Mycological Machine Learning

Background

Every year thousands of people are sickened and in some cases die as the result of consuming poisonous mushrooms ("Foraging Fatality Statistics 2016"). While foraging for mushrooms is a fun activity and a great way to learn about the natural world, one must be cautious about consuming wild mushrooms. The key to the safe consumption of wild mushrooms is correctly identifying the mushroom species. However, mycologists estimate that there are over 10,000 different species of mushrooms on the planet (Mushroom Appreciation). As a result, identifying the correct mushroom can be difficult--especially to the untrained eye. One way to help mushroom foragers correctly identify whether a mushroom is poisonous or not is to use a trained machine learning model to make accurate predictions.

Proposed Solution

The UCI Machine Learning Repository features a mushroom dataset that can be used to train a model to predict poisonous mushrooms based on various mushroom characteristics. The data is taken from the Audobon Society Field Guide, and it features 22 features and 1 class of poisonous or edible. In order to help prevent mushroom foragers from mistakenly eating poisonous mushrooms, I will train a classification model on the UCI mushroom dataset. Once the model is trained I will deploy it via AWS Sagemaker. Then I will build a web app where users will be able to enter various characteristics of a mushroom, and the model will predict whether it thinks the mushroom is poisonous or not.

Benchmark

The mushroom dataset has existed for over three decades. Therefore, it has been used in numerous machine learning projects. On Kaggle, there is a kernel titled Mushroom Classification that compares the precision and recall of seven different models trained on the dataset. The highest average precision and recall that any of the models achieve on the test data is 93 percent. (SVC, K-NN, and Random Forest each achieve 93 percent.) As a result, I will train my model with the goal of achieving a benchmark average total of at least 90 percent precision and recall on the test data, and I will consider the three aforementioned models to be the benchmark models that I will compare my trained model to.

Evaluation

As I noted above, precision and recall are the most important evaluation metrics that I will consider. However, recall will be the key metric since I want to ensure that all “positive” or poisonous mushrooms are actually labeled as poisonous. Still, I want to be careful to not have a low precision either. Therefore, I will likely compute the predictions’ F1 score to determine the “harmonic mean of precision and recall” (Koehrsen). One visual aid that will be helpful in evaluating the model’s performance is a confusion matrix. Including a confusion matrix as a visual aid will help readers better understand the interplay between precision and recall.

Project Design

I will create a Juptyer notebook within AWS Sagemaker to train and deploy my model. This is also where I will preprocess the data and display all of the relevant data visualizations. Since the issue I am examining is a classification problem, I plan to use Sagemakers’ built-in Linear Learner algorithm. Since the nanodegree course materials have focused on using Sagemaker, I intend to use Sagemaker as much as possible to gain a better understanding of how its built-in algorithms perform classification tasks.

The data itself is textual, so it will need to be encoded as integers as part of the preprocessing step. One hot encoding will be useful for this task. Since the data includes 22 features, I will have to determine if it makes sense to utilize each feature or not. Principal

Component Analysis will come in handy at this step. I will also consider other feature selection techniques to help me find the best features for training the model.

After my model is trained and deployed I will use AWS Lambda and API Gateway to create a public endpoint that is accessible via a web app. The web app will allow users to select various feature values from a dropdown menu. This data will be sent to the model, and the model will tell the user whether it believes the entered characteristics belong to a poisonous or edible mushroom.