Machine Learning Engineer Nanodegree

Capstone Project-Mushroom Classification

Lin Zhu

Jan 31st, 2017

## I. Definition

\_(approx. 1-2 pages)\_

**### Project Overview**

In this section, look to provide a high-level overview of the project in layman’s terms. Questions to ask yourself when writing this section:

- \_Has an overview of the project been provided, such as the problem domain, project origin, and related datasets or input data?\_

- \_Has enough background information been given so that an uninformed reader would understand the problem domain and following problem statement?\_

**Introduction**

Mushroom hunting, also known as mushrooming or mushroom picking, describes the activity of gathering mushrooms in the wild, typically for eating. In common sense, generally mushrooms with bright color, special cap shape or wired smell are suspicious as toxic class, but is it all of the truth? (Pictures for toxic mushrooms). The pictures show three toxic wild mushrooms that are always mistakenly classified as edible food. They all looks normal but are fatal to human being. Therefore, how to identify the edibility of a certain types of mushroom?

In this project, I utilized the mushroom dataset from UCI machine learning repository, to train and compared multiple machine learning models. The goal of our endeavors is to determine whether or not a mushroom would kill you. Given death is an extreme consequence for guessing incorrectly, which is hard to accept, the optimal model should have least error. Meanwhile, it is interesting to explore what features matter most in identifying the edibility.

**### Problem Statement**

In this section, you will want to clearly define the problem that you are trying to solve, including the strategy (outline of tasks) you will use to achieve the desired solution. You should also thoroughly discuss what the intended solution will be for this problem. Questions to ask yourself when writing this section:

- \_Is the problem statement clearly defined? Will the reader understand what you are expecting to solve?\_

- \_Have you thoroughly discussed how you will attempt to solve the problem?\_

- \_Is an anticipated solution clearly defined? Will the reader understand what results you are looking for?\_

**Dataset**

**Goal**

In this study, I am going to using it to answer the following tasks:

* Clean and preprocessing “Mushroom.csv” dataset
* Apply multiple machine learning models to predict the mushroom edibility in testing dataset.
  + Random Forest, SVM, Xgboost, Neural Network, train models with the training dataset and make predictions of testing dataset
* Compare and decide which machine-learning models perform best on this dataset?
  + Compare those evaluation metrics across models.
* Which features are most indicative of a poisonous mushroom?
  + Explore the correlation between multiple features
  + Apply dimension reduction methodologies (PCA)
  + Identify the top 3 features

**### Metrics**

In this section, you will need to clearly define the metrics or calculations you will use to measure performance of a model or result in your project. These calculations and metrics should be justified based on the characteristics of the problem and problem domain. Questions to ask yourself when writing this section:

- \_Are the metrics you’ve chosen to measure the performance of your models clearly discussed and defined?\_

- \_Have you provided reasonable justification for the metrics chosen based on the problem and solution?\_

The label column contains binary label “e” (edible) and “p” (poisonous). Incorrect prediction especially labeling “toxic” mushrooms as “edible” might bring in fatal problems. According to intimal observation, with an “e” over “p” ratio of 1.07the dataset is balanced between “edible” and “poisonous”.

Accuracy score is commonly used in binary classifier evaluation. It considers both true positive prediction and true negative predictions with equal weight.

ROC (Receiver Operator Characteristic) is also a good way to compare different classifiers, with consideration of true positive rate and false positive rate. AUC (Area under curve) computes and finds the point that is furthest from the “chance” line

(ROC curve)

Running time is another consideration. Sometimes there are trade off between the accuracy and running times. The differences in running time would increase along the data size extending. It would also generate high computing cost.

**## II. Analysis**

\_(approx. 2-4 pages)\_

**### Data Exploration**

In this section, you will be expected to analyze the data you are using for the problem. This data can either be in the form of a dataset (or datasets), input data (or input files), or even an environment. The type of data should be thoroughly described and, if possible, have basic statistics and information presented (such as discussion of input features or defining characteristics about the input or environment). Any abnormalities or interesting qualities about the data that may need to be addressed have been identified (such as features that need to be transformed or the possibility of outliers). Questions to ask yourself when writing this section:

- \_If a dataset is present for this problem, have you thoroughly discussed certain features about the dataset? Has a data sample been provided to the reader?\_

- \_If a dataset is present for this problem, are statistics about the dataset calculated and reported? Have any relevant results from this calculation been discussed?\_

- \_If a dataset is \*\*not\*\* present for this problem, has discussion been made about the input space or input data for your problem?\_

- \_Are there any abnormalities or characteristics about the input space or dataset that need to be addressed? (categorical variables, missing values, outliers, etc.)\_

**Dataset Overview**

The UCI Machine Learning repository has a mushroom dataset, consisting of descriptions of hypothetical samples corresponding to 23 species of gilled mushrooms in the Agaricus and Lepiota Family Mushroom drawn from The Audubon Society Field Guide to North American Mushrooms (1981).

The “Mushroom.csv” dataset has 8124 sets of data points. One column “Class” including “e” as edible and “p” as “poisonous”, as label. The other 22 feature columns are shown below:

- cap-shape: bell=b,conical=c,convex=x,flat=f, knobbed=k,sunken=s

- cap-surface: fibrous=f,grooves=g,scaly=y,smooth=s

- cap-color: brown=n,buff=b,cinnamon=c,gray=g,green=r,pink=p,purple=u,red=e,white=w,yellow=y

- bruises: bruises=t,no=f

- odor: almond=a,anise=l,creosote=c,fishy=y,foul=f,musty=m,none=n,pungent=p,spicy=s

- gill-attachment: attached=a,descending=d,free=f,notched=n

- gill-spacing: close=c,crowded=w,distant=d

- gill-size: broad=b,narrow=n

- gill-color: black=k,brown=n,buff=b,chocolate=h,gray=g, green=r,orange=o,pink=p,purple=u,red=e,white=w,yellow=y

- stalk-shape: enlarging=e,tapering=t

- stalk-root: bulbous=b,club=c,cup=u,equal=e,rhizomorphs=z,rooted=r,missing=?

- stalk-surface-above-ring: fibrous=f,scaly=y,silky=k,smooth=s

- stalk-surface-below-ring: fibrous=f,scaly=y,silky=k,smooth=s

- stalk-color-above-ring: brown=n,buff=b,cinnamon=c,gray=g,orange=o,pink=p,red=e,white=w,yellow=y

- stalk-color-below-ring: brown=n,buff=b,cinnamon=c,gray=g,orange=o,pink=p,red=e,white=w,yellow=y

- veil-type: partial=p,universal=u

- veil-color: brown=n,orange=o,white=w,yellow=y

- ring-number: none=n,one=o,two=t

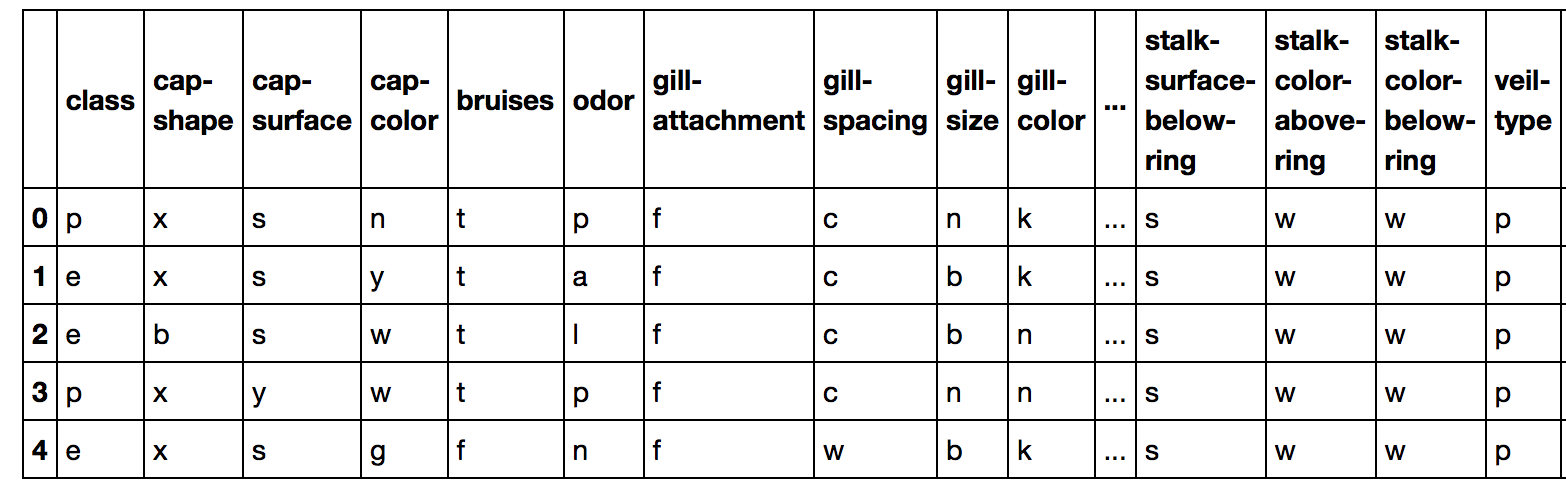
- ring-type: cobwebby=c,evanescent=e,flaring=f,large=l,none=n,pendant=p,sheathing=s,zone=z

- spore-print-color: black=k,brown=n,buff=b,chocolate=h,green=r,orange=o,purple=u,white=w,yellow=y

- population: abundant=a,clustered=c,numerous=n,scattered=s,several=v,solitary=y

- habitat: grasses=g,leaves=l,meadows=m,paths=p,urban=u,waste=w,woods=d

A sample data:



The dataset itself is well balanced between two type labels, with an edible-to-poisonous ratio of 1.07.

(balance chart)

**Dataset Abnormalities**

In the dataset, there are no NULL values. After an in-depth exploration, the feature “veil-type” contains just one unique value “p”, contributing little to the classification.

The feature “stalk-root” contains 2480 missing values filled as “?”. Considering the whole dataset only has 8124 values, I decide to keep rows with this type of missing values.

**### Exploratory Visualization**

In this section, you will need to provide some form of visualization that summarizes or extracts a relevant characteristic or feature about the data. The visualization should adequately support the data being used. Discuss why this visualization was chosen and how it is relevant. Questions to ask yourself when writing this section:

- \_Have you visualized a relevant characteristic or feature about the dataset or input data?\_

- \_Is the visualization thoroughly analyzed and discussed?\_

- \_If a plot is provided, are the axes, title, and datum clearly defined?\_

**Distribution**

Regarding each type feature, a joint plot below shows the data distribution by classes regarding each feature. The data points are generally equally distributed between two classes, a little covariance regarding specific attributes. Some features draw special attention for unbalanced distribution of certain attributes, indicating differences between edible and toxic mushrooms. For the “Oder”, edible mushrooms are mostly no special smell, while poisonous ones have foul smelling. For the “spore-print-color”, edible ones are significant in “black”, but the “poisonous” one are with high quantity of “buff” and “chocolate”.

(Distribution plot)

**Correlation**

Those features could be further classified in subgroups like “cap”, “odor”, “gill”, “stalk”, “veil”, “ring”, “spore”, or by attributes, “color”, “shape”, “surface”, ect. As features in the same group might be correlated, a heatmap was generated regarding the covariance between multiple variables (21 features in total after “veil-shape” was removed).

According to the map, those deep red or deep green blocks indicate high covariance. There are around 15 deep color blocks showing the correlation between features. For example, “veil-color” vs “gill-attachment”, “odor” vs “ring-type”. Those specific correlations would be explored in future dimension reduction studies.

(heatmap)

**### Algorithms and Techniques**

In this section, you will need to discuss the algorithms and techniques you intend to use for solving the problem. You should justify the use of each one based on the characteristics of the problem and the problem domain. Questions to ask yourself when writing this section:

- \_Are the algorithms you will use, including any default variables/parameters in the project clearly defined?\_

- \_Are the techniques to be used thoroughly discussed and justified?\_

- \_Is it made clear how the input data or datasets will be handled by the algorithms and techniques chosen?\_

Three classic machine learning models are applied and compared for full scale dataset.

### Benchmark

In this section, you will need to provide a clearly defined benchmark result or threshold for comparing across performances obtained by your solution. The reasoning behind the benchmark (in the case where it is not an established result) should be discussed. Questions to ask yourself when writing this section:

- \_Has some result or value been provided that acts as a benchmark for measuring performance?\_

- \_Is it clear how this result or value was obtained (whether by data or by hypothesis)?\_

This dataset is available on Kaggle, there are already different

## III. Methodology

\_(approx. 3-5 pages)\_

### Data Preprocessing

In this section, all of your preprocessing steps will need to be clearly documented, if any were necessary. From the previous section, any of the abnormalities or characteristics that you identified about the dataset will be addressed and corrected here. Questions to ask yourself when writing this section:

- \_If the algorithms chosen require preprocessing steps like feature selection or feature transformations, have they been properly documented?\_

- \_Based on the \*\*Data Exploration\*\* section, if there were abnormalities or characteristics that needed to be addressed, have they been properly corrected?\_

- \_If no preprocessing is needed, has it been made clear why?\_

### Implementation

In this section, the process for which metrics, algorithms, and techniques that you implemented for the given data will need to be clearly documented. It should be abundantly clear how the implementation was carried out, and discussion should be made regarding any complications that occurred during this process. Questions to ask yourself when writing this section:

- \_Is it made clear how the algorithms and techniques were implemented with the given datasets or input data?\_

- \_Were there any complications with the original metrics or techniques that required changing prior to acquiring a solution?\_

- \_Was there any part of the coding process (e.g., writing complicated functions) that should be documented?\_

### Refinement

In this section, you will need to discuss the process of improvement you made upon the algorithms and techniques you used in your implementation. For example, adjusting parameters for certain models to acquire improved solutions would fall under the refinement category. Your initial and final solutions should be reported, as well as any significant intermediate results as necessary. Questions to ask yourself when writing this section:

- \_Has an initial solution been found and clearly reported?\_

- \_Is the process of improvement clearly documented, such as what techniques were used?\_

- \_Are intermediate and final solutions clearly reported as the process is improved?\_

## IV. Results

\_(approx. 2-3 pages)\_

### Model Evaluation and Validation

In this section, the final model and any supporting qualities should be evaluated in detail. It should be clear how the final model was derived and why this model was chosen. In addition, some type of analysis should be used to validate the robustness of this model and its solution, such as manipulating the input data or environment to see how the model’s solution is affected (this is called sensitivity analysis). Questions to ask yourself when writing this section:

- \_Is the final model reasonable and aligning with solution expectations? Are the final parameters of the model appropriate?\_

- \_Has the final model been tested with various inputs to evaluate whether the model generalizes well to unseen data?\_

- \_Is the model robust enough for the problem? Do small perturbations (changes) in training data or the input space greatly affect the results?\_

- \_Can results found from the model be trusted?\_

### Justification

In this section, your model’s final solution and its results should be compared to the benchmark you established earlier in the project using some type of statistical analysis. You should also justify whether these results and the solution are significant enough to have solved the problem posed in the project. Questions to ask yourself when writing this section:

- \_Are the final results found stronger than the benchmark result reported earlier?\_

- \_Have you thoroughly analyzed and discussed the final solution?\_

- \_Is the final solution significant enough to have solved the problem?\_

## V. Conclusion

\_(approx. 1-2 pages)\_

### Free-Form Visualization

In this section, you will need to provide some form of visualization that emphasizes an important quality about the project. It is much more free-form, but should reasonably support a significant result or characteristic about the problem that you want to discuss. Questions to ask yourself when writing this section:

- \_Have you visualized a relevant or important quality about the problem, dataset, input data, or results?\_

- \_Is the visualization thoroughly analyzed and discussed?\_

- \_If a plot is provided, are the axes, title, and datum clearly defined?\_

### Reflection

In this section, you will summarize the entire end-to-end problem solution and discuss one or two particular aspects of the project you found interesting or difficult. You are expected to reflect on the project as a whole to show that you have a firm understanding of the entire process employed in your work. Questions to ask yourself when writing this section:

- \_Have you thoroughly summarized the entire process you used for this project?\_

- \_Were there any interesting aspects of the project?\_

- \_Were there any difficult aspects of the project?\_

- \_Does the final model and solution fit your expectations for the problem, and should it be used in a general setting to solve these types of problems?\_

### Improvement

In this section, you will need to provide discussion as to how one aspect of the implementation you designed could be improved. As an example, consider ways your implementation can be made more general, and what would need to be modified. You do not need to make this improvement, but the potential solutions resulting from these changes are considered and compared/contrasted to your current solution. Questions to ask yourself when writing this section:

- \_Are there further improvements that could be made on the algorithms or techniques you used in this project?\_

- \_Were there algorithms or techniques you researched that you did not know how to implement, but would consider using if you knew how?\_

- \_If you used your final solution as the new benchmark, do you think an even better solution exists?\_

-----------

\*\*Before submitting, ask yourself. . .\*\*

- Does the project report you’ve written follow a well-organized structure similar to that of the project template?

- Is each section (particularly \*\*Analysis\*\* and \*\*Methodology\*\*) written in a clear, concise and specific fashion? Are there any ambiguous terms or phrases that need clarification?

- Would the intended audience of your project be able to understand your analysis, methods, and results?

- Have you properly proof-read your project report to assure there are minimal grammatical and spelling mistakes?

- Are all the resources used for this project correctly cited and referenced?

- Is the code that implements your solution easily readable and properly commented?

- Does the code execute without error and produce results similar to those reported?

References:

1. Kaggle: <https://www.kaggle.com/acehanks/d/uciml/mushroom-classification/mushroom-poisonous>
2. Kaggle:
3. Github: <https://github.com/linzhulz/student_intervention/blob/master/student_intervention.ipynb>