You will have to form a group of 3 to 4 students.

Choose a dataset from the UC Irvine Machine Learning Repository (https://archive.ics.uci.edu/ml/index.php) with at least 5000 instances and 20 attributes for classification or regression. Compare how the different approaches seen in class perform on this dataset to predict accurately the classes or the values of the unlabeled data. You should determine what are the best hyper-parameters for each approach you are using. You could use any Python libraries.

<https://archive.ics.uci.edu/dataset/73/mushroom>

First Project: Machine Learning Analysis

This data set includes descriptions of hypothetical samples corresponding to 23 species of gilled mushrooms in the Agaricus and Lepiota Family (pp. 500-525).

Each species is identified as definitely edible, definitely poisonous, or of unknown edibility and not recommended.

This latter class was combined with the poisonous one.

The Guide clearly states that there is no simple rule for determining the edibility of a mushroom; no rule like ``leaflets three, let it be'' for Poisonous Oak and Ivy.

**Abstract**

Have you ever been walking down the streets of Windsor and been plagued by the question of determining the edibility of a mushroom? [1] While mushroom foraging has a rich tradition and a community of enthusiasts, it also poses inherent risks. This comes as some species can be toxic and potentially fatal if consumed. Reliable identification of edible and non-edible mushrooms is crucial for the safety and enjoyment of foragers and nature enthusiasts alike. This research endeavor delves into the mixture of machine learning methods with the natural world. Therefore, the main goal of this research endeavor is to develop a robust and accurate method for classifying the edibility of gilled mushrooms, thereby, providing a valuable tool for those exploring the world of mycology.

1. **Introduction**

The root of the problem lies in the difficulty of reliably distinguishing between edible and non-edible mushrooms. This challenge is compounded by the wide variety of mushroom species with subtle visual differences. Machine learning presents a compelling opportunity for tackling the task of classifying the edibility of mushrooms. Its capacity to analyze extensive datasets, detect intricate patterns, and provide data-driven insights aligns seamlessly with the complex task of mushroom identification. Through this research, we intend to harness the power of machine learning to revolutionize the way we approach the age-old practice of mushroom foraging, making it safer, more accessible, and more enjoyable for all. As we examine the details of our dataset and data preprocessing in the following sections, we will highlight the steps taken to prepare the data for our machine learning models.

1. **Dataset**

The dataset encompasses various mushroom characteristics, including cap shape, cap surface, cap color, bruises, odor, gill attachment, gill spacing, gill size, gill color, stalk shape, stalk root, stalk surface above and below the ring, stalk color above and below the ring, veil type, veil color, ring number, ring type, spore print color, population, and habitat. These attributes provide a rich and diverse set of features for our machine learning models to analyze.

\*\*mention the size of the dataset and some of the features, (edible vs poisonous)

[] the dataset somewhere

1. **Data Preprocessing**

**Cleaning and Feature Selection**

An initial visual inspection of the data presented columns that had null values. Pandas was used to load the comma separated file (csv) and using a small function [pic] the data was quickly checked for the null values. The conclusion was that only 1 column had missing values. The one column was labelled ‘stalk-root’ with a total of 2480 missing values. Since, the entire dataset contained 8416 samples this resulted in a significant portion of the data that would require data imputation. This is the process of filling in missing or null values with estimated or predicted values. Given that the 29% of the data in this column would need to be altered and inevitably affect the model’s precision, the column was dropped.

The next step was to determine what columns would help in classifying the data and which columns would not contribute. This was done by checking for the number of unique values in each of the remaining columns. [add table pic] This exposed a column that remained constant throughout the dataset. A column that is constant are considered “zero-variance” features because they do not change over the dataset and would not affect our classification. Because of this justification the ‘veil-type’ column was subsequently dropped.

Reread

As the dataset description listed missing values, the first step was to explore and know what columns had missing values, as that is a problem which is better tackled first, and through a small function we wrote and using pandas to describe the dataframe, we came to the conclusion that only 1 column had missing values which was the ‘stalk-root’ column with 2480 missing values, which was very large, so we ended up dropping the column. Next step was to do feature selection, where we determined what columns would help in classifying the data, in order to determine that, we checked for the number of unique values in each column, which is a good way to eliminate features which remain constant throughout the dataset as these won’t affect our classification. The describe function was useful here as well as it displayed number of unique values of each column. By doing this, we determined the ‘veil-type’ column to be dropped as it had a 1 unique constant value.

**Encoding and Processing Data**

In order for the models to be trained, the data needs to be processed in a way that is easy for the models to perform tasks on, the dataset consists of string data which is not ideal, so we utilize LabelEncoder() from the Sci-kit learn library in order to convert our categorical/text data into numerical values. This is useful for machine learning models, as they require numerical input. The mappings from original to numbers, are stored in a dictionary structure(map) to utilize them for future tasks.

To further preprocess the data, StandardScaler() from scikit-learn is used to standardize features, which means that it transforms the data into a form where it has a mean of 0 and standard deviation of 1, i.e. scaling it to unit variance. This is a useful and important step as it would be harder for the models to use data that to train with features on different scales. Finally, we divide the data into training data, the data that the model learns from, and testing data, data from which the model’s robustness is tested on data it hasn’t seen during training. The goal here is to classify if a mushroom is edible or poisonous, so we create 2 separate sets of data, one with the labels of edible and poisonous, which was the class column and one without it and all the other data. this makes it so that the models are exclusively trained on the feature data and evaluated against the actual labels. After the division, we perform an 80-20(80% percent training, 20% testing) split on the dataset and divide them into x\_train, x\_test, y\_train, y\_test and use them for the training and testing respectively.

1. Model Selection
2. Model Training and Evaluation
3. Hyper-Parameter Tuning
4. Comparison of Results
5. Conclusion and Findings
6. Description of Student Participation

References

[1] – fun guyz

dataset

<https://archive.ics.uci.edu/dataset/73/mushroom>