Assignment-2

Shivanshu Shekhar

April 1, 2023

1 Introduction

The Bayes classifier is a classification algorithm that is based on Bayes' theorem. It is a probabilistic approach that makes predictions about the class of an unknown instance by computing the probabilities of all possible classes given the instance's features.

The Bayes classifier assumes that each class is associated with a probability distribution that describes the probability of observing a particular feature value given the class. These probability distributions are often modeled using Gaussian distributions or other parametric models.

This assignment will look at the Gaussian class distribution assumption-based Bayes classification.

We will also at the Naive-Bayes classifier, which assumes that the features for a datapoint are conditionally independent, which basically means that the correlation between the features is zero.

For both of these, the end task is to get the posterior distribution.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \tag{1}$$

2 Questions-1,2,3

We import the data into **numpy** arrays as usual, we also split the data into train and test by using **sample** function provided by **pandas**.

We split the data into 80% train and 20% test.

We write a utility function that does all this for us:

```
def test_train_split(data_path, random_state=42):
    X = pd.read_csv(data_path)


    X_train = X.sample(frac=0.8, random_state=random_state)
    X_test = X.drop(X_train.index)
```

```python
return X_train.to_numpy(), X_test.to_numpy(), X.to_numpy()
```

3 Question-4

In this question, we estimate the class conditionals from the data, for which we use **maximum likelihood** estimates.

We assume the class conditionals to be gaussian, so we can write.

The maximum likelihood estimates (MLE) for the parameters of a multivariate Gaussian distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$ can be obtained by maximizing the log-likelihood function:

$$\mathcal{L}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) = -\frac{n}{2}\ln(2\pi) - \frac{1}{2}\ln|\boldsymbol{\Sigma}| - \frac{1}{2}\sum_{i=1}^{n}(\mathbf{x}_i - \boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{x}_i - \boldsymbol{\mu}) \tag{2}$$

where $\mathbf{x}_1, \dots, \mathbf{x}_n$ are the observed data points.

The MLE for the mean vector is simply the sample mean:

$$\hat{\boldsymbol{\mu}} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i \tag{3}$$

The MLE for the covariance matrix can be expressed as:

$$\hat{\boldsymbol{\Sigma}} = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{x}_i - \hat{\boldsymbol{\mu}})(\mathbf{x}_i - \hat{\boldsymbol{\mu}})^T \tag{4}$$

The estimated mean vector and covariance matrix can be substituted back into the multivariate Gaussian density function to obtain the estimated probability density for a given point $\mathbf{x}$:

$$f_{\hat{\boldsymbol{\mu}},\hat{\boldsymbol{\Sigma}}}(\mathbf{x}) = \frac{1}{(2\pi)^{d/2}|\hat{\boldsymbol{\Sigma}}|^{1/2}}\exp\left(-\frac{1}{2}(\mathbf{x} - \hat{\boldsymbol{\mu}})^T\hat{\boldsymbol{\Sigma}}^{-1}(\mathbf{x} - \hat{\boldsymbol{\mu}})\right) \tag{5}$$

where d is the dimension of the data points. ‘

3.1 Question-4a

In Naive Bayes the covariance matrix will be a diagonal matrix.

After fitting the data we get the below plots for dataset-1 and dataset-2:

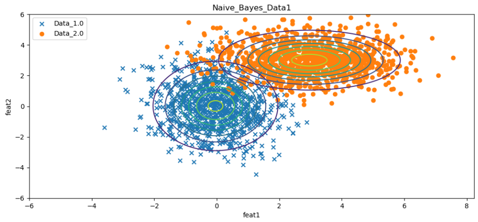


Figure 1: Dataset-1

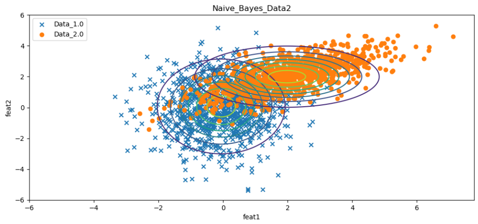


Figure 2: Dataset-2

3.2 Question-4b

In Bayes the covariance matrix will not necessarily be a diagonal matrix as we do account for correlation among the features.

After fitting the data we get the below plots for dataset-1 and dataset-2:

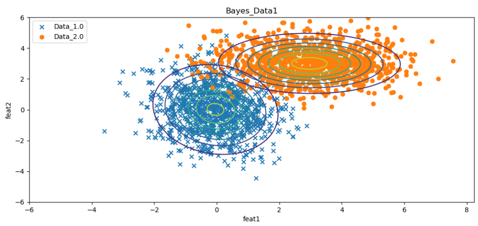


Figure 3: Dataset-1

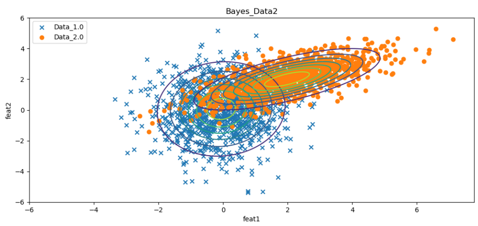


Figure 4: Dataset-2

4 Question-5

We next plot the decision boundary, the decision boundary is the line of equal probability for both the classes i.e. the line for which $P(Y = 1|X) = 0.5$.

We see that the line for both Naive Bayes and Bayes classifiers is hyperbola(i.e. quadratic) this is because we didn't assume the same covariance matrix for the two class conditionals. If that were the case then we would have ended up with a linear decision boundary(source: Bishop).

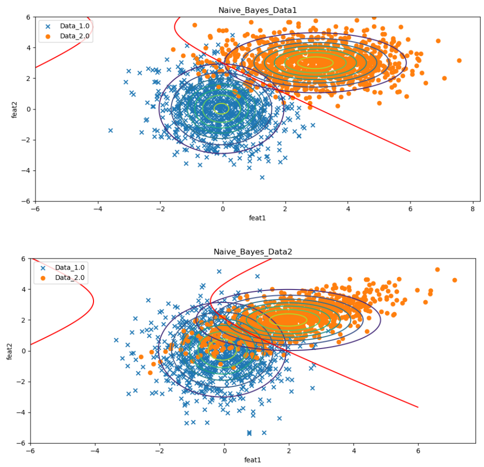


Figure 5: Complete Plots for Naive Bayes

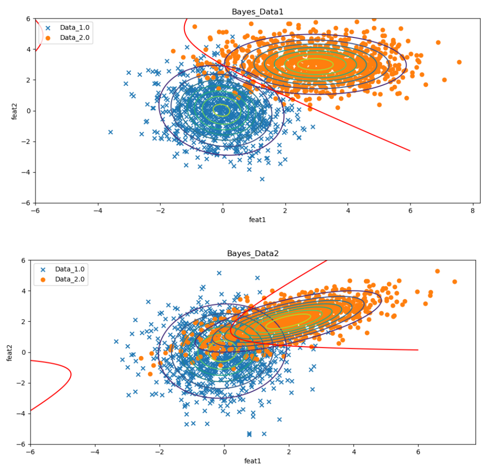


Figure 6: Complete Plots for Bayes

As we see from the above plots in dataset-1 the distribution of the class-2 datapoints is best fitted by a skewed gaussian which Naive Bayes is just incapable of doing hence Bayes should perform significantly better than Naive Bayes in the asymptotic limit of infinite data.

In the case of dataset-2, we see that both the class distribution has very less correlation between the features hence we can conclude that Bayes and Naive Bayes will perform identically in the asymptotic limit of infinite data.

5 Question-6

A confusion matrix is a table used to evaluate the performance of a machine learning model by comparing its predictions with the actual values. It is a commonly used evaluation metric in classification problems.

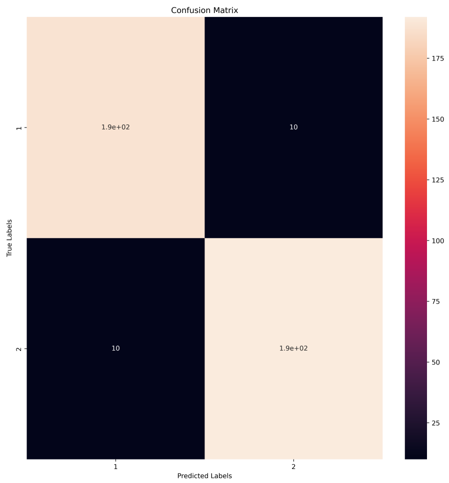
A confusion matrix displays the number of true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN) generated by a classification model.

- True Positive (TP): The model correctly predicted a positive class.
- False Positive (FP): The model incorrectly predicted a positive class.
- True Negative (TN): The model correctly predicted a negative class.
- False Negative (FN): The model incorrectly predicted a negative class.

$$\begin{bmatrix} TN & FP \\ FN & TP \end{bmatrix}$$

The matrix shows that there are four possible outcomes when comparing the actual and predicted labels.

We plot the confusion matrix for both the classifier on both the dataset.



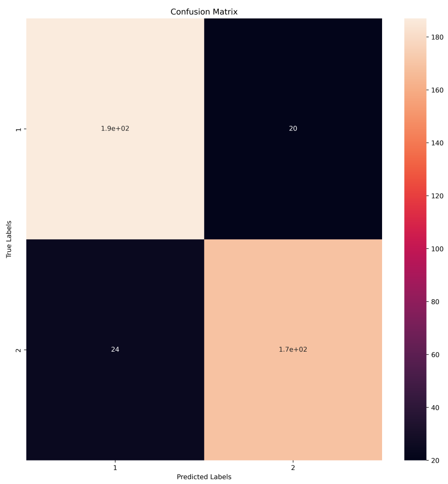
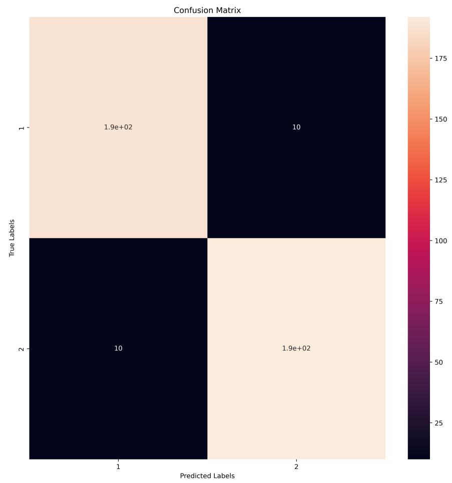


Figure 7: Confusion matrix for Naive Bayes



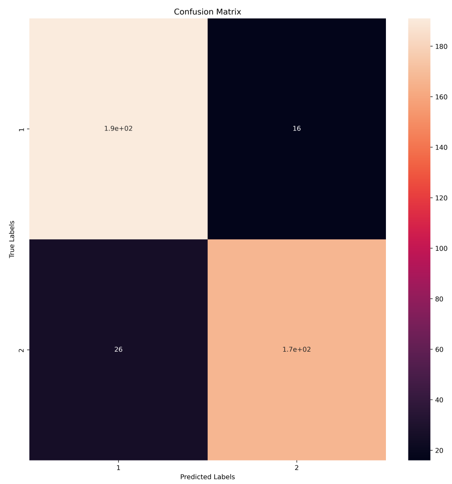


Figure 8: Complete Matrix for Bayes

6 Conclusion

We plot the confusion matrix on the testsets for both the classifiers, we observe that for the second dataset, the Bayes classifier is only marginally better than the Naive Bayes classifier which means that we could generally use Naive Bayes even for correlated data and still get somewhat good results, and in the first dataset we see that both are performing equally which was sort off expected as the dataset features have low correlation, Sometimes Naive Bayes might perform better than Bayes, this could be attributed to the fact that the test set in not accurately represent the data, we can confirm this by finding accuracy on the complete dataset from which we see that Bayes outperforms Naive Bayes.

In this assignment we saw that if the underlying assumption on the distribution of the class conditional is true, then the Bayes algorithm can perform well, we also saw that the Naive Bayes assumption although seems like a very strong assumption, in practice gives only slightly worse accuracies than the Bayes classifier

Although it should be noted that as we are using **MLE** for parameter estimation and we are estimating the mean and variance both from the given data, we will always underestimate the variance of the data, the error between the actual variance and the estimated variance decreases with the increase in a number of samples.