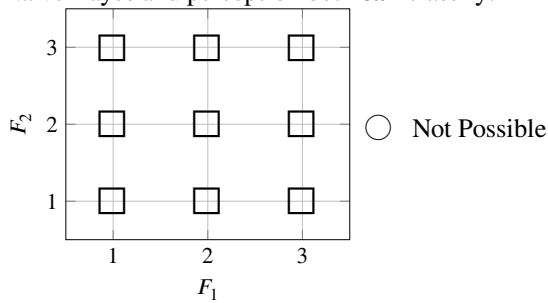


### Q1. Can You Classify Them?

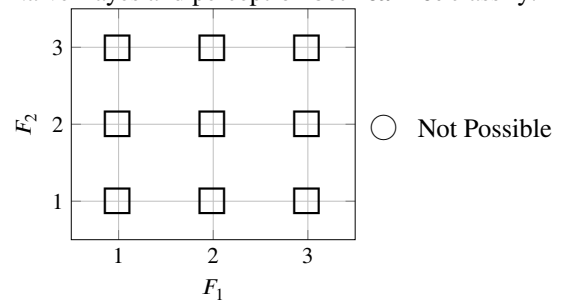
(a) For each of the 4 plots below, create a classification dataset which can or cannot be classified correctly by Naive Bayes and perceptron, as specified. Each dataset should consist of nine points represented by the boxes, shading the box  for positive class or leaving it blank  for negative class. Mark *Not Possible* if no such dataset is possible.

For *can* be classified by Naive Bayes, there should be some probability distributions  $P(Y)$  and  $P(F_1|Y), P(F_2|Y)$  for the class  $Y$  and features  $F_1, F_2$  that can correctly classify the data according to the Naive Bayes rule, and for *cannot* there should be no such distribution. For perceptron, assume that there is a bias feature in addition to  $F_1$  and  $F_2$ .

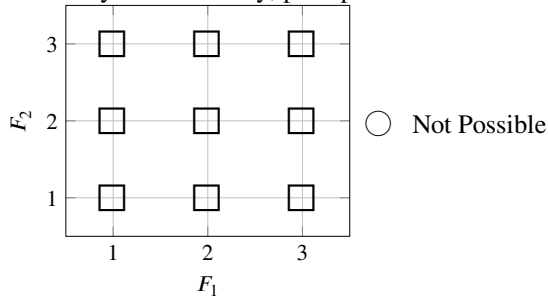
Naive Bayes and perceptron both **can** classify:



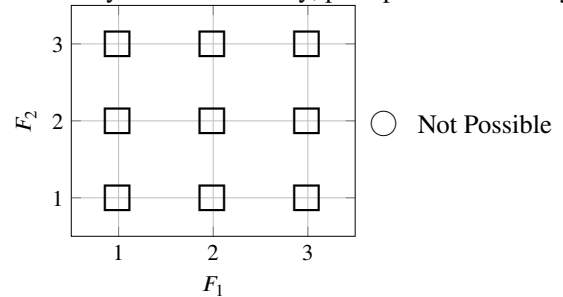
Naive Bayes and perceptron both **cannot** classify:



Naive Bayes **can** classify; perceptron **cannot** classify:



Naive Bayes **cannot** classify; perceptron **can** classify:



## Q2. 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?

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

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

### Q3. Naive Bayes: Pacman or Ghost?

You are standing by an exit as either Pacmen or ghosts come out of it. Every time someone comes out, you get two observations: a visual one and an auditory one, denoted by the random variables  $X_v$  and  $X_a$ , respectively. The visual observation informs you that the individual is either a Pacman ( $X_v = 1$ ) or a ghost ( $X_v = 0$ ). The auditory observation  $X_a$  is defined analogously. Your observations are a noisy measurement of the individual's true type, which is denoted by  $Y$ . After the individual comes out, you find out what they really are: either a Pacman ( $Y = 1$ ) or a ghost ( $Y = 0$ ). You have logged your observations and the true types of the first 20 individuals:

individual $i$	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
first observation $X_v^{(i)}$	0	0	1	0	1	0	0	1	1	1	0	1	1	0	1	1	1	0	0	0
second observation $X_a^{(i)}$	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
individual's type $Y^{(i)}$	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	0	0	0

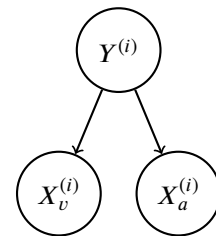
The superscript  $(i)$  denotes that the datum is the  $i$ th one. Now, the individual with  $i = 20$  comes out, and you want to predict the individual's type  $Y^{(20)}$  given that you observed  $X_v^{(20)} = 1$  and  $X_a^{(20)} = 1$ .

- (a) Assume that the types are independent, and that the observations are independent conditioned on the type. You can model this using naïve Bayes, with  $X_v^{(i)}$  and  $X_a^{(i)}$  as the features and  $Y^{(i)}$  as the labels. Assume the probability distributions take on the following form:

$$P(X_v^{(i)} = x_v | Y^{(i)} = y) = \begin{cases} p_v & \text{if } x_v = y \\ 1 - p_v & \text{if } x_v \neq y \end{cases}$$

$$P(X_a^{(i)} = x_a | Y^{(i)} = y) = \begin{cases} p_a & \text{if } x_a = y \\ 1 - p_a & \text{if } x_a \neq y \end{cases}$$

$$P(Y^{(i)} = 1) = q$$



for  $p_v, p_a, q \in [0, 1]$  and  $i \in \mathbb{N}$ .

- (i) What's the maximum likelihood estimate of  $p_v, p_a$  and  $q$ ?

$p_v =$  \_\_\_\_\_       $p_a =$  \_\_\_\_\_       $q =$  \_\_\_\_\_

- (ii) What is the probability that the next individual is Pacman given your observations? Express your answer in terms of the parameters  $p_v, p_a$  and  $q$  (you might not need all of them).

$P(Y^{(20)} = 1 | X_v^{(20)} = 1, X_a^{(20)} = 1) =$  \_\_\_\_\_

Now, assume that you are given additional information: you are told that the individuals are actually coming out of a bus that just arrived, and each bus carries *exactly* 9 individuals. Unlike before, the types of every 9 consecutive individuals are *conditionally* independent given the bus type, which is denoted by  $Z$ . Only after all of the 9 individuals have walked out, you find out the bus type: one that carries mostly Pacmans ( $Z = 1$ ) or one that carries mostly ghosts ( $Z = 0$ ). Thus, you only know the bus type in which the first 18 individuals came in:

individual $i$	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	
first observation $X_v^{(i)}$	0	0	1	0	1	0	0	1	1	1	0	1	1	0	1	1	1	0	0	0	
second observation $X_a^{(i)}$	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	
individual's type $Y^{(i)}$	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	0	0	0	
bus $j$										0										1	
bus type $Z^{(j)}$										0										1	

(b) You can model this using a variant of naïve bayes, where now 9 consecutive labels  $Y^{(i)}, \dots, Y^{(i+8)}$  are *conditionally* independent given the bus type  $Z^{(j)}$ , for bus  $j$  and individual  $i = 9j$ . Assume the probability distributions take on the following form:

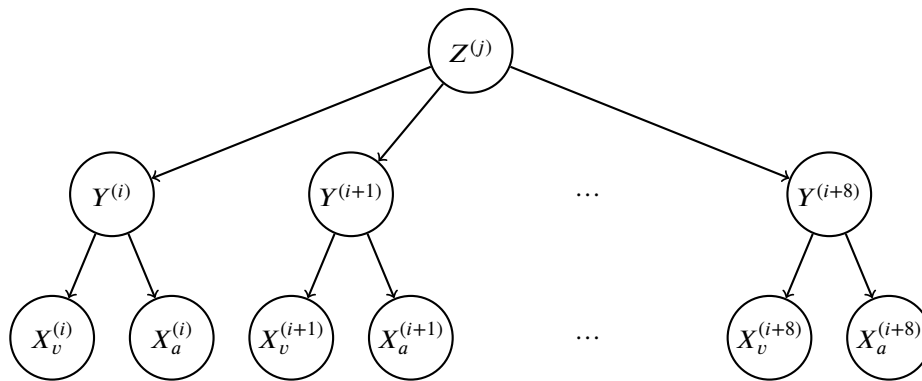
$$P(X_v^{(i)} = x_v | Y^{(i)} = y) = \begin{cases} p_v & \text{if } x_v = y \\ 1 - p_v & \text{if } x_v \neq y \end{cases}$$

$$P(X_a^{(i)} = x_a | Y^{(i)} = y) = \begin{cases} p_a & \text{if } x_a = y \\ 1 - p_a & \text{if } x_a \neq y \end{cases}$$

$$P(Y^{(i)} = 1 | Z^{(j)} = z) = \begin{cases} q_0 & \text{if } z = 0 \\ q_1 & \text{if } z = 1 \end{cases}$$

$$P(Z^{(j)} = 1) = r$$

for  $p, q_0, q_1, r \in [0, 1]$  and  $i, j \in \mathbb{N}$ .



(i) What's the maximum likelihood estimate of  $q_0, q_1$  and  $r$ ?

$q_0 =$  \_\_\_\_\_       $q_1 =$  \_\_\_\_\_       $r =$  \_\_\_\_\_

(ii) Compute the following joint probability. Simplify your answer as much as possible and express it in terms of the parameters  $p_v, p_a, q_0, q_1$  and  $r$  (you might not need all of them).

$$P(Y^{(20)} = 1, X_v^{(20)} = 1, X_a^{(20)} = 1, Y^{(19)} = 1, Y^{(18)} = 1) = \underline{\hspace{10cm}}$$