Mushroom classification for edibility

A data science exercise

Dr Srikanth Lavu, 31st May 2024

Problem Statement

- Classify given gilled mushroom from the Agaricus and Lepiota Family between edible or poisonous.

Data set - brief summary

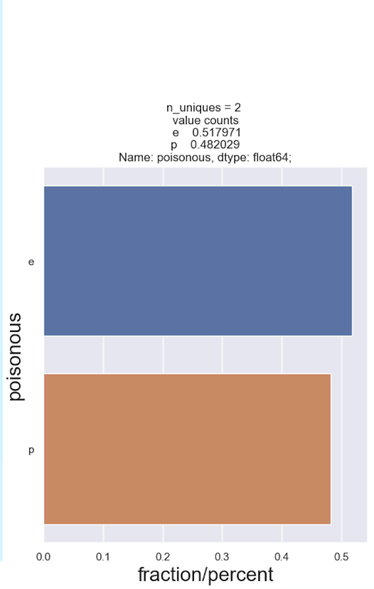
- 8124 observations provided.
  - No duplicate observations in the data.
- 22 independent variables and one dependent/target variable(‘poisonous’).
- Target variable(‘poisonous’)
  - Has two classes: edible(e), poisonous(p)
    - binary class classification problem.
  - Distribution of target variable
    - edible(‘e’) - 51.8%, poisonous(‘p’) - 48.2%
    - Dataset does not have class imbalance. This is good news because class imbalance brings new challenges.
- Independent variables
  - All variables are categorical type.
  - This requires encoding of categorical values to numerical values to use for ML model training.

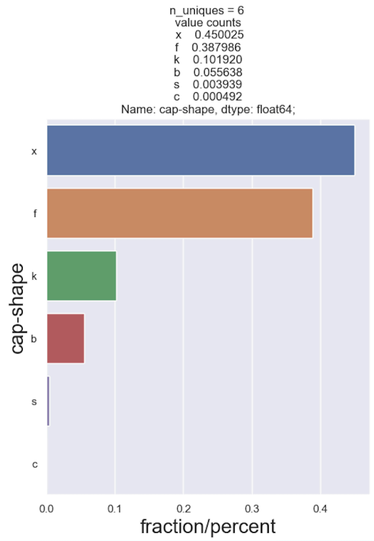
Univariate Analysis - 1 of 3

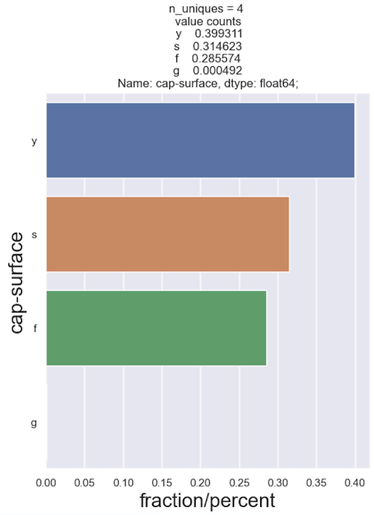
- To explore, understand and analyse distribution of individual features(independent variables)
- None of the variables has null/missing values. This is good news because ML models can’t handle missing values directly and we need to treat missing values to prepare data for model training.
- "veil-type" feature has only one value. It doesn’t carry any signal, thus should be dropped from dataset.
- "stalk-root" feature has some missing values and these are categorised into a separate category - ‘?'
- From the nature of features, tree based algorithms(Decision Tree, Random Forest, Gradient Boosting, etc.) are potential candidates for predictive modelling.

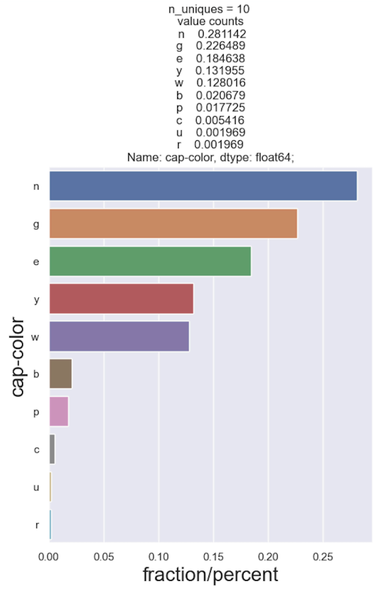
Univariate Analysis - 2 of 3

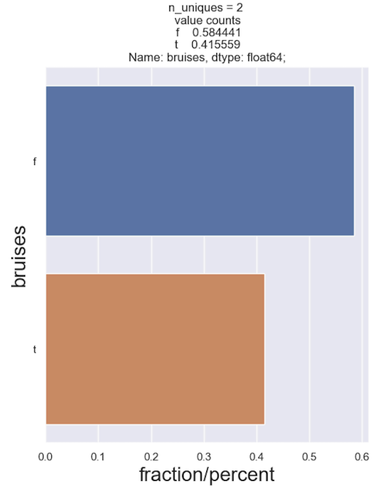
Distribution of independent features

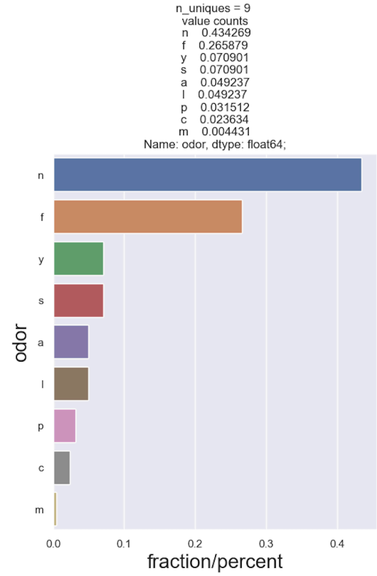


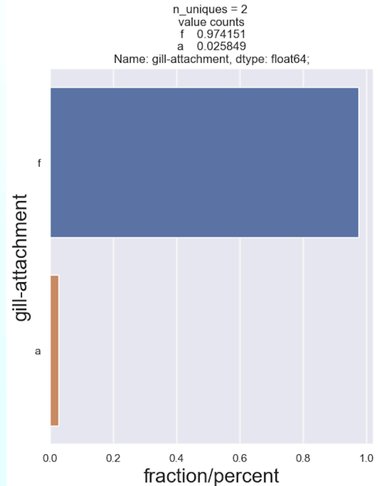


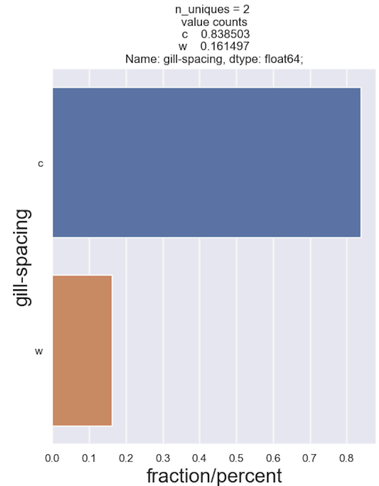


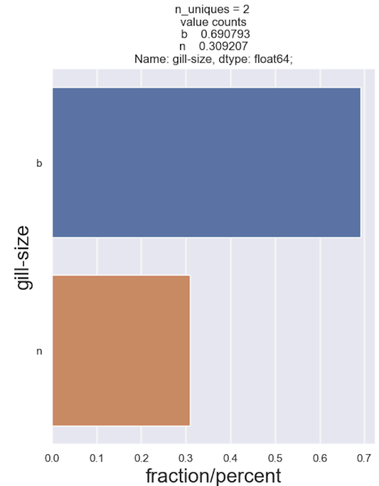


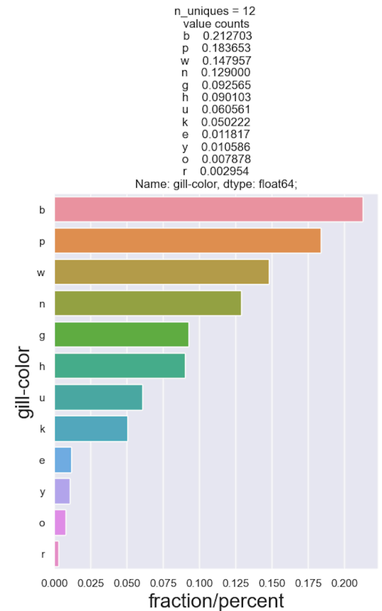


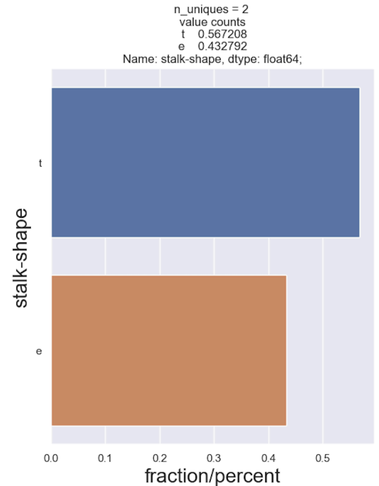


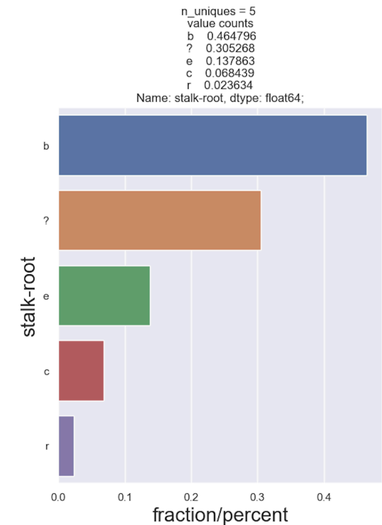






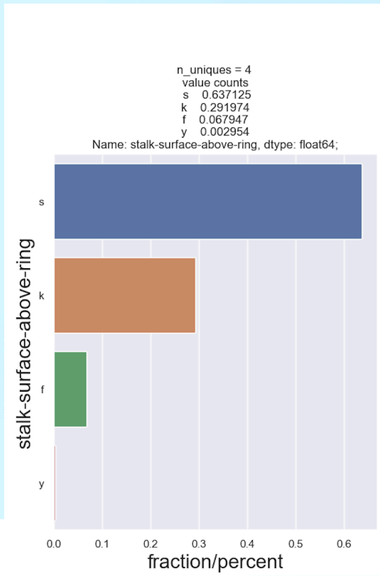


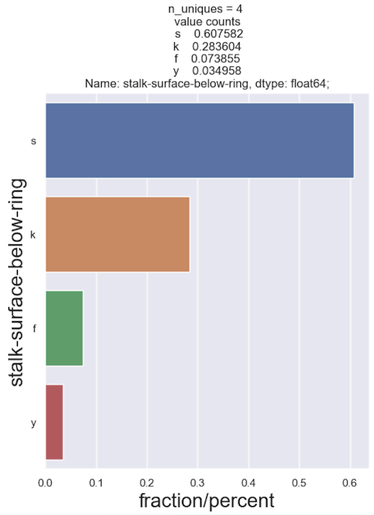


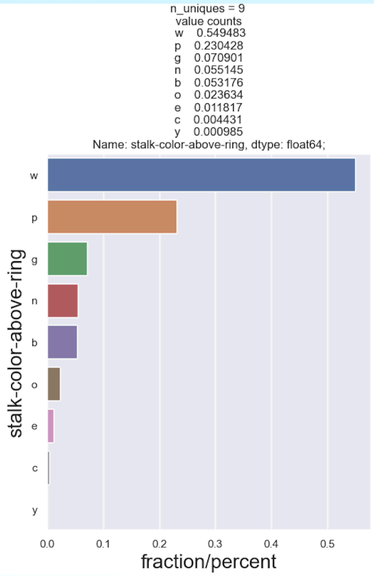


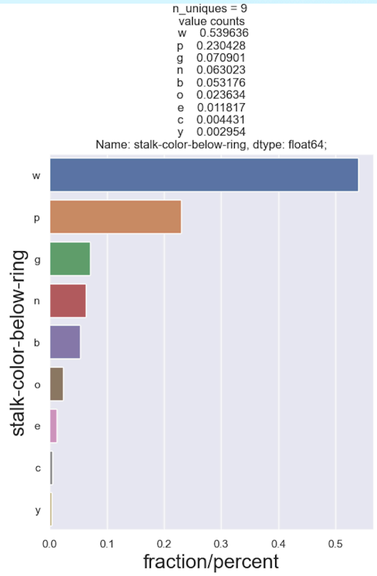
Univariate Analysis - 3 of 3

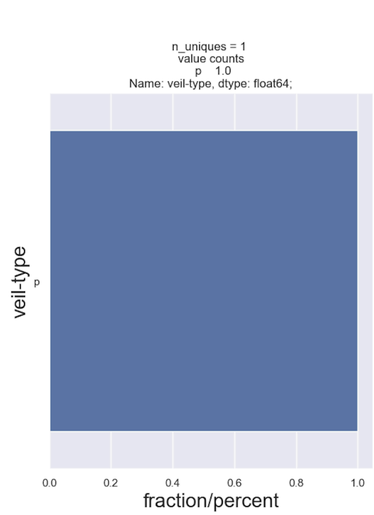
Distribution of independent features

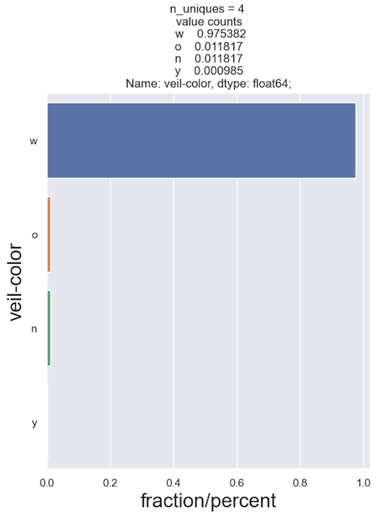


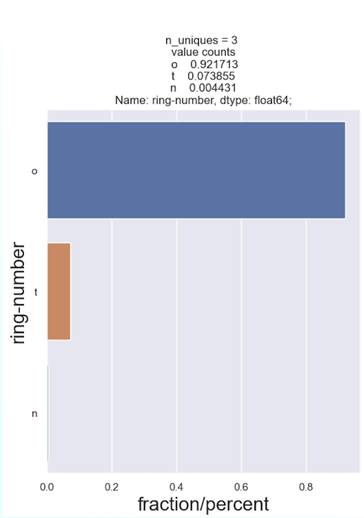


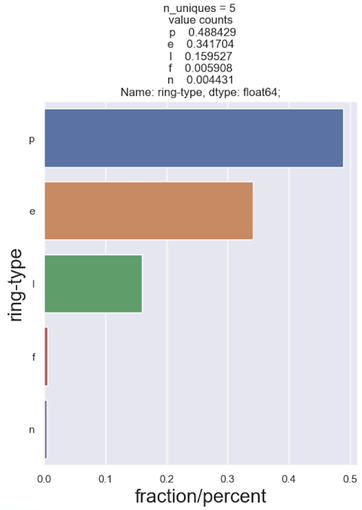


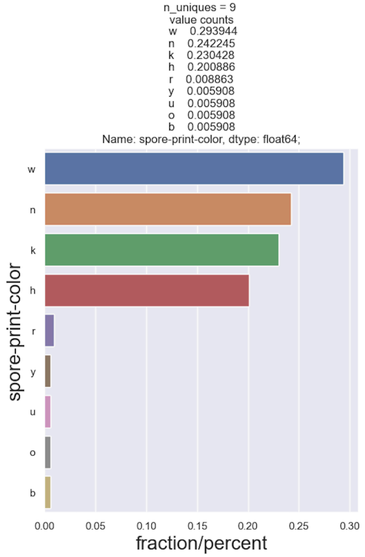


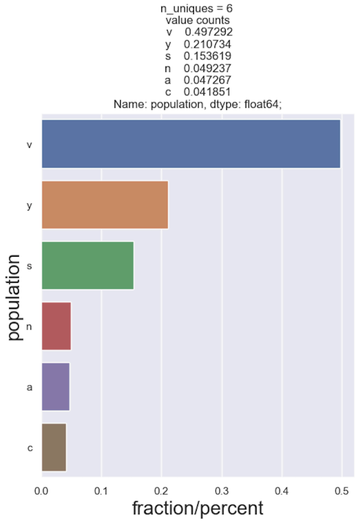


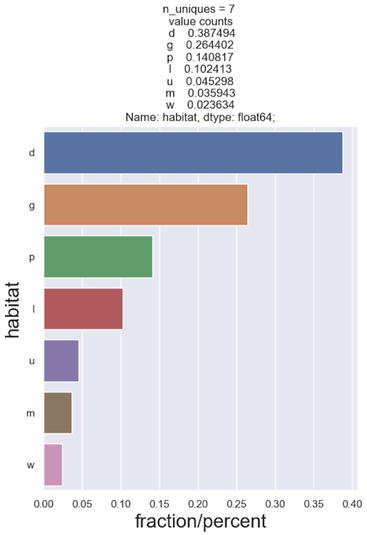










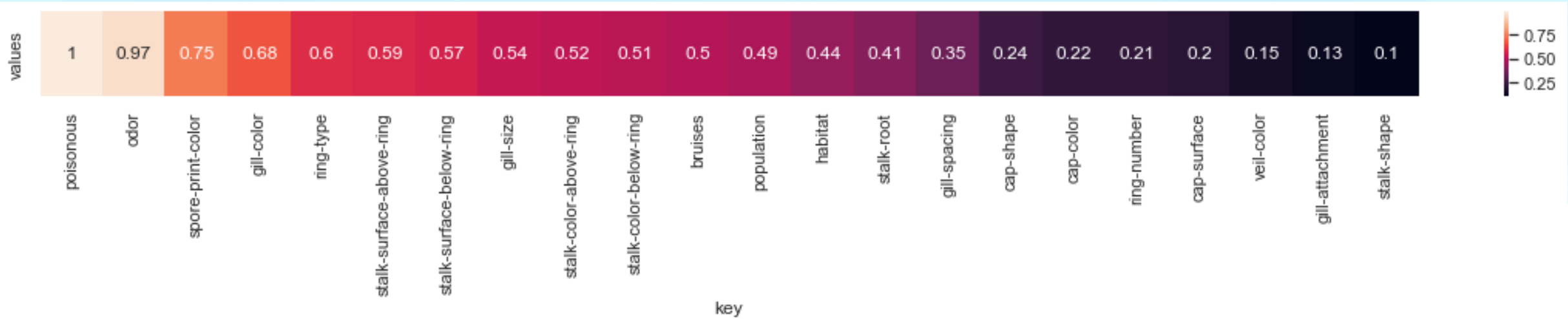


Bivariate analysis - 1 of 3

- To analyse and understand significance of each of the features in predicting target variable class, we have to perform hypothesis testing.
- Have to measure statistical significance of association between target variable (‘poisonous’) and categorical features.
- Chi-Square Test is used to determine significant association between two categorical variables with two or more unique values per variable.
- An effect size metric for the Chi-Square test of independence is Cramer's V.
- Cramer's V can be used to measure strength of the relationship between two categorical variables.
- Crammer's V is computed by taking the square root of the chi-squared statistic divided by the sample size and the minimum dimension minus 1.
- Cramer's V value varies from 0(stating no association between the variables) to 1(stating complete association between variables).
- Effect size(ES) interpretation
    - $ES > 0.6$ The result is strong. The fields are strongly associated.
    - $0.2 < ES \leq 0.6$ The result is moderate. The fields are moderately associated.
    - $ES \leq 0.2$ The result is weak. The fields are only weakly associated.

Bivariate analysis - 2 of 3

- Cramer's V value heat map



- Cramer’s V value above 0.6 indicates strong association between feature and target class
- Three features have very high association with target variable.
  - "odor" (0.97)
  - "spore-print-color" (0.75)
  - “gill-color" (0.68)

Bivariate analysis - 3 of 3

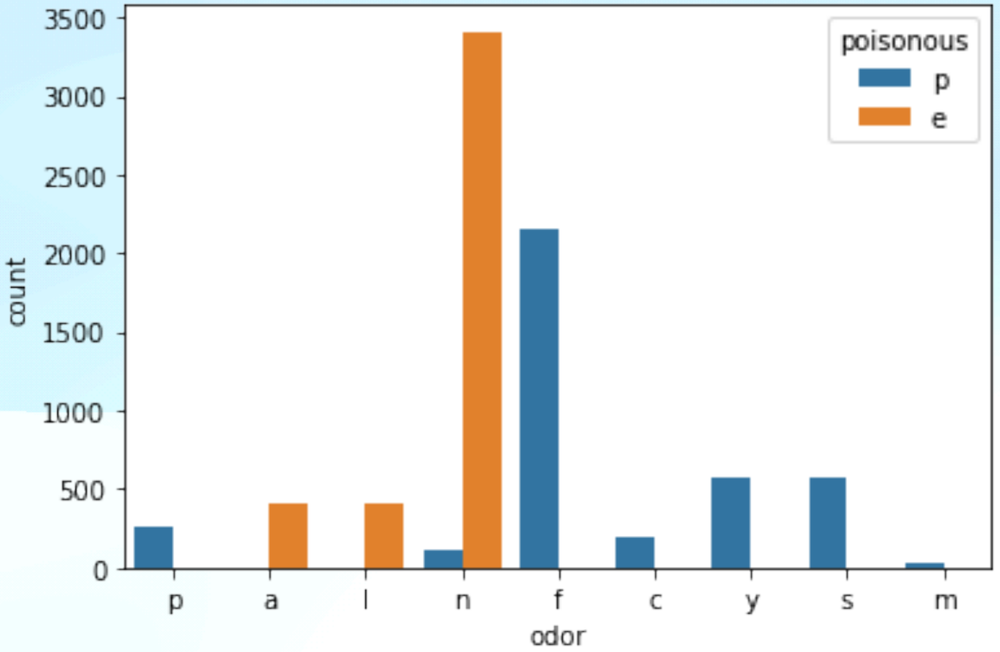
- Analyse association between the three features(‘odor’, ‘spore-print-color’, ‘gill-color’)
  - No strong association between these independent features.
    - Cramer’s V between "odor" and “spore-print-color" is 0.40
    - Cramer’s V between "odor" and “gill-color" is 0.39
    - Cramer’s V between "spore-print-color" and “gill-color" is 0.48
- Going forward we only keep the three features(‘odor’, ‘spore-print-color’, ‘gill-color’) which have strong association with the target variable.

Feature Engineering - 1 of 4

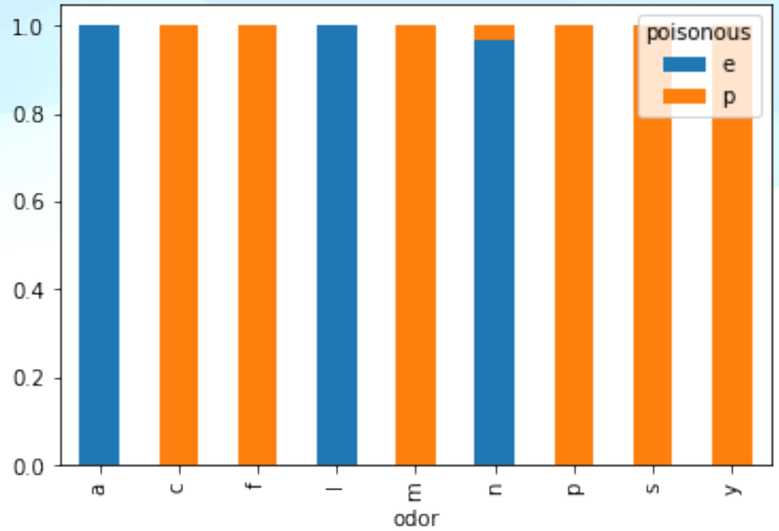
- Feature Engineering is pivotal in Model training and achieve best possible performance
- The objective of Feature Engineering is to use minimum number of high quality features to train models.
- This is required to achieve optimum balance between model complexity and accuracy.
- Also helps to overcome overfitting & underfitting issues.
- When one hot encoding is performed the count of features explode. Therefore we try to minimise unique categorical values of each of the features without loosing signal.

Feature Engineering - 2 of 4

“odor”



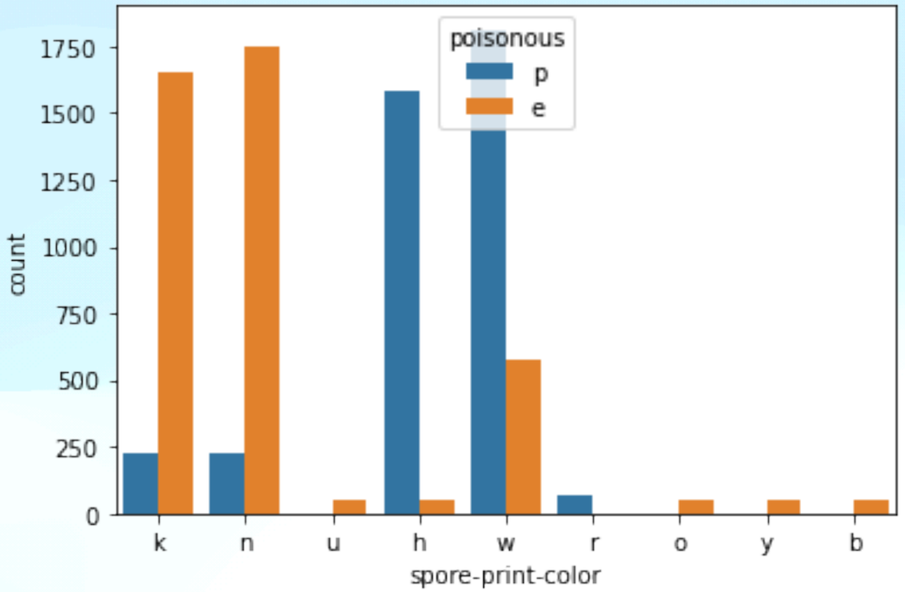
poisonous	e	p
odor		
a	1.000000	NaN
c	NaN	1.000000
f	NaN	1.000000
l	1.000000	NaN
m	NaN	1.000000
n	0.965986	0.034014
p	NaN	1.000000
s	NaN	1.000000
y	NaN	1.000000



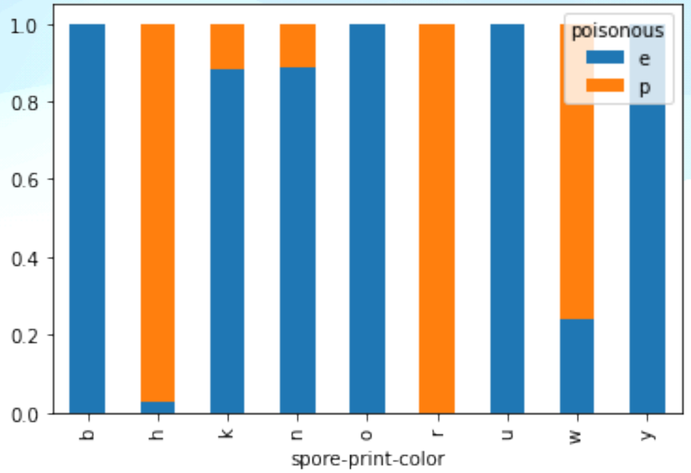
- When the feature has value in ['c', 'f', 'm', 'p', 's', ‘y’], target class is always 'poisionious'
  - Replace all of them with single unique value, e.g. ‘c’
- When the feature has value in [‘a’,'l'], target class is always ‘edible’.
  - Replace these with single unique value, e.g. ‘a’

Feature Engineering - 3 of 4

“spore-print-color”



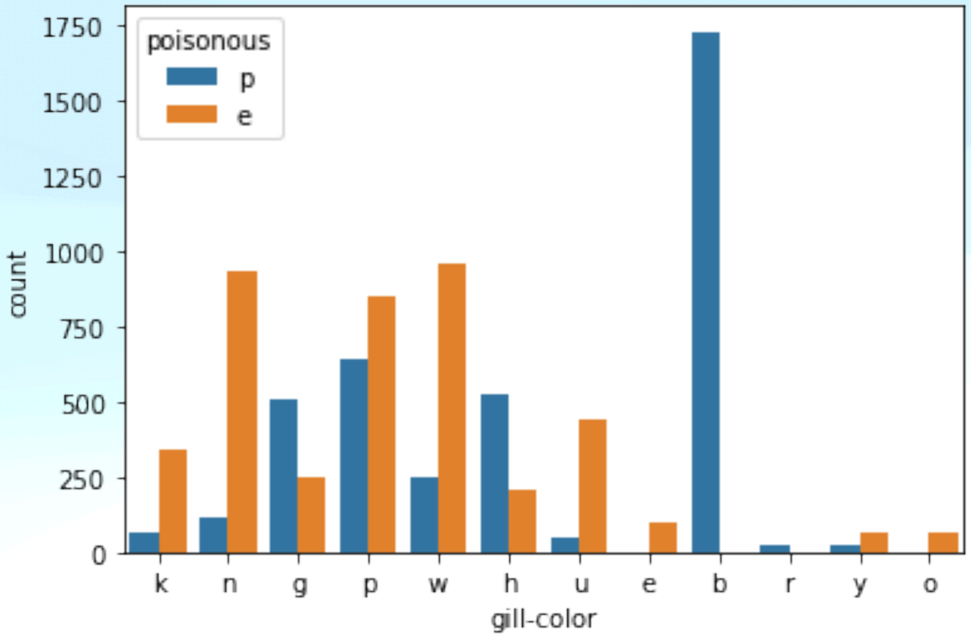
poisonous	e	p
spore-print-color		
b	1.000000	NaN
h	0.029412	0.970588
k	0.880342	0.119658
n	0.886179	0.113821
o	1.000000	NaN
r	NaN	1.000000
u	1.000000	NaN
w	0.241206	0.758794
y	1.000000	NaN



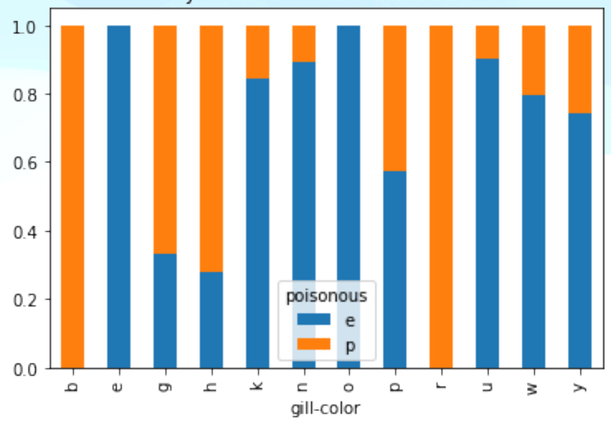
- When the feature has value in ['b', 'o', 'u', ‘y’], target class is always ‘edible'
  - Replace all of these values with single unique value, e.g. ‘b’

Feature Engineering - 4 of 4

“gill-color"



poisonous	e	p
gill-color		
b	NaN	1.000000
e	1.000000	NaN
g	0.329787	0.670213
h	0.278689	0.721311
k	0.843137	0.156863
n	0.893130	0.106870
o	1.000000	NaN
p	0.571046	0.428954
r	NaN	1.000000
u	0.902439	0.097561
w	0.795341	0.204659
y	0.744186	0.255814



- When the feature 'gill-color' has value in [‘b','r'], target class is always 'poisonous'.
  - Replace these two values with single unique value, e.g. ‘b’
- When the feature 'gill-color' has value in ['e', ‘o'], target class is always 'edible'
  - Replace these two values with single unique value, e.g. ‘e’

Model training

- The model training dataset has
  - 3 independent features: ‘odor','spore-print-color','gill-color'
  - 1 target variable: ‘poisonous’
  - 80% observations used to train models
  - 20% observations used to test models
  - Performance of each of the models reported in the next slide

Models performance

Model	Accuracy on test data	Accuracy on train data
Decision Tree	0.992	0.995
kNN	0.9907	0.995
Random Forest	0.992	0.995
XGBM	0.9907	0.995
SVM	0.992	0.995

- All models score same accuracy on training data - 99.5%
- However, SVM, Decision Tree & Random Forest models gives 99.2% accuracy on test data.
- More metrics reported in the Jupyter notebook.

Conclusion

- The three characteristics of mushrooms : ’odor’, ‘spore-print-color’,'gill-color' are reliable to classify for edibility.
- It is possible to further improve individual models performance by hyper parameter tuning using grid search.
- Stratified k-fold cross validation process helps to achieve best possible performance while controlling ‘overfitting’ issue.