Predicting Mushroom Edibility Using Supervised Learning Algorithms

Presenter

- Gabriel Angelo Vinas
- Ma. Mae Sid Santos

Background

Mushroom foraging is a popular recreational activity, but it carries significant risks as many poisonous species closely resemble edible ones, and misidentification can lead to severe poisoning or even death. Unlike some plants, there are no simple, universal rules for determining a mushroom's edibility, creating a compelling need for strong, data-driven decision support systems. This study addresses the challenge by formulating a mushroom identification as a binary classification problem, aiming to predict if a mushroom is edible or poisonous based on its physical characteristics. A critical requirement for any such model is the minimization of false negatives, that is, incorrectly classifying a poisonous mushroom as edible, as this error has the most severe consequences.

Objective

01. To develop and evaluate supervised machine learning models that can accurately predict whether a mushroom is edible or poisonous based on a set of 22 observable, physical characteristics.

02. To conduct a systematic investigation into the efficacy of four distinct and widely used classification algorithms: Logistic Regression, K-Nearest Neighbors, Random Forest, and Support Vector Machine.

03. To evaluate different strategies for handling a significant proportion of missing values in the stalk-root feature and investigate the trade-off between model complexity and predictive accuracy by systematically reducing the number of features.

Review of Related Literature

This research is grounded in the well-established UCI Mushroom Data Set, a benchmark for classification algorithms originally curated from The Audubon Society Field Guide to North American Mushrooms. Previous work on this dataset has consistently shown high classification accuracy, setting a high bar for new models. Our study conducts a comparative analysis using four foundational algorithms to address this problem. These include Logistic Regression, a statistical model that estimates the probability of a binary outcome by applying a logistic function to a linear combination of features; K-Nearest Neighbors, a non-parametric, instance-based method that classifies new data based on the majority class of its “k” closest neighbors; Random Forest, a powerful ensemble algorithm that constructs a mulititude of decision trees and uses bagging and random feature selection to reduce variance and overfitting; and Support Vector Machine, a model that finds an optimal hyperplane to maximize the margin between classes, using the”kernel trick” to handle non-linear data in high-dimensional space.

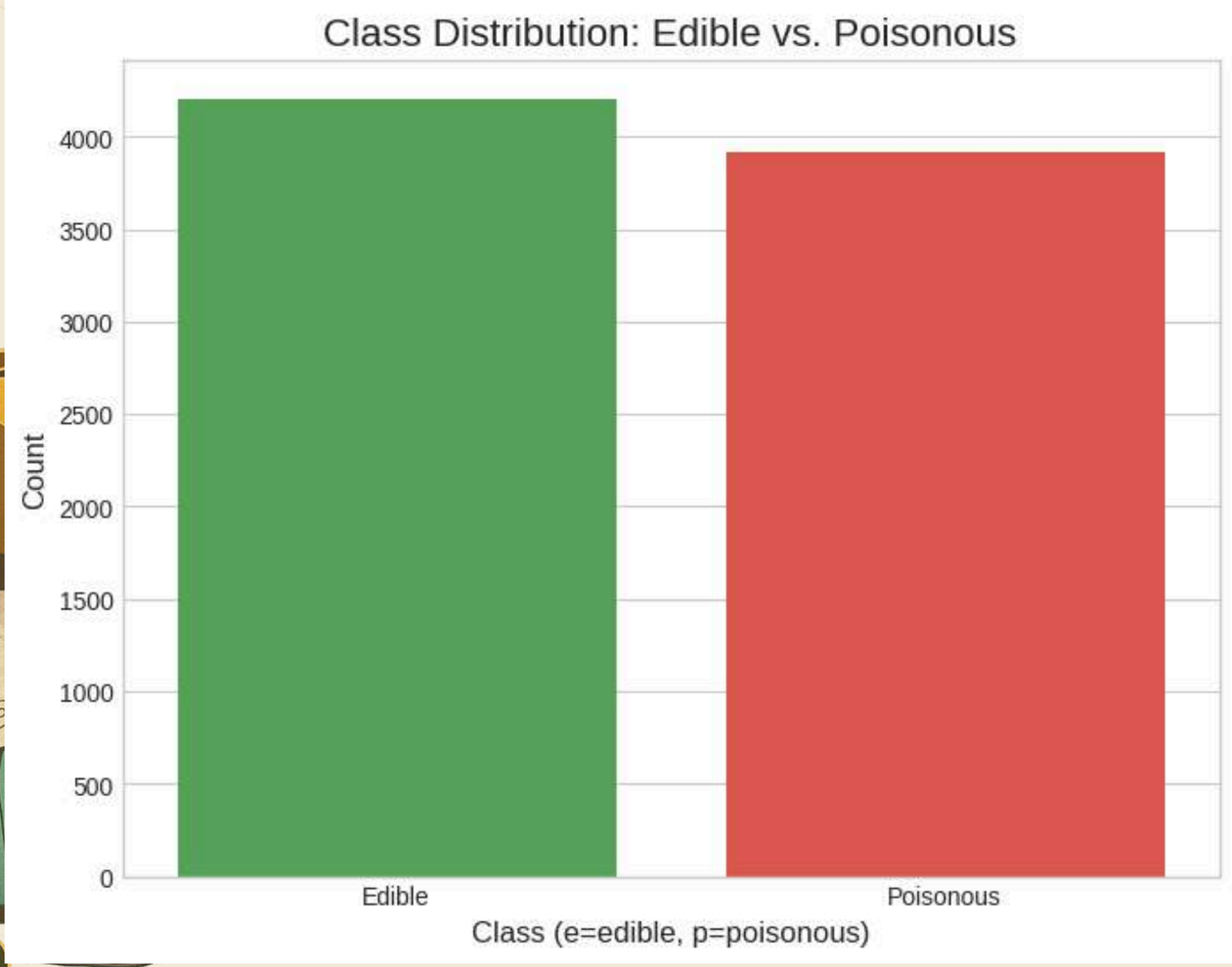
Methodology

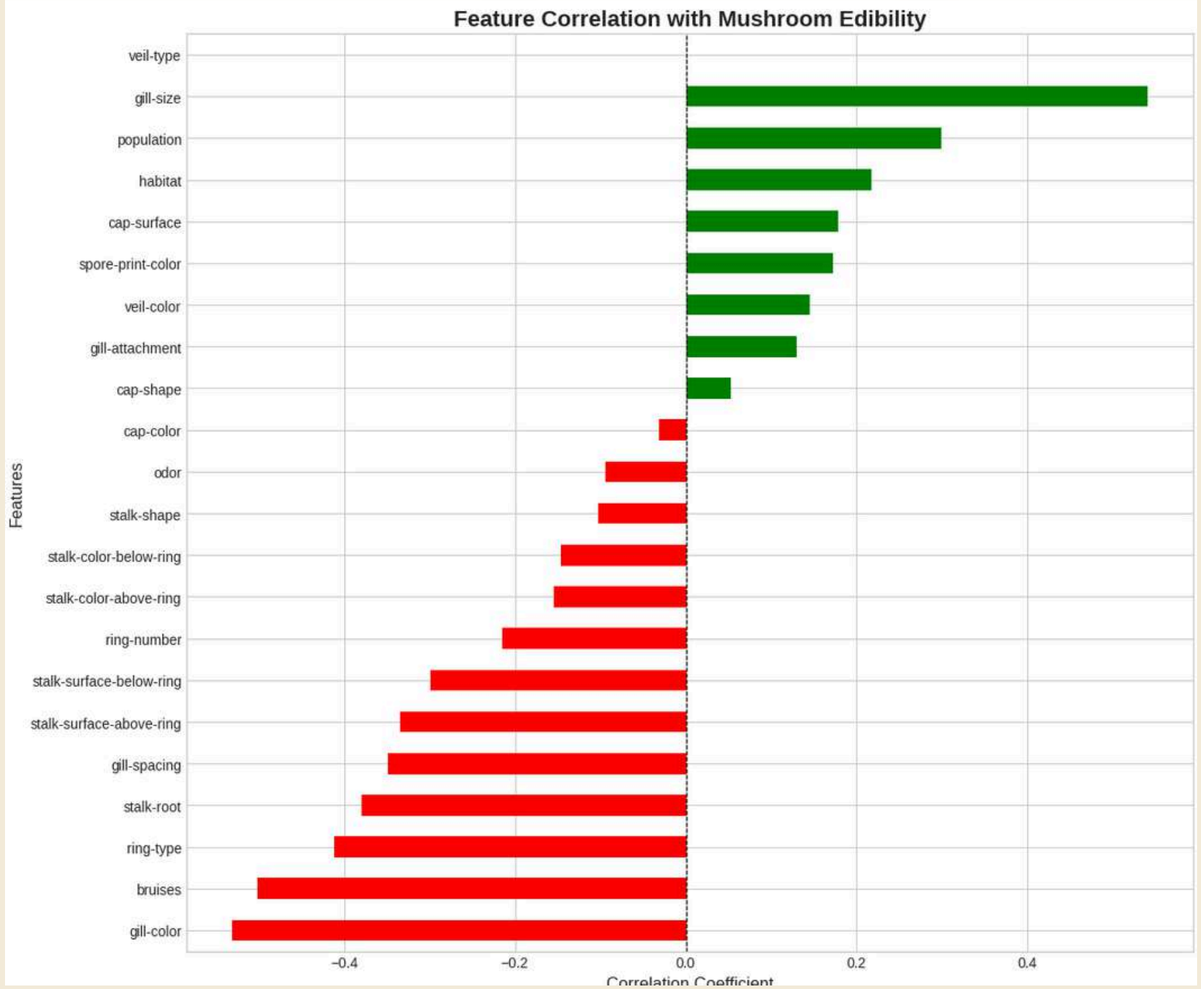
- Data Preprocessing: All 22 categorical features were incompatible with the algorithms. We employed one-hot encoding, which transformed the features into a 117-dimensional binary feature vector.
- Target Encoding: The binary target “class” was label-encoded from (e, p) to (0, 1), representing “edible” and “poisonous”.
- Implementation: All models were implemented in Python using the scikit-learn library and encapsulated within a sklearn.pipeline.Pipeline object to prevent data leakage and streamline evaluation.
- Data Split: The dataset was split into an 80% training set and a 20% testing set using stratified sampling to maintain the original class distribution.
- Evaluation Metrics: Model performance was assessed using Accuracy, Precision, Recall, and F1-Score.

Dataset

TABLE I: Selected Dataset Attribute Descriptions

Feature	Codes	Description
cap-shape	b, c, x, f, k, s	bell, conical, convex, flat, knobbed, sunken
odor	a, l, c, y, f, m, n, p, s	almond, anise, creosote, fishy, foul, etc.
gill-color	k, n, b, h, g, r, o, p, u, e, w, y	black, brown, buff, chocolate, gray, etc.
bruises	t, f	bruises, no bruises
habitat	g, l, m, p, u, w, d	grasses, leaves, meadows, paths, etc.





Experiment 1

- The Problem: A significant data quality challenge was the 'stalk-root' feature, where 2,480 instances (30.5% of the data) had a missing value denoted by '?'.

- Objective: To determine the optimal strategy for handling these missing values and quantify its impact on model performance.

Three Strategies Tested:

1. Strategy A (Categorical Treatment): Treat '?' as a distinct category, 'not observable', testing the hypothesis that its absence is informative.
2. Strategy B (Modal Imputation): Replace '?' with the most frequent (modal) value in the column, which is 'bulbous' ('b').
3. Strategy C (Feature Removal): Remove the entire 'stalk-root' feature from the dataset

Experiment 1 Results:

TABLE II: Performance Comparison of Stalk-Root Handling Strategies (Accuracy / F1-Score)

Model	Strategy A (Keep '?')	Strategy B (Impute)	Strategy C (Drop)
LR	99.94% / 99.94%	99.88% / 99.87%	99.88% / 99.87%
KNN	99.82% / 99.81%	99.82% / 99.81%	99.94% / 99.94%
RF	100.0% / 100.0%	100.0% / 100.0%	100.0% / 100.0%
SVM	100.0% / 100.0%	100.0% / 100.0%	100.0% / 100.0%

- Result: As shown in Table II, performance was "remarkably consistent" across all three strategies. The average accuracy for all models differed by less than 0.1%. Random Forest and SVM achieved 100% accuracy in all three scenarios.
- It suggests that stalk-root is not a critical feature for this dataset. and this experiment may contribute to practical implication in data gathering of mushrooms.
- Final Decision: We selected Strategy A (Keep '?')to use for our final model. This "prioritizes data integrity" and is a "more scientifically robust approach" as it avoids introducing artificial certainty through imputation or discarding data (like removal).

Experiment 2

Impact of feature selection

- Objective: To investigate the relationship between the number of features and model performance, and to determine if a simpler, more parsimonious model could achieve comparable accuracy.

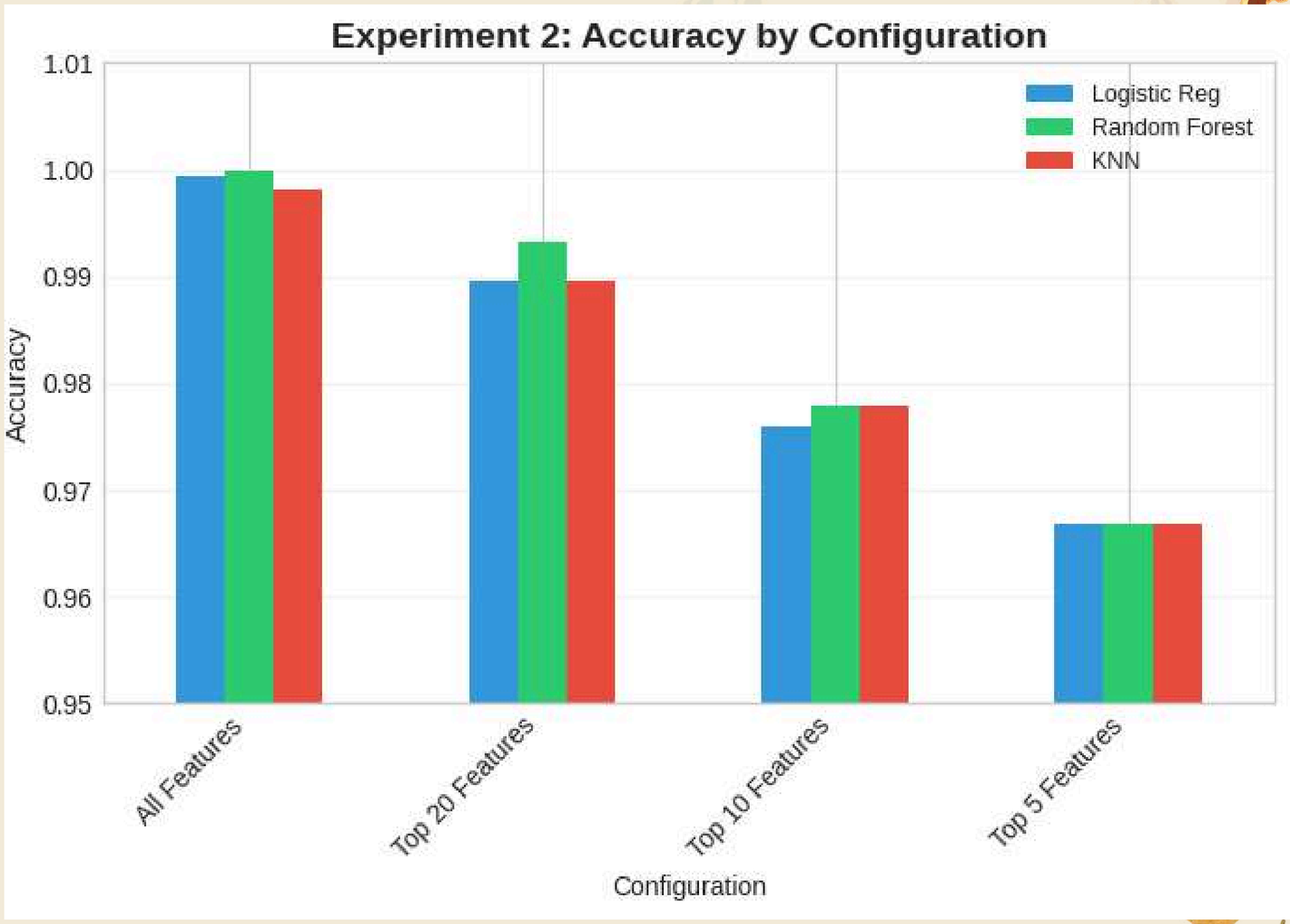
- Ranked all 117 one-hot encoded features using a Random Forest model's feature importance scores (Gini impurity reduction).
Trained and evaluated all four models on four distinct feature sets: All 117, Top 20, Top 10, and Top 5.

Experiment 2 Results:

Configuration	Average Accuracy
All Features	0.999179
Top 20 Features	0.990769
Top 10 Features	0.977231
Top 5 Features	0.966769

- Reducing features causes significant accuracy loss (3.24%)
- All features contribute meaningfully to predictions
- Model complexity is necessary for optimal performance

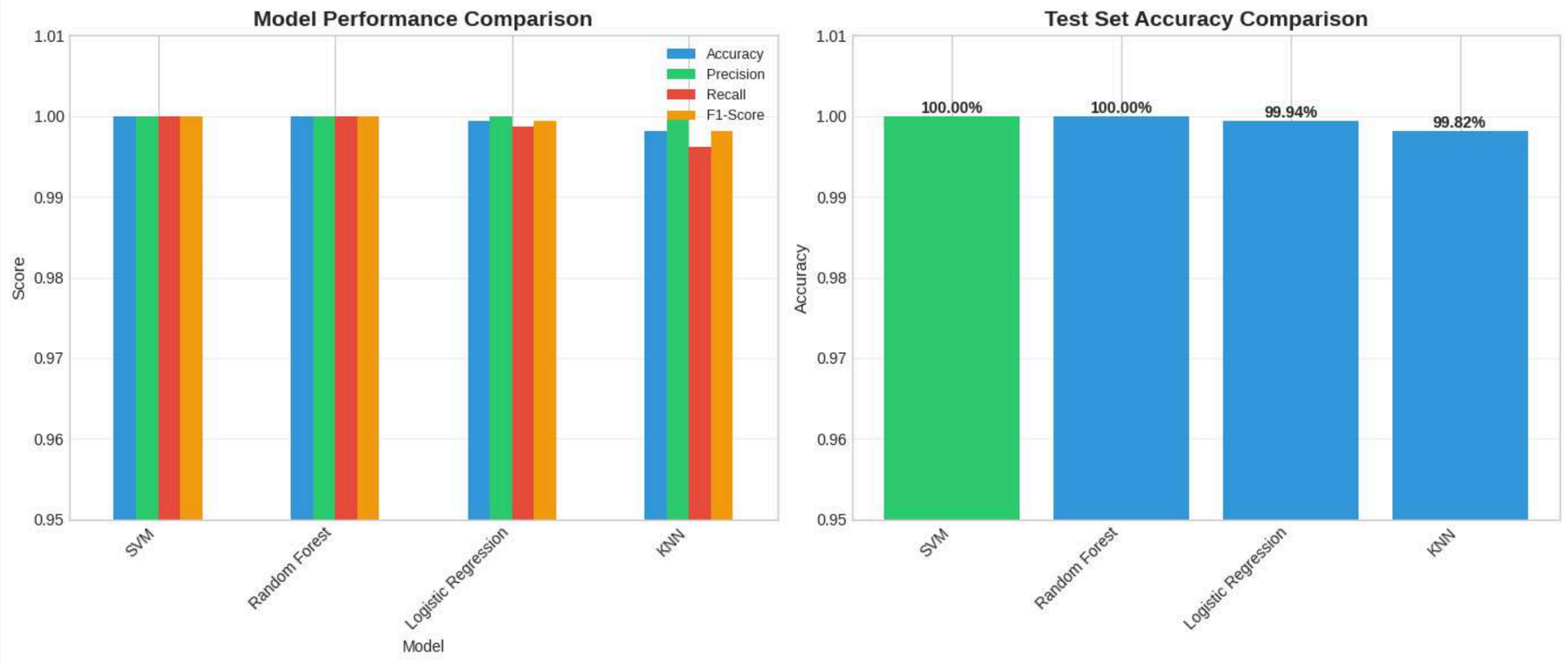
Final Decision: Use all features including the keeping of '?' in the stalk-root for production model to maintain accuracy and to keep the integrity of the original data.



Results & Discussions

TABLE III: Final Model Performance on Test Set

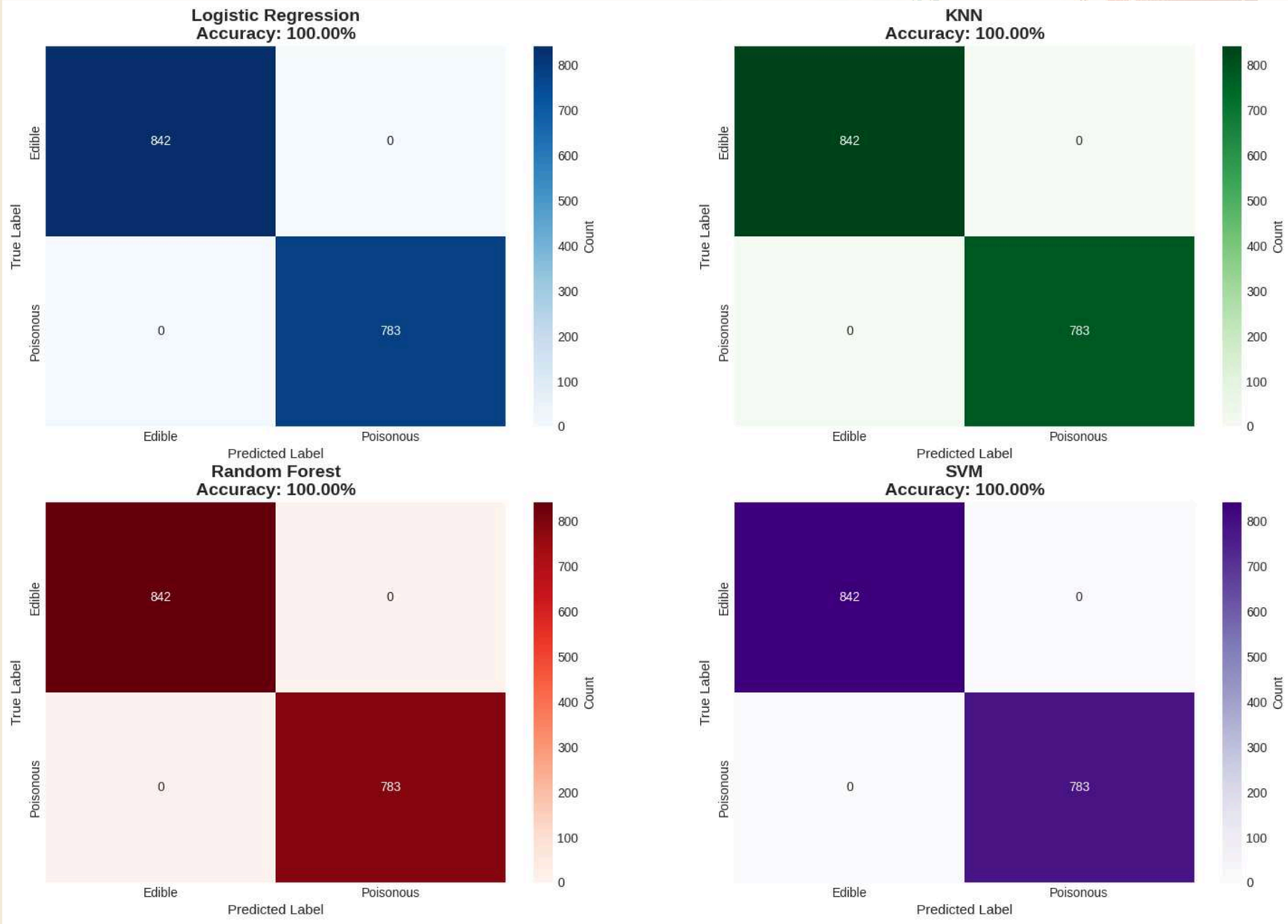
Model	Accuracy	Precision	Recall	F1-Score
LR	99.94%	100.0%	99.87%	99.94%
KNN	99.82%	100.0%	99.62%	99.81%
RF	100.0%	100.0%	100.0%	100.0%
SVM	100.0%	100.0%	100.0%	100.0%



Results & Discussions

Final Model Performance:

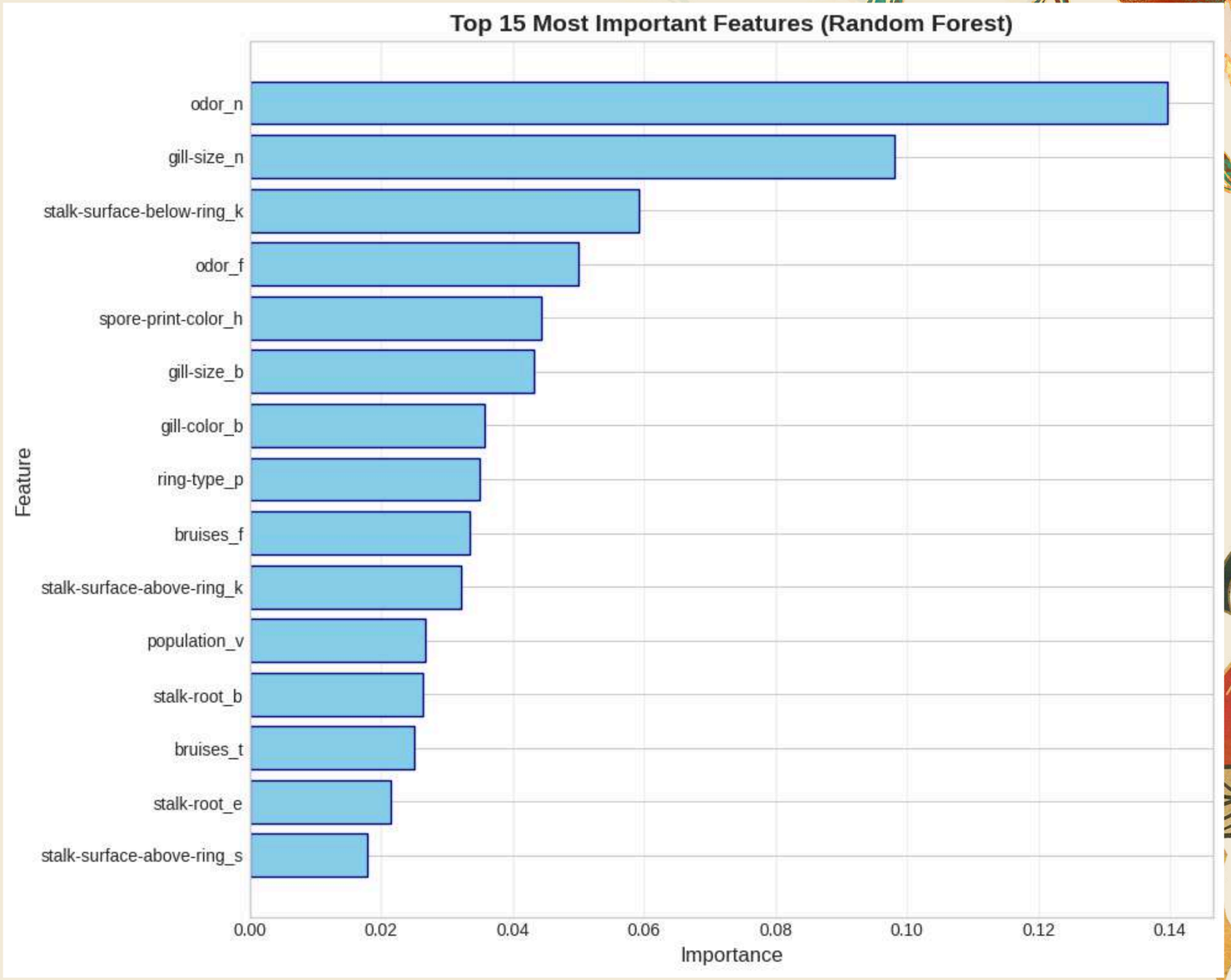
- Using all 117 features (Strategy A), both Random Forest and Support Vector Machine achieved perfect 100% scores across all metrics (Accuracy, Precision, Recall, and F1-Score) on the held-out test set.
- Critically, as shown in the confusion matrices, there were zero false negatives (a poisonous mushroom classified as edible) and zero false positives.



Results & Discussions

Dominant Predictive Features:

- Feature importance analysis revealed that features related to 'odor' are "overwhelmingly the most predictive".
- odor_n (odor=none) was the single most important feature, followed by odor_f (odor=foul). This aligns with anecdotal knowledge in mycology.



Conclusion

The analysis demonstrated that both the Random Forest and SVM models achieved perfect 100% accuracy, indicating that the features within the UCI Mushroom dataset are sufficient for fully deterministic classification of the species included. Among all features, the 'odor' attribute emerged as the most powerful and discriminative predictor, with the absence of odor or presence of a foul odor being particularly decisive. Additionally, the results showed that a comprehensive feature set is essential for optimal model performance, as the removal of any features consistently led to reduced accuracy.

However, a key limitation lies in the dataset itself. The exceptional performance is largely due to the dataset being clean and highly curated. As a result, this level of accuracy may not generalize well to the real-world complexity of wild mushrooms or species not represented in the original 23 classes.

Thank You!