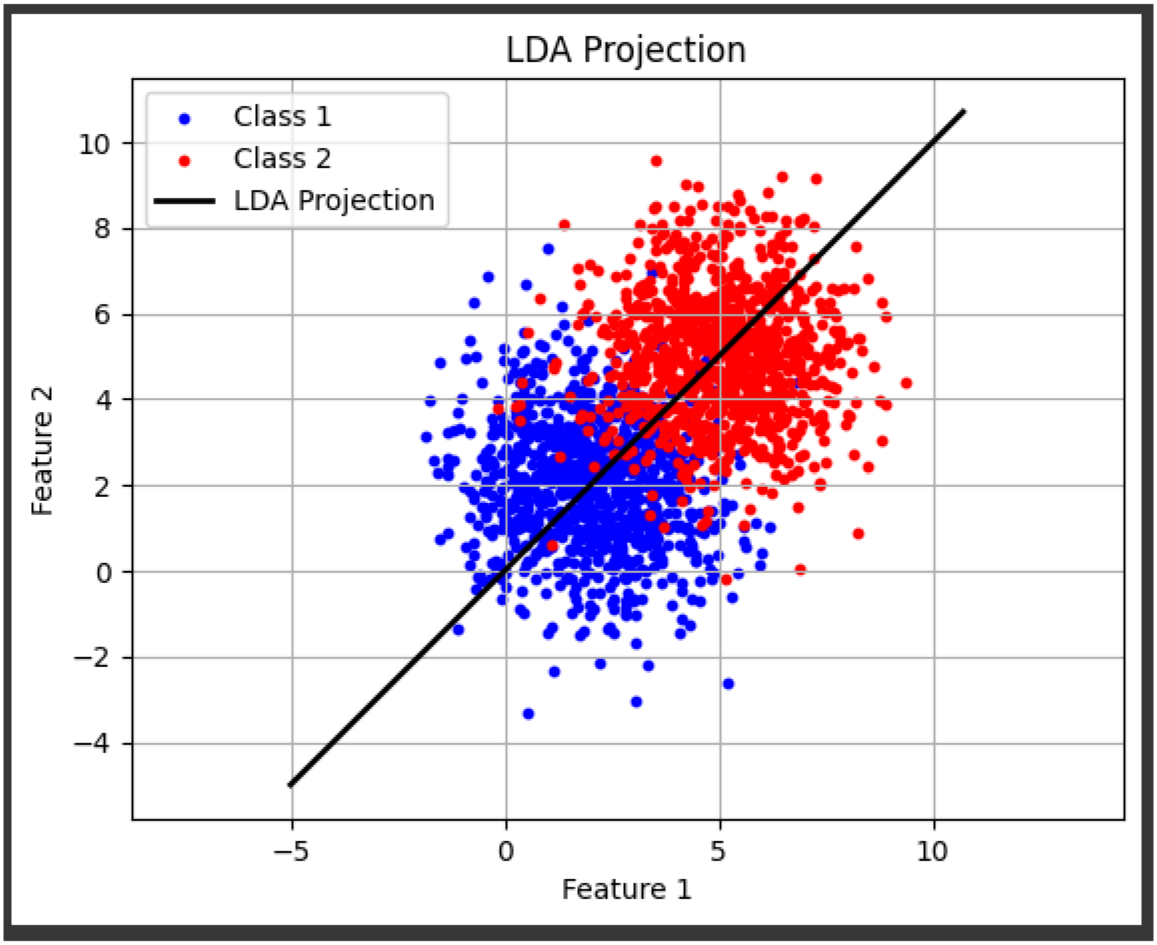
Assignment 4 Report

QUESTION-1:

COLAB: ∞ B22CS044_myLDA.ipynb

TASK-2: ∞ B22CS044_all.ipynb

To do Task 2, we used the given dataset and helper code to make the Linear Discriminant Analysis (LDA) projection vector visible. We put the LDA projection vector on a 2D scatter plot next to the original data points after computing it. This visualization made it easy to see which way the data is most split, which shows how powerful the LDA transformation is at telling the difference. The drawn vector, which started at zero, was a short way to show how the complexity was reduced. It showed how well LDA worked at making it easier to tell the difference between classes and helping with classification tasks.



TASK-3:

Linear Discriminant Analysis (LDA) was used to test the 1-NN (1-Nearest Neighbour) classifier's performance on both the original and forecast data. The classifier was accurate 88.75% of the time with the original data, but only 88.0% of the time with the forecast data.

Based on the results, it looks like there was a small drop in accuracy when the forecast data was used instead of the original data. Even though the drop in accuracy isn't very big, it shows that the LDA transformation might not have made it much easier to tell the classes apart in this dataset.

Based on this finding, the original feature space may already have enough differentiating information for the 1-NN classifier to be very accurate in this case. Another possibility is that the LDA projection may have caused some data loss or warping, which would have made the classification performance a little worse. More research, like looking into different classification methods or fine-tuning the LDA parameters, could help us figure out how to make the classification work better on this dataset.

QUESTION-2: B22CS044_all.ipynb

In this question we are given a dataset and based on that we have to develop a model that predicts whether the person plays outdoors or not. For this we use naive bayes algorithm and make our predictions.

TASK-0:

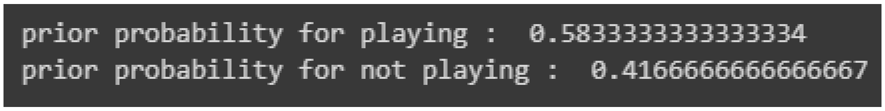
In this task we load the dataset in a dataframe and split the dataset into 12 training and 2 testing samples.

TASK-1:

Here we have to calculate the prior probabilities i.e. probabilities for playing = yes, playing = no in the whole training dataset.

We calculate the probability using the formula:
(Total occurrence of yes/no)/(Total samples)

According to my training dataset i have received the following probabilities:

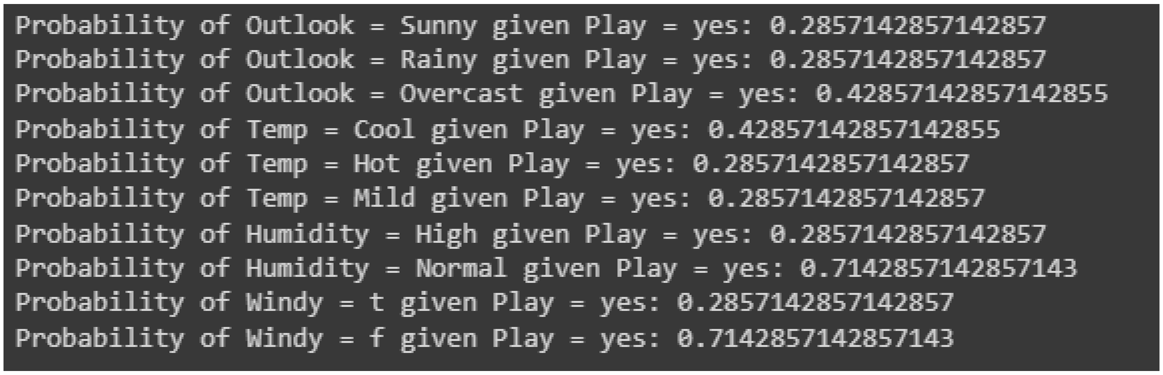


```
prior probability for playing :  0.5833333333333334
prior probability for not playing :  0.4166666666666667
```

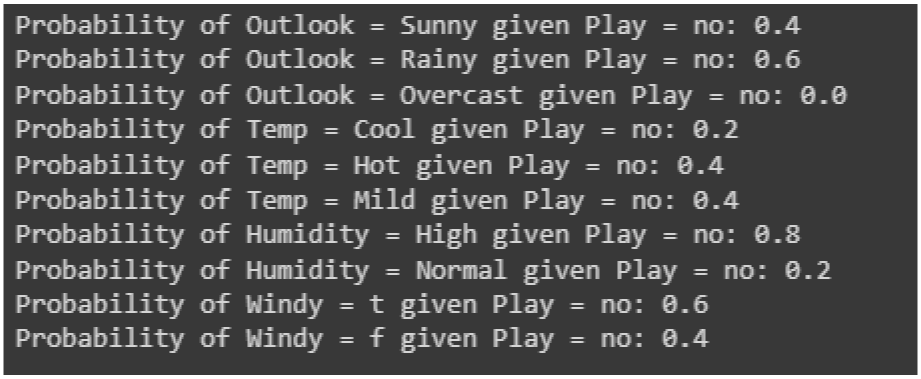
TASK-2:

Here we are required to calculate the likelihood probabilities for different features.
For example:
P(Outlook = sunny | Play = yes) =
(occurrence of outlook = sunny when play = yes) / (occurrences of Play = yes)

Similarly, we calculate probability for different features :



```
Probability of Outlook = Sunny given Play = yes: 0.2857142857142857
Probability of Outlook = Rainy given Play = yes: 0.2857142857142857
Probability of Outlook = Overcast given Play = yes: 0.42857142857142855
Probability of Temp = Cool given Play = yes: 0.42857142857142855
Probability of Temp = Hot given Play = yes: 0.2857142857142857
Probability of Temp = Mild given Play = yes: 0.2857142857142857
Probability of Humidity = High given Play = yes: 0.2857142857142857
Probability of Humidity = Normal given Play = yes: 0.7142857142857143
Probability of Windy = t given Play = yes: 0.2857142857142857
Probability of Windy = f given Play = yes: 0.7142857142857143
```



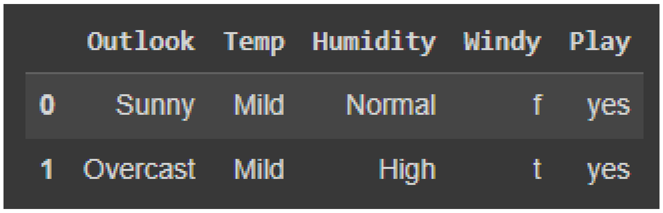
```
Probability of Outlook = Sunny given Play = no: 0.4
Probability of Outlook = Rainy given Play = no: 0.6
Probability of Outlook = Overcast given Play = no: 0.0
Probability of Temp = Cool given Play = no: 0.2
Probability of Temp = Hot given Play = no: 0.4
Probability of Temp = Mild given Play = no: 0.4
Probability of Humidity = High given Play = no: 0.8
Probability of Humidity = Normal given Play = no: 0.2
Probability of Windy = t given Play = no: 0.6
Probability of Windy = f given Play = no: 0.4
```

TASK-3:

Now we need to make predictions on the testing set on the basis of the likelihood probabilities that we calculated in the previous step.

We make the prediction by calculating posterior probability for each sample for both the target variables i.e. playing = yes, playing = no.

Let us take an example for this, the below is our test sample :



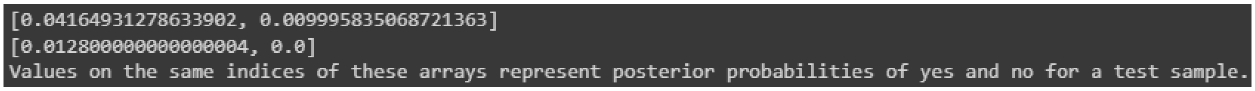
	Outlook	Temp	Humidity	Windy	Play
0	Sunny	Mild	Normal	f	yes
1	Overcast	Mild	High	t	yes

To predict whether a person would play or not we find the posterior probabilities for each of the target variables i.e. play = yes and play = no.

Posterior probability for test sample 1 (Play = yes) =
P(yes)*P(outlook = sunny)*P(Temp = mild)*P(Humidity = normal)*P(Windy = f)

Posterior probability for test sample 1 (Play = no) =
P(no)*P(outlook = sunny)*P(Temp = mild)*P(Humidity = normal)*P(Windy = f)

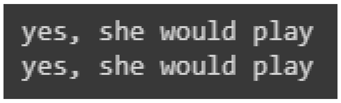
Based on these two probabilities we see the greater one and thus conclude it as the target variable.



```
[0.04164931278633902, 0.009995835068721363]
[0.012800000000000004, 0.0]
Values on the same indices of these arrays represent posterior probabilities of yes and no for a test sample.
```

TASK-4:

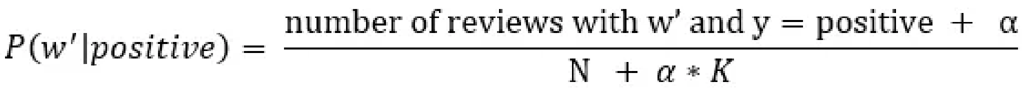
In this task on the basis of the calculated posterior probabilities we make the predictions for the target variable.
Below are the results for the test samples.



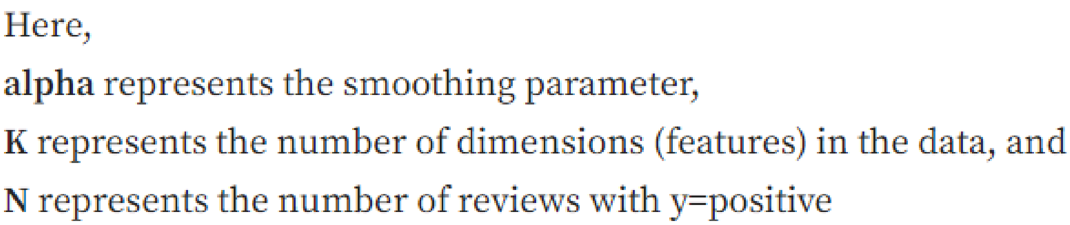
```
yes, she would play
yes, she would play
```

TASK-5:

In this task we need to perform laplace smoothing. This is used to avoid the condition of zero probability.
In our smoothing process we change the formula for likelihood probabilities.



$$P(w'|positive) = \frac{\text{number of reviews with w' and y = positive} + \alpha}{\mathrm{N} + \alpha * K}$$



Here,

alpha represents the smoothing parameter,

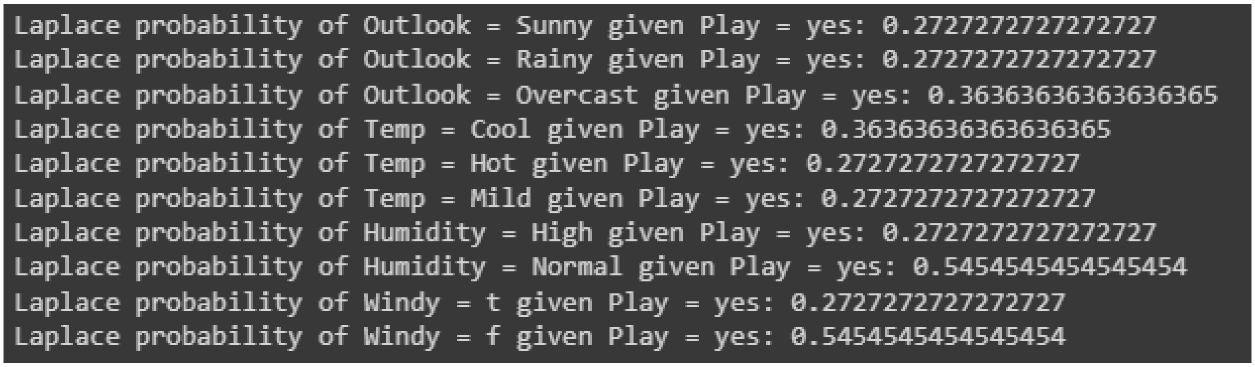
K represents the number of dimensions (features) in the data, and

N represents the number of reviews with y=positive

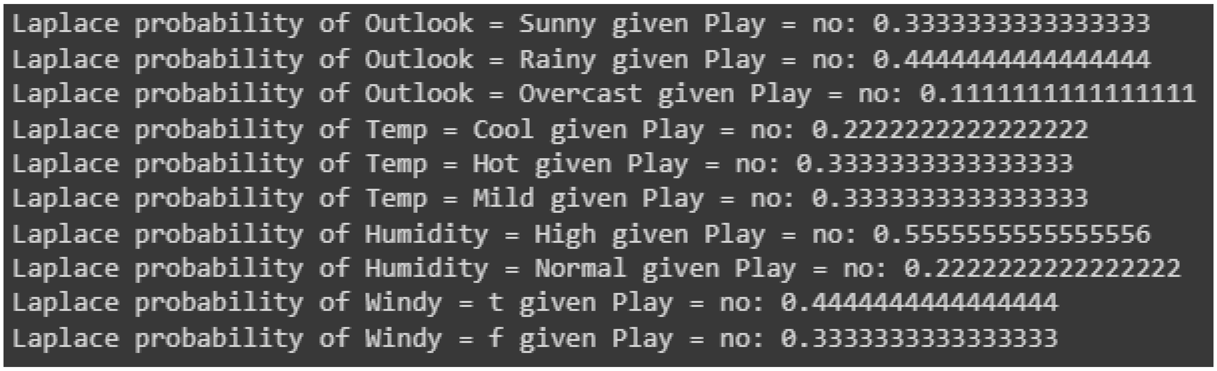
Using this formula likelihood probabilities are recalculated and the further process of calculating posterior probability remains the same and thus similarly we make predictions of the target variable.

But since the probabilities have changed so thus the final result might vary too.

Below are the laplace probabilities calculated:

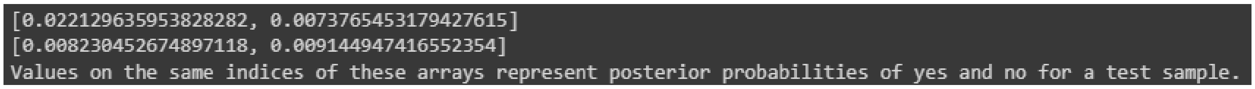


```
Laplace probability of Outlook = Sunny given Play = yes: 0.2727272727272727
Laplace probability of Outlook = Rainy given Play = yes: 0.2727272727272727
Laplace probability of Outlook = Overcast given Play = yes: 0.36363636363636365
Laplace probability of Temp = Cool given Play = yes: 0.36363636363636365
Laplace probability of Temp = Hot given Play = yes: 0.2727272727272727
Laplace probability of Temp = Mild given Play = yes: 0.2727272727272727
Laplace probability of Humidity = High given Play = yes: 0.2727272727272727
Laplace probability of Humidity = Normal given Play = yes: 0.5454545454545454
Laplace probability of Windy = t given Play = yes: 0.2727272727272727
Laplace probability of Windy = f given Play = yes: 0.5454545454545454
```



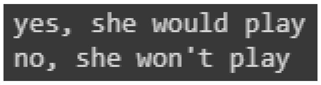
```
Laplace probability of Outlook = Sunny given Play = no: 0.3333333333333333
Laplace probability of Outlook = Rainy given Play = no: 0.4444444444444444
Laplace probability of Outlook = Overcast given Play = no: 0.1111111111111111
Laplace probability of Temp = Cool given Play = no: 0.2222222222222222
Laplace probability of Temp = Hot given Play = no: 0.3333333333333333
Laplace probability of Temp = Mild given Play = no: 0.3333333333333333
Laplace probability of Humidity = High given Play = no: 0.5555555555555556
Laplace probability of Humidity = Normal given Play = no: 0.2222222222222222
Laplace probability of Windy = t given Play = no: 0.4444444444444444
Laplace probability of Windy = f given Play = no: 0.3333333333333333
```

Posterior probabilities:



```
[0.022129635953828282, 0.0073765453179427615]
[0.008230452674897118, 0.009144947416552354]
Values on the same indices of these arrays represent posterior probabilities of yes and no for a test sample.
```

Conclusions:



```
yes, she would play
no, she won't play
```

Here, we observe that there has been a change in the concluded variable and this is because of the laplace smoothing.