1.a

|  |  |  |
| --- | --- | --- |
| **Model Summary** | | |
| Specifications | Growing Method | CRT |
| Dependent Variable | V1 |
| Independent Variables | V2, V3, V4, V5, V6, V7, V8, V9, V10, V11, V12, V13, V14, V15, V16, V17 |
| Validation | Split Sample |
| Maximum Tree Depth | 5 |
| Minimum Cases in Parent Node | 100 |
| Minimum Cases in Child Node | 50 |
| Results | Independent Variables Included | V12, V8, V11, V7, V2, V4, V10, V14, V13, V9, V15, V16, V17, V6, V3, V5 |
| Number of Nodes | 35 |
| Number of Terminal Nodes | 18 |
| Depth | 5 |

|  |  |  |
| --- | --- | --- |
| **Risk** | | |
| Sample | Estimate | Std. Error |
| Training | .632 | .004 |
| Test | .631 | .006 |
| Growing Method: CRT  Dependent Variable: V1 | | |

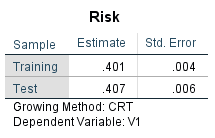
1.b

Model 1:

Maximum Tree Depth: 15

Min Cases in Parent node: 200

Min cases in Child node: 100



Model 2:

Maximum Tree Depth: 4

Min Cases in Parent node: 300

Min cases in Child node: 100

|  |  |  |
| --- | --- | --- |
| **Risk** | | |
| Sample | Estimate | Std. Error |
| Training | .746 | .004 |
| Test | .752 | .006 |
| Growing Method: CRT  Dependent Variable: V1 | | |

Model 3:

Maximum Tree Depth: 6

Min Cases in Parent node: 300

Min cases in Child node: 100

|  |  |  |
| --- | --- | --- |
| **Risk** | | |
| Sample | Estimate | Std. Error |
| Training | .557 | .004 |
| Test | .565 | .006 |
| Growing Method: CRT  Dependent Variable: V1 | | |

Model 4:

Maximum Tree Depth: 3

Min Cases in Parent node: 200

Min cases in Child node: 100

|  |  |  |
| --- | --- | --- |
| **Risk** | | |
| Sample | Estimate | Std. Error |
| Training | .819 | .003 |
| Test | .825 | .005 |
| Growing Method: CRT  Dependent Variable: V1 | | |

Model 5:

Maximum Tree Depth: 3

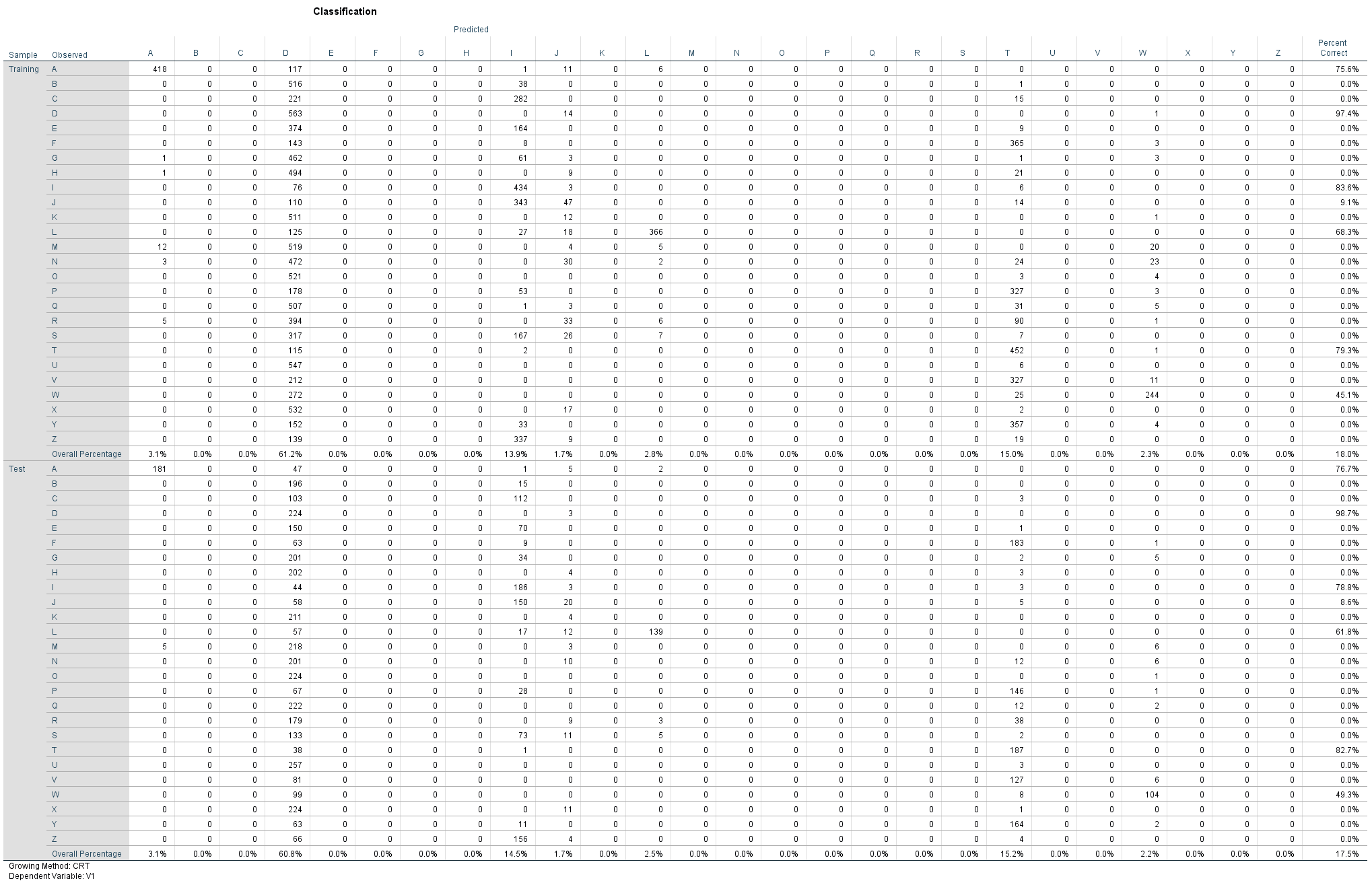
Min Cases in Parent node: 400

Min cases in Child node: 200

|  |  |  |
| --- | --- | --- |
| **Risk** | | |
| Sample | Estimate | Std. Error |
| Training | .820 | .003 |
| Test | .825 | .005 |
| Growing Method: CRT  Dependent Variable: V1 | | |

Model 5 has the best accuracy for both the training and test set. This is also the simplest model because it has a max depth of 3. I would choose model 5.

1.c.



This decision tree is only predicting a few different letters (D, T, I), so I do not think it is a very good model. I don’t think looking at just the accuracy of the model is always a good representation of the model. Looking at the sensitivity and specificity could add more insight to how the model performs.

1.d

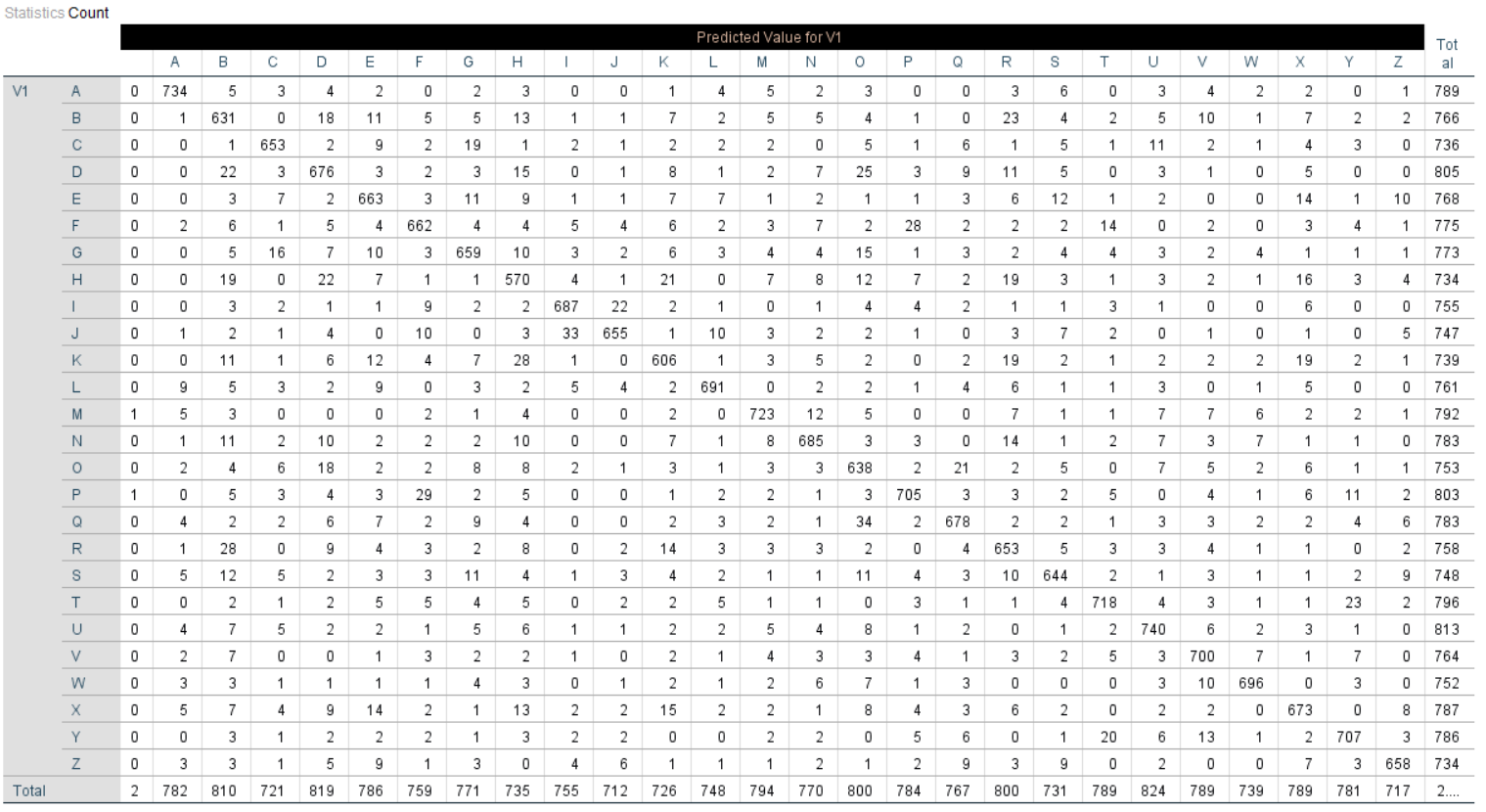
The three most important attributes for recognizing letters are x-ege mean edge count left to right (integer), y-bar mean y of on pixels in box (integer), and xybar mean x y correlation (integer)

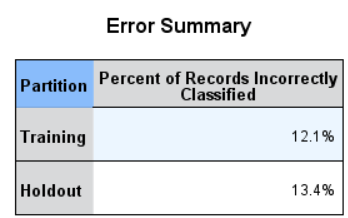
2.a.

I am not transforming the data, but I made sure to check the “Normalize features” box to ensure that all variables are normalized before the KNN classification is performed.

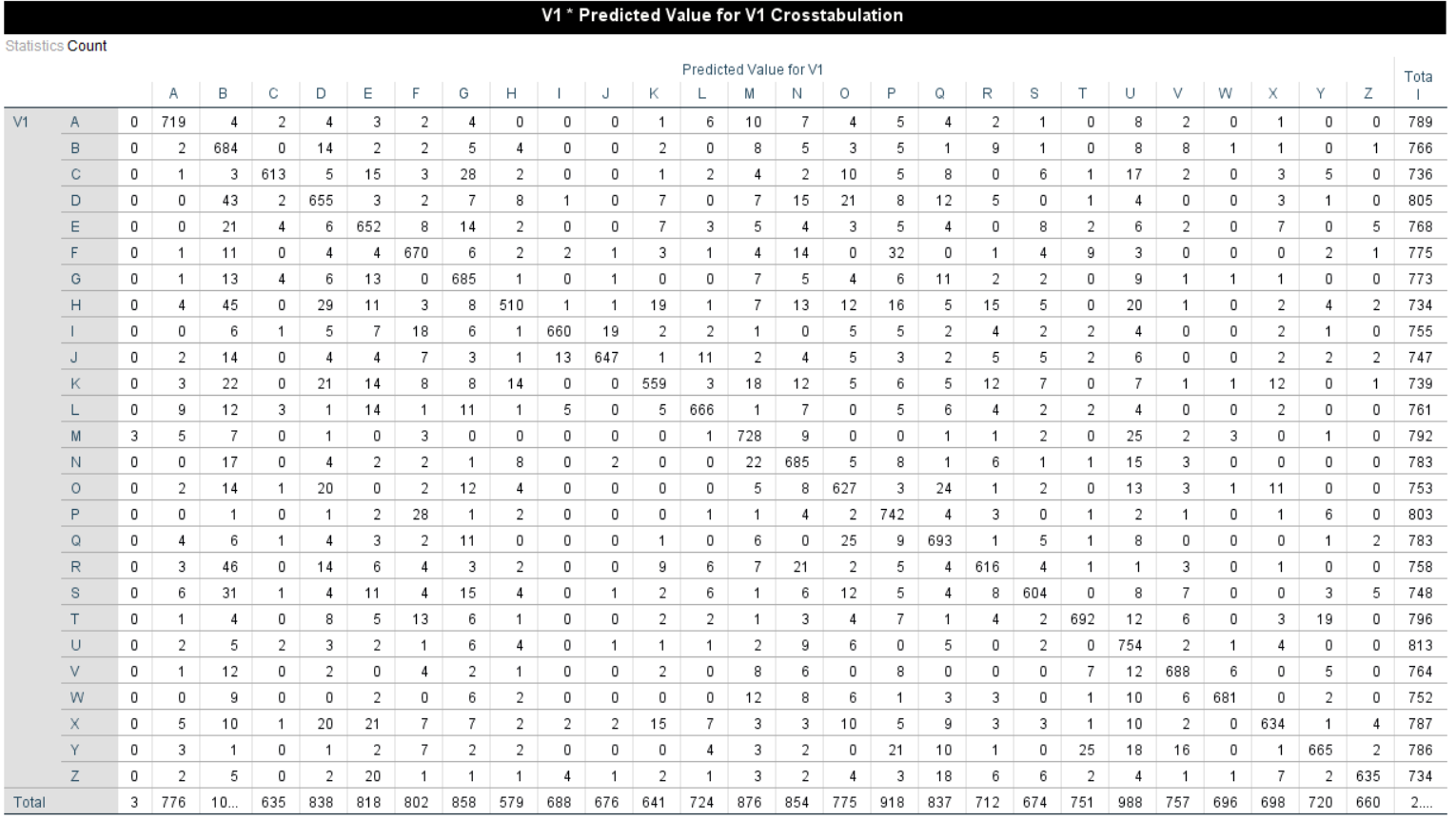
2.b

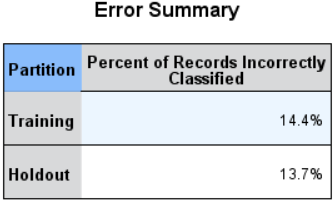
K=1



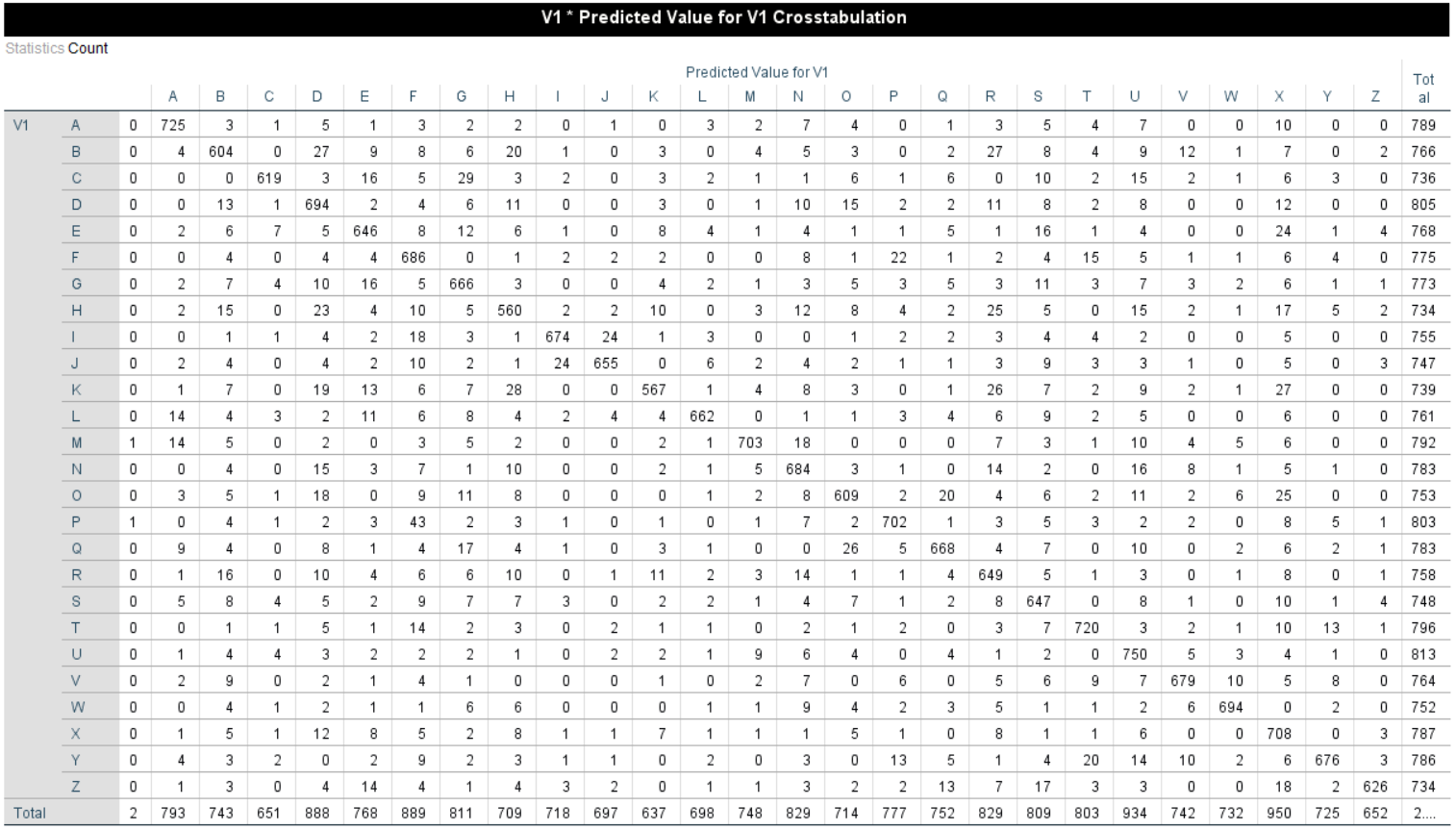


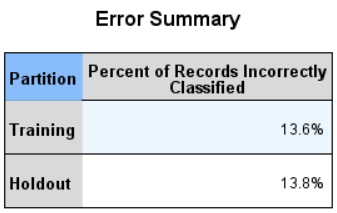
K=3



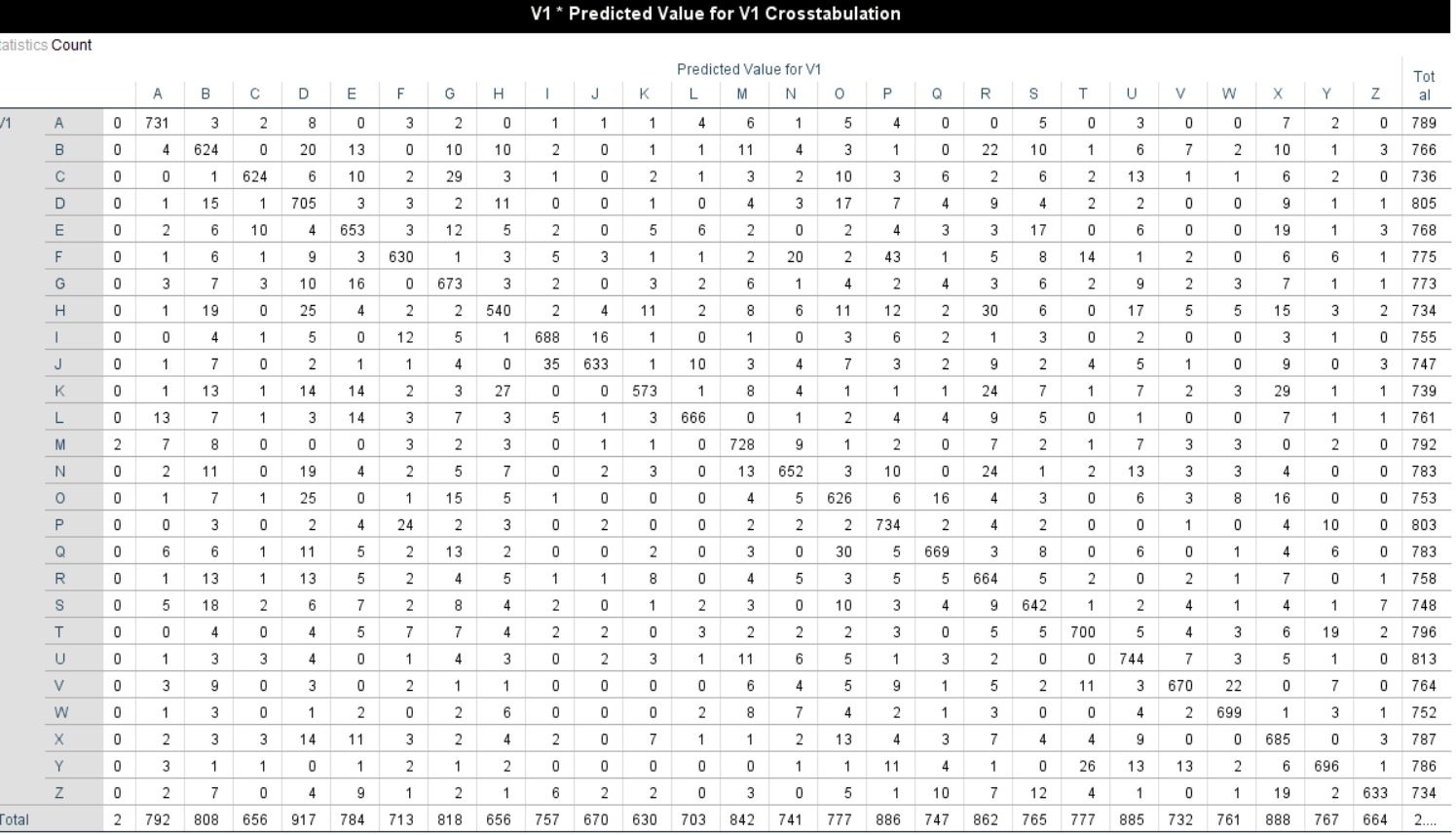


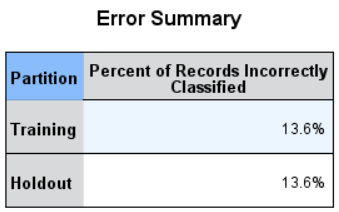
K=5





K=7





2.c

K=1 had the best training and holdout error %. The KNN is a better classifier because as shown in the classification matrices, every letter was predicted at some point. In the decision tree, only a few letters (D, T, I) were predicted for most of the data.

3.a.1

The very first cluster center is arbitrarily chosen.

The cluster centers then are calculated by taking the mean of all the points in the cluster. The centroid is not necessarily an object in the data set. This becomes the new cluster centroid, and new clusters are formed based on their distance from this new centroid.

3.a.2

Euclidian Distance > used to compare two numbers, interval-scaled numeric variables

Cosine Distance > used to compare two documents. You can create a term frequency for each document which creates a vector.

3.a.3

K =3

|  |  |  |  |
| --- | --- | --- | --- |
| **Final Cluster Centers** | | | |
|  | Cluster | | |
| 1 | 2 | 3 |
| Zscore(area) | -.23019 | -1.04171 | 1.21394 |
| Zscore(perimeter) | -.26715 | -1.00856 | 1.22275 |
| Zscore(compactness) | .41615 | -1.05540 | .53414 |
| Zscore(length) | -.36266 | -.88234 | 1.20788 |
| Zscore(width) | -.07273 | -1.12321 | 1.12090 |
| Zscore(coeef) | -.64447 | .81089 | -.06246 |
| Zscore(groove) | -.67365 | -.58237 | 1.26254 |

|  |  |  |
| --- | --- | --- |
| **Number of Cases in each Cluster** | | |
| Cluster | 1 | 75.000 |
| 2 | 65.000 |
| 3 | 70.000 |
| Valid | | 210.000 |
| Missing | | .000 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **class \* Cluster Number of Case Crosstabulation** | | | | | |
| Count | | | | | |
|  | | Cluster Number of Case | | | Total |
| 1 | 2 | 3 |
| class | 1.000 | 64 | 3 | 3 | 70 |
| 2.000 | 3 | 0 | 67 | 70 |
| 3.000 | 8 | 62 | 0 | 70 |
| Total | | 75 | 65 | 70 | 210 |

\

K=4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Final Cluster Centers** | | | | |
|  | Cluster | | | |
| 1 | 2 | 3 | 4 |
| Zscore(area) | -1.10369 | -.16037 | -.65885 | 1.23794 |
| Zscore(perimeter) | -1.05133 | -.18803 | -.72579 | 1.24428 |
| Zscore(compactness) | -1.29008 | .41688 | .20420 | .55066 |
| Zscore(length) | -.90083 | -.27181 | -.78385 | 1.22525 |
| Zscore(width) | -1.23259 | -.02187 | -.46945 | 1.14535 |
| Zscore(coeef) | .54233 | -.79480 | 1.26103 | -.03700 |
| Zscore(groove) | -.59702 | -.60326 | -.69087 | 1.28185 |

|  |  |  |
| --- | --- | --- |
| **Number of Cases in each Cluster** | | |
| Cluster | 1 | 54.000 |
| 2 | 67.000 |
| 3 | 21.000 |
| 4 | 68.000 |
| Valid | | 210.000 |
| Missing | | .000 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **class \* Cluster Number of Case Crosstabulation** | | | | | | |
| Count | | | | | | |
|  | | Cluster Number of Case | | | | Total |
| 1 | 2 | 3 | 4 |
| class | 1.000 | 1 | 0 | 31 | 38 | 70 |
| 2.000 | 0 | 57 | 13 | 0 | 70 |
| 3.000 | 62 | 0 | 0 | 8 | 70 |
| Total | | 63 | 57 | 44 | 46 | 210 |

K=5

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Final Cluster Centers** | | | | | |
|  | Cluster | | | | |
| 1 | 2 | 3 | 4 | 5 |
| Zscore(area) | .53414 | -1.09646 | -.86944 | -.17860 | 1.47902 |
| Zscore(perimeter) | .60045 | -1.00893 | -.94413 | -.21768 | 1.46244 |
| Zscore(compactness) | .16876 | -1.52901 | -.02569 | .49173 | .68097 |
| Zscore(length) | .60095 | -.81588 | -.98159 | -.32456 | 1.44521 |
| Zscore(width) | .49704 | -1.28501 | -.71733 | -.01247 | 1.36170 |
| Zscore(asymm) | .15956 | .54963 | .85530 | -.89272 | -.15948 |
| Zscore(groove) | .62723 | -.51346 | -.82523 | -.67820 | 1.46329 |

|  |  |  |
| --- | --- | --- |
| **Number of Cases in each Cluster** | | |
| Cluster | 1 | 29.000 |
| 2 | 42.000 |
| 3 | 35.000 |
| 4 | 56.000 |
| 5 | 48.000 |
| Valid | | 210.000 |
| Missing | | .000 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **class \* Cluster Number of Case Crosstabulation** | | | | | | | |
| Count | | | | | | | |
|  | | Cluster Number of Case | | | | | Total |
| 1 | 2 | 3 | 4 | 5 |
| class | 1.000 | 9 | 2 | 8 | 51 | 0 | 70 |
| 2.000 | 20 | 0 | 0 | 2 | 48 | 70 |
| 3.000 | 0 | 40 | 27 | 3 | 0 | 70 |
| Total | | 29 | 42 | 35 | 56 | 48 | 210 |

K=6

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Final Cluster Centers** | | | | | | |
|  | Cluster | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 |
| Zscore(area) | -1.11319 | .07930 | 1.49287 | -.86636 | .67858 | -.57775 |
| Zscore(perimeter) | -1.02748 | .07980 | 1.46512 | -.90990 | .76627 | -.67452 |
| Zscore(compactness) | -1.54614 | .41258 | .74367 | -.23961 | .06236 | .51486 |
| Zscore(length) | -.82305 | .01979 | 1.43631 | -.90897 | .79606 | -.83739 |
| Zscore(width) | -1.31626 | .19253 | 1.39310 | -.75508 | .57735 | -.34962 |
| Zscore(asymm) | .50750 | -.77172 | -.23120 | 1.19794 | .30807 | -.71482 |
| Zscore(groove) | -.48226 | -.35972 | 1.45740 | -.70048 | .93320 | -1.11941 |

|  |  |  |
| --- | --- | --- |
| **Number of Cases in each Cluster** | | |
| Cluster | 1 | 39.000 |
| 2 | 40.000 |
| 3 | 45.000 |
| 4 | 30.000 |
| 5 | 25.000 |
| 6 | 31.000 |
| Valid | | 210.000 |
| Missing | | .000 |

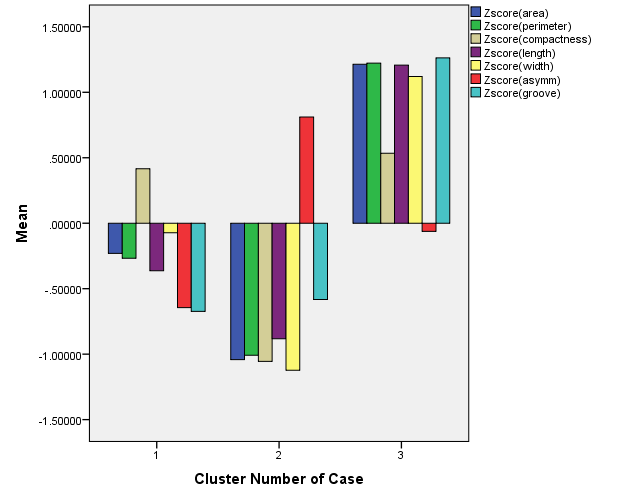
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **class \* Cluster Number of Case Crosstabulation** | | | | | | | | |
| Count | | | | | | | | |
|  | | Cluster Number of Case | | | | | | Total |
| 1 | 2 | 3 | 4 | 5 | 6 |
| class | 1.000 | 1 | 38 | 0 | 3 | 2 | 26 | 70 |
| 2.000 | 0 | 2 | 45 | 0 | 23 | 0 | 70 |
| 3.000 | 38 | 0 | 0 | 27 | 0 | 5 | 70 |
| Total | | 39 | 40 | 45 | 30 | 25 | 31 | 210 |

3.a.4

I would choose k=5, because that is where it looks like the “elbow” is.

3.a.5

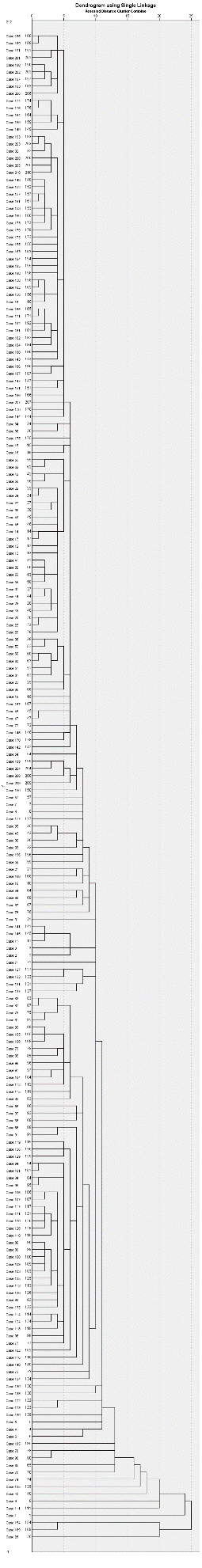
K=3 is a good number of clusters because you can easily tell that Cluster 3 has high values for all variables except asymmetry, cluster 2 has the lowest values for all variables except asymmetry and cluster 1 is somewhere in the middle.



3.a.6

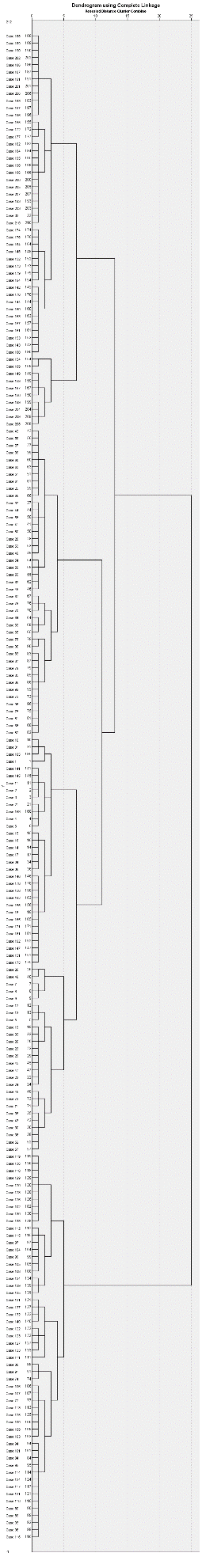
I used the z-score values for the cluster analysis, which is a way to normalize the data. This helps with the comparison and clustering since the clustering is performed based on distance. You need to have normalized variables when using k-means.

3.b.1



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **class \* Single Linkage Crosstabulation** | | | | | |
| Count | | | | | |
|  | | Single Linkage | | | Total |
| 1 | 2 | 3 |
| class | 1.000 | 1 | 68 | 1 | 70 |
| 2.000 | 0 | 70 | 0 | 70 |
| 3.000 | 0 | 68 | 2 | 70 |
| Total | | 1 | 206 | 3 | 210 |

3.b.2



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **class \* Complete Linkage Crosstabulation** | | | | | |
| Count | | | | | |
|  | | Complete Linkage | | | Total |
| 1 | 2 | 3 |
| class | 1.000 | 69 | 1 | 0 | 70 |
| 2.000 | 16 | 0 | 54 | 70 |
| 3.000 | 16 | 54 | 0 | 70 |
| Total | | 101 | 55 | 54 | 210 |

3.c

The first step to my analysis was to cluster the data using the k-means cluster method. This means that each data point is clustered into a group of other similar data points. Each cluster tries to maximize similarity while minimizing dissimilarity. The k-means method uses a calculation of distance so it is important that all variables are normalized. To normalize the variables, I calculated the z-score for each variable and used this in the clustering algorithm. In order to visualize the data, I created a bar chart which displays each of the clusters mean variable values. It can be seen in the graph, where there are 2 clusters, that Cluster 3 is easily distinguishable from the other two clusters because it has on average higher mean values for each variable with the exception of asymmetry of coefficient.

The second method I used to cluster the data was hierarchical clustering. SPSS uses agglomerative clustering which is when the clustering starts with n single clusters and forms a sequence by merging clusters. In this analysis I used minimum distance and maximum distance linkage to create two sets of clusters. The first, minimum distance (or single linkage) calculates the distance between a pair of records, one from each cluster, that are closest to each other. The second, maximum distance (or complete linkage) calculates the distance between a pair of records that are farthest from each other. The results of the complete linkage split the clusters in a way that makes the most sense based on the previously defined class. Cluster 1 contains most tuples of class 1, cluster 2 contains most tuples of class 3 and cluster 3 contains only tuples in class 2.

I think it was hard to interpret these clusters since I am not familiar with the seven geometric parameters that are included in this data set. If there are 3 classes already classified, then it would make sense that these 3 classes are put into separate clusters. In the future we could use the k-means or hierarchical clustering analysis to assign new classes if needed.