**Student’s Name: Yash Sharma**

**Roll Number: B20241**

**Mobile No: 8802131138**

**Branch:** **CSE**

# a.

|  |  |  |
| --- | --- | --- |
|  | **Prediction Outcome** | |
| **True Label** | 93 | 25 |
| 19 | 200 |

Figure 1 KNN Confusion Matrix for K = 1

|  |  |  |
| --- | --- | --- |
|  | **Prediction Outcome** | |
| **True Label** | 92 | 26 |
| 9 | 210 |

Figure 2 KNN Confusion Matrix for K = 3

|  |  |  |
| --- | --- | --- |
|  | **Prediction Outcome** | |
| **True Label** | 92 | 26 |
| 10 | 209 |

Figure 3 KNN Confusion Matrix for K = 5

**b.**

Table 1 KNN Classification Accuracy for K = 1, 3 and 5

|  |  |
| --- | --- |
| **K** | **Classification**  **Accuracy (in %)** |
| 1 | **86.944** |
| 3 | **89.614** |
| 4 | **89.318** |

# Inferences:

1. The highest classification accuracy is obtained with K =2.
2. Increasing the value of k increases the accuracy for some iterations but only up to some limit i.e., k=3 here and then their onwards it seems to be decreasing.
3. When k=1 we estimate our probability based on a single sample: the closest neighbor. This is very sensitive to all sort of distortions like noise, outliers, mislabeling of data, and so on. By using a higher value for k, we tend to be more robust against those distortions. Thus, increasing k seems to be increasing the classification accuracy.
4. As the classification accuracy increases with the increase in value of K i.e., k from 1 to 3, the number of diagonal elements increases.
5. As we increase k, the areas predicting each class will be more "smoothed", since it's the majority of the k-nearest neighbors which decide the class of any point. Thus, it makes our model predict more test samples correctly with their respective positive or negative class. Thus, we see increase in diagonal elements.
6. As the classification accuracy increases with the increase in value of K i.e., k from 1 to 3, the number of off-diagonal elements decreases.
7. As we see the diagonal elements decrease as k increases as accuracy increases thus proportionally off-diagonal elements decreases (as total test samples are fixed) as now our model makes more accurate and correct predictions (diagonal elements) thereby decreasing off-diagonal elements.

# a.

|  |  |  |
| --- | --- | --- |
|  | **Prediction Outcome** | |
| **True Label** | 111 | 7 |
| 6 | 213 |

Figure 4 KNN Confusion Matrix for K = 1 post data normalization

|  |  |  |
| --- | --- | --- |
|  | **Prediction Outcome** | |
| **True Label** | 112 | 6 |
| 5 | 214 |

Figure 5 KNN Confusion Matrix for K = 3 post data normalization

|  |  |  |
| --- | --- | --- |
|  | **Prediction Outcome** | |
| **True Label** | 111 | 7 |
| 3 | 216 |

Figure 6 KNN Confusion Matrix for K = 5 post data normalization

**b.**

Table 2 KNN Classification Accuracy for K = 1, 3 and 5 post data normalization

|  |  |
| --- | --- |
| **K** | **Classification**  **Accuracy (in %)** |
| 1 | **96.1424** |
| 3 | **96.7359** |
| 5 | **97.0326** |

# Inferences:

1. Data normalization increases classification accuracy.
2. As we know KNN method is based on finding the Euclidean distance between a given test tuple with all other input tuples. Euclidean distance calculated on non-normalized data does not give real distance between those two tuples as any attribute in data set with big range can caused a biased result while calculating the Euclidean distance, Thus, normalized data gives the accurate Euclidean distance thereby increasing the accuracy as seen.
3. The highest classification accuracy is obtained with K =5.
4. Increasing the value of k increases the accuracy.
5. When k=1 we estimate our probability based on a single sample: the closest neighbor. This is very sensitive to all sort of distortions like noise, outliers, mislabeling of data, and so on. By using a higher value for k, we tend to be more robust against those distortions. Thus, increasing k seems to be increasing the classification accuracy.
6. As the classification accuracy increases with the increase in value of K i, the number of diagonal elements increases.
7. As we increase k, the areas predicting each class will be more "smoothed", since it's the majority of the k-nearest neighbors which decide the class of any point. Thus, it makes our model predict more test samples correctly with their respective positive or negative class. Thus, we see increase in diagonal elements.
8. As the classification accuracy increases with the increase in value of K, the number of off-diagonal elements decreases.
9. As we see the diagonal elements decrease with increase in k and thereby accuracy increases. Thus, proportionally off-diagonal elements decrease (as total test samples are fixed) and now our model makes more accurate and correct predictions (diagonal elements) thereby decreasing off-diagonal elements.

|  |  |  |
| --- | --- | --- |
|  | **Prediction Outcome** | |
| **True Label** | 102 | 16 |
| 3 | 216 |

Figure 7 Confusion Matrix obtained from Bayes Classifier

The classification accuracy obtained from Bayes Classifier is 94.362 %.

Table 3 Mean for class 0 and class 1

|  |  |  |  |
| --- | --- | --- | --- |
| **S. No.** | **Attribute Name** | **Mean** | |
| **Class 0** | **Class 1** |
|  | X\_Minimum | **Removed** | |
|  | X\_Maximum | 273.418 | 723.656 |
|  | Y\_Minimum | **Removed** | |
|  | Y\_Maximum | 1583169.659 | 1431588.69 |
|  | Pixels\_Areas | 7779.663 | 585.967 |
|  | X\_Perimeter | 393.835 | 54.491 |
|  | Y\_Perimeter | 273.183 | 45.658 |
|  | Sum\_of\_Luminosity | 843350.275 | 62191.126 |
|  | Minimum\_of\_Luminosity | 53.326 | 96.236 |
|  | Maximum\_of\_Luminosity | 135.762 | 130.452 |
|  | Length\_of\_Conveyer | 1382.762 | 1480.018 |
|  | TypeOfSteel\_A300 | **Removed** | |
|  | TypeOfSteel\_A400 | **Removed** | |
|  | Steel\_Plate\_Thickness | 40.073 | 104.214 |
|  | Edges\_Index | 0.123 | 0.385 |
|  | Empty\_Index | 0.459 | 0.427 |
|  | Square\_Index | 0.592 | 0.513 |
|  | Outside\_X\_Index | 0.108 | 0.02 |
|  | Edges\_X\_Index | 0.55 | 0.608 |
|  | Edges\_Y\_Index | 0.523 | 0.831 |
|  | Outside\_Global\_Index | 0.288 | 0.608 |
|  | LogOfAreas | 3.623 | 2.287 |
|  | Log\_X\_Index | 2.057 | 1.227 |
|  | Log\_Y\_Index | 1.848 | 1.318 |
|  | Orientation\_Index | -0.314 | 0.136 |
|  | Luminosity\_Index | -0.115 | -0.116 |
|  | SigmoidOfAreas | 0.925 | 0.543 |

In Fig. 8 and 9 representing covariance matrices for class 0 and class 1 respectively the column numbers and row numbers correspond to attribute with serial number as in Table 3.



Figure 8: Covariance matrix for class 0



Figure 9: Covariance matrix for class 1

# Inferences:

1. Accuracy of Bayes Classifier is 94.362 %. The reason that it is lesser than the accuracy obtained via “KNN on normalized data” (97.003 %) can be because K-NN does better because of its inherent nature to optimize locally and make predictions based on maximum class of the k-nearest neighbors around it. But still we see Bayes Classifier still isn’t very far from the accuracy 97.003 % thus it may not be regarded as a bad approach just based upon this data prediction results.
2. As seen from covariance matrix the diagonal elements are all positive and the initial values also are very big in magnitude. They are positive because they represent the variance of the respective attribute.
3. The off-diagonal elements in the covariance matrix are both negative and positive are of varying magnitude (some of big magnitude and some of less). They simply represent the covariance between any two given attributes. The attributes “Y\_Maximum” and “Sum\_of\_Luminosity” have highest covariance while attributes “X\_Maximum” and “Y\_Maximum” have least covariance for class 0 and The attributes “Y\_Maximum” and “Sum\_of\_Luminosity” have highest covariance while attributes “Steel\_Plate\_Thickness” and “Y\_Maximum” have least covariance for class 1.

Table 4 Comparison between classifiers based upon classification accuracy

|  |  |  |
| --- | --- | --- |
| **S. No.** | **Classifier** | **Accuracy (in %)** |
|  | KNN | 89.614 |
|  | KNN on normalized data | 97.033 |
|  | Bayes | 94.362 |

# Inferences:

1. K-NN classifier on normalized data has the highest accuracy while K-NN classifier on not-normalized data has least accuracy.
2. KNN Classifiers < Bayes Classifier < K-NN Classifier on normalized data.
3. The reason behind the fact that K-NN performed on normalized data has higher accuracy than K-NN performed on normal data is as follow. As we know KNN method is based on finding the Euclidean distance between a given test tuple with all other input tuples. Euclidean distance calculated on non-normalized data does not give real distance between those two tuples as any attribute in data set with big range can caused a biased result while calculating the Euclidean distance, Thus, normalized data gives the accurate Euclidean distance thereby increasing the accuracy as seen.
4. The reason for the Bayes classifier having low accuracy than K-NN (on normalized data) is as follow. K-NN does better because of its inherent nature to optimize locally and make predictions based on maximum class of the k-nearest neighbors around it. But still we see Bayes Classifier still isn’t very far from the accuracy 97.003 % thus it may not be regarded as a bad approach just based upon this data prediction results.
5. Generally, Bayes Classifier is more accurate and beneficial on complex data sets as their K-NN method would fail due to its high time complexity and its inherent nature to optimize locally. Here, data set is not that much big and complex thus K-NN shows best accuracy among all other classifiers.