Université de Montpellier

Apprentissage statistique

TP 3: SVM

Élève :
Labourail Célia

Encadrante :
B.Bensaid

October 3, 2025

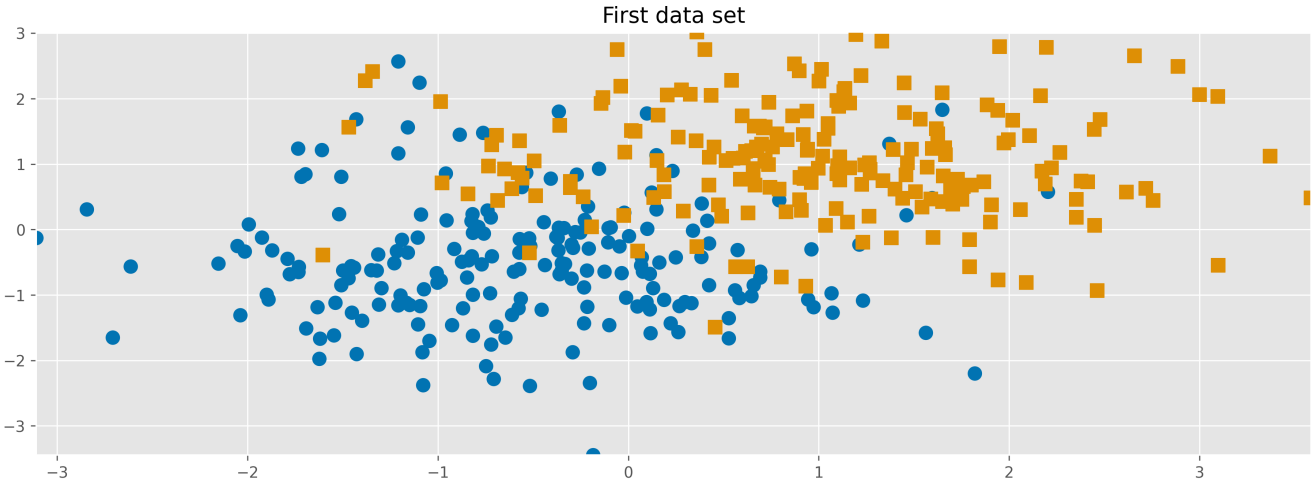
Introduction

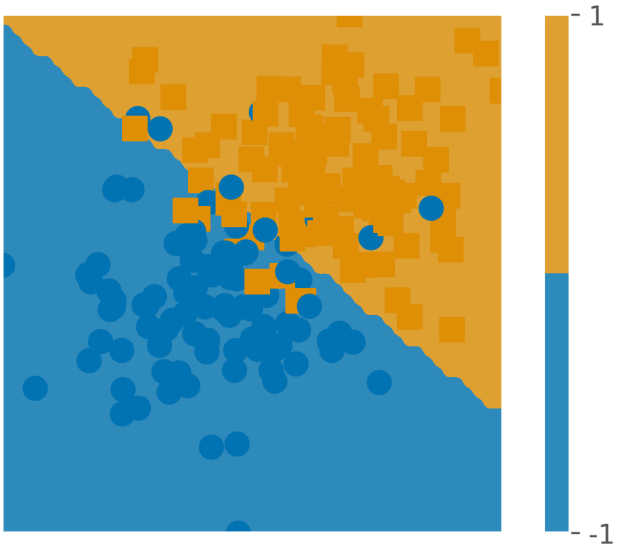
```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.svm import SVC

from svm_source import *
from sklearn import svm
from sklearn import datasets
from sklearn.utils import shuffle
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split, GridSearchCV
from sklearn.datasets import fetch_lfw_people
from sklearn.decomposition import PCA
from time import time

scaler = StandardScaler()

import warnings
warnings.filterwarnings("ignore")

plt.style.use('ggplot')
```

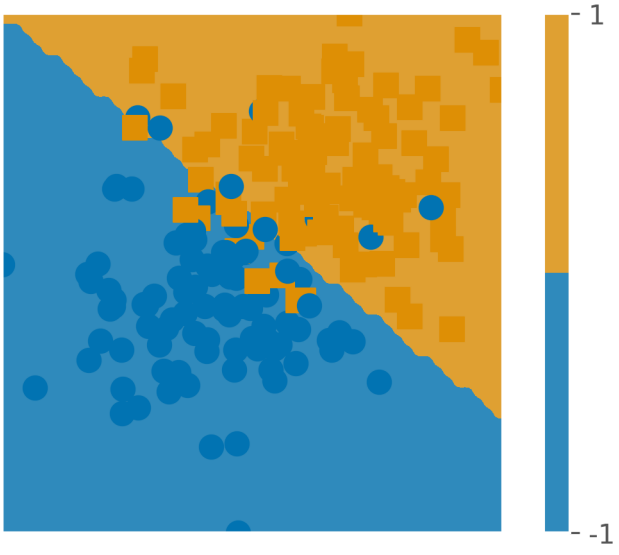
```
##############################################################################
#               Toy dataset : 2 gaussians
##############################################################################

n1 = 200
n2 = 200
mu1 = [1., 1.]
mu2 = [-1./2, -1./2]
sigma1 = [0.9, 0.9]
sigma2 = [0.9, 0.9]
X1, y1 = rand_bi_gauss(n1, n2, mu1, mu2, sigma1, sigma2)

plt.show()
plt.close("all")
plt.ion()
plt.figure(1, figsize=(15, 5))
plt.title('First data set')
plot_2d(X1, y1)

X_train = X1[::2]
Y_train = y1[::2].astype(int)
X_test = X1[1::2]
Y_test = y1[1::2].astype(int)
```

```
# fit the model with linear kernel
clf = SVC(kernel='linear')
clf.fit(X_train, Y_train)

# predict labels for the test data base
y_pred = clf.predict(X_test)

# check your score
score = clf.score(X_test, Y_test)
print('Score : %s' % score)

# display the frontiere
def f(xx):
    """Classifier: needed to avoid warning due to shape issues"""
    return clf.predict(xx.reshape(1, -1))

plt.figure()
frontiere(f, X_train, Y_train, w=None, step=50, alpha_choice=1)

# Same procedure but with a grid search
parameters = {'kernel': ['linear'], 'C': list(np.linspace(0.001, 3, 21))}
clf2 = SVC()
clf_grid = GridSearchCV(clf2, parameters, n_jobs=1)
clf_grid.fit(X_train, Y_train)

# check your score
print(clf_grid.best_params_)
print('Score : %s' % clf_grid.score(X_test, Y_test))

def f_grid(xx):
    """Classifier: needed to avoid warning due to shape issues"""
    return clf_grid.predict(xx.reshape(1, -1))

# display the frontiere
plt.figure()
frontiere(f_grid, X_train, Y_train, w=None, step=50, alpha_choice=1)
```

```
Score : 0.875
{'C': np.float64(0.3009), 'kernel': 'linear'}
Score : 0.875
```


First data set
3
2
1
0
−1
−2
−3
−3
−2
−1
0
1
2
3


1
-1


1
-1