Mushroom Classification Using Different Techniques

Name:	**Ayush Dabra**
Registration No./Roll No.:	19069
Institute/University Name:	IISER Bhopal
Program/Stream:	Data Science and Engineering
Problem Release date:	January 19, 2022
Date of Submission:	April 24, 2022

1 Introduction

There are an estimated 1.5 million [1] mushroom species in the world today. There are two sorts of mushrooms among the millions that exist worldwide: edible mushrooms and poisonous mushrooms. Many people become ill due to food poisoning because they are unaware that the mushrooms are poisonous. Therefore it is important to distinguish between poisonous and edible mushrooms. Due to similar features between edible and poisonous mushrooms, it becomes very difficult to distinguish between them. Supervised machine learning techniques are utilized in this project to classify mushrooms.

This project aims to classify whether a mushroom is edible or poisonous based on different features present in the dataset. The dataset contains 22 attributes with 7311 instances of mushrooms. Table 1 shows the features present in the dataset. Figure 1 shows the count plot for the labels and Figure 2 shows bar plots for some of the features present in the dataset.

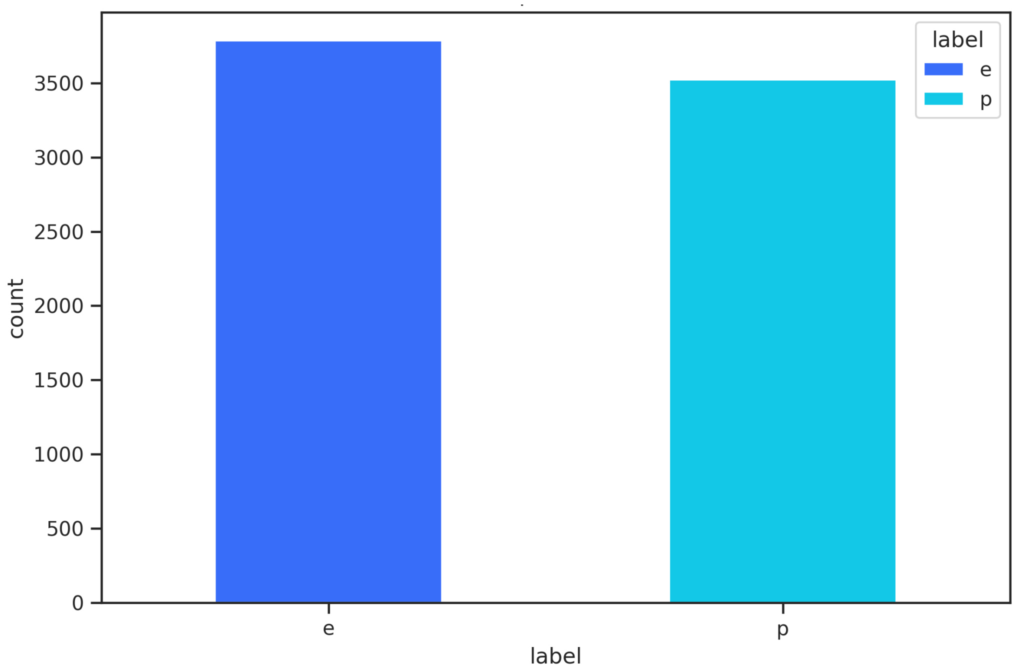

Figure 1: Countplot for label.

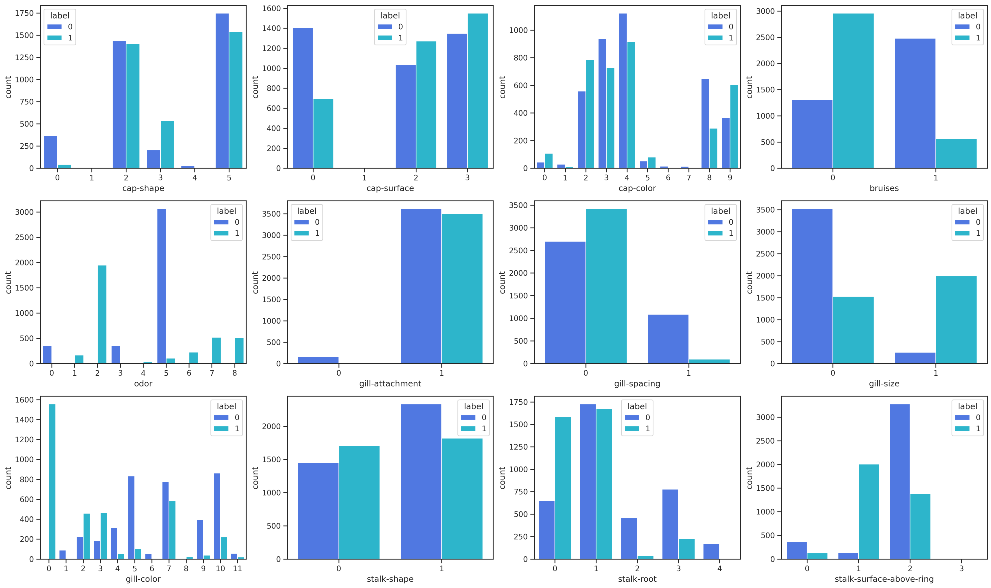

Figure 2: Bar plots for some of the features.

Table 1: Features in the mushroom dataset.

Feature	**Feature Values**
cap-shape	conical (c), convex (x), flat (f), bell (b), sunken (s), knobbed (k)
cap-surface	grooves (g), scaly (y), smooth (s), fibrous (f)
cap-colour	buff (b), cinnamon (c), grey (g), green (r), brown (n), purple (u), red (e), white (w), yellow (y), pink (p)
bruises	no (f), bruises (t)
odour	anise (l), creosote (c), fishy (y), foul (f), almond (a), none (n), pungent (p), spicy (s), musty (m)
gill-attachment	descending (d), free (f), notched (n), attached (a)
gill-spacing	crowded (w), distant (d), close (c)
gill-size	narrow (n), broad (b)
gill-color	brown (n), buff (b), chocolate (h), grey (g), black (k), orange (o), pink (p), purple (u), red (e), green (r), yellow (y), white (w)
stalk-shape	tapering (t), enlarging (e)
stalk-root	club (c), cup (u), equal (e), bulbous (b), rhizomorphs (z), rooted (r), missing (?)
stalk-surface_above_ring	scaly (y), silky (k), smooth (s), fibrous (f)
stalk-surface_below_ring	scaly (y), silky (k), smooth (s), fibrous (f)
stalk-colour-above-ring	buff (b), cinnamon (c), grey (g), orange (o), brown (n), red (e), white (w), yellow (y), pink (p)
stalk-colour-below-ring	buff (b), cinnamon (c), grey (g), orange (o), brown (n), red (e), white (w), yellow (y), pink (p)
veil-type	universal (u), partial (p)
veil-colour	orange (o), white (w), yellow (y), brown (n)
ring-number	one (o), two (t), none (n)
ring-type	evanescent (e), flaring (f), large (l), cobwebby (c), none (n), pendant (p), sheathing (s), zone (z)
spore-print-colour	brown (n), buff (b), chocolate (h), green (r), black (k), purple (u), white (w), yellow (y), orange (o)
population	clustered (c), numerous (n), abundant (a), several (v), solitary (y), scattered (s)
habitat	grasses (g), leaves (l), meadows (m), paths (p), urban (u), waste (w), woods (d)

2 Methods

The features in most machine learning models are expected to be numerical values. However, the features in the dataset are all categorical. So, the ordinal encoding technique is used to convert all the instances to numerical values. The data has been randomly divided into the train and validation sets using the 80-20 rule, which means that 80 percent (5848 instances) of the data points are in the train set and 20 percent (1463 instances) are in the validation set. The main classification algorithms used in this study are Decision Tree, Logistic Regression, Naive Bayes and Support Vector Machine. Firstly, the baseline versions of the models are trained on the dataset, then the hyper-parameters 2 of the models are tuned using grid search. The scikit-learn [2] python package is used to train the models on the dataset. The code for this project can found in this GitHub repository.

Hyperparameters are specified parameters that can be used to tune the behavior of a machine learning algorithm. These are initialized before the training and supplied to the model. A hyperparameter is a parameter whose value governs the learning process. To perform better and improve on the evaluation metric, hyperparameters are tuned by selecting the ideal values. In order to search for the best values in hyper-parameter space, GridSearchCV is used. Grid search is one of the most basic hyper-parameter tuning techniques. Hence its implementation is very straightforward. To tune models, all feasible permutations of the hyperparameters for a specific model are used, and the best-performing ones are chosen.

Table 2: Hyper-parameters of different models

Models	**Hyper-parameters Space**
Decision Tree	• criterion: [‘gini’, ‘entropy’] • max_features: [‘auto’, ‘sqrt’, ‘log2‘] • max_depth: [10, 40, 45, 60]
Logistic Regression	• penalty: [‘l1’, ‘l2’, ‘elasticnet’] • C: [0.5, 0.6, 0.7, 0.8] • solver: [‘newton-cg’, ‘lbfgs’, ‘liblinear’]
Naive Bayes	• var_smoothing: np.logspace(0,-13, num=100)
Support Vector Machine	• C: [80, 85, 90] • kernel: [‘linear’, ‘rbf’, ‘polynomial’]

3 Evaluation Criteria

The most crucial task in developing any machine learning model is assessing its performance. In this project, accuracy, precision, recall, f1-score, and macro-averaged precision metrics are used to evaluate the performance of the models.

- Accuracy is defined as the number of samples from the data set correctly predicted to the desired classes divided by the number of total samples in the data set.

$$Accuracy = \frac{No.\ of\ correctly\ predicted\ data\ points}{Total\ number\ of\ data\ points} = \frac{\text{TP+TN}}{\text{TP+FP+FN+TN}}$$

- Precision is defined as the ratio of correctly classified positive samples (True Positive) to a total number of classified positive samples (either correctly or incorrectly).

$$Precision = \frac{No.\ of\ correctly\ predicted\ positive\ points}{Total\ predicted\ positive\ points} = \frac{\text{TP}}{\text{TP+FP}}$$

- Recall is calculated as the ratio between the numbers of Positive samples correctly classified as Positive to the total number of Positive samples.

$$Recall = \frac{No.\ of\ correctly\ predicted\ positive\ points}{Total\ actual\ positive\ points} = \frac{\text{TP}}{\text{TP+FN}}$$

- F1 score is the harmonic mean between precision and recall.

$$f1\ score = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

4 Analysis of Results

Table 3 describes the performance of all the models based on different metrics. The best hyper-parameters selected by grid search are:

- Decision tree: {'criterion': 'gini', 'max_depth': 45, 'dtc__max_features': 'auto'}
- Logistic regression: {'C': 0.8, 'penalty': 'l1', 'solver': 'liblinear'}
- Naive bayes: {'var_smoothing': 0.0002848035868435799}
- SVM: {'C': 80, 'kernel': 'rbf'}

Table 3: Performance of the models.

Method	Accuracy		Precision		Recall		F1		Macro Avg. Prec.	
	Base	Tuned	Base	Tuned	Base	Tuned	Base	Tuned	Base	Tuned
Decision Tree	0.93	1.0	0.91	1.0	0.94	1.0	0.93	1.0	0.93	1.0
Logistic Reg.	0.93	0.94	0.95	0.95	0.91	0.92	0.93	0.93	0.94	0.94
Naive Bayes	0.90	0.91	0.90	0.93	0.89	0.88	0.90	0.91	0.91	0.92
SVM	0.90	1.0	0.90	1.0	0.89	1.0	0.90	1.0	0.91	1.0

From the experimental results obtained, it can be seen that the Decision Tree and Support Vector Machine gave the highest accuracy of 100%, followed by Logistic Regression with 94% test accuracy and Naive Bayes with 91% test accuracy (tuned versions).

5 Discussions and Conclusion

In this report, I have presented the methodology and the results of the mushroom classification task by experimenting with four popular supervised machine learning algorithms. Out of the four models, the Decision Tree Classifier and Support Vector Machine performed the best. Different parameters of each of the models were also tuned using grid search and better performance is obtained after tuning the hyper-parameters. For further research, a deep learning based artificial neural network can also be trained on the dataset and a comparison between machine learning and deep learning can be drawn.

References

[1] David L. Hawksworth. The magnitude of fungal diversity: the 1.5 million species estimate revisited* *paper presented at the asian mycological congress 2000 (amc 2000), incorporating the 2nd asia-pacific mycological congress on biodiversity and biotechnology, and held at the university of hong kong on 9-13 july 2000. *Mycological Research*, 105(12):1422–1432, 2001.

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