CS 188 Introduction to
Summer 2021 Artificial Intelligence

Exam Prep 6B

Q1. Perceptron

We would like to use a perceptron to train a classifier for datasets with 2 features per point and labels +1 or -1.

Consider the following labeled training data:

Features (x_1, x_2)	Label y^*
(-1,2)	1
(3,-1)	-1
(1,2)	-1
(3,1)	1

(a) Our two perceptron weights have been initialized to $w_1 = 2$ and $w_2 = -2$. After processing the first point with the perceptron algorithm, what will be the updated values for these weights?

(b) After how many steps will the perceptron algorithm converge? Write "never" if it will never converge.

Note: one steps means processing one point. Points are processed in order and then repeated, until convergence.

(c) Instead of the standard perceptron algorithm, we decide to treat the perceptron as a single node neural network and update the weights using gradient descent on the loss function.

The loss function for one data point is $Loss(y, y^*) = (y - y^*)^2$, where y^* is the training label for a given point and y is the output of our single node network for that point.

(i) Given a general activation function $g(z)$ and its derivative $g'(z)$, what is the derivative of the loss function with respect to w_1 in terms of g, g', y^*, x_1, x_2, w_1, and w_2?

$\frac{\partial Loss}{\partial w_1} =$

(ii) For this question, the specific activation function that we will use is:

$$g(z) = 1 \text{ if } z \geq 0 \text{ and } = -1 \text{ if } z < 0$$

Given the following gradient descent equation to update the weights given a single data point. With initial weights of $w_1 = 2$ and $w_2 = -2$, what are the updated weights after processing the first point?

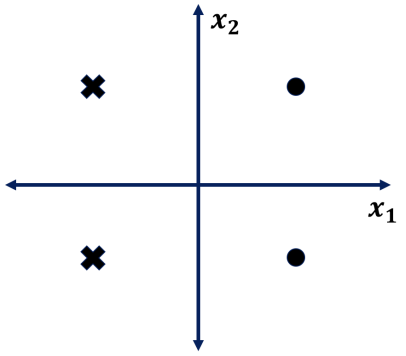
Gradient descent update equation: $w_i = w_i - \alpha \frac{\partial Loss}{\partial w_1}$

(iii) What is the most critical problem with this gradient descent training process with that activation function?

Q2. Perceptrons and Naive Bayes

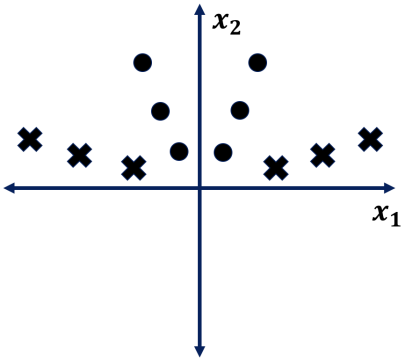
(a) For each of the datasets represented by the graphs below, please select the feature maps for which the perceptron algorithm can perfectly classify the data.

Each data point is in the form (x_1, x_2), and has some label Y, which is either a 1 (dot) or -1 (cross).



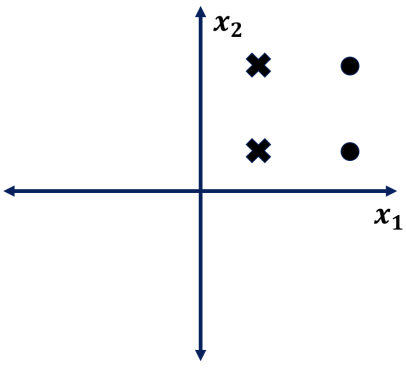
(i)

- ☐ $[x_1 \quad x_2 \quad 1]$
- ☐ $[x_1 \quad x_2 \quad x_1^2]$
- ☐ $[x_1 \quad x_2 \quad |x_1|]$
- ☐ $[x_1 \quad x_2 \quad Y]$
- ☐ $[x_1 \quad x_2]$



(ii)

- ☐ $[x_1 \quad x_2 \quad 1]$
- ☐ $[x_1 \quad x_2 \quad x_1^2]$
- ☐ $[x_1 \quad x_2 \quad |x_1|]$
- ☐ $[x_1 \quad x_2 \quad Y]$
- ☐ $[x_1 \quad x_2]$



(iii)

- ☐ $[x_1 \quad x_2 \quad 1]$
- ☐ $[x_1 \quad x_2 \quad x_1^2]$
- ☐ $[x_1 \quad x_2 \quad |x_1|]$
- ☐ $[x_1 \quad x_2 \quad Y]$
- ☐ $[x_1 \quad x_2]$

(b) Performing maximum likelihood estimation (MLE) to fit the parameters of a Bayes net to some given data (with no Laplace smoothing) leads to which of the following learning algorithms?

- ☐ Naive Bayes
- ☐ Perceptrons
- ☐ Kernelization
- ☐ Neural Networks
- ◯ None

(c) Suppose that we are trying to perform a binary classification task using Naive Bayes. Y is the label, and (X_1, X_2) are the features. The domain for the features is anywhere on the 3×3 grid centered at $(0, 0)$. In other words, X_1 and X_2 have the domain $\{-1, 0, 1\}$

Suppose that this is your dataset: (0, 1, +), (0, −1, −), (−1, 1, +), (-1, −1, −), (1, 0, +), (−1, 1, −), (0, 0, +). What is the learned value of each of the following? (Leave your answer as a simplified fraction)

(i) $P(Y = +)$

(ii) $P(X_1 = 1|Y = -)$

(iii) $P(X_2 = 0|Y = +)$

(d) Now, to decouple from the previous question, assume that the learned CPTs are below.

Y	X_1	$\Pr(X_1 \mid Y)$
+	−1	0.4
+	0	0.1
+	1	0.5
−	−1	0.6
−	0	0.3
−	1	0.1

Y	X_2	$\Pr(X_2 \mid Y)$
+	−1	0.2
+	0	0.2
+	1	0.6
−	−1	0.7
−	0	0.1
−	1	0.2

Y	$\Pr(Y)$
+	0.2
−	0.8

(i) What would be the predicted value for Y if the data point is at $(0, 0)$?

(ii) What would be the predicted value for Y if the data point is at $(1, -1)$?