

Name: AJAY GOPAL KRISHNA

SBU ID: 112688765

Name: ABHIRAM M KAUSHIK

SBU ID: 112686262

1 RESULT

We have chosen the mushroom dataset from UCI for performing feature selection.

The below plot shows the run-time for SFFS algorithm w.r.t number of top features chosen. From the plot we can observe that filter method takes significantly less time to calculate the top features when compared to wrapper methods.

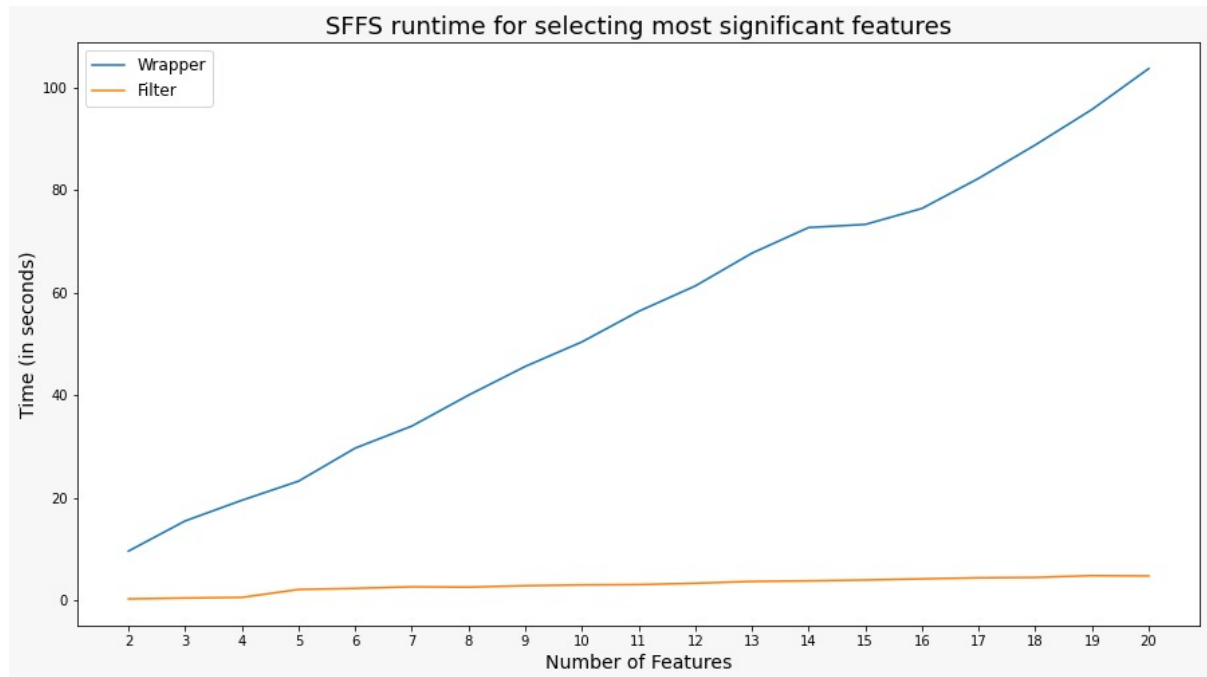


Figure 1: SFFS runtime for selecting most significant features

The below table shows the best feature indexes selected by wrapper and filter methods for $k = 15$.

Method	Feature Indices
Wrapper (k-NN)	[1, 3, 4, 5, 6, 7, 9, 11, 12, 13, 14, 15, 16, 17, 18]
Filter (mahalanobis distance)	[0, 1, 3, 5, 6, 7, 9, 10, 11, 12, 15, 16, 17, 18, 19]

We have compared the results obtained by wrapper method by using the standard mlxtend SFFS library and both results are consistent.

From the graph and table we can see that filter methods run extremely faster when compared to wrapper methods and the best features selected by both methods almost match (12/15

features).

For lower number of features the top features chosen from filter method and wrapper methods differ for our data-set. This behaviour is explained by the following paper
<https://www.sciencedirect.com/science/article/abs/pii/S0031320399000412>.

This paper verifies that for the mushroom data-set the best optimization criterion function is obtained by using 1-NN classifier.

We have implemented a class called "Distance". The "get_significance" method may be overridden to select different optimization criterion for filter methods

2 README

The python code supports the below parameters,

Option	Description	Default Value
-h, -help	show this help message and exit	
-dataset	<i>path_to_dataset</i>	mushroom.csv
-objective_type	wrapper/filter	wrapper
-features	K-best features to select	5
-folds	K-Folds cross validation. Used in wrapper	5
-floating	Select SFFS or SFS	False. SFS by default is used

To execute the **Sequential Forward Selection** algorithm with wrapper method (k-NN) using 5 fold cross validation to select 10 best features execute,

```
python feature_selection.py -dataset mushroom.csv -objective_type wrapper -features 10 -folds 5
```

To execute the **Sequential Forward Selection** algorithm with filter method (mahalanobis distance) to select 10 best features execute,

```
python feature_selection.py -dataset mushroom.csv -objective_type filter -features 10
```

To execute the **Sequential Forward Floating Selection** algorithm with wrapper method (k-NN) using 5 fold cross validation to select 10 best features execute,

```
python feature_selection.py -dataset mushroom.csv -objective_type wrapper -features 10 -folds 5 -floating
```

To execute the **Sequential Forward Floating Selection** algorithm with filter method (mahalanobis distance) to select 10 best features execute,

```
python feature_selection.py -dataset mushroom.csv -objective_type filter -features 10 -floating
```