Project 4 - Group 10

I. Introduction and Inspiration

We were inspired by the critical need for safe mushroom identification and existing tools that address this challenge. In particular, we looked at examples in Tableau that demonstrated the potential for interactive presentations of mushroom data. Our project aims to enhance these concepts by incorporating machine learning and creating a user-friendly solution that improves the assessment of mushroom edibility.

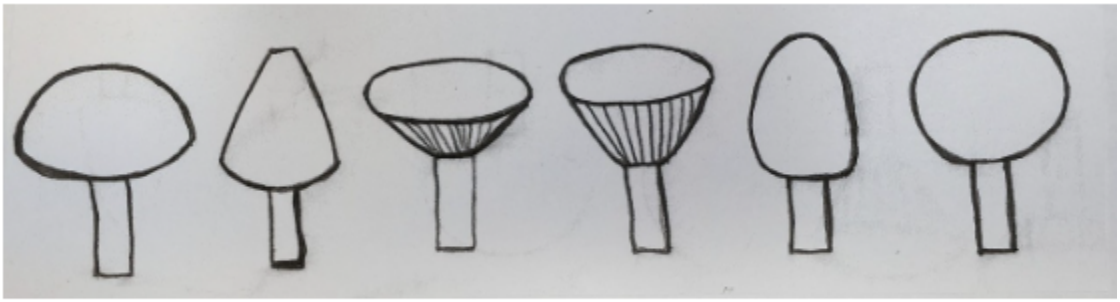
II. Mushroom Dataset (Binary Classification)

We found the dataset below by looking at different Machine Learning Datasets on the Kaggle website.

> The dataset is a cleaned version of the original Mushroom Dataset for Binary Classification Available at UCI Library. This dataset was cleaned using various techniques such as Modal imputation, one-hot encoding, z-score normalization, and feature selection (Sawhney, 2024).

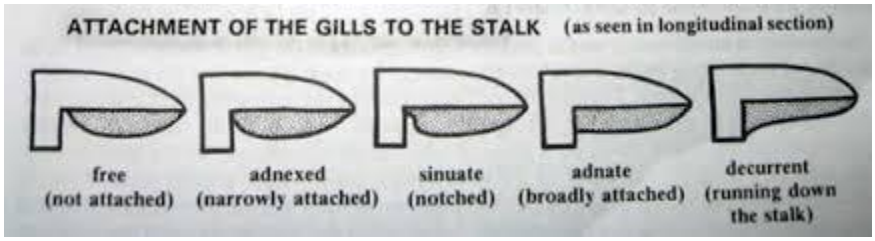
There are 9 variables including cap diameter, cap shape, gill attachment, gill color, stem height, stem width, stem color, season, and class. Class is the target variable, either

edible or poisonous. The cap shape and gill attachment are special features that require an explanation. The mushroom caps are categorized into six shapes listed below.



Convex Conical Flat Sunken Bell-Shaped Spherical

The gills have five different attachments shown below (or none).



III. Data Cleaning and Machine Learning Experiment

In our Jupyter Notebook, we first imported the pandas, matplotlib, pickle, xgboost, lightgbm, and scikit-learn machine learning libraries. Then we reviewed the rows, columns, metrics, data types, and null values. Some duplicate rows were dropped, but otherwise, the dataset was set.. The numeric and categorical features were separated and the numeric features were scaled using the Standard Scaler, before combining them into a single dataframe.

num_features = ['cap-diameter', 'stem-height', 'stem-width']

cat_features = ['cap-shape', 'gill-attachment', 'gill-color', 'stem-color', 'season', 'class']

Next, we ran a correlation analysis on the dataset. There was a strong correlation of .83 between cap diameter and stem width. Strong correlations in machine learning can lead to issues, however, since our project is to predict vs. Infer, it shouldn't hurt our final analysis.

The columns were separated into X and y variables. All the columns, except for "class", were combined into X, and "class" was put into y. We used "train test split", to separate the data into X_train, X_text, y_train, and y_test. A classification function was used for training and testing, generating metric tests, and visualizations. These are all helpful tools in analyzing which machine model performs best. The metrics used included the confusion matrix, classification report, and roc- auc score. The visuals are the confusion matrix and roc curve.

The following are the classification models that we tested:

- Logistic Regression
- Decision Tree Classifier
- Random Forest Classifier
- AdaBoost Classifier
- Extra Trees Classifier
- Gradient Boosting Classifier

- SVC
- K Neighbors
- XGB Classifier
- LGBM Classifier

After running all of the models, we compared the results from the training and testing data. We looked at the scores for precision, recall, F1-score, accuracy, the confusion matrix, the AUC, and ROC curve. We also printed the feature importance list for the top models.

Finally, we concluded that the Random Forest Classifier worked the best, with Extra Trees and XGB coming in a close second. The model that performed the worst was the Logistic Regression. Logistic Regression assumes a linear relationship between variables. If there is no correlation between the variables and the target variable, the model will not perform well. Below is a chart comparing the Logistic Regression metrics to the Random Forest.

LOGISTIC REGRESSION	TRAIN	PRECISION	RECALL	F1-SCORE	SUPPORT	ACCURACY	CONFUSION	MATRIX	AUC	ROC CURVE
	0.00	0.60	0.53	0.56	18270	0.63	9623	8647	0.68	0.67
	1.00	0.64	0.71	0.67	22029		6435	15594		
	TEST	PRECISION	RECALL	F1-SCORE	SUPPORT	ACCURACY	CONFUSION	MATRIX	AUC	
	0.00	0.59	0.52	0.55	6090	0.62	3170	2920	0.67	
	1.00	0.64	0.70	0.67	7343		2208	5135		
RANDOM FOREST	TRAIN	PRECISION	RECALL	F1-SCORE	SUPPORT	ACCURACY	CONFUSION	MATRIX	AUC	ROC CURVE
	0.00	1.00	1.00	1.00	18270	1.00	18270	0	1.00	1.00
	1.00	1.00	1.00	1.00	22029		0	22029		
	TEST	PRECISION	RECALL	F1-SCORE	SUPPORT	ACCURACY	CONFUSION	MATRIX	AUC	
	0.00	0.99	0.99	0.99	6090	0.99	6030	60	0.99	
	1.00	0.99	0.99	0.99	7343		69	7274		

What is a Random Forest Algorithm?

In supervised machine learning applications, Random Forest is a flexible and powerful ensemble learning technique that is especially useful for classification and regression issues. During the training phase, it builds a large number of decision trees and outputs the mean prediction (for regression) or the mode of the classes (for classification) of each individual tree. Random Forest is an appealing choice for many real-world applications because it is resistant to noise and outliers, manages high-dimensional datasets effectively and yields estimates of feature relevance (R., 2024).

Feature importance is a step in building a machine learning model that involves calculating the score for all input features in a model to establish the importance of each feature in the decision-making process. The higher the score for a feature, the larger effect it has on the model to predict a certain variable (Shin, 2023).

The list of feature importance for the Random Forest is below. It shows the stem width being the strongest feature. Gill attachment, cap diameter, stem color, gill color, and stem height are extremely close, with the cap shape and season having the least importance.

	Feature	Importance
5	stem-width	0.226915
2	gill-attachment	0.140512
0	cap-diameter	0.127287
6	stem-color	0.127022
3	gill-color	0.123766
4	stem-height	0.121000
1	cap-shape	0.095449
7	season	0.038049

The final We fit the data to the Random Forest model. The mushroom scaler and the mushroom model were saved as a pickle file for the next step in our project.

IV. Flask App

Our data set was presented as indexed data, as required for the model. This left all of the categorical data transformed to an index removing any strings. As such, our Random Forest model in our pickle file was expecting indexed data to be imputed to the model. This would not be user friendly as we need the categorical values to be in the web application for the user. For the model to run effectively we used both a flask application with our makePredictions function and a helper.py file to make the conversion between the index and the string values. The makePredictions function was written in python in our flask app along with our model imported as a pickle file.

The helper file was needed to convert the categorical values in the web app form back to the indexed values for the model to use to run its predictions. We also needed to return the form values selected by the user to the same data frame format that the model was using. It was necessary to have the selected values match exactly to the format in the model to prevent errors. This was done in the flask application and executes when the function is called to run the model and make predictions.

In the .js file for our prediction page we have both the function that applies when the make makePrediction function is called and several event listeners that are adjusting the values sent to the function when changed by the user. The .js is setting up both the predictions page and the form that will be used for the application. We included default values and sliders for the metric values so that there would not be null values entered onto the model. When the make predictions button is selected the values in the form are transformed into a data frame and are run through the model to output a predicted result.

V. Color Design

With our research topic being all about determining a mushroom's edibility, we wanted to select an earthy color palette to tie our visual elements together. This is represented by a brown to green color range used in elements of the web application as well as both tableau dashboards.

Our target variable is whether something is edible or poisonous, so we elected to use green to indicate poisonous. The color green has a long history of association with poisons and toxic elements as well as association with plants and nature. This allows us to

draw a natural visual association with our data through color choice. While you may not have an issue with eating a normal brown-colored mushroom, you may think twice before finding one covered in green!

VI. Dashboard Design

Mushroom Edibility - this dashboard takes up the visible space in the web browser in order to take advantage of the dense visualizations here. We begin with a scatter plot to chart stem height vs stem width to determine if any visual trends can be observed to indicate a particular size ratio that could be an indicator of a mushroom's edibility. The brown "edible" mushroom dots are densely packed on the bottom left quadrant of the plot with some scattered more toward the top and some on the right. Focusing on the green "poisonous" mushroom dots, we notice a similar trend with some distinct clustering along the bottom with a stretch of poisonous dots along a similar stem width along the entire stem height spectrum. There is similar visual striping of the plots across each individual cap shape.

Additionally, we have a tree map to show a visual representation of how cap shape and stem color combine in relation to whether a mushroom is poisonous or edible. This helps to show a visual ratio of how often these features are observed in either poisonous or edible mushrooms. Using the filters, you can narrow down to a particular shape/color and see if one color is more prominently noticed in a specific shape for example.

We finish out with some bar charts to display the count of mushrooms across a few feature types. The bar chart reveals some gill colors such as orange and gray occur much more frequently than other gill colors. Our dot plot provides a deeper visualization of the

relationship between gill color and gill attachment and how often these feature combinations are observed between both edible and poisonous mushrooms. Lastly, our stacked bar chart shows a direct visual connection for each cap shape of if a particular shape is observed to be more poisonous or more edible overall.

VII. Dashboard and Machine Learning

The relationship between gill attachment and edibility reveals specific attachment types that may be more common in edible species. The stem color analysis indicates certain colors are more frequently associated with edible mushrooms, while the seasonal data shows distinct trends in mushroom abundance, identifying optimal foraging periods. Lastly, the gill color distribution highlights prevalent colors among both edible and poisonous varieties, aiding in visual identification and enhancing forager safety. Overall, the dashboard serves as a comprehensive guide for understanding mushroom characteristics and their safety for consumption.

VIII. Bias/Limitations/Future Work

The data had many features of mushrooms, but were missing critical features that can be used to determine edibility. Bruising vs. Bleeding, cap-surface, gill-spacing, stem root and surface, veil type and color, ring type, habitat, odor, and spores were not included in the cleaned data. The spores color is often used to identify similar-looking mushrooms in the field, but was not available in this data set likely because releasing the spores can take some time and is not immediately available to individuals when identifying. Adding this

information to the model can provide critical information for the model to learn with and use to make predictions that would increase accuracy.

We also noticed that there were only 12 color values, however the stem color had 13 options for color. Similarly, some species of mushrooms do not have gills; therefore, a "none" option was included, but the data did not include a "none" option for gill color. If the mushroom does not have gills, it can not have a color.

Additionally, the nature of mushrooms is that many different species can look identical. While the model can train very well with high accuracy and recall, the act of describing the mushroom can still lead to errors in the predictions as sometimes only very subtle differences can determine species. Geographical data could help narrow down very similar mushroom species and could be added to the web application if we had that data to train our model.

The Random Forest Model also has some limitations. The complexity and memory usage can be difficult for systems with limited resources and lack of interpretability can make it hard to recognize the relationships between features and how they affect predictions.

In the future, adding additional features and location data would greatly increase accuracy and usability. We also have to trust that the person who created the dataset did it correctly. Although the dataset was clean and great for our project, future work would entail data cleaning and engineering using the original dataset.

IX. Conclusions/Reflections

Our project focuses on enhancing mushroom identification through machine learning. We explore key questions about the differences between edible and poisonous mushrooms, the influence of seasons on their safety, and the significance of color versus size in identification. Our interactive Tableau visualizations help users understand these insights easily. By combining these findings with a user-friendly design, we aim to empower foragers to identify mushrooms confidently and safely, ultimately fostering a deeper appreciation for the diverse world of mushrooms.

From our analysis we can see that while many mushrooms do share common factors such as general appearance and color, they still vary in their toxicity. Many mushroom species mimic one another making identification even by the unique factors difficult. From the visualizations of the data we see that there are generalities we can make about the characteristics of poisonous mushrooms but many factors are shared between species, edible or not. The most accurate results would be from data that is specific and detailed in ways that our model could continue to train on. Machine learning is an excellent tool for educational and identification purposes that should be used in conjunction with human expertise.

Disclaimer

The information provided is for educational purposes only and should not be used as a guide for consuming wild mushrooms. Mushroom identification is complex, and consuming the wrong species can result in serious illness or death. Always consult an expert before consuming any wild mushrooms.

Works Cited - Alphabetical order

(n.d.). *A banner of various mushrooms. Vector illustration of fungus. Drawing of voluminous fungi. Pro Vector.* Vecteezy. Retrieved September 30, 2024, from https://www.vecteezy.com/vector-art/14315476-a-banner-of-various-mushrooms-vector-illustration-of-fungus-drawing-of-voluminous-fungi

(n.d.). *Black mushroom icon*. ICONSDB.COM. Retrieved September 30, 2024, from https://www.iconsdb.com/black-icons/mushroom-icon.html

R. (2024, February 15). *What are the Advantages and Disadvantages of the Random Forest?* Geeksforgeeks.org. Retrieved October 1, 2024, from https://www.geeksforgeeks.org/what-are-the-advantages-and-disadvantages-of-random-forest/

Sawhney, P. (April 2024). *Mushroom_cleaned_csv, Version 1.* https://www.kagglecom/datasets/prishasawhney/mushroom-dataset/data

Sci Rep 11, 8134. Retrieved September 27, 2024, from https://doi.org/10.1038/s41598-021-87602-3

Shin, T. (2023, November 7). *Understanding Feature Importance in Machine Learning.* Builtin.com. Retrieved October 1, 2024, from https://builtin.com/data-science/feature-importance#:~:text=Feature%20importance%20is%20a%20 step,to%20 predict%20a%20 certain%20variable.

Wagner, D., & Hattab, G.(2021).*Secondary Mushroom [Dataset].* UCI Machine Learning Repository. Retrieved September 27, 2024, from https://doi.org/10.24432/C5FP5Q

Wagner, D., Heider, D. & Hattab, G. (2021).*Mushroom data creation, curation, and simulation to support classification tasks. UCI Machine Learning Repository. Retrieved September 27, 2024, from https://www.nature.com/articles/s41598-021-87602-3*