2022

Binary Classification of Edible and Poisonous Mushrooms using Classical ML techniques

Priyanka Borwanker - BT19CSE018

Problem Statement

Using Machine Learning to determine whether a certain mushroom is edible or poisonous based on certain features.

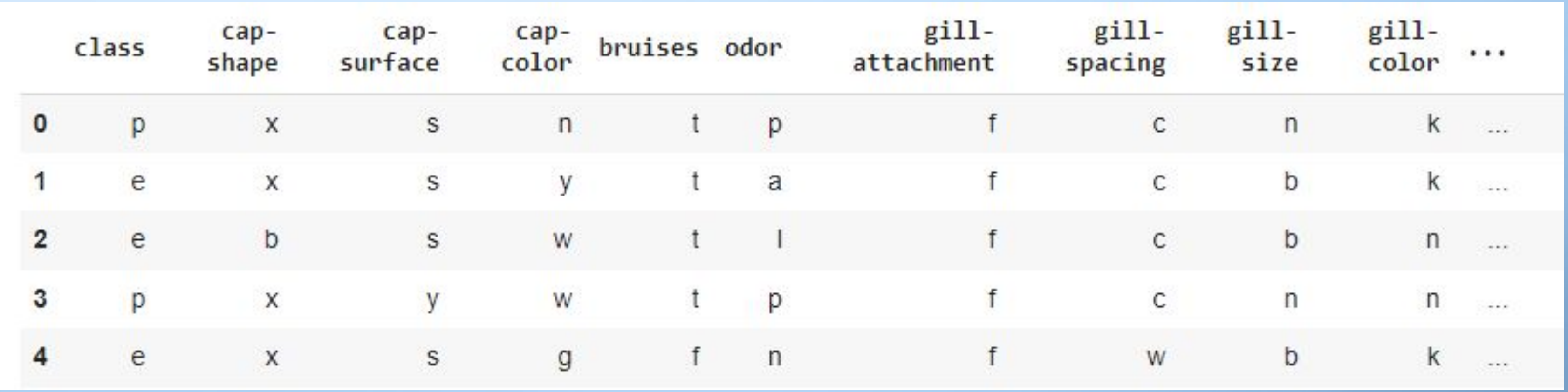
Which features are most indicative of a poisonous mushroom?

What Machine Learning models perform the best on the given dataset?

Dataset

UCI Machine Learning Repository - Mushroom Classification Dataset

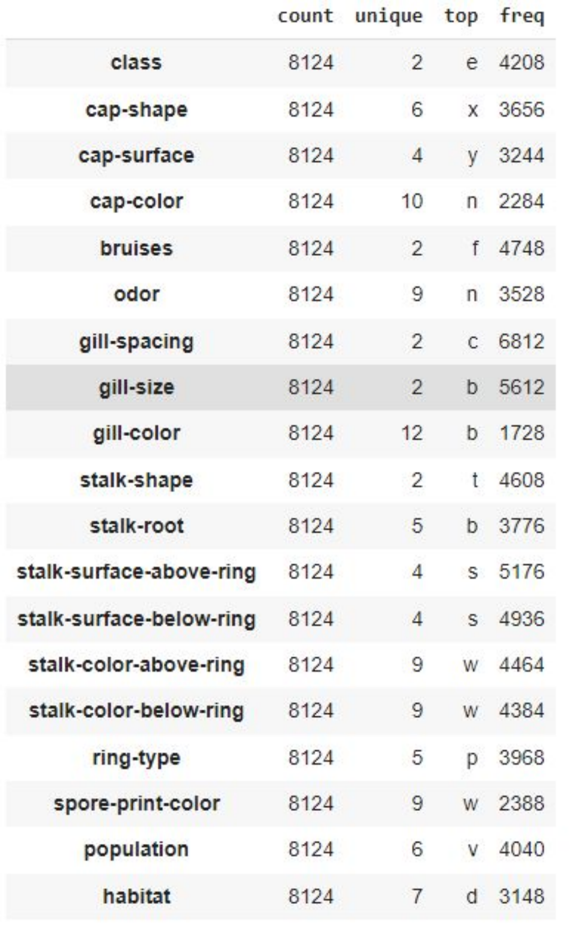
8124 rows - 23 classes, each data row classified into edible (e) or poisonous (p)



	class	cap-shape	cap-surface	cap-color	bruises	odor	gill-attachment	gill-spacing	gill-size	gill-color	...
0	p	x	s	n	t	p	f	c	n	k	...
1	e	x	s	y	t	a	f	c	b	k	...
2	e	b	s	w	t	l	f	c	b	n	...
3	p	x	y	w	t	p	f	c	n	n	...
4	e	x	s	g	f	n	f	w	b	k	...

Data Preprocessing

- ➔ Remove any missing or null attributes from the dataset.
- ➔ Dimensionality Reduction: Remove attributes with low variance for e.g. veil-type has only one value.
- ➔ Encode labels to digits



	count	unique	top	freq
class	8124	2	e	4208
cap-shape	8124	6	x	3656
cap-surface	8124	4	y	3244
cap-color	8124	10	n	2284
bruises	8124	2	f	4748
odor	8124	9	n	3528
gill-spacing	8124	2	c	6812
gill-size	8124	2	b	5612
gill-color	8124	12	b	1728
stalk-shape	8124	2	t	4608
stalk-root	8124	5	b	3776
stalk-surface-above-ring	8124	4	s	5176
stalk-surface-below-ring	8124	4	s	4936
stalk-color-above-ring	8124	9	w	4464
stalk-color-below-ring	8124	9	w	4384
ring-type	8124	5	p	3968
spore-print-color	8124	9	w	2388
population	8124	6	v	4040
habitat	8124	7	d	3148

Data after preprocessing

Proposed Solution

- SVMs:
  - Since this is a problem of binary classification, SVMs are an obvious choice.
  - not sure whether the data is linearly separable or not, so we can try SVMs with different types of kernels (linear, rbf, etc)

- Decision Tree Learning:
  - Under the assumption that all the attributes are independent of each other, we can apply decision tree learning.

- Multi Layer Perceptrons:
  - Multilayer perceptrons are a good option for supervised learning when the relationship between the attributes and output classes is not apparent.
  - A multilayer perceptron can model even non linear classification problems quite well.

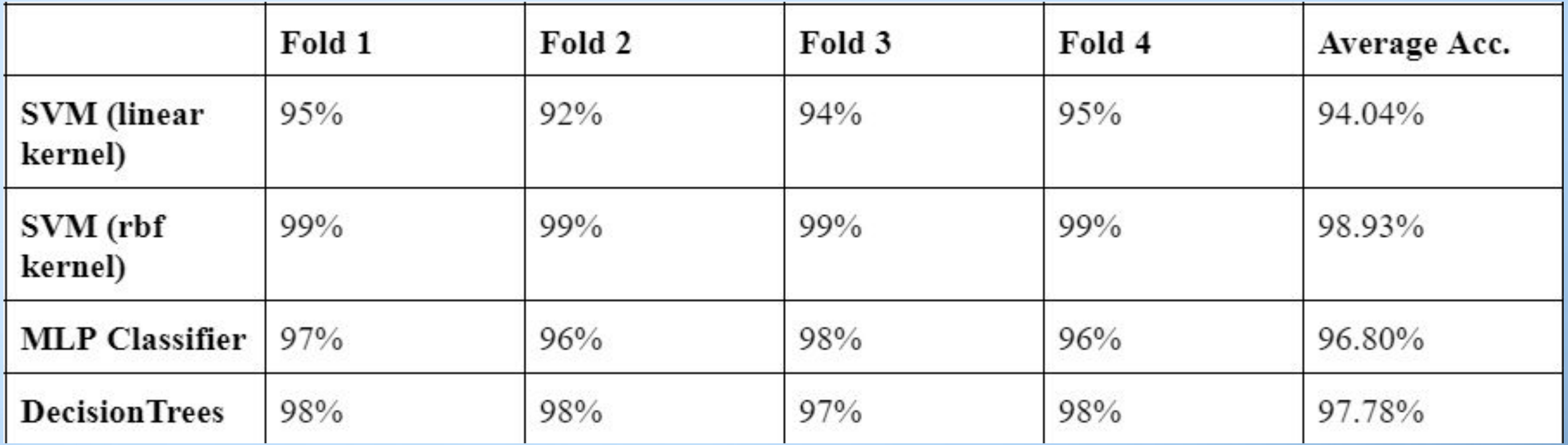
Libraries Used

- Pandas:
  - Open source, easy to use, data management framework
- Scikit-learn
  - Simple and efficient tools for predictive data analysis
  - Built on NumPy, SciPy, and matplotlib
  - Open source, commercially usable

Implementation

- SVMs implemented with two types of kernels (rbf and linear)
  - Linear - standard linear svm
  - Rbf - radial basis function, can model non-linearity in data as well
- MLP Classifier:
  - ReLu activation function
  - 1 hidden layer of 70 neurons(decreased to avoid overfitting)
  - sgd optimizer
  - Learning_rate = 0.0005 (decreased to avoid overfitting)
- Decision Tree:
  - Pre Pruning of decision trees with maximum depth set to 4
- Accuracy Measure:
  - We track when the predicted class and actual class is the same and include that in our score, else the score does not increase.
  - K-fold Cross Validation

Results



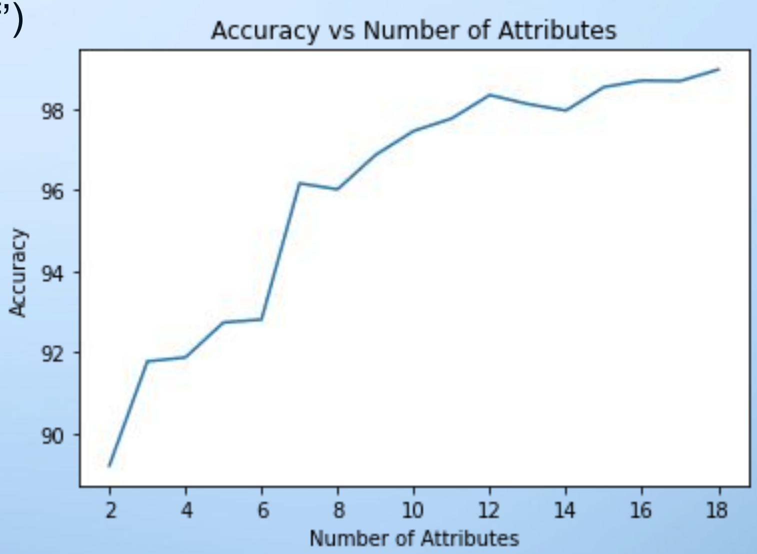
	Fold 1	Fold 2	Fold 3	Fold 4	Average Acc.
SVM (linear kernel)	95%	92%	94%	95%	94.04%
SVM (rbf kernel)	99%	99%	99%	99%	98.93%
MLP Classifier	97%	96%	98%	96%	96.80%
DecisionTrees	98%	98%	97%	98%	97.78%

➔ **SVMs with the rbf kernel are giving the best results for binary classification, which is in accordance with the fact that SVMs are still considered the gold standard for binary classification.**

Feature Selection

Can we achieve similar accuracy with less features?

- ➔ Using chi-square analysis we can pick the top most independent attributes and only use them for classification and check the accuracy.
- ➔ SVC (Kernel = ‘rbf’)

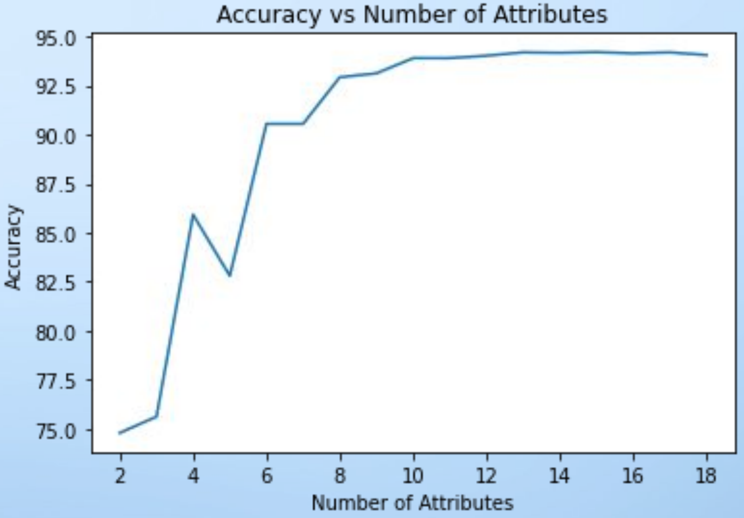


Here we can see that accuracy is increasing almost linearly with the number of features.

Feature Selection

Can we achieve similar accuracy with less features?

➔ SVC (Kernel = ‘linear’)

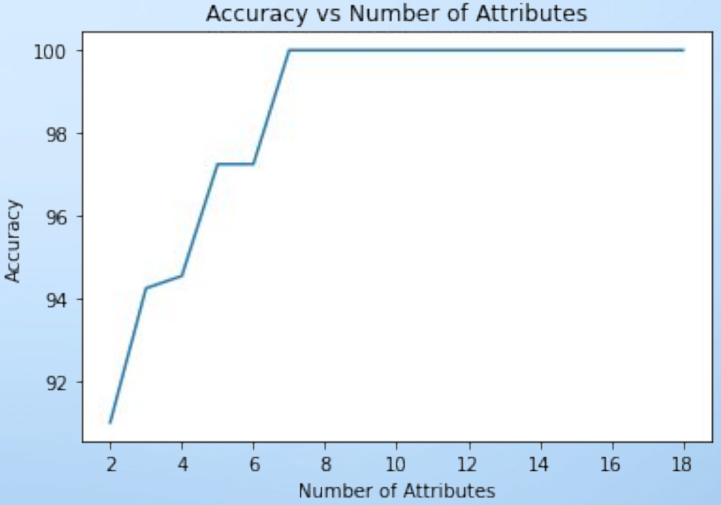


Here we can observe that accuracy is approximately constant after 10 topmost features.

Feature Selection

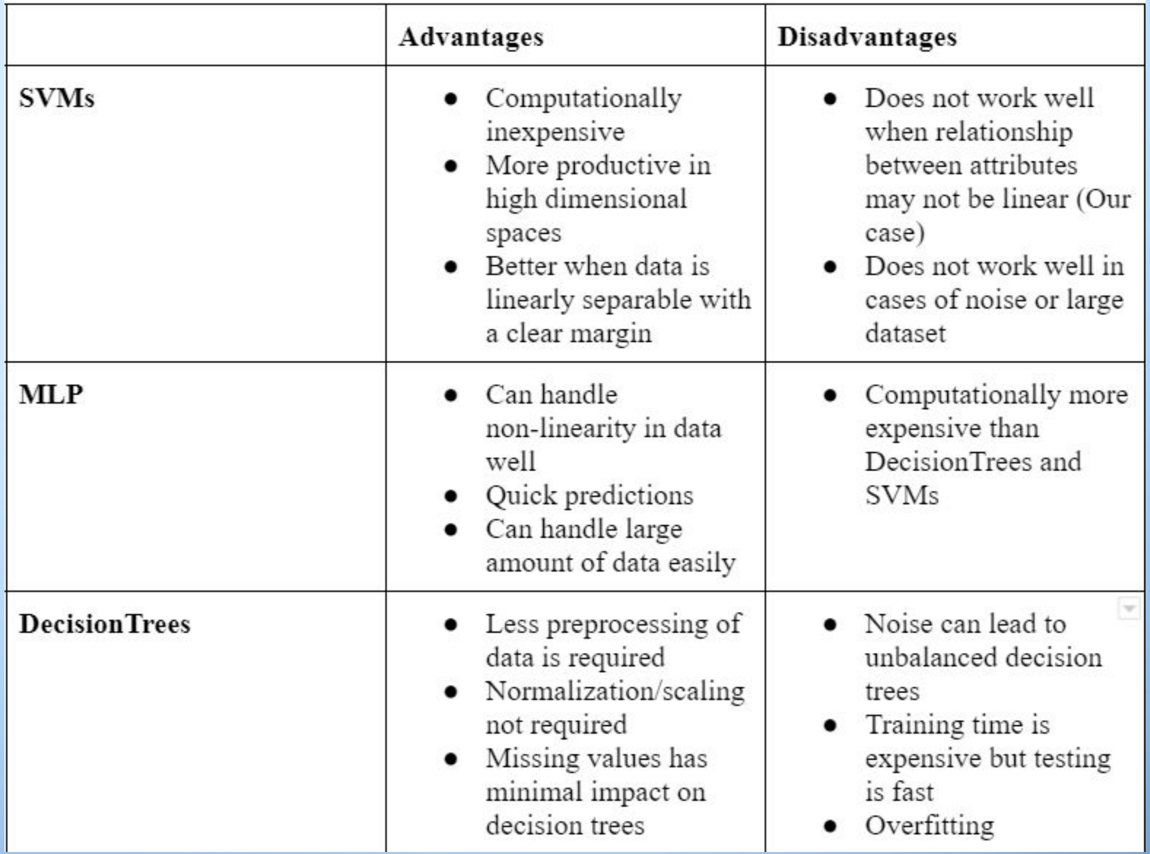
Can we achieve similar accuracy with less features?

➔ Decision Trees



In this, accuracy is the same whether we use the topmost 7 features or all the 18 features.

Conclusion



	Advantages	Disadvantages
SVMs	- Computationally inexpensive - More productive in high dimensional spaces - Better when data is linearly separable with a clear margin	- Does not work well when relationship between attributes may not be linear (Our case) - Does not work well in cases of noise or large dataset
MLP	- Can handle non-linearity in data well - Quick predictions - Can handle large amount of data easily	- Computationally more expensive than DecisionTrees and SVMs
DecisionTrees	- Less preprocessing of data is required - Normalization/scaling not required - Missing values has minimal impact on decision trees	- Noise can lead to unbalanced decision trees - Training time is expensive but testing is fast - Overfitting

Thank you.