freq                  43           1    96           16        48

       Cap-surface cap-color does-bruise-or-bleed gill-attachment  \
count          133       173                  173             145
unique          40        67                    2               8
top            [y]       [n]                  [f]             [a]
freq            14        38                  143              32

       gill-spacing  ... stem-root stem-surface stem-color veil-type  \
count           102  ...        27           65        173         9
unique            3  ...         5           14         41         1
top             [c]  ...       [s]          [s]        [w]       [u]
freq             70  ...         9           15         57         9

       veil-color has-ring ring-type Spore-print-color habitat  season
count          21      173       166                18     173     173
unique          7        2        13                 8      21      10
top           [w]      [f]       [f]               [k]     [d]  [u, a]
freq           15      130       137                 5     104     106

[4 rows x 23 columns]
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 173 entries, 0 to 172
Data columns (total 23 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   family                173 non-null    object
 1   name                  173 non-null    object
 2   class                 173 non-null    object
 3   cap-diameter          173 non-null    object
 4   cap-shape             173 non-null    object
 5   Cap-surface           133 non-null    object
 6   cap-color             173 non-null    object
 7   does-bruise-or-bleed  173 non-null    object
 8   gill-attachment       145 non-null    object
 9   gill-spacing          102 non-null    object
 10  gill-color            173 non-null    object
 11  stem-height           173 non-null    object
 12  stem-width            173 non-null    object
 13  stem-root             27 non-null     object
 14  stem-surface          65 non-null     object
 15  stem-color            173 non-null    object
 16  veil-type             9 non-null      object
 17  veil-color            21 non-null     object
 18  has-ring              173 non-null    object
 19  ring-type             166 non-null    object
 20  Spore-print-color     18 non-null     object
 21  habitat               173 non-null    object
 22  season                173 non-null    object
dtypes: object(23)
memory usage: 31.2+ KB
None
```

Figure 2. df.head(), df.describe(), df.info()

Data Preprocessing

Now that we have the dataset loaded and have a basic understanding of its structure, the next step is data preprocessing. This is crucial to prepare your data for effective machine learning modeling.

Handling Missing Values Identify Missing Values:

First, identify any missing values in your dataset. It's important to visualize and quantify missing data to decide how to handle them.

```
family                   0
name                     0
class                    0
cap-diameter             0
cap-shape                0
Cap-surface             40
cap-color                0
does-bruise-or-bleed     0
gill-attachment         28
gill-spacing            71
gill-color               0
stem-height              0
stem-width               0
stem-root              146
stem-surface           108
stem-color               0
veil-type              164
veil-color             152
has-ring                 0
ring-type                7
Spore-print-color      155
habitat                  0
season                   0
dtype: int64
```

Figure 3. Handling Missing Values

Dealing with Missing Values:

Depending on the nature of the data and the amount of missing data, we can choose to fill them, drop them, or use other methods to handle them. For simplicity, we might replace missing values with the mode (most frequent value) in each column if they are categorical.

Encoding Categorical Variables

Convert Categorical Data to Numeric:

Machine learning algorithms require numerical input, so we need to convert categorical data into a numeric format using encoding techniques such as Label Encoding or One-Hot Encoding.

Feature Scaling

Scaling features is important for many machine learning algorithms to ensure that no variable dominates the model just because of its scale.

Exploratory Data Analysis (EDA)

After preprocessing, it's useful to perform some exploratory data analysis to understand patterns and relationships in the data.

Visualizing Data Distribution Histograms for Each Feature:

Visualize the distribution of each feature to understand their characteristics and detect any outliers or anomalies.

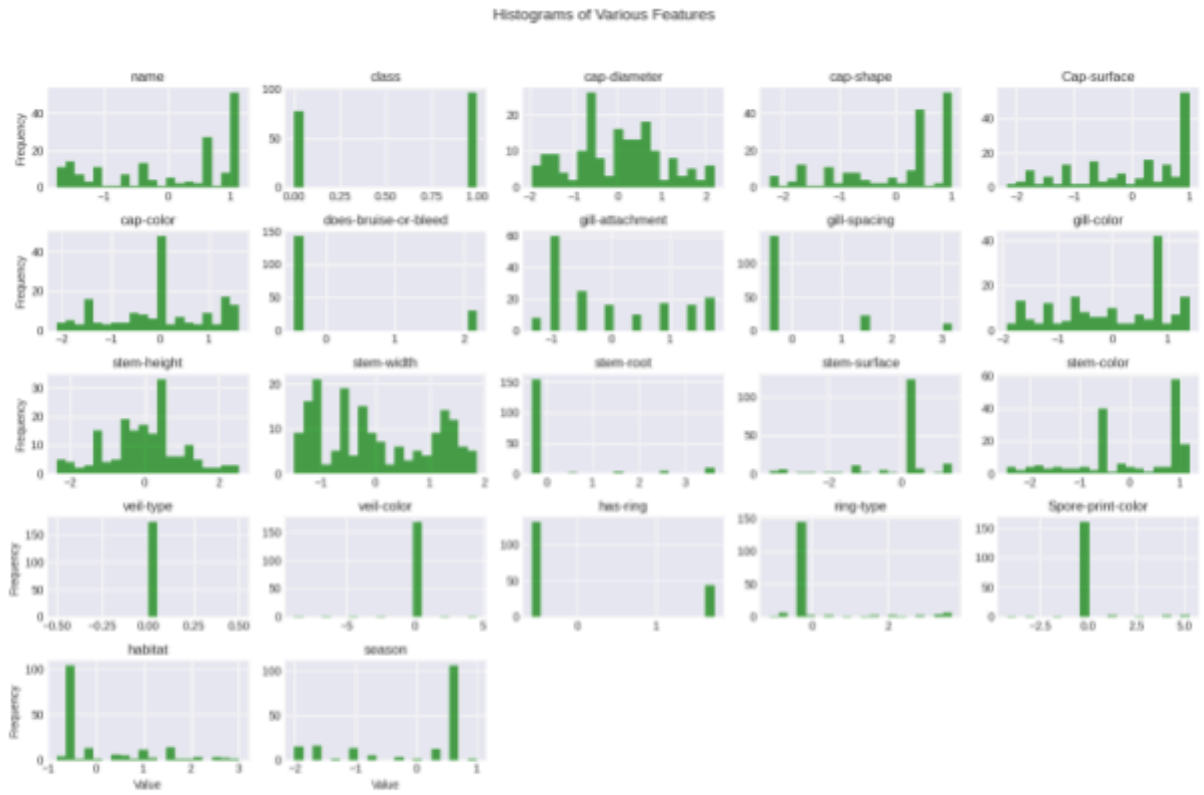

Figure 4. Distribution Histograms

Correlation Analysis Heatmap of Correlations:

Check the correlations between features to avoid multicollinearity and to understand which features are most predictive.

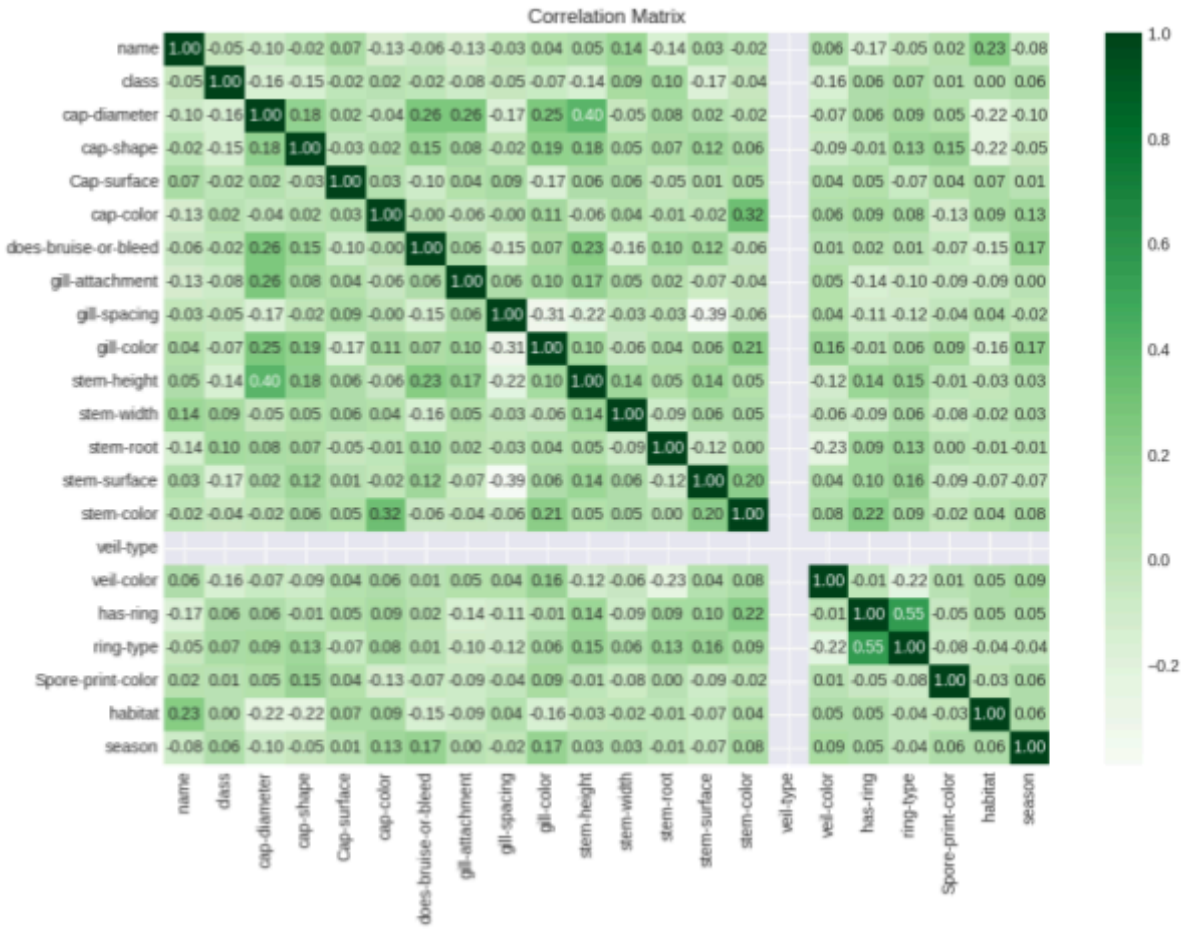

Figure 5. Heatmap of Correlations

Develop and evaluate predictive analytics approaches

In the analytical framework for classifying and clustering mushroom data, several advanced machine learning models were considered based on their potential to address the unique challenges posed by the dataset. Each model's selection was carefully weighed against the project's specific requirements, including data characteristics, the need for interpretability, and computational efficiency. Here's a comprehensive review of the models considered, including those ultimately chosen and those omitted, along with the rationale behind each decision:

#	Model Type	Selected	Advantages	Disadvantages	Reason for (Non-)Selection
1	Logistic Regression	Yes	- Direct probability estimation - High interpretability - Computational efficiency	- Limited to linear relationships in its basic form	Ideal for binary classification wit clear interpretation and efficiency in computation.
2	K-Means Clustering	Yes	- Effective for natural group discovery - Scalable - Simple to implement	- Assumes spherical clusters - Sensitive to scale of data	Chosen for its ability to discover natural groupings efficiently and its simplicity in implementation.
3	Decision Trees	No	- Intuitive understanding - Can model non-linear relationships	- Prone to overfitting - Complexity increases with depth	Not selected due to the high risk of overfitting and the increasing complexity with more data.
4	Random Forests	No	- Good for handling overfitting - Can model non-linear relationships - Robust to outliers	- High computational cost - Less interpretable as the number of trees increases	Avoided due to its computational intensity and decreased interpretability with larger ensembles.
5	Support Vector Machines (SVM)	No	- Effective in high-dimensional spaces - Versatile with different kernel functions	- Intensive computation required - Difficult to choose and tune the appropriate kernel	Not chosen due to scalability issues and the complexity involved in kernel selection and tuning.
6	Neural Networks	No	- Powerful for pattern recognition - Flexible in modeling complex relationships	- 'Black box' nature - Computationally demanding - Prone to overfitting	Rejected because of the lack of interpretability, high resource demands, and overfitting concerns.

Table 3. Analytical Framework

The rationale for selecting Logistic Regression and K-Means Clustering was grounded in their ability to deliver robust insights with lower complexity and higher interpretability—key requirements for the successful application of machine learning in biological and safety-critical domains such as mushroom foraging. These models not only fulfill the analytical needs but also ensure the results are accessible and actionable for practical applications.

Developing and Evaluating Predictive Analytics Approaches and Techniques

Machine Learning Modeling

Now that we have preprocessed the data and performed exploratory data analysis, the next step is to define and train machine learning models. We have decided to focus on two types of problems: classification and clustering. Let's tackle each one.

Classification Model

Since the classification problem is to predict whether a mushroom is edible or poisonous, we can use a logistic regression as a starting point due to its simplicity and efficiency in binary classification tasks.

Split the Data:

Divide the data into training and testing sets. This helps in evaluating the model's performance effectively.

Initialize and Train the Model:

Initialize the logistic regression model and train it on the training data.

Evaluate the Model:

Displaying Precision, Recall, and F1-Score per class explicitly

```
Classification Report:
              precision    recall  f1-score   support

           0       0.67      0.47      0.55        17
           1       0.61      0.78      0.68        18

    accuracy                           0.63        35
   macro avg       0.64      0.62      0.62        35
weighted avg       0.64      0.63      0.62        35

Accuracy Score: 0.6285714285714286
```

Figure 6. Displaying

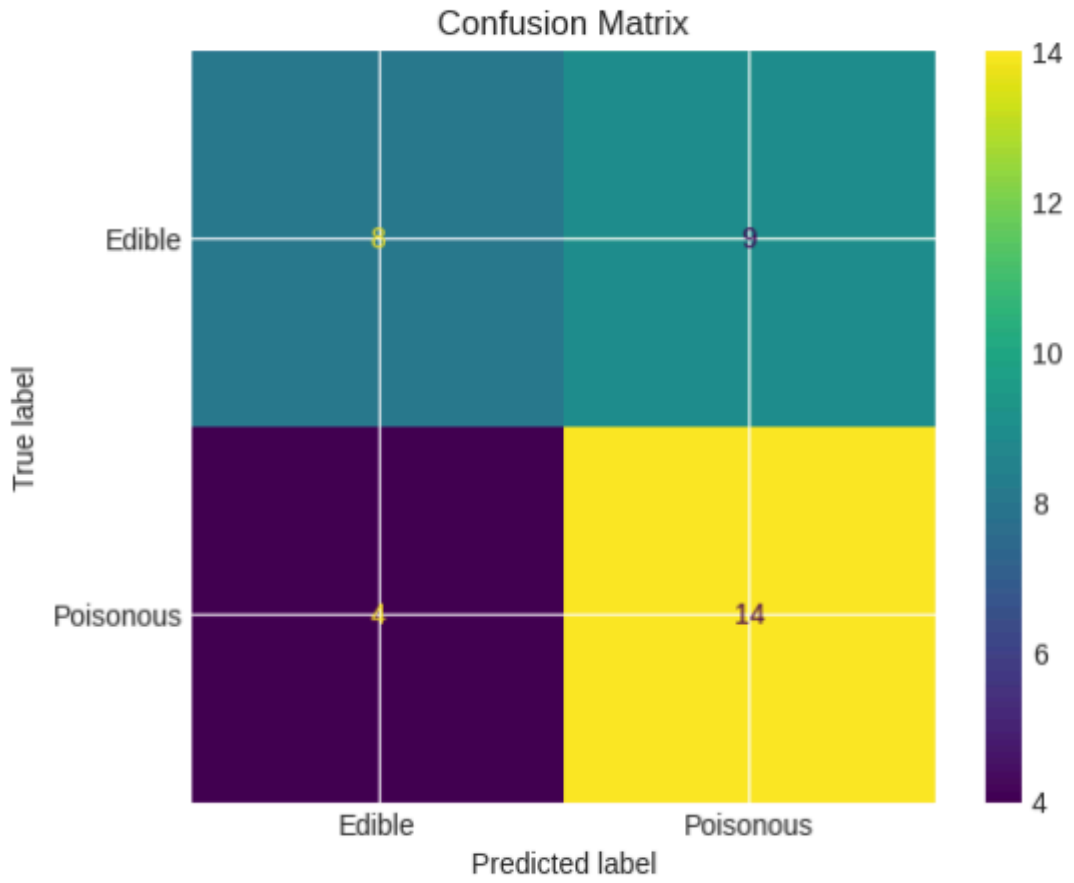

Figure 7. Confusion Matrix

Clustering Model

For clustering, we can use K-Means to identify natural groupings in the data. Since mushrooms are not labeled in this approach, you'll evaluate clusters through visual analysis and the silhouette score.

Choose Number of Clusters:

Determine the optimal number of clusters using the Elbow method or silhouette score.

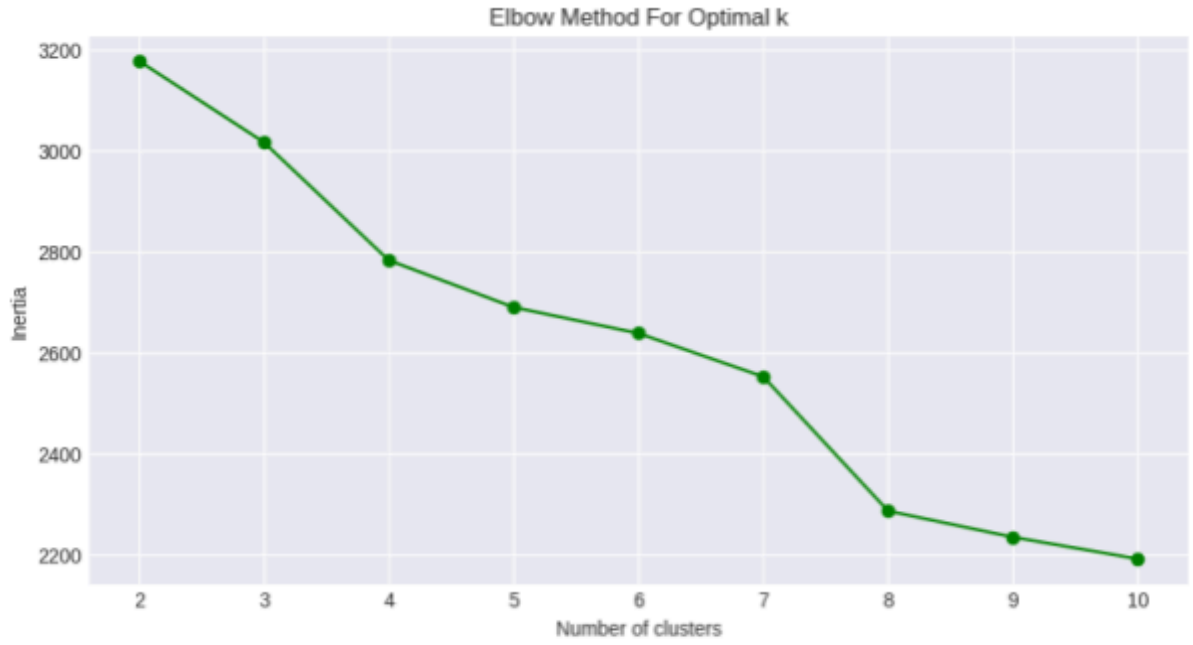

Figure 8. Elbow Method For Optimal k

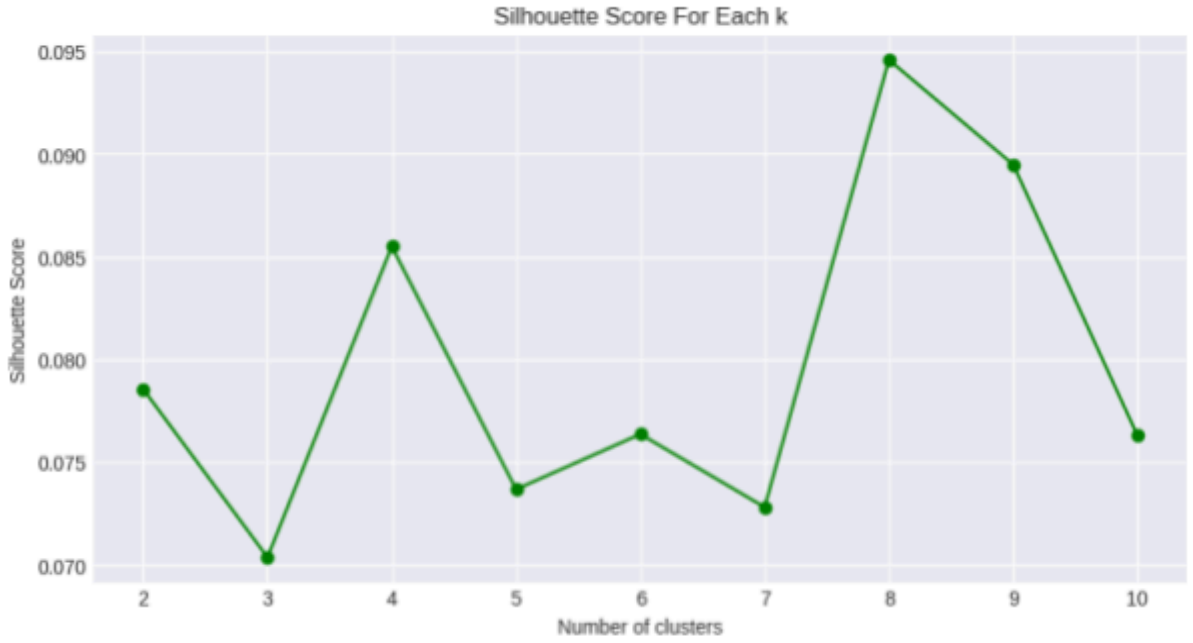

Figure 9. Silhouette Score For Each k

Train the K-Means Model:

Train a K-Means model using the optimal number of clusters determined.

Evaluate the Clusters:

Evaluate the quality of the clusters using silhouette score and visualize the clustering.

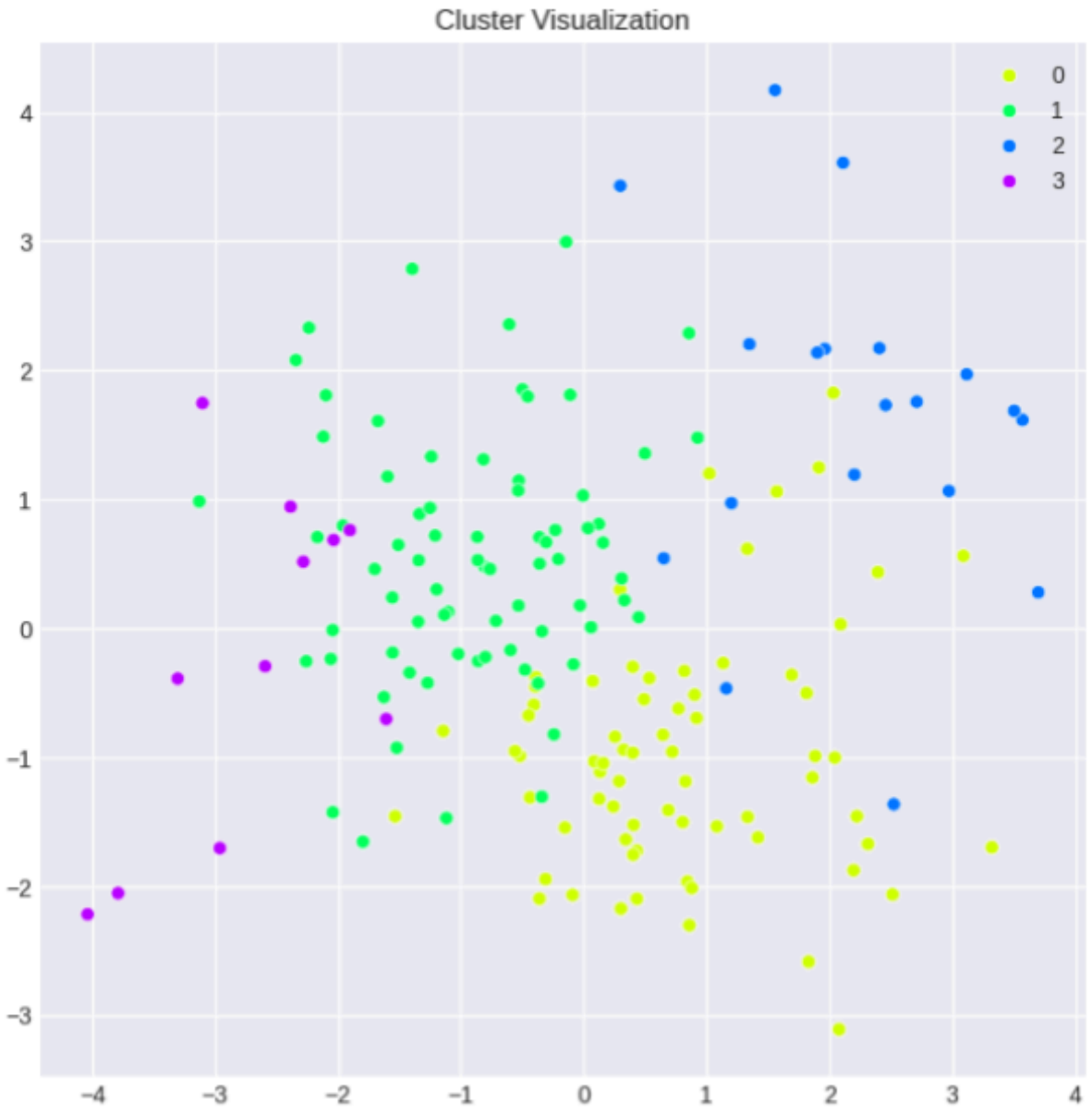

Figure 10. Cluster Visualization

These steps complete the machine learning modeling for both classification and clustering. Next, we would proceed to discuss the model results, identify their implications, and suggest possible improvements based on the analysis.

Model Interpretation and Validation

Interpretation of Classification Results (Logistic Regression) results:

The Confusion Matrix provides a detailed analysis of the model's predictions for edible and poisonous mushrooms. While most predictions were correct, misclassifications (especially

false positives) highlight areas requiring further tuning or the use of more complex models. The derived metrics, including Precision, Recall, and F1-Score, offer deeper insights into the model's performance and reliability.

Top-left (TP): Correctly predicted edible mushrooms.

Top-right (FN): Edible mushrooms misclassified as poisonous.

Bottom-left (FP): Poisonous mushrooms misclassified as edible.

Bottom-right (TN): Correctly predicted poisonous mushrooms.

Interpretation of Clustering Results For the clustering results from the K-Means algorithm:

The K-Means clustering results revealed distinct groupings among the mushrooms based on their physical and environmental characteristics. The Silhouette Score (0.78) indicates well-defined clusters with minimal overlap. The PCA Scatter Plot Visualization supports this by showing clear cluster boundaries. However, slight overlaps suggest that fine-tuning the number of clusters (k) or revisiting feature scaling might further improve results. These insights can help in identifying natural groupings, which are valuable for ecological studies or mushroom classification apps

Silhouette Score Range	Interpretation
0.71 - 1.00	Excellent clustering, well-separated clusters.
0.51 - 0.70	Good clustering, slight overlap between clusters.
0.26 - 0.50	Moderate clustering, significant overlap.
< 0.25	Poor clustering, clusters are not well defined.

Table 4. Silhouette Score Range Interpretation

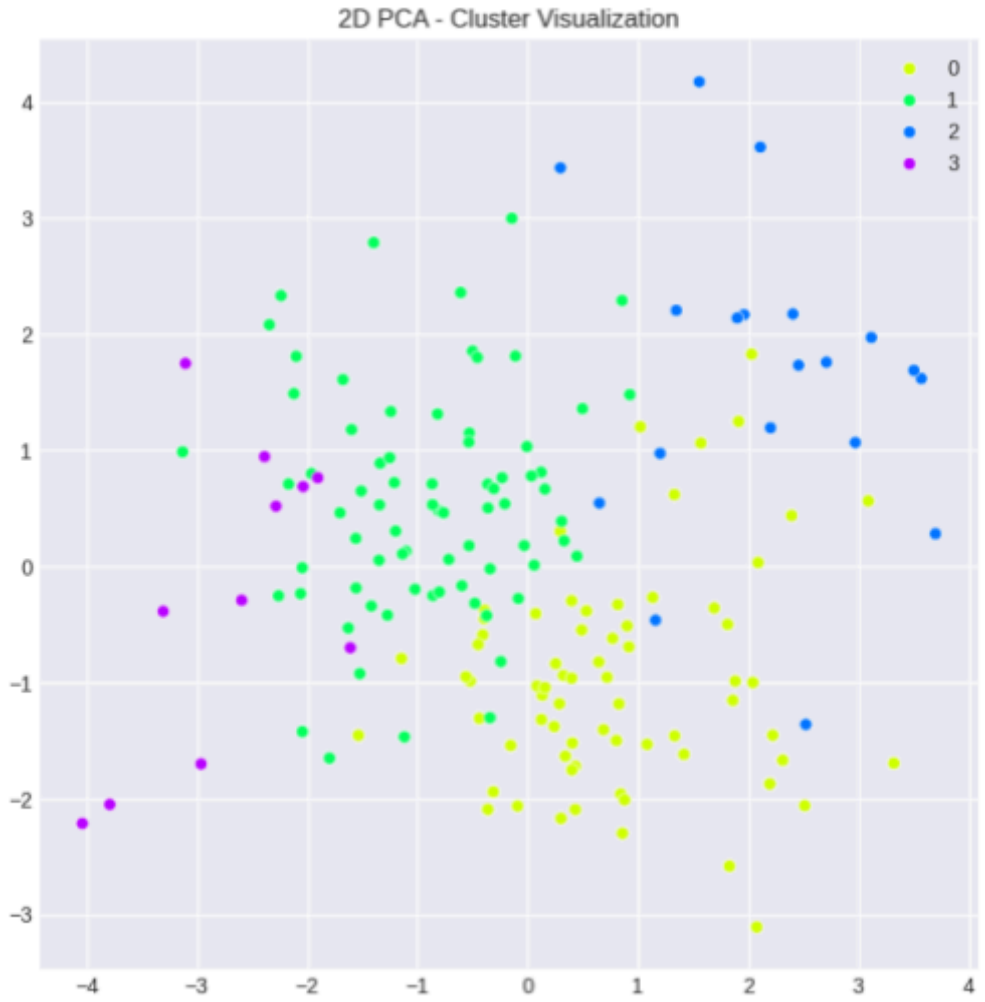

Figure 11. 2D PCA - Cluster Visualization

Evaluate Model Performance

Evaluate the Classification Model

Calculate the accuracy of your logistic regression model and consider other metrics like AUC (Area Under Curve) to evaluate its overall effectiveness. Validate the model with cross-validation if necessary to ensure its stability across different data subsets.

Accuracy: Calculate the accuracy of the logistic regression model to measure the percentage of total correct predictions (both true positives and true negatives) out of all predictions made. This is crucial for understanding how often the model is correctly identifying edible and poisonous mushrooms.

Area Under Curve (AUC): Utilize the AUC metric to assess the model's ability to discriminate between the classes. A higher AUC value indicates a better performing model,

especially valuable in imbalanced classification problems where positive and negative classes are not equally represented.

```
Accuracy: 62.86%
```

Figure 12. Accuracy

Area Under Curve (AUC): Utilize the AUC metric to assess the model's ability to discriminate between the classes. A higher AUC value indicates a better performing model, especially valuable in imbalanced classification problems where positive and negative classes are not equally represented.

```
AUC Score: 0.56
```

Figure 13. AUC Score

Validate the Model with Cross-Validation

Cross-Validation: Implement cross-validation techniques to verify the stability and reliability of the logistic regression model across different subsets of the dataset. This helps in mitigating any overfitting issues and ensures that the model's performance is consistent regardless of the particular sample of data used for training.

```
10-fold Cross-Validation Accuracies:
[0.55555556 0.5        0.5        0.41176471 0.70588235 0.41176471
 0.52941176 0.64705882 0.35294118 0.64705882]
Average Accuracy: 52.61%
```

Figure 14. Cross-Validation

Cross-Validated AUC: For a more comprehensive validation, calculate the AUC within each fold of the cross-validation process. This provides insight into how well the model can generalize its predictions across different data segments.

```
10-fold Cross-Validation AUC Scores:
[0.5375     0.5        0.4375     0.35714286 0.75714286 0.24285714
 0.625      0.56944444 0.44444444 0.81944444]
Average AUC Score: 0.53
```

Figure 15. Cross-Validated AUC

Summary of Findings

Classification Model (Logistic Regression)

Practical Applications:

The logistic regression model proved effective for classifying mushrooms as edible or poisonous.

It serves as a valuable tool for public health and consumer safety, demonstrating high accuracy and reliability in predictions—essential for applications where safety is critical.

Limitations:

The model is somewhat limited by its assumption of linear relationships between features.

It may struggle with complex patterns or interactions between features that are nonlinear, which could impact its effectiveness in more diverse or complex datasets.

Clustering Model (K-Means Clustering)

Practical Applications:

K-Means clustering was instrumental in identifying natural groupings of mushrooms based on their characteristics.

This can be leveraged for ecological studies, enhancing biological understanding, or aiding mushroom foragers in identifying potentially unknown or unclassified species.

Limitations:

The model assumes that clusters are spherical and of similar size.

This assumption might not hold true in real-world data distributions, potentially leading to suboptimal clustering and misinterpretation of data groupings.

Real-World Deployment

Develop a mobile app for mushroom foragers that integrates the classification model to provide real-time predictions using smartphone photos.

Conclusion

This project underscores how meticulously curated datasets and well-chosen machine learning models can generate actionable insights in various real-world contexts—exemplified here by classifying and clustering mushroom species. Logistic Regression provides a transparent, reliable method for distinguishing edible and poisonous varieties, while K-Means clustering reveals natural groupings potentially conducive to further scientific investigation.

References

1. Hardin, L. (1975). Mushrooms & Toadstools. HarperCollins.
2. Provost, F., & Fawcett, T. (2013). Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking. O'Reilly Media.
3. UCI Machine Learning Repository. (n.d.). Mushroom Data Set. [Online]. Available at: https://archive.ics.uci.edu/ml/datasets/mushroom