Feature selection using Recursive Feature Elimination

Monir Zaman

Feature selection

Feature selection is an important step in the machine learning workflow. It is process of identifying important predictors and remove unnecessary features.

Why needed:

- Feature selection useful to prevent overfitting
- Reduce dimensions
- Helps to understand the data

Feature selection in Machine learning workflow

- Typically applied between data-preprocessing and model building.

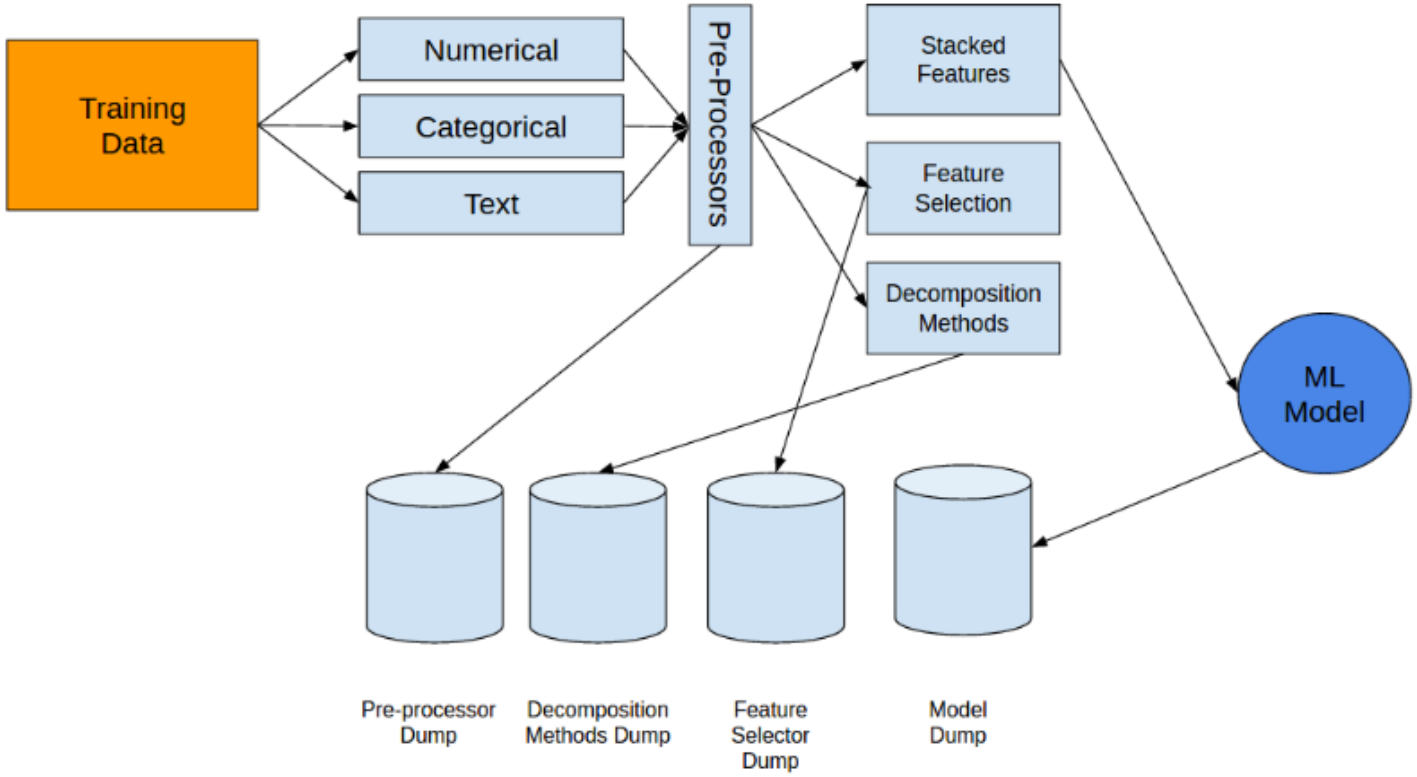


Image courtesy: http://s5047.pcdn.co/wp-content/uploads/2016/07/abhishek_25.png

Feature selection algorithm

- Univariate feature selection
  - Example: Correlation-based selection
- Multivariate feature selection
  - Example: Recursive feature elimination using Cross-Validation (RFECV)

Recursive feature elimination using Cross-Validation (RFECV)

- RFECV applies RFE on k-fold cross-validation of the training dataset
  - RFE stands for Recursive feature elimination
  - RFE recursively removes features by m features at a time. Default m = 1. It repeats the process until all the features are exhausted or number of features reached the threshold of minimum number of features to select

Recursive feature elimination (rfe)

X

Y

# bedrooms	Area	Age	Price (1k)
3	1200	10	560
3	980	13	520
3	1200	20	410
5	1600	4	880

Model fit on Train set

Train

Test

Program flow of rfe (_rfe_single_fit)

Scores (r-squared on Test set)

.78		

ranking

# bedrooms	Area	Age
1	1	1

model.feature_importance

# bedrooms	Area	Age
.6	.3	.1

Task: Calculate scores of the feature set

X

Y

# bedrooms	Area	Age	Price (1k)
3	1200	10	560
3	980	13	520
3	1200	20	410
5	1600	4	880

Program flow of rfe (_rfe_single_fit)

Scores (r-squared on Test set)

.78	.88 (delayed unitll next round)	

ranking

# bedrooms	Area	Age
1	1	2

Model fit on Train set

Train

Test

model.feature_importance

# bedrooms	Area	Age
.6	.3	.1

# bedrooms	Area	Age
.6	.3	...

Task: Remove worst performing feature

X

Y

# bedrooms	Area	Age	Price (1k)
3	1200	10	560
3	980	13	520
3	1200	20	410
5	1600	4	880

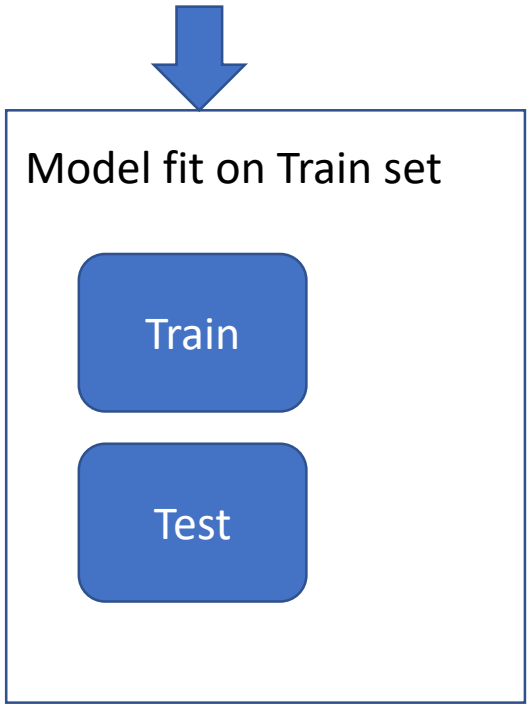
Program flow of rfe (_rfe_single_fit)

Scores (r-squared on Test set)

.78	.88	.6 (delayed unitll next round)

ranking

# bedrooms	Area	Age
1	2	3



model.feature_importance

# bedrooms	Area
.56	.44

# bedrooms	Area
.56	.44

Task: Remove worst performing feature

- Recursive feature elimination (rfe) is repeated until

  #(remaining features) >= n_features_to_select

  where n_features_to_select is a parameter passed during runtime

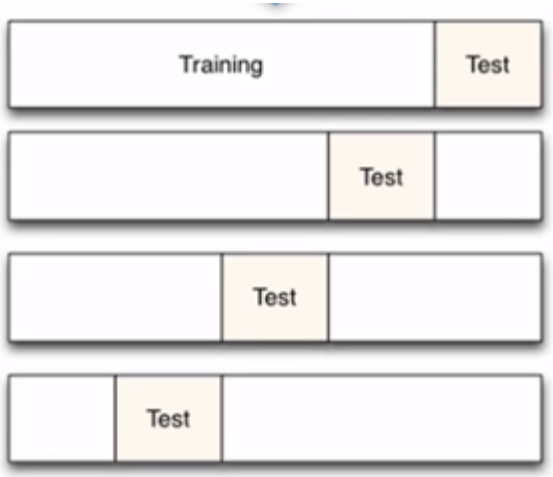
- When n_features_to_select is None, then

  n_features_to_select = Total number of features / 2

Recursive feature elimination using cross-validation (rfecv)

Program flow of rfecv.fit method

- rfecv repeats rfe with k-fold cross validation



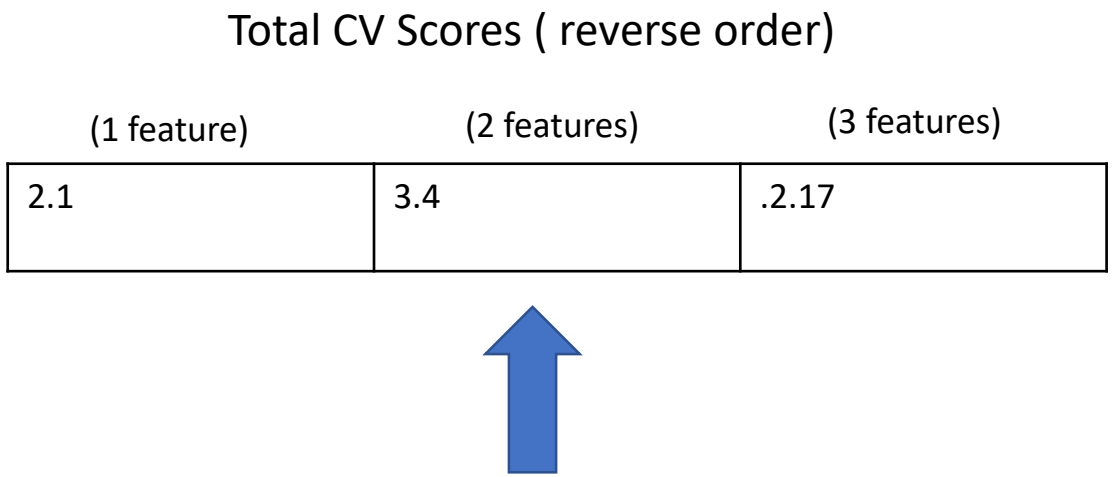
Scores (r-squared on Test set)

.56	.67	.4
.33	.8	.9
.78	.88	.2
.5	.79	.6

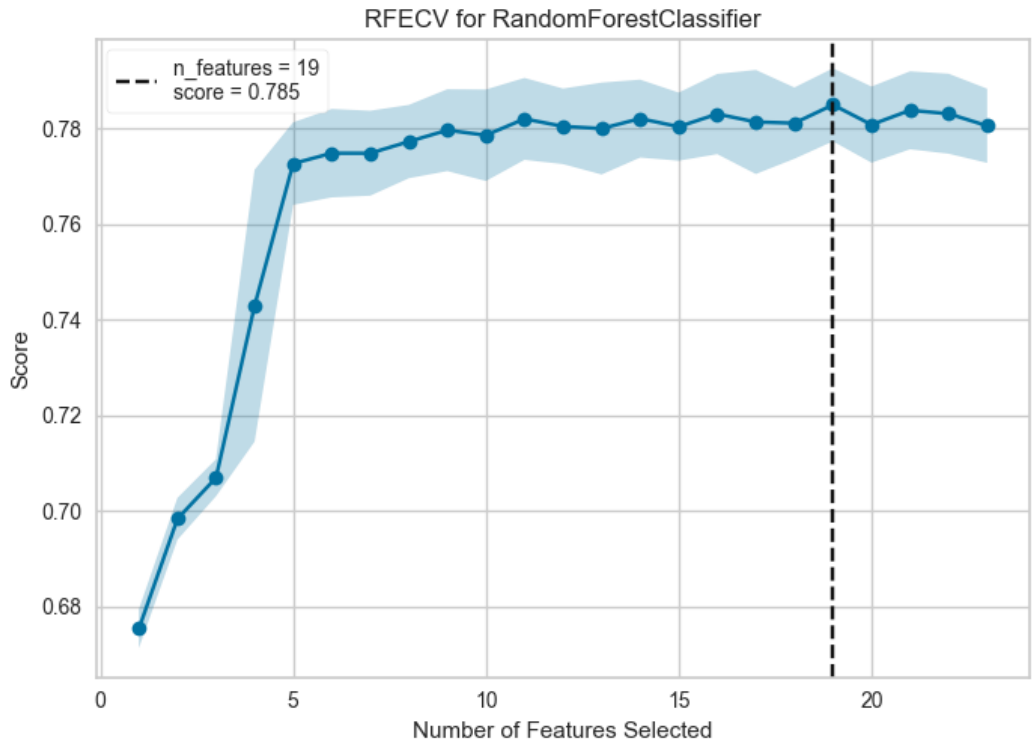
$\sum = 2.17$ $\sum = 3.14$ $\sum = 2.1$

Program flow of rfecv.fit method

- Position where cv scores reach its peak is selected as the optimal number of features (aka n_features_to_select)
- In the following example, number of optimal features is 2



- Example of CV scores plot where optimal number of features is 19



- Helps us to decide the minimum number of features needed
- Calls rfe one more time with n_features_to_select = (Total # f
  on the entire dataset.
- Last call changes ranking but not CV.scores