Machine Learning Models for Mushroom Classification

A Project Presentation by Asma Dammak

Table of contents

01 Introduction

Why Mushroom classification?

- Prevents accidental poisoning and enhances food safety.
- Manual classification is time-consuming and error-prone.
- Machine Learning (ML) can automate and improve accuracy.

Goals?

- Compare traditional ML models (Logistic Regression, KNN, Random Forest) vs. deep learning models (MLP, TabNet, ANN).
- Use Explainable AI (SHAP) to interpret model decisions.
- Identify key features influencing classification

02 Exploratory Data Analysis

Understanding the Dataset:

- Target Class:

0 = Edible, 1 = Poisonous

- 8 Features:

Cap diameter, cap shape, gill attachment & color, stem height & width, stem color, season

Data

Class Balance

Class distribution is **balanced**, meaning performance metrics such as accuracy will not be misleading.

Class Outliers

Box plot visualizations show the existence of **outliers**. All instances with a z-score >3 have been removed.

Data Quality

No missing values were found, indicating that the dataset is clean.
Categorical features properly encoded.

Correlation

The only notable correlation is between stem width and cap diameter, but seeing that we only have a few categories to start with, we will be keeping both.

Scaling

Ensures that all features contribute equally to model training, preventing biases caused by large numerical differences between variables

03 Models Building

Traditional ML Models

1 Logistic Regression

- Regularization (L2 penalty - Ridge Regression) to prevent overfitting.
- Hyperparameter tuning (Grid Search) to find optimal C-value.

2 K-Nearest Neighbour

- Hyperparameter tuning (Randomized Search) → Best k-value = 7.
- Distance metric tuning (Manhattan vs. Euclidean vs Minkowski)

3 Random Forest

- Hyperparameter tuning (Randomized Search CV).
- Max depth & number of estimators adjusted to reduce overfitting.

Deep Learning Models

1 Feedforward Neural Network (MLP)

- Hyperparameter tuning (Random Search CV).
- ReLU activation + Dropout layers to reduce overfitting.
- Early Stopping (monitor validation loss to prevent overfitting).

2 TabNet

- Tried hyperparameter tuning (Grid Search) but reverted to default model due to a poorer performance.

3 ANN

- Used Optuna (Bayesian Optimization) for hyperparameter tuning.
- Batch Normalization & Learning Rate Scheduling to improve training stability.
- Dropout Regularization to prevent overfitting.

04 Results

Table 1: Performance Summary of All Models

Model	Accuracy	Precision	Recall	F1-Score	ROC-AUC
Logistic Regression	0.64	0.65	0.64	0.64	0.70
KNN	0.72	0.73	0.72	0.72	0.72
Random Forest	0.99	0.99	0.99	0.99	1.00
MLP	0.74	0.82	0.68	0.74	0.84
TabNet	0.98	0.98	0.98	0.98	1.00
ANN	0.97	0.96	0.97	0.97	1.00

- Best Performing: **RF, TabNet, ANN**
- Best Traditional Model: Random Forest (high accuracy + explainability)
- Best Deep Learning Model: TabNet & ANN
- Weakest Models: Logistic Regression, KNN

05 SHAP Analysis

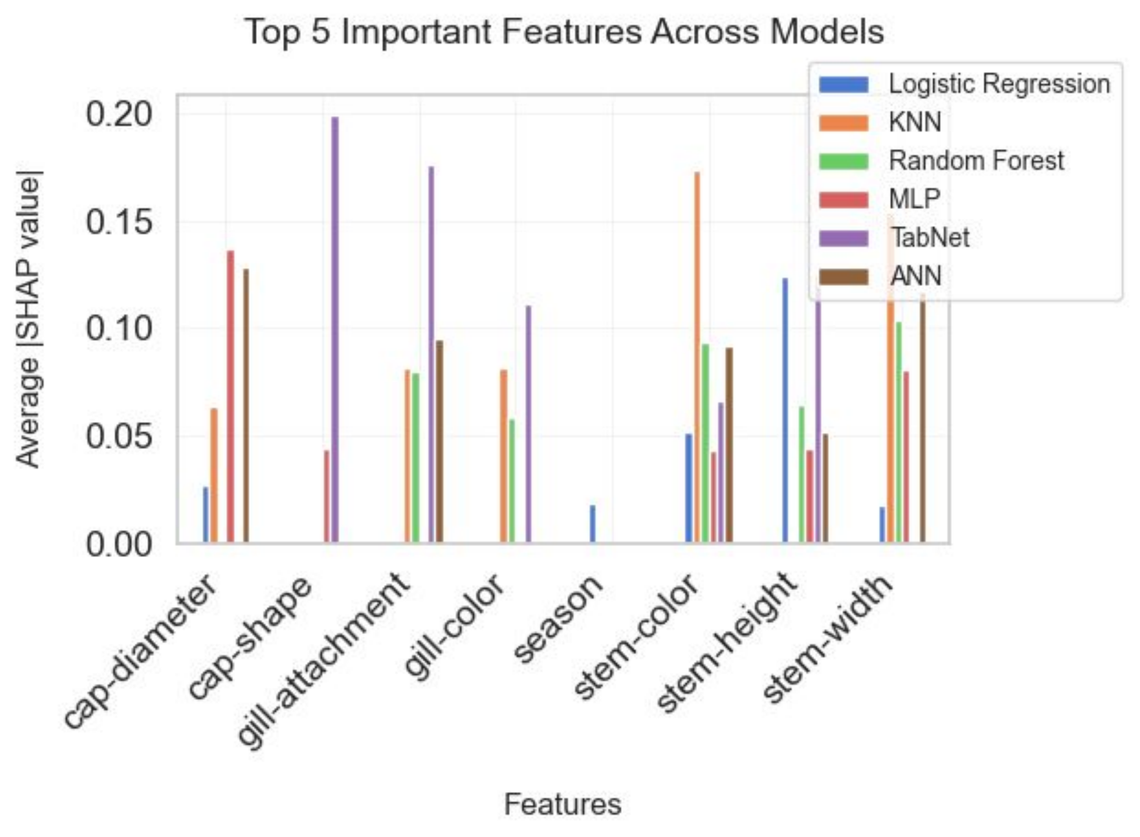
Why SHAP?

- SHAP (SHapley Additive ExPlanations) explains how models make decisions.
- Helps identify critical features affecting mushroom classification.
- Provides insights into feature interactions & individual contributions.

SHAP Feature Importance Across Models

Model-Specific Differences:

- 🔹 Logistic Regression → Focuses on stem-height, stem-color (simple linear relationships).
- 🔹 KNN → Relies on cap-shape, cap-diameter (distance-based impact).
- 🔹 Random Forest → Balanced importance across cap-shape, gill-color, gill-attachment.
- 🔹 TabNet → Uses cap-shape, gill-attachment with attention-based selection.
- 🔹 MLP & ANN → Capture complex nonlinear interactions (cap-diameter, stem-width).



Feature Importance:

- **Stem-related features** (*height*, *width*, *color*) are dominant across models.
- **Cap-diameter** is crucial for Logistic Regression, KNN, MLP, and ANN.
- **Gill-related features** impact Random Forest, TabNet, and ANN the most.
- TabNet & ANN focus on **complex interactions**, while Logistic Regression focuses on a few **dominant features**.

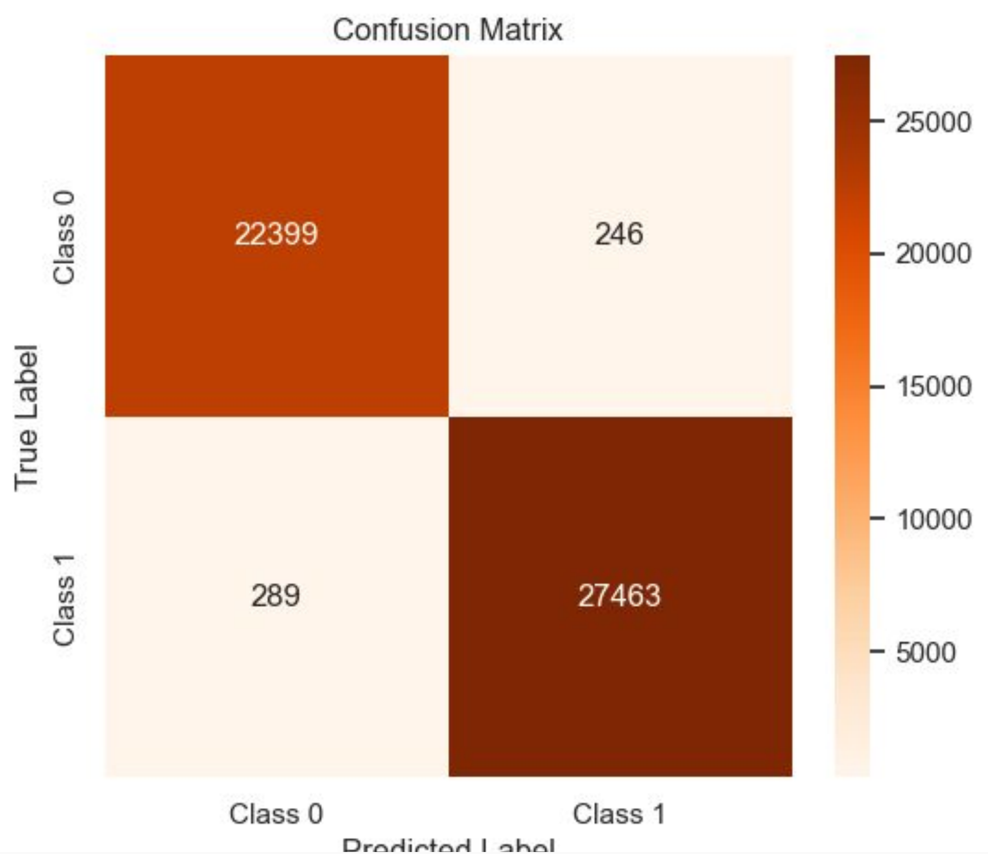
Why Random Forest is the best model?

1- Highest **metrics**

2- **Lowest False Negative ratio “1%”**

means that 1% of all actual poisonous mushrooms were incorrectly classified as edible.

⇒ In high-risk applications (like food safety), recall should be maximized to avoid any misclassification of poisonous mushrooms.

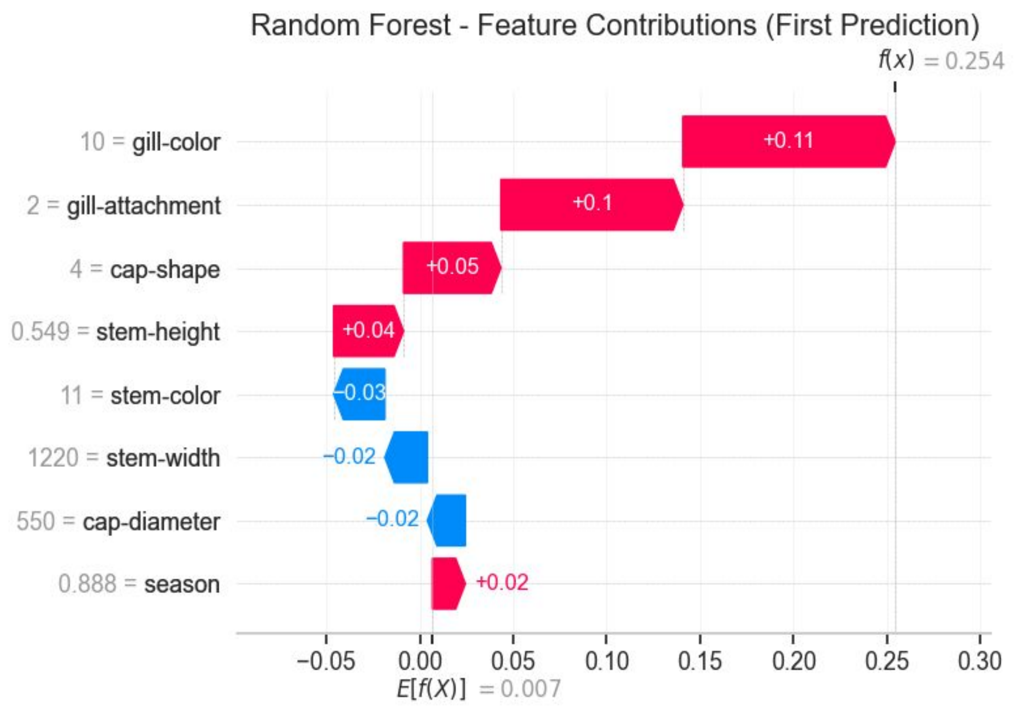


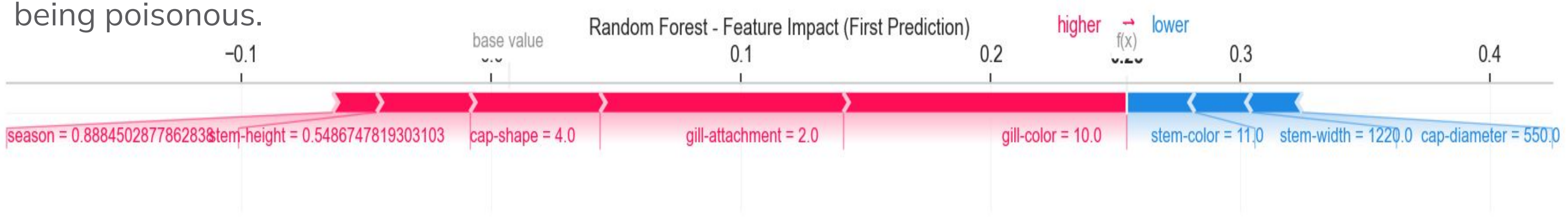
Feature Contribution:

Random Forest Waterfall & Force Plots :

→ The strongest positive influence comes from gill-color (+0.11) and gill-attachment (+0.10), suggesting that these characteristics increase the likelihood of the mushroom being classified as poisonous.

The strongest negative influence comes from stem-width (-0.02) and cap-diameter (-0.02), meaning these traits reduce the probability of the mushroom being poisonous.





06 Conclusion

Final Model Recommendation

1 Best Performing Models: Random Forest & TabNet

- High accuracy & recall → Ensures fewer misclassifications.
- Strong feature interpretation → Useful for real-world applications.
- Balanced interpretability & complexity.

2 Why Random Forest ?

- Real-world deployment → Ideal for automated mushroom classification.
- Reliable predictions → Combines accuracy & explainability.
- Computational efficiency → Compared to deep learning models requiring high resources.

Final Thoughts

- Successfully compared ML & DL models for mushroom classification.
- Integrated SHAP explainability to improve trust & interpretability.
- Identified feature importance & interactions across different models.

Thank You