Introduction to Machine Learning

Homework 2: Support Vector Machines

1) Consider multiclass classification problem with C classes, with the label $y^{(i)}$ taking values in $\{1,2,\ldots, C\}$.

a) Devise a simple, naïve approach that doesn't involve any training or averaging of data points. Why is this approach computationally expensive during test time? Why might this approach perform poorly at test time?

b) Devise a similar approach based on the class means (centroids). Neither this approach nor the approach in part (a) uses parameters **w**. Hint: We briefly discussed this approach in class on the white board. Why is this approach more computationally efficient than the approach (a). Why may it perform poorly at test time?

2) Consider the following training data:

class	x_1	x_2
+	2	1
+	3	2
+	3	0
–	1	0
–	2	0
–	1	1

a) Plot these six training points. Are the classes $\{+, -\}$ linearly separable?
b) Construct the weight vector (w_1, w_2, b) of the maximum margin hyperplane by inspection. Determine the distances between the hyperplane and each of the six training points. What is the margin of your hyperplane? Which training points are support vectors?
c) If you remove one of the support vectors, does the size of the optimal margin decrease, stay the same, or increase?

3) Let w, b be a hyperplane. Show that the distance between a training example $\mathbf{x}^{(i)}$, $y^{(i)}$ and the hyperplane is given by $\delta^{(i)} = y^{(i)}(\mathbf{w}\bullet\mathbf{x}^{(i)} + b)/|\mathbf{w}|$. Provide all details without making the derivation longer or more complicated than it has to be.

4) Beginning with the optimization problem on slide 9 on the slide in the SVM lecture, prove that the solution to the hard-margin optimization problem on page 11 provides a separating hyper-plane with maximum margin

a) Suppose that $\mathbf{w}^*$, b^* is optimal for the hard margin optimization problem on page 11. We must show that $\mathbf{w}^*$, b^* gives a hyperplane that maximizes the margin. First show that the margin for $\mathbf{w}^*$, b^* (distance from hyperplane to nearest training example) is $1/\|\mathbf{w}^*\|$. To do this, you'll want to use the explicit expression derived in class for the distance

between a training example $\mathbf{x}^{(i)}$, $y^{(i)}$ and the the hyperplane defined by $\mathbf{w}^*$ and b*. You'll also need to make use of the fact $\mathbf{w}^*$, b satisfy the constraints in the hard margin optimization problem on page 11 and that it meets at least one of the constraints with equality (since it is optimal).

b) Now let $\mathbf{z}$, d be any other separating hyperplane, and let M denote its margin for the data set. Define $\mathbf{z}' = \mathbf{z}/||\mathbf{z}||M$ and $d' = d/||\mathbf{z}||M$. Note that $||\mathbf{z}'|| = 1/M$. Show that $\mathbf{z}'$, d' is a feasible solution for the hard-margin optimization problem

c) Use the results from (a)-(b) to show that the margin for $\mathbf{w}^*$, b* is ≥ the margin for $\mathbf{z}$, d.

5) Consider a supervised machine learning problem with two features (x_1, x_2) and 4 training points $\underline{x}^{(1)}$, $\underline{x}^{(2)}$, $\underline{x}^{(3)}$, $\underline{x}^{(4)}$:

data	class	x_1	x_2
$\underline{x}^{(1)}$	+	1	0
$\underline{x}^{(2)}$	+	3	2
$\underline{x}^{(3)}$	–	1	2
$\underline{x}^{(4)}$	–	3	0

Denote w_0 & w_1 for the weights and b for the bias.

a) Prove by contradiction that there is no solution that satisfies the constraints for the hard-margin SVM problem.
b) Show that there is a solution for the soft margin SVM problem. Explicitly provide such a solution (w_0,w_1,b, $\xi^{(1)}$, $\xi^{(2)}$, $\xi^{(3)}$, $\xi^{(4)}$).

6) What python libraries are available for SVM? Do they allow for hard and soft SVM? Do they allow for kernels?

Edison Chen, ebc5802
Machine Learning
9/14/2023
HW#2 Support Vector Machines

Problem 1) Consider multiclass classification problem with C classes, with the label y(i) taking values in
{1,2,…, C}.
a) Devise a simple, naïve approach that doesn't involve any training or averaging of data points. Why is this approach computationally expensive during test time? Why might this approach perform poorly at test time?

When classifying a new point, measure the distance between it and every known point. Choose the known point closest to the new point. Classify the new point using the group of the closest known point.

This approach is computationally expensive during test time because every new point must be measured against every known point.

This approach might perform poorly at test time because it is a very local approach that does not consider the overall distribution of each class. In addition, if the training data contains noise or outliers, this method will be highly sensitive to such anomalies. A single outlier can mislead the classification of the new point.

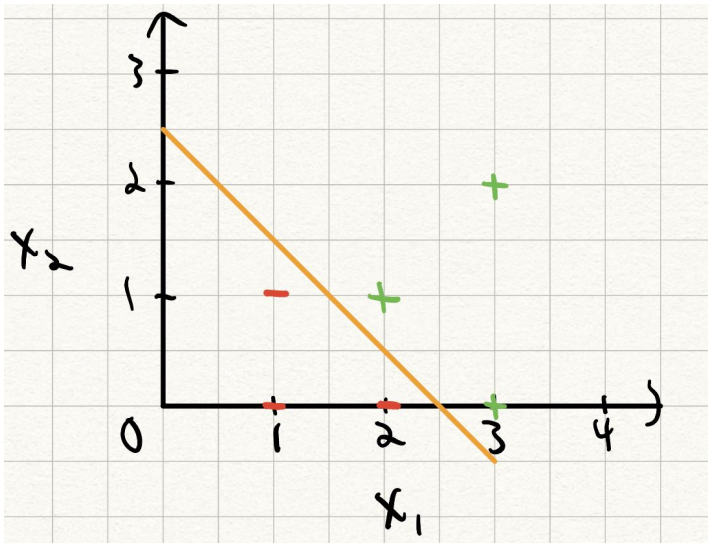
b) Devise a similar approach based on the class means (centroids). Neither this approach nor the approach in part (a) uses parameters w. Hint: We briefly discussed this approach in class on the whiteboard. Why is this approach more computationally efficient than the approach (a)? Why may it perform poorly at test time?

Take the mean of each group corresponding to one value of y(i). For example, all the values with y(i) = 1 would average to a centroid c(1). Classify the new point to the group of the closest centroid.

This approach is more computationally efficient because we are not checking the distance from every point; instead, we are just checking the distances from the centroid.

This approach may perform poorly at test time because it does not account for frequency/variations of data within each group. This approach assumes that the data for each class is uniformly distributed around the centroid, which might not be the case. If the distribution is skewed or if there are multiple clusters within a class, then the centroid may not accurately represent the class. In addition, in cases where the classes are not well-separated and their centroids are close to each other, a new point could easily be misclassified.

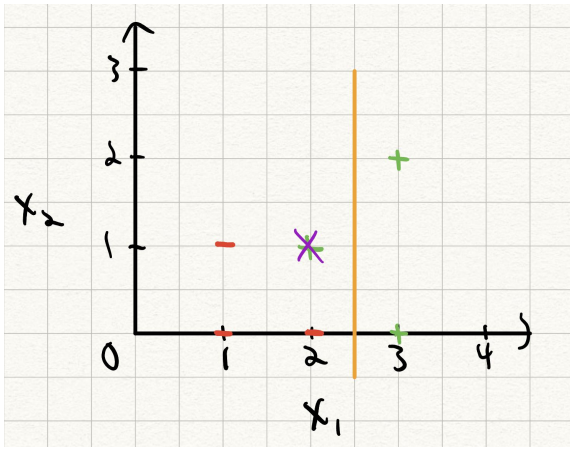
Problem 2) Consider the following training data:
a) Plot these six training points. Are the classes {+, —} linearly separable?



Yes, the classes are linearly separable. One example of a line that separates the classes is the orange line.
Equation of the orange line: x2 = -x1 + 2.5
-x1 - x2 + 2.5 = 0

b) Construct the weight vector (w1, w2, b) of the maximum margin hyperplane by inspection. Determine the distances between the hyperplane and each of the six training points. What is the margin of your hyperplane? Which training points are support vectors?
The margin of the hyperplane is (√2)/4.
Training points (1, 1), (2, 0), (2, 1), and (3, 0) are support vectors

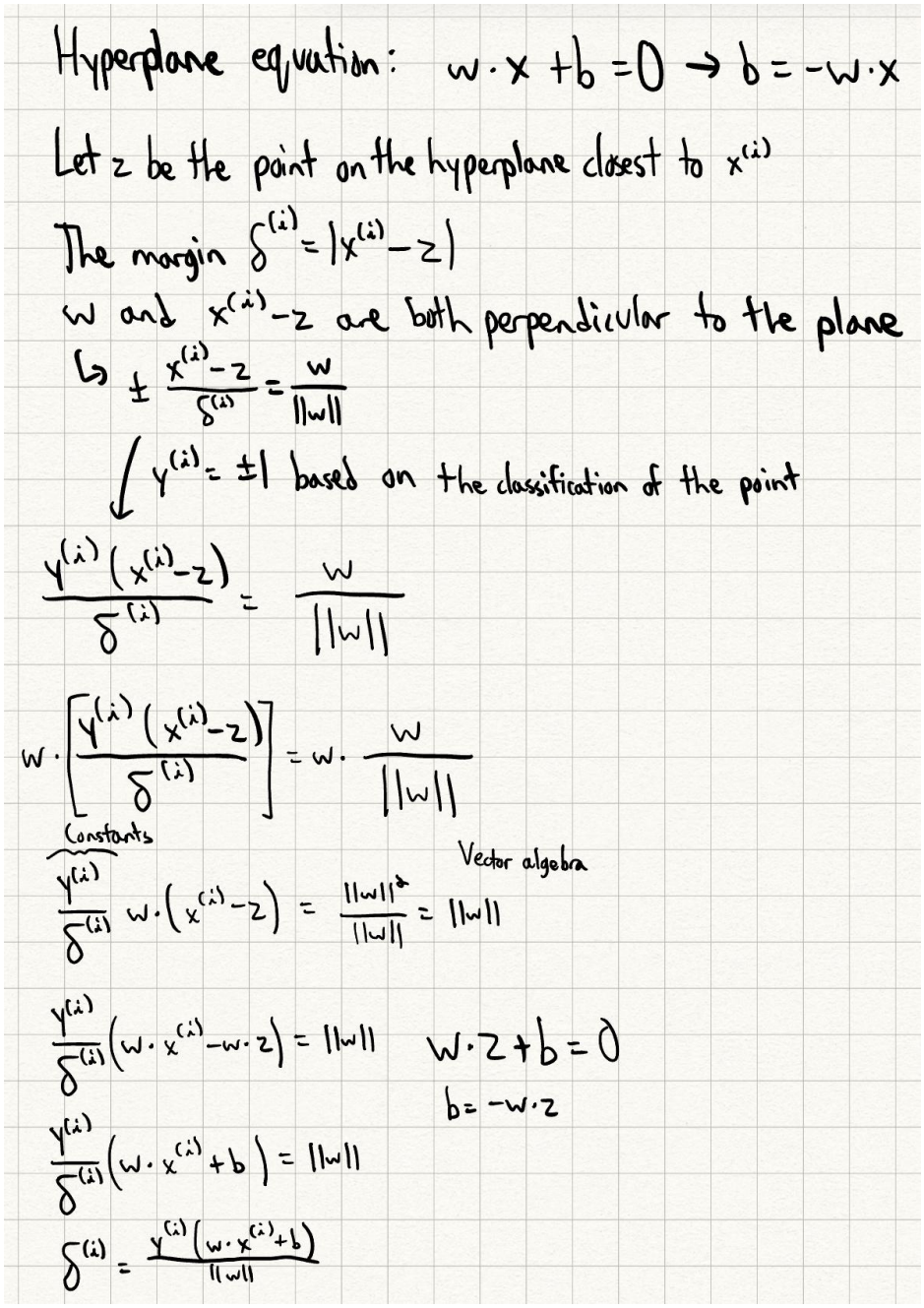
c) If you remove one of the support vectors, does the size of the optimal margin decrease, stay the same, or increase?



The size of the optimal margin increases because there are fewer support vectors. These new support vectors are more spread out than the original support vectors. For example, if you

remove the positive point (2, 1), the new optimal hyperplane would be the orange line depicted above. The new optimal margin becomes 1, which is more than (√2)/4

Problem 3) Let w, b be a hyperplane. Show that the distance between a training example x(i), y(i) and the hyperplane is given by $\delta(i) = y(i)(w \cdot x(i) + b)/|w|$. Provide all details without making the derivation longer or more complicated than it has to be.



Problem 4) Beginning with the optimization problem on slide 9 on the slide in the SVM lecture, prove that the solution to the hard-margin optimization problem on page 11 provides a separating hyper-plane with maximum margin

a) Suppose that w*, b* is optimal for the hard margin optimization problem on page 11. We must show that w*, b* gives a hyperplane that maximizes the margin. First show that the margin for w*, b* (distance from hyperplane to nearest training example) is 1/||w*||. To do this, you'll want to use the explicit expression derived in class for the distance between a training example x(i), y(i) and the the hyperplane defined by w* and b*. You'll also need to make use of the fact w*, b satisfy the constraints in the hard margin optimization problem on page 11 and that it meets at least one of the constraints with equality (since it is optimal).

a) From question 3 we know that the distance from a point to the hyperplane (margin) is equal to $\frac{y^{(i)}(w \cdot x^{(i)} + b)}{|w|}$. From slide 11, w^* and b^* are optimal. This necessarily means that there is an $j \leq m$ that satisfies $y^{(j)}(w \cdot x^{(j)} + b) = 1$

The point that corresponds to the point closest to the hyperplane is necessarily the one associated with j.

Using the equality $y^{(j)}(w \cdot x^{(j)} + b) = 1$, we can substitute the numerator in

$$\frac{y^{(i)}(w \cdot x^{(i)} + b)}{|w|} \geq \delta \rightarrow \frac{1}{|w^*|} \geq \delta \quad \text{(} \delta \text{ ← margin)}$$

Thus, w^* and b^* maximize δ (margin)

b) Now let z, d be any other separating hyperplane, and let M denote its margin for the data set. Define z' = z/||z||M and d' = d/||z||M. Note that ||z'|| = 1/M. Show that z', d' is a feasible solution for the hard-margin optimization problem

b) $z' = \frac{z}{\|z\|M}$ $d' = \frac{d}{\|z\|M}$ $\|z'\| = \frac{1}{M}$

The distance between $x^{(i)}$ and the hyperplane is $\frac{y^{(i)}(z \cdot x^{(i)} + d)}{\|z\|}$ which is $\geq M$

Dividing both sides by $\|z\|M$,

$$\frac{\frac{y^{(i)}}{\|z\|M}(z \cdot x^{(i)} + d)}{\|z\|} \geq \frac{M}{\|z\|M}$$

$$\frac{y^{(i)}\left(\frac{z}{\|z\|M} \cdot x^{(i)} + \frac{d}{\|z\|M}\right)}{\|z\|} \geq \frac{1}{\|z\|}$$

Substituting,

$$y^{(i)}(z' \cdot x^{(i)} + d') \geq 1$$

(2) Note that if w*, *b** is an optimal solution to the optimization problem on page 11, and z',d' is some other feasible solution to the optimization problem on page 11, then all of the following hold:

(i) y(i)[w*.x(i)+b*] >= 1. for i=1,...,m

(ii) y(i)[z'.x(i)+d'] >= 1 for i=1,...,m iff z', d' is a feasible solution ∴ z', d' is a feasible solution

(ii) ||w*||^2 <= ||z'||^2

c) Use the results from (a)-(b) to show that the margin for w*, b* is ³ the margin for z, d.

c) Since z', d' is a feasible solution, $\|w^*\|^2 \leq \|z\|^2$

From part a, we showed that the margin for w^*, b^* is equal to $\frac{1}{\|w^*\|}$

Let us call this margin M^*: $M^* = \frac{1}{\|w^*\|} \rightarrow \|w^*\| = \frac{1}{M^*}$

From the note in part b, we know that $\|z'\| = \frac{1}{M}$.

Substituting $\|w^*\|$ and $\|z'\|$:

$$\left(\frac{1}{M^*}\right)^2 \leq \left(\frac{1}{M}\right)^2$$

$$M^2 \leq M^{*2}$$

$$M^{*2} \geq M^2$$) Margin is always positive

$$M^* \geq M$$

∴ The margin for w^* b^* is ≥ the margin for z, d.

Problem 5) Consider a supervised machine learning problem with two features (x1, x2) and 4 training points x(1), x(2), x(3), x(4):

data	class	x_1	x_2
$\underline{x}^{(1)}$	+	1	0
$\underline{x}^{(2)}$	+	3	2
$\underline{x}^{(3)}$	–	1	2
$\underline{x}^{(4)}$	–	3	0

Denote w0 & w1 for the weights and b for the bias.

a) Prove by contradiction that there is no solution that satisfies the constraints for the hard margin SVM problem.

Assume that there is a $(w_1, w_2), b$ that satisfies

$y^{(i)}(\mathbf{w} \cdot \mathbf{x}^{(i)} + b) \geq 1$ for all i = 1,2,…,m

$-1[(w_0, w_1)\cdot(1,2)+b] \geq 1$

$-1[(w_0, w_1)\cdot(3,0)+b] \geq 1$

$1[(w_0, w_1)\cdot(1,0)+b] \geq 1$

$1[(w_0, w_1)\cdot(3,2)+b] \geq 1$

$\downarrow$

$-w_0 - 2w_1 - b \geq 1$

$-3w_0 - 0w_1 - b \geq 1$

$w_0 + 0w_1 + b \geq 1$

$3w_0 + 2w_1 + b \geq 1$

} Summing all 4 inequalities

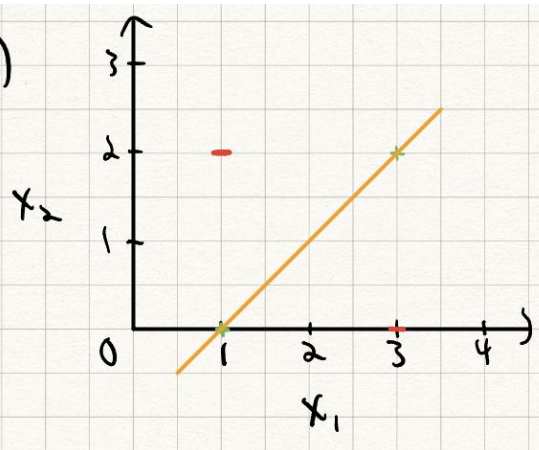
$0 \geq 4$

$\emptyset$ Contradiction!

$\therefore$ there is no $(w_0, w_1), b$ that satisfies the constraints for the hard margin SVM problem

b) Show that there is a solution for the soft margin SVM problem. Explicitly provide such a solution (w0,w1,b, ξ(1), ξ(2), ξ(3), ξ(4)).

b)



$x_2 = x_1 - 1$

$x_1 - x_2 - 1 = 0$

$w_0 = 1 \quad w_1 = -1 \quad b = -1$

$-w_0 - 2w_1 - b \geq 1 - \xi_3$

$-3w_0 - 0w_1 - b \geq 1 - \xi_4$

$w_0 + 0w_1 + b \geq 1 - \xi_1$

$3w_0 + 2w_1 + b \geq 1 - \xi_2$

$\downarrow$

$-(1) - 2(-1) - (-1) \geq 1 - \xi_3$

$-3(1) - 0(-1) - (-1) \geq 1 - \xi_4$

$(1) + 0(-1) + (-1) \geq 1 - \xi_1$

$3(1) + 2(-1) + (-1) \geq 1 - \xi_2$

$\downarrow$

$2 \geq 1 - \xi_3 \quad \xi_3 \geq -1$

$-2 \geq 1 - \xi_4 \rightarrow \xi_4 \geq 3$ A valid solution $\xi_1 = 1$, $\xi_2 = 1$, $\xi_3 = -1$, $\xi_4 = 3$

$0 \geq 1 - \xi_1 \quad \xi_1 \geq 1$

$0 \geq 1 - \xi_2 \quad \xi_2 \geq 1$

$(w_0, w_1, b, \xi_1, \xi_2, \xi_3, \xi_4)$

$= (1, -1, -1, 1, 1, -1, 3)$

(1, -1, -1, 1, 1, -1, 3)

Problem 6) What python libraries are available for SVM? Do they allow for hard and soft SVM? Do they allow for kernels?

scikit-learn
Both Hard/Soft SVM
Allows for kernels

cvxopt
Primarily used for hard-margin SVM but can be adapted for soft-margin.
Have to implement kernel methods manually

TensorFlow / PyTorch
Both Hard/Soft SVM
Have to implement kernel methods manually