Assignment 2 Report

Yufei Liu

260561054

**Question 1**

The 3 generated datasets, *DS1-test.csv, DS1-train.csv* and *DS1-valid.csv* are stored under directory *hwk2\_datasets*.

**Question 2**

1.

a.

*Accuracy = 0.95875*

*Precision = 0.9508599508599509*

*Recall = 0.9675*

*F1 measure = 0.9591078066914498*

The parameters are stored as a text file ***Assignment2\_260561054\_2\_1\_a.txt***under directory *hwk2\_datasets*.

b.

*w0 = 1.9463125787853652*

*w1 = [[ 1.02193507]*

*[-0.61584269]*

*[-0.44156509]*

*[-0.27127862]*

*[-0.74051604]*

*[-0.29672473]*

*[ 1.30321243]*

*[-1.71359568]*

*[-2.13595237]*

*[ 0.62525576]*

*[-0.91837962]*

*[-0.91067288]*

*[ 1.1556343 ]*

*[ 0.99645741]*

*[-0.38852566]*

*[ 0.91743229]*

*[ 2.18270508]*

*[-0.48336646]*

*[-0.1325924 ]*

*[-0.33832297]]*

The parameters are stored as a text file ***Assignment2\_260561054\_2\_1\_b.txt***under directory *hwk2\_datasets*.

**Question 3**

By setting range of K from 1 to 20, when K = 2 the model gives the best fit with

*Best K = 2*

*Accuracy = 0.49375*

*Precision = 0.4746450304259635*

*Recall = 0.6157894736842106*

*F1 measure = 0.5360824742268041*

The k-NN classifier performs considerably worse than GDA. When testing the k-NN classifier, I used SciKit learn to normalize the dataset and which is necessary because without normalization, one feature may dominate the distance measure.

The k-NN classifier is worse here because our datasets have really high dimensions, the distances are going to be less representative.

The parameters are stored as a text file ***Assignment2\_260561054\_3\_b.txt***under directory *hwk2\_datasets*.

**Question 4**

The 3 generated datasets, *DS2-test.csv, DS2-train.csv* and *DS2-valid.csv* are stored under directory *hwk2\_datasets*.

**Question 5**

1.

a.

*Accuracy = 0.535*

*Precision = 0.5333333333333333*

*Recall = 0.56*

*F1 measure = 0.5463414634146342*

The parameters are stored as a text file ***Assignment2\_260561054\_5\_1\_a.txt***under directory *hwk2\_datasets*.

b.

*w0 = 0.00944858874154364*

*w1 = [[-0.01587187]*

*[-0.03097841]*

*[ 0.02603874]*

*[-0.00485993]*

*[ 0.00947184]*

*[ 0.00059935]*

*[ 0.01926517]*

*[ 0.08241396]*

*[-0.02991102]*

*[ 0.01269773]*

*[-0.01779393]*

*[ 0.00205633]*

*[ 0.00282658]*

*[-0.02220273]*

*[-0.0197494 ]*

*[ 0.02114536]*

*[-0.02914988]*

*[-0.03820653]*

*[ 0.01388067]*

*[ 0.01200783]]*

The parameters are stored as a text file ***Assignment2\_260561054\_5\_1\_b.txt***under directory *hwk2\_datasets*.

2.

*Best K = 4*

*Accuracy = 0.5275*

*Precision = 0.5267034990791897*

*Recall = 0.7027027027027027*

*F1 measure = 0.6021052631578947*

The k-NN classifier’s performance does not change that much on dataset 2 comparing to GDA, who’s accuracy, precision and recall drops dramatically. We will discuss the reason in the next question.

3.

The parameters are stored as a text file ***Assignment2\_260561054\_5\_3.txt***under directory *hwk2\_datasets*.

**Question 6**

GDA classifier

The measures drop dramatically from dataset 1 to dataset 2. This is expected because dataset 2 is generated by a mixture of 3 Gaussians with different covariance matrix.

It violates the assumption of GDA:

* All classes share the same covariance matrix
* Class conditional densities are Gaussian

k-NN classifier

By applying k-NN classifier on both dataset 1 and dataset 2, the result did not fluctuate too much comparing to the GDA classifier. This is because k-NN classifier does not have explicit assumptions on dataset. However, k-NN classifier has much worse performance on dataset 1.