EE559
HW8
1
(a)
In data set 1, we use Linear Kernel.
When I used parameter C = 1, the accuracy is 0.9. When I used parameter C = 100, the accuracy is 1.0.
In the plot and the accuracy result, we could know that the larger C can increase the accuracy and make the decision boundary move more left and top.

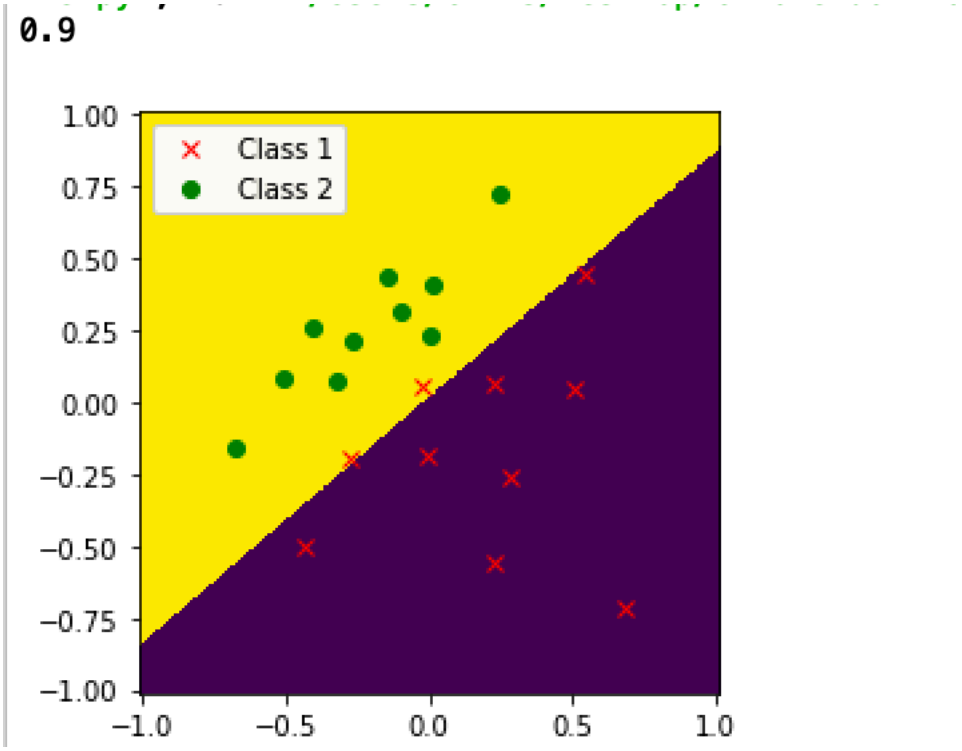

Fig1. Linear Kernel with C = 1

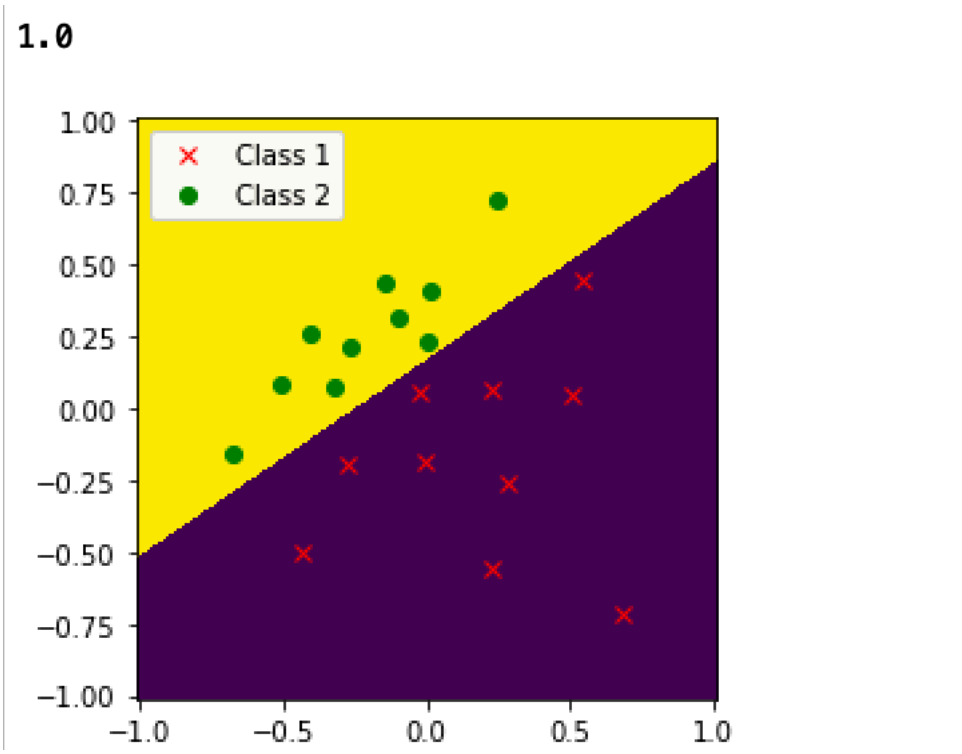

Fig2. Linear Kernel with C = 100

(b)
When C = 100 with Linear Kernel, the support vectors are

```
[[-0.023855   0.06042  ]
 [ 0.54579    0.45029  ]
 [ 0.0064864  0.23394  ]]
```

W (W1 and W2) are [[-7.11966384 10.40264821]]
W0 is [-1.79836766]
g(x) = W*X.T + W0

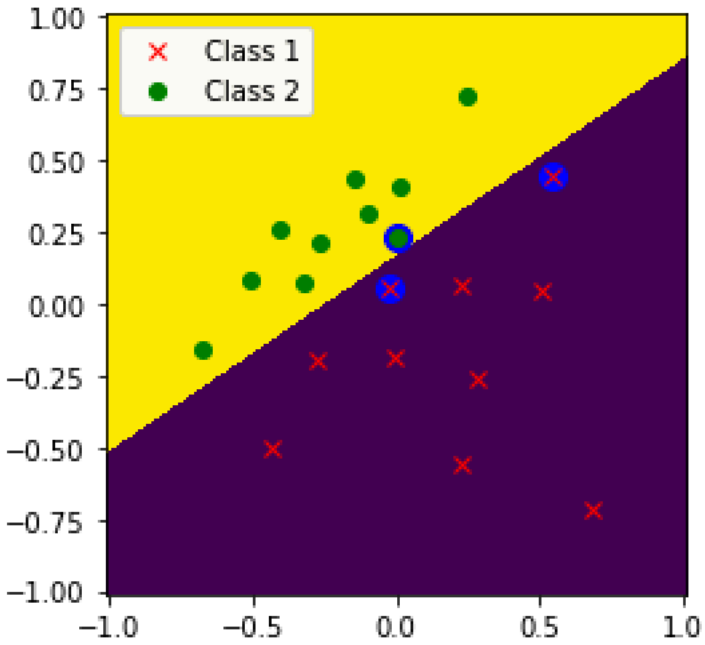

Fig2. Linear Kernel with C = 100, circled the support vectors

(c)
g(x) for each support vector are
[-1.00000007]
[-1.00000052]
[0.58904688]

In this result, the two of the support vectors in class 1 are on the margin boundary and one in class 2 is in the gap of two margin boundary.
In my opinion the one in class 2 may satisfied the the constraint zi(w.T * ui + wo) >= 1 - εi, and in this situation, the svm can get the minimize J(w)

(d)
Use Gaussian (RBF) Kernel with gamma parameter set to ‘auto’. Set C = 50, the accuracy is 0.95

```
0.95
```

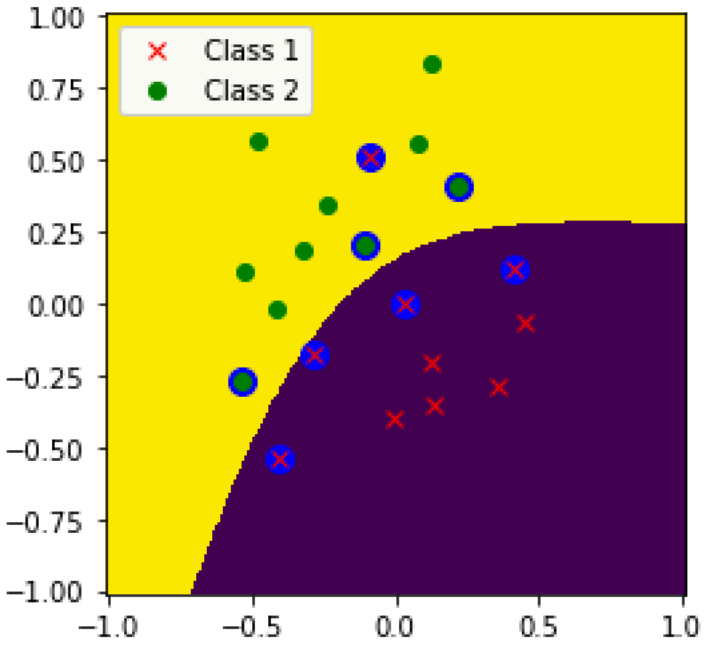

Fig3. RBF Kernel with C = 50

Use Gaussian (RBF) Kernel with gamma parameter set to ‘auto’. Set C = 5000, the accuracy is 1.0

```
1.0
```

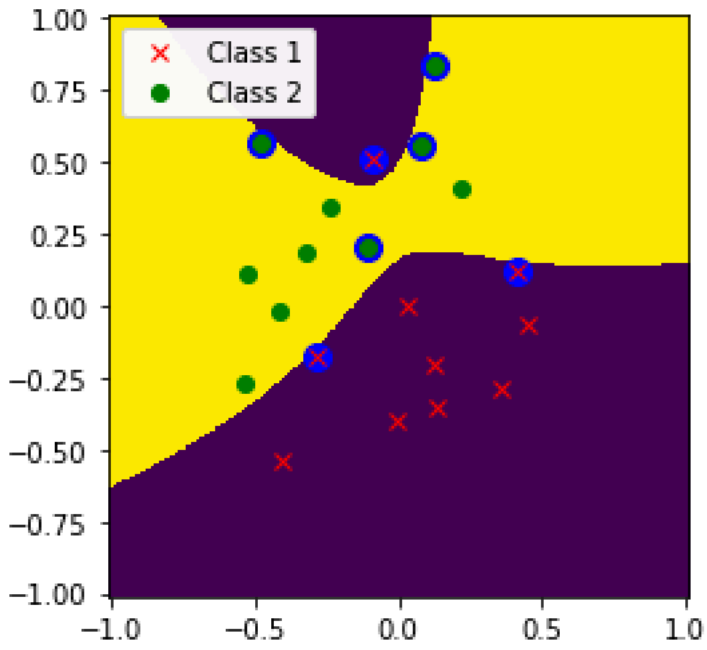

Fig4. RBF Kernel with C = 5000

The decision boundary is nonlinear, because it use Gaussian (RBF) Kernel, which use the exponential function and Eulidean distance to calculate the dual lagrangian Optimization.

With different C, separating the decision regions are quite different. The larger C get will ignore the less relevant support vector. Like in the C=5000, the lower purple region ignore 4 points in class 1 and 2, which is more far away than the decision boundary, which means it is increase the training accuracy. And the larger C get will detect more relevant region which lower C may overlooked. Like in the C=5000, the upper purple region detected the point in class 1 is not from the yellow margin. It will give a more support vector of that region to get more precisely region which lower C cannot covered.

(e)
Use Gaussian (RBF) Kernel with default C parameter. Set gamma = 10, the accuracy is 0.95

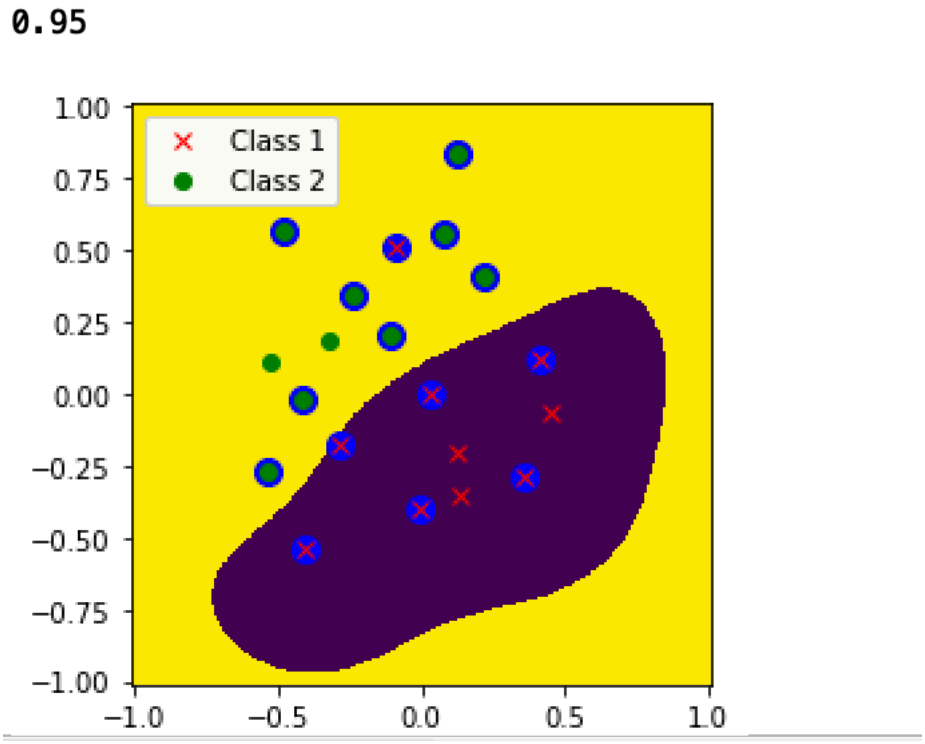

Fig5. RBF Kernel with gamma = 10

Use Gaussian (RBF) Kernel with default C parameter. Set gamma = 50, the accuracy is 1.0

```
1.0
```

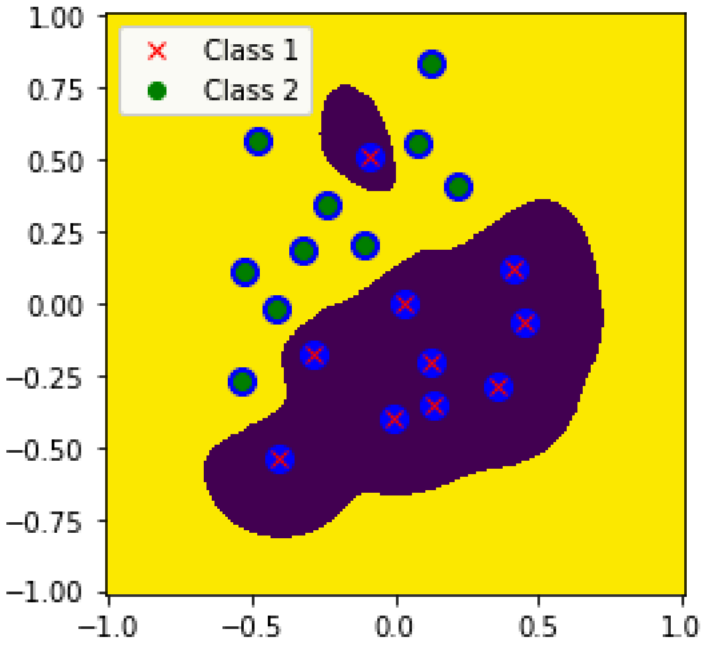

Fig6. RBF Kernel with gamma = 50

Use Gaussian (RBF) Kernel with default C parameter. Set gamma = 5000, the accuracy is 1.0

```
1.0
```

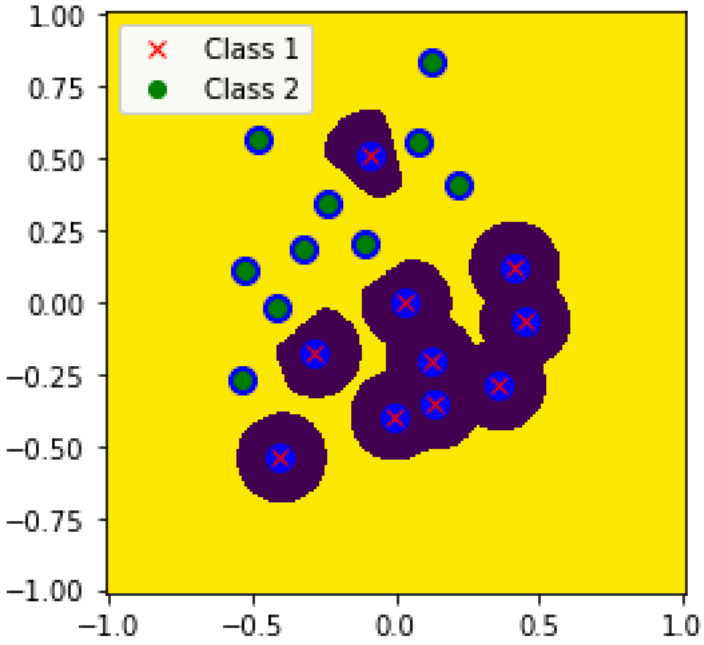

Fig7. RBF Kernel with gamma = 5000

The larger gamma will get a more specific boundary and will include more support vector. I think both gamma = 50 and gamma = 5000 are overfitted because in order to increase the accuracy, the model include the all points as the support vector. It is obviously to see that the support vector parameters are not robust when we using all.

2.
(a)
Use Gaussian (RBF) Kernel. Set gamma = 1, C = 1.
The average cross validation accuracy is 0.842483660130719.

(b)

(i)

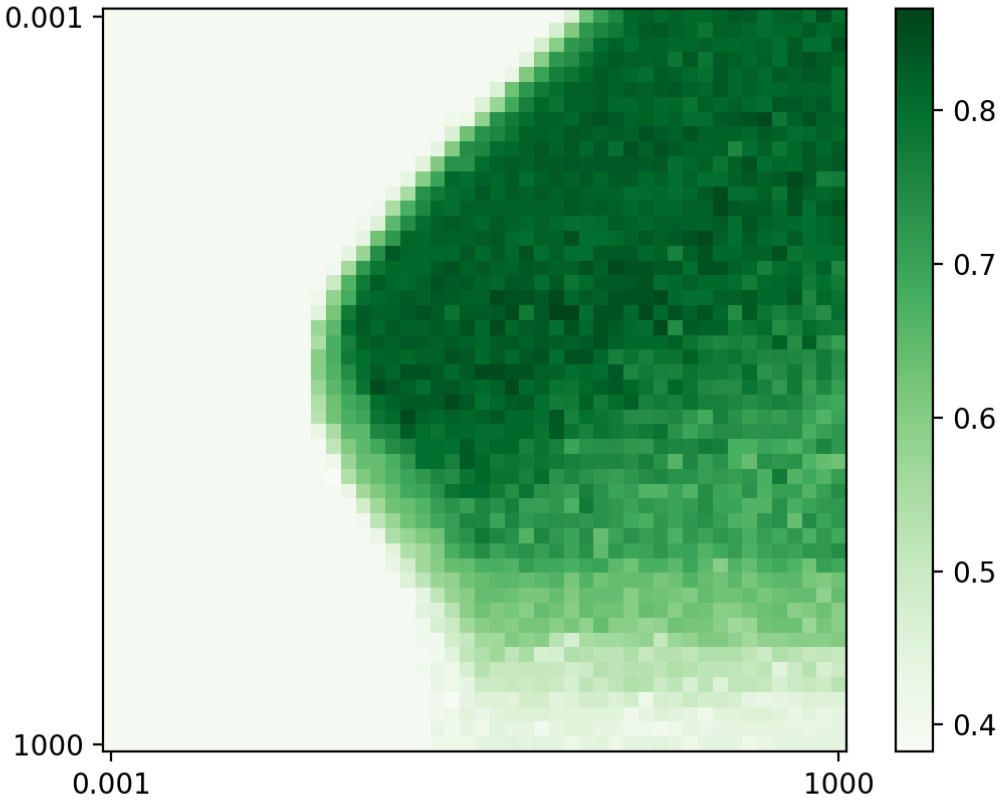

Fig7. Visualize ACC

(ii)
The max can be founded in graph, but may spent a little time
The max accuracy = 0.8660130718954248
Its standard deviation = 0.06391374907061416
Gamma = 0.28117686979742307
C = 6.25055192527397

(c)
(i)
The 20 chosen pairs of [γ, C] are [[0.49417133613238334, 2.6826957952797246], [0.655128556859551, 10.985411419875572], [0.28117686979742307, 2.023589647725158], [0.3727593720314938, 25.595479226995334], [0.655128556859551, 2.6826957952797246], [0.012648552168552958, 1000.0], [0.49417133613238334, 1.151395399326447], [2.023589647725158, 2.6826957952797246], [0.655128556859551, 6.25055192527397], [0.09102981779915217, 184.20699693267164], [1.151395399326447, 3.5564803062231287], [0.28117686979742307, 8.286427728546842], [0.655128556859551, 10.985411419875572],

[0.029470517025518096, 244.205309454865], [0.029470517025518096, 59.636233165946365], [0.03906939937054617, 429.1934260128778], [0.28117686979742307, 10.985411419875572], [0.0071968567300115215, 244.205309454865], [0.21209508879201905, 44.98432668969444], [0.655128556859551, 3.5564803062231287]] (20 pairs of [γ, C])

(ii)
the best values for [γ, C] (over the T=20) = [0.655128556859551, 2.6826957952797246]
Its accuracy = 0.83781045751634
Its standard deviation = 0.08670145277949695
Yes. It runs more well-defined or more reproducible now.

(d)
The final accuracy with test set = 0.8089887640449438
Yes. This is estimate within approximately 1 standard deviation of my mean cross-validation accuracy from (c) (ii).