

In [1]:

```

from sklearn.datasets import load_breast_cancer
from sklearn.metrics import classification_report, confusion_matrix
from sklearn.datasets import load_breast_cancer
from sklearn.svm import SVC
from sklearn.model_selection import GridSearchCV
from sklearn.model_selection import train_test_split

```

In [2]:

```

dataset = load_breast_cancer()
X=dataset.data
Y=dataset.target
X_train, X_test, y_train, y_test = train_test_split(
    X,Y,test_size = 0.30, random_state = 101)

```

In [3]:

```

model = SVC()
model.fit(X_train, y_train)
predictions = model.predict(X_test)
print(classification_report(y_test, predictions))

```

	precision	recall	f1-score	support
0	0.95	0.85	0.90	66
1	0.91	0.97	0.94	105
accuracy			0.92	171
macro avg	0.93	0.91	0.92	171
weighted avg	0.93	0.92	0.92	171

In [5]:

```

param_grid = {'C': [0.1, 1, 10, 100],
              'gamma': [1, 0.1, 0.01, 0.001, 0.0001],
              'gamma':['scale', 'auto'],
              'kernel': ['linear']}

grid = GridSearchCV(SVC(), param_grid, refit = True, verbose = 3,n_jobs=-1)

grid.fit(X_train, y_train)

print(grid.best_params_)

```

Fitting 5 folds for each of 8 candidates, totalling 40 fits  
{'C': 100, 'gamma': 'scale', 'kernel': 'linear'}

In [6]:

```
grid_predictions = grid.predict(X_test)
print(classification_report(y_test, grid_predictions))
```

	precision	recall	f1-score	support
0	0.95	0.89	0.92	66
1	0.94	0.97	0.95	105
accuracy			0.94	171
macro avg	0.94	0.93	0.94	171
weighted avg	0.94	0.94	0.94	171

In [22]:

```
from sklearn.pipeline import make_pipeline
from sklearn.model_selection import RandomizedSearchCV
from sklearn.preprocessing import StandardScaler
import scipy as sc

param_distributions = [{ 'svc__C': sc.stats.expon(scale=100),
                        'svc__gamma': sc.stats.expon(scale=.1),
                        'svc__kernel': ['rbf'] },
                      { 'svc__C': sc.stats.expon(scale=100),
                        'svc__kernel': ['linear'] }]

rs = RandomizedSearchCV(estimator=make_pipeline(StandardScaler(), SVC()),
                        param_distributions=param_distributions,
                        cv=10, scoring='accuracy', refit=True, n_jobs=1,
                        random_state=1)

rs.fit(X_train, y_train)
print(rs.best_params_)
```

```
{'svc__C': 37.581919839690876, 'svc__gamma': 0.0743489684087931, 'svc__kernel': 'rbf'}
```

In [23]:

```
rs_predictions = rs.predict(X_test)
print(classification_report(y_test, rs_predictions))
```

	precision	recall	f1-score	support
0	0.96	0.97	0.96	66
1	0.98	0.97	0.98	105
accuracy			0.97	171
macro avg	0.97	0.97	0.97	171
weighted avg	0.97	0.97	0.97	171

In [ ]: