CONTENTS

student’s names:

Shorok abu nimer – 208103556

Ragad mograby – 207608407

Background:

In this research, we examined different types of mushrooms to see if they are poisonous or edible. for this purpose, we used two files the first one was Primary data.

Relevant information for the Primary data:

This dataset includes 173 species of mushrooms with caps from various families and one entry for each species. Each species is identified as definitely edible, definitely poisonous, or of unknown edibility and not recommended (the latter class was combined with the poisonous class). Of the 20 variables, 17 are nominal and 3 are metrical. The values of each nominal variable are a set of possible values and for the metrical variables a range of possible values.

the second one was Secondary data:

Relevant information for the Secondary data:

This dataset includes 61069 hypothetical mushrooms with caps based on 173 species (353 mushrooms per species). Each mushroom is identified as definitely edible, definitely poisonous, or of unknown edibility and not recommended (the latter class was combined with the poisonous class). Of the 20 variables, 17 are nominal and 3 are metrical.

Question:

The question we answered is:

Is a particular mushroom According to its features poisonous or edible?

We didn't change the question, we just reformulated it in a better way, because from what we found in the research, it is possible to know if the mushroom is poisonous or edible from its features.

Methods:

<u>Packages:</u>

Pandas

Seaborn

numpy

matplotlib.pyplot

sklearn

The steps we have taken to answer the question:

1. exploring the data (exploratory data analysis):

a. in the file Explore_primary.py, we wrote primary_print_more_info function to print information about the first csv file (primary data).

head, shape, describe, info, is_null, index of not null col, name of not null col.

We have removed from the database all the columns containing null values.

We are left with 12 relevant columns.

Create new data frame without the columns that have null values

b. In the file exploratory_data.py, , we wrote print_more_info function to print information about the second csv file (secondary data).

head, shape, describe, info, is_null, index of not null col, name of not null col.

Create new data frame without the columns that have null values

Put every column in np array.

2. Look for correlations between different variables and the target variable.

We wrote plot functions to draw the columns and find the correlation between the features and the target.

a. in the file Explore_primary.py

The function plot_ptarget is to show the poison and edible mushrooms by index (the index is the 173 species).

The function get_index_pri(str) takes string = (name of specie) and returns the index of the name(str).

b. In the file exploratory_data.py

test_part function

:param target: the target of the data set (class column: p/e)
:param arr1: x
:param arr2: y
:param n1: first index or default = 0
:param n2: last index or default = len = 61069
:return: drawing slice of arr1 and arr2 with coloring the poison vals in red and the edible vals in blue

plot_test_part function

:return: plot all the columns(11 column) using test part function

We searched the internet for information about mushrooms and how to identify poisonous mushrooms.

We found that there are features of poisonous mushrooms, so we tried to plot these features to find a relationship between the feature and the poisonous mushrooms.

The features we found:

has ring = t
gill color = white
cap color = red / orange = e / o
stem color = red / white = e / w

But through the drawing, we did not find any relationship that helps us to know the poisonous mushrooms from the edible mushrooms, and it became clear to us that this differs from one type of mushroom to another, so we decided to take specific types of mushrooms and find information about them and then draw them to find the relationship between the features and their edibility.

From these drawings, we also found a great similarity in the features between two types of mushrooms, one of which is poisonous and one of them is edible.

Therefore, we knew that we would not be able to reduce the remaining features (12 columns) because each one of them will help us to find the type of mushroom and finding if its edible or poisonous, with the help of the Random Forest algorithm.

3. The Random Forest Algorithm

We used the Random Forest because One of the most important features of the Random Forest Algorithm is that it can handle the data set categorical variables as in the case of classification and we chassed to work with the classification so we decided that this algorithm will help us to solve our problem. It builds decision trees on different samples and takes their majority vote for classification

They are the steps of the Random Forest that we worked in:

Step 1: we worked and arranged at our data set.

Step 2: Individual decision trees are constructed for each sample.

Step 3: Each decision tree will generate an output.

Step 4: Final output is considered based on Majority Voting or Averaging for Classification.

a. We took the data
b. Convert the char values to int
c. Split the data

The first way we split it to 70% training and 30% test

Print the shape of train and test data

used RandomForestClassifier (as we learned in the class lec12)

predict y => rf_pred = rf.predict(X_test)

and compared between rf_pred and y_test by plotting them.

Also we print the accuracy that shows us that the algorithm succeed to predict the target (rf_pred).

We used another way to split the data 50-30-20

50% training

30% testing

20% validation

And print the results

Results:

We will show the results according to the steps we have taken to answer the question:

1. exploring the data (exploratory data analysis):
a. Explore_primary.py

primary_print_more_info function

```
primary more info:
head
           family              name class  ... Spore-print-color habitat     season
0  Amanita Family         Fly Agaric     p  ...               NaN     [d]  [u, a, w]
1  Amanita Family        Panther Cap     p  ...               NaN     [d]     [u, a]
2  Amanita Family  False Panther Cap     p  ...               NaN     [d]     [u, a]
3  Amanita Family        The Blusher     e  ...               NaN     [d]     [u, a]
4  Amanita Family          Death Cap     p  ...               NaN     [d]     [u, a]

[5 rows x 23 columns]
shape
(173, 23)
descripe:
                   family        name class  ... Spore-print-color habitat  season
count                 173         173   173  ...                18     173     173
unique                 23         173     2  ...                 8      21      10
top     Tricholoma Family  Fly Agaric     p  ...               [k]     [d]  [u, a]
freq                   43           1    96  ...                 5     104     106

[4 rows x 23 columns]
```

```
info
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 173 entries, 0 to 172
Data columns (total 23 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   family                173 non-null    object
 1   name                  173 non-null    object
 2   class                 173 non-null    object
 3   cap-diameter          173 non-null    object
 4   cap-shape             173 non-null    object
 5   Cap-surface           133 non-null    object
 6   cap-color             173 non-null    object
 7   does-bruise-or-bleed  173 non-null    object
 8   gill-attachment       145 non-null    object
 9   gill-spacing          102 non-null    object
 10  gill-color            173 non-null    object
 11  stem-height           173 non-null    object

 11  stem-height           173 non-null    object
 12  stem-width            173 non-null    object
 13  stem-root             27 non-null     object
 14  stem-surface          65 non-null     object
 15  stem-color            173 non-null    object
 16  veil-type             9 non-null      object
 17  veil-color            21 non-null     object
 18  has-ring              173 non-null    object
 19  ring-type             166 non-null    object
 20  Spore-print-color     18 non-null     object
 21  habitat               173 non-null    object
 22  season                173 non-null    object
dtypes: object(23)
memory usage: 31.2+ KB
```

```
is_null
family                   0
name                     0
class                    0
cap-diameter             0
cap-shape                0
Cap-surface             40
cap-color                0
does-bruise-or-bleed     0
gill-attachment         28
gill-spacing            71
gill-color               0
stem-height              0
stem-width               0
stem-root              146
stem-surface           108
stem-color               0
veil-type              164
veil-color             152
has-ring                 0
ring-type                7
Spore-print-color      155
habitat                  0
season                   0
dtype: int64
```

index of not null col

[0, 1, 2, 3, 4, 6, 7, 10, 11, 12, 15, 18, 21, 22]

name of not null col:

['family', 'name', 'class', 'cap-diameter', 'cap-shape', 'cap-color', 'does-bruise-or-bleed', 'gill-color', 'stem-height', 'stem-width', 'stem-color', 'has-ring', 'habitat', 'season']

b. exploratory_data.py

```
secondary data info
head
  class  cap-diameter cap-shape  ... spore-print-color habitat season
0     p         15.26         x  ...               NaN       d      w
1     p         16.60         x  ...               NaN       d      u
2     p         14.07         x  ...               NaN       d      w
3     p         14.17         f  ...               NaN       d      w
4     p         14.64         x  ...               NaN       d      w

[5 rows x 21 columns]
shape
(61069, 21)
descripe:
       cap-diameter   stem-height    stem-width
count  61069.000000  61069.000000  61069.000000
mean       6.733854      6.581538     12.149410
std        5.264845      3.370017     10.035955
min        0.380000      0.000000      0.000000
25%        3.480000      4.640000      5.210000
50%        5.860000      5.950000     10.190000
75%        8.540000      7.740000     16.570000
max       62.340000     33.920000    103.910000
```

```
info
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 61069 entries, 0 to 61068
Data columns (total 21 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   class                 61069 non-null  object
 1   cap-diameter          61069 non-null  float64
 2   cap-shape             61069 non-null  object
 3   cap-surface           46949 non-null  object
 4   cap-color             61069 non-null  object
 5   does-bruise-or-bleed  61069 non-null  object
 6   gill-attachment       51185 non-null  object
 7   gill-spacing          36006 non-null  object
 8   gill-color            61069 non-null  object
 9   stem-height           61069 non-null  float64
 10  stem-width            61069 non-null  float64
 11  stem-root             9531 non-null   object
 12  stem-surface          22945 non-null  object
 13  stem-color            61069 non-null  object
 14  veil-type             3177 non-null   object
 15  veil-color            7413 non-null   object
 16  has-ring              61069 non-null  object
 17  ring-type             58598 non-null  object
 18  spore-print-color     6354 non-null   object
```

```
 19  habitat               61069 non-null  object
 20  season                61069 non-null  object
dtypes: float64(3), object(18)
memory usage: 9.8+ MB
```

```
is_null
class                      0
cap-diameter               0
cap-shape                  0
cap-surface            14120
cap-color                  0
does-bruise-or-bleed       0
gill-attachment         9884
gill-spacing           25063
gill-color                 0
stem-height                0
stem-width                 0
stem-root              51538
stem-surface           38124
stem-color                 0
veil-type              57892
veil-color             53656
has-ring                   0
ring-type               2471
spore-print-color      54715
habitat                    0
season                     0
dtype: int64
```

index of not null col

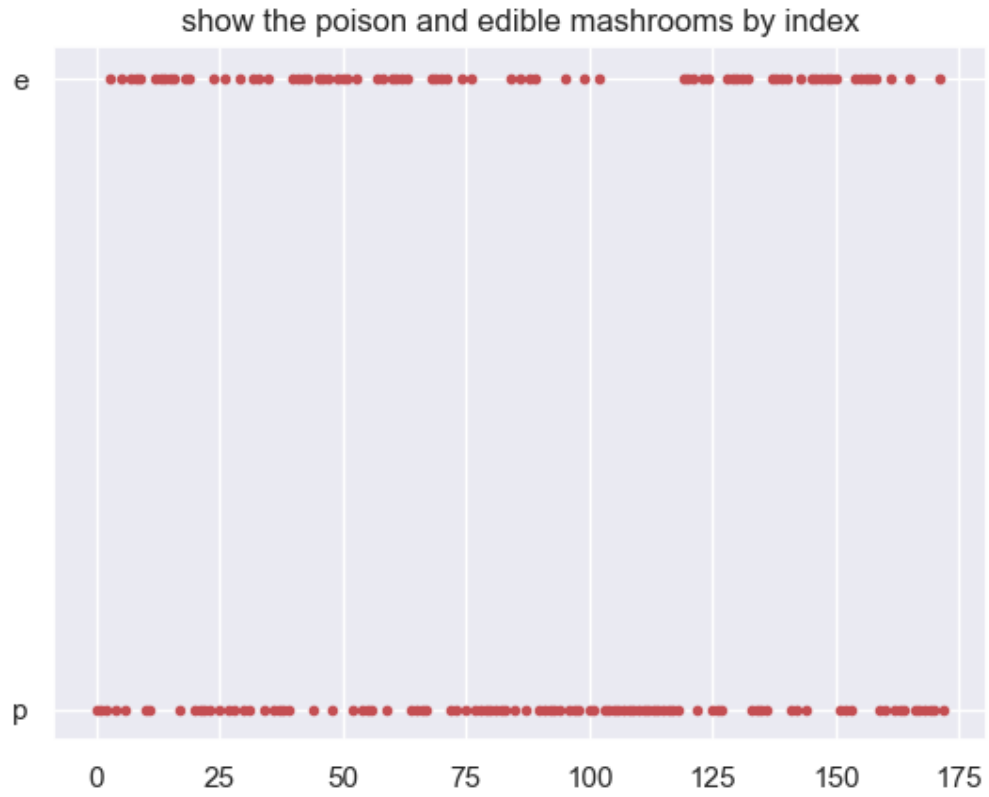
[0, 1, 2, 4, 5, 8, 9, 10, 13, 16, 19, 20]

name of not null col:

['class', 'cap-diameter', 'cap-shape', 'cap-color', 'does-bruise-or-bleed', 'gill-color', 'stem-height', 'stem-width', 'stem-color', 'has-ring', 'habitat', 'season']

2. Look for correlations between different variables and the target variable.
a. Explore_primary.py

plot_ptarget function



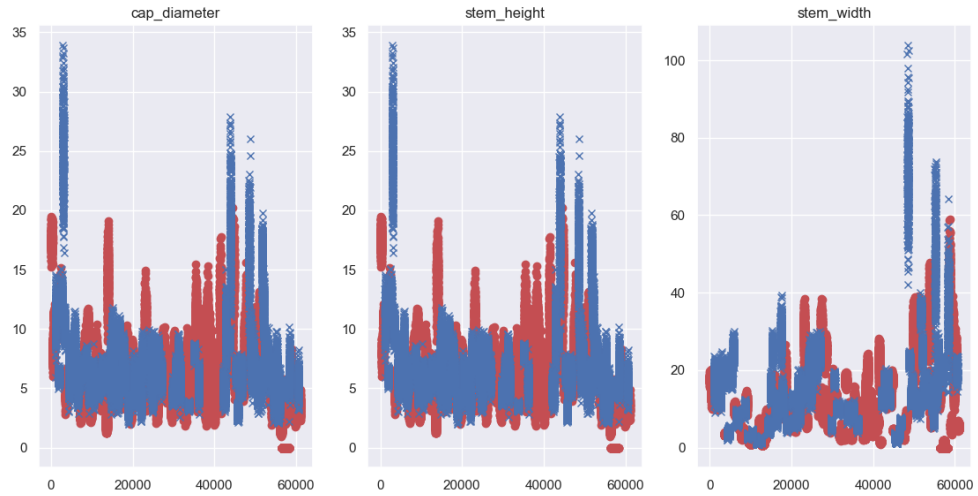
b. exploratory_data.py

A. plot_test_part function

blue = edible

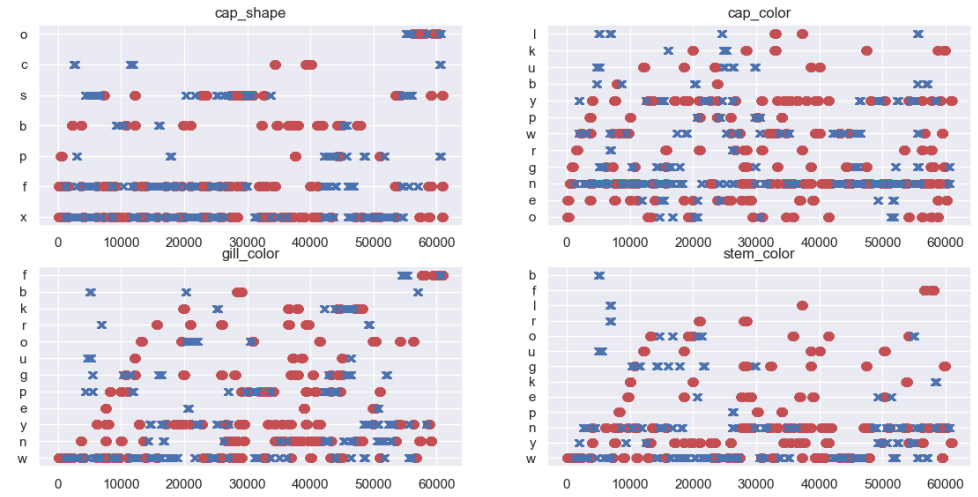
red = poisonous

cap_diameter, stem_hieght, stem_width



Cap_shape, cap_color, gill_color,stem_color

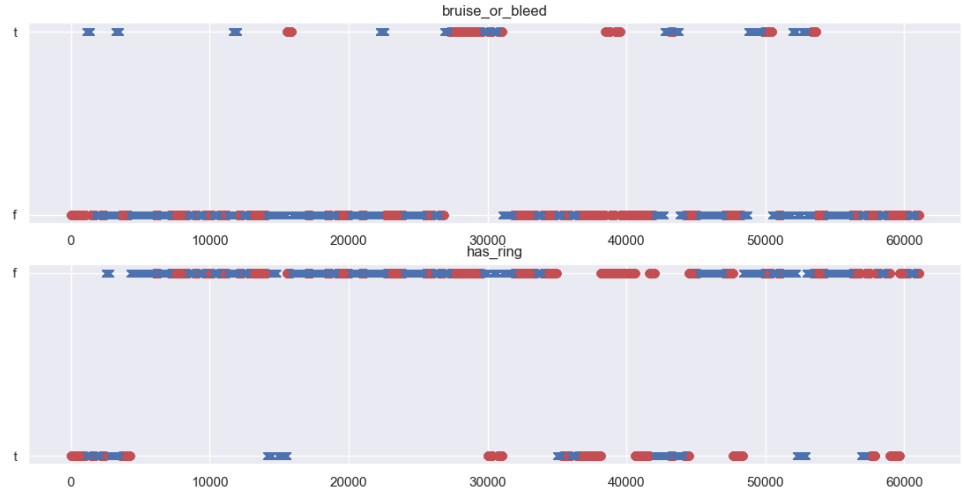
brown=n, buff=b, gray=g, green=r, pink=p, purple=u, red=e, white=w, yellow=y, blue=l, orange=o, black=k, none=f



Bruise or bleed, has ring

ring=t, none=f

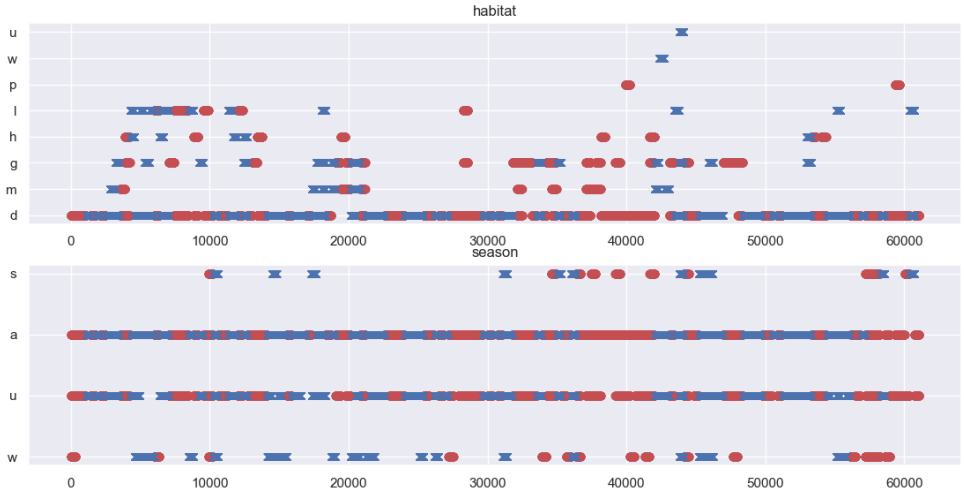
bruises-or-bleeding=t,no=f



Habitat , season

grasses=g, leaves=l, meadows=m, paths=p, heaths=h, urban=u, waste=w, woods=d

spring=s, summer=u, autumn=a, winter=w



The previous drawings did not help us much in finding a relationship between the features of the mushroom and knowing if it is poisonous or edible. Small but insufficient details can be noted, for example:

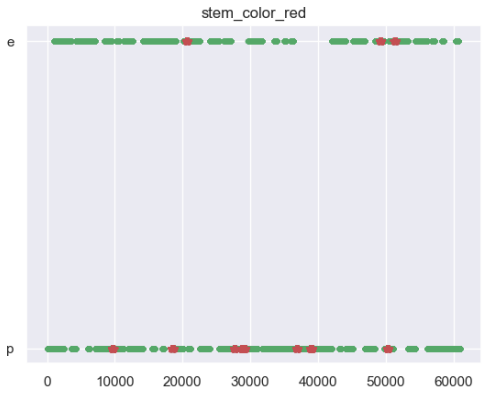
Mushrooms with cap diameter > 20 Or stem height > 20 Or stem width > 60 Is edible

Most of mushrooms with cap color = orange (o) or red(e) is poisonous

So we searched in the internet and plotted out the features we found

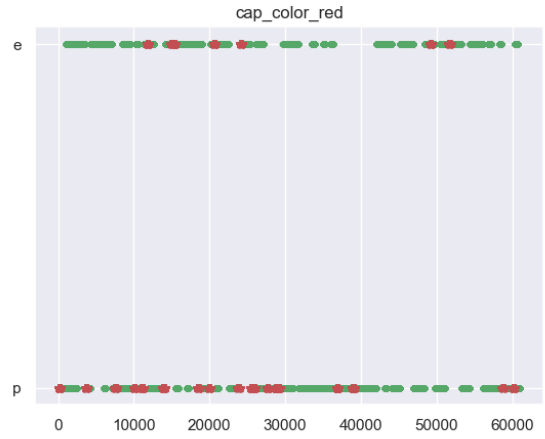
has ring = t
gill color = white
cap color = red / orange = e / o
stem color = red / white = e / w

B. plot_poisonous function



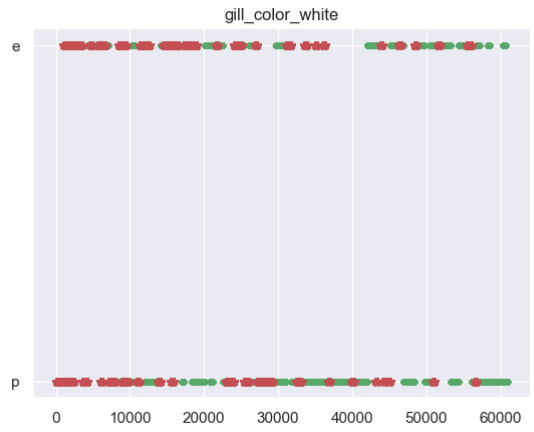
Most of the mushrooms that have red stem color are poisonous

Cap color red



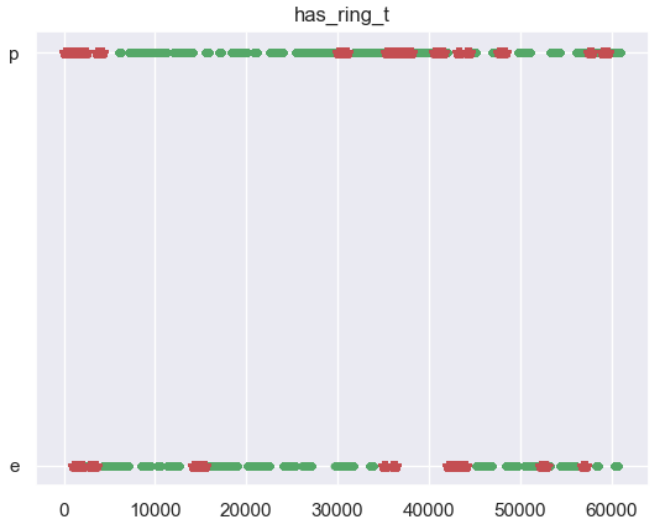
Most of the mushrooms that have red cap color are poisonous

Gill color white



The first 10000 mushrooms are poisonous, regardless of the color of the gill

Has ring = t

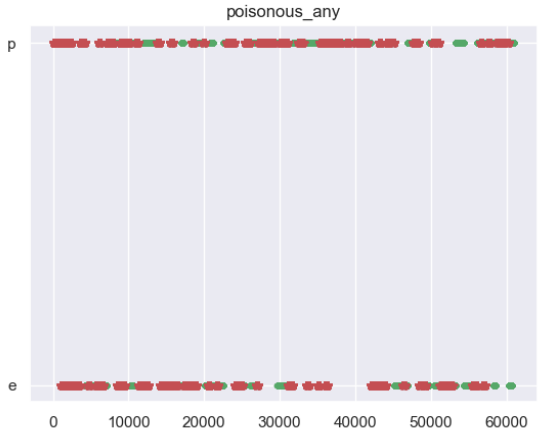


Many kinds of mushrooms that have ring are poisonous

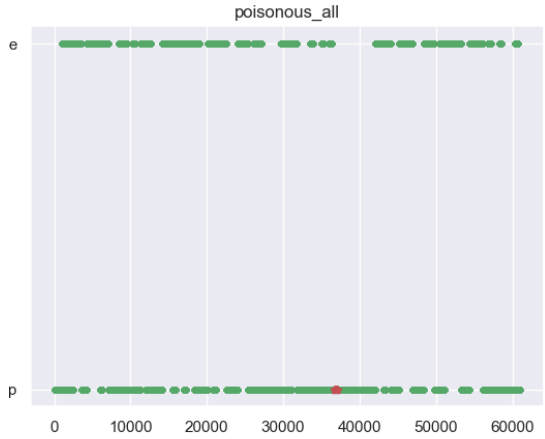
Poisonous any

The intent of this graph is that if any of the four features mentioned above are achieved, will the mushroom be poisonous or edible?

has ring = t
gill color = white
cap color = red = e
stem color = red = e

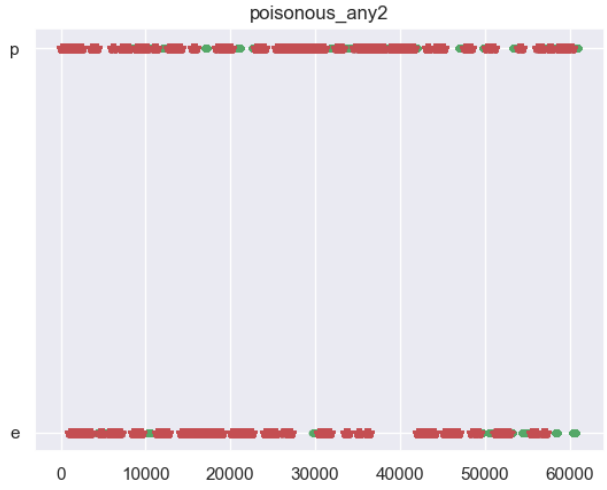


The intent of this graph is that if each of the four features mentioned above are achieved, will the mushroom be poisonous or edible?

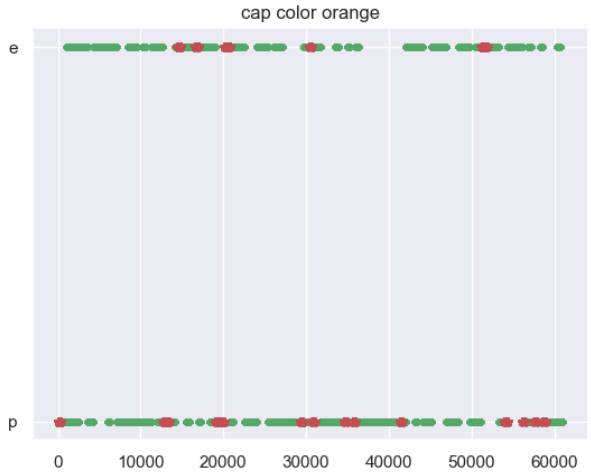


poisonous_any2

poisonous 2:
has ring = t
gill color = white
cap color = red / orange = e / o
stem color = red / white = e / w



Cap color orange



Most of them is poisonous (cap color = o)

This function made us look at basic features to check if the mushroom was poisonous or edible, but it was also not enough because even these features differ from one type of mushroom to another.

So we looked at specific types of mushrooms and saw their features.

C. plot_test_part_examples function

Fly Agaric

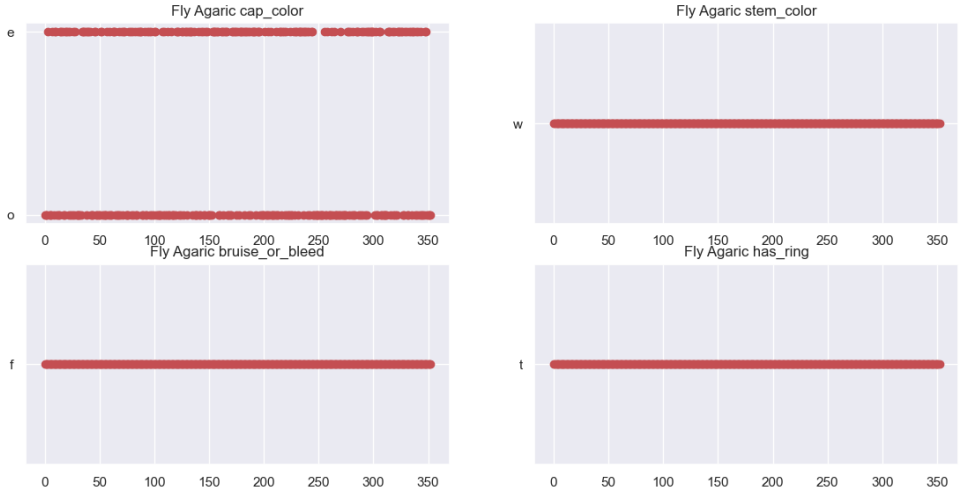
Poisonous

Cap color = orange or red

Stem color = white

Bruise or bleed = f

Has ring = t



https://www.woodlandtrust.org.uk/trees-woods-and-wildlife/fungi-and-lichens/fly-agaric/

Death Cap (Amanita phalloides)

poisonous

False Deathcap

edible

Stem color

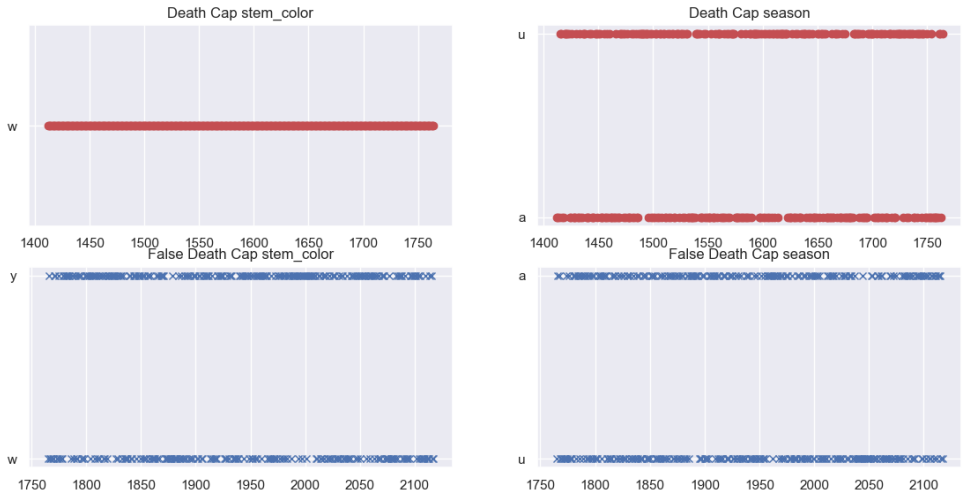
Death cap: white

False death cap: white or yellow

Season

Death cap: summer=u or autumn=a

False death cap: summer=u or autumn=a



https://www.woodlandtrust.org.uk/trees-woods-and-wildlife/fungi-and-lichens/deathcap/

https://www.wildfooduk.com/mushroom-guide/false-death-cap/

When looking at specific types of mushrooms, it can help us more in determining whether they are poisonous or edible, but we can also be mistaken because there are types of mushrooms that similar in their features, but one of them is poisonous and the other is edible.

So after looking at the files and drawing the different relationships between mushrooms, their features and types, and examining whether they are poisonous or edible in the end, we chose to proceed to the next step, which is the random forest algorithm, with its help we can get the best results to find out if the mushroom is poisonous or edible.

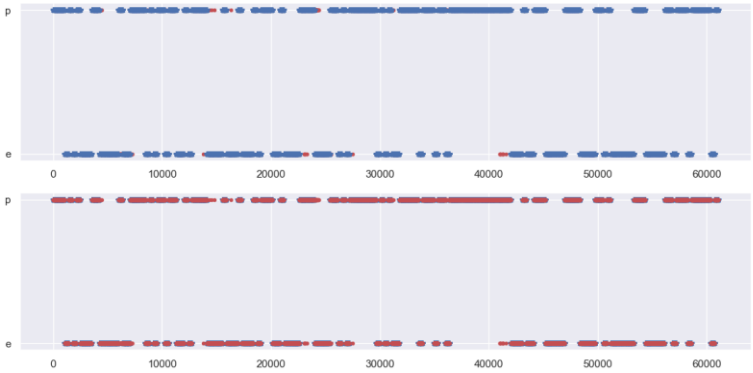
3. The Random Forest Algorithm

We have divided the information in two ways as mentioned in the previous section

a. Train = 70% , test = 30%

```
random forest
splitting the data into train and test 70-30
X_train shape =  (42748, 11)
X_test shape =  (18321, 11)
y_train shape =  (42748,)
y_test shape =  (18321,)
Random Forest accuracy: 0.994814693521096
[[ 8056    52]
 [   43 10170]]
```

We also plotted the predicted values vs the y test



The values are similar

The accuracy = 0.99481

Through the results, we find that this algorithm is successful and appropriate for this research, as the accuracy of the results is high and the expected value is very close to the original value.

b. Train = 50%, test = 30%, validation = 20%

```
another way to split the data 50-30-20
X_train2 shape =  (30534, 11)
X_test2 shape =  (21375, 11)
X_validation shape =  (9160, 11)
y_train2 shape =  (30534,)
y_test2 shape =  (21375,)
y_validation shape =  (9160,)
Random Forest accuracy: 0.9947134502923977
[[ 9425    65]
 [   48 11837]]
validation
Random Forest accuracy: 0.9959606986899563
[[3988   18]
 [  19 5135]]
```

Discussion

At the end of the research and through the results we obtained, we can say that we have found a way to know the poisonous or edible mushrooms through their features.

That is, we can collect new data about different types of mushrooms and use algorithm random forest again, and come to a good conclusion about whether the mushrooms are poisonous or edible, and this was exactly our research question, (depending on the features of the mushroom, is it poisonous or edible?).

The results show us that the algorithm was accurate and successful, as the accuracy rate was 0.99.

The new idea that came to our mind after this research is that we can work on a project that receives different forms (pictures) of mushrooms and through the (AI) this mushroom can be researched and take data about it (its features) that will help us determine if it is poisonous or edible, a project like this can That it is good and that it be used by researchers or by people who like to walk in nature and try to find mushrooms and pick them to eat (I don't know if it is legal :))

Even similar research can be done on other types of food.