Based on the program developed in the lecture, implement a nonlinear Kernel analysis based on SVM for (a) polynomial kernels (poly) of at least 2 different degrees of your choice, and (b) for radial base functions (rbf). Apply the analysis to the breast cancer database, or a database of your choice. Study the classification performance in terms of suitable measures of your choice (e.g., accuracy, precision, recall, f1, k-folds) as a function of the hyperparameters $\gamma$ (gamma) and degree (for polynomials), and gamma and penalty C for rbf kernels. It is on you if you want to use GridSearch, RandomSearch, hyperopt, or just to try a few combinations of the hyperparameters. Hint: Hyperopt may require you to install or update some libraries outside the Jupyter notebook. The program must end with the output ”Hyper, hyper!”.

# 0. Imports, global variables/constants, and datasets

In [1]:

```
from sklearn import svm

import pandas as pd

from sklearn.metrics import precision_recall_fscore_support

import numpy as np

# Load datasets
from sklearn import datasets
cancer = datasets.load_breast_cancer()
print("Features: ", cancer.feature_names)
print("Labels: ", cancer.target_names)
print(cancer.data.shape)
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(
    cancer.data,
    cancer.target,
    test_size=0.3,
    random_state=77
)

import warnings
warnings.filterwarnings("ignore")
```

```
Features:  ['mean radius' 'mean texture' 'mean perimeter' 'mean area'
 'mean smoothness' 'mean compactness' 'mean concavity'
 'mean concave points' 'mean symmetry' 'mean fractal dimension'
 'radius error' 'texture error' 'perimeter error' 'area error'
 'smoothness error' 'compactness error' 'concavity error'
 'concave points error' 'symmetry error' 'fractal dimension error'
 'worst radius' 'worst texture' 'worst perimeter' 'worst area'
 'worst smoothness' 'worst compactness' 'worst concavity'
 'worst concave points' 'worst symmetry' 'worst fractal dimension']
Labels:  ['malignant' 'benign']
(569, 30)
```

# 1. (a) Nonlinear Kernel analysis based on SVM for polynomial kernels, 2 degrees

```
In [2]: clf_poly = svm.SVC(
            kernel='poly',
            degree=2,
            gamma='auto'
        )
```

# 2. (b) Nonlinear Kernel analysis based on SVM for radial base functions

```
In [3]: clf_rbf = svm.SVC(kernel='rbf')
```

# 3. Apply analysis to the breast cancer database (or another database)

```
In [4]: clf_poly.fit(X_train, y_train)
        clf_rbf.fit(X_train, y_train)
```

```
Out[4]: SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
            decision_function_shape='ovr', degree=3, gamma='auto_deprecated',
            kernel='rbf', max_iter=-1, probability=False, random_state=None,
            shrinking=True, tol=0.001, verbose=False)
```

# 4. Study classification performance (accuracy, precision, recall, f1, k-folds) as a function of the hyperparameters $\gamma$ and degree (for polynomials), and $\gamma$ and penalty $C$ for rbf kernels.

I decided to try different parameters and plot the test scores.

In [5]:

```
def get_accuracy_precision_recall_f_score(clf_instance, X_train, y_tr
ain, X_test, y_test):
    clf_instance.fit(
        X_train,
        y_train
    )

    y_pred = clf_instance.predict(X_test)

    accuracy = clf_instance.score(X_test, y_test)

    precision, recall, f_score, _ = precision_recall_fscore_support(y
_test, y_pred)

    return accuracy, precision, recall, f_score

def analyze_svm(hyperparameters, X_train, y_train, X_test, y_test, po
lynomials=True):
    df_results = pd.DataFrame.from_dict(
        {
            'accuracy': [],
            'precision': [],
            'recall': [],
            'f_score': [],
            'gamma': [],
            'degree': [],
            'penalty_c': []
        }
    )

    if polynomials:
        for gamma in hyperparameters['gamma']:
            for degree in hyperparameters['degree']:

                clf_poly = svm.SVC(
                    kernel='poly',
                    degree=degree,
                    gamma=gamma
                )

                accuracy, precision, recall, f_score = get_accuracy_p
recision_recall_f_score(
                    clf_poly,
                    X_train,
                    y_train,
                    X_test,
                    y_test
                )

                df_intermediate = pd.DataFrame.from_dict(
                    {
                        'accuracy': accuracy,
                        'precision': precision,
                        'recall': precision,
                        'f_score': f_score,
                        'gamma': gamma,
                        'degree': degree,
                        'penalty_c': None
                    }
                )
                df_results = df_results.append(
                    df_intermediate,
                    ignore_index=True
                )

    else:
        for gamma in hyperparameters['gamma']:
            for penalty_c in hyperparameters['penalty_c']:

                clf_rbf = svm.SVC(
                    kernel='rbf',
                    gamma=gamma,
                    C=penalty_c
                )

                accuracy, precision, recall, f_score = get_accuracy_p
recision_recall_f_score(
                    clf_rbf,
                    X_train,
                    y_train,
                    X_test,
                    y_test
                )

                df_intermediate = pd.DataFrame.from_dict(
                    {
                        'accuracy': accuracy,
                        'precision': precision,
                        'recall': precision,
                        'f_score': f_score,
                        'gamma': gamma,
                        'degree': None,
                        'penalty_c': penalty_c
                    }
                )
                df_results = df_results.append(
                    df_intermediate,
                    ignore_index=True
                )

    return df_results

list_gamma = ['scale', 'auto', 'auto_deprecated']
#list_gamma = [0.1, 1]#, 'auto', 'auto_deprecated']

hyperparameters = {
    'gamma': list_gamma,
    'degree': [0, 1, 2]#, 3]
}
df_results = analyze_svm(hyperparameters, X_train, y_train, X_test, y
_test, polynomials=True)
print('***POLY***')
print('Sorted by accuracy')
print(df_results.sort_values(by='accuracy', ascending=False))
print('\nSorted by precision')
print(df_results.sort_values(by='precision', ascending=False))
print('\nSorted by recall')
print(df_results.sort_values(by='recall', ascending=False))
print('\nSorted by f-score')
print(df_results.sort_values(by='f_score', ascending=False))

hyperparameters = {
    'gamma': list_gamma,
    'penalty_c': [0.1, 1, 10]#, 100]
}
df_results = analyze_svm(hyperparameters, X_train, y_train, X_test, y
_test, polynomials=False)
print('\n\n\n')
print('***RBF***')
print('Sorted by accuracy')
print(df_results.sort_values(by='accuracy', ascending=False))
print('\nSorted by precision')
print(df_results.sort_values(by='precision', ascending=False))
print('\nSorted by recall')
print(df_results.sort_values(by='recall', ascending=False))
print('\nSorted by f-score')
print(df_results.sort_values(by='f_score', ascending=False))
```

```
***POLY***
Sorted by accuracy
    accuracy  precision    recall   f_score            gamma  degree
penalty_c
17  0.970760   0.964286  0.964286  0.977376  auto_deprecated     2.0
     None
16  0.970760   0.983051  0.983051  0.958678  auto_deprecated     2.0
     None
11  0.970760   0.964286  0.964286  0.977376             auto     2.0
     None
10  0.970760   0.983051  0.983051  0.958678             auto     2.0
     None
8   0.964912   0.966667  0.966667  0.950820             auto     1.0
     None
15  0.964912   0.963964  0.963964  0.972727  auto_deprecated     1.0
     None
14  0.964912   0.966667  0.966667  0.950820  auto_deprecated     1.0
     None
9   0.964912   0.963964  0.963964  0.972727             auto     1.0
     None
5   0.906433   0.878049  0.878049  0.931034            scale     2.0
     None
4   0.906433   0.979167  0.979167  0.854545            scale     2.0
     None
3   0.900585   0.870968  0.870968  0.927039            scale     1.0
     None
2   0.900585   0.978723  0.978723  0.844037            scale     1.0
     None
7   0.637427   0.637427  0.637427  0.778571             auto     0.0
     None
1   0.637427   0.637427  0.637427  0.778571            scale     0.0
     None
6   0.637427   0.000000  0.000000  0.000000             auto     0.0
     None
12  0.637427   0.000000  0.000000  0.000000  auto_deprecated     0.0
     None
13  0.637427   0.637427  0.637427  0.778571  auto_deprecated     0.0
     None
0   0.637427   0.000000  0.000000  0.000000            scale     0.0
     None

Sorted by precision
    accuracy  precision    recall   f_score            gamma  degree
penalty_c
10  0.970760   0.983051  0.983051  0.958678             auto     2.0
     None
16  0.970760   0.983051  0.983051  0.958678  auto_deprecated     2.0
     None
4   0.906433   0.979167  0.979167  0.854545            scale     2.0
     None
2   0.900585   0.978723  0.978723  0.844037            scale     1.0
     None
14  0.964912   0.966667  0.966667  0.950820  auto_deprecated     1.0
     None
8   0.964912   0.966667  0.966667  0.950820             auto     1.0
     None
17  0.970760   0.964286  0.964286  0.977376  auto_deprecated     2.0
```

```
     None
11  0.970760   0.964286  0.964286  0.977376             auto     2.0
     None
15  0.964912   0.963964  0.963964  0.972727  auto_deprecated     1.0
     None
9   0.964912   0.963964  0.963964  0.972727             auto     1.0
     None
5   0.906433   0.878049  0.878049  0.931034            scale     2.0
     None
3   0.900585   0.870968  0.870968  0.927039            scale     1.0
     None
1   0.637427   0.637427  0.637427  0.778571            scale     0.0
     None
7   0.637427   0.637427  0.637427  0.778571             auto     0.0
     None
13  0.637427   0.637427  0.637427  0.778571  auto_deprecated     0.0
     None
12  0.637427   0.000000  0.000000  0.000000  auto_deprecated     0.0
     None
6   0.637427   0.000000  0.000000  0.000000             auto     0.0
     None
0   0.637427   0.000000  0.000000  0.000000            scale     0.0
     None

Sorted by recall
    accuracy  precision    recall   f_score            gamma  degree
penalty_c
10  0.970760   0.983051  0.983051  0.958678             auto     2.0
     None
16  0.970760   0.983051  0.983051  0.958678  auto_deprecated     2.0
     None
4   0.906433   0.979167  0.979167  0.854545            scale     2.0
     None
2   0.900585   0.978723  0.978723  0.844037            scale     1.0
     None
14  0.964912   0.966667  0.966667  0.950820  auto_deprecated     1.0
     None
8   0.964912   0.966667  0.966667  0.950820             auto     1.0
     None
17  0.970760   0.964286  0.964286  0.977376  auto_deprecated     2.0
     None
11  0.970760   0.964286  0.964286  0.977376             auto     2.0
     None
15  0.964912   0.963964  0.963964  0.972727  auto_deprecated     1.0
     None
9   0.964912   0.963964  0.963964  0.972727             auto     1.0
     None
5   0.906433   0.878049  0.878049  0.931034            scale     2.0
     None
3   0.900585   0.870968  0.870968  0.927039            scale     1.0
     None
1   0.637427   0.637427  0.637427  0.778571            scale     0.0
     None
7   0.637427   0.637427  0.637427  0.778571             auto     0.0
     None
13  0.637427   0.637427  0.637427  0.778571  auto_deprecated     0.0
     None
```

```
12  0.637427   0.000000  0.000000  0.000000  auto_deprecated     0.0
     None
6   0.637427   0.000000  0.000000  0.000000             auto     0.0
     None
0   0.637427   0.000000  0.000000  0.000000            scale     0.0
     None

Sorted by f-score
    accuracy  precision    recall   f_score            gamma  degree
penalty_c
17  0.970760   0.964286  0.964286  0.977376  auto_deprecated     2.0
     None
11  0.970760   0.964286  0.964286  0.977376             auto     2.0
     None
15  0.964912   0.963964  0.963964  0.972727  auto_deprecated     1.0
     None
9   0.964912   0.963964  0.963964  0.972727             auto     1.0
     None
10  0.970760   0.983051  0.983051  0.958678             auto     2.0
     None
16  0.970760   0.983051  0.983051  0.958678  auto_deprecated     2.0
     None
8   0.964912   0.966667  0.966667  0.950820             auto     1.0
     None
14  0.964912   0.966667  0.966667  0.950820  auto_deprecated     1.0
     None
5   0.906433   0.878049  0.878049  0.931034            scale     2.0
     None
3   0.900585   0.870968  0.870968  0.927039            scale     1.0
     None
4   0.906433   0.979167  0.979167  0.854545            scale     2.0
     None
2   0.900585   0.978723  0.978723  0.844037            scale     1.0
     None
7   0.637427   0.637427  0.637427  0.778571             auto     0.0
     None
1   0.637427   0.637427  0.637427  0.778571            scale     0.0
     None
13  0.637427   0.637427  0.637427  0.778571  auto_deprecated     0.0
     None
6   0.637427   0.000000  0.000000  0.000000             auto     0.0
     None
12  0.637427   0.000000  0.000000  0.000000  auto_deprecated     0.0
     None
0   0.637427   0.000000  0.000000  0.000000            scale     0.0
     None




***RBF***
Sorted by accuracy
    accuracy  precision    recall   f_score            gamma degree
penalty_c
4   0.941520   0.964286  0.964286  0.915254            scale   None
     10.0
5   0.941520   0.930435  0.930435  0.955357            scale   None
```

```
     10.0
2   0.912281   0.979592  0.979592  0.864865            scale   None
      1.0
3   0.912281   0.885246  0.885246  0.935065            scale   None
      1.0
0   0.894737   0.978261  0.978261  0.833333            scale   None
      0.1
1   0.894737   0.864000  0.864000  0.923077            scale   None
      0.1
12  0.637427   0.000000  0.000000  0.000000  auto_deprecated   None
      0.1
16  0.637427   0.000000  0.000000  0.000000  auto_deprecated   None
     10.0
15  0.637427   0.637427  0.637427  0.778571  auto_deprecated   None
      1.0
14  0.637427   0.000000  0.000000  0.000000  auto_deprecated   None
      1.0
13  0.637427   0.637427  0.637427  0.778571  auto_deprecated   None
      0.1
9   0.637427   0.637427  0.637427  0.778571             auto   None
      1.0
11  0.637427   0.637427  0.637427  0.778571             auto   None
     10.0
10  0.637427   0.000000  0.000000  0.000000             auto   None
     10.0
8   0.637427   0.000000  0.000000  0.000000             auto   None
      1.0
7   0.637427   0.637427  0.637427  0.778571             auto   None
      0.1
6   0.637427   0.000000  0.000000  0.000000             auto   None
      0.1
17  0.637427   0.637427  0.637427  0.778571  auto_deprecated   None
     10.0

Sorted by precision
    accuracy  precision    recall   f_score            gamma degree
penalty_c
2   0.912281   0.979592  0.979592  0.864865            scale   None
      1.0
0   0.894737   0.978261  0.978261  0.833333            scale   None
      0.1
4   0.941520   0.964286  0.964286  0.915254            scale   None
     10.0
5   0.941520   0.930435  0.930435  0.955357            scale   None
     10.0
3   0.912281   0.885246  0.885246  0.935065            scale   None
      1.0
1   0.894737   0.864000  0.864000  0.923077            scale   None
      0.1
15  0.637427   0.637427  0.637427  0.778571  auto_deprecated   None
      1.0
13  0.637427   0.637427  0.637427  0.778571  auto_deprecated   None
      0.1
11  0.637427   0.637427  0.637427  0.778571             auto   None
     10.0
9   0.637427   0.637427  0.637427  0.778571             auto   None
      1.0
```

```
7   0.637427   0.637427  0.637427  0.778571             auto   None
      0.1
17  0.637427   0.637427  0.637427  0.778571  auto_deprecated   None
     10.0
10  0.637427   0.000000  0.000000  0.000000             auto   None
     10.0
8   0.637427   0.000000  0.000000  0.000000             auto   None
      1.0
12  0.637427   0.000000  0.000000  0.000000  auto_deprecated   None
      0.1
14  0.637427   0.000000  0.000000  0.000000  auto_deprecated   None
      1.0
6   0.637427   0.000000  0.000000  0.000000             auto   None
      0.1
16  0.637427   0.000000  0.000000  0.000000  auto_deprecated   None
     10.0

Sorted by recall
    accuracy  precision    recall   f_score            gamma degree
penalty_c
2   0.912281   0.979592  0.979592  0.864865            scale   None
      1.0
0   0.894737   0.978261  0.978261  0.833333            scale   None
      0.1
4   0.941520   0.964286  0.964286  0.915254            scale   None
     10.0
5   0.941520   0.930435  0.930435  0.955357            scale   None
     10.0
3   0.912281   0.885246  0.885246  0.935065            scale   None
      1.0
1   0.894737   0.864000  0.864000  0.923077            scale   None
      0.1
15  0.637427   0.637427  0.637427  0.778571  auto_deprecated   None
      1.0
13  0.637427   0.637427  0.637427  0.778571  auto_deprecated   None
      0.1
11  0.637427   0.637427  0.637427  0.778571             auto   None
     10.0
9   0.637427   0.637427  0.637427  0.778571             auto   None
      1.0
7   0.637427   0.637427  0.637427  0.778571             auto   None
      0.1
17  0.637427   0.637427  0.637427  0.778571  auto_deprecated   None
     10.0
10  0.637427   0.000000  0.000000  0.000000             auto   None
     10.0
8   0.637427   0.000000  0.000000  0.000000             auto   None
      1.0
12  0.637427   0.000000  0.000000  0.000000  auto_deprecated   None
      0.1
14  0.637427   0.000000  0.000000  0.000000  auto_deprecated   None
      1.0
6   0.637427   0.000000  0.000000  0.000000             auto   None
      0.1
16  0.637427   0.000000  0.000000  0.000000  auto_deprecated   None
     10.0
```

```
Sorted by f-score
    accuracy  precision    recall   f_score            gamma degree
penalty_c
5   0.941520   0.930435  0.930435  0.955357            scale   None
     10.0
3   0.912281   0.885246  0.885246  0.935065            scale   None
      1.0
1   0.894737   0.864000  0.864000  0.923077            scale   None
      0.1
4   0.941520   0.964286  0.964286  0.915254            scale   None
     10.0
2   0.912281   0.979592  0.979592  0.864865            scale   None
      1.0
0   0.894737   0.978261  0.978261  0.833333            scale   None
      0.1
15  0.637427   0.637427  0.637427  0.778571  auto_deprecated   None
      1.0
13  0.637427   0.637427  0.637427  0.778571  auto_deprecated   None
      0.1
11  0.637427   0.637427  0.637427  0.778571             auto   None
     10.0
9   0.637427   0.637427  0.637427  0.778571             auto   None
      1.0
7   0.637427   0.637427  0.637427  0.778571             auto   None
      0.1
17  0.637427   0.637427  0.637427  0.778571  auto_deprecated   None
     10.0
10  0.637427   0.000000  0.000000  0.000000             auto   None
     10.0
8   0.637427   0.000000  0.000000  0.000000             auto   None
      1.0
12  0.637427   0.000000  0.000000  0.000000  auto_deprecated   None
      0.1
14  0.637427   0.000000  0.000000  0.000000  auto_deprecated   None
      1.0
6   0.637427   0.000000  0.000000  0.000000             auto   None
      0.1
16  0.637427   0.000000  0.000000  0.000000  auto_deprecated   None
     10.0
```

## 5. Hyper, hyper!

In [6]:

```
print('Hyper, hyper!')
```

```
Hyper, hyper!
```