An Analysis of Edible and Poisonous Mushrooms

Masters of Professional Studies in Informatics, Northeastern University

ALY 6040: Data Mining Applications

Mohammed Saif Wasay

NUID: 002815958

Prof: Harpreet Sharma

30th September 2024

Table of Contents

Abstract

Mushroom foraging has gained popularity due to the increasing interest in natural food sources among culinary enthusiasts. However, the potential dangers of consuming poisonous mushrooms necessitate a comprehensive understanding of their distinguishing characteristics. This report examines a dataset containing various attributes of mushrooms to classify them as edible or poisonous. By utilizing statistical analysis and machine learning techniques, we identify key features that correlate with mushroom edibility and propose recommendations for safe foraging practices.

1. Introduction

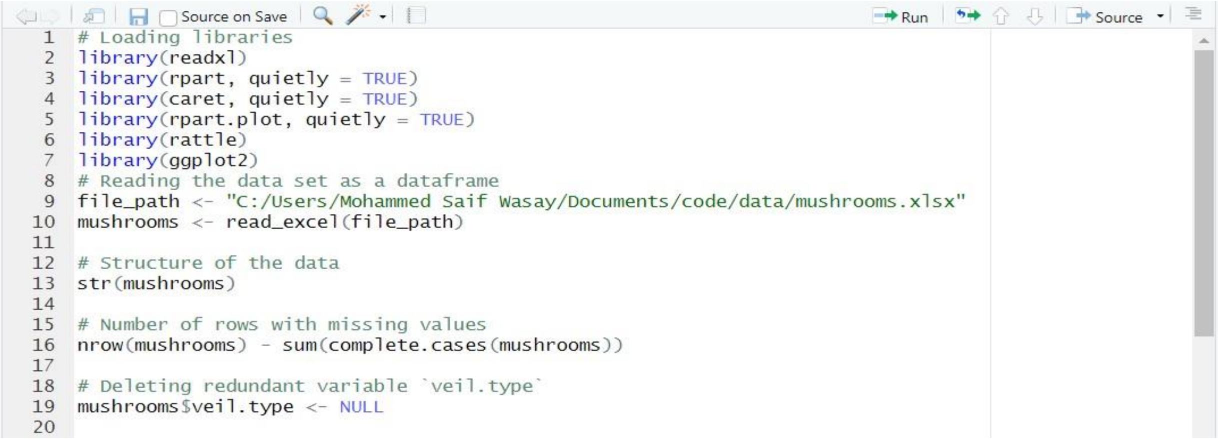
Mushrooms are a vital component of many ecosystems and have long been a part of human diets across cultures. While they offer significant nutritional benefits, the consumption of wild mushrooms carries risks, primarily due to the presence of poisonous species. This report aims to analyze a dataset of mushrooms, exploring attributes such as cap shape, cap color, odor, and gill attachment. By employing classification techniques, the objective is to develop a model that effectively differentiates between edible and poisonous mushrooms, thereby enhancing the safety of foragers.

2. Methodology & Data Collection

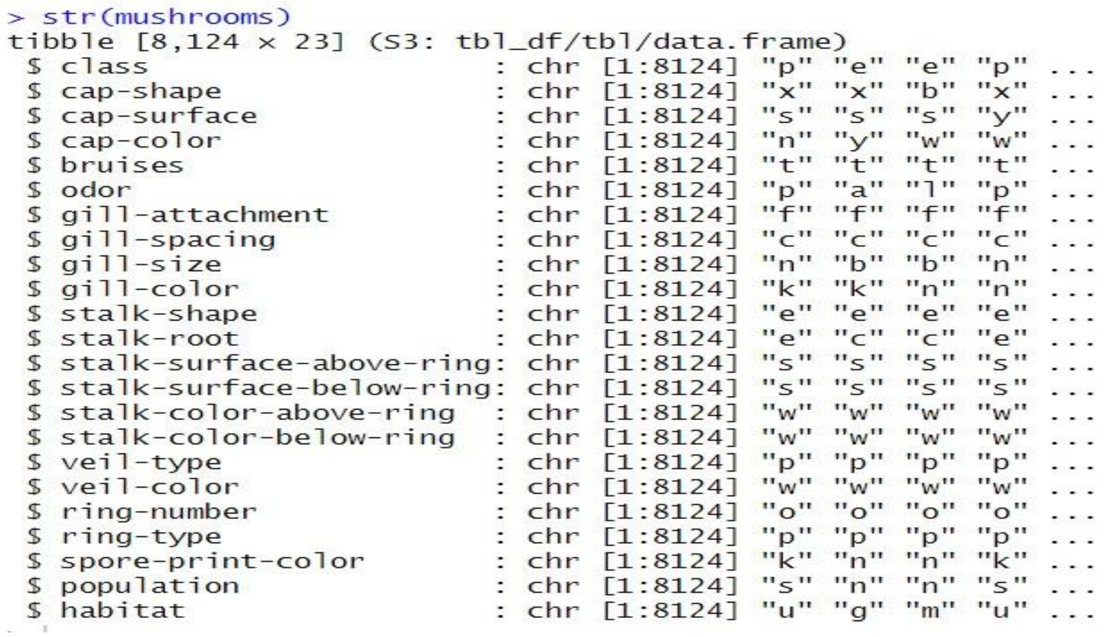
The dataset used in this analysis is derived from an Excel file containing various characteristics of mushroom specimens. Each entry in the dataset includes categorical variables that describe physical attributes and the edibility classification (edible or poisonous). The primary focus is on features such as cap shape, cap color, odor, and gill attachment, among others.

3. Data Preparation

Initially, the dataset was loaded into the R programming environment. The structure of the data was examined to understand the types of variables present. A critical step in data preparation involved identifying and removing redundant variables, specifically the veil.type column, which was deemed unnecessary for the analysis. The dataset was also checked for missing values to ensure a clean dataset for modeling.



```r
# Loading libraries
library(readxl)
library(rpart, quietly = TRUE)
library(caret, quietly = TRUE)
library(rpart.plot, quietly = TRUE)
library(rattle)
library(ggplot2)
# Reading the data set as a dataframe
file_path <- "C:/Users/Mohammed Saif Wasay/Documents/code/data/mushrooms.xlsx"
mushrooms <- read_excel(file_path)

# Structure of the data
str(mushrooms)

# Number of rows with missing values
nrow(mushrooms) - sum(complete.cases(mushrooms))

# Deleting redundant variable `veil.type`
mushrooms$veil.type <- NULL

```



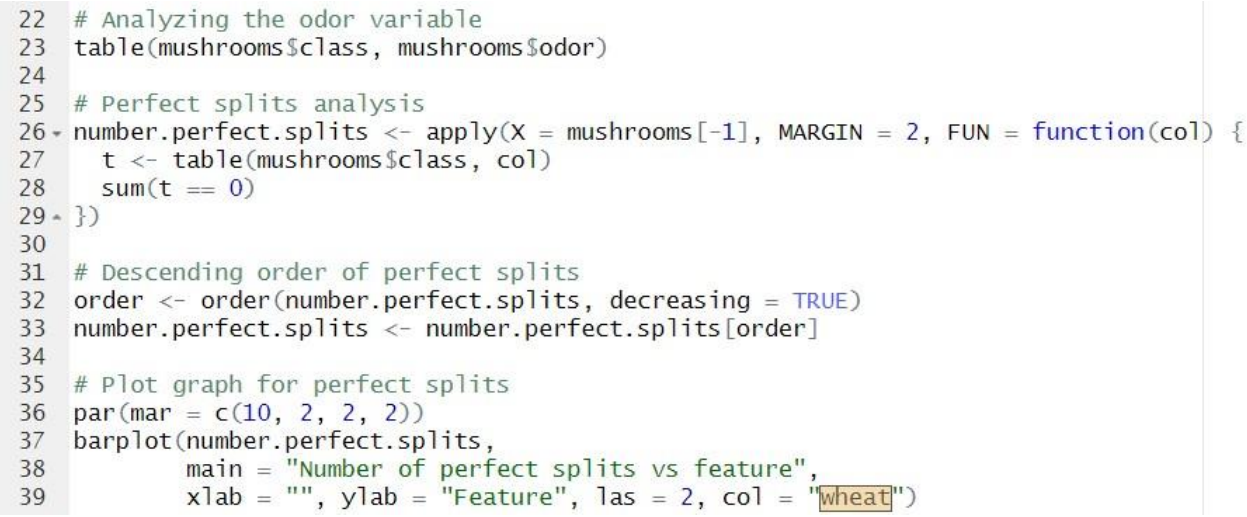
```
> str(mushrooms)
tibble [8,124 × 23] (S3: tbl_df/tbl/data.frame)
 $ class                   : chr [1:8124] "p" "e" "e" "p" ...
 $ cap-shape               : chr [1:8124] "x" "x" "b" "x" ...
 $ cap-surface             : chr [1:8124] "s" "s" "s" "y" ...
 $ cap-color               : chr [1:8124] "n" "y" "w" "w" ...
 $ bruises                 : chr [1:8124] "t" "t" "t" "t" ...
 $ odor                    : chr [1:8124] "p" "a" "l" "p" ...
 $ gill-attachment         : chr [1:8124] "f" "f" "f" "f" ...
 $ gill-spacing            : chr [1:8124] "c" "c" "c" "c" ...
 $ gill-size               : chr [1:8124] "n" "b" "b" "n" ...
 $ gill-color              : chr [1:8124] "k" "k" "n" "n" ...
 $ stalk-shape             : chr [1:8124] "e" "e" "e" "e" ...
 $ stalk-root              : chr [1:8124] "e" "c" "c" "e" ...
 $ stalk-surface-above-ring: chr [1:8124] "s" "s" "s" "s" ...
 $ stalk-surface-below-ring: chr [1:8124] "s" "s" "s" "s" ...
 $ stalk-color-above-ring  : chr [1:8124] "w" "w" "w" "w" ...
 $ stalk-color-below-ring  : chr [1:8124] "w" "w" "w" "w" ...
 $ veil-type               : chr [1:8124] "p" "p" "p" "p" ...
 $ veil-color              : chr [1:8124] "w" "w" "w" "w" ...
 $ ring-number             : chr [1:8124] "o" "o" "o" "o" ...
 $ ring-type               : chr [1:8124] "p" "p" "p" "p" ...
 $ spore-print-color       : chr [1:8124] "k" "n" "n" "k" ...
 $ population              : chr [1:8124] "s" "n" "n" "s" ...
 $ habitat                 : chr [1:8124] "u" "g" "m" "u" ...
```

4. Exploratory Data Analysis (EDA)

Prior to building the classification model, exploratory data analysis was conducted to visualize and understand the relationships between different features and their impact on mushroom edibility. This involved:

4.1. **Creating Cross-Tabulations**: Cross-tabulations were generated for key features against the class (edible or poisonous) to identify patterns and frequencies.

4.2. **Visualizations**: Bar plots were created to depict the distribution of mushroom characteristics, aiding in the identification of significant attributes related to edibility.



```r
# Analyzing the odor variable
table(mushrooms$class, mushrooms$odor)

# Perfect splits analysis
number.perfect.splits <- apply(X = mushrooms[-1], MARGIN = 2, FUN = function(col) {
  t <- table(mushrooms$class, col)
  sum(t == 0)
})

# Descending order of perfect splits
order <- order(number.perfect.splits, decreasing = TRUE)
number.perfect.splits <- number.perfect.splits[order]

# Plot graph for perfect splits
par(mar = c(10, 2, 2, 2))
barplot(number.perfect.splits,
        main = "Number of perfect splits vs feature",
        xlab = "", ylab = "Feature", las = 2, col = "wheat")
```

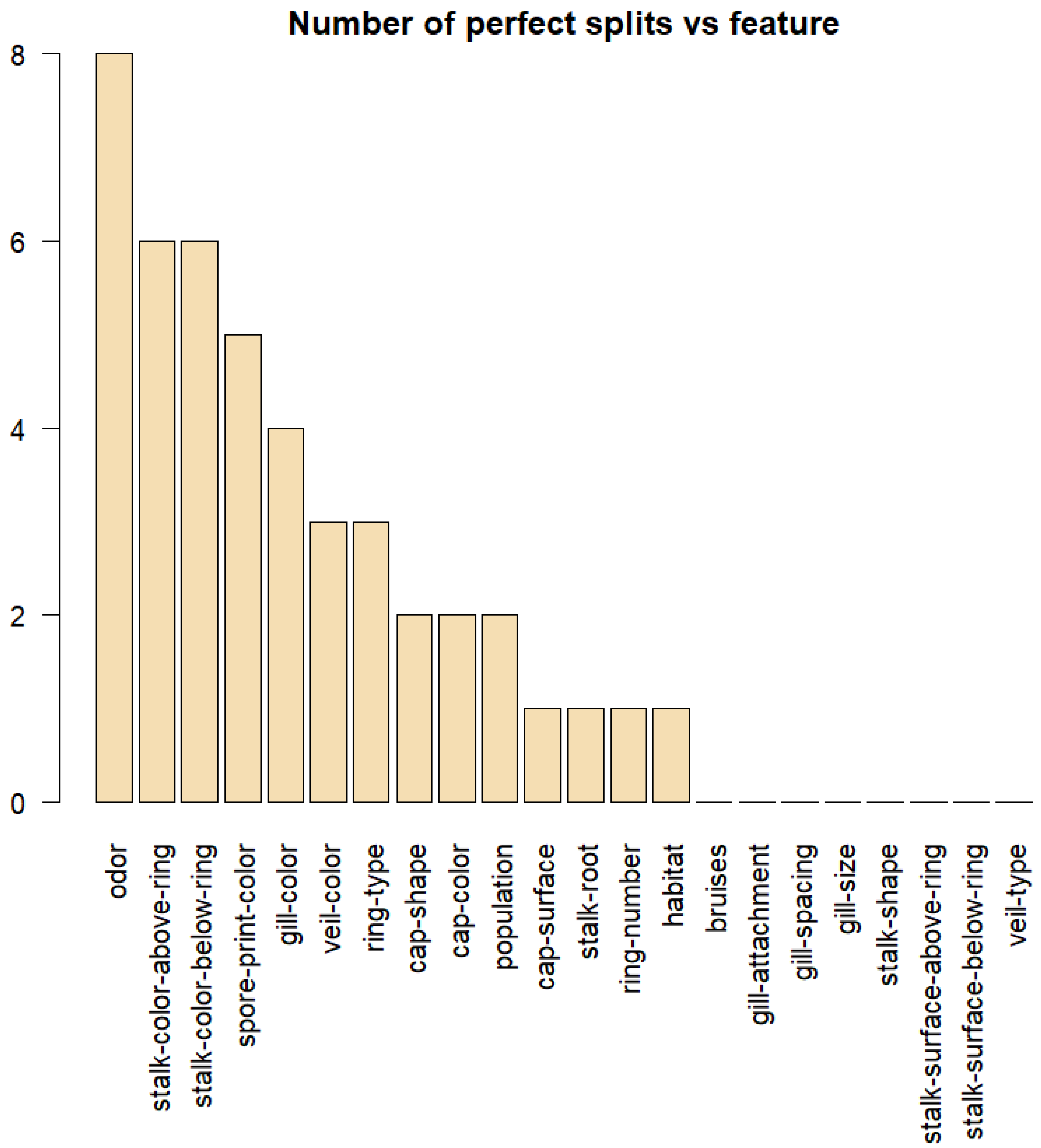


Figure 1: No of Perfect Splits Vs Features.

5. Model Development

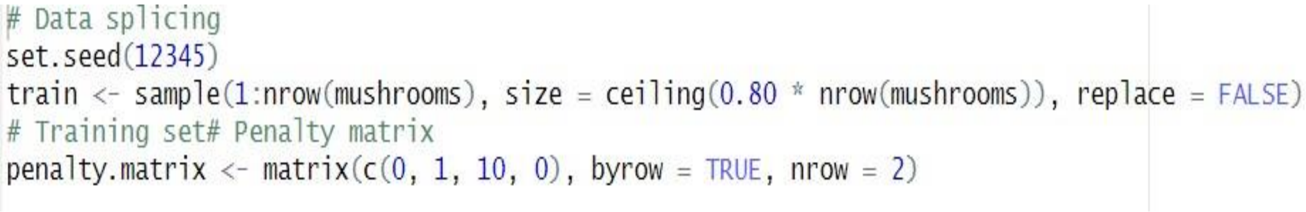
The classification model was built using the R package rpart, which is designed for recursive partitioning and regression trees. The process included the following steps:

5.1. Data Splitting: The dataset was randomly divided into training (80%) and testing (20%) sets to evaluate the model's performance.

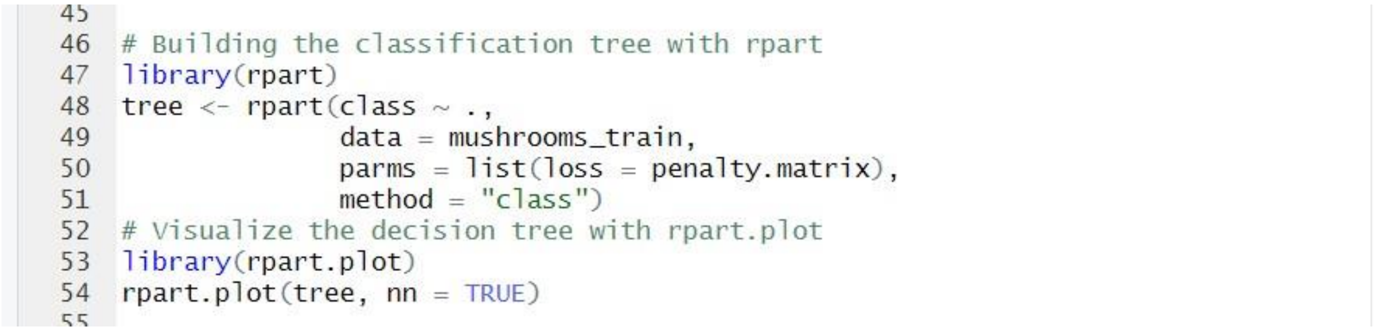
5.2. Building the Decision Tree: A classification tree was constructed using the training data, incorporating a penalty matrix to manage misclassifications, particularly emphasizing the cost of incorrectly classifying edible mushrooms as poisonous.

5.3. Pruning the Tree: To avoid overfitting, the tree was pruned based on the optimal complexity parameter determined from cross-validated error rates.

5.4. Model Testing: Predictions were made on the test set, and a confusion matrix was generated to assess the accuracy and performance of the model.



```
# Data splicing
set.seed(12345)
train <- sample(1:nrow(mushrooms), size = ceiling(0.80 * nrow(mushrooms)), replace = FALSE)
# Training set# Penalty matrix
penalty.matrix <- matrix(c(0, 1, 10, 0), byrow = TRUE, nrow = 2)
```



```
# Building the classification tree with rpart
library(rpart)
tree <- rpart(class ~ .,
              data = mushrooms_train,
              parms = list(loss = penalty.matrix),
              method = "class")
# Visualize the decision tree with rpart.plot
library(rpart.plot)
rpart.plot(tree, nn = TRUE)
```

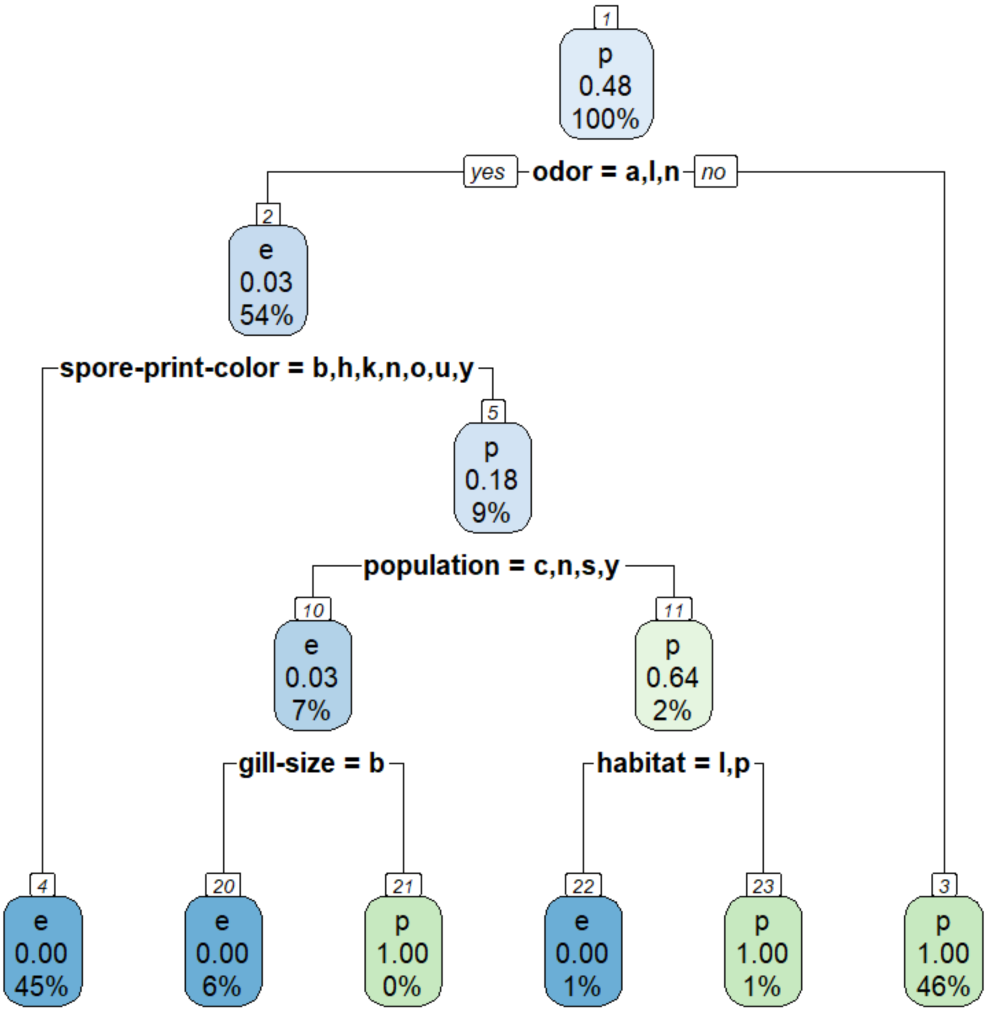
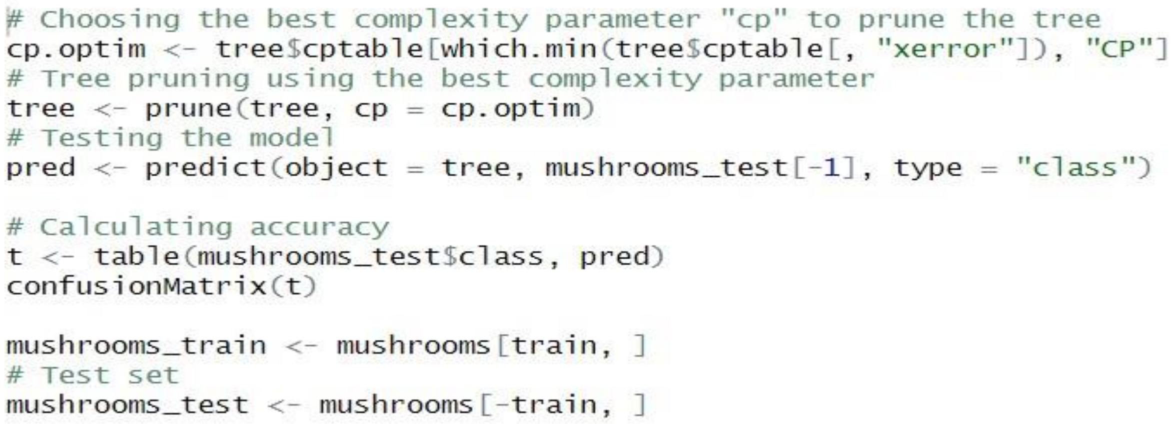
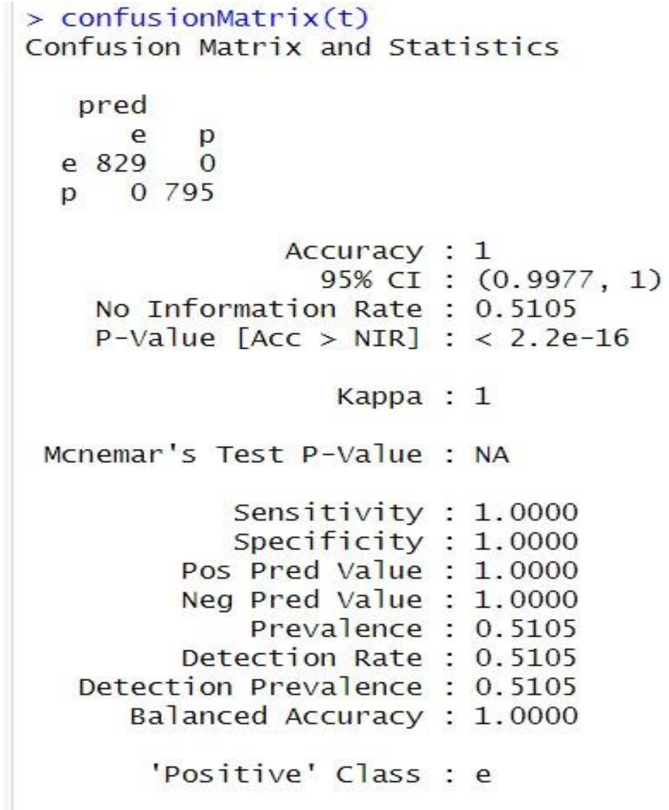


Figure 2: Decision Tree



```
# Choosing the best complexity parameter "cp" to prune the tree
cp.optim <- tree$cptable[which.min(tree$cptable[, "xerror"]), "CP"]
# Tree pruning using the best complexity parameter
tree <- prune(tree, cp = cp.optim)
# Testing the model
pred <- predict(object = tree, mushrooms_test[-1], type = "class")

# Calculating accuracy
t <- table(mushrooms_test$class, pred)
confusionMatrix(t)

mushrooms_train <- mushrooms[train, ]
# Test set
mushrooms_test <- mushrooms[-train, ]
```



```
> confusionMatrix(t)
Confusion Matrix and Statistics

   pred
      e   p
  e 829   0
  p   0 795

               Accuracy : 1
                 95% CI : (0.9977, 1)
    No Information Rate : 0.5105
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 1

 Mcnemar's Test P-Value : NA

            Sensitivity : 1.0000
            Specificity : 1.0000
         Pos Pred Value : 1.0000
         Neg Pred Value : 1.0000
             Prevalence : 0.5105
         Detection Rate : 0.5105
   Detection Prevalence : 0.5105
      Balanced Accuracy : 1.0000

       'Positive' Class : e
```

Summary of Findings

Model Performance

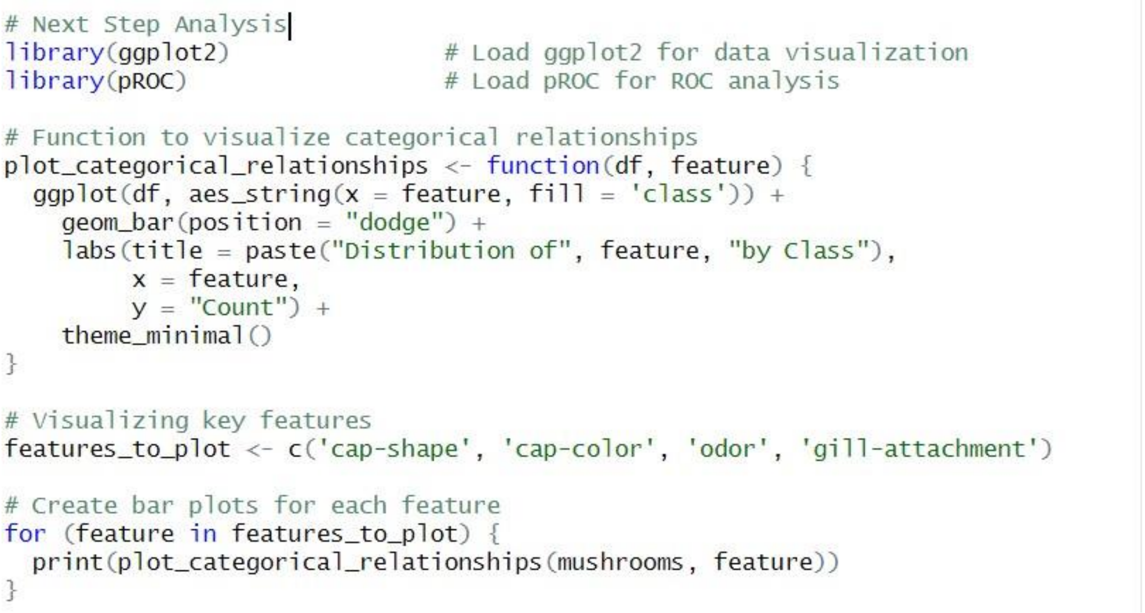
Appendix and Next Steps Analysis

Additional Exploratory Data Analysis (EDA)

To enhance understanding and validate findings, further exploratory analysis should be conducted. This could include:

Visualizing Relationships: Employing the `ggplot2` package to create comprehensive visualizations that highlight relationships between categorical variables and the class label.

Investigating Environmental Factors: Analyzing how environmental factors, such as soil type and moisture conditions, affect mushroom characteristics and edibility.



```r
# Next Step Analysis
library(ggplot2)                  # Load ggplot2 for data visualization
library(pROC)                     # Load pROC for ROC analysis

# Function to visualize categorical relationships
plot_categorical_relationships <- function(df, feature) {
  ggplot(df, aes_string(x = feature, fill = 'class')) +
    geom_bar(position = "dodge") +
    labs(title = paste("Distribution of", feature, "by Class"),
         x = feature,
         y = "Count") +
    theme_minimal()
}

# Visualizing key features
features_to_plot <- c('cap-shape', 'cap-color', 'odor', 'gill-attachment')

# Create bar plots for each feature
for (feature in features_to_plot) {
  print(plot_categorical_relationships(mushrooms, feature))
}
```

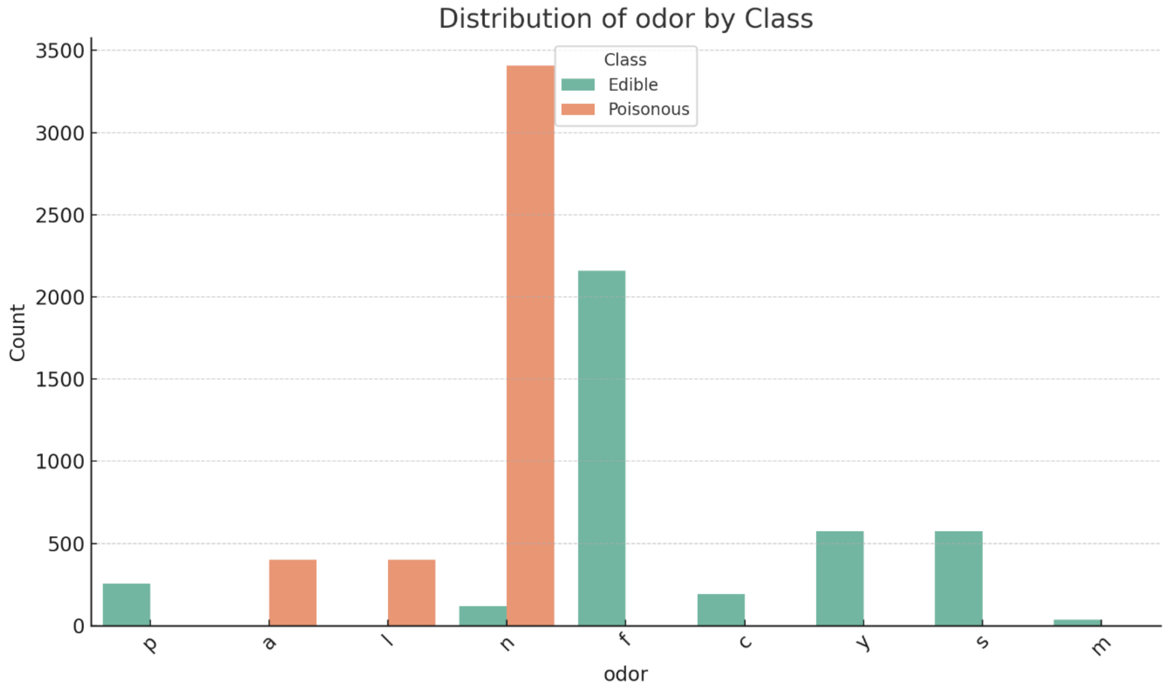


Figure 3: Distribution of Odor by Class

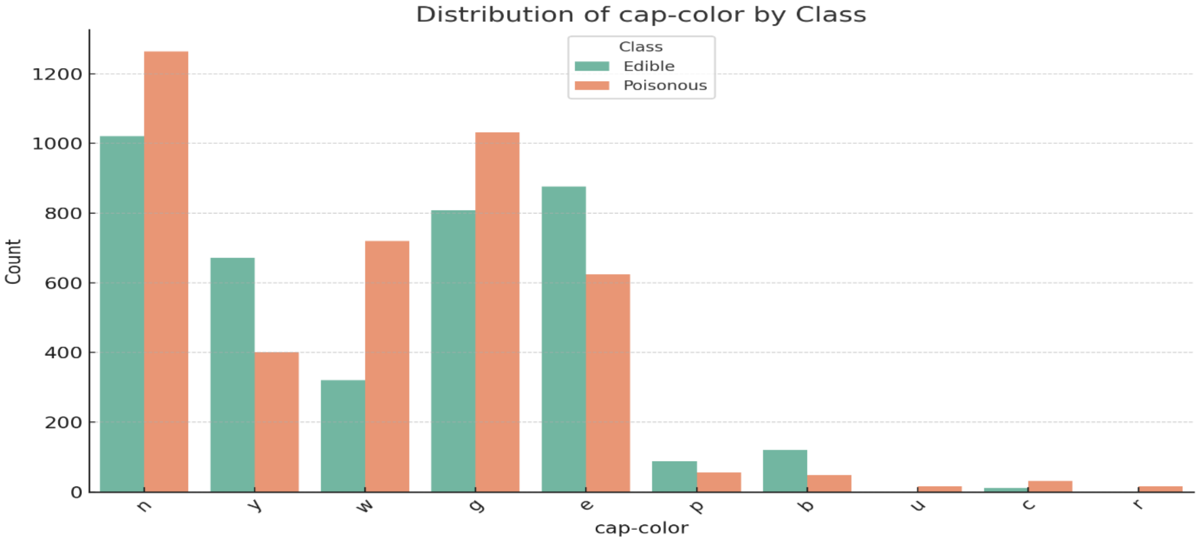


Figure 4: Distribution of Cap-Color by Class

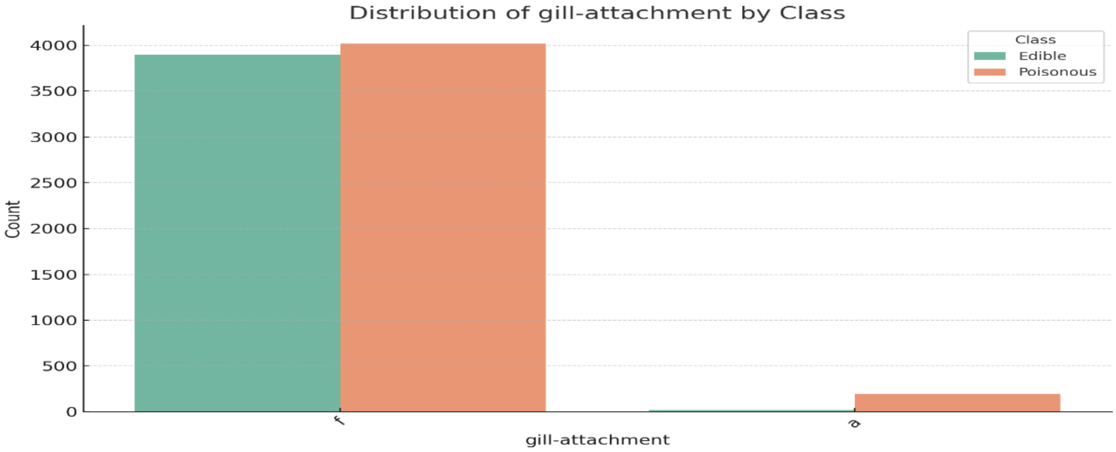


Figure 5: Distribution of gill-attachment by Class

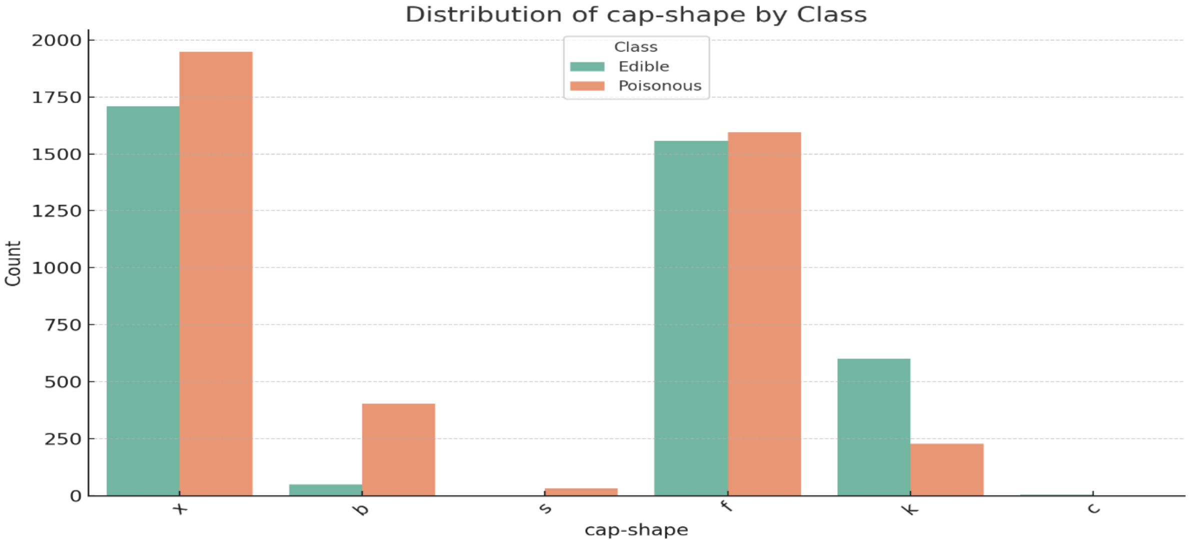


Figure 6: Distribution of Cap-Shape by Class

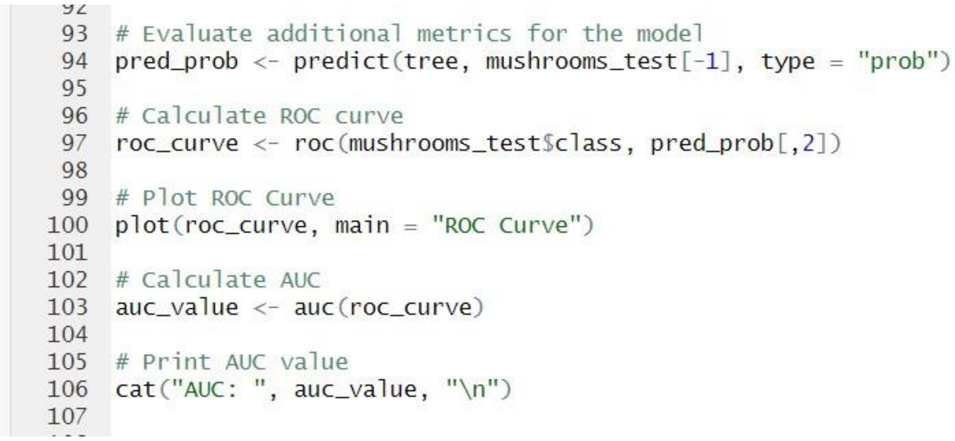
5. Insights from the Visualizations:

1. **Cap Shape**: The shapes 'x' and 'f' appear frequently in both classes, while 'b' shows a strong association with edible mushrooms.
2. **Cap Color**: The colors brown (n) and white (w) are common in both classes, with red (r) and purple (u) exclusively associated with edible mushrooms.
3. **Odor**: The absence of odor (n) is strongly indicative of edibility, while certain strong odors correlate with poisonous mushrooms.
4. **Gill Attachment**: Most mushrooms with free gills (f) belong to both classes, but attached gills (a) appear more frequently in edibles.

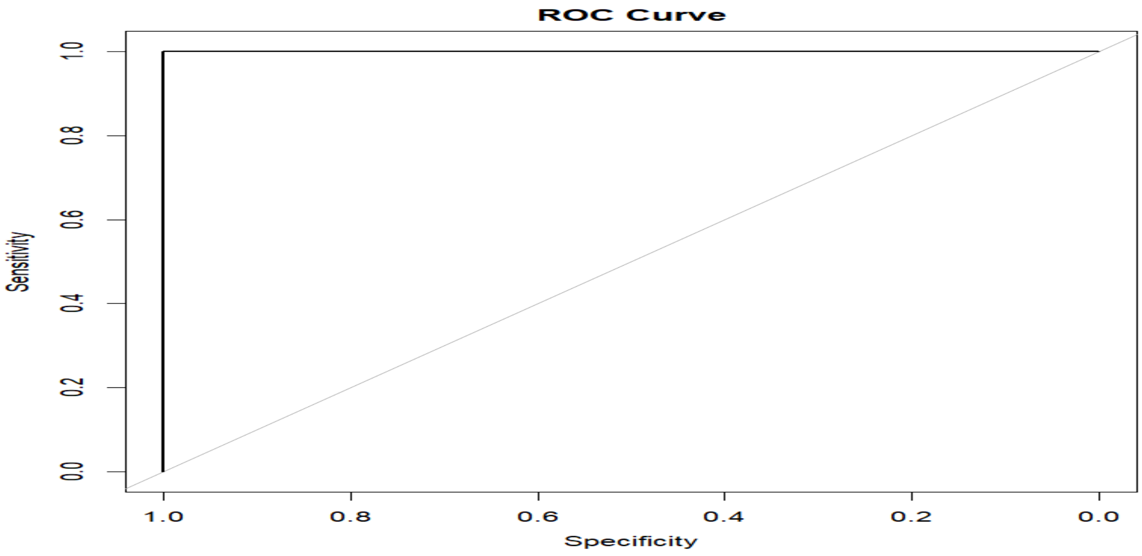
These visualizations help in understanding the relationships between various features and the classification of mushrooms.

6.Model Evaluation Metrics

In addition to the confusion matrix, incorporating metrics such as precision, recall, and F1-score would provide a more nuanced understanding of model performance, particularly in identifying rare classes (poisonous mushrooms). Here, we are incorporationg ROC Curve.



```
# Evaluate additional metrics for the model
pred_prob <- predict(tree, mushrooms_test[-1], type = "prob")

# Calculate ROC curve
roc_curve <- roc(mushrooms_test$class, pred_prob[,2])

# Plot ROC Curve
plot(roc_curve, main = "ROC Curve")

# Calculate AUC
auc_value <- auc(roc_curve)

# Print AUC value
cat("AUC: ", auc_value, "\n")

```



Future Modeling Approaches

Future steps could include:

- **Testing Other Classification Algorithms**: Exploring the effectiveness of other algorithms such as **Random Forest**, **Support Vector Machines (SVM)**, or Gradient Boosting to compare performance metrics.
- **Hyperparameter Tuning**: Implementing techniques such as Grid Search or Random Search for hyperparameter tuning to optimize model performance.
- **Cross-Validation**: Utilizing k-fold cross-validation to ensure that the model generalizes well to unseen data.

Conclusion

This analysis demonstrates the importance of recognizing specific mushroom characteristics to ensure safe foraging. By focusing on features such as cap shape, color, odor, and gill attachment, novice foragers can make informed decisions about mushroom edibility. The insights gained from this study highlight the potential for further research and development of more robust predictive models, ultimately contributing to safer practices in mushroom foraging.

References:

1. R Documentation, An introduction to R. Retrieved 30th September 2024 from https://cran.r-project.org/doc/manuals/r-release/R-intro.html#Related-software-and-documentation
2. Albusairi, F. (2023, March 26). Mastering Simple R Visualizations: From Scatter Plots to Heat Maps. . Retrieved 30th September 2024 from https://www.linkedin.com/pulse/mastering-simple-r-visualizations-from-scatterplots-heat-albusairi/.