Assignment 5

Due Date: Saturday, November 17th, by midnight

Total number of points: 50 points

Problem 1 (25 points):

Download the seeds dataset from **http://archive.ics.uci.edu/ml/datasets/seeds#**

The examined data group comprised kernels belonging to three different varieties of wheat: Kama, Rosa and Canadian, 70 elements each, randomly selected for the experiment. High quality visualization of the internal kernel structure was detected using a soft X-ray technique. It is non-destructive and considerably cheaper than other more sophisticated imaging techniques like scanning microscopy or laser technology. The images were recorded on 13x18 cm X-ray KODAK plates. Studies were conducted using combine harvested wheat grain originating from experimental fields, explored at the Institute of Agrophysics of the Polish Academy of Sciences in Lublin.

Attribute information:

To construct the data, seven geometric parameters of wheat kernels were measured:
1. area A,
2. perimeter P,
3. compactness C = 4*pi*A/P^2,
4. length of kernel,
5. width of kernel,
6. asymmetry coefficient
7. length of kernel groove.

All of these parameters were real-valued continuous.

The last attribute in the data file represents the class label.

i. (15 points) Perform k-means clustering using all the attributes with the except of the class label, vary the number of clusters from 3 to 4 to 5 to 6 and report:
   a. How the cluster centers were calculated
      i. **Initially, cluster centers were chosen as random cases. Then distances to centers from all cases were calculated using the Euclidean distance measure. After that, cases were assigned to their nearest cluster centers. New centers were defined by the average of coordinates for all the points assigned to each specific cluster. This process was repeated until no significant change was observed or up to 100 iterations were completed.**

   b. What similarity measure was used
      i. **Euclidean Distance**

c. For each k, report the following:
   i. Final cluster centers

Final Cluster Centers (k=3)

	Cluster		
	1	2	3
area_A	18.72	11.96	14.65
perimeter_P	16.30	13.27	14.46
compactness_C	.8851	.8522	.8792
length_of_kernel	6.209	5.229	5.564
width_of_kernel	3.723	2.873	3.278
asymmetry_coefficient	3.6036	4.7597	2.6489
length_of_kernelgroove	6.066	5.089	5.192

Final Cluster Centers
(k=3 normalized)

	Cluster		
	1	2	3
area_A_transformed	.39	.76	.12
perimeter_P_transformed	.42	.80	.18
compactness_C_transformed	.67	.70	.38
length_of_kernel_transformed	.37	.73	.19
width_of_kernel_transformed	.47	.77	.16
asymmetry_coefficient_transformed	.27	.37	.50
length_of_kernelgroove_transformed	.32	.76	.28

Final Cluster Centers (k=4)

	Cluster			
	1	2	3	4
area_A	11.94	14.42	17.75	19.52
perimeter_P	13.27	14.35	15.88	16.65
compactness_C	.8515	.8795	.8840	.8844
length_of_kernel	5.229	5.524	6.048	6.350
width_of_kernel	2.867	3.253	3.614	3.812
asymmetry_coefficient	4.8040	2.5904	3.1649	4.1641
length_of_kernelgroove	5.095	5.127	5.921	6.184

Final Cluster Centers
(k=4 normalized)

	Cluster			
	1	2	3	4
area_A_transformed	.11	.50	.80	.27
perimeter_P_transformed	.17	.54	.83	.30
compactness_C_transformed	.33	.64	.71	.66
length_of_kernel_transformed	.19	.49	.77	.25
width_of_kernel_transformed	.14	.55	.81	.37
asymmetry_coefficient_transformed	.55	.32	.35	.27
length_of_kernelgroove_transformed	.30	.48	.79	.21

Final Cluster Centers (k=5)

	Cluster				
	1	2	3	4	5
area_A	16.56	14.69	19.15	12.09	11.98
perimeter_P	15.39	14.47	16.47	13.31	13.29
compactness_C	.8782	.8809	.8871	.8571	.8508
length_of_kernel	5.888	5.572	6.269	5.217	5.241
width_of_kernel	3.481	3.286	3.773	2.901	2.880
asymmetry_coefficient	4.1095	2.4079	3.4604	3.3438	5.6733
length_of_kernelgroove	5.725	5.159	6.127	5.005	5.122

Final Cluster Centers
(k=5 normalized)

	Cluster				
	1	2	3	4	5
area_A_transformed	.81	.10	.20	.57	.39
perimeter_P_transformed	.84	.16	.22	.63	.42
compactness_C_transformed	.72	.28	.61	.62	.67
length_of_kernel_transformed	.78	.19	.18	.57	.37
width_of_kernel_transformed	.82	.11	.30	.61	.47
asymmetry_coefficient_transformed	.35	.52	.44	.40	.22
length_of_kernelgroove_transformed	.80	.31	.21	.62	.30

Final Cluster Centers (k=6)

	Cluster					
	1	2	3	4	5	6
area_A	11.83	14.24	16.41	18.95	12.32	19.58
perimeter_P	13.22	14.26	15.32	16.39	13.42	16.65
compactness_C	.8500	.8793	.8783	.8868	.8580	.8877
length_of_kernel	5.216	5.494	5.864	6.247	5.266	6.316
width_of_kernel	2.844	3.234	3.463	3.745	2.951	3.835
asymmetry_coefficient	4.1684	2.3165	3.8501	2.7235	6.3367	5.0815
length_of_kernelgroove	5.076	5.062	5.690	6.119	5.122	6.144

Final Cluster Centers
(k=6 normalized)

	Cluster					
	1	2	3	4	5	6
area_A_transformed	.10	.39	.20	.85	.76	.57
perimeter_P_transformed	.16	.42	.22	.90	.77	.63
compactness_C_transformed	.28	.67	.61	.63	.82	.60
length_of_kernel_transformed	.19	.37	.18	.86	.67	.58
width_of_kernel_transformed	.11	.47	.30	.81	.82	.60
asymmetry_coefficient_transformed	.52	.23	.44	.40	.30	.41
length_of_kernelgroove_transformed	.31	.30	.21	.86	.71	.64

ii. Number of elements in each cluster

Number of Cases in each Cluster (k=3)

Cluster	1	61.000
	2	77.000
	3	72.000
Valid		210.000
Missing		.000

Number of Cases in each Cluster (k=3 normalized)

Cluster	1	70.000
	2	63.000
	3	77.000
Valid		210.000
Missing		.000

Number of Cases in each Cluster (k=4)

Cluster	1	75.000
	2	67.000
	3	40.000
	4	28.000
Valid		210.000
Missing		.000

Number of Cases in each Cluster (k=4 normalized)

Cluster	1	62.000
	2	46.000
	3	51.000
	4	51.000
Valid		210.000
Missing		.000

Number of Cases in each Cluster (k=5)

Cluster	1	25.000
	2	51.000
	3	48.000
	4	44.000
	5	42.000
Valid		210.000
Missing		.000

Number of Cases in each Cluster (k=5 normalized)

Cluster	1	46.000
	2	50.000
	3	37.000
	4	28.000
	5	49.000
Valid		210.000
Missing		.000

Number of Cases in each Cluster (k=6)

Cluster	1	56.000
	2	54.000
	3	31.000
	4	33.000
	5	21.000
	6	15.000
Valid		210.000
Missing		.000

Number of Cases in each Cluster (k=6 normalized)

Cluster	1	50.000
	2	50.000
	3	37.000
	4	24.000
	5	24.000
	6	25.000
Valid		210.000
Missing		.000

iii. The class distribution within each cluster

Cluster Number of Case * Class Crosstabulation (k=3)

		Class			
		1	2	3	Total
Cluster Number of Case	1	1	60	0	61
	2	9	0	68	77
	3	60	10	2	72
Total		70	70	70	210

- Cluster 1: 60/61 (Class 2) = 0.984 - good
- Cluster 2: 68/77 (Class 3) = 0.883 - okay
- Cluster 3: 60/72 (Class 1) = 0.833 - okay
- Average Score: (60+68+60)/(210)= 0.895
- Weighted Score by Class: (1/3)(60/70) + (1/3(60/70) + (1/3)(68/70) = 0.895

Cluster Number of Case * Class Crosstabulation (k=3 normalized)

		Class			
		1	2	3	Total
Cluster Number of Case	1	58	9	3	70
	2	2	61	0	63
	3	10	0	67	77
Total		70	70	70	210

- Cluster 1: 58/70 (Class 1) = 0.828 - good
- Cluster 2: 61/63 (Class 2) = 0.968 - okay
- Cluster 3: 67/77 (Class 3) = 0.870 - okay
- Average Score: (58+61+67)/(210)= 0.886
- Weighted Score by Class: (1/3)(58/70) + (1/3(61/70) + (1/3)(67/70) = 0.885

Cluster Number of Case * Class Crosstabulation (k=4)

		Class			
		1	2	3	Total
Cluster Number of Case	1	8	0	67	75
	2	58	6	3	67
	3	4	36	0	40
	4	0	28	0	28
Total		70	70	70	210

- Cluster 1: 67/75 (Class 3) = 0.893 - okay
- Cluster 2: 58/67 (Class 1) = 0.866 - okay
- Cluster 3: 36/40 (Class 2) = 0.900 - okay
- Cluster 4: 28/28 (Class 2) = 1.000 - good
- Average Score: (67+58+36+28)/(210) = 0.900
- Weighted Score by Class: (1/3)(58/70) + (1/3(64/70) + (1/3)(67/70) = 0.900

Cluster Number of Case * Class Crosstabulation (k=4 normalized)

		Class			
		1	2	3	Total
Cluster Number of Case	1	1	0	61	62
	2	27	19	0	46
	3	0	51	0	51
	4	42	0	9	51
Total		70	70	70	210

- Cluster 1: 61/62 (Class 3) = 0.984 - good
- Cluster 2: 27/46 (Class 1) = 0.587 - terrible
- Cluster 3: 51/51 (Class 2) = 1.000 - good
- Cluster 4: 42/51 (Class 1) = 0.824 - okay
- Average Score: (61+27+51+42)/(210) = 0.862
- Weighted Score by Class: (1/3)(69/70) + (1/3)(51/70) + (1/3)(61/70) = 0.862

Cluster Number of Case * Class Crosstabulation (k=5)

		Class			
		1	2	3	Total
Cluster Number of Case	1	6	19	0	25
	2	48	3	0	51
	3	0	48	0	48
	4	14	0	30	44
	5	2	0	40	42
Total		70	70	70	210

- Cluster 1: 19/25 (Class 2) = 0.760 - bad
- Cluster 2: 48/51 (Class 1) = 0.941 - good
- Cluster 3: 48/48 (Class 2) = 1.000 - good
- Cluster 4: 30/44 (Class 3) = 0.682 - bad
- Cluster 5: 40/42 (Class 3) = 0.952 - good
- Average Score: (19+48+48+20+40)/(210)= 0.833
- Weighted Score by Class: (1/3)(48/70) + (1/3)(67/70) + (1/3)(70/70) = 0.881

Cluster Number of Case * Class Crosstabulation (k=5 normalized)

		Class			
		1	2	3	Total
Cluster Number of Case	1	0	46	0	46
	2	1	0	49	50
	3	16	0	21	37
	4	6	22	0	28
	5	47	2	0	49
Total		70	70	70	210

- Cluster 1: 46/46 (Class 2) = 1.000 - good
- Cluster 2: 49/50 (Class 3) = 0.980 - good
- Cluster 3: 21/37 (Class 3) = 0.568 - terrible
- Cluster 4: 22/28 (Class 2) = 0.786 - bad
- Cluster 5: 47/49 (Class 1) = 0.959 - good
- Average Score: (46+49+21+22+47)/(210)= 0.881
- Weighted Score by Class: (1/3)(47/70) + (1/3)(68/70) + (1/3)(70/70) = 0.881

Cluster Number of Case * Class Crosstabulation (k=6)

		Class			
		1	2	3	Total
Cluster Number of Case	1	7	0	49	56
	2	52	0	2	54
	3	9	22	0	31
	4	0	33	0	33
	5	2	0	19	21
	6	0	15	0	15
Total		70	70	70	210

- Cluster 1: 49/56 (Class 3) = 0.875 – okay | Cluster 2: 52/54 (Class 1) = 0.963 - good
- Cluster 3: 22/31 (Class 2) = 0.710 - bad
- Cluster 4: 33/33 (Class 2) = 1.000 - good
- Cluster 5: 19/21 (Class 3) = 0.904 - okay
- Cluster 6: 15/15 (Class 2) = 1.000 - good
- Average Score: (49+52+22+33+19+15)/(210) = 0.905
- Weighted Score by Class: (1/3)(52/70) + (1/3(70/70) + (1/3)(68/70) = 0.905

Cluster Number of Case * Class Crosstabulation (k=6 normalized)

		Class			
		1	2	3	Total
Cluster Number of Case	1	1	0	49	50
	2	48	2	0	50
	3	16	0	21	37
	4	0	24	0	24
	5	1	23	0	24
	6	4	21	0	25
Total		70	70	70	210

- Cluster 1: 49/50 (Class 3) = 0.980 - good | Cluster 2: 48/50 (Class 1) = 0.960 - good
- Cluster 3: 21/37 (Class 3) = 0.568 - terrible
- Cluster 4: 24/24 (Class 2) = 1.000 - good
- Cluster 5: 23/24 (Class 2) = 0.958 - good
- Cluster 6: 21/25 (Class 2) = 0.840 - okay
- Average Score: (49+48+21+24+23+21)/(210) = 0.886
- Weighted Score by Class: (1/3)(48/70) + (1/3)((24+23+21)/70) + (1/3)((49+21)/70) = 0.886

iv. In your opinion, which k should be selected? Explain your selection.

- **I believe a k value of 3 is optimal for this dataset to group the data. But I acknowledge that a k-value of 6 performed the best on multiple metrics I tried using to decide the best k-value. The choice of k-value in this scenario depends on how much cluster impurity is acceptable, and how effective the cluster is on average at getting points from the same class into a given cluster. The k=6 value for example, had higher mark in terms of the average score (sum of the correctly grouped kernels/total number of kernels) and weighted score (sum of the fraction of correctly grouped points for each class).**
- **Given that there are three classes, ideally there would be three clusters for a k-value of 3. This isn't necessarily the case, but especially because there are only 210 data points, I think a k-value of 3 will also help prevent overfitting, in case clustering was used on future wheat kernel samples to say create class labels.**
- **In my Non-Normalized Results table, I summarize the results of k-means clustering on the data prior to normalization. Interestingly, the k-means algorithm performed better on the non-normalized data than on the normalized data. This could indicate that some geometric parameters are more important than others in terms of grouping together kernels. As a fix to this, the parameters could receive a weighting relative to their importance to the model. Overall, k=4 and k=6 performed the best in terms of my average score and weighted score metrics, but notably mis grouped many kernels belonging to Class 1.**
- **In my Normalized Results table, I summarize the results of k-means clustering on the data following normalization. The k-values of 3 and 6 performed the best across all metrics for the normalized data. With nearly identical scores. K=6 did very well at grouping class 2 points together and class 3 points together, but mis grouped many class one points in with the other classes. K=3 did better at grouping like-Class 1 points, average at Class 2, and good at Class 3.**
- **Overall, 3 clusters is a simpler model, and it performed about as well at k=6, so I think a k-value of 3 is optimal.**

Non-Normalized Results					
	Average Score	#cases from Class 1 in a Cluster of majority Class 1	#cases from Class 2 in a Cluster of majority Class 2	#cases from Class 3 in a Cluster of majority Class 3	Weighted Score by Class (pseudo-accuracy)
K=3	0.895	60	60	68	0.895
K=4	0.900	58	64	67	0.900
K=5	0.833	48	67	70	0.881
K=6	0.905	52	70	68	0.905

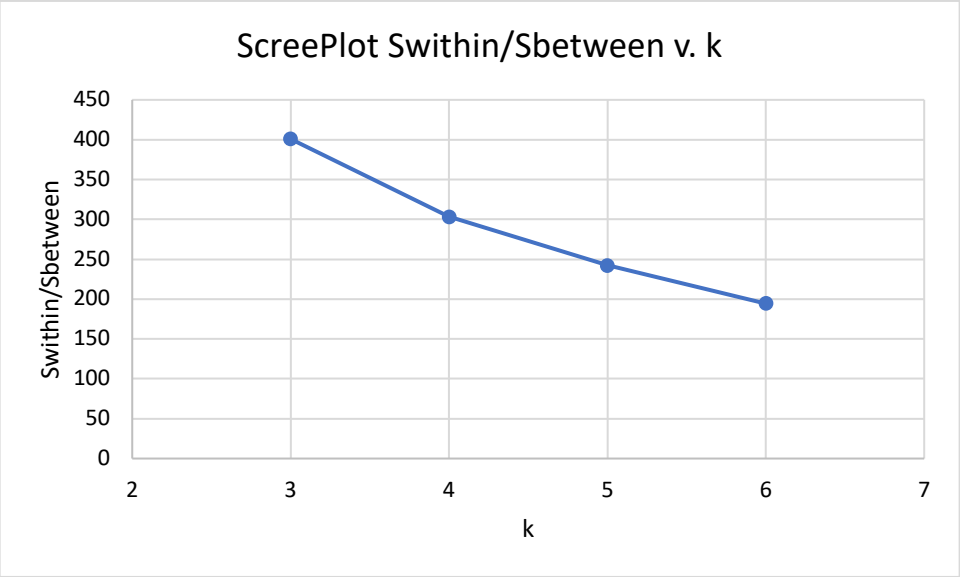
Normalized Results					
	Average Score	#cases from Class 1 in a Cluster of majority Class 1	#cases from Class 2 in a Cluster of majority Class 2	#cases from Class 3 in a Cluster of majority Class 3	Weighted Score by Class (pseudo-accuracy)
K=3	0.886	58	61	67	0.885
K=4	0.862	69	51	61	0.862
K=5	0.881	47	68	70	0.881
K=6	0.886	48	68	70	0.886

- **The table above, my optimal k-value of 3 is highlighted in green. It did well on average across classes, without to many mislabeled points per class.**
- **I also wanted to see if an analysis of the S-within / S-between ratio could help identify the best k value for clustering. I summarize the results of my calculations below. I found that k=6 minimized the S-within to S-between ratio, which would indicate the highest intraclass similarity and lowest intergroup similarity. That said, an increasing k value should always improve the ratio, up until a knee-point which may come around k=6. I performed these calculations on the non-normalized data because that data had the highest scores for my metrics above.**

SSE Calculations in Excel for non-normalized (within groups):

	SSE Cluster 1	SSE Cluster 2	SSE Cluster 3	SSE Cluster 4	SSE Cluster 5	SSE Cluster 6	S_{Within} (Sum(SSE))
K=3	184.1	195.7	207.5				587.3
K=4	187.6	173.9	93.4	61.5			516.4
K=5	36.9	82.3	118.2	64.2	83.9		385.5
K=6	77.3	108.1	53.1	30.2	42.2	25.8	336.7

NON-NORMALIZED	S-Within	S-Between	S-Within / S-Between
K=3	587.3	1.465	400.9
K=4	516.4	1.703	303.2
K=5	385.5	1.591	242.3
K=6	336.7	1.732	194.4



- **The scree plot shows an improving ratio with increasing k values, with a knee-point likely around a k value of 6. Despite this, I still believe a k value of 3 is more appropriate for this dataset. If k was equal to the number of data points in our dataset (210) it would also have a smaller ratio, but that wouldn't do us any good.**

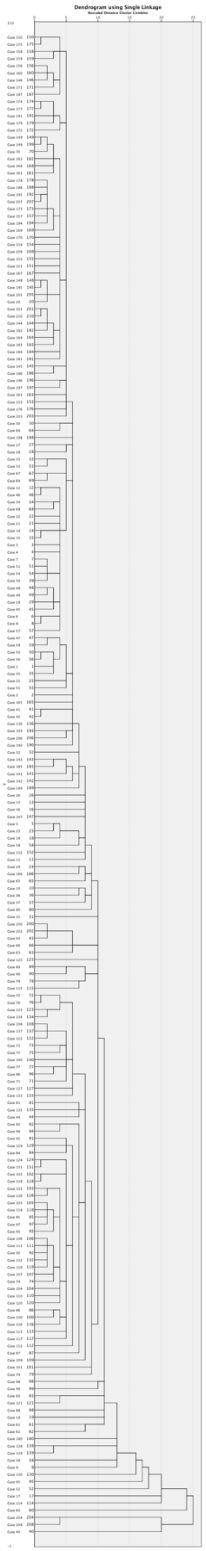
v. For the selected k in iv, analyze and report if the normalization of the attributes will influence the clustering results.

- **I normalized the data using min-max normalization in the range [0,1].**
- **Average score normalized: 0.886**
- **Weighted score normalized: 0.885**
- **Number of mis grouped points per cluster normalized: Total = 24**
  - **Cluster 1: 12**
  - **Cluster 2: 2**
  - **Cluster 3: 10**
- **Average score non-normalized: 0.895**
- **Weighted Score non-normalized: 0.895**
- **Number of mis grouped points per cluster non-normalized: Total = 22**
  - **Cluster 1: 1**
  - **Cluster 2: 9**
  - **Cluster 3: 12**
- **Yes, normalizing the data does influence the results. The k-means algorithm appears to have actually worked better on the non-normalized data in terms of the number of points that were erroneously grouped within each cluster. For the non-normalized data there were only 22 mis grouped points, while for the normalized group there were 24 mis grouped points.**
- **Generally, the normalization of attributes is recommended for the k-means algorithm to work effectively. Given that I used Euclidean Distance as my distance measure, there is also a slight decrease in the efficiency of the algorithm with increasing number of attributes, but it should still generalize better following normalization.**

ii. (10 points) Perform hierarchical clustering using all attributes except the class label as follows:

i. Apply single linkage algorithm and report

1. The dendogram



2. The class distribution at the level of the dendogram where there are only three clusters.

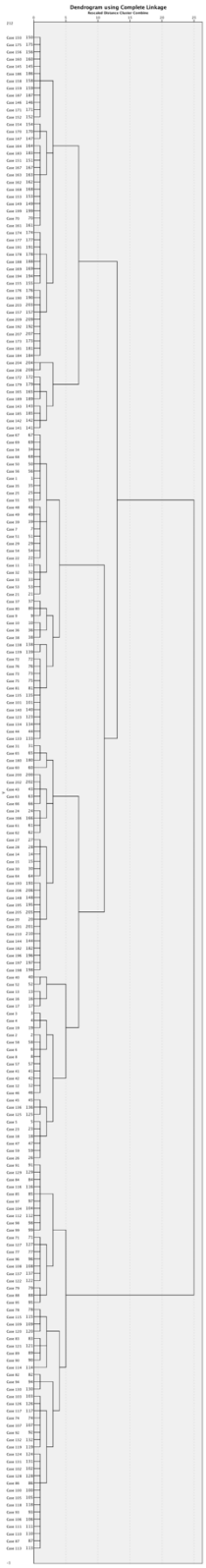
Single Linkage * Class Crosstabulation

Count

		Class			
		1	2	3	Total
Single Linkage	1	68	70	68	206
	2	1	0	2	3
	3	1	0	0	1
Total		70	70	70	210

- **Clearly Single Linkage with 3 clusters messed up terribly and put almost all points in cluster 1.**
- **This method is not suitable for this dataset.**
- **In terms of 'purity':**
  - **Cluster 1: 70/206 = 0.340 - terrible**
  - **Cluster 2: 2/3 = 0.666 - terrible**
  - **Cluster 3: 1/1 = 1.000 - good**
  - **Average: (70+2+1)/210 = 0.348 – Clearly awful grouping**
  - **Weighted Score by Class: (1/3)(1/70) + (1/3)(70/70) + (1/3)(2/70) = 0.348**

ii. Apply complete linkage and report
    1. The dendogram



2. The class distribution at level of the dendogram where there are only three clusters.

Complete Linkage * Class Crosstabulation

Count

		Class			
		1	2	3	Total
Complete Linkage	1	69	16	16	101
	2	1	0	54	55
	3	0	54	0	54
Total		70	70	70	210

- **Complete Linkage with 3 clusters performs much better than Single Linkage did, assigning 69 cases from Class 1 to Cluster 1, 54 cases from Class 2 to Cluster 3, and 54 cases from Class 3 to Cluster 2.**
- **In terms of 'purity':**
  - **Cluster 1: 69/101 = 0.683**
  - **Cluster 2: 54/55 = 0.982**
  - **Cluster 3: 54/54 = 1.000**
  - **Average: 0.843**
  - **Weighted Score by Class: (1/3)(69/70) + (1/3)(54/70) + (1/3)(54/70) = 0.843**

iii. (2.5 points) Compare the results with hierarchical clustering and k-means algorithm.

	K-means Purity (k=3 normalized)	Hierarchical Purity
Cluster 1:	58/70 = 0.828	69/101 = 0.683
Cluster 2:	61/63 = 0.968	54/55 = 0.982
Cluster 3:	67/77 = 0.870	54/54 = 1.00
Average Purity:	0.889	0.888
Average Score:	0.886	0.842

- **K-means clustering yields a higher average purity and average score within clusters than hierarchical clustering does, as displayed in the table above. That said, they are close in value so looking further to see that hierarchical clustering placed 32 points from Classes 2 and 3 to a cluster with majority Class 1 points, demonstrates poor performance. While the dataset only holds 210 cases, I would still argue that k-means does a better job of clustering the data in general.**

iv. (2.5 points) Create an executive summary (~half a page) that outlines the problem, summarizes the data, describes the methodology, summarizes the results, and makes recommendations. When creating it, imagine that you will give this summary to someone who is not an expert in data mining.

Executive Summary

Our goal in this problem was to analyze a dataset containing geometric parameters of wheat kernels from three different species of wheat: Kama, Rosa, and Canadian. More specifically, the dataset contains 70 Kama, 70 Rosa, and 70 Canadian cases, for a total of 210 wheat kernels that had their measurements taken. Each kernel had the following measurements taken: area, perimeter, compactness, length, width, asymmetry coefficient, and groove length.

In this problem, we wanted to determine how well different clustering algorithms, k-means and hierarchical, could group similar wheat kernels together, to see if the measurements provided in the dataset can be used to accurately tell the three different species of wheat apart. If the clustering algorithm does a good job of grouping each species of wheat together based on geometric parameters, then new kernels could be labelled with their corresponding species name based on their measurements. In order to determine the effectiveness of these methods, I reported the relative purity of each cluster for the different parameter settings of each algorithm. Basically, the if a cluster has a lot of points its similar to or of the same species, it's a better cluster and the algorithm is doing a good job of grouping each species based on the wheat kernel measurements.

The first algorithm tested was the k-means algorithm, which iteratively assigns cases to clusters based on their distance, or similarity to, their nearest cluster center. Cluster centers are chosen at random initially but get updated as the average of coordinates for all the points assigned to that cluster in the previous iteration of the algorithm. Basically, points or cases that are close to each other if you were to plot them on a graph get grouped together. With this algorithm, you define the number of clusters "k" you think will be present in the data to best group that data. In my analysis, I found that k=3 or three clusters did the best job of grouping the data in general. Other cluster numbers did well, but I think k=3 will work the best in general, which is important when considering the model being applied to new data.

The second algorithm tested was agglomerative hierarchical clustering, in which data points are also grouped together based on their distance to other data points. This algorithm basically groups the points closest to each other, then groups points with their closest groups of points, then groups with groups, and so on until a specified number of groups/clusters have been formed. In this problem, we had to test this method using the single linkage and complete linkage distance measures to form groups. Single linkage joins a point or group with a point or group that is closest to it, and complete linkage joins a point or group with a point or group that is furthest away but still within that group. Complete linkage worked much better than single linkage did for this problem. The reason is likely that data points were very close to each other, and so complete linkage allowed for better cluster forming or agglomeration on average. Single linkage was more susceptible to groups forming between points that belonged to different species but were close in value.

Overall, k-means provided better cluster purity than hierarchical clustering did on this dataset. Specifically, for non-normalized data, k-means for k=3 yielded an average cluster purity of 0.889 and average score of 0.886, while hierarchical provided an average purity of 0.888 and average score of 0.842 for complete linkage (single linkage was terrible in both metrics).

While the clustering algorithms performed relatively well on this dataset, I would recommend collecting more examples. Only 70 examples from each species is a low number, and more examples could improve the effectiveness of the clustering algorithms as well as any the accuracy provided by classification algorithms later on. For what is available, if clustering must be performed, use k=3 using the k-means algorithm and normalizing the attributes using min-max normalization in the range [0,1].

Problem 2 (25 points):

On the same data used in Problem 1, create a decision tree classification model for the three different varieties of wheat: Kama, Rosa and Canadian.

a. Use 10-fold cross validation and at least five different configurations to produce a decision tree classifier. Report the results obtained for the different configurations and chose one as being the best among the configurations you tried. Explain your answer.

max depth	np	nc	Overall Percent Correct	Complexity	#Nodes	#Terminal Nodes	SPSS Depth	Top 3 Important Features
50	8	4	97.1%	rules=7, depth=5	13	7	5	Area_A, Perimeter_P, width_of_kernal
50	10	5	97.1%	rules=7, depth=5	13	7	5	Area_A, Perimeter_P, width_of_kernal
50	14	7	97.1%	rules=6, depth=5	11	6	5	Area_A, Perimeter_P, width_of_kernal
50	18	9	91.9%	rules=3, depth=2	5	3	2	Area_A, Perimeter_P, width_of_kernal
50	22	11	91.9%	rules=3, depth=2	5	3	2	Area_A, Perimeter_P, width_of_kernal
50	40	20	91.9%	rules=3, depth=2	5	3	2	Area_A, Perimeter_P, width_of_kernal
50	100	50	91.9%	rules=3, depth=2	5	3	2	Area_A, Perimeter_P, width_of_kernal

- **Explanation: The best configuration for the decision tree is a tree with np=14, nc=7, 6 rules, a depth of 5, and Area_A, Perimeter_P, and Width_of_kernel as the top three important features. This configuration yields a 97.1% overall percent correct. Increasing np and nc beyond this point leads to a lower overall percent correct. Reducing np and nc to values lower than 14 and 7 increases the number of terminal nodes or rules from 6 to 7. Therefore, np=14 and nc=7 yielded the best results in terms of overall percentage correct and the simplest model.**

b. For the best tree configuration, report the misclassification matrix and interpret it. In your opinion, is accuracy a good way to interpret the performance of the model? If not, suggest other measures.

Classification

	Predicted			
Observed	1	2	3	Percent Correct
1	68	1	1	97.1%
2	2	68	0	97.1%
3	2	0	68	97.1%
Overall Percentage	34.3%	32.9%	32.9%	97.1%

Growing Method: CRT

Dependent Variable: Class

- **Accuracy, or overall percent correct, is a good way to interpret the performance of the model in this case. It shows how the observed results compared to the predicted and how well the decision tree predicted a wheat kernel's species based on its geometric parameters.**
- **Other metrics for model performance include sensitivity and specificity, but in terms of classifying wheat kernels, there doesn't seem to be a clear reason to optimize for either. For the purpose of this problem/assignment/dataset the overall percent correct metric is sufficient to describe how well this model performed at the wheat kernel classification.**

c. What are the most important three attributes for classifying the wheat data?

- **The top three most important attributes are: Area_A, Perimeter_P, and Width_of_kernel.**
- **This seems to indicate that overall kernel size is an important distinguishing characteristic for telling apart different species of wheat kernel. A larger kernel may suggest one species of wheat over another.**

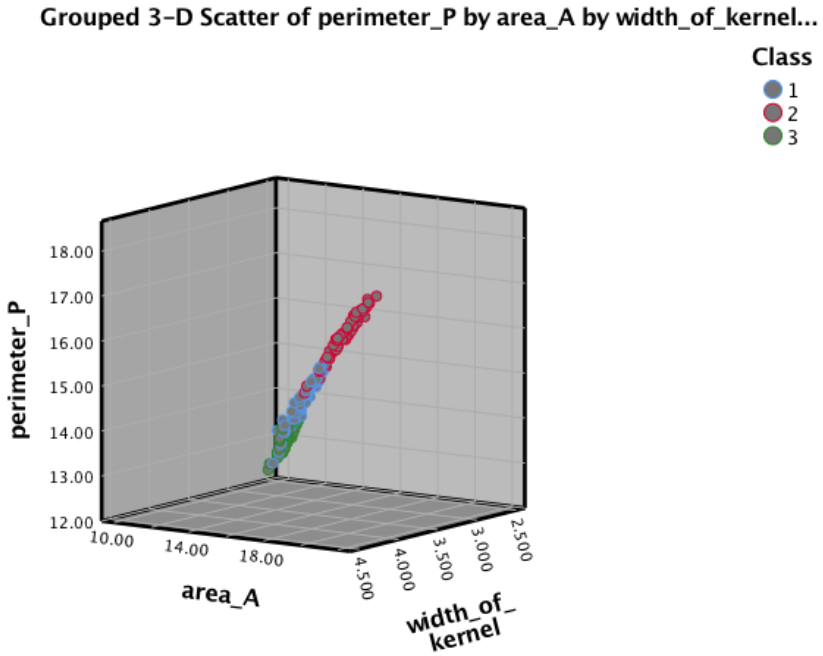
Independent Variable Importance

Independent Variable	Importance	Normalized Importance
area_A	.499	100.0%
perimeter_P	.484	97.0%
width_of_kernel	.426	85.4%

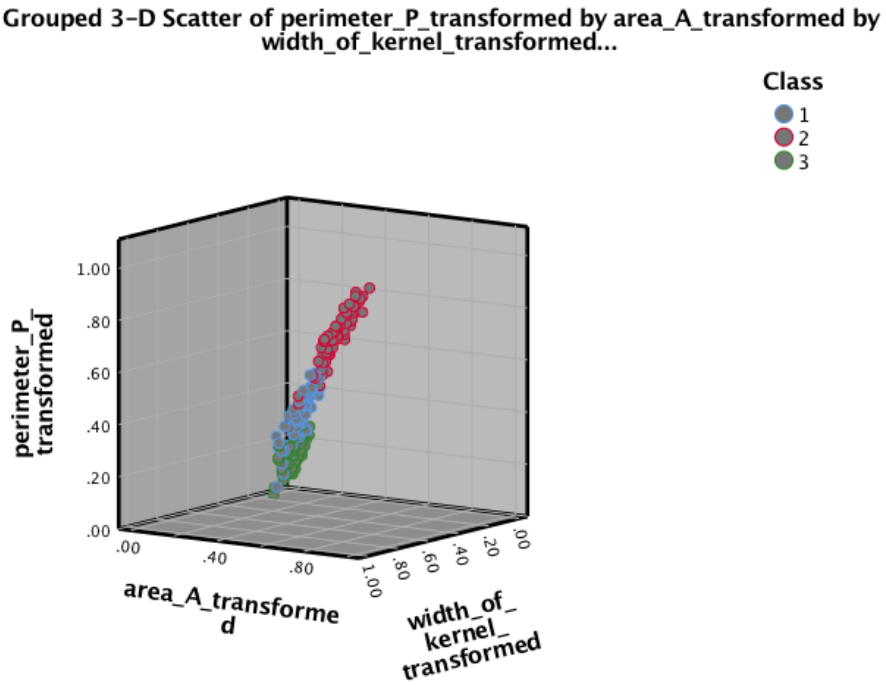
Growing Method: CRT

Dependent Variable: Class

d. Create a graph that will allow you to visualize the data in the 3-dimensional space of the most important attributes. Interpret the graph.



- **Above is a 3D scatter plot of the three most important features within the dataset. Class 1 blue is Kama, Class 2 red is Rosa, and Class 3 green is Canadian. This indicates that Rosa kernels have the largest size on average, then Kama, and Canadian being generally the smallest.**
- **It is also important to note that there are not 3 distinct clusters, which makes sense as wheat kernels of different species share many characteristics. If it was comparing a wheat and rice kernels, maybe there would be more distinct clusters.**



- **I also report here the 3D scatterplot for the normalized data. It performs just as well as the non-normalized data when classifying using a decision tree. The 3D scatterplot shows the variation in the classes more clearly though for the most important features.**

e. Are there any other techniques that can help identify variables for data visualization? Explain your answer and include any analysis you will perform to answer this question.

- **We can perform correlation analysis to identify the most important predictor variables in this dataset given that our variables are continuous. I attempt to perform this analysis using the stepwise method under linear regression analysis. Also, by looking at the correlation among variables, we can determine how best to perform data reduction. Among the most correlated features, we might not need all of them in order to build an efficient model, and they can therefore be reduced.**
- **I report below the descriptive statistics, model summary, ANOVA table, and an example of a scatterplot of some highly correlated variables.**

Descriptive Statistics

	Mean	Std. Deviation	N
Class	2.00	.818	210
area_A	14.8475	2.90970	210
perimeter_P	14.5593	1.30596	210
compactness_C	.870999	.0236294	210
length_of_kernel	5.62853	.443063	210
width_of_kernel	3.25860	.377714	210
asymmetry_coefficient	3.700201	1.5035571	210
length_of_kernelgroove	5.40807	.491480	210

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.577[a]	.333	.330	.670
2	.680[b]	.463	.458	.603
3	.691[c]	.477	.469	.596
4	.840[d]	.705	.699	.449

a. Predictors: (Constant), asymmetry_coefficient
b. Predictors: (Constant), asymmetry_coefficient, compactness_C
c. Predictors: (Constant), asymmetry_coefficient, compactness_C, length_of_kernelgroove
d. Predictors: (Constant), asymmetry_coefficient, compactness_C, length_of_kernelgroove, length_of_kernel

ANOVA[a]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	46.654	1	46.654	103.958	.000[b]
	Residual	93.346	208	.449		
	Total	140.000	209			
2	Regression	64.799	2	32.400	89.184	.000[c]
	Residual	75.201	207	.363		
	Total	140.000	209			
3	Regression	66.790	3	22.263	62.645	.000[d]
	Residual	73.210	206	.355		
	Total	140.000	209			
4	Regression	98.725	4	24.681	122.585	.000[e]
	Residual	41.275	205	.201		
	Total	140.000	209			

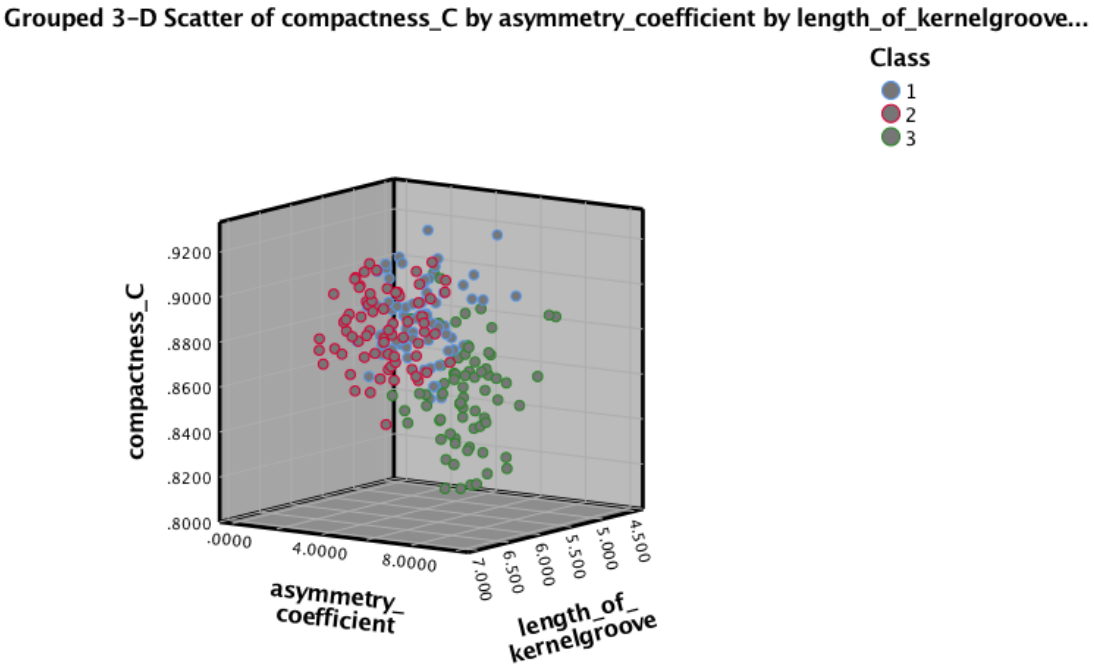
a. Dependent Variable: Class

b. Predictors: (Constant), asymmetry_coefficient

c. Predictors: (Constant), asymmetry_coefficient, compactness_C

d. Predictors: (Constant), asymmetry_coefficient, compactness_C, length_of_kernelgroove

e. Predictors: (Constant), asymmetry_coefficient, compactness_C, length_of_kernelgroove, length_of_kernel



- **As you can see here, these variables have a higher degree of correlation and overlap than the three most important features that I previously identified for building my decision tree. It is better to visualize less correlated features because they can be more easily separated by the rules of the decision tree leading to better classification.**