Project report on Classification of mushrooms according to their edibility

Ivan Smirnov, Maryana Smirnova

Thursday 6th April, 2023

1 Introduction

Mushrooms are a popular food item around the world. However, some mushrooms are poisonous and can cause serious health problems or even death if consumed. In this project, the aim is to uncover the key indicators that differentiate between edible and poisonous mushrooms. The goal is to implement a machine learning model that can leverage these attributes to accurately classify mushrooms and provide valuable insights for safe mushroom hunting.

To accomplish this objective, an initial step involves conducting an in-depth exploration of the dataset and performing exploratory data analysis (EDA) to gain insights into the distribution of features and their relationship with the target variable. Subsequently, data preprocessing techniques are employed to address missing values, encode categorical variables, and scale numerical features. Lastly, a variety of machine learning models are trained on the preprocessed data, and their performance is assessed using diverse evaluation metrics.

1.1 Dataset

The dataset used in this project is called the "Secondary mushroom dataset," created by Dennis Wagner. It consists of 61,069 hypothetical mushrooms with caps, representing 173 species (353 mushrooms per species). Each mushroom is classified as definitely edible, definitely poisonous, or of unknown edibility and not recommended (combined with the poisonous class). The dataset contains 20 variables, including 17 nominal and 3 metrical variables (cap-diameter, stem-height, and stem-width).

The creation of this dataset was inspired by Jeff Schlimmer's Mushroom Data Set from the UCI Machine Learning repository [Schlimmer, 1987], which simulated data for 23 species. However, our dataset expands on this by including mushrooms from Patrick Hardin's source book "Mushrooms & Toadstools" [Hardin, 1999].

The dataset consists of features that describe different characteristics of mushrooms, including cap-shape and cap-color, veil-type, and gill-attachment. The dataset is divided into two distinct classes: edible and poisonous. The "Secondary mushroom dataset" was generated using Python scripts, which involved randomizing both nominal and metrical variables [Wagner, 2020].

The dataset exhibits a balanced distribution with 45% (27,181) edible instances and 55% (33,888) poisonous instances. While accuracy may be sufficient for evaluating model performance, additional metrics such as precision, recall, and ROC-AUC will be employed to thoroughly assess model effectiveness and ensure a more robust evaluation considering the inherent class imbalance in the dataset.

The following is a detailed overview of the dataset's features:

Class Information:

- **class**: poisonous (p), edible (e)

Features Information:
(n: nominal, m: metrical; nominal values as sets of values)

1. **cap-diameter (m)**: float number in cm
2. **cap-shape (n)**: bell (b), conical (c), convex (x), flat (f), sunken (s), spherical (p), others (o)
3. **cap-surface (n)**: fibrous (i), grooves (g), scaly (y), smooth (s), shiny (h), leathery (l), silky (k), sticky (t), wrinkled (w), fleshy (e)
4. **cap-color (n)**: brown (n), buff (b), gray (g), green (r), pink (p), purple (u), red (e), white (w), yellow (y), blue (l), orange (o), black (k)
5. **does-bruise-bleed (n)**: bruises or bleeding (t), no (f)
6. **gill-attachment (n)**: adnate (a), adnexed (x), decurrent (d), free (e), sinuate (s), pores (p), none (f), unknown ($?$)
7. **gill-spacing (n)**: close (c), distant (d), none (f)
8. **gill-color (n)**: see cap-color + none (f)
9. **stem-height (m)**: float number in cm

10. **stem-width (m)**: float number in mm
11. **stem-root (n)**: bulbous (b), swollen (s), club (c), cup (u), equal (e), rhizomorphs (z), rooted (r)
12. **stem-surface (n)**: see cap-surface + none (f)
13. **stem-color (n)**: see cap-color + none (f)
14. **veil-type (n)**: partial (p), universal (u)
15. **veil-color (n)**: see cap-color + none (f)
16. **has-ring (n)**: ring (t), none (f)
17. **ring-type (n)**: cobwebby (c), evanescent (e), flaring (r), grooved (g), large (l), pendant (p), sheathing (s), zone (z), scaly (y), movable (m), none (f), unknown ($?$)
18. **spore-print-color (n)**: see cap color
19. **habitat (n)**: grasses (g), leaves (l), meadows (m), paths (p), heaths (h), urban (u), waste (w), woods (d)
20. **season (n)**: spring (s), summer (u), autumn (a), winter (w)

2 Data

The data section describes the dataset, including its size, structure, and preprocessing steps such as removing duplicates and missing values, and selecting relevant features. Exploratory analysis was conducted to identify informative features values for mushroom identification.

2.1 Data prepossessing

During the initial stage of the project, the dataset was thoroughly explored to address any issues related to duplicates and missing values. The class distribution remained balanced at a ratio of 45/55. However, it was observed that several features exhibited a significant number of null values, exceeding 50,000 in count. Despite attempts to drop these features, all rows were lost as a result. Consequently, 9 features, including cap-surface, gill-attachment, gill-spacing, stem-root, stem-surface, veil-type, veil-color, ring-type, and spore-print-color, were eliminated from the analysis. Ultimately, the dataset was refined to include 11 remaining

features: cap-diameter, cap-shape, cap-color, does-bruise-or-bleed, gill-color, has-ring, stem-color, stem-width, stem-height, habitat, and season.

After **preliminary** cleaning and preprocessing the dataset, various machine learning models were trained on the data. All models showed exceptional performance, particularly the K-nearest neighbors (KNN) model, which achieved approximately 99.99% accuracy. Due to the unusually high accuracy, further investigation was conducted to validate the results and identify potential issues or biases in the modeling process.

Further analysis revealed that removing problematic features caused numerous duplicate nominal values to emerge within the dataset, as shown in Figure 1. As a result, all duplicate instances were eliminated during the final stage of data preprocessing, reducing the dataset size to 3393 instances from the original 60923, with 11 remaining features. This caused the class distribution to become more imbalanced, with a 60/40 ratio of poisonous to edible mushrooms. Despite the increased imbalance, accuracy remained an acceptable evaluation metric, but additional metrics such as precision, recall, and AUC were used to further validate model performance.

class	cap-diameter	cap-shape	cap-color	does-bruise-or-bleed	gill-color	has-ring	stem-color	stem-width	stem-height	habitat	season
p	14.07	x	o	f	w	t	w	17.74	17.80	d	w
p	14.64	x	o	f	w	t	w	17.20	16.53	d	w
p	15.34	x	o	f	w	t	w	18.79	17.84	d	u
p	13.55	f	e	f	w	t	w	16.88	16.04	d	w
p	13.40	x	o	f	w	t	w	17.14	17.95	d	u
p	17.37	x	o	f	w	t	w	18.27	18.10	d	u
p	16.56	x	o	f	w	t	w	18.11	18.89	d	u
p	15.54	f	e	f	w	t	w	17.87	18.26	d	a
p	15.19	x	e	f	w	t	w	17.67	17.42	d	a
p	16.16	x	o	f	w	t	w	18.90	19.46	d	w

Figure 1: Illustration of nominal duplicate instances

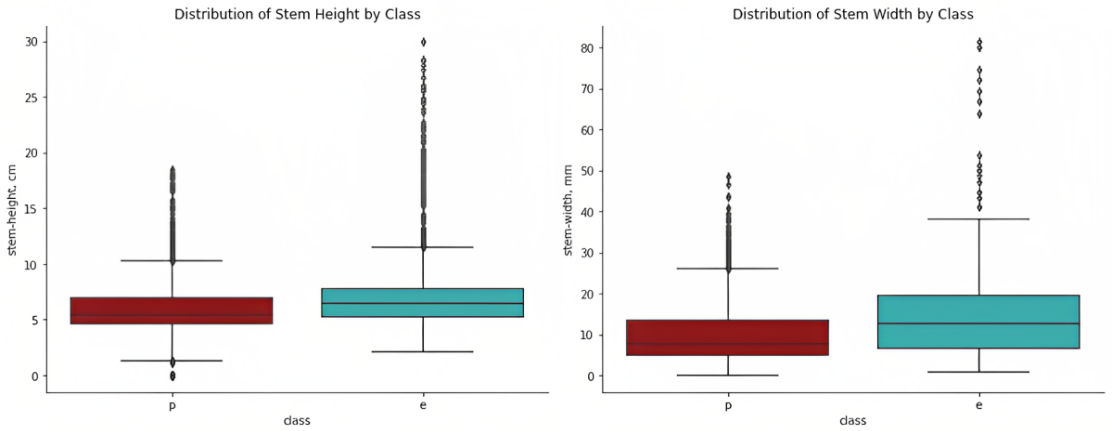
After completing the process of discovering and preprocessing the dataset, all nominal values were appropriately labeled, rendering the dataset ready for training the machine learning models. However, prior to commencing the training phase, a preliminary data analysis was conducted on the unlabeled data to gain insights and assess its characteristics.

2.2 Data analysis

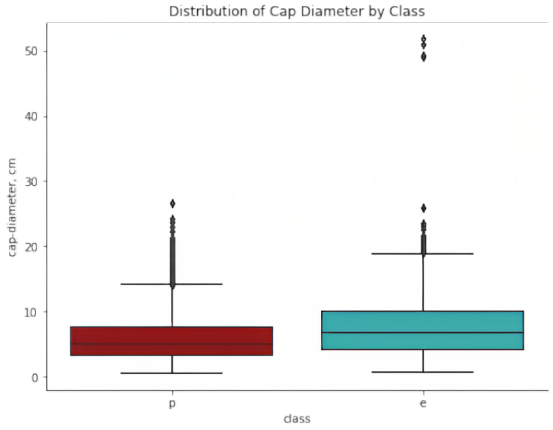
During the exploratory data analysis phase, the primary objective was to identify the most informative feature values for distinguishing between edible and poisonous

mushrooms. This was achieved through comprehensive analysis using various plots and meticulous data exploration.

Exploring the data and examining the box plots (refer to Figure 2) revealed several observations about the relationship between certain numerical features and mushroom edibility. For stem-height, heights exceeding 20cm strongly suggest edibility, while the smallest stem-heights of 1-2cm indicate poisonousness. Similarly, for stem-width, larger values starting from 50mm tend to indicate edibility. For cap-diameter, no significant differences were observed within a close range, but an extreme value of approximately 50cm suggests edibility.



(a) Distribution of stem-height and stem-width by class



(b) Distribution of cap-diameter by class

Figure 2: Boxplot comparison of edible (e) and poisonous (p) mushrooms for stem-height, stem-width and cap-diameter

After analyzing numerical values, the focus shifted to examining nominal values within the dataset. For each unique value within the features, excluding

'cap-diameter,' 'stem-height,' and 'stem-width,' the difference in percentage between edible and poisonous mushrooms was calculated. This aimed to identify key indicators for classification and reveal the most effective attribute values for distinguishing between edible and poisonous mushrooms.

Certain feature values had high percentages of either poisonous or edible mushrooms based on their colors. For example, cap-shapes 'bell' (b) and 'other' (o) had 91.38% and 83.33% poisonous mushrooms, respectively. Cap-colors 'black' (k) and 'green' (r) had 84.54% and 86.96% poisonous mushrooms, respectively. Stem-colors 'black' (k), 'green' (r), 'red' (e), and 'yellow' (y) had 97.75%, 82.61%, 93.48%, and 87.81% poisonous mushrooms, respectively. Conversely, cap-color 'buff' (b) had 80.95% edible mushrooms, and season 'winter' (w) had 74.82% edible mushrooms. These findings suggest that certain values of cap-shape, cap-color, stem-color, and season can be strong indicators of mushroom edibility.

These findings are summarized in Table 1:

Class	**Feature**	**Feature Value**	**Percentage**
p	cap-shape	b	91.38%
p	cap-shape	o	83.33%
p	cap-color	k	84.54%
p	cap-color	r	86.96%
p	does-bruise-or-bleed	t	76.15%
p	gill-color	y	77.41%
p	stem-color	k	97.75%
p	stem-color	l	82.61%
p	stem-color	r	93.48%
p	stem-color	y	87.81%
e	cap-color	b	80.95%
e	season	w	74.82%

Table 1: Percentage comparison of edible (e) and poisonous (p) mushrooms for specific feature values

Bar plots were created to show the size differences between edible and poisonous mushrooms across specific features, providing a visual representation. An example is the cap-color bar plot in Figure 3. These plots give readers a comprehensive overview and help them better understand the analysis results.

The examination of the data revealed differences in specific feature values, such as stem-color, cap-color, cap-shape, does-bruise-or-bleed, and gill-color, between edible and poisonous mushrooms. These differences highlight the potential of these

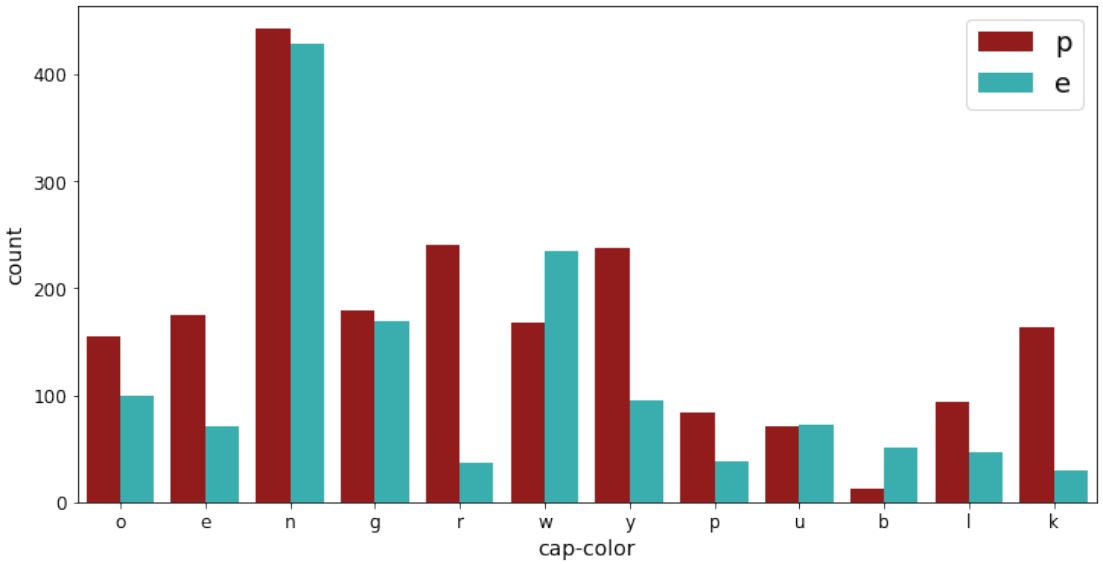


Figure 3: Barplot of mushroom class distribution for cap-color

feature values as indicators of mushroom edibility. For instance, cap color 'buff' and season 'winter' had high percentages of edible mushrooms, while other feature values were more commonly associated with poisonous mushrooms. These findings offer valuable insights into distinguishing attributes of edible and poisonous mushrooms based on specific feature values.

3 Method

This section presents the methodology employed for the analysis of a specific dataset, focusing on the utilization of various machine learning models and the evaluation of their performance.

To ensure a reliable evaluation of model performance, the dataset was divided into a training set and a test set with a ratio of 80/20, respectively. The training set was further used for hyperparameter tuning and model selection through cross-validation. The following machine learning (ML) models were chosen for analysis: Random Forest Classifier (RFC), Linear Discriminant Analysis (LDA), Quadratic Discriminant Analysis (QDA), k-Nearest Neighbors (KNN), and Support Vector Machines (SVM).

3.1 Models choice

The selection of these models was based on their diverse characteristics and potential strengths in handling the mushroom dataset. RFC model was chosen for its ensemble learning approach, which combines multiple decision trees to improve accuracy and robustness. Additionally, RFC is well-suited for handling binary and categorical features, making it suitable for the mushroom dataset.

LDA and QDA were chosen based on their discriminant analysis nature. LDA assumes normally distributed classes with equal covariance matrices, while QDA relaxes this assumption, allowing for different covariance matrices for each class. These models were selected to explore the effects of linear and quadratic decision boundaries, respectively, on the classification performance.

KNN is a non-parametric classification algorithm that assigns labels based on the majority vote of the nearest neighbors in the feature space. It was chosen to evaluate the impact of different values of k on the classification performance and to compare the results of KNN on our initial and preprocessed datasets.

SVM, specifically with the Radial Basis Function (RBF) kernel, was also included as a model of interest. SVMs aim to find an optimal hyperplane that maximally separates classes in the feature space. By tuning the hyperparameters, such as the regularization parameter C and the kernel parameter gamma, the performance of the SVM model was assessed.

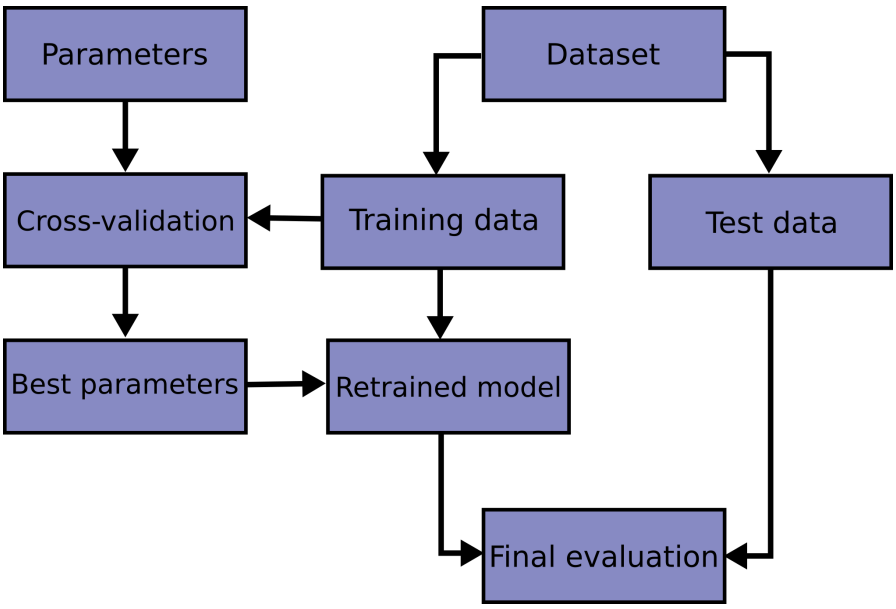


Figure 4: Illustration of cross-validation in our hyperparameter tuning process

3.2 Hyperparameters tuning

To assess the performance of the machine learning (ML) models and optimize their hyperparameters, cross-validation on the training set was employed. This technique helps mitigate issues related to overfitting and provides a more reliable evaluation of the models' generalization capabilities.

In our study, k-fold cross-validation was utilized, where the dataset is divided into k equally sized folds. Each fold serves as a validation set once, while the remaining k-1 folds are used for training the model. This process is repeated k times, ensuring that each fold is used as the validation set exactly once. The results from each iteration are then averaged to obtain a robust estimate of the model's performance. In our case, k is 5, resulting in a 5-fold cross-validation approach.

Figure 4 illustrates the process of k-fold cross-validation. The image is adapted from the scikit-learn documentation [Pedregosa et al., 2011].

3.3 Examples

Table 2 displays the cross-validated accuracy and area under the ROC curve for KNN models with different values of k. As k increases, the model becomes less sensitive to noisy data but may oversmooth the decision boundaries, potentially resulting in lower accuracy. In our case, it was observed that the KNN model achieved the highest cross-validated accuracy and AuROC when k was set to 3, with a CV accuracy of 0.8895 and CV AuROC of 0.9432. Overall, These results suggest that a moderate value of k, such as 3, strikes a good balance between capturing local patterns and avoiding overfitting.

k	**CV Accuracy**	**CV AuROC**
1	0.9049	0.9041
2	0.8596	0.9334
3	0.8895	0.9432
4	0.8666	0.9416
5	0.8670	0.9377
6	0.8545	0.9366
7	0.8596	0.9343
8	0.8489	0.9318
9	0.8475	0.9275
10	0.8419	0.9250

Table 2: Cross-Validated performance of K-Nearest Neighbors with varying k. The gray row represents the model which will be used for test predictions.

Since the KNN algorithm assigns the class label based on the majority vote of its k nearest neighbors, the class probabilities are determined by the fraction of neighbors belonging to each class, which are then used to calculate the AuROC.

Additionally, Table 3 presents the cross-validated accuracy and cross-validated AuROC for Support Vector Machines (SVM) models with different values of the regularization parameter C and the kernel parameter gamma.

C	**Gamma**	**CV Accuracy**	**CV AuROC**
1	0.1	0.9101	0.9601
5	0.1	0.9175	0.9669
10	0.1	0.9170	0.9668
100	0.1	0.9167	0.9665

Table 3: Cross-Validated performance of Support Vector Machines with varying C and Gamma. The gray row represents the model which will be used for test predictions.

In SVM, the C parameter controls the regularization strength, while the gamma parameter defines the kernel width.

For C = 1 and gamma = 0.1, the model achieves a CV accuracy of 0.9101 and a CV AuROC of 0.9601. This indicates that the SVM with a lower regularization strength and a moderate kernel width performs reasonably well, but there is potential for improvement.

Increasing the value of C to 5 while keeping gamma at 0.1 results in improved performance. This suggests that a slightly stronger regularization with a similar kernel width leads to better generalization and predictive performance.

Interestingly, when C is further increased to 10 and 100, CV accuracy and CV AuROC don't improve significantly. This suggests that a very high regularization strength does not yield substantial improvements in performance compared to the previous settings.

Table 4 showcases the performance of Random Forest Classifier (RFC) with different values of max_depth and n_estimators.

From the results, following trends can be observed:

1. Increasing the number of estimators generally leads to slightly higher mean ROC and mean accuracy scores. This indicates that increasing the number of decision trees in the forest improves the model's overall performance.

2. When comparing different max_depth values, it is observed that higher values (e.g., depth 20) tend to have slightly lower mean accuracy compared to depth

max_depth	n_estimators	CV AuROC	CV Accuracy
15	500	0.99386	0.96868
15	1000	0.99389	0.96720
15	2000	0.99404	0.96794
20	500	0.99412	0.96683
20	1000	0.99429	0.96794
20	2000	0.99434	0.96757
None	100	0.99422	0.96646

Table 4: Performance of Random Forest Classifier with varying max_depth and n_estimators. The gray row represents the model which will be used for test predictions.

15. This suggests that increasing the max_depth beyond a certain point may lead to overfitting or increased variance in the model's predictions.

However, it's important to note that the differences in performance between different max_depth values and n_estimators are relatively small.

In summary, the Random Forest Classifier demonstrates strong performance across various combinations of max_depth and n_estimators. Increasing the number of estimators generally improves the model's performance, while the impact of max_depth on performance is less pronounced.

4 Results and Discussion

This section presents the results of the analysis conducted using the ML models mentioned in the methodology section. The evaluation metrics, including accuracy, AuROC, and confusion matrices, provide insights into the models' performance on the test set.

4.1 Models Performance

The performance of each ML model was evaluated using various metrics. Table 5 shows the accuracy scores on a test set for different models.

The analysis of the ML models revealed varying performance across different models. The RFC models exhibited the highest accuracy scores, suggesting their effectiveness in predicting the edibility of mushrooms. This method justified itself, as it was expected at the stage of data analysis. The ensemble learning approach of RFC, combining multiple decision trees, allowed for capturing complex interactions between attributes and led to accurate predictions.

Model	Accuracy	AuROC
RFC (max_depth=20, n_estimators=2000)	0.979	0.998
SVM (C=5, gamma=0.1)	0.951	0.989
KNN (k=3)	0.925	0.966
QDA	0.748	0.813
LDA	0.732	0.763

Table 5: Accuracy and ROC scores of chosen models

On the other hand, LDA and QDA demonstrated moderate to low accuracy scores. These models may have struggled to capture the intricate relationships among attributes, leading to reduced performance.

KNN achieved good accuracy when the number of neighbors was set to 3, but its performance deteriorated as the value of k increased. This sensitivity to the choice of neighbors suggests that KNN's effectiveness heavily relies on the local patterns in the dataset.

SVM demonstrated competitive performance, with accuracy scores exceeding 0.95 for different values of the regularization parameter C and the kernel parameter gamma. The high AuROC values suggest that SVM was successful in discriminating between edible and poisonous mushrooms, contributing to its overall strong performance.

In conclusion, the RFC models and SVM exhibited superior performance in predicting the edibility of mushrooms, while LDA, QDA, and KNN demonstrated varying degrees of effectiveness. The analysis of evaluation metrics, including accuracy, AuROC, and confusion matrices, provided valuable insights into the models' performance and highlighted their strengths and limitations. These findings contribute to the understanding of the ML models' applicability for classifying mushrooms based on their attributes.

4.2 Receiver Operating Characteristic (ROC) Analysis

To further evaluate the performance of the ML models, ROC analysis was conducted. Figure 5 shows the ROC curves for the different models.

The ROC curves provide insights into the trade-off between the true positive rate (sensitivity) and the false positive rate (1 - specificity) for different classification thresholds. A model with a higher AuROC indicates a better ability to distinguish between the classes.

The RFC models exhibited high AuROC values, indicating excellent discrimination between edible and poisonous mushrooms. SVM and KNN showed moderate AuROC values, suggesting a relatively good ability to distinguish between

the classes. LDA and QDA demonstrated lower AuROC values, indicating a lower discriminative power.

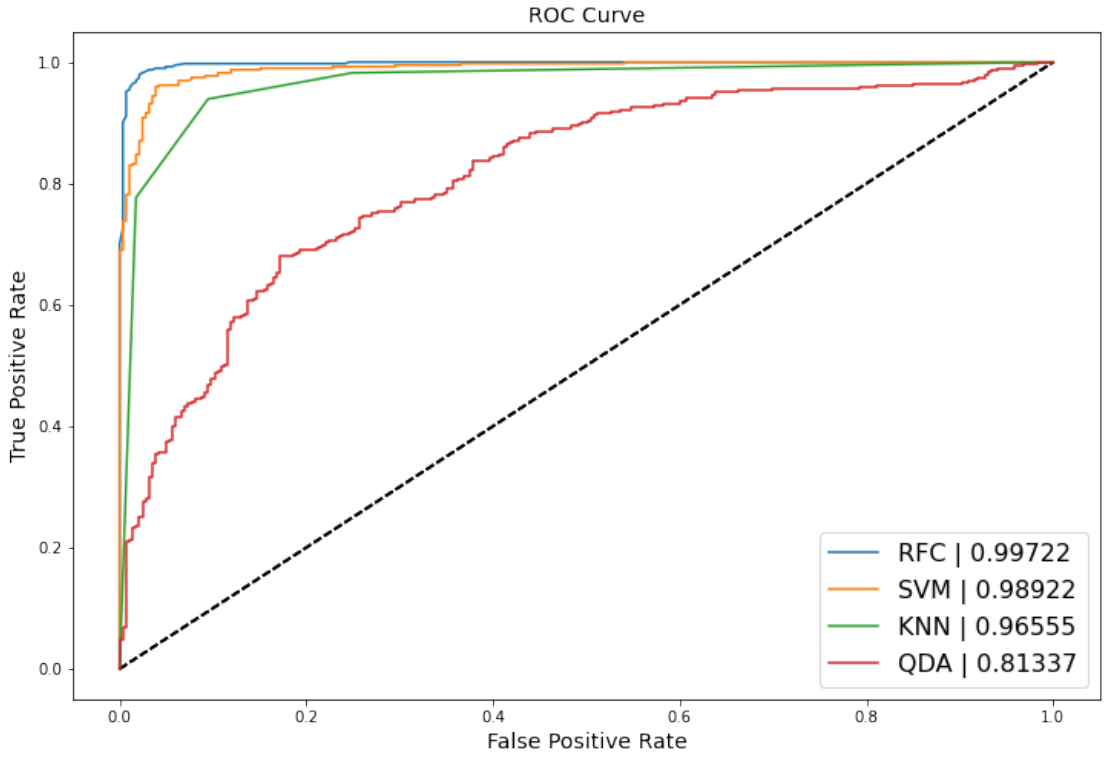


Figure 5: ROC curves for chosen models

4.3 Confusion Matrix Analysis

Confusion matrices provide a detailed breakdown of the model's predictions, showing the true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). Figure 6 presents the confusion matrix for the RFC with chosen parameters.

The confusion matrix reveals that the RFC model correctly predicted 274 samples as edible (true negatives) and 391 samples as poisonous (true positives). However, there were 11 false positive predictions, where edible mushrooms were misclassified as poisonous, and 3 false negative predictions, where poisonous mushrooms were misclassified as edible.

These results indicate that the RFC model performed well in predicting both edible and poisonous mushrooms, with a small number of misclassifications. The high true positive rate (recall) suggests that the model's predictions were reli-

able, minimizing the risk of false negatives and ensuring that potentially harmful mushrooms were correctly identified as poisonous.

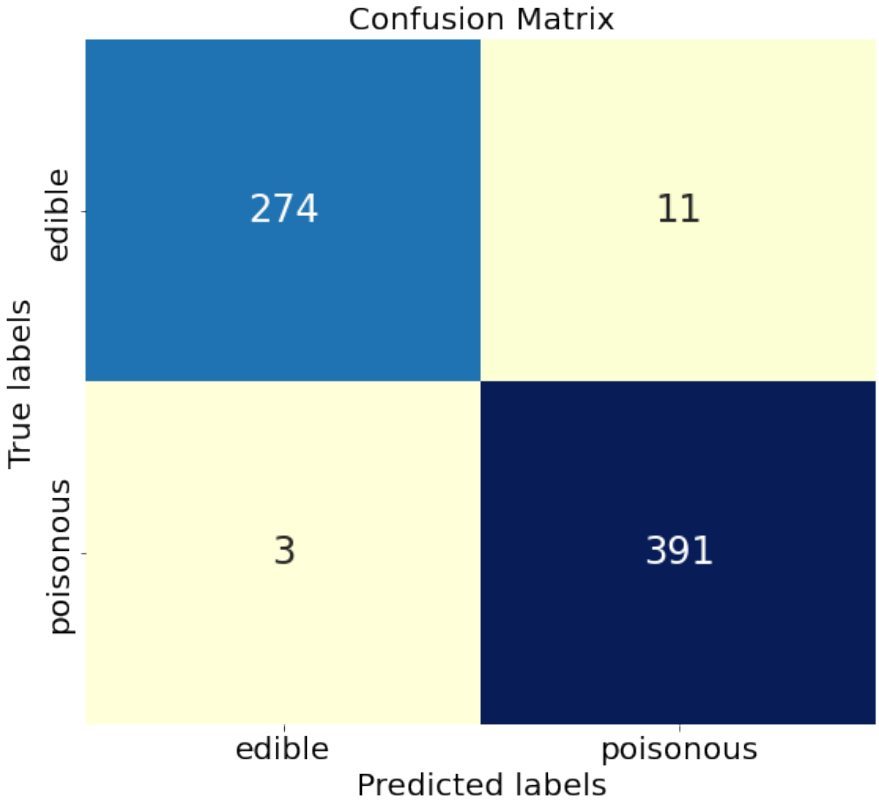


Figure 6: Confusion matrix for Random Forest Classifier

4.4 Feature importance analysis

Permutation importance is a technique used to evaluate the importance of features in a machine learning model. It measures the decrease in model performance when the values of a specific feature are randomly permuted while keeping other features unchanged. The underlying assumption is that if a feature is important for the model, permuting its values should result in a significant decrease in performance.

Since the model has a good predictive power, it increases the reliability of the feature importances obtained. Figure 7 displays features that are most relevant for the prediction results of our model. Surprisingly, cap-color, which was initially and intuitively considered as potentially important happens to be one of the least important, while does-bruise-or-bleed is significantly higher. Moreover, information about the stem (stem-color, stem-height, stem-width) plays an important role too.

This opens prospects for new analysis and a new look at the mushroom's edibility identification.

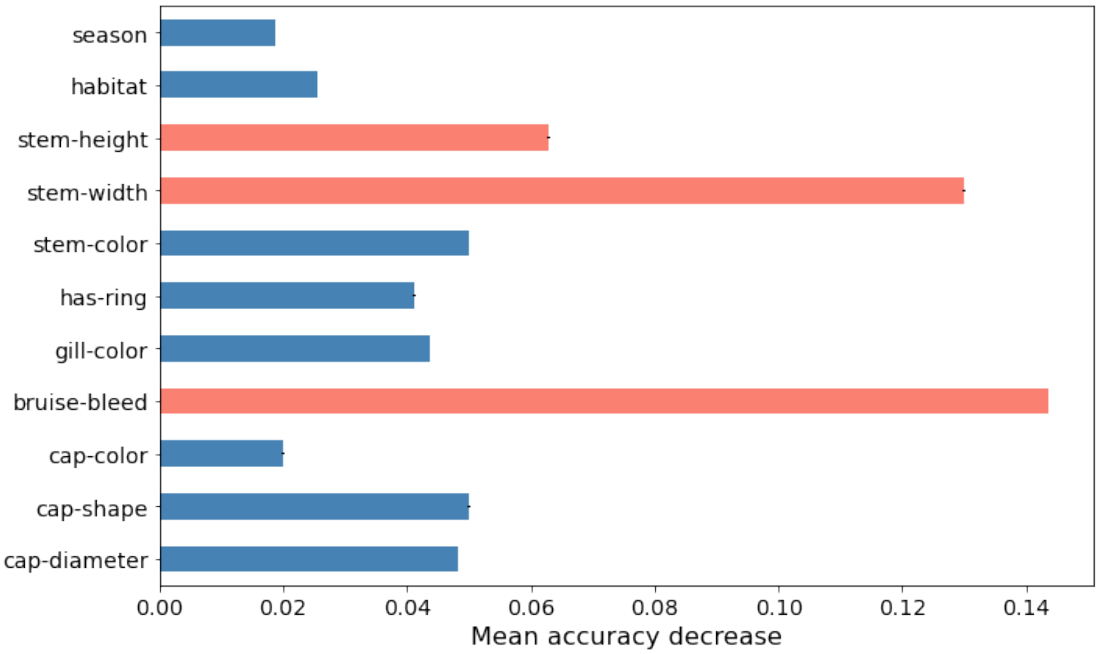


Figure 7: Feature importance using permutation on full RFC model

5 Summary

The objective of our study was to develop ML model capable of accurately differentiating between edible and poisonous mushrooms. The RFC model demonstrated exceptional predictive power, achieving high accuracy in classifying mushrooms and ensuring the safety of mushroom consumers.

Our analysis revealed that attributes related to the mushroom's stem, such as stem color, width, and height, were highly relevant in distinguishing between edible and poisonous mushrooms. Additionally, the feature "does-bruise-or-bleed" was also identified as a significant indicator. Combined with our data analysis insights, these findings contribute to the field of mycology and promote safe mushroom consumption practices by providing valuable insights for accurately classifying mushrooms based on their attributes.

References

[Hardin, 1999] Hardin, P. (1999). *Mushrooms & Toadstools*. Zondervan.

[Pedregosa et al., 2011] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

[Schlimmer, 1987] Schlimmer, J. (1987). Mushroom data set.

[Wagner, 2020] Wagner, D. (2020). Repository of mushroom data sets and python scripts. `https://mushroom.mathematik.uni-marburg.de/files/`.