In [109…
```
import pandas as pd
import numpy as np
from numpy.random import seed
```

In [110…
```
import warnings
warnings.filterwarnings("ignore")
```

In [111…
```
n = 900
```

In [112…
```
seed(100)
```

In [113…
```
df1 = pd.read_csv("./DATA/default_of_credit_card_clients.csv")
df1.head()
```

Out[113…

	Unnamed: 0	ID	LIMIT_BAL	SEX	EDUCATION	MARRIAGE	AGE	PAY_1	PAY_2	PAY_3	...	BILL_AMT4	BILL_AMT5	BILL_AMT6	PAY_AMT1	PAY_AMT2	PAY_AMT3	P
0	0	1	20000	2	2	1	24	2	2	-1	...	0	0	0	0	689	0	
1	1	2	120000	2	2	2	26	-1	2	0	...	3272	3455	3261	0	1000	1000	
2	2	3	90000	2	2	2	34	0	0	0	...	14331	14948	15549	1518	1500	1000	
3	3	4	50000	2	2	1	37	0	0	0	...	28314	28959	29547	2000	2019	1200	
4	4	5	50000	1	2	1	57	-1	0	-1	...	20940	19146	19131	2000	36681	10000	

5 rows × 26 columns

In [114…
```
x = list(np.arange(30000))
idx = list(np.random.choice(x,n))
df = df1.loc[idx,:]
df.shape
```

Out[114…
```
(900, 26)
```

In [115…
```
cols = df.columns.tolist()
cols
```

Out[115…
```
['Unnamed: 0',
 'ID',
 'LIMIT_BAL',
 'SEX',
 'EDUCATION',
 'MARRIAGE',
 'AGE',
 'PAY_1',
 'PAY_2',
 'PAY_3',
 'PAY_4',
 'PAY_5',
 'PAY_6',
 'BILL_AMT1',
 'BILL_AMT2',
 'BILL_AMT3',
 'BILL_AMT4',
 'BILL_AMT5',
 'BILL_AMT6',
 'PAY_AMT1',
 'PAY_AMT2',
 'PAY_AMT3',
 'PAY_AMT4',
 'PAY_AMT5',
 'PAY_AMT6',
 'DPNM']
```

In [116…
```
del cols[0]
del cols[0]
del cols[-1]
cols
```

Out[116…
```
['LIMIT_BAL',
 'SEX',
 'EDUCATION',
 'MARRIAGE',
 'AGE',
 'PAY_1',
 'PAY_2',
 'PAY_3',
 'PAY_4',
 'PAY_5',
 'PAY_6',
 'BILL_AMT1',
 'BILL_AMT2',
 'BILL_AMT3',
 'BILL_AMT4',
 'BILL_AMT5',
 'BILL_AMT6',
 'PAY_AMT1',
 'PAY_AMT2',
 'PAY_AMT3',
 'PAY_AMT4',
 'PAY_AMT5',
 'PAY_AMT6']
```

In [117…
```
len(df)
```

Out[117…
```
900
```

In [118…
```
from sklearn import preprocessing
from sklearn.model_selection import train_test_split, GridSearchCV
data = df[cols].values
X = preprocessing.StandardScaler().fit(data).transform(data)
X.shape
```

Out[118…
```
(900, 23)
```

In [119…
```
y = df['DPNM'].values
y.shape
```

Out[119…
```
(900,)
```

In [120…
```
x_tr, x_t, y_tr, y_t = train_test_split(X, y, test_size=0.3, random_state=100)
x_tr.shape, x_t.shape, y_tr.shape, y_t.shape
```

Out[120…
```
((630, 23), (270, 23), (630,), (270,))
```

In [121…
```
from sklearn import svm
from sklearn.svm import SVC
```

In [122…
```
clf = GridSearchCV(svm.SVC(gamma='auto'), {'C':[1,10,20], 'kernel':['rbf','linear']}, cv=5, return_train_score=False)
```

In [123…
```
clf.fit(x_tr, y_tr)
res = pd.DataFrame(clf.cv_results_)
```

In [124…
```
res.columns
```

Out[124…
```
Index(['mean_fit_time', 'std_fit_time', 'mean_score_time', 'std_score_time',
       'param_C', 'param_kernel', 'params', 'split0_test_score',
       'split1_test_score', 'split2_test_score', 'split3_test_score',
       'split4_test_score', 'mean_test_score', 'std_test_score',
       'rank_test_score'],
      dtype='object')
```

In [125…
```
res[ ['param_C', 'param_kernel', 'mean_test_score'] ]
```

Out[125…

	param_C	param_kernel	mean_test_score
0	1	rbf	0.807937
1	1	linear	0.796825
2	10	rbf	0.807937
3	10	linear	0.796825
4	20	rbf	0.790476
5	20	linear	0.796825

In [126…
```
res['mean_test_score'].max()
```

Out[126…
```
0.807936507936508
```