2. Describe the goal of all clustering methods.

The clustering task does not try to classify, estimate, or predict the value of a target variable. Instead, clustering algorithms seek to segment the entire data set into relatively homogeneous subgroups or clusters, where the similarity of the records within the cluster is maximized, and the similarity to records outside this cluster is minimized.

6. Suppose that we have the following data:

A B c d e f g h I j

(2,0) (1,2) (2,2) (3,2) (2,3) (3,3) (2,4) (3,4) (4,4) (3,5)

Identify the cluster by applying the k-means algorithm, with k=2. Try using initial cluster centers as far apart as possible.

|  |  |  |
| --- | --- | --- |
| point | x | y |
| a | 2 | 0 |
| b | 1 | 2 |
| c | 2 | 2 |
| d | 3 | 2 |
| e | 2 | 3 |
| f | 3 | 3 |
| g | 2 | 4 |
| h | 3 | 4 |
| i | 4 | 4 |
| j | 3 | 5 |

Step 1: We have already known that we are interested in k = 2 clusters.

Step 2: Randomly assign k (2) records to be the initial cluster center locations. For this example, I assigned the cluster centers to be m1 = (1,2) and m2 = (4,4).

Step 3 (first pass): For each record, find the nearest cluster center. Table below contains the (rounded) Euclidean distances between each point and each cluster center m1 and m2, along with an indication of which cluster center the point is nearest to. Therefore, cluster 1 contains points {a, b, c, d, e}, and cluster 2 contains points {f, g, h, I, j}.

|  |  |  |  |
| --- | --- | --- | --- |
| point | Distance from m1 | Distance from m2 | Cluster membership |
| A | 2.24 | 4.47 | C1 |
| B | 0 | 3.6 | C1 |
| C | 1 | 2.83 | C1 |
| D | 2 | 2.24 | C1 |
| E | 1.41 | 2.24 | C1 |
| F | 2.24 | 1.41 | C2 |
| G | 2.24 | 2 | C2 |
| H | 2.83 | 1 | C2 |
| I | 3.6 | 0 | C2 |
| J | 3.6 | 1.41 | C2 |



Step 4 (first pass): For each of the k clusters find the cluster centroid and update the location of each cluster center to the new value of the centroid. The centroid for cluster 1 is [(2+1+2+3+2)/5, (0+2+2+2+3)/5] = (2,1.8). The centroid for cluster 2 is [(3+2+3+4+3)/5, (3+4+4+4+5)/5] = (3,4).



Step 5: Repeat steps 3 and 4 until convergence or termination. The centroids have moved, so we go back to step 3 for our second pass through the algorithm.

Step 3 (second pass): For each record, find the nearest cluster center. Table below shows the distances between each point and each updated cluster center m1 = (2,1.8) and m2 = (3,4), together with the resulting cluster membership. There hasn’t been any shift from cluster 2 to cluster 1 or vice versa. All records remain in the same clusters as previously.

|  |  |  |  |
| --- | --- | --- | --- |
| point | Distance from m1 | Distance from m2 | Cluster membership |
| A | 1.8 | 4.12 | C1 |
| B | 1.02 | 1.83 | C1 |
| C | 0.2 | 2.24 | C1 |
| D | 1.02 | 2 | C1 |
| E | 1.2 | 2.41 | C1 |
| F | 1.56 | 1 | C2 |
| G | 2.2 | 1 | C2 |
| H | 2.42 | 0 | C2 |
| I | 2.97 | 1 | C2 |
| J | 3.35 | 1 | C2 |

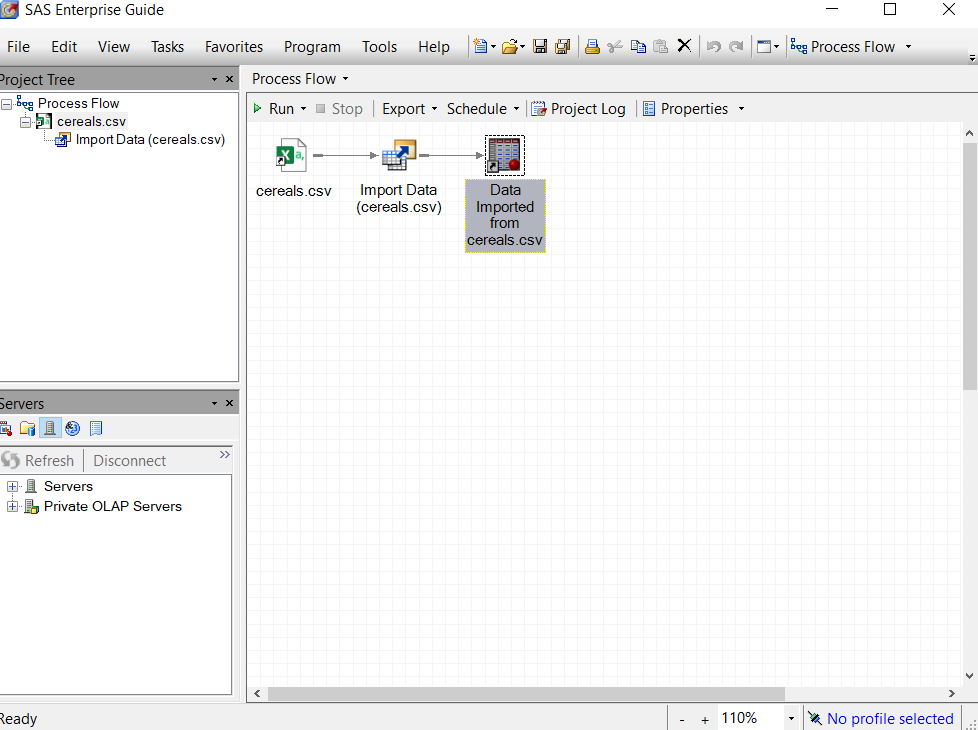


Step 4 (second pass): For each of the k clusters, I found the cluster centroid and update the location of each cluster center to the new value of the centroid. The new centroid for cluster 1 is [(2+1+2+3+2)/5, (0+2+2+2+3)/5] = (2,1.8). The centroid for cluster 2 is [(3+2+3+4+3)/5, (3+4+4+4+5)/5] = (3,4)

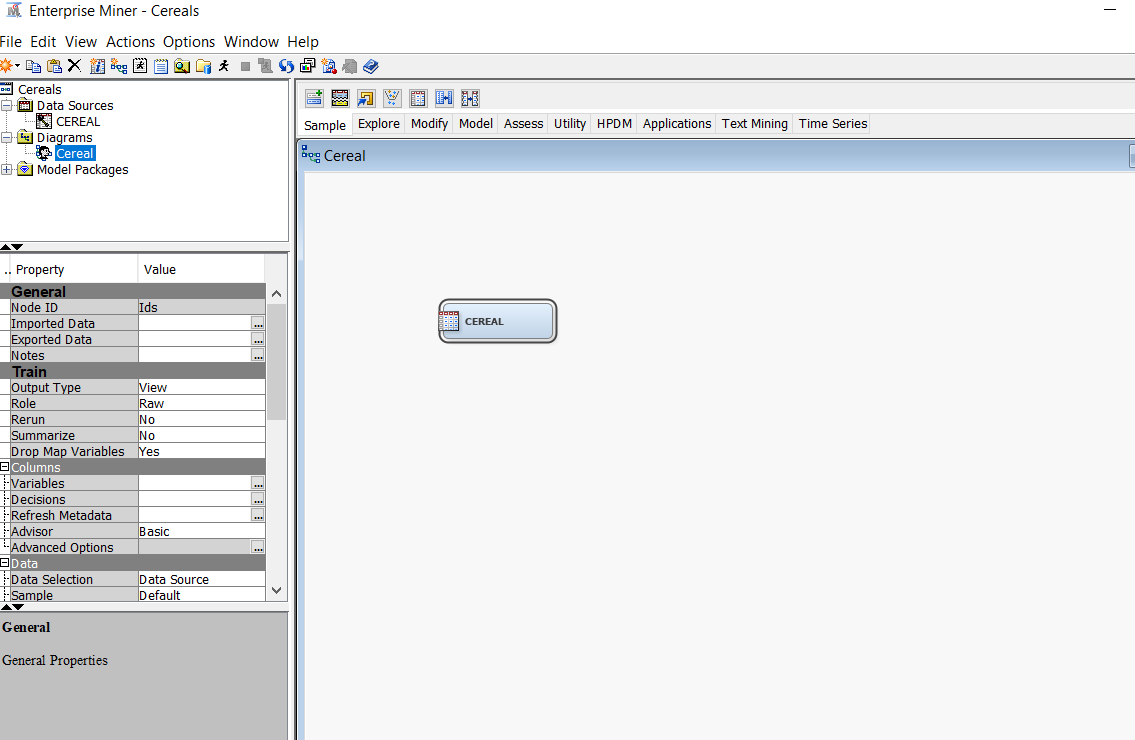
Since the centroids remain unchanged, the algorithm terminates.

12. Using all of the variables except name and rating, run the k-means algorithm with k = 5 to identify clusters within the data.

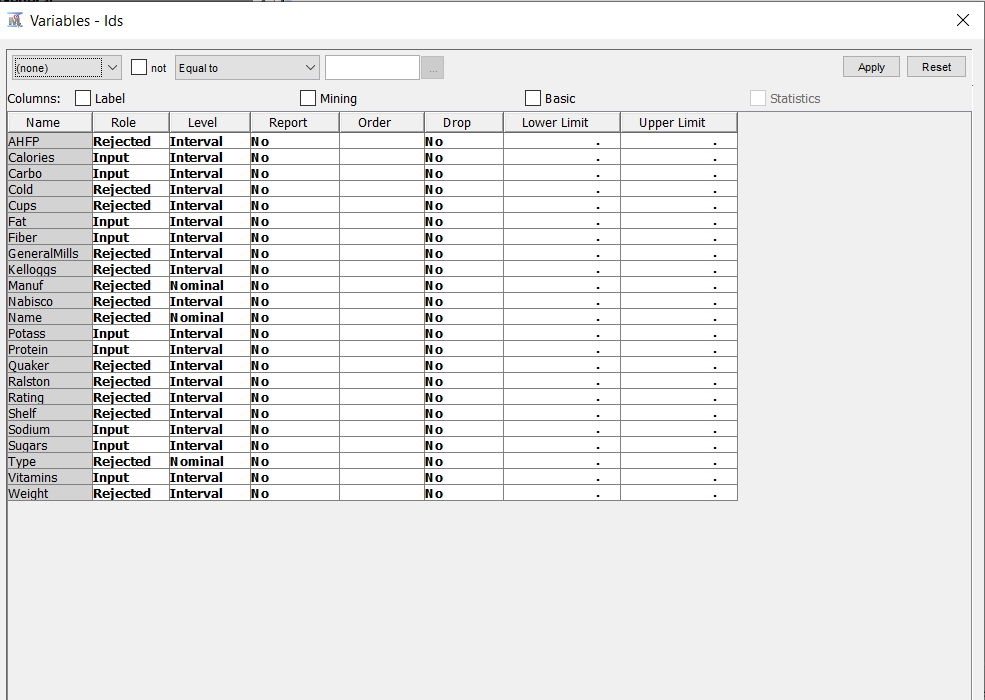
Same all other project, first use SaS EG to convert the data to sas7bdat format.



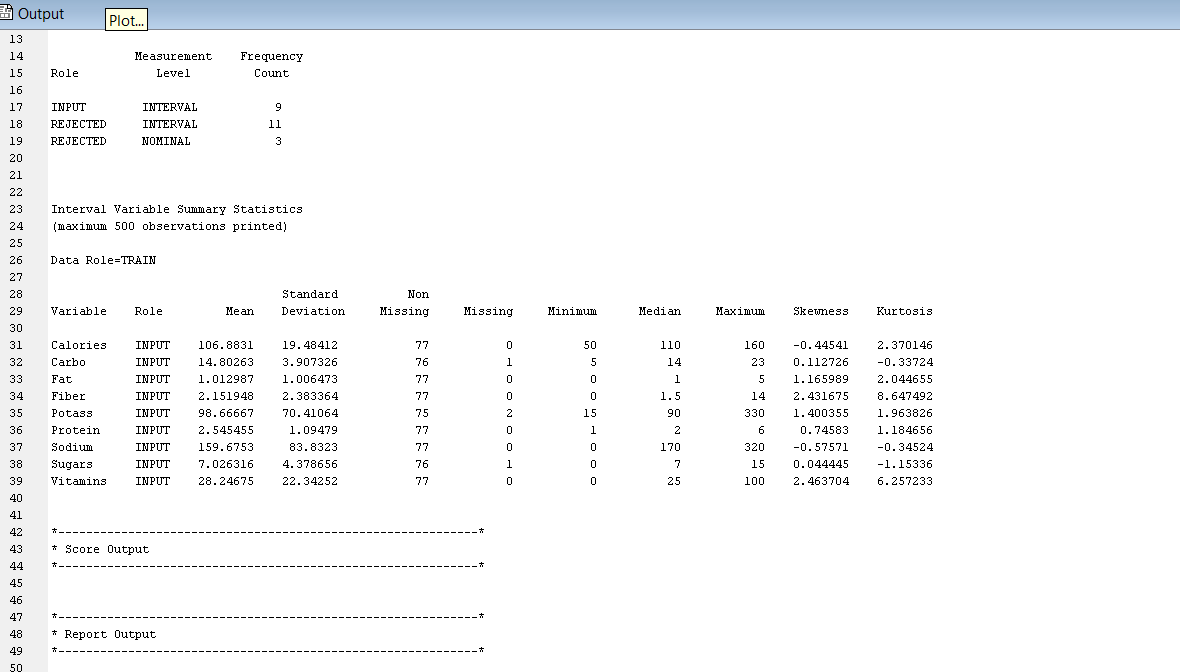
Then open up SaS EM and import the data:



Except for the variables that have been mentioned in the question, I rejected others.

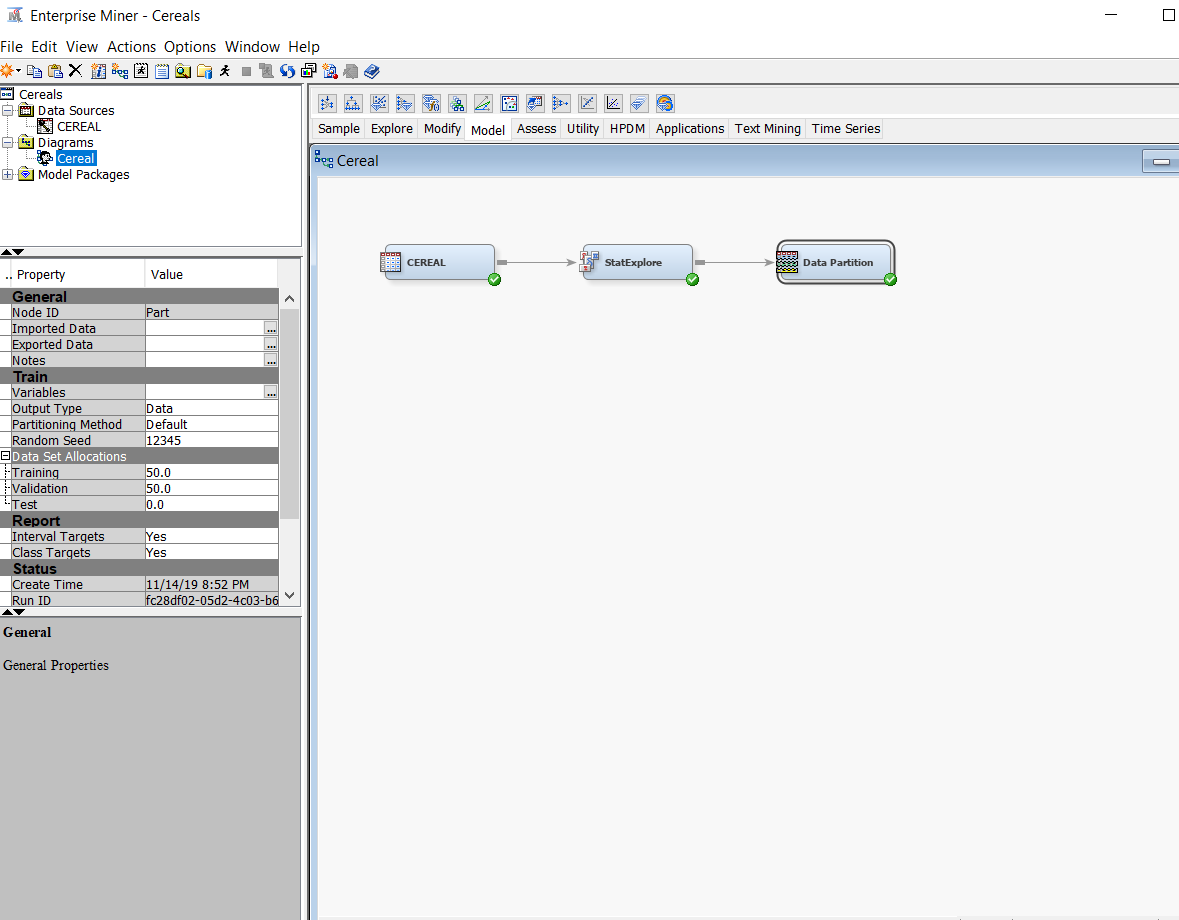
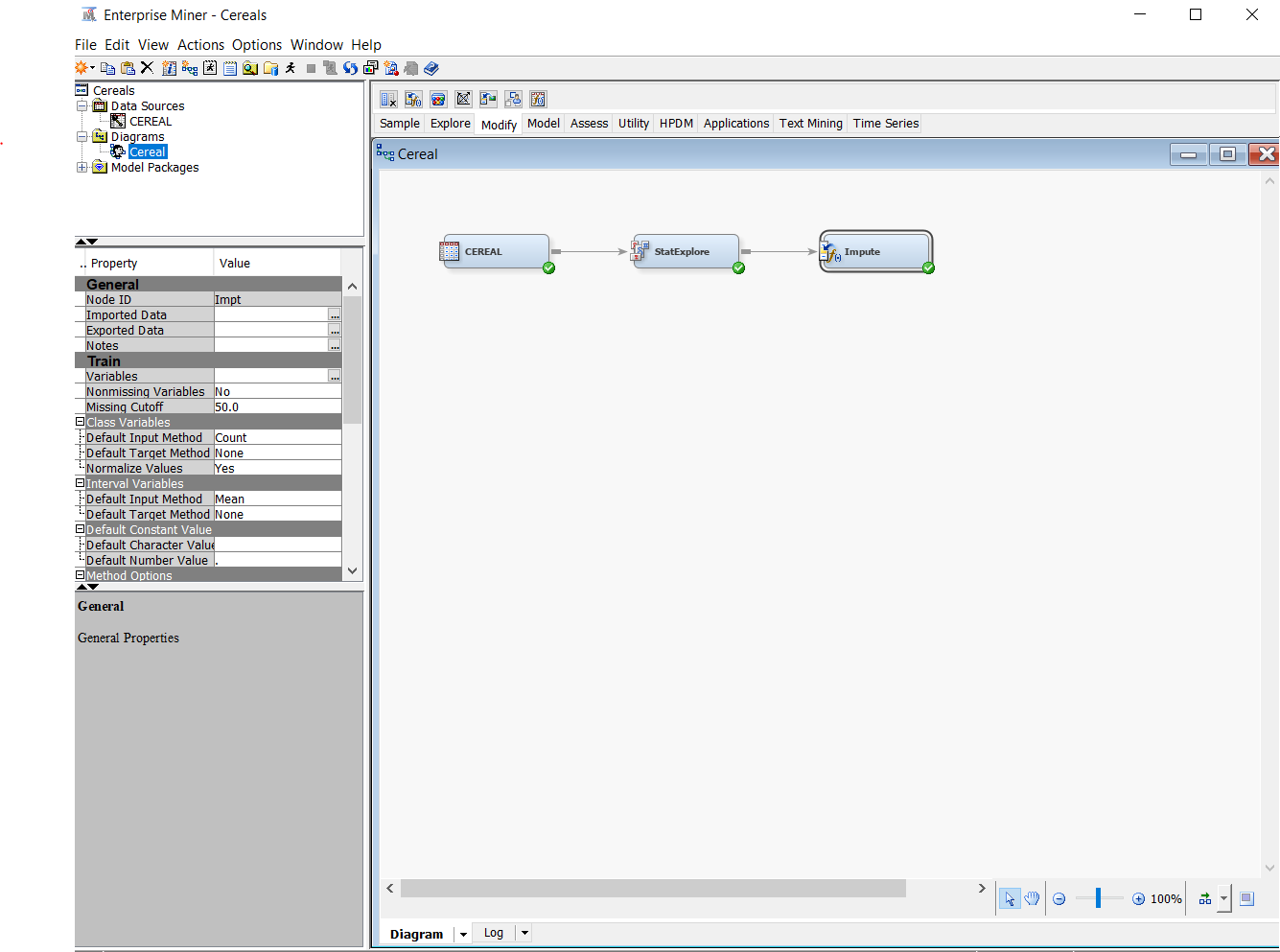


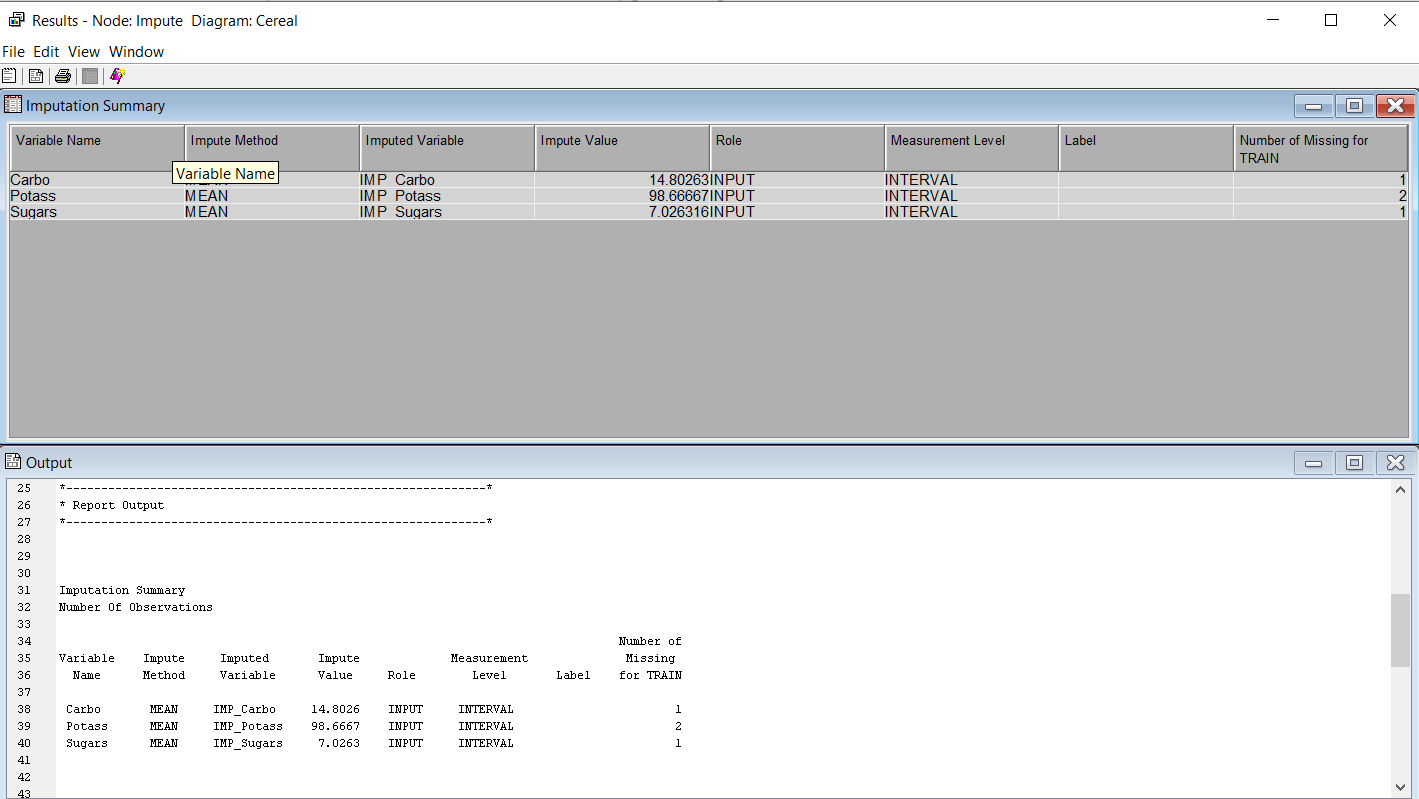
Use statexplore to see if we have any missing values or not:



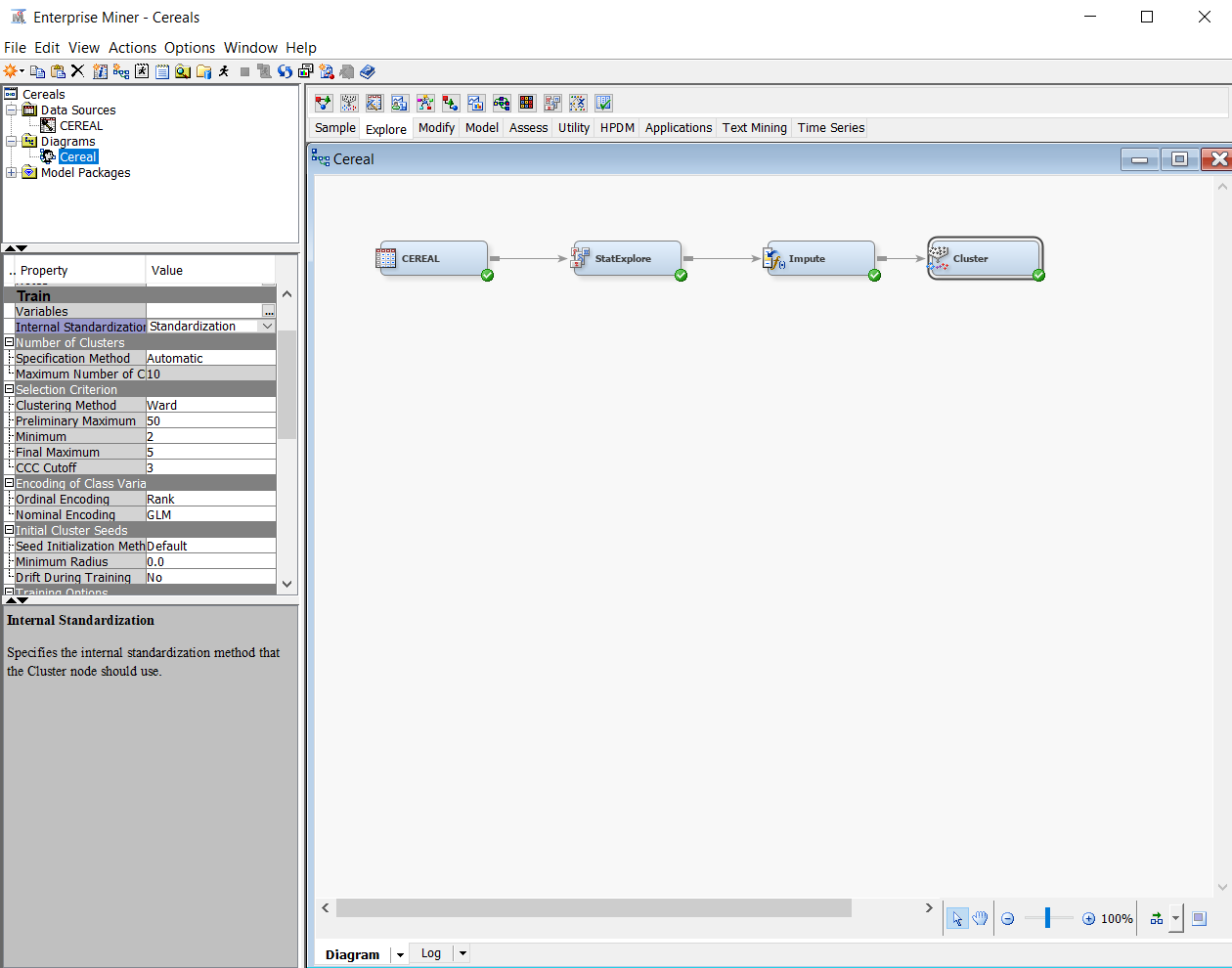
As we can see there are a few missing values so to fill them. I used the imputation with mean values.

Here we have the result of imputation:



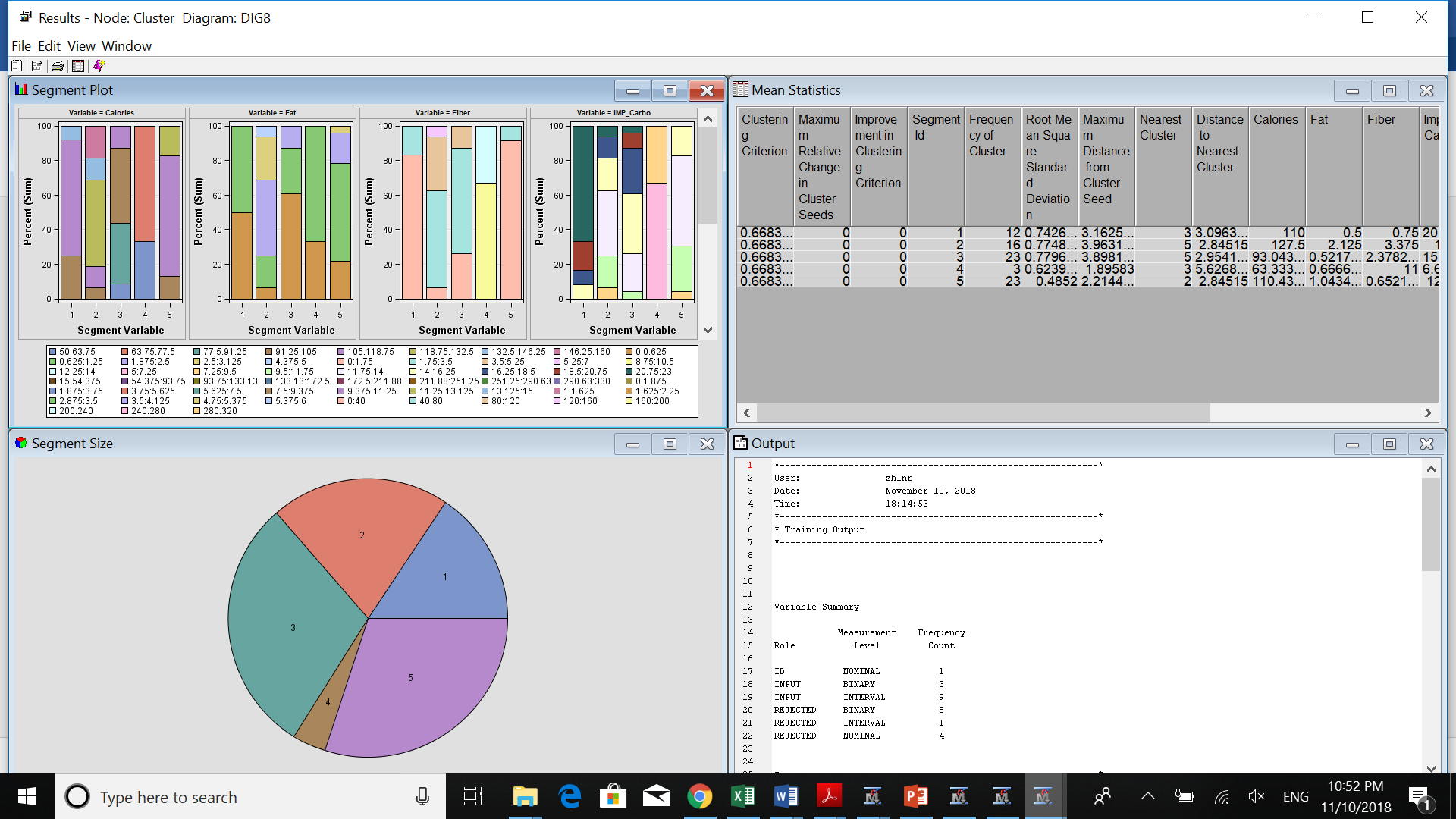


And to cluster the dataset, I used cluster node:

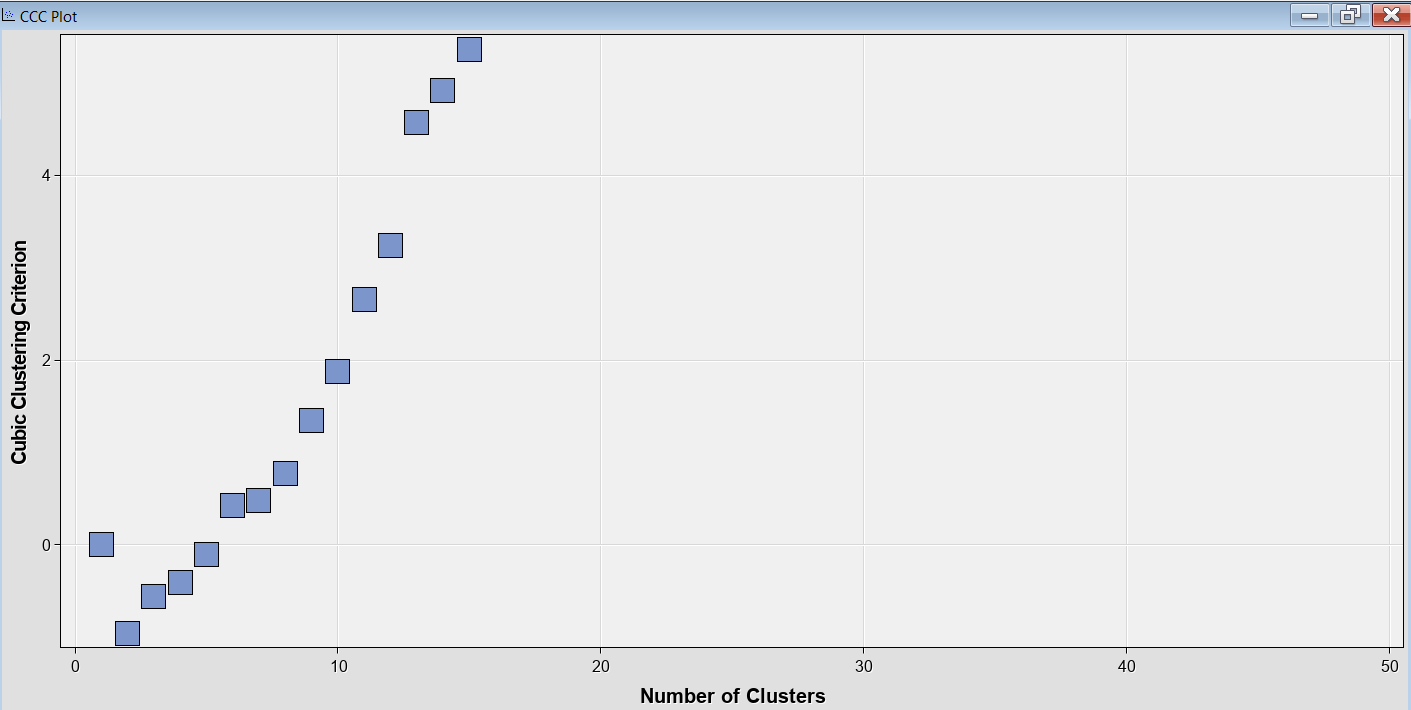




Then I ran the node and got the below result:



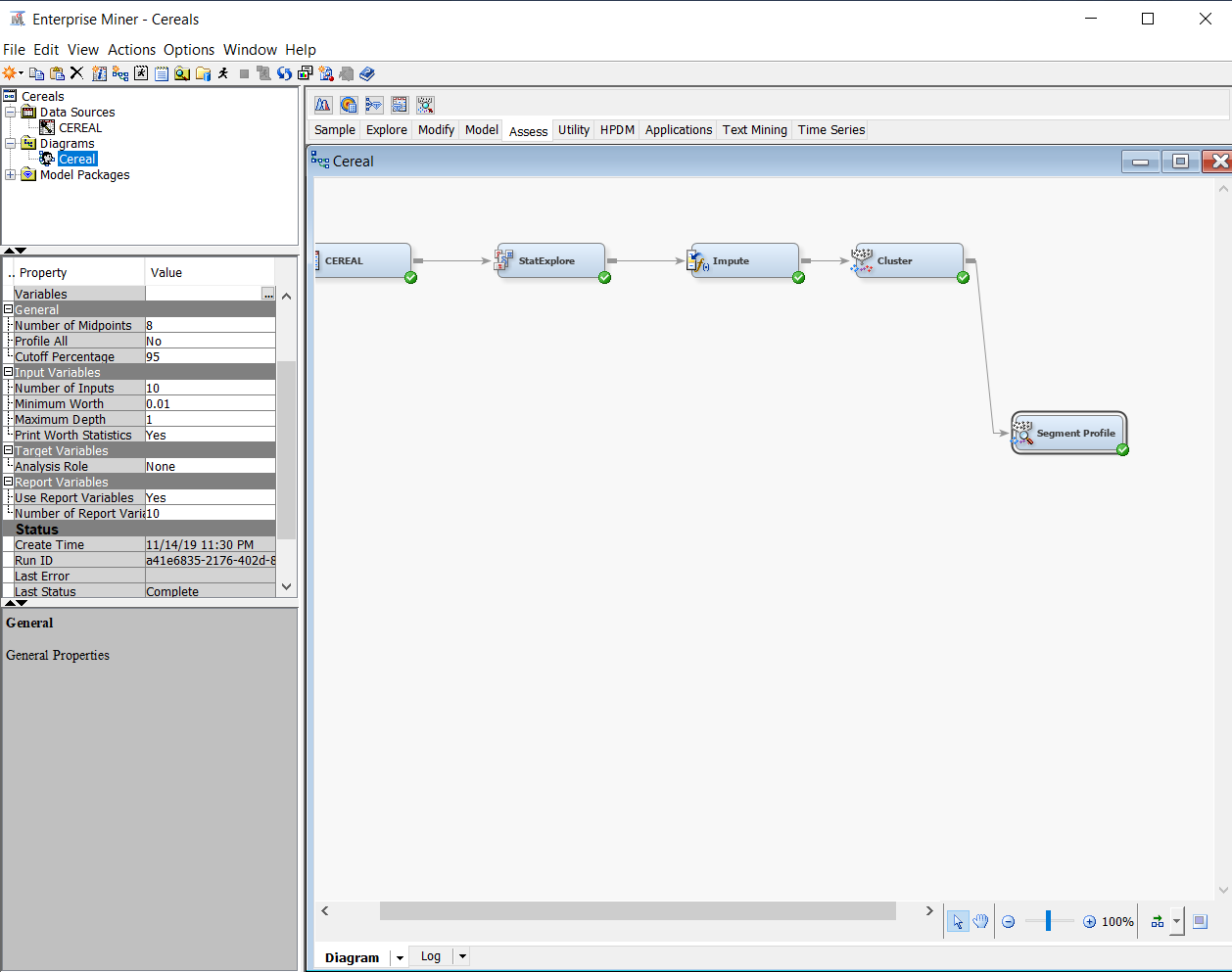
And From the view section we click on CCC plot and look at the graph.

13. Develop clustering profiles that clearly describe the characteristics of the cereals within the cluster.

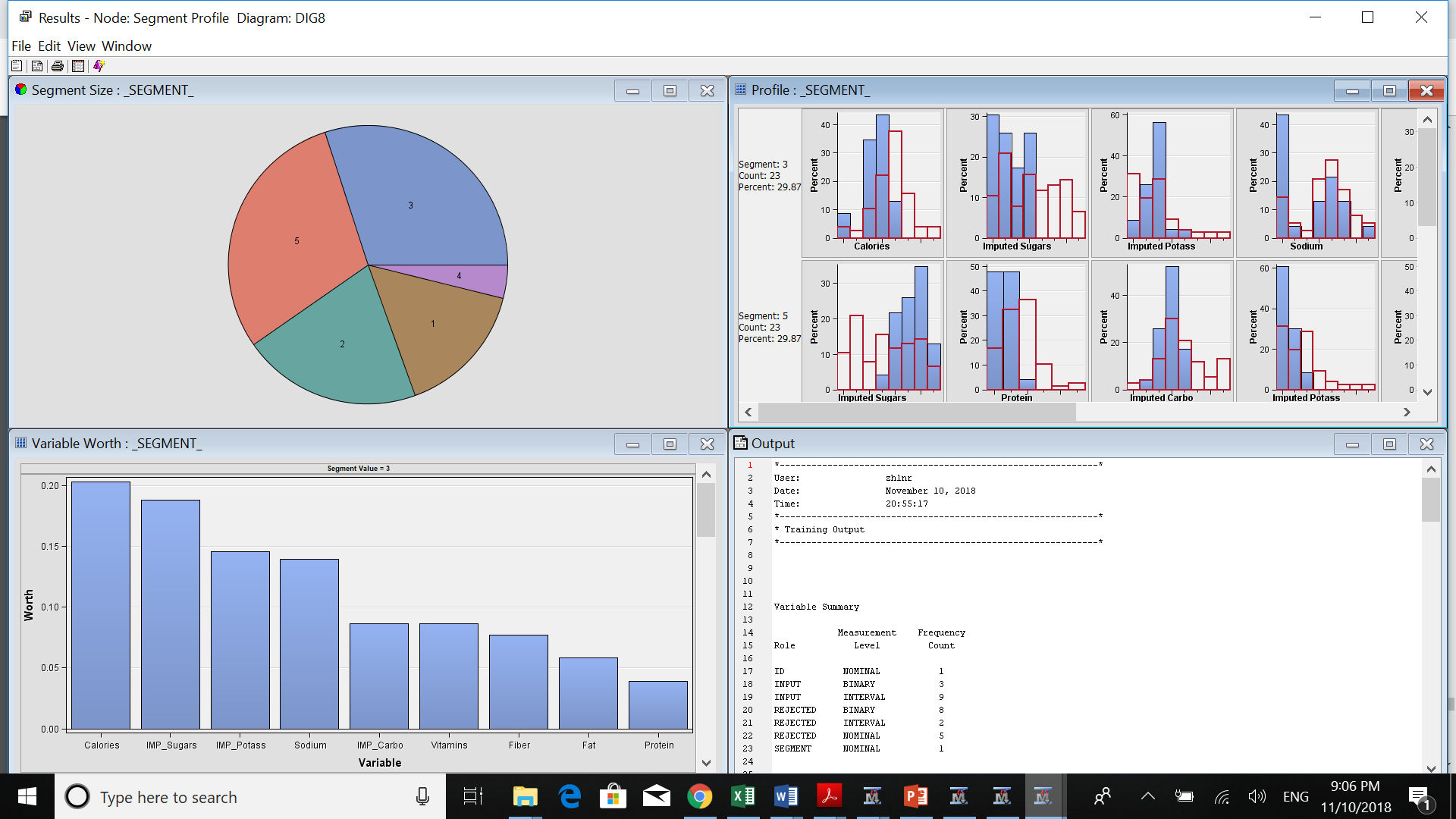
Based on the ccc plot, we can decide the number of k (clusters) within the data which is 3.

13. Develop clustering profiles that clearly describe the characteristics of the cereals within the cluster.

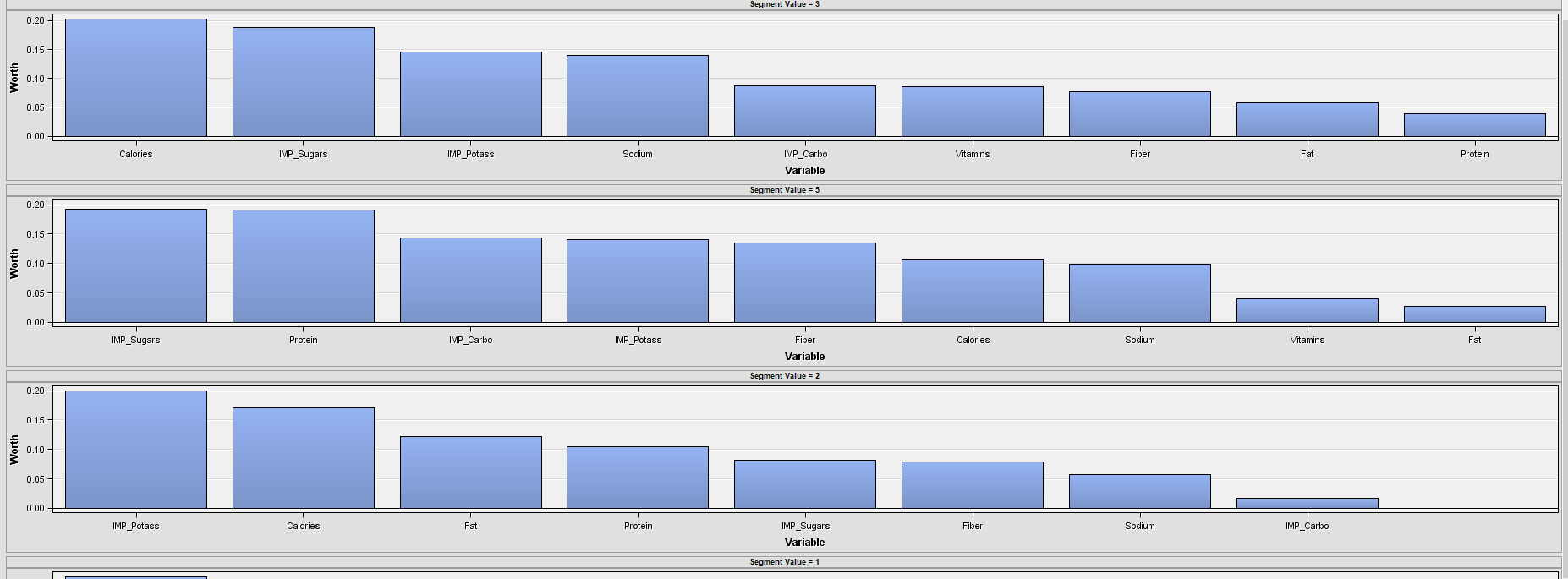
I used Segment profile to differentiate different clusters within the dataset. Because we have more than 3 variables to compare, we cannot do that by scatter plot.

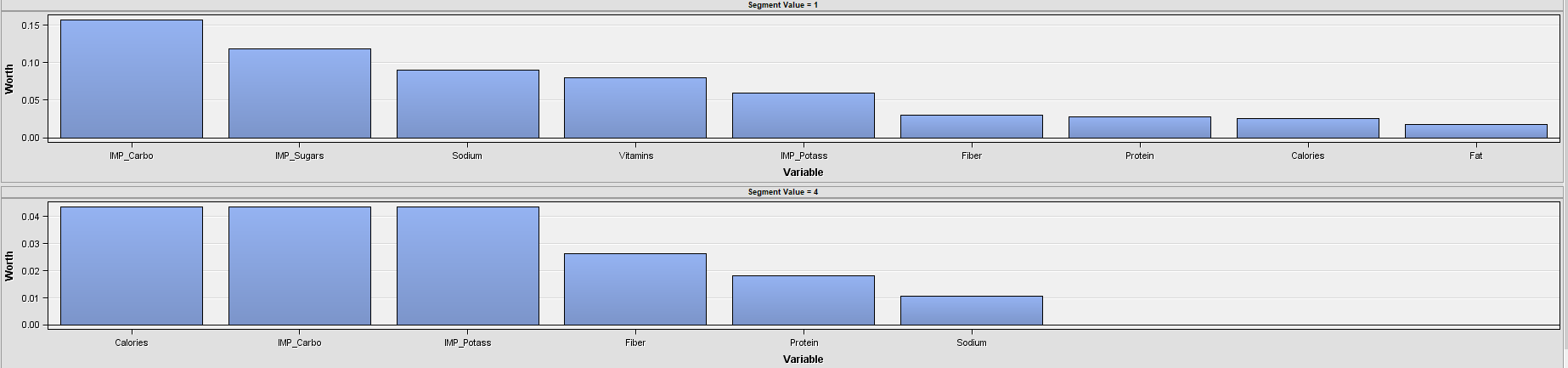


And take a look at the result:

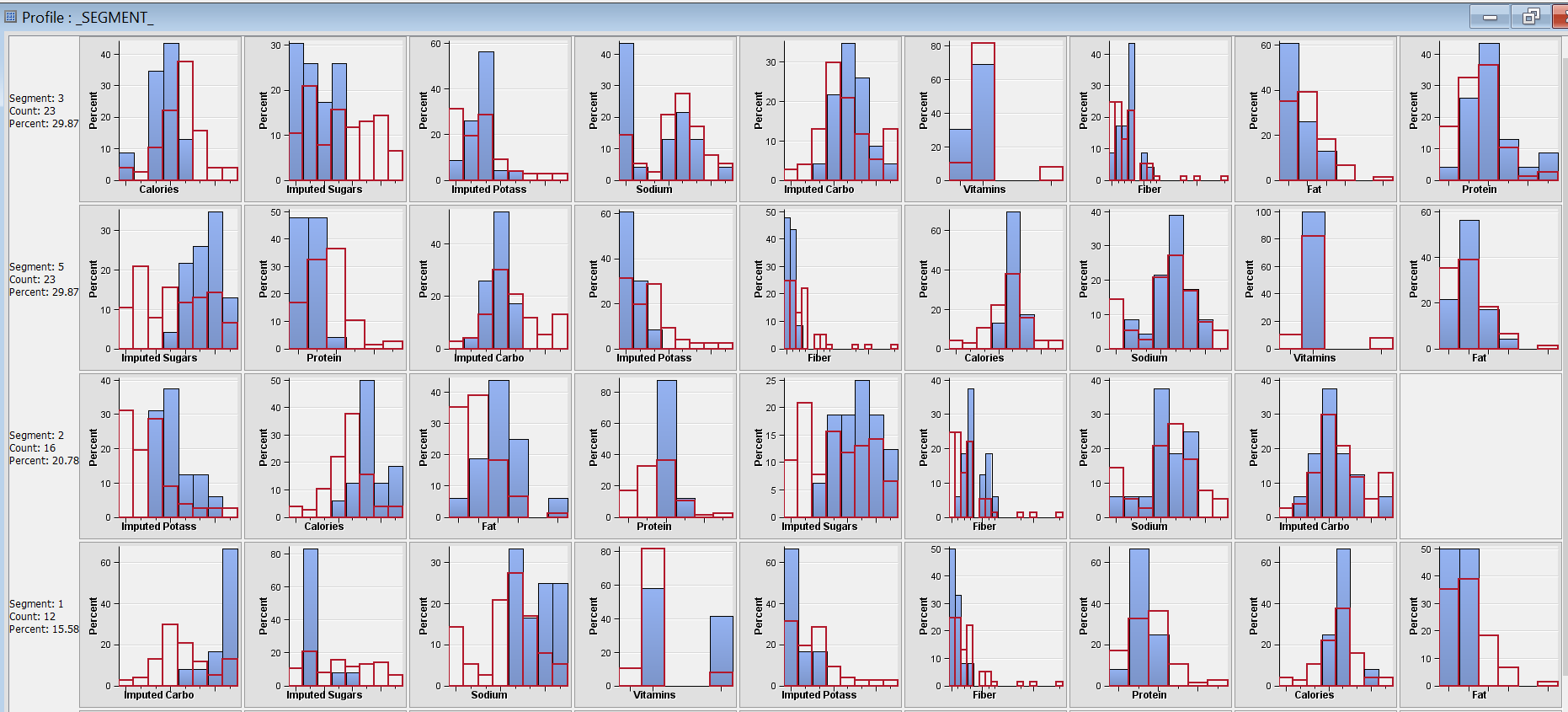


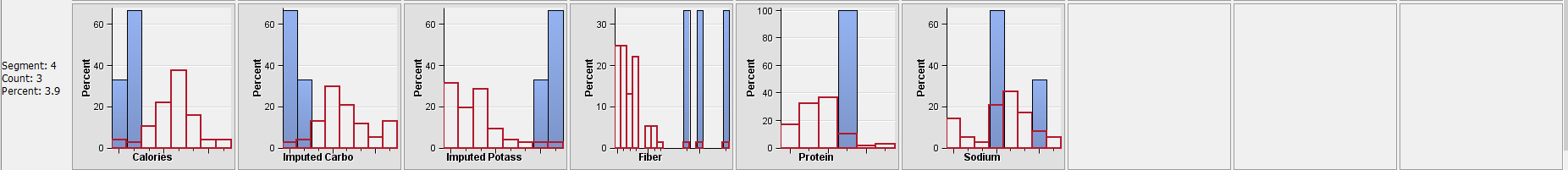
Here, we can see the worth of different variables in each segment:





So, in segment 1 potassium, calories and fat are 3 most important variables. In segment 2 sugar, protein and carbohydrates, in segment 3 calories, carbohydrates and potassium, in segment 4, Carbohydrates, sugar and sodium and in segment 5 we have calories, sugar and potassium as the most 3 important variables.





Here we can compare the values. For example, in segment 3 we can see the cereals have less calories, less sugar. At the same time, they have approximately the average amount of potass, sodium, vitamins and other variables. In segment 5, the cereals have much higher sugars than the others. But less potassium, fiber, fat, protein and approximately equal number of calories and sodium to the average. In segment 2 the cereals have higher amount of sugars, potass and calories. But average amount of sodium, fiber and carbohydrates. The cereals in segment 1 have high amount of carbo, sodium and low amount of fat, potass, fiber and sugar. The cereals in segment 4 have low number of calories, carbohydrates and high amount of potass, fiber, protein average amount of sodium.