University for the Creative Arts

BERLIN SCHOOL OF BUSINESS & INNOVATION

Essay / Assignment Title: Unveiling Mushroom Mysteries: A Python-Powered Machine Learning Journey

Programme title: Predictive Analytics and Machine Learning using Python

Name: VICTORY CHIEMEKA OKEZIE

Year: 2025

CONTENTS

Statement of compliance with academic ethics and the avoidance of plagiarism

I honestly declare that this dissertation is entirely my own work and none of its part has been copied from printed or electronic sources, translated from foreign sources and reproduced from essays of other researchers or students. Wherever I have been based on ideas or other people texts I clearly declare it through the good use of references following academic ethics.

(In the case that is proved that part of the essay does not constitute an original work, but a copy of an already published essay or from another source, the student will be expelled permanently from the postgraduate program).

Name and Surname (Capital letters): VICTORY CHIEMEKA OKEZIE

………..……….

Date: ....2............./....1....../......2025...

INTRODUCTION

This study is used to calculate a set of exercise questions about a hypothetical database of each species of mushroom described by four attributes. The study calculated the entropy, misclassification rate, information gain, weighted average, and it showed which attribute that will be selected for building decision tree.

Additionally, the study explores the application of seven machine learning algorithms to predict whether a mushroom is edible or poisonous based on its physical attributes. These algorithms were evaluated on a comprehensive dataset, and their performance was assessed using key metrics such as accuracy, precision, recall, and F1-score.

This report will not only demonstrate the effectiveness of machine learning in solving classification tasks but also provides a comparative analysis of models to identify the best approach for mushroom classification. By combining theoretical concepts with practical applications, this study highlights the power and versatility of machine learning in addressing real-world challenges.

CHAPTER ONE: DECISION TREE AND INFORMATION GAIN CALCULATIONS

Q1. The formula for entropy is H(S) = -p1log2(p1) – p2log2(p2). Where p1 is the proportion of edible mushrooms and p2 is the proportion of poisonous mushrooms. I used reference from Bishop, C.M., 2006. *Pattern Recognition and Machine Learning*. 1st ed. New York: Springer. To calculate entropy, I first need to know what is p1 and p2, using total samples N = 10, p1 = edible mushroom/N which is p1=5/10=0.5, p1 is 0.5 and p2=poisonous mushroom/N = 5/10= 0.5. Therefore, p2 is 0.5. Entropy = -(0.5log2(0.5) + 0.5log2(0.5)), = -(0.5*-1+0.5*-1) = 1.0. Therefore, entropy of the data is H = 1.0

Q2. Now I solve for information gain of cap-color relative to these training examples. The formular for information gain is IG = H(entropy) - ∑|Dv|/|D|H(Dv). Where H is the entropy of the whole dataset which is 1, Dv is the subset of the dataset corresponding to each value of the attribute A and H(D) is the entropy of each subset. I will first group data by cap color from table 2 which will be w(samples) = {1,4,5} is edible = 3, poisonous = 0. y(samples) = {2,3,6,9,10} is edible = 2, poisonous = 3. g(samples) = {7} is Edible = 0, poisonous = 1 and x(samples) = {8} is edible = 0, poisonous = 1. Now I calculate H(Dv) for each subset which is w: p1=3/3=1, p2=0/3=0. H(D) of w = -(1log2(1) + 0log2(0)) = 0, y: p1=2/5=0.4, p2=3/5 = 0.6. H(D) of y = -(0.4log2(0.4) + 0.6log2(0.6)) = 0.971. g: p1=0, p2=1. H(D) of g = -(0log2(0) + 1log2(1)) = 0. r: p1 = 0, p2= 1. H(D) of r = 0

Now we calculate the weighted entropy. Formula for weighted entropy is ∑|Dv|/|D|H(Dv). Weighted entropy = 3/10 * 0 + 5/10 * 0.971 + 1/10 * 0 + 1/10 * 0. Weighted entropy is 0.4855. Finally, Information gain is IG = H-Weighted entropy, = 1-0.4855 = 0.5145. Information Gain is 0.5145

Q3. The next attribute that should be selected is the one with the highest information gain. Based on intuition habitat should be the next attribute because it might have a higher IG due to better split differentiation. I will have to calculate to determine if habitat is unique and no other attribute have the same IG. Calculation is below.

Entropy of the whole dataset (H) is 1.0 and Total Samples: N is 10. I will calculate information gain for cap shape which is c(samples) = {1,4,5,6,10} is edible = 3 and poisonous = 2, f(samples) = {2,3,7} is edible = 2 and poisonous = 1, b(samples) = {8,9} is edible = 0 and poisonous = 2. H(D) of c is p1= 3/5, p2 = 2/5. H(D) of c = -(3/5log2(3/5) + 2/5log2(2/5)) = 0.971. H(D) of f is p1= 2/3, p2 = 1/3. H(D) of f = -(2/3log2(2/3) + 1/3log2(1/3)) = 0.918. H(D) of b is p1= 0, p2 = 1. H(D) of f = -(0log2(0) + 1log2(1)) = 0. The weighted entropy is 5/10 * 0.971 + 3/10 * 0.918 + 2/10 * 0, = 0.4855 + 0.2751 + 0, = 0.7609. weighted average of cap shape is 0.7609. Information gain is H – weighted average, = 1.0 – 0.7609, = 0.2391. Information gain of cap shape is 0.2391

I will calculate information gain for odor which is p(samples) = {1,3,7,9} is edible = 2 and poisonous = 2, n(samples) = {2,6,8} is edible = 1 and poisonous = 2, a(samples) = {4,5} is edible = 2 and poisonous = 0, s(samples) = {10} is edible = 0 and poisonous = 1. H(D) of p is p1= 2/4, p2 = 2/4. H(D) of p = -(2/4log2(2/4) + 2/4log2(2/4)) = 1.0. H(D) of n is p1= 1/3, p2 = 2/3. H(D) of n = -(1/3log2(1/3) + 2/3log2(2/3)) = 0.918. H(D) of a is p1= 1, p2 = 0. H(D) of a = -(1log2(1) + 0log2(0)) = 0. H(D) of s is p1= 0, p2 = 1. H(D) of s = -(0log2(0) + 1log2(1)) = 0. The weighted entropy is 4/10 * 1 + 3/10 * 0.918 + 2/10 * 0 + 1/10 * 0, = 0.4 +0.2754 + 0 + 0, = 0.6754. weighted average of odor is 0.6754. Information gain is H – weighted average, = 1.0 – 0.6754, = 0.3246. Information gain of odor is 0.3246

I will calculate information gain for habitat which is m(samples) = {1,7,9,10} is edible = 1 and poisonous = 3, u(samples) = {2,3,4} is edible = 3 and poisonous = 0, l(samples) = {5,6,8} is edible = 1 and poisonous = 2. H(D) of m is p1= 1/4, p2 = 3/4. H(D) of m = -(1/4log2(1/4) + 3/4log2(3/4)) = 0.811. H(D) of u is p1= 1, p2 = 0. H(D) of u = -(1log2(1) + 0log2(0)) = 0. H(D) of l is p1= 1/3, p2 = 2/3. H(D) of l = -(1/3log2(1/3) + 2/3log2(2/3)) = 0.918. The weighted entropy is 4/10 * 0.811 + 3/10 * 0 + 3/10 * 0.918, = 0.3244 + 0 + 0.2754, = 0.5998. weighted average of habitat is 0.5998. Information gain is H – weighted average, = 1.0 – 0.5998, = 0.4002. Information gain of habitat is 0.4002. The attribute with the highest information gain is Habitat. Therefore, the next attribute selected for the building the decision tree is Habitat. Habitat is unique because there is no other attribute with similar information gain and habitat has the highest.

Q4. The formula for expected misclassification rate is Error = min (P1, P2). Where P1 is 0.5 and P2 is 0.5. Error = min (0.5,0.5) = 0.5. Therefore, Expected Misclassification rate is 0.5.

CHAPTER TWO: EXPLORATORY AND CLASSIFICATION ALGORITHMS ON MUSHROOM DATASET

I will explore, classify and perform machine learning algorithms that predict whether a mushroom is edible or poisonous in this chapter. The dataset I am using was already provided by Usman Akhtar (September 2024). *Mushrooms-dataset*. Available at: https://github.com/usmanakhtar/CryptocurrencyDataset/. (Accessed: 4 January 2025). All my calculations and analysis have been coded in my jupyter notebook file. The dataset has 8124 rows and 23 columns. The rows values are categorical, which means in order to predict, I will have to convert categorical values to float or numerical values using label encoder. The dataset is clean and they are no missing values.

I used the code .describe() to find the exploratory data analysis. Due to the fact that the dataset is a categorical dataset, the code showed the count, unique, top and the amount of frequency of the rows values that appears in the dataset. Then I performed a value count of all the main columns that was used to select which is best for decision tree back in the first chapter one. Then I visualized each columns row distribution in the dataset. Below is a sample of one of the visualizations (The class distribution). See Figure 1.

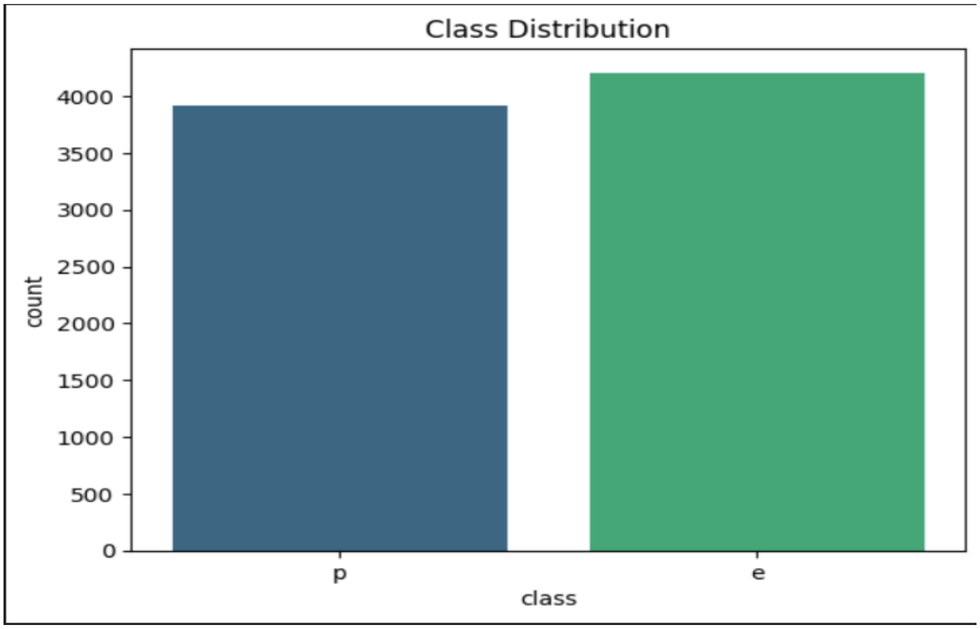


Figure 1: Class Distribution

I made a pair plot of the relationship between each column variables. In my jupyter notebook file, the pair plot shows the interconnectivity of each five columns with the rest of the columns. I only used five columns each for interpretability because if I had used all the columns, the pair plot would have been so big and hard to read. The below Figure 2 shows the sample of one of these pair plot. The graph shows how each column values e.g the labels e, y appears in other columns and it shows how many times these values appear. My jupyter notebook file includes the summary statistics, distribution analysis and the visualizations of relationship between variables.

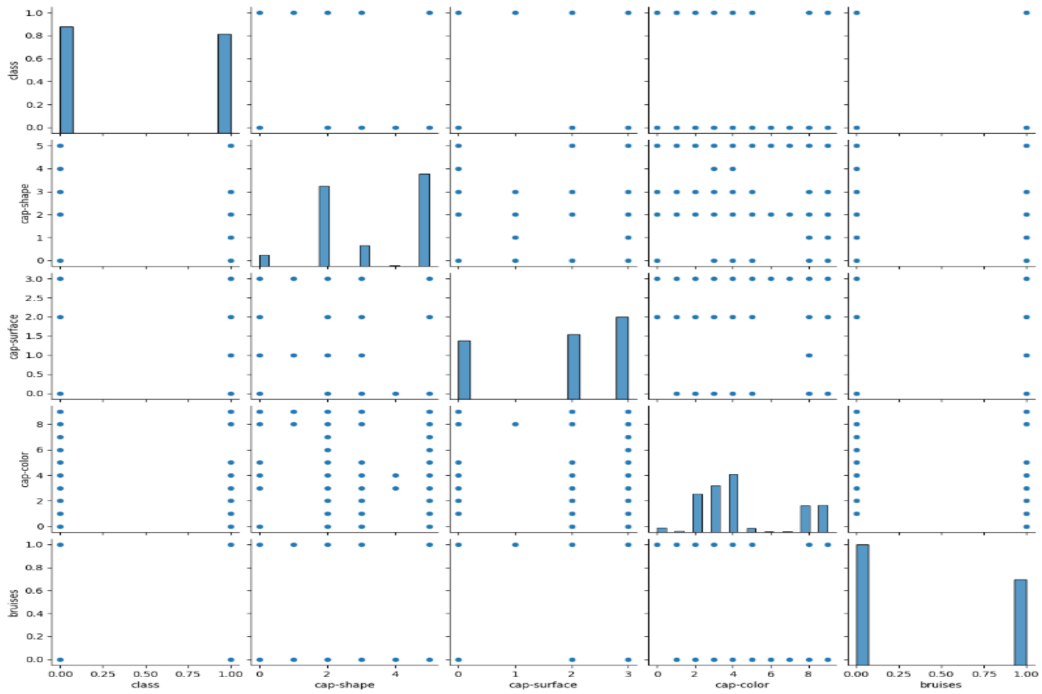
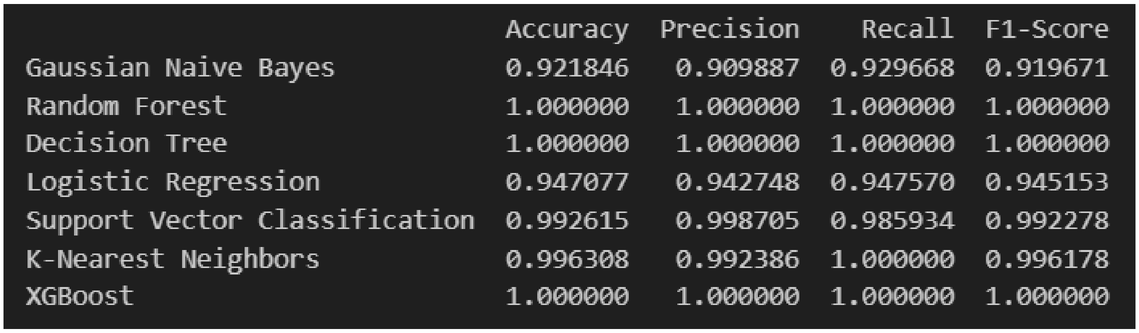


Figure 2: Pair plot Showing Relationship of The First Five Columns

I implemented seven different machine learning algorithms to predict whether a mushroom is edible or poisonous. For each algorithm, I reported the accuracy, precision, recall and F1 score. I will explain which is the best algorithm and how it predicted the analysis accurately. My target variable is class column and the rest of the column was used has features. After training and testing it, each algorithm performed well (See figure 3).



```
                               Accuracy  Precision    Recall  F1-Score
Gaussian Naive Bayes           0.921846   0.909887  0.929668  0.919671
Random Forest                  1.000000   1.000000  1.000000  1.000000
Decision Tree                  1.000000   1.000000  1.000000  1.000000
Logistic Regression            0.947077   0.942748  0.947570  0.945153
Support Vector Classification  0.992615   0.998705  0.985934  0.992278
K-Nearest Neighbors            0.996308   0.992386  1.000000  0.996178
XGBoost                        1.000000   1.000000  1.000000  1.000000
```

Figure 3: The Performance of The Seven Model

The best performing model was Random Forest, Decision Tree and XGBoost (See Figure, 4,5 and 6) because with each having a score of one across the accuracy, precision, recall and F1 score. The three measured the accurate proportion of correctly classifying samples out of all samples. And with a perfect accuracy (1.00) means that the models predicts both edible and poisonous mushrooms correctly 100% of the time.

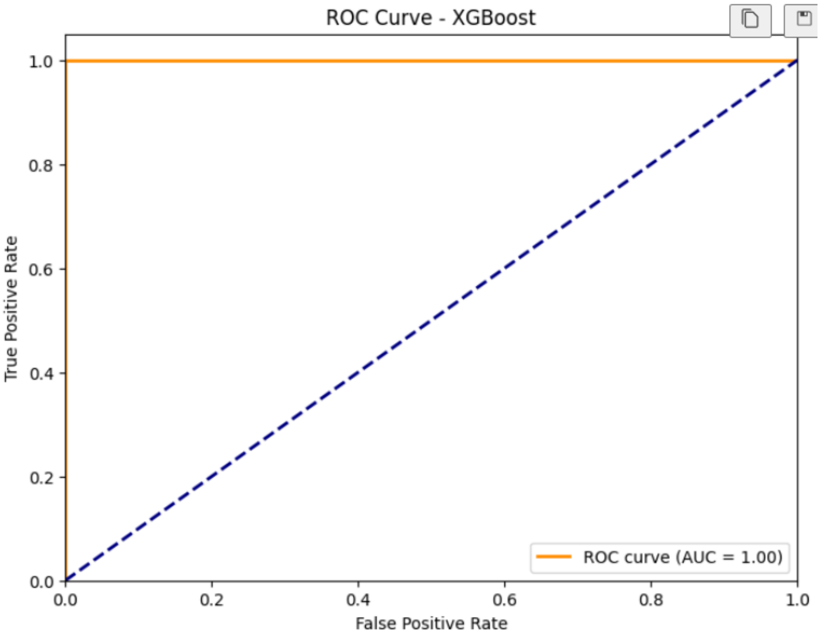


Figure 4: XGBoost ROC Curve

The above graph shows the ROC curve of XGBoost. The closer the curve is to the top left corner, the better the model is at distinguishing between edible and poisonous mushroom. An AUC closer to one means the model is excellent at classification.

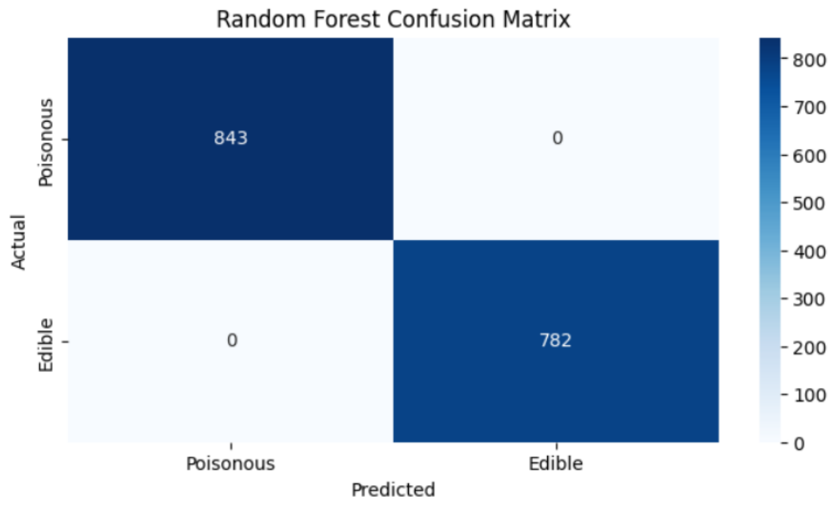


Figure 5: Random Forest Confusion Matrix

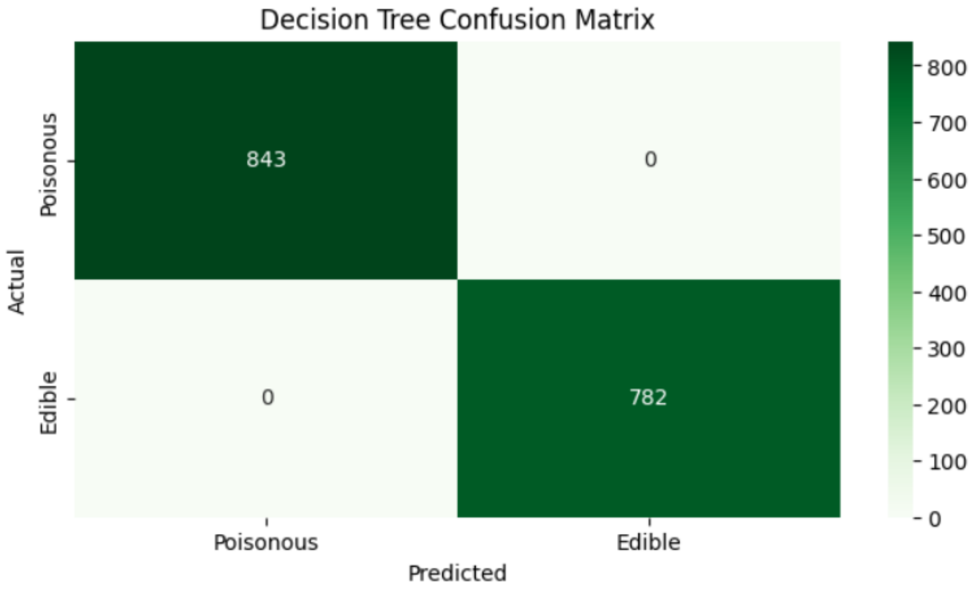


Figure 6: Decision Tree Confusion Matrix

Confusion matrix is a graphical visualization that shows how well the models classify the edible and poisonous mushroom. The above graphs show the diagonal values that represent correct classification, while the off-diagonal values represent misclassification. In the case of decision tree and random forest, they both have a misclassification of zero in both edible and poisonous, which means they both accurate in predicting whether a mushroom is edible or poisonous.

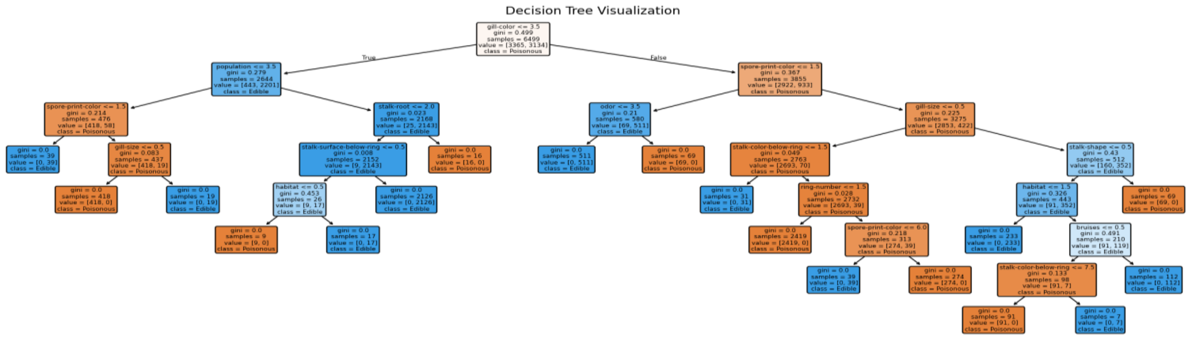

Decision Tree Visualization
gill-color <= 3.5 gini = 0.499 samples = 6499 value = [3365, 3134] class = Poisonous
True
False
population <= 3.5 gini = 0.279 samples = 2644 value = [443, 2201] class = Edible
spore-print-color <= 1.5 gini = 0.367 samples = 3855 value = [2922, 933] class = Poisonous
spore-print-color <= 1.5 gini = 0.214 samples = 476 value = [418, 58] class = Poisonous
stalk-root <= 2.0 gini = 0.023 samples = 2168 value = [25, 2143] class = Edible
odor <= 3.5 gini = 0.21 samples = 580 value = [69, 511] class = Edible
gill-size <= 0.5 gini = 0.225 samples = 3275 value = [2853, 422] class = Poisonous
gini = 0.0 samples = 39 value = [0, 39] class = Edible
gill-size <= 0.5 gini = 0.083 samples = 437 value = [418, 19] class = Poisonous
stalk-surface-below-ring <= 0.5 gini = 0.008 samples = 2152 value = [9, 2143] class = Edible
gini = 0.0 samples = 16 value = [16, 0] class = Poisonous
gini = 0.0 samples = 511 value = [0, 511] class = Edible
gini = 0.0 samples = 69 value = [69, 0] class = Poisonous
stalk-color-below-ring <= 1.5 gini = 0.049 samples = 2763 value = [2693, 70] class = Poisonous
stalk-shape <= 0.5 gini = 0.43 samples = 512 value = [160, 352] class = Edible
gini = 0.0 samples = 418 value = [418, 0] class = Poisonous
gini = 0.0 samples = 19 value = [0, 19] class = Edible
habitat <= 0.5 gini = 0.453 samples = 26 value = [9, 17] class = Edible
gini = 0.0 samples = 2126 value = [0, 2126] class = Edible
gini = 0.0 samples = 31 value = [0, 31] class = Edible
ring-number <= 1.5 gini = 0.028 samples = 2732 value = [2693, 39] class = Poisonous
habitat <= 1.5 gini = 0.326 samples = 443 value = [91, 352] class = Edible
gini = 0.0 samples = 69 value = [69, 0] class = Poisonous
gini = 0.0 samples = 9 value = [9, 0] class = Poisonous
gini = 0.0 samples = 17 value = [0, 17] class = Edible
gini = 0.0 samples = 2419 value = [2419, 0] class = Poisonous
spore-print-color <= 6.0 gini = 0.218 samples = 313 value = [274, 39] class = Poisonous
gini = 0.0 samples = 233 value = [0, 233] class = Edible
bruises <= 0.5 gini = 0.491 samples = 210 value = [91, 119] class = Edible
gini = 0.0 samples = 39 value = [0, 39] class = Edible
gini = 0.0 samples = 274 value = [274, 0] class = Poisonous
stalk-color-below-ring <= 7.5 gini = 0.133 samples = 98 value = [91, 7] class = Poisonous
gini = 0.0 samples = 112 value = [0, 112] class = Edible
gini = 0.0 samples = 91 value = [91, 0] class = Poisonous
gini = 0.0 samples = 7 value = [0, 7] class = Edible

Figure 7: Decision Tree Visualization

Figure 7 shows the structure of the decision tree model. Each node represents a decision based on feature, and the tree splits data based on feature thresholds. However, if I was to pick one machine learning model based on personal preference then I will pick XGBoost because XGBoost will be preferred for its robustness and efficiency on larger or noisier datasets. XGBoost also uses gradient boosting that reduces overfitting and improve generalization. Below table 1 is a detailed comparison of the seven models.

Model	Accuracy	Precision	Recall	F1-Score	Performance
Gaussian Naïve Bayes	0.921846	0.909887	0.929668	0.919671	Performs well but misclassifies some edible/poisonous mushroom
Random Forest	1.000000	1.000000	1.000000	1.000000	Perfect performance. Predicts all edible and poisonous mushrooms correctly. It handles categorical

					features and interactions very effectively.
Decision Tree	1.000000	1.000000	1.000000	1.000000	Perfect performance. Predicts all edible and poisonous mushrooms correctly. Handles categorical and interactions very effectively.
Logistic Regression	0.947077	0.942748	0.947570	0.945153	Performs decently but struggles with complex, non-linear patterns, some misclassifications occur for edible and poisonous mushrooms.
Support Vector Classification	0.992615	0.998705	0.985934	0.992278	Very strong performance. Slightly misses perfect accuracy due to a few

					errors. Handles high-dimensional data well but requires more computational resources.
K-Nearest Neighbors	0.996308	0.992386	1.000000	0.996178	Nearly perfect. Handles the dataset well but is slightly sensitive to the choice of the number of neighbors (k) and distance metrics. A few misclassifications might occur.
XGBoost	1.000000	1.00000	1.000000	1.000000	Perfect performance. It is optimized for speed, accuracy and generalization which makes it a perfect choice for this dataset.

Table 1: Comparison of The Seven Models

CONCLUSION

In the first chapter of this analysis, I successfully performed calculations on the decision tree and figured out what attribute should be used in building the tree. By delving into concepts like entropy and information gain, I was able to understand decision tree algorithms and I was able to highlight how these metrics help identify the most informative features that will be useful in splitting the dataset and which attribute I can select for building a decision tree.

I explored the classification of mushrooms as either edible or poisonous using seven machine learning algorithms. Through rigorous evaluation and visualization, we assessed each model's performance in terms of accuracy, precision, recall, and F1-score. Our results revealed that Random Forest, Decision Tree, and XGBoost achieved perfect scores across all metrics. These models demonstrated their strength in handling categorical data, leveraging feature importance and capturing complex patterns in the dataset. Among them, XGBoost stands out as the most robust and efficient due to its gradient boosting approach which reduces overfitting and enhances predictive accuracy.

In conclusion, this study shows how we can use entropy and information gain to find the best attribute that should be used for decision tree and the power of machine learning in addressing critical classification tasks, such as identifying edible or poisonous mushrooms. For optimal results, I recommend XGBoost, Random Forest or Decision Tree because they balance interpretability, efficiency and predictive accuracy. This analysis not only provides a practical solution for mushroom classification but also highlights the utility of machine learning in solving complex classification problems.

BIBLIOGRAPHY

References

Usman Akhtar (September 2024). *Mushrooms-dataset*. Available at: https://github.com/usmanakhtar/CryptocurrencyDataset/ (Accessed: 4 January 2025).

Bishop, C.M., 2006. *Pattern Recognition and Machine Learning*. 1st ed. New York: Springer.

List of Figures

List of Tables