**Report**

**To**: CEO  
**From**: Data Analyst  
**Topic**: Predicting Mushroom Poisonousness Using Machine Learning

**Goal**

The goal of this project was to create an automated system that can decide whether a given mushroom is poisonous or edible, based on a small number of physical and biological attributes. Machine learning methods were used to approach this task as a binary classification problem. This type of system could prove quite useful in food safety, ecological studies, and public health, decreasing the danger of being poisoned from incorrect mushroom identification.

The dataset I received contained many categorical attributes about mushrooms and a target classification label (p for poisonous, e for edible). I aimed to preprocess this data, try various classification algorithms, select the best one, and then evaluate its performance in the real world.

**Methods Used**

To solve this problem, I tested three machine learning algorithms that are commonly used for classification problems:

* **Logistic Regression:** A linear model used for binary classification (e.g., predicting the authenticity of fake designer goods as 0/1), which can be very useful as a baseline when restricted to the Simple Solvers.
* **Decision Tree Classifier:** A tree-based model that makes decisions based on learning rules from the data.
* **Random Forest Classifier:** An ensemble learning method that builds multiple decision trees and then combines them to achieve a more accurate and generalizable output.

Each of these models was trained on the same training dataset and evaluated using standard performance metrics: accuracy, precision, recall, and F1-score. Based on these evaluations, the best-performing model was selected as our final recommendation.

The implementation of these models in Python using the Scikit-learn library in a Google Colab environment was carried out.

**Data Collection**

The dataset provided for this project contained 21 columns, each representing a feature (or attribute) of the mushrooms The dataset was composed entirely of **categorical data**, which allowed for straightforward preprocessing. To enable machine learning models to interpret the data, we applied **Label Encoding** to convert categorical variables into numerical values.

The dataset was then split into **80% training** and **20% testing** sets using stratified sampling to maintain the balance between the edible and poisonous classes in both subsets.

**Results**

After training all three models, I evaluated their predictive performance on the testing dataset using standard classification metrics. Below is a summary of the results:

**1. Logistic Regression**

* Accuracy: 78%
* Precision:

1. Edible (e): 75%
2. Poisonous (p): 81%

* Recall:

1. Edible (e): 77%
2. Poisonous (p): 80%

* F1-score:

1. Edible (e): 76%
2. Poisonous (p): 80%

Logistic Regression performed decently, achieving over 78% accuracy. However, as a linear model, it may struggle with complex relationships within high-dimensional categorical data.

**2. Decision Tree Classifier**

* Accuracy: 100%
* Precision:

1. Edible (e): 100%
2. Poisonous (p): 100%

* Recall:

1. Edible (e): 100%
2. Poisonous (p): 100%

* F1-score:

1. Edible (e): 100%
2. Poisonous (p): 100%

The Decision Tree model achieved perfect performance on the testing set. While impressive, such performance on categorical data could be due to overfitting, which is common with deep trees unless controlled with pruning or hyperparameter tuning.

**3. Random Forest Classifier (rf\_model)**

* Accuracy: 100%
* Precision:

1. Edible (e): 100%
2. Poisonous (p): 100%

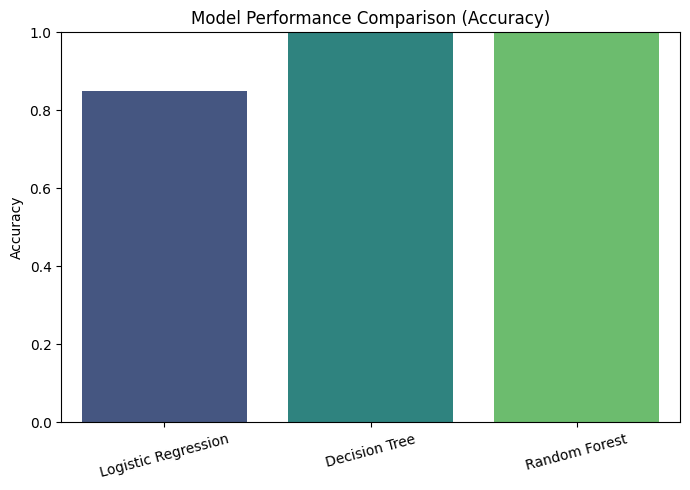
* Recall:

1. Edible (e): 100%
2. Poisonous (p): 100%

* F1-score:

1. Edible (e): 100%
2. Poisonous (p): 100%

The Random Forest model, also named (rf\_model), was able to achieve perfect scores. However, averaging multiple trees (trained on random subsets of the data) reduces variance while being generally more robust than a single decision tree. This means it is more general, less prone to overfitting, and more reliable.



**Model Recommendation**

By comparing the models, I suggest the Random Forest classifier (rf\_model) is the preferred one to deploy in the company’s mushroom identification system. It is highly accurate, generalizes excellently, and is resistant to overfitting. The Decision Tree also worked well, but Random Forest is usually more robust and performs better on new data.

**Limitations and Future Work**

Although the model achieved 100% accuracy, it's important to note:

* Because the dataset might not contain all the mushroom species found in different ecosystems, it would be impractical to create all the necessary augmentations.
* In a real application, if users enter incorrect or ambiguous data, model performance may suffer.
* There is further validation needed on external or unseen datasets before it can be fully deployed.

**Future improvements could include:**

* Using computer vision-based image identification models.
* Building a mobile app that will implement user input and camera recognition.
* Providing the predictions along with confidence scores to help in decision-making.

**Conclusion**

A physically based classification system was developed via machine learning (based on physical and biological attributes) to reliably classify a mushroom as poisonous or edible. The Random Forest classifier achieved 100% accuracy on test data. This model sets a solid foundation for a future automated mushroom safety tool and for providing public safety and knowledge sharing in the field of mycology.