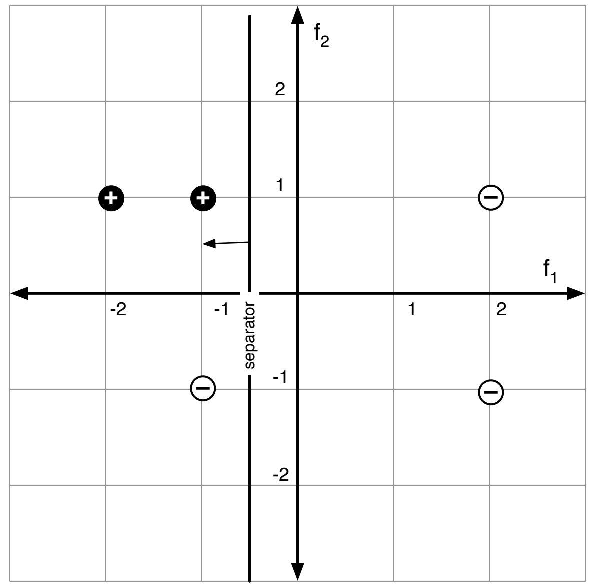
Recitation 12 Solutions

1.

18.17 The examples map from $[x_1, x_2]$ to $[x_1, x_1, x_2]$ coordinates as follows:
$[-1, -1]$ (negative) maps to $[-1, +1]$
$[-1, +1]$ (positive) maps to $[-1, -1]$
$[+1, -1]$ (positive) maps to $[+1, -1]$
$[+1, +1]$ (negative) maps to $[+1, +1]$
Thus, the positive examples have $x_1 x_2 = -1$ and the negative examples have $x_1 x_2 = +1$. The maximum margin separator is the line $x_1 x_2 = 0$, with a margin of 1. The separator corresponds to the $x_1 = 0$ *and* $x_2 = 0$ axes in the original space—this can be thought of as the limit of a hyperbolic separator with two branches.

2.



Data points are: Negative: (-1, -1) (2, 1) (2, -1) Positive: (-2, 1) (-1, 1)

Recall that the perceptron algorithm uses the extended form of the data points in which a 1 is added as the 0th component.

1. Assume that the initial value of the weight vector for the perceptron is $[0, 0, 1]$, that the data points are examined in the order given above and that the rate (step size) is 1.0. Give the weight vector after one iteration of the algorithm (one pass through all the data points):

   Only point $x_2 = (2, 1)$ is misclassified. Using the extended form $x'_2 = (1, 2, 1)$, we have

   $$[0, 0, 1] * x'_2 = +1$$

   We update the weight vector using the extended form (times $y_2 = -1$):

   $$w \leftarrow w + y_2 x'_2$$

   and we get the new weight vector

   $$w = [-1, -2, 0]$$

2. Draw the separator corresponding to the weights after this iteration on the graph at the top of the page.

3. Would the algorithm stop after this iteration or keep going? Explain.

   No. The new weight vector misclassifies the negative point (-1, -1), whose margin will be +1.

4. If we add a positive point at (1,-1) to the other points and retrain the perceptron, what would the perceptron algorithm do? Explain.

   The data is no longer linearly separable and so the perceptron would loop forever.