A close up of a sign

Description automatically generated

Assignment - 01

**Machine Learning & Pattern Recognition**

COMP 8740 Fall 2019

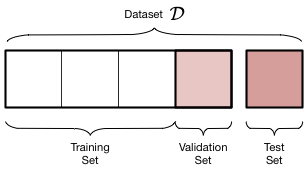
**Project Supervisor**: Dr. Luis Rueda

**Authors**: Mayank Semwal

**1).** All the datasets stored inside the folder named “SampleDataset/...”.

Architecture followed for Model classification:

**Training & Testing**: Use of Sklearn’s library “train\_test\_split” for creating a train and test dataset and divided the dataset into ratio of 70% for training and 30% for test.



**Validation test**: Sklearn’s 10-fold cross validation score to validate the training results.

* Pandas interpolate function is used in this assignment to fill null values in the dataset.

**2).** Classifiers and five performance measures in a table after explanation:

a). **Linear Discriminant Analysis (LDA)**: Using scikit-learn which contains library known as discriminant\_analysis and contains “LinearDiscriminantAnalysis” to compute LDA.

* For fitting the model, “fit\_transform” is used which first calls fit function and then transforms the same data.
* For classification, LDA is mostly used for dimensionality reduction but the library also contains predict method. LDA predict method is used in this assignment.

b). **Quadratic Discriminant Analysis (QDA)**:

* QDA and LDA share the same library of scikit-learn mentioned above.
* Only difference is QDA does not have fit\_transform method and it automatically transforms the data when fitting.

c). **Naïve Bayes**: Scikit-learn or Sklearn has a library “naive\_bayes” which contains distribution and kernels.

* For distribution I have considered Gaussian Naïve Bayes (GNB) because value of features in the datasets are continuous and follow a normal distribution of data. GNB can take values from negative infinity to positive infinity. Gaussian (Normal distribution) is also easy to work with because we just need to estimate the mean and the standard deviation from data.
* The main reason for not choosing Multinomial Naïve Bayes (MNB) is that MNB cannot work with negative values which our dataset has in abundance. MNB is mostly used for frequency occurrence and count of a word in a text classification problem which cannot be negative that’s whys MNB will not work in the datasets used in this assignment.
* Bernoulli Naïve Bayes (BNB) works well when the values of features are of binary nature i.e. 1 (true) or 0 (false), negative or positive. Because of such nature BNB will binarize the input feature values, and decision rule of BNB will create linear boundary which is not suited for datasets in this assignment.

d). **Support Vector Machine (SVM)**: Sklearn contains a library known as **svm** which contains SVC (support vector classification) method and the implementation is based on libsvm.

* RBF kernel of SVM is chosen because the dataset in the assignment is a nonlinear problems and RBF creates nonlinear combinations of features to uplift the samples onto a higher dimensional feature space where then uses a linear decision boundary to separate your classes. Whereas Linear, Sigmoid and Poly kernels of SVM models are unable to fit the data into specific curve due to the nature of data’s shape.

**Performance measures for each Dataset of a classifier:** Except the asked metrics, I have used five others.

* R2 score: To determine how close the data are to the fitted regression line. It is also known as the coefficient of determination.
* F1 Score: Weighted average of Precision and Sensitivity.
* Time Taken: Total time taken by classifier to fit and predict.
* Misclassified samples: To determine number of samples classifier not able to classify.

**1). CIRCLE:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CLASSFIERS | LDA | QDA | SVM(RBF) | Naïve Bayes(Gaussian) |
| Accuracy | 33 % | 99 % | 100 % | 99% |
| Sensitivity (Recall) | 56 % | 100 % | 100 % | 100% |
| Specificity (True Negative Rate) | 11 % | 98 % | 100% | 98% |
| Precision (PPV) | 37 % | 98 % | 100% | 98% |
| Negative Predictive Value(NPV) | 21 % | 100 % | 100% | 100% |
| F1 Score | 45 % | 99 % | 100% | 99% |
| Geometric mean | 20 % | 98 % | 100% | 98% |
| R2 Score | -169.76 | 95.99 | 100% | 95.99% |
| Misclassified Samples | 202 | 3 | 0 | 3 |
| Time Taken | 4.18sec | 0.61sec | 1.91sec | 2.45sec |

**2). SPIRAL:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CLASSFIERS | LDA | QDA | SVM(RBF) | Naïve Bayes(Gaussian) |
| Accuracy | 72% | 72% | 100% | 70% |
| Sensitivity (Recall) | 70% | 70% | 100% | 66% |
| Specificity (True Negative Rate) | 73% | 73% | 99% | 73% |
| Precision (PPV) | 71% | 71% | 99% | 70% |
| Negative Predictive Value(NPV) | 72% | 72% | 100% | 70% |
| F1 Score | 71% | 71% | 100% | 68% |
| Geometric mean | 72% | 72% | 99% | 71% |
| R2 Score | -13.41 | -13.41 | 98.67% | -21.42% |
| Misclassified Samples | 85 | 85 | 1 | 91 |
| Time Taken | 2.56sec | 2.12sec | 3.85sec | 1.57sec |

**3). MOON:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CLASSFIERS | LDA | QDA | SVM(RBF) | Naïve Bayes(Gaussian) |
| Accuracy | 87% | 85% | 100% | 87% |
| Sensitivity (Recall) | 88% | 87% | 100% | 88% |
| Specificity (True Negative Rate) | 86% | 82% | 99% | 86% |
| Precision(PPV) | 86% | 83% | 99% | 86% |
| Negative Predictive Value(NPV) | 89% | 87% | 100% | 89% |
| F1 Score | 87% | 85% | 100% | 87% |
| Geometric mean | 86 | 82% | 99% | 86% |
| R2 Score | 49.31% | 38.64% | 98.67% | 49.31% |
| Misclassified Samples | 38 | 46 | 1 | 38 |
| Time Taken | 3.03sec | 0.73sec | 2.49sec | 1.50sec |

**4). HALF KERNEL:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CLASSFIERS | LDA | QDA | SVM(RBF) | Naïve Bayes(Gaussian) |
| Accuracy | 71% | 94% | 100% | 96% |
| Sensitivity (Recall) | 79% | 99% | 100% | 100% |
| Specificity (True Negative Rate) | 63% | 89% | 100% | 92% |
| Precision(PPV) | 67% | 90% | 100% | 92% |
| Negative Predictive Value(NPV) | 77% | 99% | 100% | 100% |
| F1 Score | 72% | 94% | 100% | 96% |
| Geometric mean | 65% | 90% | 100% | 92% |
| R2 Score | -16.19 | 75.96% | 100% | 83.97% |
| Misclassified Samples | 58 | 12 | 0 | 8 |
| Time Taken | 2.76sec | 1.79sec | 3.47sec | 2.82sec |

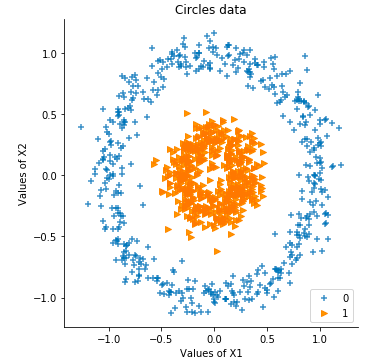
**5). Two Gaussian33:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CLASSFIERS | LDA | QDA | SVM(RBF) | Naïve Bayes(Gaussian) |
| Accuracy | 99% | 99% | 99% | 99% |
| Sensitivity (Recall) | 100% | 99% | 99% | 99% |
| Specificity (True Negative Rate) | 97% | 99% | 99% | 98% |
| Precision(PPV) | 95% | 99% | 99% | 98% |
| Negative Predictive Value(NPV) | 100% | 99% | 99% | 99% |
| F1 Score | 99% | 99% | 99% | 99% |
| Geometric mean | 97% | 99% | 99% | 98% |
| R2 Score | 94.66% | 95.99% | 95.99% | 94.66% |
| Misclassified Samples | 4 | 3 | 3 | 4 |
| Time Taken | 2.55sec | 1.84sec | 2.83sec | 1.91sec |

**6). Two Gaussian 42:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CLASSFIERS | LDA | QDA | SVM(RBF) | Naïve Bayes(Gaussian) |
| Accuracy | 89% | 94% | 93% | 88% |
| Sensitivity (Recall) | 95% | 97% | 98% | 93% |
| Specificity (True Negative Rate) | 83% | 91% | 89% | 84% |
| Precision(PPV) | 84% | 91% | 89% | 85% |
| Negative Predictive Value(NPV) | 94% | 97% | 98% | 93% |
| F1 Score | 89% | 94% | 94% | 89% |
| Geometric mean | 84% | 91% | 89% | 84% |
| R2 Score | 54.64% | 75.99% | 73.32% | 53.31% |
| Misclassified Samples | 34 | 18 | 20 | 35 |
| Time Taken | 2.57sec | 2.03sec | 2.62sec | 1.53sec |

**3).** For 2D plots, using seaborn library which contains method “sns.implot” to plot.

 A screenshot of a cell phone

Description automatically generated

*Fig 1: Circles Fig 2: Half Kernel*

A close up of a piece of paper

Description automatically generated A close up of a logo

Description automatically generated

*Fig 3: Moon Fig 4: Spiral*

A close up of a map

Description automatically generated A close up of a map

Description automatically generated

*Fig 5: Two Gaussian 33 Fig 6: Two Gaussian 42*

**4).** Performance (accuracy only) for each classifier and dataset.

1. **Linear Discriminant Analysis (LDA):** 
   * 1. Circle Dataset: “learning\_curve” method of Sklearn is used for image below.

|  |  |
| --- | --- |
| **Score** | **In %** |
| Training Accuracy | 35.5714 |
| Validation Accuracy | 34.2710 |
| Predicted Test Accuracy | 33.00 |

A close up of a map

Description automatically generated

Validation Score is to check for overfitting or underfitting. This can also be assured from learning curve above (10 K-fold cross validation over 70% of data i.e. 700 records). There is underfitting because LDA is unable to capture the trend of the data and does not fit the data well. And even after 5 epochs there is no sign of better fitness of dataset with LDA.

A picture containing map

Description automatically generated

**Conclusion:** LDA is bad for the circle dataset as the predicted accuracy is very poor and from the decision plot boundary can be seen the linear line plotted by LDA, which concludes that the dataset is not linearly separable. As mentioned in a table in part 2, total misclassified samples = 202 and negative R2 Score also proves LDA is not a good approach to classify this dataset.

* + 1. Spiral Dataset:

|  |  |
| --- | --- |
| **Score** | **In %** |
| Training Accuracy | 76.4286 |
| Validation Accuracy | 76.2920 |
| Predicted Test Accuracy | 72.0 |

**A close up of a map

Description automatically generated**

Though there is no sign of underfitting or overfitting, but due to the nature of dataset LDA did its best to classify the dataset and also learning rate gradually increased which is a good sign but not good enough to classify 28% of the records and negative R2 Score also proves LDA is not a good approach to classify this dataset.

**A close up of a map

Description automatically generated**

**Conclusion:** LDA is not a good option for the spiral dataset as from the decision plot boundary can be seen that the linear line plotted by LDA, which concludes that the dataset is not linearly separable. As mentioned in a table in part 2 total misclassified samples = 85 also proves LDA is not a good approach to classify this dataset.

* + 1. Moon Dataset:

|  |  |
| --- | --- |
| **Score** | **In %** |
| Training Accuracy | 88.8571 |
| Validation Accuracy | 88.4272 |
| Predicted Test Accuracy | 87.0 |

**A close up of a map

Description automatically generated**

There is no sign of underfitting or overfitting, but due to the nature of dataset LDA did its best to classify the dataset and also learning rate gradually increased which is a good sign and performed well for this dataset than above two datasets.

**A close up of a logo

Description automatically generated**

**Conclusion:** LDA is not the best option for the Moon dataset but performed better than QDA, Naïve Bayes only behind SVM. From the decision plot boundary can be seen that the linear line plotted by LDA, which concludes that the dataset is not linearly separable. As mentioned in a table in part 2 total misclassified samples = 38 also proves LDA is not a good approach to classify this dataset.

* + 1. Half Kernel:

|  |  |
| --- | --- |
| **Score** | **In %** |
| Training Accuracy | 66.5 |
| Validation Accuracy | 66.39 |
| Predicted Test Accuracy | 71.0 |

**A close up of a map

Description automatically generated** There is slight underfitting, due to the shape of data points, LDA is not able to understand the nature of data and hence not able to classify correctly, and also learning rate gradually decreased which is a bad sign and unable to classify 29% of records and negative R2 score proves LDA is not able to classify data.

**A close up of a piece of paper

Description automatically generated**

**Conclusion:** LDA is the worst for this dataset. From the decision plot boundary can be seen that the linear line plotted by LDA, which concludes that the dataset is not linearly separable. As mentioned in a table in part 2 total misclassified samples = 58 and R2 score= -16.18 also proves LDA is not a good approach to classify this dataset.

* + 1. Two Gaussian33:

|  |  |
| --- | --- |
| **Score** | **In %** |
| Training Accuracy | 98.2857 |
| Validation Accuracy | 98.2916 |
| Predicted Test Accuracy | 95.0 |

**A close up of a map

Description automatically generated**

LDA performed very well for two gaussian 33 dataset, it is visible from the learning curve as well that model is able to adapt to the shape of data points after 3 epochs only and R2 score of 94.66 to prove LDA fitted efficiently w.r.t regression line. LDA is just marginally behind other classifiers.

**A close up of a map

Description automatically generated**

**Conclusion:** LDA is the good for this dataset. From the decision plot boundary can be seen that the linear line plotted by LDA, which concludes that the dataset is linearly separable. As mentioned in a table in part 2 total misclassified samples = 4 also proves LDA is a good approach to classify this dataset. Considering the simplicity of LDA its better for this dataset.

* + 1. Two Gaussian42:

|  |  |
| --- | --- |
| **Score** | **In %** |
| Training Accuracy | 90.2857 |
| Validation Accuracy | 89.9901 |
| Predicted Test Accuracy | 89.0 |

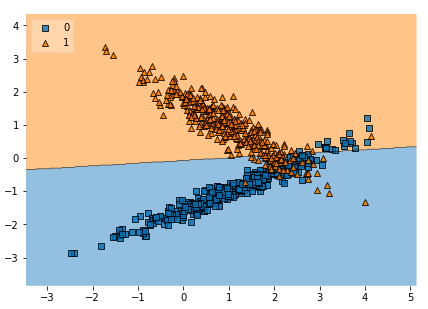
**A screenshot of a cell phone

Description automatically generated**

LDA performed average for two gaussian 42 dataset but is not suitable because of closeness

of data points, visible from the learning curve as well that model is able to adapt to the shape of data points after 5 epochs only but after that gap between training and validation score is

increasing.

****

**Conclusion:** LDA is the not good for this dataset. From the decision plot boundary which concludes that the dataset is linearly separable but due to closeness of data point not able to efficiently separate linearly. Total misclassified samples = 34 also proves LDA is a not good approach to classify this dataset compared to others.

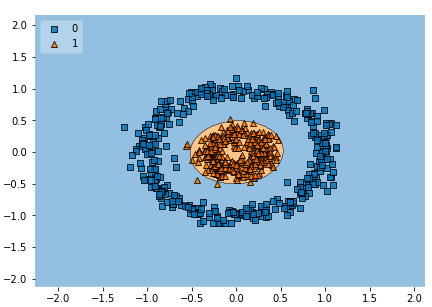
1. **Quadratic Discriminant Analysis (QDA):** 
   * 1. Circle Dataset:

|  |  |
| --- | --- |
| **Score** | **In %** |
| Training Accuracy | 99.1429 |
| Validation Accuracy | 99.0039 |
| Predicted Test Accuracy | 99.00 |

A close up of text on a white background

Description automatically generated

There is overfitting because training loss is more than validation loss. Though training score is still increasing even after 7 epochs but there is still slight overfitting.



**Conclusion:** QDA is good for the circle dataset as the predicted accuracy is excellent and

from the decision plot boundary can be seen the quadratic line plotted by QDA. Total

misclassified samples = 3 and R2 Score =96 also proves QDA is good to classify this dataset

and only behind SVM’s RBF.

* + 1. Spiral Dataset:

|  |  |
| --- | --- |
| **Score** | **In %** |
| Training Accuracy | 76.5714 |
| Validation Accuracy | 76.2920 |
| Predicted Test Accuracy | 72.0 |

**A close up of a map

Description automatically generated**

Though there is no sign of underfitting or overfitting, but due to the nature of dataset QDA did its best to classify the dataset and also learning rate gradually increased which is a good sign but not good enough to classify 28% of the records and negative R2 Score also proves QDA is not a good approach to classify this dataset.

**A close up of a piece of paper

Description automatically generated**

**Conclusion:** QDA is not the best option for the spiral dataset as from the decision plot boundary can be seen that the linear line plotted by QDA, which concludes that the dataset is not linearly separable does not fit into quadratic equation as well. Total misclassified samples = 85 also proves QDA is not a good approach to classify this dataset.

* + 1. Moon Dataset:

|  |  |
| --- | --- |
| **Score** | **In %** |
| Training Accuracy | 86.5714 |
| Validation Accuracy | 86.5697 |
| Predicted Test Accuracy | 85.0 |

**A close up of a map

Description automatically generated**

There is no sign of underfitting or overfitting, but due to the nature of this dataset, QDA performed worse than LDA for this dataset, Accuracy is less than LDA and also learning curve is decreasing in every epoch signifying overfitting as training loss is more.

A close up of a logo

Description automatically generated

**Conclusion:** QDA is a bad option for the Moon dataset. From the decision plot boundary can be seen that the linear line plotted by QDA, which concludes that the dataset is not linearly separable and does not fit data points into quadratic equation well. Total misclassified samples = 46 also proves QDA is not a good approach to classify this dataset.

* + 1. Half Kernel:

|  |  |
| --- | --- |
| **Score** | **In %** |
| Training Accuracy | 95.6250 |
| Validation Accuracy | 95.1212 |
| Predicted Test Accuracy | 94.0 |

A screenshot of text

Description automatically generated

QDA is a good classifier for this dataset which can be observed from learning curve. Accuracy is good but the issue in QDA is misclassified samples which are more compared to SVM and Naïve Bayes.

A close up of a map

Description automatically generated

**Conclusion:** QDA is good for this dataset. Misclassified samples = 12 and R2 score= 75 signifies

QDA is a good approach to classify this dataset being simple in execution than other classifiers

Doing slightly better than QDA.

* + 1. Two Gaussian33:

|  |  |
| --- | --- |
| **Score** | **In %** |
| Training Accuracy | 99.1429 |
| Validation Accuracy | 99.2836 |
| Predicted Test Accuracy | 99.0 |

**A screenshot of a cell phone

Description automatically generated**

QDA performed best for two gaussian 33 dataset, it is visible from the learning curve as well that model is able to adapt to the shape of data points after 3 epochs only and R2 score= 96

to prove QDA fitted efficiently w.r.t regression line. QDA outperformed SVM in time taken to

classify else accuracy is and misclassified samples are similar for both.

**A close up of a map

Description automatically generated**

**Conclusion:** QDA is the best for this dataset. From the decision plot boundary can be seen that the quadratic line plotted by QDA, which concludes that the dataset fits well into quadratic equation. As mentioned in a table in part 2 total misclassified samples = 3. Considering the

simplicity of QDA its better for this dataset than SVM. Choice is between GNB and QDA.

* + 1. Two Gaussian42:

|  |  |
| --- | --- |
| **Score** | **In %** |
| Training Accuracy | 95.4286 |
| Validation Accuracy | 95.1298 |
| Predicted Test Accuracy | 94.0 |

**A close up of a map

Description automatically generated**

QDA performed best for two gaussian 42 dataset, it is visible from the learning curve as well

that model is able to adapt to the shape of data points and still learning to classify all records and R2 score of 75 to prove QDA fitted efficiently w.r.t regression line. QDA outperformed

SVM in accuracy, time taken, R2 score and misclassified samples.

**A close up of a map

Description automatically generated**

**Conclusion:** QDA is the best for this dataset. From the decision plot boundary can be seen that the quadratic line plotted by QDA that fits data well into quadratic equation. Misclassified

samples = 18 which is less as compared to all other classifiers.

1. **Support Vector Machine (SVM):** 
   * 1. Circle Dataset:

|  |  |
| --- | --- |
| **Score** | **In %** |
| Training Accuracy | 100 |
| Validation Accuracy | 100 |
| Predicted Test Accuracy | 100 |

A close up of a map

Description automatically generated

SVM performed excellent for this dataset with an accuracy of 100%, Its visible from the learning curve as well that after 3 epochs model fit well enough. Value of Gamma= 0.6 and Cost= 0.5 even with these small values SVM is able to fit all the data points perfectly.

A close up of a logo

Description automatically generated

**Conclusion:** SVM is best for the circle dataset as the predicted accuracy is excellent and from the decision plot boundary can be seen that non-linear line plotted by SVM incorporated all

support vectors. Misclassified samples = 0 and R2 Score =100 also proves SVM is best to classify this dataset.

* + 1. Spiral Dataset:

|  |  |
| --- | --- |
| **Score** | **In %** |
| Training Accuracy | 100 |
| Validation Accuracy | 99.2815 |
| Predicted Test Accuracy | 100 |

**A close up of a map

Description automatically generated**

SVM performed excellent for this dataset with an accuracy of 100%, Its visible from the learning curve as well that after 3 epochs model fit well enough. Value of Gamma= 1 and Cost= 0.5 even with these small values SVM is able to fit all the data points perfectly.

**A close up of a map

Description automatically generated**

**Conclusion:** SVM is best and outperforms all other classifiers by large margin in this

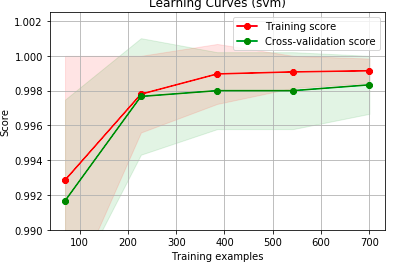
dataset. Predicted accuracy is excellent and from the decision plot boundary can be

seen that non-linear line plotted by SVM incorporated all support vectors. Misclassified

samples = 1 and R2 Score =100 also proves SVM is best to classify this dataset.

* + 1. Moon Dataset:

|  |  |
| --- | --- |
| **Score** | **In %** |
| Training Accuracy | 100 |
| Validation Accuracy | 100 |
| Predicted Test Accuracy | 99.67 |

****

SVM performed excellent for this dataset with an accuracy of 100%, Its visible from the learning curve as well that after 3 epochs model fit well enough. Value of Gamma= 2 and Cost= 1 even though values had to be increased to acquire better accuracy then SVM is able to fit all the data points perfectly.

A close up of a map

Description automatically generated

**Conclusion:** SVM is best and outperforms all other classifiers by large margin in this

dataset. Predicted accuracy is excellent and from the decision plot boundary can be

seen that non-linear line plotted by SVM incorporated all support vectors. Misclassified

samples = 1 and R2 Score =98.67 also proves SVM is best to classify this dataset.

* + 1. Half Kernel:

|  |  |
| --- | --- |
| **Score** | **In %** |
| Training Accuracy | 100 |
| Validation Accuracy | 100 |
| Predicted Test Accuracy | 100 |

A close up of text on a white background

Description automatically generated

SVM performed good for this dataset with an accuracy of 100%, Its visible from the learning curve as well that after 2 epochs model fit well enough. Value of Gamma= 0.01 and Cost= 0.5 even with these small values SVM is able to fit all the data points perfectly.

A close up of a map

Description automatically generated

**Conclusion:** SVM is good for this dataset. Predicted accuracy is excellent and from the decision plot boundary can be seen that non-linear line plotted by SVM incorporated all support vectors with maximum margin. Misclassified samples = 3 and R2 Score =95.99. Only issue is the time taken by SVM to classify.

* + 1. Two Gaussian33:

|  |  |
| --- | --- |
| **Score** | **In %** |
| Training Accuracy | 99.5714 |
| Validation Accuracy | 98.9958 |
| Predicted Test Accuracy | 99.0 |

**A close up of a map

Description automatically generated**

SVM performed good for this dataset with an accuracy of 99%, Its visible from the learning curve as well that after 2 epochs model fit well enough. Value of Gamma= 3 and Cost= 1 even though values had to be increased to acquire better accuracy then SVM is able to fit all the data points perfectly.

**A close up of a map

Description automatically generated**

**Conclusion:** SVM is similar to all the classifier for this dataset. Misclassified samples = 3 and

R2 Score =95.99. There is no point of using SVM in this dataset because QDA is fast as well.

* + 1. Two Gaussian42:

|  |  |
| --- | --- |
| **Score** | **In %** |
| Training Accuracy | 94.8571 |
| Validation Accuracy | 94.42 |
| Predicted Test Accuracy | 93.0 |

**A close up of a map

Description automatically generated**

SVM performed good for this dataset with an accuracy of 93%, Its visible from the learning curve as well that after 4 epochs model fit well enough. Value of Gamma= 1 and Cost= 1 is used to fit the data. But when compared with QDA it did not performed better than QDA.

**A close up of a map

Description automatically generated**

**Conclusion:** SVM is not good for this dataset, as SVM is complicated and has more misclassified samples = 20 and R2 Score =73.32. There is no point of using SVM in this dataset because QDA is fast, accurate and less misclassified samples as well. For this dataset.

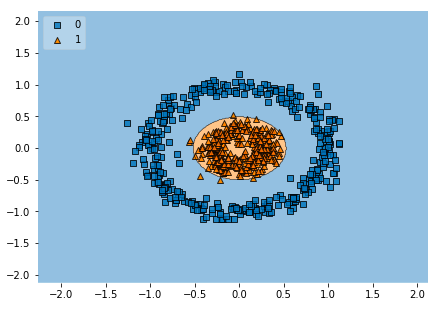
1. **Gaussian Naïve Bayes (GNB):** 
   * 1. Circle Dataset:

|  |  |
| --- | --- |
| **Score** | **In %** |
| Training Accuracy | 99.2857 |
| Validation Accuracy | 98.9998 |
| Predicted Test Accuracy | 99 |

A close up of a map

Description automatically generated

Accuracy is very high similar to QDA. From the learning curve its clear there is overfitting because training loss is more than validation loss. Though training score is still increasing even after 7 epochs so there is just slight overfitting.



**Conclusion:** GNB is good for the circle dataset as the predicted accuracy is excellent and

from the decision plot boundary can be seen that non-linear line plotted by GNB incorporated almost all data points. Misclassified samples = 3 and R2 Score = 95.99 also proves GNB is good and only falling behind SVM by slightest.

.

* + 1. Spiral Dataset:

|  |  |
| --- | --- |
| **Score** | **In %** |
| Training Accuracy | 74.8571 |
| Validation Accuracy | 74.8530 |
| Predicted Test Accuracy | 70 |

**A close up of a map

Description automatically generated**

Though there is no sign of underfitting or overfitting, but due to the nature of dataset GNB did its best to classify the dataset and also learning rate gradually increased which is a good sign but not good enough to classify 30% of the records and negative R2 Score also proves GNB is not a good approach to classify this dataset.

**A close up of a map

Description automatically generated**

**Conclusion** GNB is bad option for the spiral dataset because of low accuracy and from the decision plot boundary can be seen that the linear line is plotted by GNB, but the dataset is not linearly separable. Misclassified samples = 91 also proves GNB is not a good approach to classify this dataset.

* + 1. Moon Dataset:

|  |  |
| --- | --- |
| **Score** | **In %** |
| Training Accuracy | 88.8571 |
| Validation Accuracy | 88.5700 |
| Predicted Test Accuracy | 87.0 |

**A screenshot of a cell phone

Description automatically generated**

Accuracy of GNB is higher than QDA but from the learning curve its clear there is overfitting because training loss is more than validation loss. And misclassification number is higher as well. GNB tries to fit non-linear data into linear line. So, GNB is no the good option.

**A close up of a logo

Description automatically generated**

**Conclusion:** GNB is bad option for the spiral dataset because of average accuracy and from the decision plot boundary can be seen that the linear line is plotted by GNB, but the dataset is not linearly separable. Misclassified samples = 38 also proves GNB is not a good approach to classify this dataset.

* + 1. Half Kernel:

|  |  |
| --- | --- |
| **Score** | **In %** |
| Training Accuracy | 96.7500 |
| Validation Accuracy | 96.2479 |
| Predicted Test Accuracy | 96 |

A screenshot of a cell phone

Description automatically generated

GNB performs well for this dataset giving high accuracy and best learning curve because it is learning in each epoch. GNB is good for this dataset. Predicted accuracy is excellent and from the decision plot boundary can be seen that non-linear line plotted by GNB incorporated most data points.

A close up of a map

Description automatically generated

**Conclusion:** Misclassified samples = 8 and R2 Score =83.97. Despite good accuracy SVM and QDA is still performing better than GNB. QDA classifies in half the time and less misclassifications.

So, GNB is not a good option by compared to others.

* + 1. Two Gaussian33:

|  |  |
| --- | --- |
| **Score** | **In %** |
| Training Accuracy | 99.1429 |
| Validation Accuracy | 99.2815 |
| Predicted Test Accuracy | 99.0 |

**A close up of text on a white background

Description automatically generated**

GNB performs well for this dataset giving high accuracy. But from learning curve it can be seen that after 2 epochs learning rate varied for validation score and training score is continuously falling. GNB is good for this dataset. Predicted accuracy is excellent and from the decision plot boundary can be seen that non-linear line plotted by GNB incorporated most data points.

**A close up of a map

Description automatically generated**

**Conclusion:** GNB performed better than SVM w.r.t to time taken and provides same accuracy, Misclassified samples = 4 and R2 Score =94.66. GNB and QDA created the same decision boundary when classifying and in almost same time. GNB is a good model for this dataset and choice is between QDA and GNB.

* + 1. Two Gaussian42:

|  |  |
| --- | --- |
| **Score** | **In %** |
| Training Accuracy | 91.8571 |
| Validation Accuracy | 91.9966 |
| Predicted Test Accuracy | 88.0 |

**A screenshot of a cell phone

Description automatically generated**

GNB performed average for this dataset with an accuracy of 88%, Its visible from the learning curve as well that after 2 epochs model is stagnant and not learning at all. R2 Score =53.32 is very low because of high density data and model is not able to fit data on regression line.

**A close up of a map

Description automatically generated**

**Conclusion:** GNB is not a good option for this dataset because of high misclassified samples=35. There is no point of using SVM in this dataset because QDA is fast, accurate and less misclassified samples as well. For this dataset. Even Bernoulli NB and LDA performed better than GNB.

**5).** For each classifier: which classifier is the best for that particular dataset, why?

1. **CIRCLE, SPIRAL, MOON** and **HALF-KERNEL:** For these datasets SVM’s RBF kernel is the best. The reason to select SVM is the non-linear behaviour of dataset and also negligible misclassified samples and less amount of time taken to classify the dataset with higher accuracy by SVM. Kernel trick avoids the explicit mapping which is required to get linear learning algorithms to learn a nonlinear or decision boundary.

The RBF kernel is defined as:

*RBF (x, x’) = exp [ −γ ||x – x’||****2****]*

where, x – x’ is squared Euclidian distance and “γ” is a parameter that sets the “spread” of the kernel, which is Gamma parameter and it gives flexibility to fit data efficiently around decision boundary. RBF also penalises misclassified samples using cost function.

K (x, x’) = {ψ(x), ψ(x’)}

Where ψ is a function that projections vectors x into a new vector space. RBF’s trick is to compute the similarities between data points in a high dimensional space without computing where the points lie in this space and projects them into infinite dimensional feature space, where separating hyperplane or boundary can be easily marked.

This is the reason SVM’s RBF outperforms above mentioned datasets by classifying non-linear dataset.

1. **Two Gaussian 33** and **42:** For these two datasets SVM and QDA performed the best, but QDA will be the best classifier to select, The reason to choose QDA is the linear behavior of dataset and less misclassified samples, complexity as compared to SVM, even the time taken by an algorithm to classify the data is less by QDA which will be very useful for large dataset.

QDA assumes that the predictor variables X is of multivariate Gaussian or normal distribution, also each class has its covariance and can apprehend the differing covariances and provide more accurate non-linear classification decision boundaries. Probability density function for a multivariate Gaussian distribution is:



QDA computes separate \boldsymbol\Sigma and \boldsymbol\mu for each class. When data points of different classes merge with each other, use of QDA is better and also dataset of gaussian distribution and easy to put into quadratic equation. **So, as in above two datasets there are not many points mixed with each other, SVM is more ambitious and complex approach**.

**6).** Sketch of proof of how Naïve Bayes that uses Gaussian distributions is a simplified form of the quadratic (optimal Bayesian) classifier.

* Naïve Bayes assume **xj** is conditionally independent given **y**. In other words,

P (xj | y) = P (xj | y, xk). P (x| y) is assumed to be a Gaussian distribution with mean μ and covariance matrix Σ. ***If the Σ is unconstrained, which finally end up with cross product terms, leading to quadratic decision boundaries****.*

The gaussian normal density function is defined as:

A close up of a watch

Description automatically generated

By using Bayes rule, discriminant function becomes:

A picture containing object

Description automatically generated

After eliminating constant terms from above equation:

A picture containing object

Description automatically generated

Then take natural logs and we gets the below equation.



Above expression is called a quadratic discriminant function.

* To prove let’s compare with Quadratic optimal Bayesian equation. QDA allows different covariance matrices, Σy for each class y.

A screenshot of a cell phone

Description automatically generated

If in the QDA model we assume that the covariance matrices are diagonal, the inputs are then assumed to be conditionally independent in each class. In this case, the quadratic discriminant function becomes:

A picture containing object

Description automatically generated

Hence, we can see Naïve Bayes that uses Gaussian distributions is a simplified form of the quadratic (optimal Bayesian) classifier.