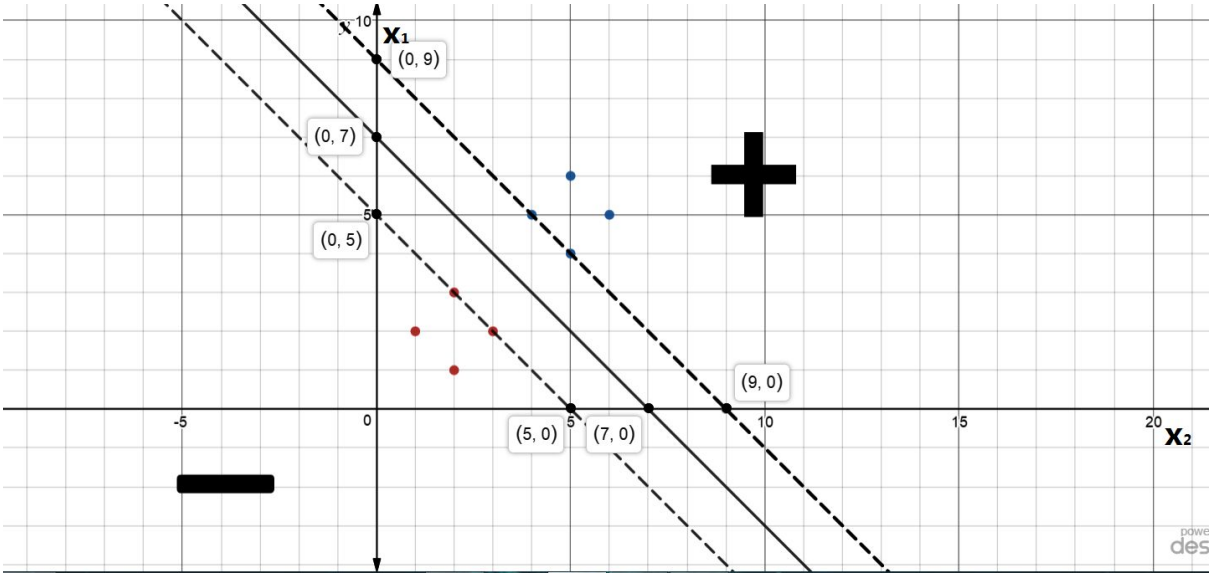
1.

(a)



(b)

From the graph, we can see that for the boundary, w = (1, 1) and b = -7.

Thus, margin = 1/||w|| = 1/ $\sqrt{2}$

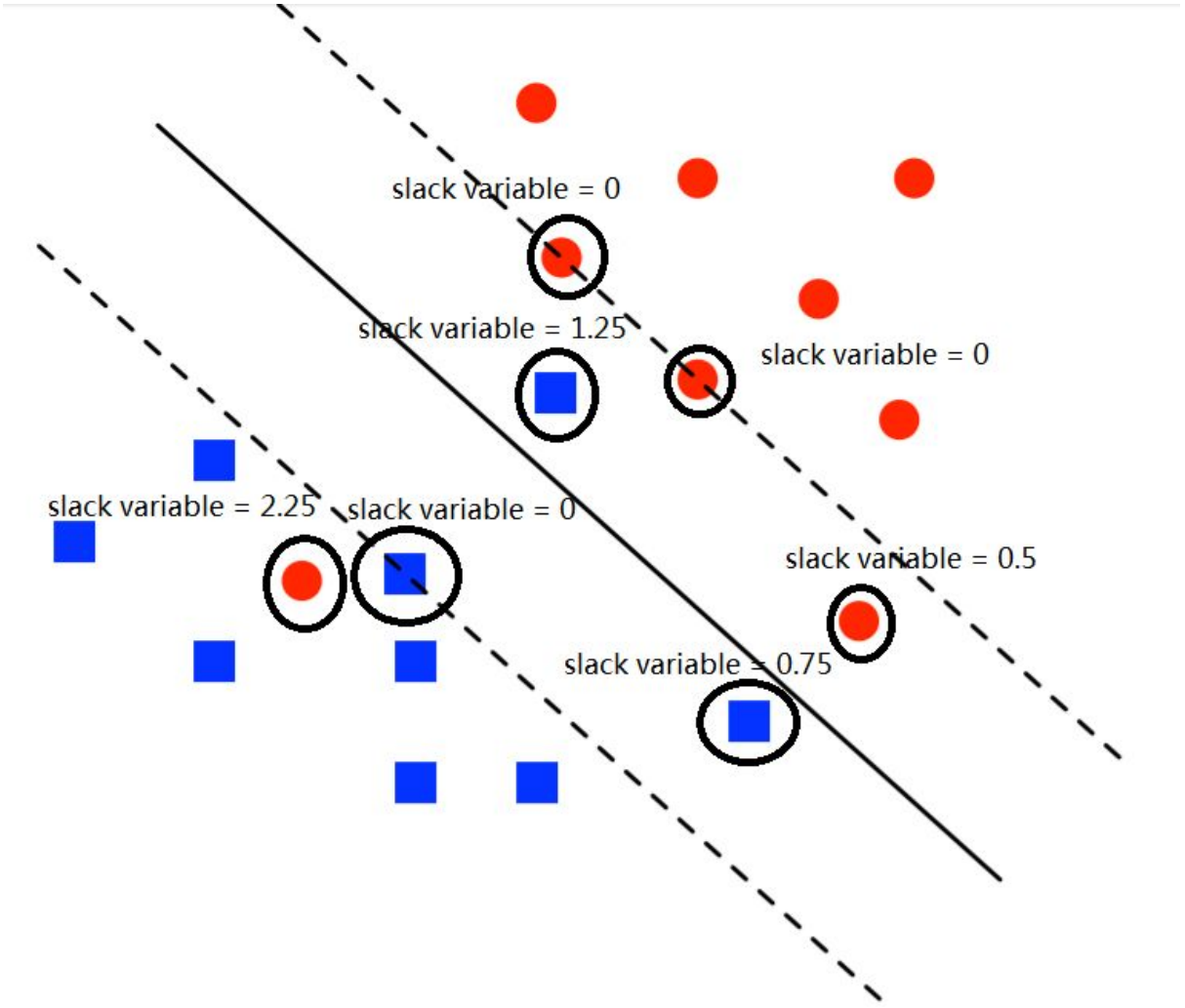
(c)

From the graph, we can see that for the boundary, w = (1, 1) and b = -7.

2.

(a)

In the graph below, all support vectors are circled with black, and their slack variables are indicated directly above the circles.



(b)

Suppose the factor C in the soft-margin SVM optimization problem were increased, we would expect the margin to decrease.

3.
(a)
Answer: possibly false
Justification: a_i represents the number of times an update occurred on point i, which could be greater than 1.

(b)
Answer: necessarily true
Justification: since a_i represents the number of times an update occurred on point i, and we have a total of k updates on all our point, we have $\sum_i a_i = k$

(c)
Answer: necessarily true
Justification: a_i would be 0 if and only if there were no updates on point i. Since we only have k updates, a has at most k nonzero coordinates.

(d)
Answer: necessarily true
Justification: since the training data converges after k updates, we know that the training data must be linearly separable, otherwise it would never converge.

4.

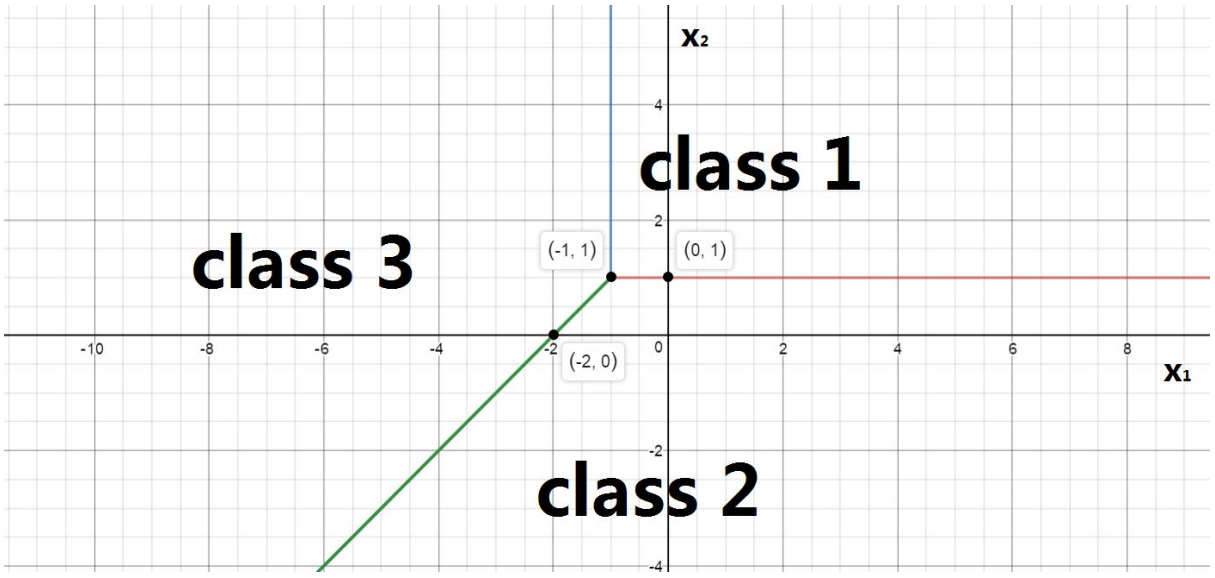
(a)

In the primal optimization problem, w has d variables, b is a single variable, and epsilon has n variables. Thus, there are a total of d+1+n variables.

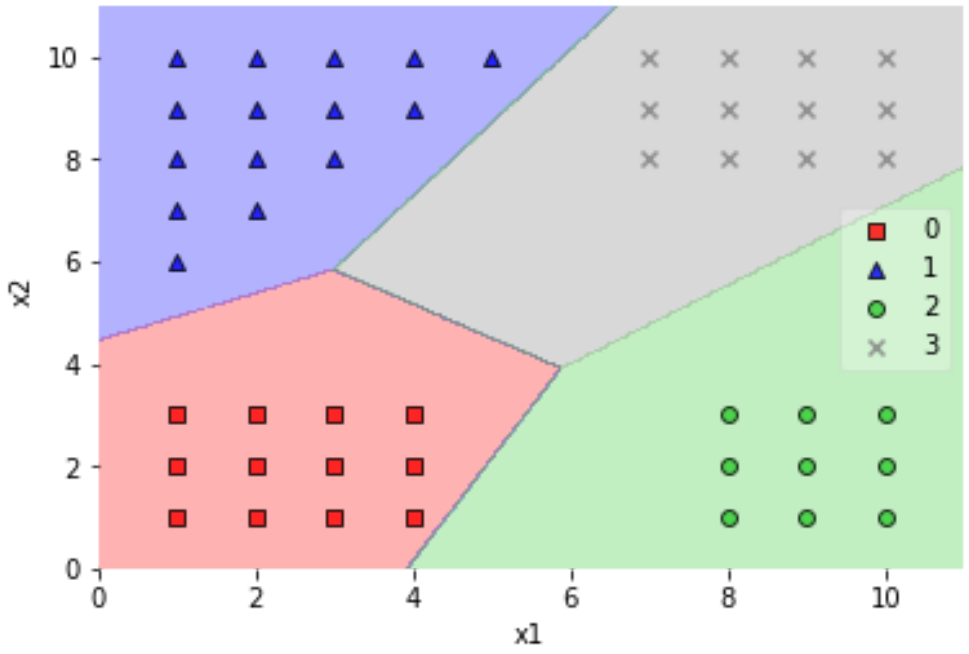
(b)

In the dual optimization problem, a has n vriables. Thus, there are a total of n variables.

5.

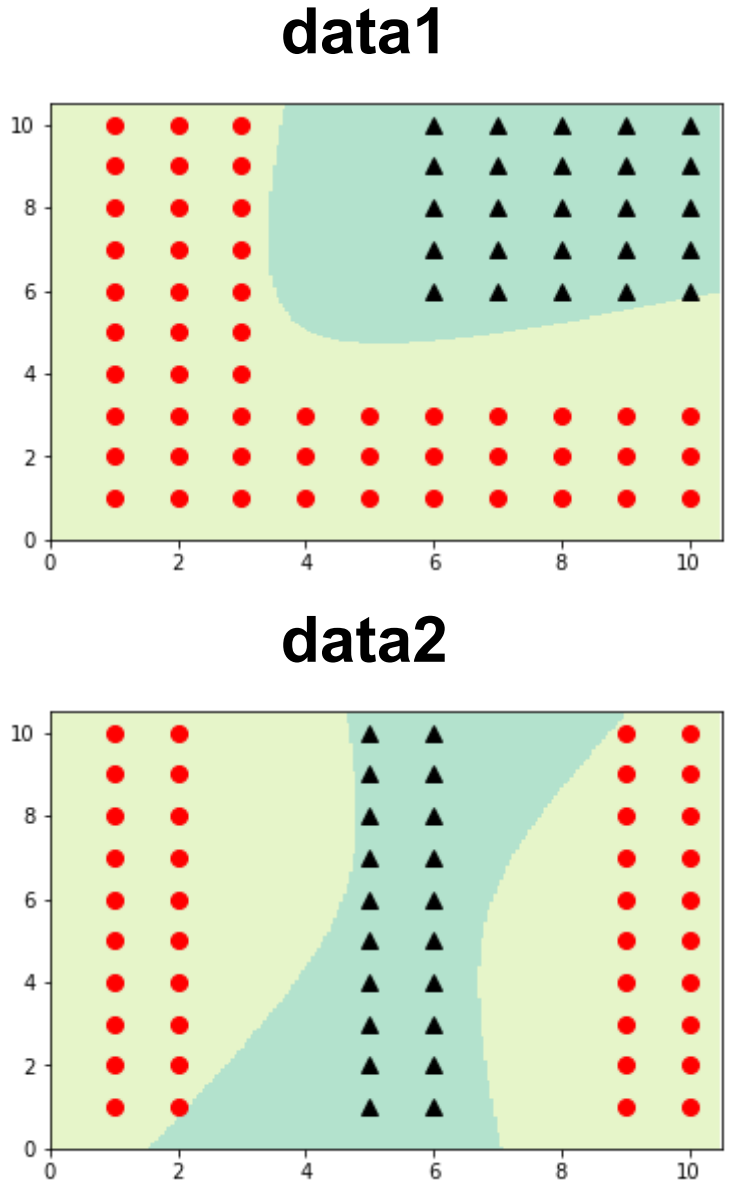


6.



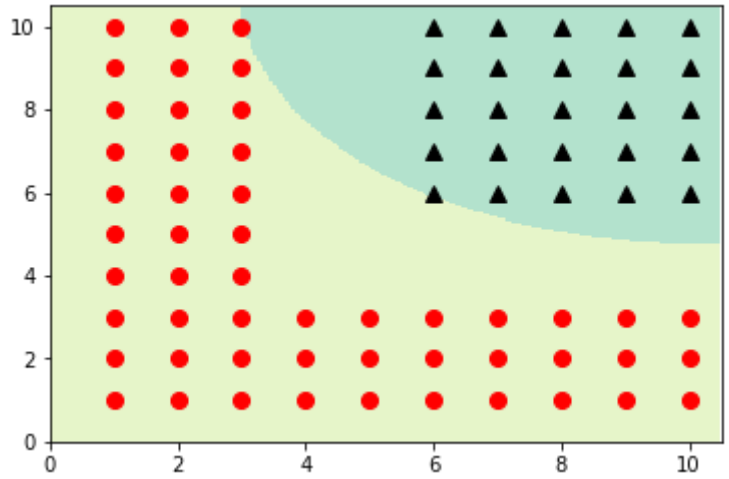
7.

(b) Using kernel Perceptron with quadratic kernel

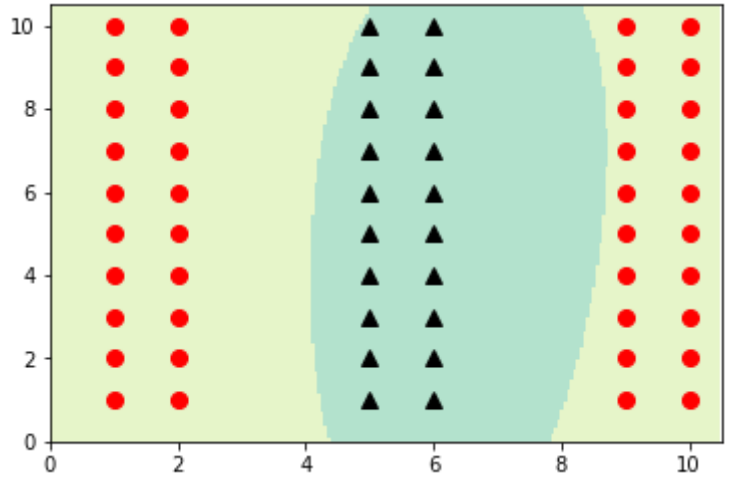


(c) kernel Perceptron with RBG kerne

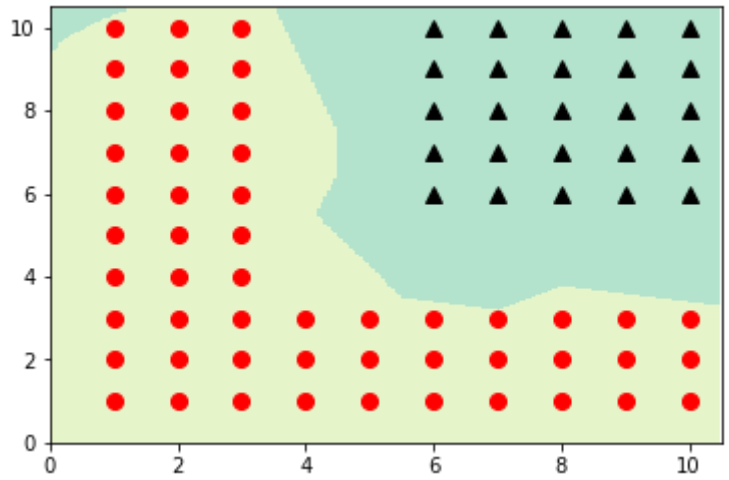
data1(sigma=0.1)



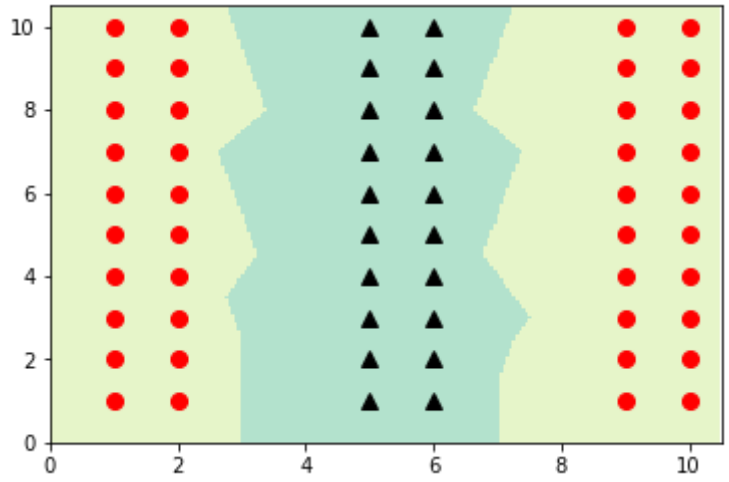
data2(sigma=0.1)



data1(sigma=10)



data2(sigma=10)



8.
(b)

C	Training Error	Test Error
0.01	11.81%	12.60%
0.1	10.74%	11.33%
1.0	13.18%	13.92%
10.0	11.01%	11.67%
100.0	13.73%	14.67%

From the table, we can see that the error rates never drop. Therefore, the data is clearly not linearly separable.

(c)

C	Training Error	Test Error	# of support vectors
1.0	0.00%	0.0194%	8652