Q1. Naïve Bayes

You are given a naïve bayes model, shown below, with label $Y$ and features $X_1$ and $X_2$. The conditional probabilities for the model are parametrized by $p_1$, $p_2$ and $q$.

\[ \begin{array}{ccc}
X_1 & Y & P(X_1|Y) \\
0 & 0 & p_1 \\
1 & 0 & 1 - p_1 \\
0 & 1 & 1 - p_1 \\
1 & 1 & p_1 \\
\end{array} \quad \begin{array}{ccc}
X_2 & Y & P(X_2|Y) \\
0 & 0 & p_2 \\
1 & 0 & 1 - p_2 \\
0 & 1 & 1 - p_2 \\
1 & 1 & p_2 \\
\end{array} \]

Note that some of the parameters are shared (e.g. $P(X_1 = 0|Y = 0) = P(X_1 = 1|Y = 1) = p_1$).

(a) Given a new data point with $X_1 = 1$ and $X_2 = 1$, what is the probability that this point has label $Y = 1$? Express your answer in terms of the parameters $p_1$, $p_2$ and $q$ (you might not need all of them).

\[ P(Y = 1|X_1 = 1, X_2 = 1) = \]

(b) What are the maximum likelihood estimates for $p_1$, $p_2$ and $q$?

\[ p_1 = \quad p_2 = \quad q = \]
Q2. Machine Learning: Potpourri

(a) What is the minimum number of parameters needed to fully model a joint distribution \( P(Y, F_1, F_2, ..., F_n) \) over label \( Y \) and \( n \) features \( F_i \)? Assume binary class where each feature can possibly take on \( k \) distinct values.

(b) Under the Naive Bayes assumption, what is the minimum number of parameters needed to model a joint distribution \( P(Y, F_1, F_2, ..., F_n) \) over label \( Y \) and \( n \) features \( F_i \)? Assume binary class where each feature can take on \( k \) distinct values.

(c) You suspect that you are overfitting with your Naive Bayes with Laplace Smoothing. How would you adjust the strength \( k \) in Laplace Smoothing?

   - [ ] Increase \( k \)
   - [ ] Decrease \( k \)

(d) While using Naive Bayes with Laplace Smoothing, increasing the strength \( k \) in Laplace Smoothing can:

   - [ ] Increase training error
   - [ ] Decrease training error
   - [ ] Increase validation error
   - [ ] Decrease validation error

(e) It is possible for the perceptron algorithm to never terminate on a dataset that is linearly separable in its feature space.

   - [ ] True
   - [ ] False

(f) If the perceptron algorithm terminates, then it is guaranteed to find a max-margin separating decision boundary.

   - [ ] True
   - [ ] False

(g) In multiclass perceptron, every weight \( w_y \) can be written as a linear combination of the training data feature vectors.

   - [ ] True
   - [ ] False

(h) For binary class classification, logistic regression produces a linear decision boundary.

   - [ ] True
   - [ ] False

(i) In the binary classification case, logistic regression is exactly equivalent to a single-layer neural network with a sigmoid activation and the cross-entropy loss function.
(j) (i) You train a linear classifier on 1,000 training points and discover that the training accuracy is only 50%. Which of the following, if done in isolation, has a good chance of improving your training accuracy?

☐ Add novel features  ☐ Train on more data  ☐ Train on less data

(ii) You now try training a neural network but you find that the training accuracy is still very low. Which of the following, if done in isolation, has a good chance of improving your training accuracy?

☐ Add more hidden layers  ☐ Add more units to the hidden layers
Q3. Neural Networks: Representation

For each of the piecewise-linear functions below, mark all networks from the list above that can represent the function exactly on the range \( x \in (-\infty, \infty) \). In the networks above, \( \text{relu} \) denotes the element-wise ReLU nonlinearity: \( \text{relu}(z) = \max(0, z) \). The networks \( G_i \) use 1-dimensional layers, while the networks \( H_i \) have some 2-dimensional intermediate layers.

(a)  
\[ \begin{array}{c}
\text{Networks:} \\
G_1: \quad x \rightarrow \ast \rightarrow y \\
G_2: \quad x \rightarrow \ast \rightarrow + \rightarrow y \\
G_3: \quad x \rightarrow \ast \rightarrow \text{relu} \rightarrow y \\
G_4: \quad x \rightarrow \ast \rightarrow \text{relu} \rightarrow + \rightarrow y \\
G_5: \quad x \rightarrow \ast \rightarrow \text{relu} \rightarrow + \rightarrow \ast \rightarrow + \rightarrow y \\
\end{array} \]

(b)  
\[ \begin{array}{c}
\text{Networks:} \\
H_1: \quad x \rightarrow \ast \rightarrow y \\
H_2: \quad x \rightarrow \ast \rightarrow + \rightarrow y \\
H_3: \quad x \rightarrow \ast \rightarrow \text{relu} \rightarrow y \\
H_4: \quad x \rightarrow \ast \rightarrow \text{relu} \rightarrow + \rightarrow y \\
H_5: \quad x \rightarrow \ast \rightarrow \text{relu} \rightarrow + \rightarrow \ast \rightarrow + \rightarrow y \\
\end{array} \]

\( \bigcirc \) None of the above