```
In [1]: import pandas as pd
        import numpy as np
```

```
In [2]: # load the data
        # https://www.kaggle.com/datasets/ted-mehta/kepler-data-science-exoplanets-finding?select=cumulative.csv
        df = pd.read_csv('cumulative.csv')
        df.head(10)
```

| | rowid | kepid | kepoi_name | kepler_name | koi_disposition | koi_pdisposition | koi_score | koi_fpflag_nt | koi_fpflag_ss | koi_fpflag_co | koi_fpflag_ec | koi_period | koi_period_err1 | koi_period_err2 | koi_time0bk | koi_time0bk_err1 | koi_time0bk_err2 | koi_eccen | koi_eccen_err1 | koi_eccen_err2 | koi_longp | koi_longp_err1 | koi_longp_err2 | koi_impact | koi_impact_err1 | koi_impact_err2 | koi_duration | koi_duration_err1 | koi_duration_err2 | koi_depth | koi_depth_err1 | koi_depth_err2 | koi_prad | koi_prad_err1 | koi_prad_err2 | koi_teq | koi_teq_err1 | koi_teq_err2 | koi_insol | koi_insol_err1 | koi_insol_err2 | koi_model_snr | koi_tce_plnt_num | koi_tce_delivname | koi_slogg | koi_slogg_err1 | koi_slogg_err2 | koi_slogg_err2 | koi_slogg_err2

```
from sklearn.feature_selection import rfe
# Feature extraction
model = LogisticRegression()
rfe = RFE(model, 3)
rfe = RFE(estimator=LogisticRegression(), n_features_to_select=3)
fit = rfe.fit(X, Y)
print("Num Features: %d",%fit.n_features_)
print("Selected Features: %s",%fit.support_)
print("Feature Ranking: %s",%fit.ranking_)
# summarize the selection of the attributes
print(zip(range(0,9),
          fit.ranking_,
          dataset.columns))
# show the selected features
features = fit.support_
print(features,dtype=int,sep=',')
```

```
Num Features: %d, 3
Selected Features: %s, [True False False False False False False False False]
Feature Ranking: %s, [1 2 3 4 5 6 7]
<zip_iterator_1>
[True,False,False,False,False,False,False,False,False]
```

	Feature_name	Selected	RFE_ranking
0	id	True	1
1	Clump_thickness	False	2
2	Uniform_cell_size	False	3
3	Uniform_cell_shape	False	4
4	Marginal_adhesion	False	5
5	Single_epithelial_cell_size	False	6
6	Bare_nuclei	False	7
7	Bland_chromatin	False	8
8	Normal_nucleoli	False	9

```
# Feature extraction
model = LogisticRegression()
rfe = RFE(model, 3)
rfe = RFE(estimator=LogisticRegression(), n_features_to_select=3)
fit = rfe.fit(X, Y)
print("Num Features: %d",%fit.n_features_)
print("Selected Features: %s",%fit.support_)
print("Feature Ranking: %s",%fit.ranking_)
# summarize the selection of the attributes
print(zip(range(0,9),
          fit.ranking_,
          dataset.columns))
# show the selected features
features = fit.support_
print(features,dtype=int,sep=',')
```

```
Num Features: %d, 3
Selected Features: %s, [True False False False False False False False False]
Feature Ranking: %s, [1 2 3 4 5 6 7]
<zip_iterator_2>
[True,False,False,False,False,False,False,False,False]
```

	Feature_name	Selected	RFE_ranking
0	id	True	1
1	Clump_thickness	False	2
2	Uniform_cell_size	False	3
3	Uniform_cell_shape	False	4
4	Marginal_adhesion	False	5
5	Single_epithelial_cell_size	False	6
6	Bare_nuclei	False	7
7	Bland_chromatin	False	8
8	Normal_nucleoli	False	9

backward elimination using SFS

```
# Feature Extraction with Recursive Feature Elimination
from sklearn.feature_selection import RFE
from sklearn.linear_model import LogisticRegression
# feature extraction
model = LogisticRegression()
rfe = RFE(model, 3)
fit = rfe.fit(X, Y)
print("Num Features: %d",%fit.n_features_)
print("Selected Features: %s",%fit.support_)
print("Feature Ranking: %s",%fit.ranking_)
# summarize the selection of the attributes
print(zip(range(0,9),
          fit.ranking_,
          dataset.columns))
# show the selected features
features = fit.support_
print(features,dtype=int,sep=',')
```

```
Num Features: %d, 3
Selected Features: %s, [True False False False False False False False False]
Feature Ranking: %s, [1 2 3 4 5 6 7]
<zip_iterator_3>
[True,False,False,False,False,False,False,False,False]
```

	Feature_name	id_score	chi_score	Feature_score
0	id	0.013333333333333	2.006669	2.019999333333333
1	Clump_thickness	0.139333333333333	16.118096	16.257429333333333
2	Uniform_cell_size	0.268666666666667	1.000025	1.268691666666667
3	Uniform_cell_shape	0.164666666666667	1.307511	1.472177666666667
4	Marginal_adhesion	0.096666666666667	0.814726	0.911392666666667
5	Single_epithelial_cell_size	0.047333333333333	1.057334	1.104666666666667
6	Bare_nuclei	0.207666666666667	4.306261	4.513927666666667
7	Bland_chromatin	0.153666666666667	1.546964	1.700630666666667
8	Normal_nucleoli	0.094666666666667	0.989191	1.083857666666667

4.3 Step-wise Selection — SFFS) from mlxtend

```
# Feature Extraction with Recursive Feature Elimination
from sklearn.feature_selection import RFE
from sklearn.linear_model import LogisticRegression
# feature extraction
model = LogisticRegression()
rfe = RFE(model, 3)
fit = rfe.fit(X, Y)
print("Num Features: %d",%fit.n_features_)
print("Selected Features: %s",%fit.support_)
print("Feature Ranking: %s",%fit.ranking_)
# summarize the selection of the attributes
print(zip(range(0,9),
          fit.ranking_,
          dataset.columns))
# show the selected features
features = fit.support_
print(features,dtype=int,sep=',')
```

```
Num Features: %d, 3
Selected Features: %s, [True False False False False False False False False]
Feature Ranking: %s, [1 2 3 4 5 6 7]
<zip_iterator_4>
[True,False,False,False,False,False,False,False,False]
```

	Feature_name	id_score	chi_score	Feature_score
0	id	0.013333333333333	2.006669	2.019999333333333
1	Clump_thickness	0.139333333333333	16.118096	16.257429333333333
2	Uniform_cell_size	0.268666666666667	1.000025	1.268691666666667
3	Uniform_cell_shape	0.164666666666667	1.307511	1.472177666666667
4	Marginal_adhesion	0.096666666666667	0.814726	0.911392666666667
5	Single_epithelial_cell_size	0.047333333333333	1.057334	1.104666666666667
6	Bare_nuclei	0.207666666666667	4.306261	4.513927666666667
7	Bland_chromatin	0.153666666666667	1.546964	1.700630666666667
8	Normal_nucleoli	0.094666666666667	0.989191	1.083857666666667

```
from sklearn.feature_selection import SelectKBest
# feature extraction
model = SelectKBest(k=3)
fit = model.fit(X, Y)
print("Num Features: %d",%fit.n_features_)
print("Selected Features: %s",%fit.support_)
print("Feature Ranking: %s",%fit.ranking_)
# summarize the selection of the attributes
print(zip(range(0,9),
          fit.ranking_,
          dataset.columns))
# show the selected features
features = fit.support_
print(features,dtype=int,sep=',')
```

```
Num Features: %d, 3
Selected Features: %s, [True False False False False False False False False]
Feature Ranking: %s, [1 2 3 4 5 6 7]
<zip_iterator_5>
[True,False,False,False,False,False,False,False,False]
```

	Feature_name	id_score	chi_score	Feature_score
0	id	0.013333333333333	2.006669	2.019999333333333
1	Clump_thickness	0.139333333333333	16.118096	16.257429333333333
2	Uniform_cell_size	0.268666666666667	1.000025	1.268691666666667
3	Uniform_cell_shape	0.164666666666667	1.307511	1.472177666666667
4	Marginal_adhesion	0.096666666666667	0.814726	0.911392666666667
5	Single_epithelial_cell_size	0.047333333333333	1.057334	1.104666666666667
6	Bare_nuclei	0.207666666666667	4.306261	4.513927666666667
7	Bland_chromatin	0.153666666666667	1.546964	1.700630666666667
8	Normal_nucleoli	0.094666666666667	0.989191	1.083857666666667

```
from sklearn import matplotlib
from matplotlib import pyplot
# feature extraction
model = LogisticRegression()
rfe = RFE(model, 3)
fit = rfe.fit(X, Y)
print("Num Features: %d",%fit.n_features_)
print("Selected Features: %s",%fit.support_)
print("Feature Ranking: %s",%fit.ranking_)
# summarize the selection of the attributes
print(zip(range(0,9),
          fit.ranking_,
          dataset.columns))
# show the selected features
features = fit.support_
print(features,dtype=int,sep=',')
```

```
Num Features: %d, 3
Selected Features: %s, [True False False False False False False False False]
Feature Ranking: %s, [1 2 3 4 5 6 7]
<zip_iterator_6>
[True,False,False,False,False,False,False,False,False]
```

	Feature_name	id_score	chi_score	Feature_score
0	id	0.013333333333333	2.006669	2.019999333333333
1	Clump_thickness	0.139333333333333	16.118096	16.257429333333333
2	Uniform_cell_size	0.268666666666667	1.000025	1.268691666666667
3	Uniform_cell_shape	0.164666666666667	1.307511	1.472177666666667
4	Marginal_adhesion	0.096666666666667	0.814726	0.911392666666667
5	Single_epithelial_cell_size	0.047333333333333	1.057334	1.104666666666667
6	Bare_nuclei	0.207666666666667	4.306261	4.513927666666667
7	Bland_chromatin	0.153666666666667	1.546964	1.700630666666667
8	Normal_nucleoli	0.094666666666667	0.989191	1.083857666666667

```
from sklearn.feature_selection import SelectFromModel
# feature extraction
model = LogisticRegression()
rfe = RFE(model, 3)
fit = rfe.fit(X, Y)
print("Num Features: %d",%fit.n_features_)
print("Selected Features: %s",%fit.support_)
print("Feature Ranking: %s",%fit.ranking_)
# summarize the selection of the attributes
print(zip(range(0,9),
          fit.ranking_,
          dataset.columns))
# show the selected features
features = fit.support_
print(features,dtype=int,sep=',')
```

```
Num Features: %d, 3
Selected Features: %s, [True False False False False False False False False]
Feature Ranking: %s, [1 2 3 4 5 6 7]
<zip_iterator_7>
[True,False,False,False,False,False,False,False,False]
```

	Feature_name	id_score	chi_score	Feature_score
0	id	0.013333333333333	2.006669	2.019999333333333
1	Clump_thickness	0.139333333333333	16.118096	16.257429333333333
2	Uniform_cell_size	0.268666666666667	1.000025	1.268691666666667
3	Uniform_cell_shape	0.164666666666667	1.307511	1.472177666666667
4	Marginal_adhesion	0.096666666666667	0.814726	0.911392666666667
5	Single_epithelial_cell_size	0.047333333333333	1.057334	1.104666666666667
6	Bare_nuclei	0.207666666666667	4.306261	4.513927666666667
7	Bland_chromatin	0.153666666666667	1.546964	1.700630666666667
8	Normal_nucleoli	0.094666666666667	0.989191	1.083857666666667

```
from sklearn import matplotlib
from matplotlib import pyplot
# feature extraction
model = LogisticRegression()
rfe = RFE(model, 3)
fit = rfe.fit(X, Y)
print("Num Features: %d",%fit.n_features_)
print("Selected Features: %s",%fit.support_)
print("Feature Ranking: %s",%fit.ranking_)
# summarize the selection of the attributes
print(zip(range(0,9),
          fit.ranking_,
          dataset.columns))
# show the selected features
features = fit.support_
print(features,dtype=int,sep=',')
```

```
Num Features: %d, 3
Selected Features: %s, [True False False False False False False False False]
Feature Ranking: %s, [1 2 3 4 5 6 7]
<zip_iterator_8>
[True,False,False,False,False,False,False,False,False]
```

	Feature_name	id_score	chi_score	Feature_score
0	id	0.013333333333333	2.006669	2.019999333333333
1	Clump_thickness	0.139333333333333	16.118096	16.257429333333333
2	Uniform_cell_size	0.268666666666667	1.000025	1.268691666666667
3	Uniform_cell_shape	0.164666666666667	1.307511	1.472177666666667
4	Marginal_adhesion	0.096666666666667	0.814726	0.911392666666667
5	Single_epithelial_cell_size	0.047333333333333	1.057334	1.104666666666667
6	Bare_nuclei	0.207666666666667	4.306261	4.513927666666667
7	Bland_chromatin	0.153666666666667	1.546964	1.700630666666667
8	Normal_nucleoli	0.094666666666667	0.989191	1.083857666666667

```
from sklearn import matplotlib
from matplotlib import pyplot
# feature extraction
model = LogisticRegression()
rfe = RFE(model, 3)
fit = rfe.fit(X, Y)
print("Num Features: %d",%fit.n_features_)
print("Selected Features: %s",%fit.support_)
print("Feature Ranking: %s",%fit.ranking_)
# summarize the selection of the attributes
print(zip(range(0,9),
          fit.ranking_,
          dataset.columns))
# show the selected features
features = fit.support_
print(features,dtype=int,sep=',')
```

```
Num Features: %d, 3
Selected Features: %s, [True False False False False False False False False]
Feature Ranking: %s, [1 2 3 4 5 6 7]
<zip_iterator_9>
[True,False,False,False,False,False,False,False,False]
```

	Feature_name	id_score	chi_score	Feature_score
0	id	0.013333333333333	2.006669	2.019999333333333
1	Clump_thickness	0.139333333333333	16.118096	16.257429333333333
2	Uniform_cell_size	0.268666666666667	1.000025	1.268691666666667
3	Uniform_cell_shape	0.164666666666667	1.307511	1.472177666666667
4	Marginal_adhesion	0.096666666666667	0.814726	0.911392666666667
5	Single_epithelial_cell_size	0.047333333333333	1.057334	1.104666666666667
6	Bare_nuclei	0.207666666666667	4.306261	4.513927666666667
7	Bland_chromatin	0.153666666666667	1.546964	1.700630666666667
8	Normal_nucleoli	0.094666666666667	0.989191	1.083857666666667

```
from sklearn import matplotlib
from matplotlib import pyplot
# feature extraction
model = LogisticRegression()
rfe = RFE(model, 3)
fit = rfe.fit(X, Y)
print("Num Features: %d",%fit.n_features_)
print("Selected Features: %s",%fit.support_)
print("Feature Ranking: %s",%fit.ranking_)
# summarize the selection of the attributes
print(zip(range(0,9),
          fit.ranking_,
          dataset.columns))
# show the selected features
features = fit.support_
print(features,dtype=int,sep=',')
```

```
Num Features: %d, 3
Selected Features: %s, [True False False False False False False False False]
Feature Ranking: %s, [1 2 3 4 5 6 7]
<zip_iterator_10>
[True,False,False,False,False,False,False,False,False]
```

	Feature_name	id_score	chi_score	Feature_score
0	id	0.013333333333333	2.006669	2.019999333333333
1	Clump_thickness	0.139333333333333	16.118096	16.257429333333333
2	Uniform_cell_size	0.268666666666667	1.000025	1.268691666666667
3	Uniform_cell_shape	0.164666666666667	1.307511	1.472177666666667
4	Marginal_adhesion	0.096666666666667	0.814726	0.911392666666667
5	Single_epithelial_cell_size	0.047333333333333	1.057334	1.104666666666667
6	Bare_nuclei	0.207666666666667	4.306261	4.513927666666667
7	Bland_chromatin	0.153666666666667	1.546964	1.700630666666667
8	Normal_nucleoli	0.094666666666667	0.989191	1.083857666666667

```
from sklearn import matplotlib
from matplotlib import pyplot
# feature extraction
model = LogisticRegression()
rfe = RFE(model, 3)
fit = rfe.fit(X, Y)
print("Num Features: %d",%fit.n_features_)
print("Selected Features: %s",%fit.support_)
print("Feature Ranking: %s",%fit.ranking_)
# summarize the selection of the attributes
print(zip(range(0,9),
          fit.ranking_,
          dataset.columns))
# show the selected features
features = fit.support_
print(features,dtype=int,sep=',')
```

```
Num Features: %d, 3
Selected Features: %s, [True False False False False False False False False]
Feature Ranking: %s, [1 2 3 4 5 6 7]
<zip_iterator_11>
[True,False,False,False,False,False,False,False,False]
```

	Feature_name	id_score	chi_score	Feature_score
0	id	0.013333333333333	2.006669	2.019999333333333
1	Clump_thickness	0.139333333333333	16.118096	16.257429333333333
2	Uniform_cell_size	0.268666666666667	1.000025	1.268691666666667
3	Uniform_cell_shape	0.164666666666667	1.307511	1.472177666666667
4	Marginal_adhesion	0.096666666666667	0.814726	0.911392666666667
5	Single_epithelial_cell_size	0.047333333333333	1.057334	1.104666666666667
6	Bare_nuclei	0.207666666666667	4.306261	4.513927666666667
7	Bland_chromatin	0.153666666666667	1.546964	1.700630666666667
8	Normal_nucleoli	0.094666666666667	0.989191	1.083857666666667

```
from sklearn import matplotlib
from matplotlib import pyplot
# feature extraction
model = LogisticRegression()
rfe = RFE(model, 3)
fit = rfe.fit(X, Y)
print("Num Features: %d",%fit.n_features_)
print("Selected Features: %s",%fit.support_)
print("Feature Ranking: %s",%fit.ranking_)
# summarize the selection of the attributes
print(zip(range(0,9),
          fit.ranking_,
          dataset.columns))
# show the selected features
features = fit.support_
print(features,dtype=int,sep=',')
```

```
Num Features: %d, 3
Selected Features: %s, [True False False False False False False False False]
Feature Ranking: %s, [1 2 3 4 5 6 7]
<zip_iterator_12>
[True,False,False,False,False,False,False,False,False]
```

	Feature_name	id_score	chi_score	Feature_score
0	id	0.013333333333333	2.006669	2.019999333333333
1	Clump_thickness	0.139333333333333	16.118096	16.257429333333333
2	Uniform_cell_size	0.268666666666667	1.000025	1.268691666666667
3	Uniform_cell_shape	0.164666666666667	1.307511	1.472177666666667
4	Marginal_adhesion	0.096666666666667	0.814726	0.911392666666667
5	Single_epithelial_cell_size	0.047333333333333	1.057334	1.104666666666667
6	Bare_nuclei	0.207666666666667	4.306261	4.513927666666667
7	Bland_chromatin	0.153666666666667	1.546964	1.700630666666667
8	Normal_nucleoli	0.094666666666667	0.989191	1.083857666666667

```
from sklearn import matplotlib
from matplotlib import pyplot
# feature extraction
model = LogisticRegression()
rfe = RFE(model, 3)
fit = rfe.fit(X, Y)
print("Num Features: %d",%fit.n_features_)
print("Selected Features: %s",%fit.support_)
print("Feature Ranking: %s",%fit.ranking_)
# summarize the selection of the attributes
print(zip(range(0,9),
          fit.ranking_,
          dataset.columns))
# show the selected features
features = fit.support_
print(features,dtype=int,sep=',')
```

```
Num Features: %d, 3
Selected Features: %s, [True False False False False False False False False]
Feature Ranking: %s, [1 2 3 4 5 6 7]
<zip_iterator_13>
[True,False,False,False,False,False,False,False,False]
```

	Feature_name	id_score	chi_score	Feature_score
0	id	0.013333333333333	2.006669	2.019999333333333
1	Clump_thickness	0.139333333333333	16.118096	16.257429333333333
2	Uniform_cell_size	0.268666666666667	1.000025	1.268691666666667
3	Uniform_cell_shape	0.164666666666667	1.307511	1.472177666666667
4	Marginal_adhesion	0.096666666666667	0.814726	0.911392666666667
5	Single_epithelial_cell_size	0.047333333333333	1.057334	1.104666666666667
6	Bare_nuclei	0.207666666666667	4.306261	4.513927666666667
7	Bland_chromatin	0.153666666666667	1.546964	1.700630666666667
8	Normal_nucleoli	0.094666666666667	0.989191	1.083857666666667

```
from sklearn import matplotlib
from matplotlib import pyplot
# feature extraction
model = LogisticRegression()
rfe = RFE(model, 3)
fit = rfe.fit(X, Y)
print("Num Features: %d",%fit.n_features_)
print("Selected Features: %s",%fit.support_)
print("Feature Ranking: %s",%fit.ranking_)
# summarize the selection of the attributes
print(zip(range(0,9),
          fit.ranking_,
          dataset.columns))
# show the selected features
features = fit.support_
print(features,dtype=int,sep=',')
```

```
Num Features: %d, 3
Selected Features: %s, [True False False False False False False False False]
Feature Ranking: %s, [1 2 3 4 5 6 7]
<zip_iterator_14>
[True,False,False,False,False,False,False,False,False]
```

	Feature_name	id_score	chi_score	Feature_score
0	id	0.013333333333333	2.006669	2.019999333333333
1	Clump_thickness	0.139333333333333	16.118096	16.257429333333333
2	Uniform_cell_size	0.268666666666667	1.000025	1.268691666666667
3	Uniform_cell_shape	0.164666666666667	1.307511	1.472177666666667
4	Marginal_adhesion	0.096666666666667	0.814726	0.911392666666667
5	Single_epithelial_cell_size	0.047333333333333	1.057334	1.104666666666667
6	Bare_nuclei	0.207666666666667	4.306261	4.513927666666667
7	Bland_chromatin	0.153666666666667	1.546964	1.700630666666667
8	Normal_nucleoli	0.094666666666667	0.989191	1.083857666666667

```
from sklearn import matplotlib
from matplotlib import pyplot
# feature extraction
model = LogisticRegression()
rfe = RFE(model, 3)
fit = rfe.fit(X, Y)
print("Num Features: %d",%fit.n_features_)
print("Selected Features: %s",%fit.support_)
print("Feature Ranking: %s",%fit.ranking_)
# summarize the selection of the attributes
print(zip(range(0,9),
          fit.ranking_,
          dataset.columns))
# show the selected features
features = fit.support_
print(features,dtype=int,sep=',')
```

```
Num Features: %d, 3
Selected Features: %s, [True False False False False False False False False]
Feature Ranking: %s, [1 2 3 4 5 6 7]
<zip_iterator_15>
[True,False,False,False,False,False,False,False,False]
```

	Feature_name	id_score	chi_score	Feature_score
0	id	0.013333333333333	2.006669	2.019999333333333
1	Clump_thickness	0.139333333333333	16.118096	16.257429333333333
2	Uniform_cell_size	0.268666666666667	1.000025	1.268691666666667
3	Uniform_cell_shape	0.164666666666667	1.307511	1.472177666666667
4	Marginal_adhesion	0.096666666666667	0.814726	0.911392666666667
5	Single_epithelial_cell_size	0.047333333333333	1.057334	1.104666666666667
6	Bare_nuclei	0.207666666666667	4.306261	4.513927666666667
7	Bland_chromatin	0.153666666666667	1.546964	1.700630666666667
8	Normal_nucleoli	0.094666666666667	0.989191	1.083857666666667

```
from sklearn import matplotlib
from matplotlib import pyplot
# feature extraction
model = LogisticRegression()
rfe = RFE(model, 3)
fit = rfe.fit(X, Y)
print("Num Features: %d",%fit.n_features_)
print("Selected Features: %s",%fit.support_)
print("Feature Ranking: %s",%fit.ranking_)
# summarize the selection of the attributes
print(zip(range(0,9),
          fit.ranking_,
          dataset.columns))
# show the selected features
features = fit.support_
print(features,dtype=int,sep=',')
```

```
Num Features: %d, 3
Selected Features: %s, [True False False False False False False False False]
Feature Ranking: %s, [1 2 3 4 5 6 7]
<zip_iterator_16>
[True,False,False,False,False,False,False,False,False]
```

	Feature_name	id_score	chi_score	Feature_score
0	id	0.013333333333333	2.006669	2.019999333333333
1	Clump_thickness	0.139333333333333	16.118096	16.257429333333333
2	Uniform_cell_size	0.268666666666667	1.000025	1.268691666666667
3	Uniform_cell_shape	0.164666666666667	1.307511	1.472177666666667
4	Marginal_adhesion	0.096666666666667	0.814726	0.911392666666667
5	Single_epithelial_cell_size	0.047333333333333	1.057334	1.104666666666667
6	Bare_nuclei	0.207666666666667	4.306261	4.513927666666667
7	Bland_chromatin	0.153666666666667	1.546964	1.700630666666667
8	Normal_nucleoli	0.094666666666667	0.989191	1.083857666666667

```
from sklearn import matplotlib
from matplotlib import pyplot
# feature extraction
model = LogisticRegression()
rfe = RFE(model, 3)
fit = rfe.fit(X, Y)
print("Num Features: %d",%fit.n_features_)
print("Selected Features: %s",%fit.support_)
print("Feature Ranking: %s",%fit.ranking_)
# summarize the selection of the attributes
print(zip(range(0,9),
          fit.ranking_,
          dataset.columns))
# show the selected features
features = fit.support_
print(features,dtype=int,sep=',')
```

```
Num Features: %d, 3
Selected Features: %s, [True False False False False False False False False]
Feature Ranking: %s, [1 2 3 4 5 6 7]
<zip_iterator_17>
[True,False,False,False,False,False,False,False,False]
```

	Feature_name	id_score	chi_score	Feature_score
0	id	0.013333333333333	2.006669	2.019999333333333
1	Clump_thickness	0.139333333333333	16.118096	16.257429333333333
2	Uniform_cell_size	0.268666666666667	1.000025	1.268691666666667
3	Uniform_cell_shape	0.164666666666667	1.307511	1.472177666666667
4	Marginal_adhesion	0.096666666666667	0.814726	0.911392666666667
5	Single_epithelial_cell_size	0.047333333333333	1.057334	1.104666666666667
6	Bare_nuclei	0.207666666666667	4.306261	4.513927666666667
7	Bland_chromatin	0.153666666666667	1.546964	1.700630666666667
8	Normal_nucleoli	0.094666666666667	0.989191	1.083857666666667

```
from sklearn import matplotlib
from matplotlib import pyplot
# feature extraction
model = LogisticRegression()
rfe = RFE(model, 3)
fit = rfe.fit(X, Y)
print("Num Features: %d",%fit.n_features_)
print("Selected Features: %s",%fit.support_)
print("Feature Ranking: %s",%fit.ranking_)
# summarize the selection of the attributes
print(zip(range(0,9),
          fit.ranking_,
          dataset.columns))
# show the selected features
features = fit.support_
print(features,dtype=int,sep=',')
```

```
Num Features: %d, 3
Selected Features: %s, [True False False False False False False False False]
Feature Ranking: %s, [1 2 3 4 5 6 7]
<zip_iterator_18>
[True,False,False,False,False,False,False,False,False]
```

	Feature_name	id_score	chi_score	Feature_score
0	id	0.013333333333333	2.006669	2.019999333333333
1	Clump_thickness	0.139333333333333	16.118096	16.257429333333333
2	Uniform_cell_size	0.268666666666667	1.000025	1.268691666666667
3	Uniform_cell_shape	0.164666666666667	1.307511	1.472177666666667
4	Marginal_adhesion	0.096666666666667	0.814726	0.911392666666667
5	Single_epithelial_cell_size	0.047333333333333	1.057334	1.104666666666667
6	Bare_nuclei	0.207666666666667	4.306261	4.513927666666667
7	Bland_chromatin	0.153666666666667	1.546964	1.700630666666667
8	Normal_nucleoli	0.094666666666667	0.989191	1.083857666666667

```
from sklearn import matplotlib
from matplotlib import pyplot
# feature extraction
model = LogisticRegression()
rfe = RFE(model, 3)
fit = rfe.fit(X, Y)
print("Num Features: %d",%fit.n_features_)
print("Selected Features: %s",%fit.support_)
print("Feature Ranking: %s",%fit.ranking_)
# summarize the selection of the attributes
print(zip(range(0,9),
          fit.ranking_,
          dataset.columns))
# show the selected features
features = fit.support_
print(features,dtype=int,sep=',')
```

```
Num Features: %d, 3
Selected Features: %s, [True False False False False False False False False]
Feature Ranking: %s, [1 2 3 4 5 6 7]
<zip_iterator_19>
[True,False,False,False,False,False,False,False,False]
```

	Feature_name	id_score	chi_score	Feature_score
0	id	0.013333333333333	2.006669	2.019999333333333
1	Clump_thickness	0.139333333333333	16.118096	16.257429333333333
2	Uniform_cell_size	0.268666666666667	1.000025	1.268691666666667
3	Uniform_cell_shape	0.164666666666667	1.307511	1.472177666666667
4	Marginal_adhesion	0.096666666666667	0.814726	0.911392666666667
5	Single_epithelial_cell_size	0.047333333333333	1.057334	1.104666666666667
6	Bare_nuclei	0.207666666666667	4.306261	4.513927666666667
7	Bland_chromatin	0.153666666666667	1.546964	1.700630666666667
8	Normal_nucleoli	0.094666666666667	0.989191	1.083857666666667

```
from sklearn import matplotlib
from matplotlib import pyplot
# feature extraction
model = LogisticRegression()
rfe = RFE(model, 3)
fit = rfe.fit(X, Y)
print("Num Features: %d",%fit.n_features_)
print("Selected Features: %s",%fit.support_)
print("Feature Ranking: %s",%fit.ranking_)
# summarize the selection of the attributes
print(zip(range(0,9),
          fit.ranking_,
          dataset.columns))
# show the selected features
features = fit.support_
print(features,dtype=int,sep=',')
```

```
Num Features: %d, 3
Selected Features: %s, [True False False False False False False False False]
Feature Ranking: %s, [1 2 3 4 5 6 7]
<zip_iterator_20>
[True,False,False,False,False,False,False,False,False]
```

	Feature_name	id_score	chi_score	Feature_score
0	id	0.013333333333333	2.006669	2.019999333333333
1	Clump_thickness	0.139333333333333	16.118096	16.257429333333333
2	Uniform_cell_size	0.268666666666667	1.000025	1.268691666666667
3	Uniform_cell_shape	0.164666666666667	1.307511	1.472177666666667
4	Marginal_adhesion	0.096666666666667	0.814726	0.911392666666667
5	Single_epithelial_cell_size	0.047333333333333	1.057334	1.104666666666667
6	Bare_nuclei	0.207666666666667	4.306261	4.513927666666667
7	Bland_chromatin	0.153666666666667	1.546964	1.700630666666667
8	Normal_nucleoli	0.094666666666667	0.989191	1.083857666666667

```
from sklearn import matplotlib
from matplotlib import pyplot
# feature extraction
model = LogisticRegression()
rfe = RFE(model, 3)
fit = rfe.fit(X, Y)
print("Num Features: %d",%fit.n_features_)
print("Selected Features: %s",%fit.support_)
print("Feature Ranking: %s",%fit.ranking_)
# summarize the selection of the attributes
print(zip(range(0,9),
          fit.ranking_,
          dataset.columns))
# show the selected features
features = fit.support_
print(features,dtype=int,sep=',')
```

```
Num Features: %d, 3
Selected Features: %s, [True False False False False False False False False]
Feature Ranking: %s, [1 2 3 4 5 6 7]
<zip_iterator_21>
[True,False,False,False,False,False,False,False,False]
```

	Feature_name	id_score	chi_score	Feature_score
0	id	0.013333333333333	2.006669	2.019999333333333
1	Clump_thickness	0.139333333333333	16.118096	16.257429333333333
2	Uniform_cell_size	0.268666666666667	1.000025	1.268691666666667
3	Uniform_cell_shape	0.164666666666667	1.307511	1.472177666666667
4	Marginal_adhesion	0.096666666666667	0.814726	0.911392666666667
5	Single_epithelial_cell_size	0.047333333333333	1.057334	1.104666666666667
6	Bare_nuclei	0.207666666666667	4.306261	4.513927666666667
7	Bland_chromatin	0.153666666666667	1.546964	1.700630666666667
8	Normal_nucleoli	0.094666666666667	0.989191	1.083857666666667

```
from sklearn import matplotlib
from matplotlib import pyplot
# feature extraction
model = LogisticRegression()
rfe = RFE(model, 3)
fit = rfe.fit(X, Y)
print("Num Features: %d",%fit.n_features_)
print("Selected Features: %s",%fit.support_)
print("Feature Ranking: %s",%fit.ranking_)
# summarize the selection of the attributes
print(zip(range(0,9),
          fit.ranking_,
          dataset.columns))
# show the selected features
features = fit.support_
print(features,dtype=int,sep=',')
```

```
Num Features: %d, 3
Selected Features: %s, [True False False False False False False False False]
Feature Ranking: %s, [1 2 3 4 5 6 7]
<zip_iterator_22>
[True,False,False,False,False,False,False,False,False]
```

	Feature_name	id_score	chi_score	Feature_score
0	id	0.013333333333333	2.006669	2.019999333333333
1	Clump_thickness	0.139333333333333	16.118096	16.257429333333333
2	Uniform_cell_size	0.268666666666667	1.000025	1.268691666666667
3	Uniform_cell_shape	0.164666666666667	1.307511	1.472177666666667
4	Marginal_adhesion	0.096666666666667	0.814726	0.911392666666667
5	Single_epithelial_cell_size	0.047333333333333	1.057334	1.104666666666667
6	Bare_nuclei	0.207666666666667	4.306261	4.513927666666667
7	Bland_chromatin	0.153666666666667	1.546964	1.700630666666667
8	Normal_nucleoli	0.094666666666667	0.989191	1.083857666666667

```
from sklearn import matplotlib
from matplotlib import pyplot
# feature extraction
model = LogisticRegression()
rfe = RFE(model, 3)
fit = rfe.fit(X, Y)
print("Num Features: %d",%fit.n_features_)
print("Selected Features: %s",%fit.support_)
print("Feature Ranking: %s",%fit.ranking_)
# summarize the selection of the attributes
print(zip(range(0,9),
          fit.ranking_,
          dataset.columns))
# show the selected features
features = fit.support_
print(features,dtype=int,sep=',')
```

```
Num Features: %d, 3
Selected Features: %s, [True False False False False False False False False]
Feature Ranking: %s, [1 2 3 4 5 6 7]
<zip_iterator_23>
[True,False,False,False,False,False,False,False,False]
```

	Feature_name	id_score	chi_score	Feature_score
0	id	0.013333333333333	2.006669	2.019999333333333