Cemhan Kaan Özaltan
21902695

Homework 1 Report

Student - Grade	H	L	F
S_M	87%	21%	4%
S_U	13%	79%	96%
Total	64%	24%	12%

Fig. 1. Probability table.

Question 1.1

81% of high grades belong to motivated students, and high grades make up 64% of all grades. Therefore, their product is the probability that a student is motivated and got a high grade. Applying this to all grades (total probability):

$$P(S_M) = 0.87 \cdot 0.64 + 0.21 \cdot 0.24 + 0.04 \cdot 0.12 = 0.612$$

Question 1.2

By the Bayes theorem:

$$P(H|S_M) = \frac{P(S_M|H)P(H)}{P(S_M)}$$

And:

$$P(S_M) = 0.612$$
$$P(H) = 0.64$$
$$P(S_M|H) = 0.87$$

Therefore:

$$P(H|S_M) = \frac{0.87 \cdot 0.64}{0.612} = 0.9098$$

Question 1.3

By the Bayes theorem:

$$P(H|S_U) = \frac{P(S_U|H)P(H)}{P(S_U)}$$

And:

$$P(S_U) = 1 - P(S_M) = 1 - 0.612 = 0.388$$
$$P(H) = 0.64$$
$$P(S_U|H) = 0.13$$

Therefore:

$$P(H|S_U) = \frac{0.13 \cdot 0.64}{0.388} = 0.2144$$

Question 2.1
1. Sample counts for classes (training data):
Athletics (0): 77
Cricket (1): 86
Football (2): 198
Rugby (3): 114
Tennis (4): 77

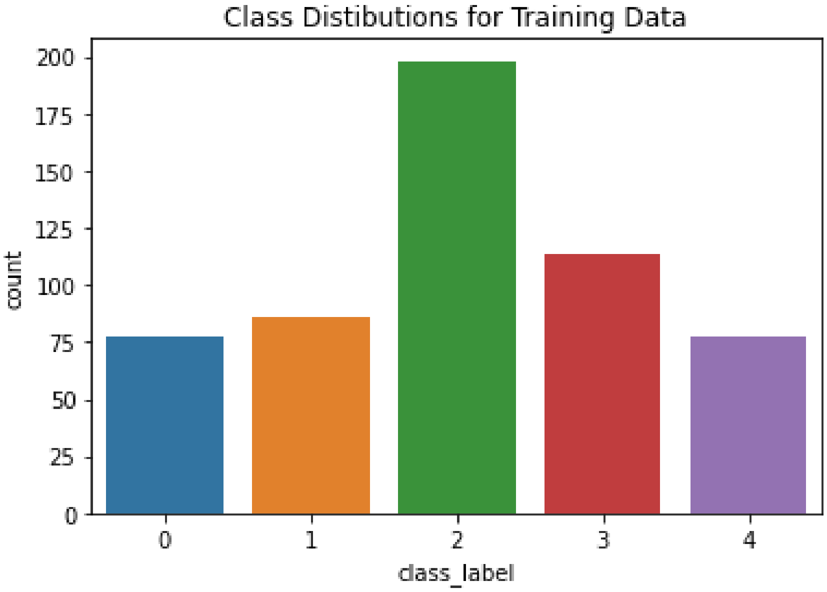

Fig. 2. Class distributions for training data.

2. The dataset is slightly skewed towards the football (2) class. This creates a problem for the Naïve Bayes classifier as it is trained on this data, its parameters take values accordingly, and therefore may develop a bias towards the football class while predicting. In order to mitigate this and make the distribution more uniform, we can use additive smoothing (Dirichlet prior) by adding an equal number of identical samples to each class.

3. They have similar distributions as the following information shows:
Sample counts for classes (validation data):

Athletics (0): 24
Cricket (1): 38
Football (2): 67
Rugby (3): 33
Tennis (4): 23

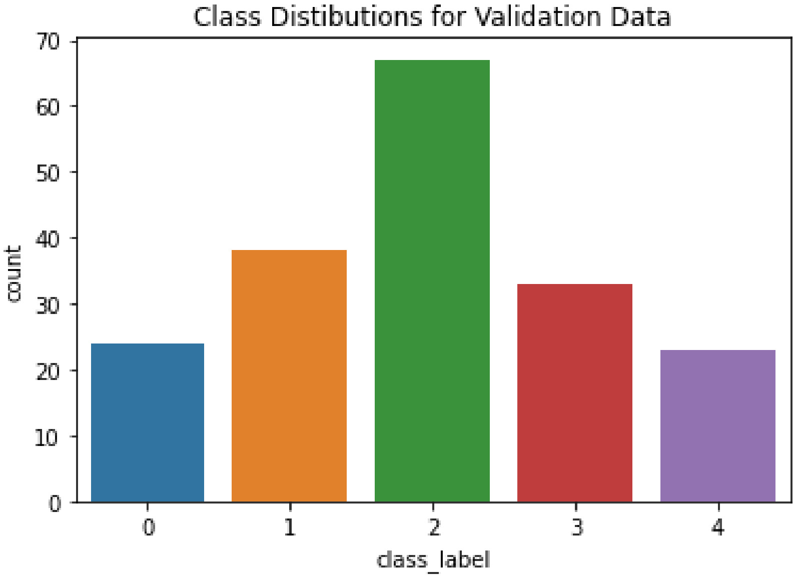

Fig. 3. Class distributions for validation data.

This means that the training and validation split is a good one in our case. If this split was bad and the class distributions of the training and validation sets were different, the π parameter would be misleading. For example, $\pi_{y=y_k}$ where $y_k = football$ would be biased as the N_{y_k}value would be considerably higher than the others due to the training dataset, if the class distribution of the validation dataset was completely even (unlike the training dataset which is skewed towards the football class).

4. In the case of this assignment, the reported accuracy is not affected since the class distributions of the training and validation datasets are almost identical. However, in another dataset, the accuracy may be affected. The extent of this effect would be determined by the class distribution differences between the training and validation datasets.

Question 2.2
Accuracy: 96.75675675675676%
Wrong prediction count: 6
Note: Here, $-\infty$ is simulated with the value -10^{15} (NINF variable).

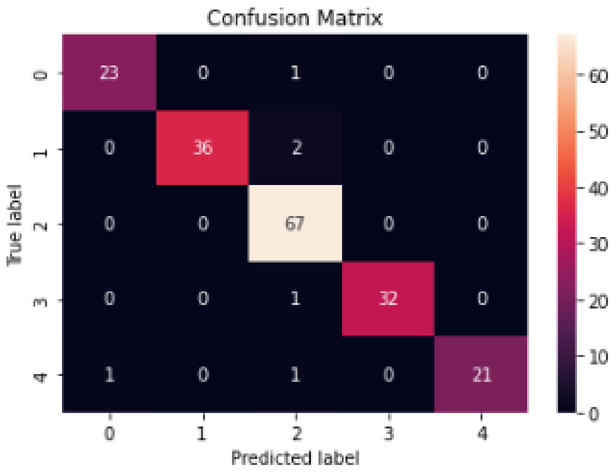

Fig. 4. Confusion matrix.

Question 2.3

Accuracy (with Dirichlet prior): 97.2972972972973%
Wrong prediction count (with Dirichlet prior): 5

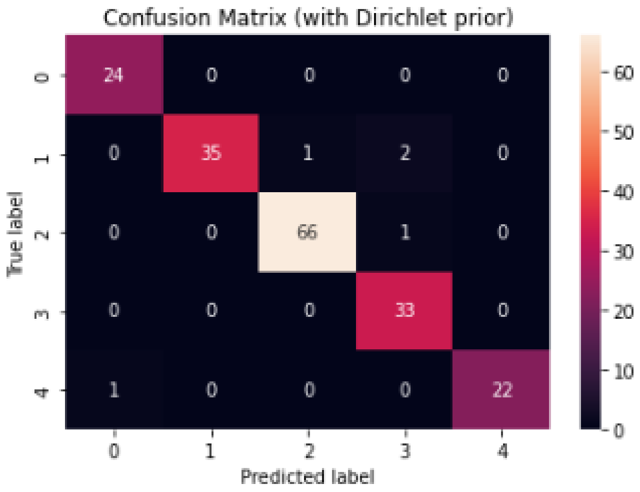

Fig. 5. Confusion matrix (with Dirichlet prior).

Question 2.4

When we compare the results of Questions 2.2 and 2.3, we see that the accuracy increased by around 0.5% as there exists one less wrong prediction. This is due to the usage of the Dirichlet prior (additive smoothing). By adding an instance for each word in each sample in the dataset, the zero probabilities of words that do not exist in a specific sample are removed since they now have an instance. This is useful in our case since many zero values exist in the features of our dataset samples. This is also due to the fact that there are no stop words (common words for the problem domain). This means that the counted words are rarer in nature, and therefore are more likely to take a zero value, making additive smoothing more useful.

Note: In Question 2.2, when the power of 10 is chosen as an even number, even though the resulting number is not made positive since the minus is not under the power, the accuracy changes and becomes the same with the one of Question 2.3 (even when the even power is smaller than the odd one). I am not quite sure what the reason for this is, but since when I use a regular large negative number with no powers, I get the same results in the report, I used an even power in my implementation.