Mehmet Berk Türkçapar
21902570

CS-464: Introduction to Machine Learning
Homework 1

Part 1: The Online Shopping Case

Question 1.1

$$P(F_P) = (0.95 * 0.45) + (0.6 * 0.3) + (0.1 * 0.25) = 0.6325$$

Question 1.2

$$P(P|F_P) = \frac{0.95 * 0.45}{P(F_P)} = 0.6759$$

Question 1.3

$$P(P|F_N) = \frac{(1-0.95) * 0.45}{1-P(F_P)} = 0.0612$$

Part 2: Spam Email Detection

While implementing the code I have used numpy and pandas libraries. I have used numpy to deal with arrays and numerical operations and pandas to read and write csv files and create dataframes.

Question 2.1

1. The percentage of spam emails in the `y_train.cvs` is 28.6%.

2. The training data set is skewed towards the not spam (0) class. Having such an imbalance training set can have major impacts on the performance of the model. In this case, since the number of ‘not spam’ samples are much more than ‘spam’ samples the model will have a bias towards the ‘not spam’ class. Because of the majority of ‘not spam’ samples in the training data the model will be able to predict this majority class much more accurately while neglecting the ‘spam’ class.

3. Since the model will have a bias towards the majority class (in this case ‘not spam’ class), the model will mostly predict the given inputs as ‘not spam’. For example, if a test set includes 100 samples, 90 of them belong to the majority class, our biased model may classify all 100 samples as the majority class.

Then the accuracy of the model will be 90% which seems like a very good accuracy. However, this reported accuracy is misleading. In reality the model is not able to detect the minority class at all.

Question 2.2 (Multinomial Naive Bayes without Dirichlet Prior)

For this question after taking the logarithm of conditional probabilities I have checked and changed the -inf values to -1e12 in order to avoid any overflow issues. I have used a 'with' statement in order to prevent any warnings.

Confusion Matrix:

mnb0

	Not Spam	Spam
Not Spam	703.0	15.0
Spam	28.0	289.0

Accuracy and Number of Wrong Predictions:

```
---------Multinomial Naive Bayes with alpha: 0---------
Accuracy: 0.9585
Number of wrong predictions: 43
```

Question 2.3 (Multinomial Naive Bayes with Dirichlet Prior)

Confusion Matrix:

mnb5

	Not Spam	Spam
Not Spam	681.0	37.0
Spam	17.0	300.0

Accuracy and Number of Wrong Predictions:

```
---------Multinomial Naive Bayes with alpha: 5---------
Accuracy: 0.9478
Number of wrong predictions: 54
```

As it can be seen from the data the accuracy slightly decreased. Dirichlet prior may have caused some bias in the model. In my case, the alpha value chosen may not have been a correct assumption about the distribution of data. This may have lowered the accuracy. Specifically, from the confusion matrices, it can be seen that the correct 'Not Spam' predictions lowered from 703 to 681 whereas the correct 'Spam' predictions increased from 289 to 300. As stated before, the training data set is skewed towards 'Not Spam' class. Introducing the Dirichlet prior caused a bias towards the 'Spam' class.

Question 2.4 (Bernoulli Naive Bayes)

Confusion Matrix:

bnb

	Not Spam	Spam
Not Spam	695	23
Spam	32	285

Accuracy and Number of Wrong Predictions:

```
---------Bernoulli Naive Bayes---------
Accuracy: 0.9469
Number of wrong predictions: 55
```

The Bernoulli Naive Bayes has a lower accuracy than both of the previous models. Bernoulli Naive Bayes is more suitable in cases in which the presence of an attribute can be indicated with binary variables. In our case the frequency of words is important. So, the Bernoulli Naive Bayes is less suitable for our case. As a result, Bernoulli Naive Bayes performed slightly worse than the Multinomial Naive Bayes model.

Mehmet Berk Türkçapar
21902570

Question 2.5

As explained in the previous answers, the Multinomial Naive Bayes model is more suitable for this case. In the given scenario the frequency of words is important so we are dealing with discrete count data. So, it is not suitable for the Bernoulli Naive Bayes Model. Accuracy may be a deceiving performance metric for our case since both the train and test data sets are skewed towards the ‘Not Spam’ class. Because of this skewed training data the model may be biased towards ‘Not Spam’ class and since the test data is also skewed towards ‘Not Spam’ class a high accuracy can be seen while in fact the model is not good at detecting ‘Spam’ classes. However, as it can be seen from the confusion matrices the models had many correct ‘Spam’ classifications as well.