

EE 559 Homework 5

Rakshitha Panduranga

USC ID-7890-1614-34

Email ID- rpandura@usc.edu (<mailto:rpandura@usc.edu>)

Problem 1

I tried for the first dataset. But, I did not get the right accuracy.

In [1]:

```
1 import numpy as np
2 from numpy import genfromtxt
```

In [3]:

```
1 sytrain_data = np.genfromtxt('synthetic1_train.csv', delimiter=',')
2 sytest_data = np.genfromtxt('synthetic1_test.csv', delimiter=',')
```

In [4]:

```
1 sy_train_len = len(sytrain_data)
2 sy_test_len = len(sytest_data)
```

In [5]:

```
1 sy_train_len
```

Out[5]:

100

In [7]:

```
1 random_perm = np.random.permutation(sy_train_len)
```

In [97]:

1 random_perm

Out[97]:

```
array([ 4,  2, 67,  9, 34, 62,  3, 30, 22, 31,  8, 14, 15, 50, 28,  6, 81,
       94,  1, 45, 75, 70, 36, 65, 85, 39, 84, 59, 54, 91, 97, 52, 96, 83,
      51, 19, 87, 89, 48, 74, 40, 47, 56, 90, 42, 82, 93, 32, 25, 10, 95,
      24, 98, 37, 11, 18, 64, 66, 26, 27, 77, 71, 41,  0, 86, 58, 23, 99,
     49, 46, 16, 69, 80, 43, 44, 35, 88, 78, 55,  7, 57, 17, 61, 68, 20,
     72, 73, 21, 79,   5, 76, 63, 92, 13, 60, 38, 29, 53, 33, 12])
```

In [98]:

1 sytrain_data

Out[98]:

```
array([[ 4.9553e-01,  1.3221e+00,  1.0000e+00],
       [-7.6615e+00, -5.7644e+00,  1.0000e+00],
       [-5.1141e-01,  1.9586e+00,  1.0000e+00],
       [ 2.3214e+00,  5.0545e+00,  1.0000e+00],
       [-2.0195e+00,  3.4092e-01,  1.0000e+00],
       [-2.9861e+00, -3.3190e+00,  1.0000e+00],
       [-6.5189e+00, -4.7777e+00,  1.0000e+00],
       [ 2.6641e-01,  2.3592e+00,  1.0000e+00],
       [-4.8785e+00, -3.5146e+00,  1.0000e+00],
       [-4.8374e+00, -2.8104e+00,  1.0000e+00],
       [ 7.0224e-01,  3.3218e+00,  1.0000e+00],
       [ 5.7500e+00,  6.9757e+00,  1.0000e+00],
       [-6.8286e+00, -5.7670e+00,  1.0000e+00],
       [-1.7667e+00, -5.6320e-01,  1.0000e+00],
       [ 4.0435e-01,  1.8636e+00,  1.0000e+00],
       [-1.5761e+00, -6.3041e-01,  1.0000e+00],
       [ 2.1320e+00,  5.0140e+00,  1.0000e+00],
       [-5.1850e+00, -3.5251e+00,  1.0000e+00],
       [ 3.2005e+00,  2.3888e+00,  1.0000e+00],
       [-5.9964e-01,  1.9578e+00,  1.0000e+00],
       [-1.5437e+00,  3.2573e-01,  1.0000e+00],
       [-7.8170e+00, -5.1727e+00,  1.0000e+00],
       [-3.6526e+00, -3.2991e+00,  1.0000e+00],
       [-7.0529e-01,  5.0600e-03,  1.0000e+00],
       [-3.0048e+00, -6.8425e-01,  1.0000e+00],
       [-2.3935e+00, -1.9034e+00,  1.0000e+00],
       [-9.4264e-01, -4.6249e-01,  1.0000e+00],
       [-1.9343e+00,  2.3832e-01,  1.0000e+00],
       [-1.2830e+00, -7.6194e-01,  1.0000e+00],
       [-3.8848e-01,  1.1461e+00,  1.0000e+00],
       [-1.1151e+00,  2.0278e+00,  1.0000e+00],
       [-1.7090e+00,  8.7660e-01,  1.0000e+00],
       [ 1.9344e+00,  4.0634e+00,  1.0000e+00],
       [-4.2990e+00, -2.2258e+00,  1.0000e+00],
       [-5.9648e+00, -6.0329e+00,  1.0000e+00],
       [-4.1375e+00, -8.2818e-01,  1.0000e+00],
       [-4.0518e+00, -1.7807e+00,  1.0000e+00],
       [ 3.9041e+00,  6.4578e+00,  1.0000e+00],
       [-7.0233e+00, -4.9286e+00,  1.0000e+00],
       [-1.1953e+00,  2.5819e+00,  1.0000e+00],
       [-1.2685e+00,  1.1212e+00,  1.0000e+00],
       [ 1.3147e-01,  2.5571e+00,  1.0000e+00],
       [-6.8244e+00, -5.7910e+00,  1.0000e+00],
       [ 4.1980e+00,  5.8591e+00,  1.0000e+00],
       [-3.3886e+00, -9.8005e-01,  1.0000e+00],
       [ 1.3651e+00,  2.1773e+00,  1.0000e+00],
       [-5.7778e+00, -3.8964e+00,  1.0000e+00],
       [ 7.5680e+00,  8.2219e+00,  1.0000e+00],
       [-3.9977e+00, -1.9570e+00,  1.0000e+00],
       [-8.2418e+00, -4.6721e+00,  1.0000e+00],
       [-1.0683e+00, -4.0974e+00,  2.0000e+00],
       [ 5.4001e+00,  2.4341e+00,  2.0000e+00],
       [ 5.1599e+00,  2.5871e+00,  2.0000e+00],
       [ 3.0834e+00, -1.0177e+00,  2.0000e+00],
       [ 4.1844e+00,  1.9339e+00,  2.0000e+00],
```

```
[ 3.9144e-01, -1.5853e+00, 2.0000e+00],  
[ 4.4168e+00, 1.2120e-01, 2.0000e+00],  
[ 6.2230e+00, 5.0323e+00, 2.0000e+00],  
[ 1.2892e+00, -2.9687e-01, 2.0000e+00],  
[ 8.6351e+00, 5.7328e+00, 2.0000e+00],  
[ 1.8482e+00, -2.9414e+00, 2.0000e+00],  
[-8.7547e-01, -4.3877e+00, 2.0000e+00],  
[-4.2572e-01, -7.3642e+00, 2.0000e+00],  
[-9.2330e-01, -5.1256e+00, 2.0000e+00],  
[-8.1171e-02, -2.4433e+00, 2.0000e+00],  
[ 1.0514e+00, -3.0568e+00, 2.0000e+00],  
[ 6.3585e+00, 2.4300e+00, 2.0000e+00],  
[ 2.7006e+00, -2.5733e-01, 2.0000e+00],  
[-7.5879e-01, -1.2143e+00, 2.0000e+00],  
[ 5.1726e+00, 8.6644e-01, 2.0000e+00],  
[ 2.2665e+00, -8.2415e-01, 2.0000e+00],  
[ 7.1770e-01, -1.7864e+00, 2.0000e+00],  
[ 8.0069e+00, 4.1797e+00, 2.0000e+00],  
[ 1.0234e+00, -2.7906e+00, 2.0000e+00],  
[ 4.8224e+00, 2.2581e+00, 2.0000e+00],  
[ 1.5403e+00, -1.2853e+00, 2.0000e+00],  
[ 6.2164e+00, 4.5979e+00, 2.0000e+00],  
[ 6.6096e+00, 3.8412e+00, 2.0000e+00],  
[ 3.1687e+00, -4.2044e-01, 2.0000e+00],  
[ 4.2380e+00, 1.7191e+00, 2.0000e+00],  
[-1.1387e-01, -1.2830e+00, 2.0000e+00],  
[ 2.0702e+00, -8.5223e-01, 2.0000e+00],  
[ 2.3850e+00, -1.1126e-01, 2.0000e+00],  
[ 3.0375e+00, 4.9356e-02, 2.0000e+00],  
[ 4.4861e+00, 2.4781e+00, 2.0000e+00],  
[ 1.7149e+00, -1.6982e+00, 2.0000e+00],  
[ 2.2596e+00, -6.1605e-01, 2.0000e+00],  
[ 3.8097e+00, 5.3026e-02, 2.0000e+00],  
[ 2.2318e+00, -2.1668e+00, 2.0000e+00],  
[-2.0323e-01, -2.6196e+00, 2.0000e+00],  
[ 7.2327e+00, 6.2794e+00, 2.0000e+00],  
[ 4.0678e+00, 1.5098e+00, 2.0000e+00],  
[ 4.9854e-01, -1.0909e+00, 2.0000e+00],  
[ 7.5293e+00, 3.8676e+00, 2.0000e+00],  
[-2.1824e+00, -2.9826e+00, 2.0000e+00],  
[ 3.8817e+00, -8.3276e-03, 2.0000e+00],  
[ 7.0057e-01, -2.7706e+00, 2.0000e+00],  
[ 5.1785e+00, 1.4548e+00, 2.0000e+00],  
[ 3.4706e+00, 1.5744e+00, 2.0000e+00],  
[ 6.6011e+00, 3.8681e+00, 2.0000e+00]])
```

In [101]:

```
1 sytrainreflected_data0 = [];
2 sytrainreflected_data1 = [];
3 sytrainreflected_data2 = [];
4 sytrainreflected_data3 = [];
5 reflected_data = [];
6 for i in range(0,sy_train_len):
7     sytrainreflected_data3.append(sytrain_data[random_perm[i],2]);
8     if(sytrain_data[random_perm[i],2] == 2):
9         sytrainreflected_data0.append(-sytrain_data[random_perm[i],0]);
10        sytrainreflected_data1.append(-sytrain_data[random_perm[i],1]);
11        sytrainreflected_data2.append(-1);
12    else:
13        sytrainreflected_data0.append(sytrain_data[random_perm[i],0]);
14        sytrainreflected_data1.append(sytrain_data[random_perm[i],1]);
15        sytrainreflected_data2.append(1);
16 reflected_data = np.column_stack((sytrainreflected_data0,sytrainreflected_data1,sytrainreflected_data2))
```

In [102]:

1 reflected_data

Out[102]:

```
array([[-2.0195e+00,  3.4092e-01,  1.0000e+00,  1.0000e+00],
       [-5.1141e-01,  1.9586e+00,  1.0000e+00,  1.0000e+00],
       [-2.7006e+00,  2.5733e-01, -1.0000e+00,  2.0000e+00],
       [-4.8374e+00, -2.8104e+00,  1.0000e+00,  1.0000e+00],
       [-5.9648e+00, -6.0329e+00,  1.0000e+00,  1.0000e+00],
       [ 4.2572e-01,  7.3642e+00, -1.0000e+00,  2.0000e+00],
       [ 2.3214e+00,  5.0545e+00,  1.0000e+00,  1.0000e+00],
       [-1.1151e+00,  2.0278e+00,  1.0000e+00,  1.0000e+00],
       [-3.6526e+00, -3.2991e+00,  1.0000e+00,  1.0000e+00],
       [-1.7090e+00,  8.7660e-01,  1.0000e+00,  1.0000e+00],
       [-4.8785e+00, -3.5146e+00,  1.0000e+00,  1.0000e+00],
       [ 4.0435e-01,  1.8636e+00,  1.0000e+00,  1.0000e+00],
       [-1.5761e+00, -6.3041e-01,  1.0000e+00,  1.0000e+00],
       [ 1.0683e+00,  4.0974e+00, -1.0000e+00,  2.0000e+00],
       [-1.2830e+00, -7.6194e-01,  1.0000e+00,  1.0000e+00],
       [-6.5189e+00, -4.7777e+00,  1.0000e+00,  1.0000e+00],
       [-2.0702e+00,  8.5223e-01, -1.0000e+00,  2.0000e+00],
       [ 2.1824e+00,  2.9826e+00, -1.0000e+00,  2.0000e+00],
       [-7.6615e+00, -5.7644e+00,  1.0000e+00,  1.0000e+00],
       [ 1.3651e+00,  2.1773e+00,  1.0000e+00,  1.0000e+00],
       [-1.5403e+00,  1.2853e+00, -1.0000e+00,  2.0000e+00],
       [-2.2665e+00,  8.2415e-01, -1.0000e+00,  2.0000e+00],
       [-4.0518e+00, -1.7807e+00,  1.0000e+00,  1.0000e+00],
       [-1.0514e+00,  3.0568e+00, -1.0000e+00,  2.0000e+00],
       [-1.7149e+00,  1.6982e+00, -1.0000e+00,  2.0000e+00],
       [-1.1953e+00,  2.5819e+00,  1.0000e+00,  1.0000e+00],
       [-4.4861e+00, -2.4781e+00, -1.0000e+00,  2.0000e+00],
       [-8.6351e+00, -5.7328e+00, -1.0000e+00,  2.0000e+00],
       [-4.1844e+00, -1.9339e+00, -1.0000e+00,  2.0000e+00],
       [-4.0678e+00, -1.5098e+00, -1.0000e+00,  2.0000e+00],
       [-5.1785e+00, -1.4548e+00, -1.0000e+00,  2.0000e+00],
       [-5.1599e+00, -2.5871e+00, -1.0000e+00,  2.0000e+00],
       [-7.0057e-01,  2.7706e+00, -1.0000e+00,  2.0000e+00],
       [-3.0375e+00, -4.9356e-02, -1.0000e+00,  2.0000e+00],
       [-5.4001e+00, -2.4341e+00, -1.0000e+00,  2.0000e+00],
       [-5.9964e-01,  1.9578e+00,  1.0000e+00,  1.0000e+00],
       [-3.8097e+00, -5.3026e-02, -1.0000e+00,  2.0000e+00],
       [ 2.0323e-01,  2.6196e+00, -1.0000e+00,  2.0000e+00],
       [-3.9977e+00, -1.9570e+00,  1.0000e+00,  1.0000e+00],
       [-4.8224e+00, -2.2581e+00, -1.0000e+00,  2.0000e+00],
       [-1.2685e+00,  1.1212e+00,  1.0000e+00,  1.0000e+00],
       [ 7.5680e+00,  8.2219e+00,  1.0000e+00,  1.0000e+00],
       [-4.4168e+00, -1.2120e-01, -1.0000e+00,  2.0000e+00],
       [-7.2327e+00, -6.2794e+00, -1.0000e+00,  2.0000e+00],
       [-6.8244e+00, -5.7910e+00,  1.0000e+00,  1.0000e+00],
       [-2.3850e+00,  1.1126e-01, -1.0000e+00,  2.0000e+00],
       [-7.5293e+00, -3.8676e+00, -1.0000e+00,  2.0000e+00],
       [ 1.9344e+00,  4.0634e+00,  1.0000e+00,  1.0000e+00],
       [-2.3935e+00, -1.9034e+00,  1.0000e+00,  1.0000e+00],
       [ 7.0224e-01,  3.3218e+00,  1.0000e+00,  1.0000e+00],
       [-3.8817e+00,  8.3276e-03, -1.0000e+00,  2.0000e+00],
       [-3.0048e+00, -6.8425e-01,  1.0000e+00,  1.0000e+00],
       [-3.4706e+00, -1.5744e+00, -1.0000e+00,  2.0000e+00],
       [ 3.9041e+00,  6.4578e+00,  1.0000e+00,  1.0000e+00],
       [ 5.7500e+00,  6.9757e+00,  1.0000e+00,  1.0000e+00],
```

```
[ 3.2005e+00,  2.3888e+00,  1.0000e+00,  1.0000e+00],
[ 8.1171e-02,  2.4433e+00, -1.0000e+00,  2.0000e+00],
[ -6.3585e+00, -2.4300e+00, -1.0000e+00,  2.0000e+00],
[ -9.4264e-01, -4.6249e-01,  1.0000e+00,  1.0000e+00],
[ -1.9343e+00,  2.3832e-01,  1.0000e+00,  1.0000e+00],
[ -6.6096e+00, -3.8412e+00, -1.0000e+00,  2.0000e+00],
[ -7.1770e-01,  1.7864e+00, -1.0000e+00,  2.0000e+00],
[ 1.3147e-01,  2.5571e+00,  1.0000e+00,  1.0000e+00],
[ 4.9553e-01,  1.3221e+00,  1.0000e+00,  1.0000e+00],
[ -2.2596e+00,  6.1605e-01, -1.0000e+00,  2.0000e+00],
[ -1.2892e+00,  2.9687e-01, -1.0000e+00,  2.0000e+00],
[ -7.0529e-01,  5.0600e-03,  1.0000e+00,  1.0000e+00],
[ -6.6011e+00, -3.8681e+00, -1.0000e+00,  2.0000e+00],
[ -8.2418e+00, -4.6721e+00,  1.0000e+00,  1.0000e+00],
[ -5.7778e+00, -3.8964e+00,  1.0000e+00,  1.0000e+00],
[ 2.1320e+00,  5.0140e+00,  1.0000e+00,  1.0000e+00],
[ -5.1726e+00, -8.6644e-01, -1.0000e+00,  2.0000e+00],
[ 1.1387e-01,  1.2830e+00, -1.0000e+00,  2.0000e+00],
[ 4.1980e+00,  5.8591e+00,  1.0000e+00,  1.0000e+00],
[ -3.3886e+00, -9.8005e-01,  1.0000e+00,  1.0000e+00],
[ -4.1375e+00, -8.2818e-01,  1.0000e+00,  1.0000e+00],
[ -2.2318e+00,  2.1668e+00, -1.0000e+00,  2.0000e+00],
[ -3.1687e+00,  4.2044e-01, -1.0000e+00,  2.0000e+00],
[ -3.9144e-01,  1.5853e+00, -1.0000e+00,  2.0000e+00],
[ 2.6641e-01,  2.3592e+00,  1.0000e+00,  1.0000e+00],
[ -6.2230e+00, -5.0323e+00, -1.0000e+00,  2.0000e+00],
[ -5.1850e+00, -3.5251e+00,  1.0000e+00,  1.0000e+00],
[ 8.7547e-01,  4.3877e+00, -1.0000e+00,  2.0000e+00],
[ 7.5879e-01,  1.2143e+00, -1.0000e+00,  2.0000e+00],
[ -1.5437e+00,  3.2573e-01,  1.0000e+00,  1.0000e+00],
[ -8.0069e+00, -4.1797e+00, -1.0000e+00,  2.0000e+00],
[ -1.0234e+00,  2.7906e+00, -1.0000e+00,  2.0000e+00],
[ -7.8170e+00, -5.1727e+00,  1.0000e+00,  1.0000e+00],
[ -4.2380e+00, -1.7191e+00, -1.0000e+00,  2.0000e+00],
[ -2.9861e+00, -3.3190e+00,  1.0000e+00,  1.0000e+00],
[ -6.2164e+00, -4.5979e+00, -1.0000e+00,  2.0000e+00],
[ 9.2330e-01,  5.1256e+00, -1.0000e+00,  2.0000e+00],
[ -4.9854e-01,  1.0909e+00, -1.0000e+00,  2.0000e+00],
[ -1.7667e+00, -5.6320e-01,  1.0000e+00,  1.0000e+00],
[ -1.8482e+00,  2.9414e+00, -1.0000e+00,  2.0000e+00],
[ -7.0233e+00, -4.9286e+00,  1.0000e+00,  1.0000e+00],
[ -3.8848e-01,  1.1461e+00,  1.0000e+00,  1.0000e+00],
[ -3.0834e+00,  1.0177e+00, -1.0000e+00,  2.0000e+00],
[ -4.2990e+00, -2.2258e+00,  1.0000e+00,  1.0000e+00],
[ -6.8286e+00, -5.7670e+00,  1.0000e+00,  1.0000e+00]])
```

In [167]:

```
1 initial_weight = [0.1,0.1,0.1]
2 halt1 = 0;
3 sol = np.zeros(3);
4 flag = 0;
5 dot_prod1 = 1;
6 jw = np.zeros([sy_train_len,4])
7 sum_weight = sum(initial_weight)
8 for j in range(0,1000):
9     k=0;
10    for i in range(0,sy_train_len):
11        dot_prod = reflected_data[i,0]*initial_weight[0] + reflected_data[i,1]*initial_weight[1];
12        if(dot_prod <= 0):
13            initial_weight[0] = reflected_data[i,0]+initial_weight[0];
14            initial_weight[1] = reflected_data[i,1]+initial_weight[1];
15            initial_weight[2] = reflected_data[i,2]+initial_weight[2];
16        if(dot_prod > 0 and dot_prod1<=0):
17            halt1 = 1;
18        elif(dot_prod >0 and dot_prod1 >0):
19            halt1 = halt1 + 1;
20        if(halt1 == sy_train_len):
21            j = 1000;
22            print("they are linearly separable");
23            sol = initial_weight;
24            print(sol);
25        if(j ==999):
26            jw[k,0:3]=initial_weight;
27            jw[k,3] = dot_prod;
28            k = k+1;
29            flag = 1;
30            dot_prod1 = dot_prod;
31        if(flag ==1):
32            for l in range(0,sy_train_len):
33                if(jw[l,3] == np.min(reflected_data[:,3])):
34                    sol = jw[l,0:3];
35 data0 = sytrain_data[:,0];
36 data1 = sytrain_data[:,1];
37 data2 = np.ones(sy_train_len);
38 data = np.column_stack((data0,data1,data2))
39 labels = [];
40 for i in range(0,len(data)):
41     pout = data[i];
42     temp = np.dot(pout,sol)
43     if(temp>0):
44         labels.append(1);
45     elif(temp<0):
46         labels.append(2);
47     else:
48         labels.append(0);
49 count = 0;
50 for j in range(0,sy_train_len):
51     if(sytrain_data[i,2] == labels[i]):
52         count = count+1;
53
54 acc = (count/sy_train_len)*100;
55
56
```

In [169]:

1	acc
---	-----

Out[169]:

0.0