P2-DiabetesData

Jeetendra Gan

11/4/2019

I have written a python script to read the text data and convert it to csv. The feature names have been simplified.

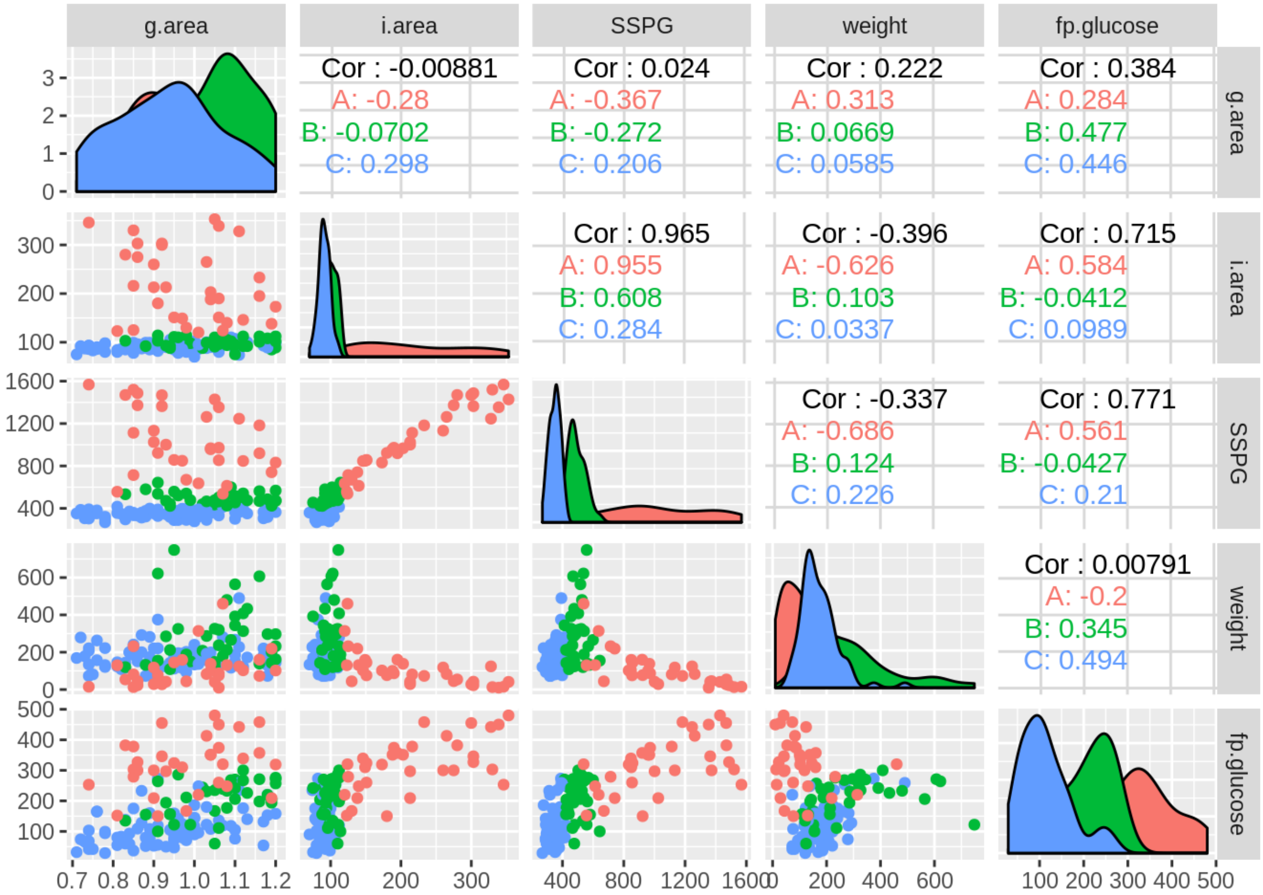
```
##      g.area           i.area         SSPG            weight
##  Min.   :0.7100   Min.   : 70   Min.   : 269.0   Min.   : 10.0
##  1st Qu.:0.8800   1st Qu.: 90   1st Qu.: 352.0   1st Qu.:118.0
##  Median :0.9800   Median : 97   Median : 413.0   Median :156.0
##  Mean   :0.9773   Mean   :122   Mean   : 543.6   Mean   :186.1
##  3rd Qu.:1.0800   3rd Qu.:112   3rd Qu.: 558.0   3rd Qu.:221.0
##  Max.   :1.2000   Max.   :353   Max.   :1568.0   Max.   :748.0
##    fp.glucose        class
##  Min.   : 29.0   Min.   :1.000
##  1st Qu.:100.0   1st Qu.:2.000
##  Median :159.0   Median :3.000
##  Mean   :184.2   Mean   :2.297
##  3rd Qu.:257.0   3rd Qu.:3.000
##  Max.   :480.0   Max.   :3.000
```

```
## [1] "g.area"     "i.area"     "SSPG"       "weight"     "fp.glucose"
## [6] "class"
```

Here are the prior probabilities of each class from class 1, class 2, followed by class 3, resp.

```
## [1] 0.2275862 0.2482759 0.5241379
```

Here is the pariwise scatter-plot for each of the five variables



Classes have difference covariance matrix The variances of feature distribution for classes are different as can be seen in the plot above. Consider, i.area, the variance for class 1 is larger than that for classes 2, and 3.

The classes are not multi-variate normal LDA assumes that the density of features for a given class is normall distributed. It can be seen from the image that the feature densities do not have an ideal normal distribution but are very close to normal. For i.area and SSPG, the distribution of class A(1), may not look normal, but it can be assumed to be normal with a large variance.

Test-train division

The data is split into 70% training and 25% test set. In the training data, the data points belonging to class 1, class 2, and class 3 are shown below.

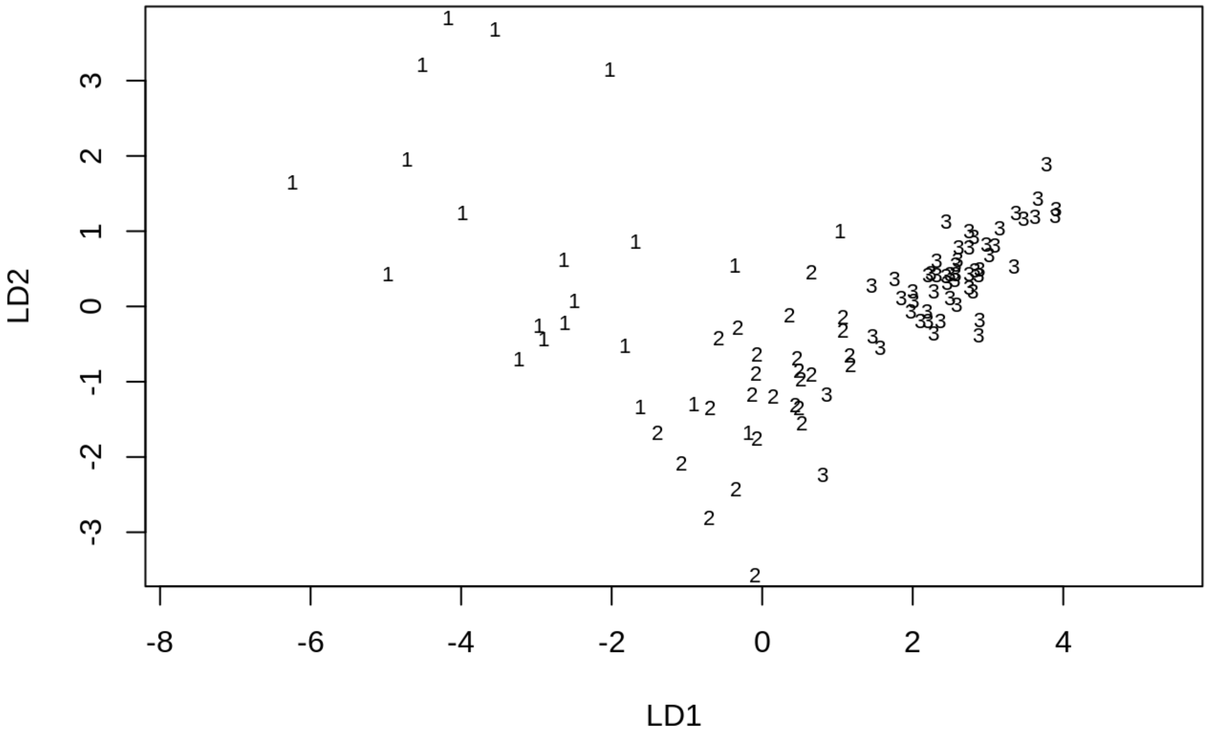
```
## [1] 21 26 54
```

The the test data, the count of data points belonging to class 1, class 2, and class 3 are shown below.

```
## [1] 12 10 22
```

LDA

```
## Call:
## lda(class ~ ., data = trainData)
##
## Prior probabilities of groups:
##         1         2         3
## 0.2079208 0.2574257 0.5346535
##
## Group means:
##      g.area    i.area     SSPG   weight fp.glucose
## 1 0.9795238 204.57143 998.7619 128.7143   306.3810
## 2 1.0484615 100.76923 504.3846 277.4231   214.7308
## 3 0.9416667  91.42593 352.1852 171.0741   116.0926
##
## Coefficients of linear discriminants:
##                     LD1          LD2
## g.area     -0.979571205 -3.420970121
## i.area      0.039091321  0.033732203
## SSPG       -0.013423147 -0.005573391
## weight     -0.001082993 -0.005957180
## fp.glucose -0.005304303  0.000193961
##
## Proportion of trace:
##    LD1    LD2
## 0.8922 0.1078
```



The predictions done by LDA are as follows:

```
##
##      1  2  3
##   1 11  0  0
##   2  1  7  1
##   3  0  3 21
```

The miss-classification percentage for LDA on test data is:

```
## [1] 11.36364
```

LDAs performance on train data:

```
##
##      1  2  3
##   1 16  0  0
##   2  4 23  2
##   3  1  3 52
```

Miss-classifications on train data:

```
## [1] 9.90099
```

QDA

```
## Call:
## qda(class ~ ., data = trainData)
##
## Prior probabilities of groups:
##         1         2         3
## 0.2079208 0.2574257 0.5346535
##
## Group means:
##      g.area    i.area     SSPG   weight fp.glucose
## 1 0.9795238 204.57143 998.7619 128.7143   306.3810
## 2 1.0484615 100.76923 504.3846 277.4231   214.7308
## 3 0.9416667  91.42593 352.1852 171.0741   116.0926
```

The predictions done by QDA on test data are as follows:

```
##
##      1  2  3
##   1 12  1  0
##   2  0  7  1
##   3  0  2 21
```

The miss-classification percentage of QDA on test data is:

```
## [1] 9.090909
```

Performance of QDA on train data:

```
##
##      1  2  3
##   1 12  1  0
##   2  0  7  1
##   3  0  2 21
```

The miss-classification percentage of QDA on train data is:

```
## [1] 3.960396
```

Performance of LDA and QDA

I did multiple tests by setting different seed values. In general QDA performed better than LDA. As seen above, for the current seed, QDA’s test and train error are lesser than LDA’s test and train error. The above observation might be due to the violation of the rule/assumption that all the features have the same variance. Violation of the above rule may make LDA a weaker candidate. QDA works when the variances are not the same.

Classification results for test sample

Following is the test sample:

```
##   g.area i.area SSPG weight fp.glucose
## 1   0.98    122  544    186        184
```

LDA assigns this individual to:

```
## [1] 3
## Levels: 1 2 3
```

The posterior probabilities are shown below. It can be seen that LDA assigns the data point to class 3 with a probability which is only slightly greater than that for class 2.

```
##             1        2         3
## 1 0.002148242 0.475594 0.5222577
```

QDA assigns the individual to:

```
## [1] 2
## Levels: 1 2 3
```

The posterior probabilities are shown below. We can say that QDA classifies the data point very confidently to class 2.

```
##           1         2            3
## 1 0.4135972 0.5863852 1.756332e-05
```