

NAMA : Wandu Yusuf Kurniawan
NIM : 1301218601
KELAS : IFX-45-GAB

Perhatikan pemrograman SVM berikut ini:

```
[ ] import pickle
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline

from cvxopt import matrix, solvers
```

```
#DATA
COLORS = ['red', 'blue']

x_train = np.array([[0.20, 0.20],[-0.10, 0.50],[0.30, 0.20],
                    [0.10, 0.50],[0.30, 0.10],[0.00, 0.30],
                    [0.40, 0.50],[-0.30, -0.30],[0.10, -0.40],
                    [-0.10, -0.30],[0.00, -0.50],[-0.30, -0.10],
                    [-0.50, -0.40],[-0.40, -0.20]])

y_train = np.array([1., 1., 1., 1., 1., 1., 1., -1., -1., -1., -1., -1., -1., -1.])

x_test = np.array([[-0.40, 0.40],[ -0.50, 0.50],[ -0.40, 0.50],[ -0.30, 0.40],
                   [ 0.50, 0.00],[ 0.40, 0.10],[ 0.50, 0.40],[ 0.10, 0.30],
                   [ 0.20, 0.30],[ -0.30, 0.50],[ 0.30, 0.40],[ 0.10, 0.20],
                   [ 0.50, 0.20],[ -0.10, 0.30],[ 0.00, 0.50],[ 0.50, -0.50],
                   [ 0.50, -0.40],[ -0.50, 0.10],[ -0.10, -0.20],[ 0.00, -0.30],
                   [ 0.30, -0.30],[ -0.40, -0.10],[ 0.30, -0.40],[ 0.10, -0.30],
                   [ 0.10, -0.50],[ -0.20, -0.20],[ -0.30, -0.20],[ -0.50, -0.10],
                   [-0.40, -0.40],[ -0.50, -0.50],[ -0.20, -0.50],[ -0.20, -0.40]])

y_test = [1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., -1., -1., -1.,
          -1., -1., -1., -1., -1., -1., -1., -1., -1., -1., -1., -1.]

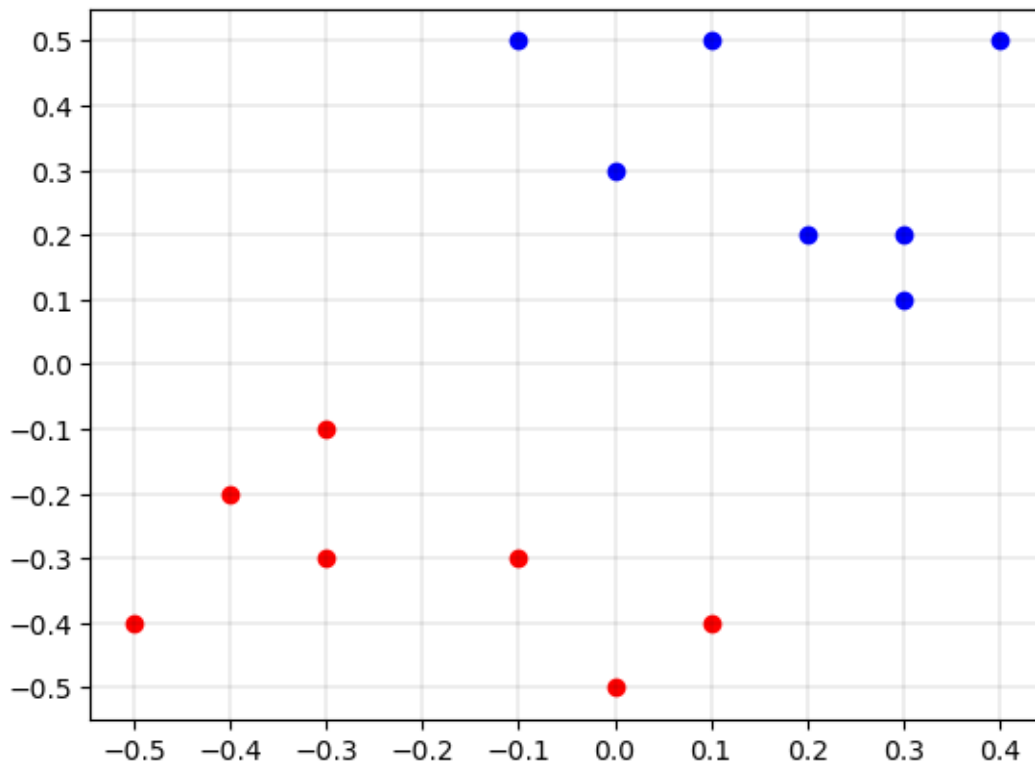
n_train, D = x_train.shape
n_test,_ = x_test.shape
```

```
[ ] def scatter_plot(ax, x, y):
    unique = np.unique(y)
    for li in range(len(unique)):
        x_s = x[y==unique[li]]
        ax.set_xticks(np.arange(-1, 1, 0.1))
        ax.set_yticks(np.arange(-1, 1., 0.1))
        ax.scatter(x_s[:,0],x_s[:,1],c=COLORS[li])
        ax.grid(b=True, which='both', color='k', linestyle='-',linewidth=.1)
```

```
fig, ax = plt.subplots()
scatter_plot(ax, x_train, y_train)
plt.show()
```

Pertanyaan 1 (point 20):

Apa yang akan dikeluarkan setelah baris ini di run?





```
K = y_train[:,None]*x_train
Kkt = K.dot(K.T)
P = matrix(Kkt)
q = matrix(-np.ones((n_train,1)))
G = matrix(-np.eye(n_train))
h = matrix(np.zeros(n_train))
A = matrix(y_train.reshape(1,-1))
b = matrix(np.zeros(1))

solvers.options['show_progress']=True
qp_solver = solvers.qp(P, q, G, h, A, b)

alpha = np.array(qp_solver['x'])
```



```
pcost      dcost      gap    pres    dres
0: -3.9566e+00 -9.9644e+00 2e+01 4e+00 2e+00
1: -5.3822e+00 -8.3614e+00 4e+00 4e-01 2e-01
2: -7.3460e+00 -8.2900e+00 1e+00 7e-02 3e-02
3: -7.9669e+00 -8.0139e+00 5e-02 5e-04 2e-04
4: -7.9986e+00 -8.0008e+00 2e-03 6e-06 2e-06
5: -7.9999e+00 -8.0001e+00 2e-04 2e-09 8e-10
6: -8.0000e+00 -8.0000e+00 3e-05 3e-15 2e-16
7: -8.0000e+00 -8.0000e+00 4e-06 2e-15 2e-16
Optimal solution found.
```

```
[ ]
```

```
print('alpha = ',np.round(alpha.T,2))
```

```
alpha = [[ 0.  0.  0.  0.  0.  8.  0.  0.  0.  0.  8.  0.  0.]]
```

```
[ ]
```

```
w = np.sum(alpha*y_train[:,None] * x_train, axis=0)
print('w = ',w)
```

```
w = [ 2.39852179  3.20110867]
```



```
th = (alpha> 1e-4).reshape(-1)
```

```
b = y_train[th]-x_train[th].dot(w)
bias = b[0]
print('bias = ',bias)
```



```
bias = 0.0396673992178
```

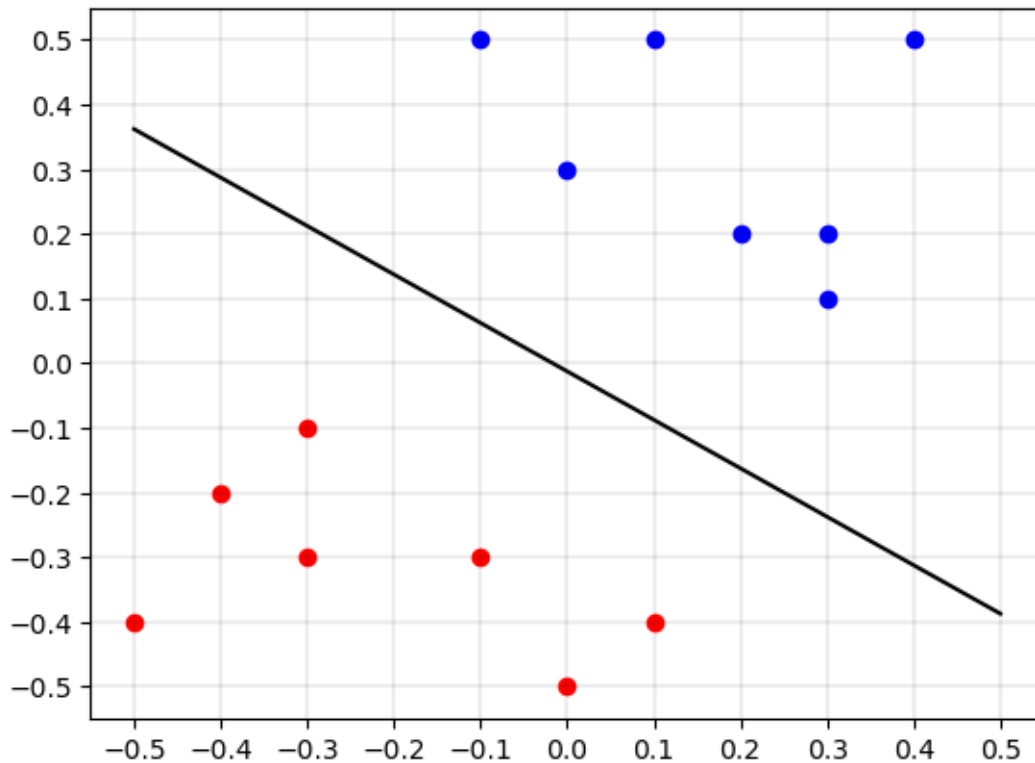
```
[ ]
```

```
norm = np.linalg.norm(w)
w, bias = w/norm, bias/norm
fig, ax = plt.subplots()

m = -w[0]/w[1]
intercept = -bias/w[1]
x2 = np.arange(-.5,1)
ax.plot(x2, x2*m + intercept, 'k-')
scatter_plot(ax, x_train, y_train)
plt.show()
```

Pertanyaan-2 (point 20):

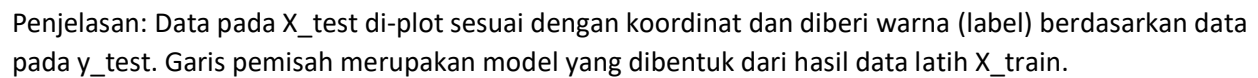
Apa output setelah baris ini di run?



```
norm = np.linalg.norm(w)
w, bias = w/norm, bias/norm
fig, ax = plt.subplots()

m = -w[0]/w[1]
intercept = -bias/w[1]
x2 = np.arange(-.5,1)
ax.plot(x2 , x2*m + intercept , 'k-')
scatter_plot(ax, x_test, y_test)
plt.show()
```

Pertanyaan-3 (point 20):
 Apa output setelah baris ini di run? Jelaskan!



1301218601 WandiyusufKurniawan IFX-45-GAB Tugas-1

pada `y_train`. Kemudian model tersebut diuji menggunakan data `X_test` dan label pada `y_test`. Ternyata, output dari soal 3 menunjukkan pemisahan label yang benar-benar sesuai dengan garis pemisah. Hal tersebut didukung dengan hasil `y_pred_train` dan `y_pred_test` memiliki akurasi sempurna karena kedua array tersebut benar-benar sesuai dengan label yang diberikan oleh `y_train` dan `y_test`.

Pertanyaan-5 (point 20):

Sebutkan contoh penggunaan algoritma SVM yang anda ketahui! Jelaskan dengan cara menjabarkan cara kerja algoritma SVM.

Contohnya pada klasifikasi artikel pada berita

1. Sebelum diproses menggunakan SVM, data mentah berupa artikel atau corpus di-preprocessing terlebih dahulu (record yang bolong dihapus, ubah semua kata menjadi huruf kecil kemudian ditokenisasi/dipisah-pisah, hapus stop words seperti kata sambung atau kata ganti, hapus kata yang mengandung angka atau symbol, dan lemmatisasi kata, seperti mengambil kata beri dari memberi).
2. Siapkan variable `X_train` (data latih prediktor), `X_test` (data uji prediktor), `y_train` (data latih target), dan `y_test` (data latih target) untuk memisahkan data yang sudah di-preprocessing dengan `train_test_split`. Untuk kasus ini data `X` (prediktor) adalah kata-kata yang sudah disaring dari artikel, dan data `y` (target) adalah `label_1` atau `label_2` dengan ukuran data uji sebesar 20%.
3. Label pada variable `y_train` dan `y_test` di hot encode terlebih dahulu untuk memudahkan klasifikasi. Misalnya `label_1` diberi encode 0 dan `label_2` diberi encode 1 dengan `.fit_transform()`.
4. Gunakan algoritma TF-IDF (Term Frequency – Inverse Document Frequency) untuk mengonversi data prediktor (kata-kata) menjadi vector dengan `.fit()` kata-kata saringan (gabungan `X_train` dan `X_test`) kemudian `.transform()` setiap `X_train` dan `X_test` untuk mendapatkan nilai yang bisa digunakan untuk klasifikasi SVM. Nilai tersebut masing-masing dimasukkan ke dalam variable `X_train_tfidf` dan `X_test_tfidf`.
5. Inisialisasi SVM dengan kernel 'linear' untuk membuat garis pemisah, kemudian `.fit()` `X_train_tfidf` dan `y_train`. Algoritma tersebut akan mencari jarak terjauh antar label yang terdekat dengan garis pemisah untuk menentukan nilai kemiringan/bobot dan bias ($y=wx+b$).
6. Setelah model dibuat, `.predict()` data `X_test_tfidf`, kemudian hitung akurasi dengan `y_test`. Semakin tinggi nilai akurasinya, semakin baik modelnya dan direkomendasikan untuk digunakan dalam klasifikasi artikel pada berita (misalnya 'hiburan' atau 'teknologi').