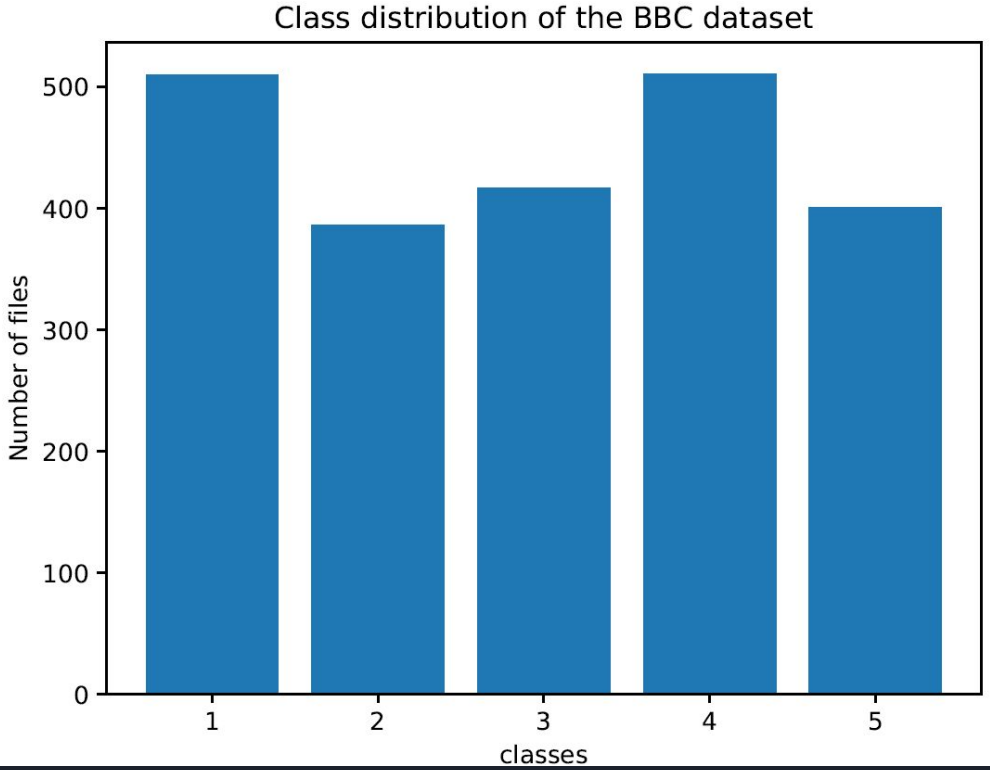
# SOEN 472 A1

Abdalla Osman
Juan Sebastian Hoyos
Samil Mehedi

# Task 1 dataset




1=business (510)

2=entertainment (386)

3=politics (417)

4=sports (511)

5=tech (401)

The dataset is balanced

```
i)
Number of zeros in business: 18500
 With percentage of 5.054879598232705
Number of zeros in entertainment: 18814
 With percentage of 5.140675933035141
Number of zeros in politics: 19275
 With percentage of 5.266638067888399
Number of zeros in sport: 19939
 With percentage of 5.448067259954698
Number of zeros in tech: 18490
 With percentage of 5.0521472308823085
```

Looking at the this we see that we have a lot of words that appear zero times in the files. If any of these words appear on the predict it will cause the probability of that class to be zero.

By applying a smoothing we can ensure no words “appear” zero times

```
i)
Number of zeros in business: 0
 With percentage of 0.0
Number of zeros in entertainment: 0
 With percentage of 0.0
Number of zeros in politics: 0
 With percentage of 0.0
Number of zeros in sport: 0
 With percentage of 0.0
Number of zeros in tech: 0
 With percentage of 0.0
```

c)

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.96 | 0.95 | 0.95 | 97 |
| 1 | 1.00 | 0.94 | 0.97 | 70 |
| 2 | 0.95 | 0.97 | 0.96 | 87 |
| 3 | 0.99 | 1.00 | 1.00 | 107 |
| 4 | 0.95 | 0.99 | 0.97 | 84 |
| accuracy | | | 0.97 | 445 |
| macro avg | 0.97 | 0.97 | 0.97 | 445 |
| weighted avg | 0.97 | 0.97 | 0.97 | 445 |

Nothing changes for these two. Normal because nothing changed in the training. Not even the training set.

c)

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.96 | 0.95 | 0.95 | 97 |
| 1 | 1.00 | 0.94 | 0.97 | 70 |
| 2 | 0.95 | 0.97 | 0.96 | 87 |
| 3 | 0.99 | 1.00 | 1.00 | 107 |
| 4 | 0.95 | 0.99 | 0.97 | 84 |
| accuracy | | | 0.97 | 445 |
| macro avg | 0.97 | 0.97 | 0.97 | 445 |
| weighted avg | 0.97 | 0.97 | 0.97 | 445 |

Slightly more performant, because of the applied smoothing.

**As the smoothing value increases. There is a larger bias**

Same performance. Why? We applied 0.9 smoothing

c)

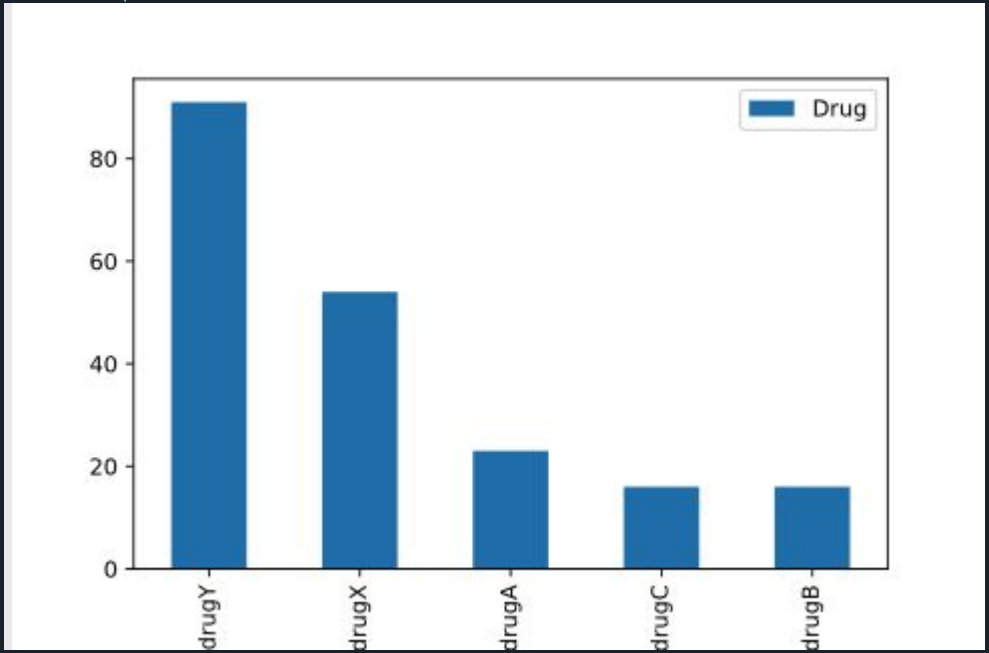
| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.97 | 0.96 | 0.96 | 97 |
| 1 | 1.00 | 0.97 | 0.99 | 70 |
| 2 | 0.98 | 0.98 | 0.98 | 87 |
| 3 | 1.00 | 1.00 | 1.00 | 107 |
| 4 | 0.97 | 1.00 | 0.98 | 84 |
| accuracy | | | 0.98 | 445 |
| macro avg | 0.98 | 0.98 | 0.98 | 445 |
| weighted avg | 0.98 | 0.98 | 0.98 | 445 |

Decrease in performance. If the smoothing is too big the difference between probabilities is smaller

c)

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.96 | 0.95 | 0.95 | 97 |
| 1 | 1.00 | 0.94 | 0.97 | 70 |
| 2 | 0.95 | 0.97 | 0.96 | 87 |
| 3 | 0.99 | 1.00 | 1.00 | 107 |
| 4 | 0.95 | 0.99 | 0.97 | 84 |
| accuracy | | | 0.97 | 445 |
| macro avg | 0.97 | 0.97 | 0.97 | 445 |
| weighted avg | 0.97 | 0.97 | 0.97 | 445 |

# Task 2 Dataset




- The features that are provided within the dataset are age, sex, blood pressure and cholesterol of the patients and the target is the drug that the patient uses

- This is the distribution of the instances in each class.

- As shown in the chart, the dataset is unbalanced

# Discussion

```
************************Averages************************


GaussianNB average accuracy, average macro-average F1 and average weighted-average F1 respectively:
0.8800000000000001 | 0.8678341013824884 | 0.8873917050691243

Base Decision Tree average accuracy, average macro-average F1 and average weighted-average F1 respectively:
0.9800000000000001 | 0.9892930679478381 | 0.9798901853122854

Top Decision Tree average accuracy, average macro-average F1 and average weighted-average F1 respectively:
0.9800000000000001 | 0.9892930679478381 | 0.9798901853122854

Perceptron average accuracy, average macro-average F1 and average weighted-average F1 respectively:
0.7 | 0.4047008547008547 | 0.6491452991452992

Base MLP average accuracy, average macro-average F1 and average weighted-average F1 respectively:
0.6079999999999999 | 0.2557158550990678 | 0.5156227984544657

Top MLP average accuracy, average macro-average F1 and average weighted-average F1 respectively:
0.6399999999999999 | 0.31953585070744384 | 0.5768446976550362

************************Standard Deviations************************


Standard deviation GaussianNB: 1.1102230246251565e-16 | 0.0 | 1.1102230246251565e-16

Standard deviation Base Decision Tree: 1.1102230246251565e-16 | 0.0 | 1.1102230246251565e-16

Standard deviation Top Decision Tree: 1.1102230246251565e-16 | 0.0 | 1.1102230246251565e-16

Standard deviation Perceptron: 0.0 | 0.0 | 1.1102230246251565e-16

Standard deviation Base MLP: 0.016000000000000014 | 0.010061684008796281 | 0.0178692467345062

Standard deviation Top MLP: 0.02529822128134706 | 0.041299714350699464 | 0.0313385026621842a)
```

Does the same model give you the same performance every time?

The performance is almost the same every time as the tests are being run on the same training set, which further explains why the average standard deviation of the models is approximately 0. An interesting observation is that the Base MLP and the Top MLP have a larger standard deviation compared to the other models. This can be because there is a minor disperse as the MLPs reach a global optima stochastically.

Accuracy-wise, the models for GaussianNB, Base Decision Tree and Top Decision Tree had a much better accuracy than Perceptron, Base MLP and Top MLP. Reasons being is that they are trained to predict Drug X and Y, since the dataset is unbalanced.

# Teamwork Contribution

Juan Sebastian Hoyos:

Task 1

Abdalla Osman:

Task 2 Question 6d till the end

Samil Mehedi:

Task 2 Question 1 to Question 6c