Identification of edible mushrooms

Jose Maldonado

Abstract

This project analyses a mushrooms dataset, using machine learning methodologies (in particular, decision tree and random forest classification algorithms), to determine the best way to differentiate edible from poisonous mushrooms and to check how effective are mushroom hunting tips.

Motivation

Mushroom hunting is an outdoor activity which is becoming more and more popular in many parts or the world, but it can be a dangerous activity since deadly mushrooms sometimes look like perfectly edible mushrooms, and can collected and eaten by mistake.

This project will analyze a dataset of mushrooms in order to determine the best way to differentiate edible from poisonous mushrooms.

Dataset(s)

This project uses the “Mushroom Classification” dataset, downloaded from Kaggle, which includes descriptions of hypothetical samples corresponding to different species of gilled mushrooms, each one identified as definitely edible, definitely poisonous, or of unknown edibility and not recommended (this latter class was combined with the poisonous one).

Data Preparation and Cleaning

No preparation nor cleaning was need, since the dataset is correctly organized in a .csv file and the meaning of each column is described on the overview (on the Kaggle site).

Research Question(s)

- It’s possible to differentiate edible from poisonous mushrooms based solely on their physical aspect and smell?
- Which feature is the most indicative of a poisonous mushroom?
- How effective are common mushroom hunting tips?

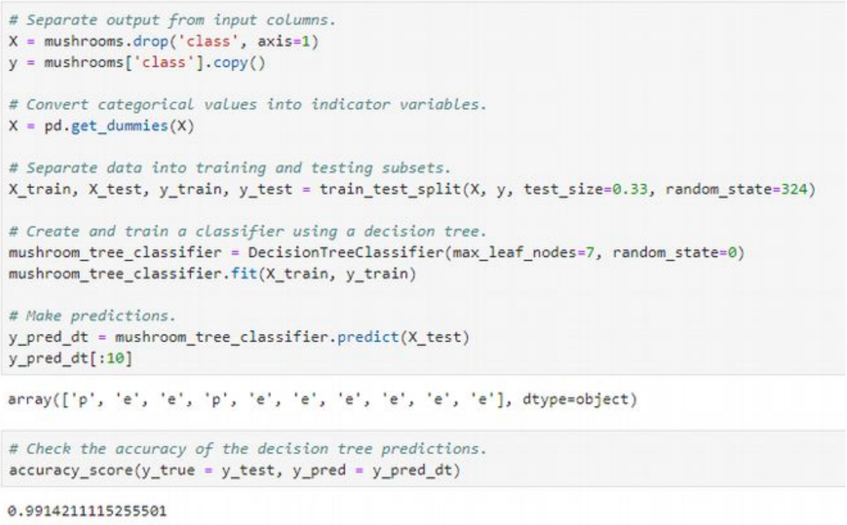
Methods

Initial analysis was done using a Decision Tree classification algorithm, due to his flexibility, ease to use, resilience and comprehensive nature.

Later, due to indications of overfitting, a Random Forest classification algorithm was used, since it mitigates the risk of overfitting, with the only downside of a higher computational cost (with was no problem in this case, since the dataset wasn’t very large).

Findings – Classification (1/2)

The first attempt to build a model for differentiate edible from poisonous mushrooms was done using a Decision Tree algorithm. The result was a model with an accuracy of 99%.



```
# Separate output from input columns.
X = mushrooms.drop('class', axis=1)
y = mushrooms['class'].copy()

# Convert categorical values into indicator variables.
X = pd.get_dummies(X)

# Separate data into training and testing subsets.
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.33, random_state=324)

# Create and train a classifier using a decision tree.
mushroom_tree_classifier = DecisionTreeClassifier(max_leaf_nodes=7, random_state=0)
mushroom_tree_classifier.fit(X_train, y_train)

# Make predictions.
y_pred_dt = mushroom_tree_classifier.predict(X_test)
y_pred_dt[:10]
```

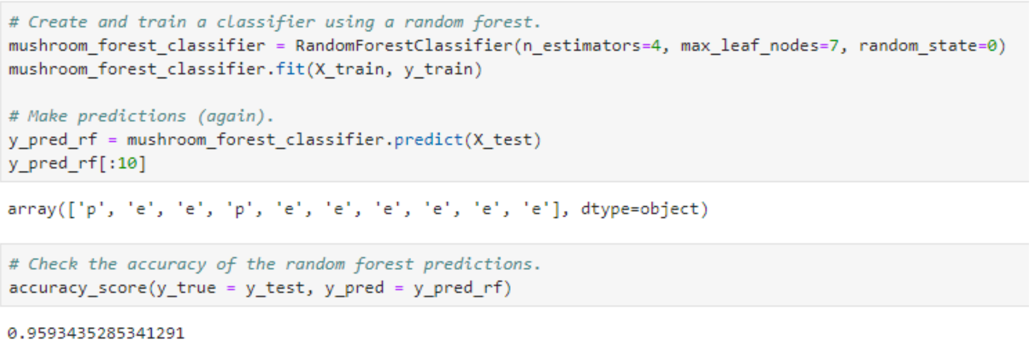
```
array(['p', 'e', 'e', 'p', 'e', 'e', 'e', 'e', 'e', 'e'], dtype=object)
```

```
# Check the accuracy of the decision tree predictions.
accuracy_score(y_true = y_test, y_pred = y_pred_dt)
```

```
0.9914211115255501
```

Findings – Classification (2/2)

In order to verify that this high accuracy value was not produced by an overfitting error from the Decision Tree algorithm, a second model was build, this time using a Random Forest algorithm (since this algorithm tends to limit overfitting).
This new model had an accuracy of 95%, discarding suspects of overfitting, and confirming the feasibility for differentiate edible from poisonous mushrooms based on their aspect and smell.



```
# Create and train a classifier using a random forest.
mushroom_forest_classifier = RandomForestClassifier(n_estimators=4, max_leaf_nodes=7, random_state=0)
mushroom_forest_classifier.fit(X_train, y_train)

# Make predictions (again).
y_pred_rf = mushroom_forest_classifier.predict(X_test)
y_pred_rf[:10]
```

```
array(['p', 'e', 'e', 'p', 'e', 'e', 'e', 'e', 'e', 'e'], dtype=object)
```

```
# Check the accuracy of the random forest predictions.
accuracy_score(y_true = y_test, y_pred = y_pred_rf)
```

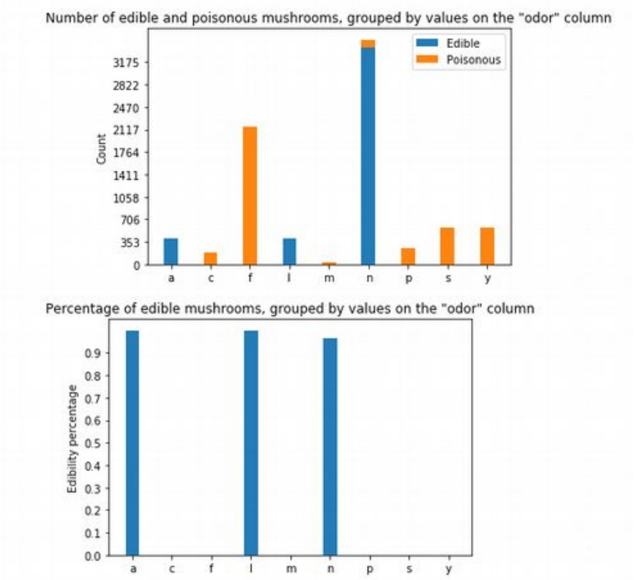
```
0.9593435285341291
```

Findings – Indicative features (1/4)

In order to identify which features are the most useful for identify if a mushrooms are edible or poisonous, a pair of plot bars (one indicating the number of edible and poisonous mushrooms for each value on a feature, and another indicating the percentage of edible mushrooms for each value) were draw for each one of the features in the dataset.

Findings – Indicative features (2/4)

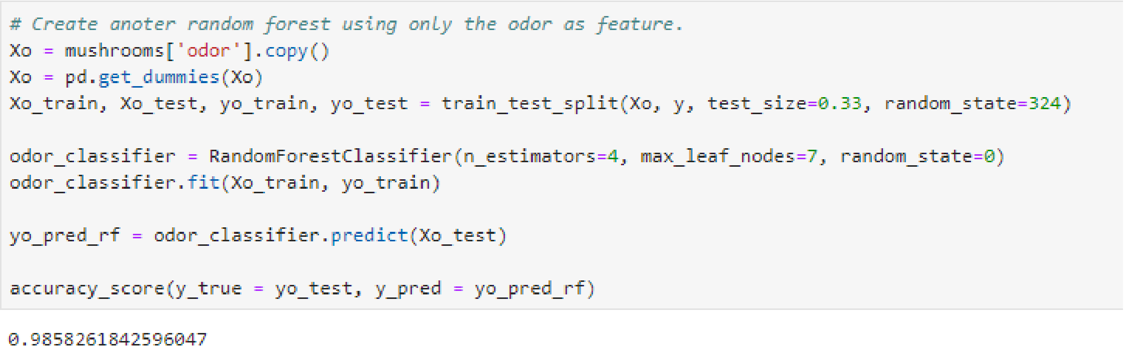
From all the features, the “odor” feature was found to be the best indicator, since edible and poisonous mushrooms are separated by the values of this feature:


Number of edible and poisonous mushrooms, grouped by values on the "odor" column

Percentage of edible mushrooms, grouped by values on the "odor" column

Findings – Indicative features (3/4)

In order to verify this finding, a third model mas build, this time using the “odor” as the only feature, and the result was an accuracy of 98%.
This finding makes completely sense with biology, since most mammals evolved their noses to being able to detect edibles by the smell.

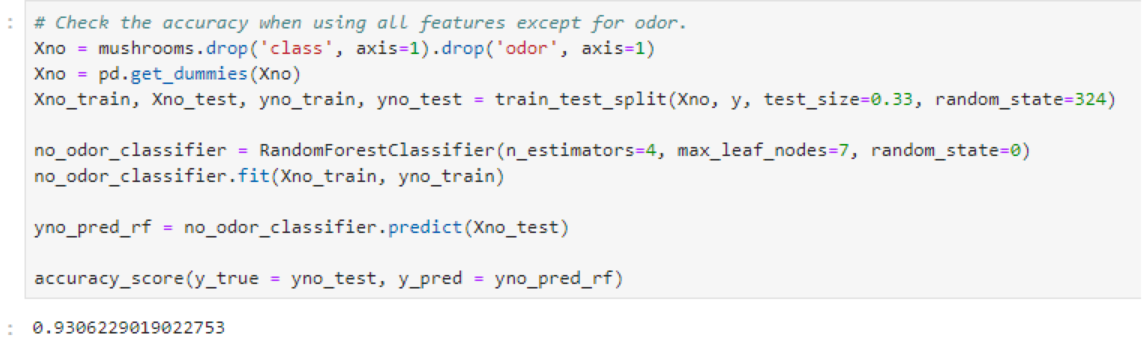


```
# Create anoter random forest using only the odor as feature.
Xo = mushrooms['odor'].copy()
Xo = pd.get_dummies(Xo)
Xo_train, Xo_test, yo_train, yo_test = train_test_split(Xo, y, test_size=0.33, random_state=324)

odor_classifier = RandomForestClassifier(n_estimators=4, max_leaf_nodes=7, random_state=0)
odor_classifier.fit(Xo_train, yo_train)

yo_pred_rf = odor_classifier.predict(Xo_test)

accuracy_score(y_true = yo_test, y_pred = yo_pred_rf)
```

```
0.9858261842596047
```

Findings – Indicative features (4/4)

However, the human nose is not very sensitive, so a fourth model was created, removing the “odor” column, in order to check if it’s possible to classify mushrooms using only their aspect. The result was an accuracy of 93% (not as good as the odor’s accuracy, but still pretty effective).



```
# Check the accuracy when using all features except for odor.
Xno = mushrooms.drop('class', axis=1).drop('odor', axis=1)
Xno = pd.get_dummies(Xno)
Xno_train, Xno_test, yno_train, yno_test = train_test_split(Xno, y, test_size=0.33, random_state=324)

no_odor_classifier = RandomForestClassifier(n_estimators=4, max_leaf_nodes=7, random_state=0)
no_odor_classifier.fit(Xno_train, yno_train)

yno_pred_rf = no_odor_classifier.predict(Xno_test)

accuracy_score(y_true = yno_test, y_pred = yno_pred_rf)
```

```
0.9306229019022753
```

Findings – Hunting tips (1/3)

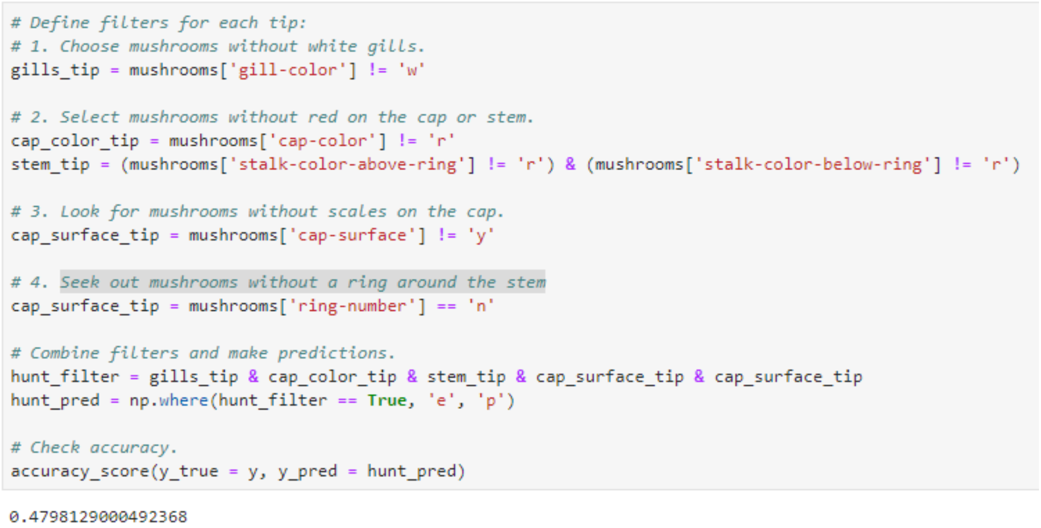
Common hunting tips are:

- Choose mushrooms without white gills.
- Select mushrooms without red on the cap or stem.
- Look for mushrooms without scales on the cap.
- Seek out mushrooms without a ring around the stem.

In order to verify the accuracy of these tips, they were applied to the dataset, and then the accuracy was calculated.

Findings – Hunting tips (2/3)

Surprisingly, the accuracy was only 47%.

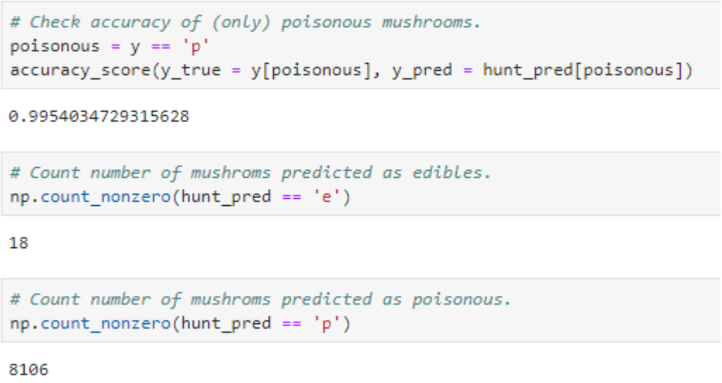


```
# Define filters for each tip:
# 1. Choose mushrooms without white gills.
gills_tip = mushrooms['gill-color'] != 'w'

# 2. Select mushrooms without red on the cap or stem.
cap_color_tip = mushrooms['cap-color'] != 'r'
stem_tip = (mushrooms['stalk-color-above-ring'] != 'r') & (mushrooms['stalk-color-below-ring'] != 'r')

# 3. Look for mushrooms without scales on the cap.
cap_surface_tip = mushrooms['cap-surface'] != 'y'

# 4. Seek out mushrooms without a ring around the stem
cap_surface_tip = mushrooms['ring-number'] == 'n'

# Combine filters and make predictions.
hunt_filter = gills_tip & cap_color_tip & stem_tip & cap_surface_tip & cap_surface_tip
hunt_pred = np.where(hunt_filter == True, 'e', 'p')

# Check accuracy.
accuracy_score(y_true = y, y_pred = hunt_pred)
```

```
0.4798129000492368
```

Findings – Hunting tips (3/3)

However, in a further analysis, it was found that the accuracy for detecting poisonous mushrooms was 99%. Which indicates that these particular group of tips, tend to classify most mushrooms as poisonous, in order to prevent classifying poisonous mushrooms as edibles.



```
# Check accuracy of (only) poisonous mushrooms.
poisonous = y == 'p'
accuracy_score(y_true = y[poisonous], y_pred = hunt_pred[poisonous])
```

```
0.9954034729315628
```

```
# Count number of mushroms predicted as edibles.
np.count_nonzero(hunt_pred == 'e')
```

```
18
```

```
# Count number of mushroms predicted as poisonous.
np.count_nonzero(hunt_pred == 'p')
```

```
8106
```

Limitations

The dataset includes only mushrooms from the Agaricus and Lepiota families, and some data was generated based on the book “The Audubon Society Field Guide to North American Mushrooms”. As consequence the findings may no be applicable to mushrooms from other families or to mushrooms from other regions.

Conclusions

- It’s perfectly feasible to differentiate edible from poisonous mushrooms based only in their aspect and smell.

- In most cases, poisonous mushrooms can be identified based only by their smell (although it’s also possible to identify them based only in their physical aspect).

- Common mushroom hunting tips, in order to avoid fake positives (mark poisonous mushrooms as edibles), produce a lot number of fake negatives (mark edibles mushrooms as poisonous).

Acknowledgements

The dataset was obtained from Kaggle (https://www.kaggle.com/datasets).

No feedback was received from neither colleagues nor friends.

References

- https://www.kaggle.com/uciml/mushroom-classification
- https://scikit-learn.org/stable/index.html
- https://www.edx.org/course/python-for-data-science-0