1 Naive Bayes

In this question, we will train a Naive Bayes classifier to predict class labels $Y$ as a function of input features $A$ and $B$. $Y$, $A$, and $B$ are all binary variables, with domains 0 and 1. We are given 10 training points from which we will estimate our distribution.

\[
\begin{array}{c|cccccc}
A & 1 & 1 & 1 & 1 & 0 & 1 \\
B & 1 & 0 & 0 & 1 & 1 & 1 \\
Y & 1 & 1 & 0 & 0 & 0 & 0 \\
\end{array}
\]

1. What are the maximum likelihood estimates for the tables $P(Y)$, $P(A|Y)$, and $P(B|Y)$?

\[
\begin{array}{c|c|c|c|c}
Y & P(Y) & A & Y & P(A|Y) & B & Y & P(B|Y) \\
0 & 3/5 & 0 & 0 & 1/6 & 0 & 0 & 1/3 \\
0 & 3/5 & 1 & 0 & 5/6 & 1 & 0 & 2/3 \\
1 & 2/5 & 0 & 1 & 1/4 & 0 & 1 & 1/4 \\
1 & 2/5 & 1 & 1 & 3/4 & 1 & 1 & 3/4 \\
\end{array}
\]

2. Consider a new data point $(A = 1, B = 1)$. What label would this classifier assign to this sample?

\[
P(Y = 0, A = 1, B = 1) = P(Y = 0)P(A = 1|Y = 0)P(B = 1|Y = 0) \\
= (3/5)(5/6)(2/3) \\
= 1/3
\]

\[
P(Y = 1, A = 1, B = 1) = P(Y = 1)P(A = 1|Y = 1)P(B = 1|Y = 1) \\
= (2/5)(3/4)(3/4) \\
= 9/40
\]

Our classifier will predict label 0.

3. Let’s use Laplace Smoothing to smooth out our distribution. Compute the new distribution for $P(A|Y)$ given Laplace Smoothing with $k = 2$. 

\[
P(A = 1|Y = 0) = P(A = 1) + k/N \\
P(A = 1|Y = 1) = P(A = 1) + k/N \\
= (1/6) + (2/5)(k/N)
\]
### 2 Perceptron

You want to predict if movies will be profitable based on their screenplays. You hire two critics A and B to read a script you have and rate it on a scale of 1 to 4. The critics are not perfect; here are five data points including the critics’ scores and the performance of the movie:

<table>
<thead>
<tr>
<th>#</th>
<th>Movie Name</th>
<th>A</th>
<th>B</th>
<th>Profit?</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Pellet Power</td>
<td>1</td>
<td>1</td>
<td>-</td>
</tr>
<tr>
<td>2</td>
<td>Ghosts!</td>
<td>3</td>
<td>2</td>
<td>+</td>
</tr>
<tr>
<td>3</td>
<td>Pac is Bac</td>
<td>2</td>
<td>4</td>
<td>+</td>
</tr>
<tr>
<td>4</td>
<td>Not a Pizza</td>
<td>3</td>
<td>4</td>
<td>+</td>
</tr>
<tr>
<td>5</td>
<td>Endless Maze</td>
<td>2</td>
<td>3</td>
<td>-</td>
</tr>
</tbody>
</table>

#### (a) First, you would like to examine the linear separability of the data. Plot the data on the 2D plane above; label profitable movies with + and non-profitable movies with – and determine if the data are linearly separable.

The data are linearly separable.

#### (b) Now you decide to use a perceptron to classify your data. Suppose you directly use the scores given above as features, together with a bias feature. That is \( f_0 = 1, f_1 = \text{score given by } A \) and \( f_2 = \text{score given by } B \).

Run one pass through the data with the perceptron algorithm, filling out the table below. Go through the data points in order, e.g. using data point #1 at step 1.

<table>
<thead>
<tr>
<th>step</th>
<th>Weights</th>
<th>Score</th>
<th>Correct?</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>([-1, 0, 0])</td>
<td>(-1 \cdot 1 + 0 \cdot 1 + 0 \cdot 1 = -1)</td>
<td>yes</td>
</tr>
<tr>
<td>2</td>
<td>([-1, 0, 0])</td>
<td>(-1 \cdot 1 + 0 \cdot 3 + 0 \cdot 2 = -1)</td>
<td>no</td>
</tr>
<tr>
<td>3</td>
<td>([0, 3, 2])</td>
<td>(0 \cdot 1 + 3 \cdot 2 + 2 \cdot 4 = 14)</td>
<td>yes</td>
</tr>
<tr>
<td>4</td>
<td>([0, 3, 2])</td>
<td>(0 \cdot 1 + 3 \cdot 3 + 2 \cdot 4 = 17)</td>
<td>yes</td>
</tr>
<tr>
<td>5</td>
<td>([0, 3, 2])</td>
<td>(0 \cdot 1 + 3 \cdot 2 + 2 \cdot 3 = 12)</td>
<td>no</td>
</tr>
</tbody>
</table>

Final weights: \([-1, 1, -1]\)

#### (c) Have weights been learned that separate the data? With the current weights, points will be classified as positive if \(-1 \cdot 1 + 1 \cdot A + -1 \cdot B \geq 0\), or \( A - B \geq 1 \). So we will have incorrect predictions for data points 3:

\[-1 \cdot 1 + 1 \cdot 2 + -1 \cdot 4 = -3 < 0\]

and 4:

\[-1 \cdot 1 + 1 \cdot 3 + -1 \cdot 4 = -2 < 0\]

Note that although point 2 has \( w \cdot f = 0 \), it will be classified as positive (since we classify as positive if \( w \cdot f \geq 0 \)).

#### (d) More generally, irrespective of the training data, you want to know if your features are powerful enough to allow you to handle a range of scenarios. Circle the scenarios for which a perceptron using the features above can indeed perfectly classify movies which are profitable according to the given rules:
(a) Your reviewers are awesome: if the total of their scores is more than 8, then the movie will definitely be profitable, and otherwise it won’t be. Can classify (consider weights $[-8, 1, 1]$)

(b) Your reviewers are art critics. Your movie will be profitable if and only if each reviewer gives either a score of 2 or a score of 3. Cannot classify

(c) Your reviewers have weird but different tastes. Your movie will be profitable if and only if both reviewers agree. Cannot classify