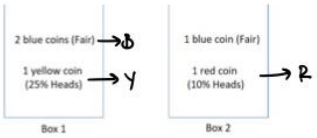
CS464 Introduction to Machine Learning
Fall 2024-25
Homework 1 Report

Question 1

1 Probability Review [30 pts]



There are 2 boxes in a room. The first box contains 2 blue coins and 1 yellow coin. The second box contains 1 blue and 1 red coin. The blue coins are fair. However, the yellow coin has 25% and red coin has 10% chance of landing heads.
You randomly select a coin from one of the boxes and toss it two times.
Question 1.1 [10 pts] What is the probability that you get two tails in a row?
Question 1.2 [10 pts] You toss the coin two times and got two tails in a row. What is the probability that the selected coin was fair?
Question 1.3 [10 pts] You toss the coin two times and got two tails in a row. What is the probability that the selected coin was the red one?
Note: Give your answers in 5 decimal points.

(1) X : denoting probability of 2 tails in a row

$P(X|B) = \frac{1}{2} \cdot \frac{1}{2} = 0.25$, $P(X|Y) = (0.75)(0.75) = 0.5625$, $P(X|R) = (0.9)(0.9) = 0.81$

$P(X|Box1) = P(B|Box1) \cdot P(X|B) + P(Y|Box1) \cdot P(X|Y) = \frac{2}{3} * 0.25 + \frac{1}{3} * 0.5625$

$= 0.1666667 + 0.1875$

$= 0.35416667$

$P(X|Box2) = P(B|Box2) \cdot P(X|B) + P(R|Box2) \cdot P(X|R)$

$= 0.25 * 0.5 + \frac{1}{2} * 0.81 = 0.125 + 0.405 = 0.53$

$P(X) = P(Box1) \cdot P(X|Box1) + P(Box2) \cdot P(X|Box2)$ Marginalization

$= \frac{1}{2} * 0.3541667 + \frac{1}{2} * 0.53 = 0.1770833 4 + 0.265$

$= 0.44208334 \simeq 0.44208$

(2)

$P(B) = P(B|Box1) * P(Box1) + P(B|Box2) P(Box2) = \frac{2}{3} * \frac{1}{2} + \frac{1}{2} * \frac{1}{2} = \frac{7}{12}$

$P(B|X) = \frac{P(X|B) * P(B)}{P(X)} = \frac{0.25 * 0.58333333}{0.44208334} = 0.58333333$

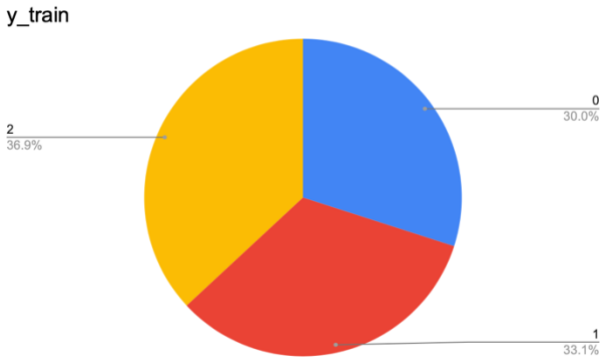
$= 0.32987747 \simeq 0.32988$

(3) $P(R) = P(R|Box1) * P(Box1) + P(R|Box2) * P(Box2) = 0 + \frac{1}{4} = 0.25$

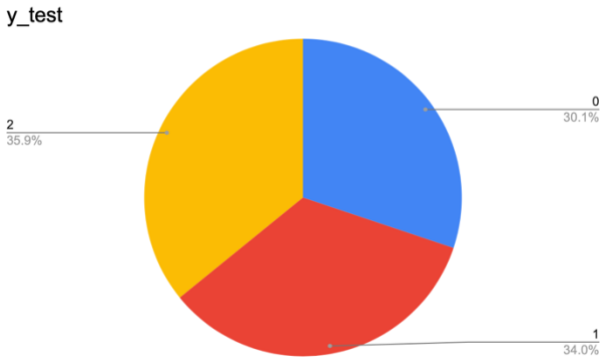
$P(R|X) = \frac{P(X|R) * P(R)}{P(X)} = \frac{0.81 * 0.25}{0.44208334} = 0.45805843 \simeq 0.45806$

Question 3.1

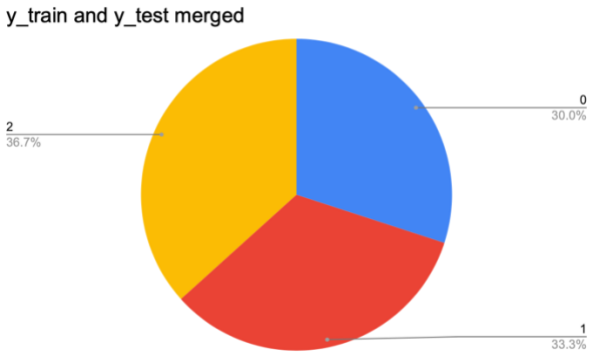
1.



{1: 0.331304347826087, 2: 0.3691304347826087, 0: 0.2995652173913044}



{2: 0.3585714285714286, 0: 0.30142857142857143, 1: 0.34}



{2: 0.36666666666666664, 0: 0.3, 1: 0.3333333333333333}

2. Priors = {1: 0.331304347826087, 2: 0.3691304347826087, 0: 0.2995652173913044}

3. It is nearly balanced. Yes, it affects. With an imbalanced training set, one class is much larger than the other, and a model may be skewed towards predicting the majority class and perform poorly on the minority class. That can lead to unreasonably high accuracy,

as most of the time, the model is simply guessing the most common class rather than learning something about the minority class. This can result in poor classification of the minority class.

4.
```
Occurrences of 'good' in positive documents: 207
Occurrences of 'bad' in positive documents: 12
ln(P(good | Y = positive)): -4.287609775546563
ln(P(bad | Y = positive)): -7.135421919023932
```

Question 3.2

```
Results without Smoothing:
Accuracy: 0.581
Confusion Matrix:
[138, 41, 32]
[76, 78, 84]
[32, 28, 191]
```

The expected (true) values are represented by rows, and columns represent the predicted values as 0, 1, and 2, respectively.

Question 3.3

```
Results with Smoothing:
Accuracy: 0.649
Confusion Matrix:
[151, 45, 15]
[73, 86, 79]
[13, 21, 217]
```

The expected (true) values are represented by rows, and columns represent the predicted values as 0, 1, and 2, respectively.

Applying the Dirichlet prior ($\alpha = 1$) greatly improved the accuracy of the multinomial Naive Bayes model from 58.1% (without smoothing) to 64.9%. The smoothing effect also means that each word, even unseen in the training set, has a small non-zero probability over all classes. Hence, this prevents the model from assigning a probability of zero to words during classification, hence reducing the number of misclassifications. The confusion matrix shows that the model correctly predicts more instances for all classes and decreases the number of misclassified instances, especially for underrepresented classes, with smoothing. Overall, the smoothing effect helps the model to generalize better, i.e., the model becomes more robust to unseen data. The small value α is added to each word count, which makes the model less likely to overfit the training data and instead gives a fairer estimate of word probabilities for each class so it can perform better classification.

Question 3.4

```
Results with Bernoulli:
Accuracy: 0.641
Confusion Matrix:
[113, 90, 8]
[29, 180, 29]
[19, 76, 156]
```

The expected (true) values are represented by rows, and columns represent the predicted values as 0, 1, and 2, respectively.

The Bernoulli model is a binary data model where we handle feature presence or absence, and the multinomial model is a count model that captures feature counts.

From the multinomial model with smoothing (at 3.3), the best result was obtained with an accuracy of 0.649, compared to 0.641 by the Bernoulli (binomial) model (at 3.4) and 0.581 by the unsmoothed multinomial model (at 3.1). Smoothing improved performance by reducing misclassifications and better handling rare features. However, the Bernoulli model with Laplacian smoothing was effective but slightly less capable of capturing the nuances in feature frequency than the smoothed multinomial approach. The lowest accuracy was found with the unsmoothed multinomial model, suggesting that the lack of smoothing negatively affected classification reliability. Overall, smoothing over the multinomial model gave the best accuracy, with the smallest number of misclassifications, compared to the smoothed Bernoulli and unsmoothed multinomial model in my case.