Feature Selection Tool

@date:2017-08-28
@author:charleshen

File:

feature_selection.py

main function:

```
SFE(df, estimator, param_grid, im_method = 'seq', sel_method = 'sfe',
      vip_feat = 20, Forward = True, max_feat = 50, batch_feat = 1,
      para_search_step = 1, random_state = None, n_jobs = 4,
      cv = 5, randcv = True)
```

PARAS1: parameter of im_method, get feature importance of each feature, support:

'fscore': F-score of features
'pcc': Person corr. Coefficient of features
'tree': optimal tree’s feature importance(gini importance)
'lasso': weights of features for optimal lasso regression model
'elnet': weights of features for optimal elnet regression model
'seq': sequence of the features(default, namely feature importance is feature’s sequence(order))

PARAS2: parameter of sel_method, method of get best feature numbers, support:

'sfe': Stepwise Feature Elimination, if Forward is True, then feature addiation(add features one by one using recursive try algorithm)

'ofe': Ordered Feature Elimination, if Forward is True, then feature addiation(add features one by one with the original order or feature importance order)

PARAS3: other parameters:

vip_feat = 20: very important features, will be fixed in the elimination or addition
Forward = True: if forward is true, features will be added one by one instead of elimination

max_feat = 50: max number of features

batch_feat = 1: batch size of features for addition or elimination

para_search_step = 1: step of search best parameters during addition or elimination

random_state = None

n_jobs = 4: number of threads for parallelization

cv = 5: cross validation’s fold, default cross validated score is ROC-AUC for classification, R-squared for regression

randcv = True: if True, then random search method will be used in the grid-search.

Tests:

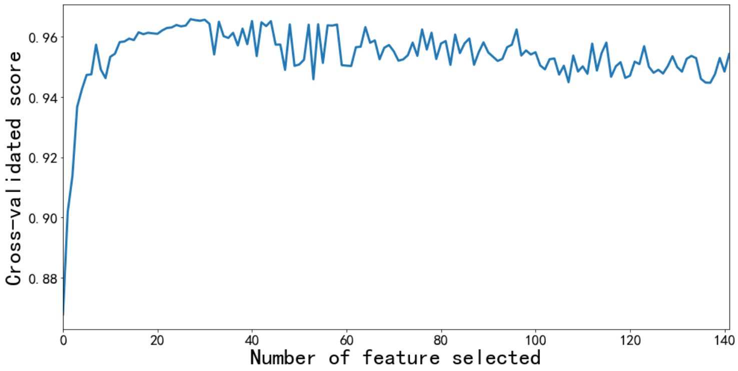
the tool has been tested on dataset of bairong(bairong_train.csv)

OFE test:

estimator = SVM()

im_method= ‘tree’

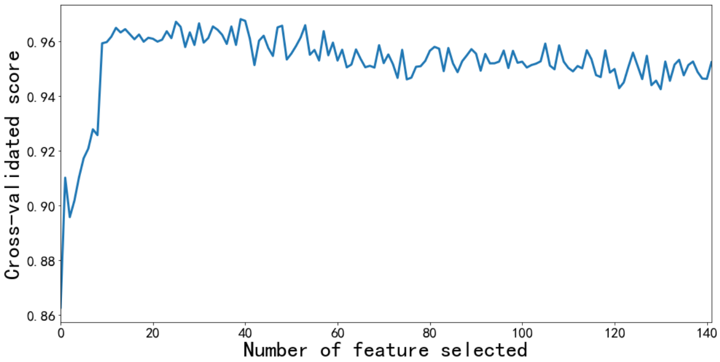
sel_method = ‘ofe’:



estimator = SVM()

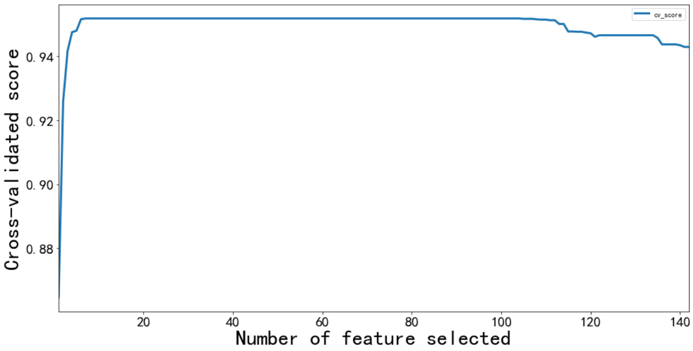
im_method= ‘lasso’

sel_method = ‘ofe’

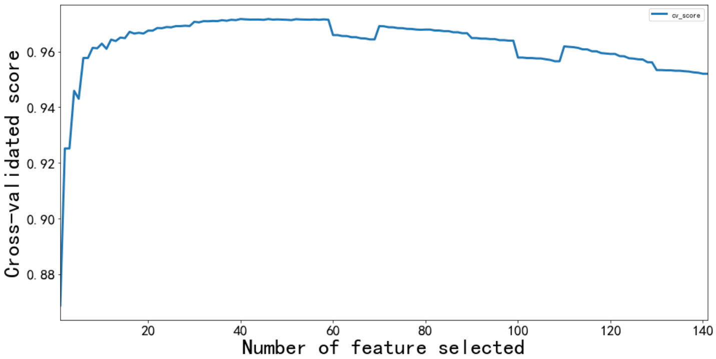


SFE test:

estimator = DecisionTreeClassifier()
im_method= ‘tree’
sel_method = ‘sfe’



estimator = SVM()
im_method= ‘tree’
sel_method = ‘sfe’



Conclusions:

- Both OFE and SFE can be used to select best number of features to avoid over-fitting;
- SFE is a better method to select best number of features(Because SFE can raise cross validated score step by step)
- OFE is more faster than SFE