Module 2 | Technique Practice

Trang Tran

CPS, Northeastern University

ALY6040 | Data Mining

Professor Winnie Li

Apr 23, 2023

 # Analysis and Interpretation

This data set provides descriptions of mushroom samples which includes 8124 observations and 23 variables. The provided code performs various tasks related to building and evaluating a classification tree model for the mushroom dataset:

- After checking the structure of the dataset using [str()] function, we see that all variables are categorical data.

```
> 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" ...
```

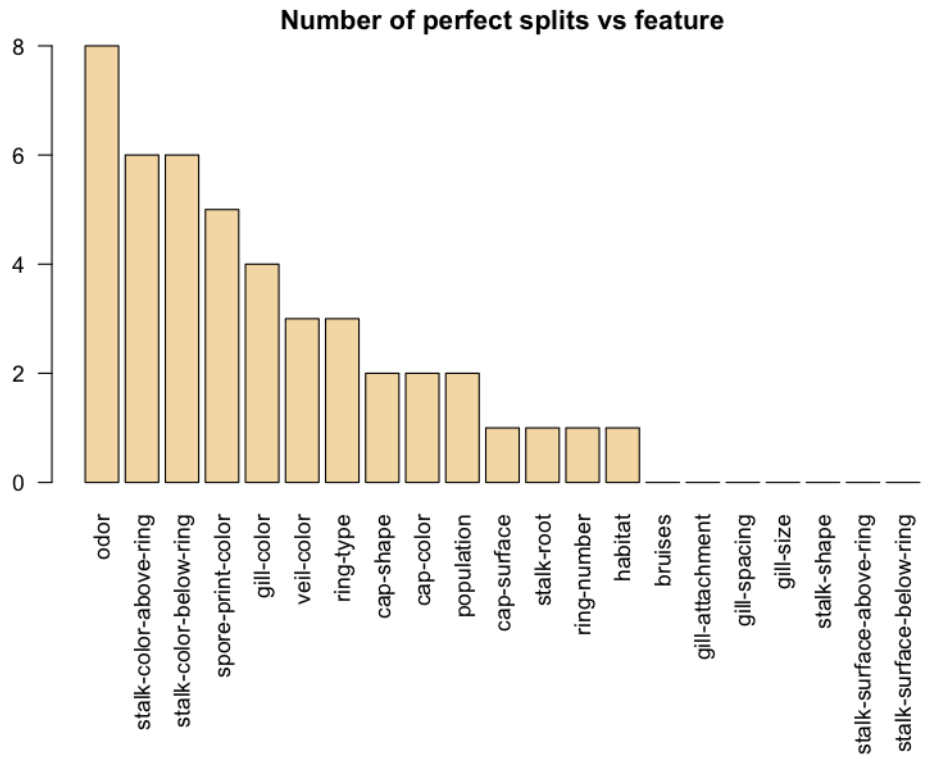
- The number of rows with missing values (= 0) is calculated by subtracting the number of rows with complete cases from the total number of rows in the data frame.

```
> # number of rows with missing values
> nrow(mushrooms) - sum(complete.cases(mushrooms))
[1] 0
```

- We delete the ‘veil-type’ column (a redundant variable) because it has only one unique value ‘p’ that makes no meaning in classification.

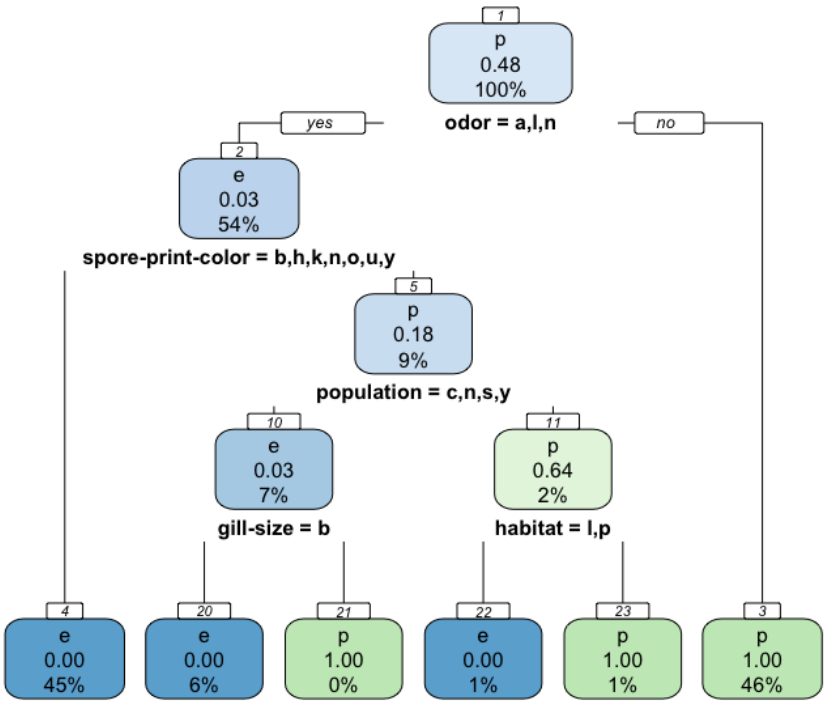
- The [apply] function is used to calculate the number of perfect splits for each variable in the data frame, excluding the class variable. The variable with the highest number of perfect splits is considered the best variable for classification.

Figure 1



- The bar plot above is created to visualize the number of perfect splits for each variable in the dataset, with the variables sorted in descending order based on the number of perfect splits. We capture that the variable ‘odor’ has the highest perfect split number (8).
- The dataset is randomly split into training and testing sets using the sample function. The training set contains 80% of the data, and the testing set contains the remaining 20%.
- A penalty matrix is created to be used for the classification tree using the matrix function.
- The [rpart] function is used to build the classification tree model with the class variable as the response and all other variables as predictors. The rpart.plot function is used to visualize the decision tree.

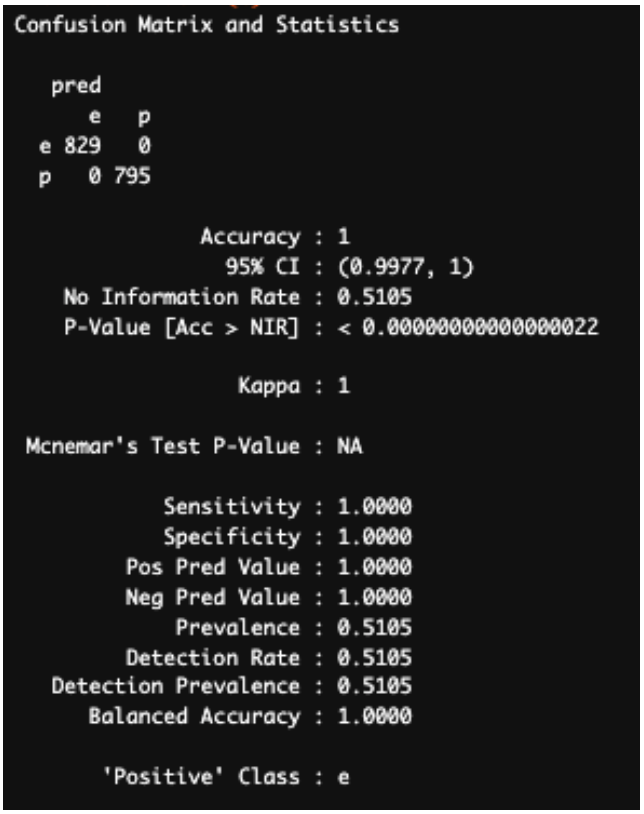
Figure 2: Decision Tree



- The optimal complexity parameter (cp) for pruning the tree is determined using the [cptable] function and the which.min function. It returns the row number of the cptable matrix where the cross-validated error (xerror) is the smallest. The cptable matrix contains information about the cost-complexity parameter (CP) for each node in the classification tree, and the corresponding values of the misclassification error, the cross-validated error, and the complexity parameter used for pruning. The [prune] function is used to prune the tree using the optimal complexity parameter.
- Finally, the tree model is tested using the [predict] function on the testing set, and the accuracy is calculated using the [confusionMatrix] function. In this case, the confusion matrix shows that all of the 829 samples in the 'e' class were correctly predicted as 'e', and all of the 795 samples in the 'p' class were correctly predicted as 'p'. There are no false

positives or false negatives, so the model achieved perfect accuracy. The rest of the output shows various performance metrics for the model, including sensitivity, specificity, positive predictive value (PPV), negative predictive value (NPV), prevalence, detection rate, and detection prevalence.

Figure 3: Confusion Matrix and Statistics on testing model



- Overall, this output indicates that the classification model performed very well on the test dataset, achieving perfect accuracy and high values for other performance metrics. However, it's important to note that the high performance on this test dataset does not guarantee 100% that the model will perform as well on new, unseen data.

 # Recommendations

The analysis provides some insights into the most important features in predicting the edibility of mushrooms. The bar plot of perfect splits vs feature shows that odor is the most important feature in determining whether a mushroom is edible or poisonous, with 8 perfect splits. This implies that certain types of odors are highly indicative of the mushroom's edibility, while others are highly indicative of its toxicity. Other important features include ‘stalk-color-above-ring’, ‘stalk-color-below-ring’, ‘spore-print-color’, and ‘gill-color’.

Based on these results, it is recommended that mushroom growers, sellers, and consumers should pay close attention to the odor and other features of mushrooms when determining their edibility. It may also be useful to incorporate additional features into the analysis, such as geographic location, soil types, and growing conditions, to better understand the combined factors that influence mushroom poisonousness. Additionally, the model could be further refined and tested using additional datasets to ensure its accuracy and generalizability.

References

1. UCI. Mushroom Data Set. Retrieved April 23, 2023 from.

   https://archive.ics.uci.edu/ml/datasets/mushroom