Shorabh Dhandharia
A20361514

HW2 – Generative Learning

Problem Statement

In this assignment, I have implemented different algorithms for generative learning. For Gaussian Discriminant Analysis I have used the "iris" dataset and for Naïve Bayes I have used "Spambase" dataset. For cross validation, I have used previous assignment's technique of 10-fold cross validation. I have used the Python's inbuilt function to compute and confusion matrix and calculated precision, recall, F-measure and accuracy manually.

Proposed Solution

1. *1D 2-Class Gaussian Discriminant Analysis*
   Implement a 2 class dataset with continuous single feature by estimating the model parameters and computing discriminant function for each classes.
2. *nD 2-Class Gaussian Discriminant Analysis*
   Implement a 2 class dataset with continuous n-dimensional features by estimating the model parameters and computing discriminant function for each classes.
3. *nD k-Class Gaussian Discriminant Analysis*
   Implement a k class dataset with continuous n-dimensional features by estimating the model parameters and computing discriminant function for each classes.
4. *Naïve Bayes with Bernoulli Features*
   Implement a 2 class dataset with binary features derived from text documents, by estimating the model parameters and computing discriminant function for each classes using Naïve Bayes assumption.
5. *Naïve Bayes with Binomial Features*
   Derive the parameter estimate equations for Naïve Bayes Binomial Features using maximum likelihood.
   Implement a 2 class dataset with discrete features derived from text documents, by estimating the model parameters and computing discriminant function for each classes using Naïve Bayes assumption.

Classify the examples and compute confusion matrix, precision, recall and F-measure for all the algorithms and plot the precision recall curve for nD 2-Class Gaussian Discriminant Analysis

Implementation Details

For GDA I am using the iris dataset which consists of 4 features – length and width of sepals and petals. The algorithm uses these features to classify to which of the three species – Iris Setosa, Iris virginica and Iris versicolor.

For Naïve Bayes I have used text classification dataset called as Spambase which tells whether an email is a spam or not. It contains around 57 features which are frequencies of different words used to classify spam or not. For the one with Bernoulli Features I replaced all the decimal values to 1 since the algorithm only works on binomial data. It will hence classify spam or not based on the words present in that example. Whereas for Binomial features I assumed each example as a document and considered a random value for its length to calculate the word count and total no. of words in that document. The algorithm then classifies into spam or not based on the count of words in that document.

The Implementation of the code is almost similar in all the five algorithms as below:

- Reading the entire dataset into Z matrix
- Reshuffling the rows to perform cross validation correctly
- Split the matrix into X and Y
- Strip the data to get float values
- Modify feature vector in case of Naïve Bayes-Bernoulli and compute document length, word count, total no. of words in case of Naive Bayes with Binomial features
- Apply Cross Validation
    - Divide the Data into Training and Test
    - Calculate count of each class in Training data
    - Estimate the model parameters
    - Compute the discriminant function
    - Classify the examples and compute different measures

Results and Discussions

Below Tables Are Measures Calculated without Cross Validation –

Without cross validation, Precision, Recall and Accuracy comes more than 0.85 since the dataset is very small. But as dataset increases, we can see from Naïve Bayes Analysis that it decreases.

Gaussian Discriminant Analysis:

1D 2-Class	*nD 2-Class*	*nD k-Class*
Confusion Matrix: [[42 8] [8 42]] Precision is: [0.84, 0.84] Recall is: [0.84, 0.84] F-Measure is: [0.84, 0.84] Accuracy is: 0.84	Confusion Matrix: [[49 1] [0 50]] Precision is: [0.98,1.0] Recall is: [1.0, 0.98] F-Measure is: [0.99,0.99] Accuracy is: 0.99	Confusion Matrix [[50 0 0] [0 49 1] [0 1 49]] Precision is: [1.0,0.98,0.98] Recall is: [1.0,0.98,0.98] F-Measure is: [1.0,0.98,0.98] Accuracy is: 0.99

Naïve Bayes Analysis:

Bernoulli Features	*Binomial Features*
Confusion Matrix: [[2600 188] [334 1479]] Precision is: [0.93,0.82] Recall is: [0.89,0.89] F-Measure is: [0.91,0.85] Accuracy is: 0.89	Confusion Matrix [[2242 546] [79 1734]] Precision is: [0.80,0.96] Recall is: [0.97,0.76] F-Measure is: [0.88,0.85] Accuracy is: 0.86

Below Tables Are Measures Calculated with Cross Validation –

After applying cross validation, the size of test data comes down to 10 and hence the algorithm almost 99.5% of the time classifies correctly. And hence the Precision, Recall and Accuracy almost comes to 1.0 in case of GDA but in case of Naïve Bayes the measures remains almost the same as without Cross Validation.

Gaussian Discriminant Analysis:

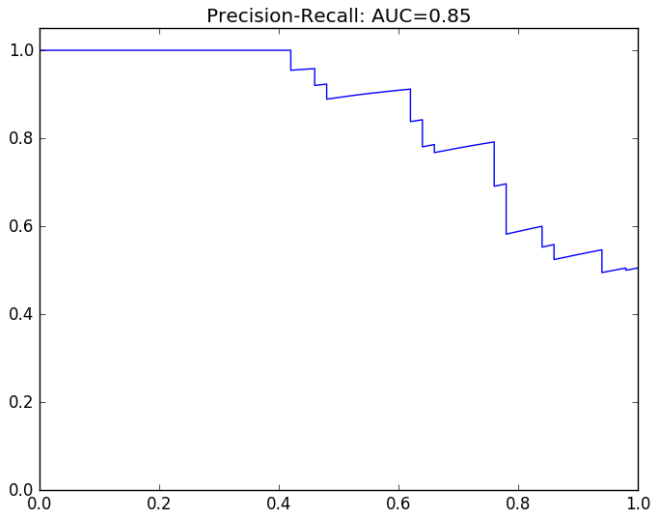
1D 2-Class	*nD 2-Class*	*nD k-Class*
Confusion Matrix: [[4 0] [0 6]] Precision is: [1.0, 1.0] Recall is: [1.0, 1.0] F-Measure is: [1.0, 1.0] Accuracy is: 1.0	Confusion Matrix: [[7 0] [0 3]] Precision is: [1.0, 1.0] Recall is: [1.0, 1.0] F-Measure is: [1.0, 1.0] Accuracy is: 1.0	Confusion Matrix: [[5 0 0] [0 7 1] [0 0 2]] Precision is: [1.0, 0.875, 1.0] Recall is: [1.0, 1.0, 0.67] F-Measure is: [1.0, 0.93, 0.80] Accuracy is: 0.93

Naïve Bayes Analysis:

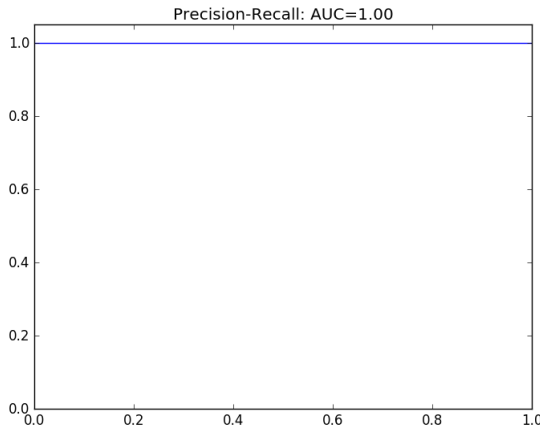
Bernoulli Features	*Binomial Features*
Confusion Matrix: [[278 17] [40 126]] Precision is: [0.94, 0.76] Recall is: [0.87, 0.88] F-Measure is: [0.91, 0.82] Accuracy is: 0.88	Confusion Matrix: [[210 64] [8 179]] Precision is: [0.77, 0.96] Recall is: [0.96, 0.74] F-Measure is: [0.85, 0.83] Accuracy is: 0.84

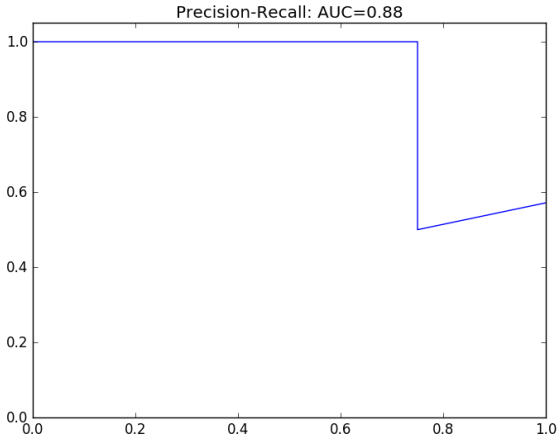
Precision Recall Curve for nD 2-Class Gradient Discriminant Analysis:

The below figure is the PR curve taken for nD 2-Class GDA without cross validation. I have also calculated the area under the curve (AUC) which can be seen in the title of the graph which in this case is 0.85



The below figures are the PR curve taken for nD 2-Class GDA with Cross Validation for different folds. The area under the curve (AUC) can be seen in the title of the graph.





From the figures, I understand that when I was plotting without cross validation there were around 100 samples and hence the graph was as it should have been. But when I was plotting with cross validation there were only 10 samples whose precision were always equal to 1 as we can see from the above left hand figure. In one or two folds out of 10, I got one curve like the right hand figure. Also, the AUC for the left one is 1 for exactly the same reason mentioned before and for the right hand is 0.88.

References

http://scikit-learn.org/stable/auto_examples/model_selection/plot_precision_recall.html

Note:

The derivation document is placed along with this report in the same folder.

All the codes have been executed and evaluated for the datasets mentioned above.
To run the code successfully you need to change the file name and location in all the codes, references are provided below.
1D2Class.py:5, ND2Class.py:7, NDKClass.py:5, NB_Bernoulli.py:5, NB_Binomial.py:7