IC 272: DATA SCIENCE - III

LAB ASSIGNMENT – IV

Data classification using K-nearest neighbor classifier and Bayes classifier with unimodal Gaussian density

Student's Name: Yash Sharma **Mobile No: 8802131138**

Roll Number: B20241 **Branch: CSE**

1 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.

2 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.

3

	Prediction Outcome	
True Label	102	16
	3	216

Figure 7 Confusion Matrix obtained from Bayes Classifier

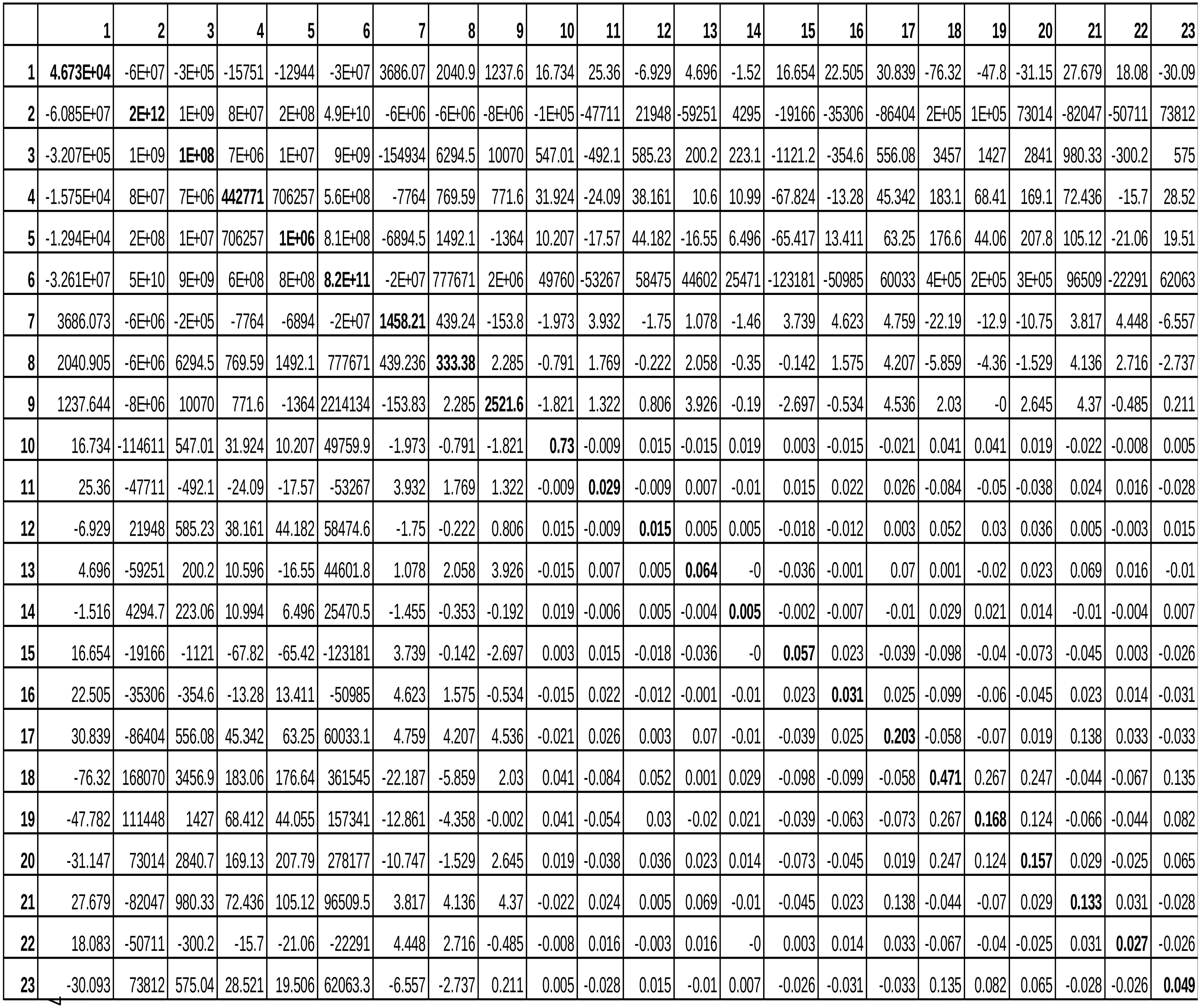
The classification accuracy obtained from Bayes Classifier is 94.362 %.

Table 3 Mean for class 0 and class 1

S. No.	Attribute Name	Mean	
		Class 0	Class 1
1.	X_Minimum	Removed	
2.	X_Maximum	273.418	723.656
3.	Y_Minimum	Removed	
4.	Y_Maximum	1583169.659	1431588.69
5.	Pixels_Areas	7779.663	585.967
6.	X_Perimeter	393.835	54.491
7.	Y_Perimeter	273.183	45.658
8.	Sum_of_Luminosity	843350.275	62191.126
9.	Minimum_of_Luminosity	53.326	96.236
10.	Maximum_of_Luminosity	135.762	130.452
11.	Length_of_Conveyer	1382.762	1480.018
12.	TypeOfSteel_A300	Removed	
13.	TypeOfSteel_A400	Removed	
14.	Steel_Plate_Thickness	40.073	104.214
15.	Edges_Index	0.123	0.385
16.	Empty_Index	0.459	0.427
17.	Square_Index	0.592	0.513
18.	Outside_X_Index	0.108	0.02
19.	Edges_X_Index	0.55	0.608
20.	Edges_Y_Index	0.523	0.831
21.	Outside_Global_Index	0.288	0.608
22.	LogOfAreas	3.623	2.287

23.	Log_X_Index	2.057	1.227
24.	Log_Y_Index	1.848	1.318
25.	Orientation_Index	-0.314	0.136
26.	Luminosity_Index	-0.115	-0.116
27.	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.



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
1	**4.673E+04**	-6E+07	-3E+05	-15751	-12944	-3E+07	3686.07	2040.9	1237.6	16.734	25.36	-6.929	4.696	-1.52	16.654	22.505	30.839	-76.32	-47.8	-31.15	27.679	18.08	-30.09
2	-6.085E+07	**2E+12**	1E+09	8E+07	2E+08	4.9E+10	-6E+06	-6E+06	-8E+06	-1E+05	-47711	21948	-59251	4295	-19166	-35306	-86404	2E+05	1E+05	73014	-82047	-50711	73812
3	-3.207E+05	1E+09	**1E+08**	7E+06	1E+07	9E+09	-154934	6294.5	10070	547.01	-492.1	585.23	200.2	223.1	-1121.2	-354.6	556.08	3457	1427	2841	980.33	-300.2	575
4	-1.575E+04	8E+07	7E+06	**442771**	706257	5.6E+08	-7764	769.59	771.6	31.924	-24.09	38.161	10.6	10.99	-67.824	-13.28	45.342	183.1	68.41	169.1	72.436	-15.7	28.52
5	-1.294E+04	2E+08	1E+07	706257	**1E+06**	8.1E+08	-6894.5	1492.1	-1364	10.207	-17.57	44.182	-16.55	6.496	-65.417	13.411	63.25	176.6	44.06	207.8	105.12	-21.06	19.51
6	-3.261E+07	5E+10	9E+09	6E+08	8E+08	**8.2E+11**	-2E+07	777671	2E+06	49760	-53267	58475	44602	25471	-123181	-50985	60033	4E+05	2E+05	3E+05	96509	-22291	62063
7	3686.073	-6E+06	-2E+05	-7764	-6894	-2E+07	**1458.21**	439.24	-153.8	-1.973	3.932	-1.75	1.078	-1.46	3.739	4.623	4.759	-22.19	-12.9	-10.75	3.817	4.448	-6.557
8	2040.905	-6E+06	6294.5	769.59	1492.1	777671	439.236	**333.38**	2.285	-0.791	1.769	-0.222	2.058	-0.35	-0.142	1.575	4.207	-5.859	-4.36	-1.529	4.136	2.716	-2.737
9	1237.644	-8E+06	10070	771.6	-1364	2214134	-153.83	2.285	**2521.6**	-1.821	1.322	0.806	3.926	-0.19	-2.697	-0.534	4.536	2.03	-0	2.645	4.37	-0.485	0.211
10	16.734	-114611	547.01	31.924	10.207	49759.9	-1.973	-0.791	-1.821	**0.73**	-0.009	0.015	-0.015	0.019	0.003	-0.015	-0.021	0.041	0.041	0.019	-0.022	-0.008	0.005
11	25.36	-47711	-492.1	-24.09	-17.57	-53267	3.932	1.769	1.322	-0.009	**0.029**	-0.009	0.007	-0.01	0.015	0.022	0.026	-0.084	-0.05	-0.038	0.024	0.016	-0.028
12	-6.929	21948	585.23	38.161	44.182	58474.6	-1.75	-0.222	0.806	0.015	-0.009	**0.015**	0.005	0.005	-0.018	-0.012	0.003	0.052	0.03	0.036	0.005	-0.003	0.015
13	4.696	-59251	200.2	10.596	-16.55	44601.8	1.078	2.058	3.926	-0.015	0.007	0.005	**0.064**	-0	-0.036	-0.001	0.07	0.001	-0.02	0.023	0.069	0.016	-0.01
14	-1.516	4294.7	223.06	10.994	6.496	25470.5	-1.455	-0.353	-0.192	0.019	-0.006	0.005	-0.004	**0.005**	-0.002	-0.007	-0.01	0.029	0.021	0.014	-0.01	-0.004	0.007
15	16.654	-19166	-1121	-67.82	-65.42	-123181	3.739	-0.142	-2.697	0.003	0.015	-0.018	-0.036	-0	**0.057**	0.023	-0.039	-0.098	-0.04	-0.073	-0.045	0.003	-0.026
16	22.505	-35306	-354.6	-13.28	13.411	-50985	4.623	1.575	-0.534	-0.015	0.022	-0.012	-0.001	-0.01	0.023	**0.031**	0.025	-0.099	-0.06	-0.045	0.023	0.014	-0.031
17	30.839	-86404	556.08	45.342	63.25	60033.1	4.759	4.207	4.536	-0.021	0.026	0.003	0.07	-0.01	-0.039	0.025	**0.203**	-0.058	-0.07	0.019	0.138	0.033	-0.033
18	-76.32	168070	3456.9	183.06	176.64	361545	-22.187	-5.859	2.03	0.041	-0.084	0.052	0.001	0.029	-0.098	-0.099	-0.058	**0.471**	0.267	0.247	-0.044	-0.067	0.135
19	-47.782	111448	1427	68.412	44.055	157341	-12.861	-4.358	-0.002	0.041	-0.054	0.03	-0.02	0.021	-0.039	-0.063	-0.073	0.267	**0.168**	0.124	-0.066	-0.044	0.082
20	-31.147	73014	2840.7	169.13	207.79	278177	-10.747	-1.529	2.645	0.019	-0.038	0.036	0.023	0.014	-0.073	-0.045	0.019	0.247	0.124	**0.157**	0.029	-0.025	0.065
21	27.679	-82047	980.33	72.436	105.12	96509.5	3.817	4.136	4.37	-0.022	0.024	0.005	0.069	-0.01	-0.045	0.023	0.138	-0.044	-0.07	0.029	**0.133**	0.031	-0.028
22	18.083	-50711	-300.2	-15.7	-21.06	-22291	4.448	2.716	-0.485	-0.008	0.016	-0.003	0.016	-0	0.003	0.014	0.033	-0.067	-0.04	-0.025	0.031	**0.027**	-0.026
23	-30.093	73812	575.04	28.521	19.506	62063.3	-6.557	-2.737	0.211	0.005	-0.028	0.015	-0.01	0.007	-0.026	-0.031	-0.033	0.135	0.082	0.065	-0.028	-0.026	**0.049**

Figure 8: Covariance matrix for class 0

IC 272: DATA SCIENCE - III
LAB ASSIGNMENT – IV
Data classification using K-nearest neighbor classifier and Bayes classifier with unimodal Gaussian density

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
1	**256526.31**	1E+08	-22255	1101.1	-1974	-2E+06	-1224.8	-744	13220	-1933	8.914	-3.806	10.89	1.504	6.695	-5.018	-16.56	-13.78	5.306	-21.2	-25.9	-8.452	-14.22
2	111783525	**3E+12**	3E+08	2E+07	5E+06	3.3E+10	-4E+06	-43296	4E+06	-4E+07	23556	-19251	-38010	13457	64533	-22199	-74705	15298	64300	-63427	-1E+05	-14718	-37675
3	-22254.624	3E+08	**5E+06**	178492	129451	4.9E+08	-15632	-300.3	-23835	4262.2	-47.65	35.619	-90.63	52.91	-101.64	-96.06	55.178	653.1	330.8	355.1	65.419	-32.38	218.9
4	1101.079	2E+07	178492	**9807.2**	5546.9	1.9E+07	-570.12	30.15	-1447	282.11	-1.332	4.156	-7.318	3.972	-4.85	-9.176	-2.152	36.62	23.56	16.86	-3.758	-1.119	15.51
5	-1973.565	5E+06	129451	5546.9	**5000.6**	1.3E+07	-557.42	-79.15	-1139	438.56	-2.244	2.952	-6.496	1.204	-8.612	-2.367	7.11	29.03	10.68	21.03	11.045	-1.556	13.01
6	-2334975.6	3E+10	5E+08	2E+07	1E+07	**5.1E+10**	-1E+06	84723	-3E+06	343512	-4689	3985.1	-9653	5578	-10535	-10272	5462.3	67783	34740	36735	6364.1	-2282	22865
7	-1224.809	-4E+06	-15632	-570.1	-557.4	-1E+06	**733.909**	348.05	-993.3	-204.8	1.066	0.591	0.775	-0.15	0.427	-0.833	-2.224	-5.043	-1.3	-3.287	-2.503	3.684	-1.984
8	-744.043	-43296	-300.3	30.15	-79.15	84723	348.045	**406.46**	-381.1	-205.4	0.429	-0.025	-0.267	0.044	0.878	-1.09	-2.018	-1.504	0.678	-2.165	-2.874	2.786	-0.96
9	13220.079	4E+06	-23835	-1447	-1139	-3E+06	-993.31	-381.1	**23101**	1243.4	-0.09	-5.16	2.468	-0.7	6.591	1.971	-3.138	-7.953	-1.44	-10.57	-7.431	-4.547	-5.967
10	-1932.619	-4E+07	4262.2	282.11	438.56	343512	-204.84	-205.4	1243.4	**5645.3**	-1.331	0.699	-1.134	-0.17	-3.443	2.058	6.623	3.627	-1.38	5.403	7.846	-1.662	2.39
11	8.914	23556	-47.65	-1.332	-2.244	-4688.9	1.066	0.429	-0.09	-1.331	**0.09**	-0.001	0.011	0	0.008	-0.003	-0.017	-0.012	0.005	-0.017	-0.024	0.005	-0.004
12	-3.806	-19251	35.619	4.156	2.952	3985.08	0.591	-0.025	-5.16	0.699	-0.001	**0.02**	-0.002	0.001	-0.012	-0.011	-0.008	0.026	0.022	0.022	-0.004	0.002	0.024
13	10.893	-38010	-90.63	-7.318	-6.496	-9652.6	0.775	-0.267	2.468	-1.134	0.011	-0.002	**0.082**	-0	0.02	0.015	-0.016	-0.053	-0.02	-0.033	-0.021	0.001	-0.028
14	1.504	13457	52.909	3.972	1.204	5577.97	-0.151	0.044	-0.698	-0.165	0	0.001	-0.003	**0.002**	0.002	-0.005	-0.005	0.012	0.012	0.001	-0.008	0	0.005
15	6.695	64533	-101.6	-4.85	-8.612	-10535	0.427	0.878	6.591	-3.443	0.008	-0.012	0.02	0.002	**0.065**	-0.014	-0.068	-0.066	0.011	-0.086	-0.103	0.004	-0.045
16	-5.018	-22199	-96.06	-9.176	-2.367	-10272	-0.833	-1.09	1.971	2.058	-0.003	-0.011	0.015	-0.01	-0.014	**0.049**	0.064	-0.025	-0.06	0.024	0.086	-0.007	-0.017
17	-16.564	-74705	55.178	-2.152	7.11	5462.3	-2.224	-2.018	-3.138	6.623	-0.017	-0.008	-0.016	-0.01	-0.068	0.064	**0.227**	0.048	-0.07	0.113	0.229	-0.015	0.022
18	-13.781	15298	653.05	36.62	29.028	67782.7	-5.043	-1.504	-7.953	3.627	-0.012	0.026	-0.053	0.012	-0.066	-0.025	0.048	**0.271**	0.116	0.177	0.073	-0.019	0.147
19	5.306	64300	330.78	23.557	10.681	34740.3	-1.299	0.678	-1.44	-1.376	0.005	0.022	-0.021	0.012	0.011	-0.058	-0.073	0.116	**0.119**	0.017	-0.101	0	0.065
20	-21.204	-63427	355.12	16.864	21.025	36734.8	-3.287	-2.165	-10.57	5.403	-0.017	0.022	-0.033	0.001	-0.086	0.024	0.113	0.177	0.017	**0.178**	0.169	-0.017	0.103
21	-25.896	-119870	65.419	-3.758	11.045	6364.12	-2.503	-2.874	-7.431	7.846	-0.024	-0.004	-0.021	-0.01	-0.103	0.086	0.229	0.073	-0.1	0.169	**0.302**	-0.019	0.041
22	-8.452	-14718	-32.38	-1.119	-1.556	-2282.4	3.684	2.786	-4.547	-1.662	0.005	0.002	0.001	0	0.004	-0.007	-0.015	-0.019	0	-0.017	-0.019	**0.025**	-0.009
23	-14.221	-37675	218.95	15.508	13.014	22864.8	-1.984	-0.96	-5.967	2.39	-0.004	0.024	-0.028	0.005	-0.045	-0.017	0.022	0.147	0.065	0.103	0.041	-0.009	**0.102**

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.

4

Table 4 Comparison between classifiers based upon classification accuracy

S. No.	Classifier	Accuracy (in %)
1.	KNN	89.614
2.	KNN on normalized data	97.033
3.	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.