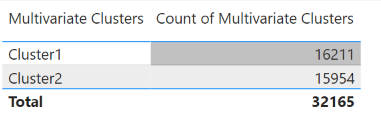
Assignment3

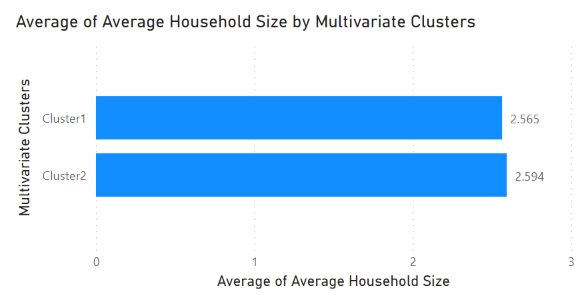
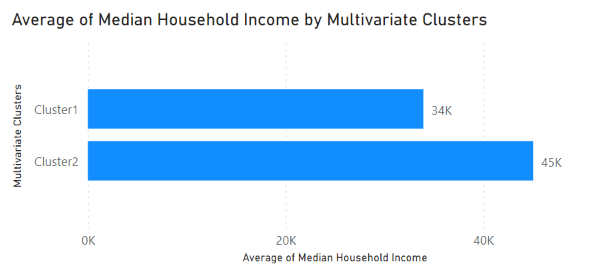
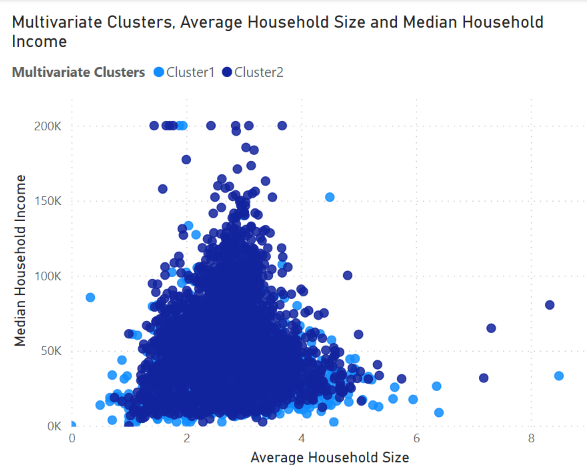
Author: Jingyi Wu (jingyiw2)

**PART A:**

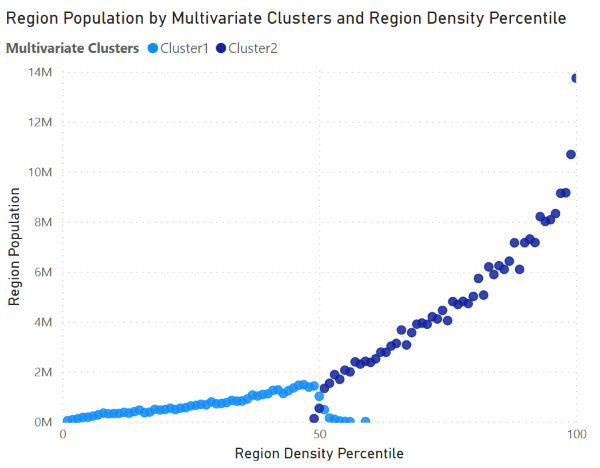
1. Here we removed region\_id for clustering, since it makes no sense and is not an attribute of people.
2. Use PowerBI to do clustering and the result is as below. 1013 instances contain missing values in their attributes. Therefore, only 32165 instances are considered here for clustering and PowerBI automatically clustered them into 2 groups.



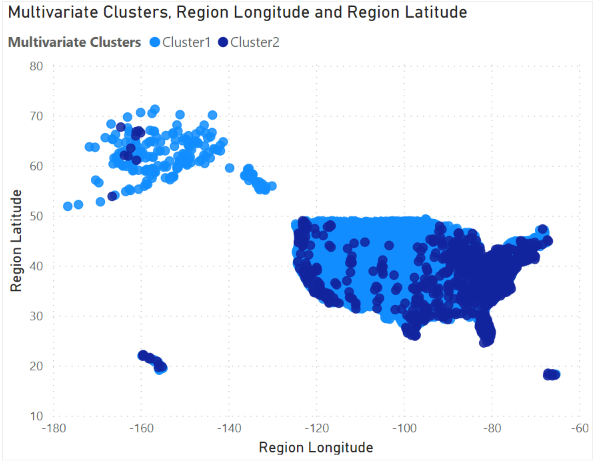
Here we can have a look at the feature distribution for each cluster to find out the features about clusters.

From the above charts, we can see that regions in cluster 1 generally have a slightly smaller household size and lower median household income.



As for the aspect of population size and density, regions in cluster1 have a lower population density and smaller population on average.



As for location side, regions in cluster 1 tend to have a higher latitude and mostly locate in the northern and middle USA. The east coast of USA and west coast of USA are mostly in cluster2.

In general, these charts show that cluster 2 include mostly regions from coastal areas of USA, which have a larger population and higher population density and people here have a slightly bigger family and higher household income. This is in accord with reality.

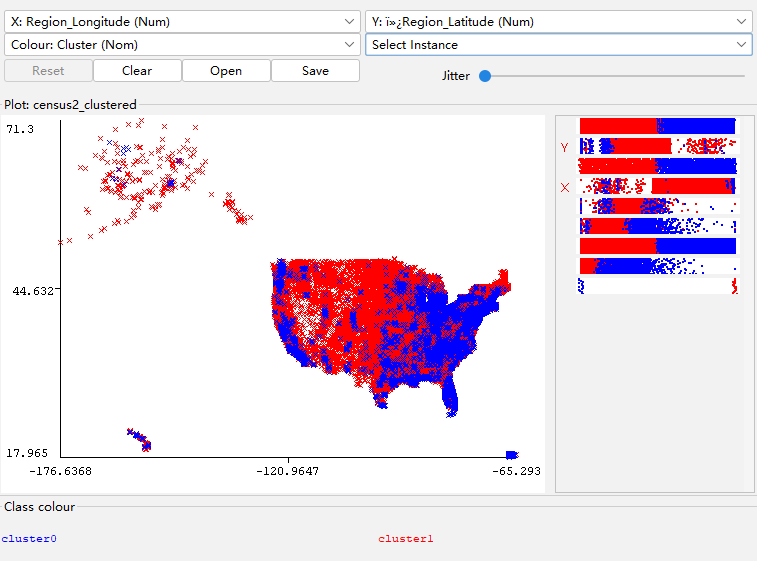
1. Apply simple k-means in weka and we will start with k=2, the result is as below.

Clustered Instances

0 15957 ( 50%)

1. 16208 ( 50%)

Within cluster sum of squared errors: 34188.93. The visualization is similar to PowerBI’s result.



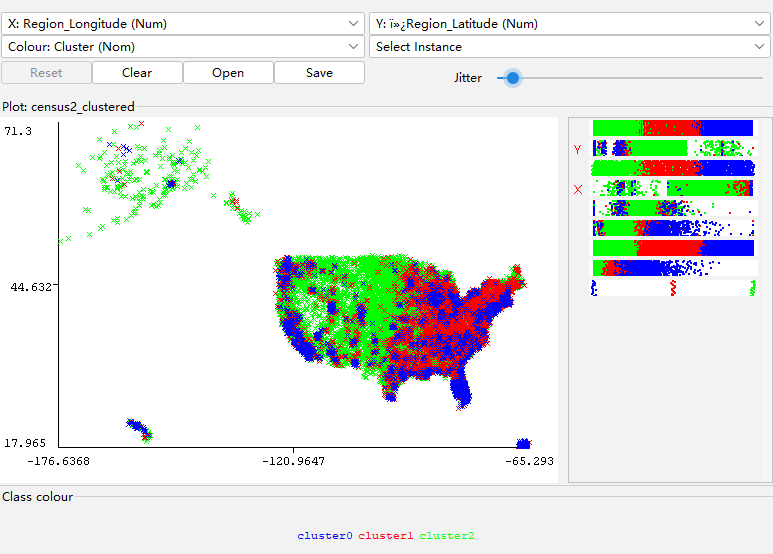
Try k=3

Clustered Instances, within cluster error here is 33738.91.

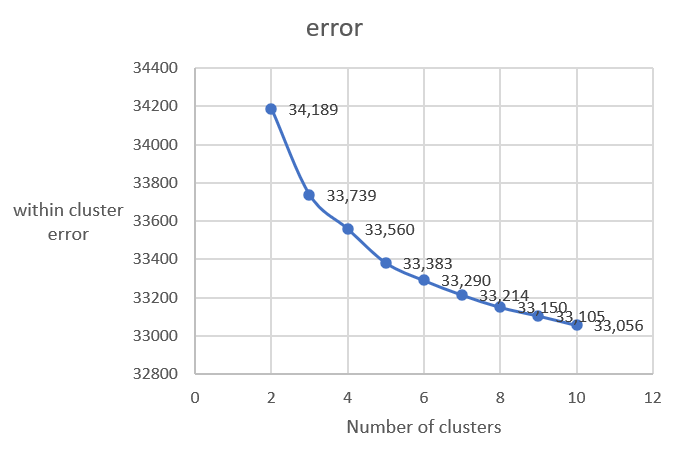
0 10818 ( 34%)

1 11223 ( 35%)

2 10124 ( 31%)

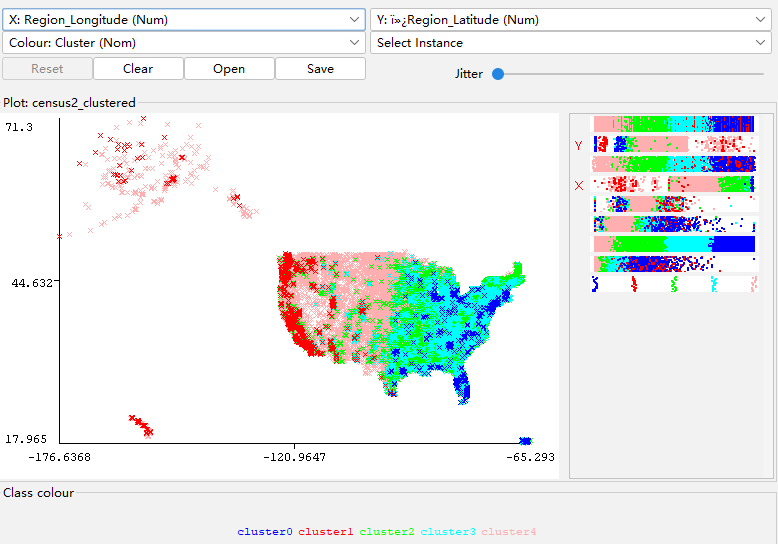


Similarly, we can get a plot of k number and within cluster sum of squared error, and as we increase k, some cluster get really small and include only a small portion of instances, like 5%.

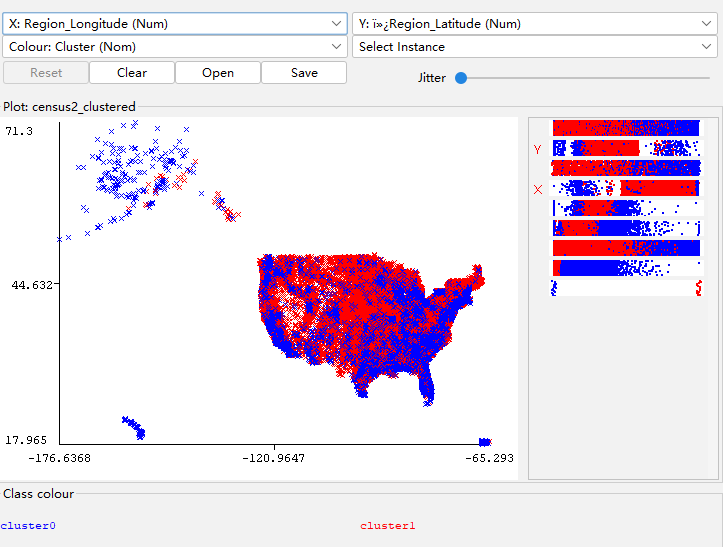


As we can see, increasing the number of clusters will decrease within cluster error but the heterogeneity between different clusters will decrease as well. Here choose a k which makes the error drop most quickly, so k=2 is indeed the best choice.

K=5 may also work. We can get more nuanced clusters.

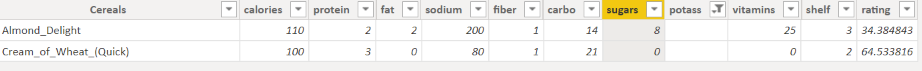
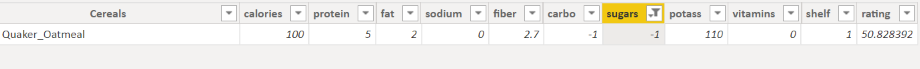


1. Try EM algorithm for clustering. K=2, we can see the result is mostly similar to k-means, but some minority instances are classified into different cluster.



Combined with reality, the upper left instances are from Alaska, the third-least populous and mostly sparsely populated state. It should be classified into cluster0, which include less populated areas. Therefore, in this case, k-means is a better option.

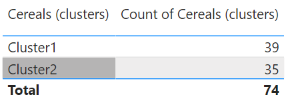
**PART B:**

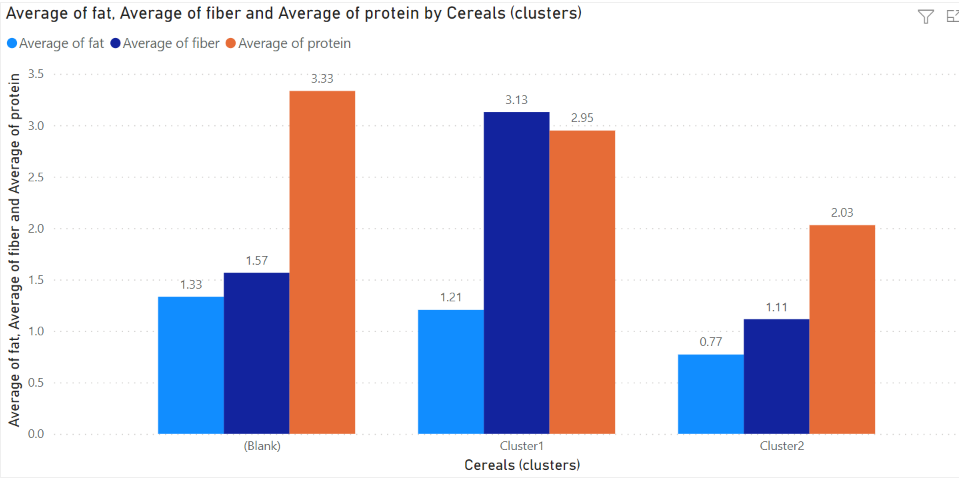
1. There are missing values in potass column and wrong value in carbon and sugar columns. 

Since this data has no apparent numeric relationship with existing data and there is no way for us to forcast a reasonable value here, for clustering I will use what weka automatically does – replace missing values with mean/mode. As for the wrong values(it’s impossible that sugar and carbon here are -1), I will set them to nan value also and do the same processing.

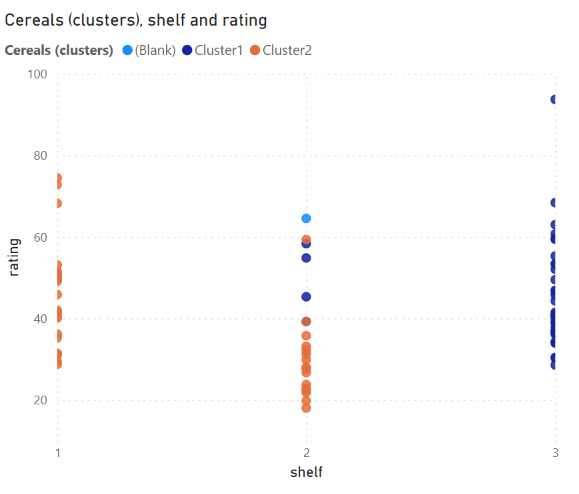
Also transform shelf to nominal value.

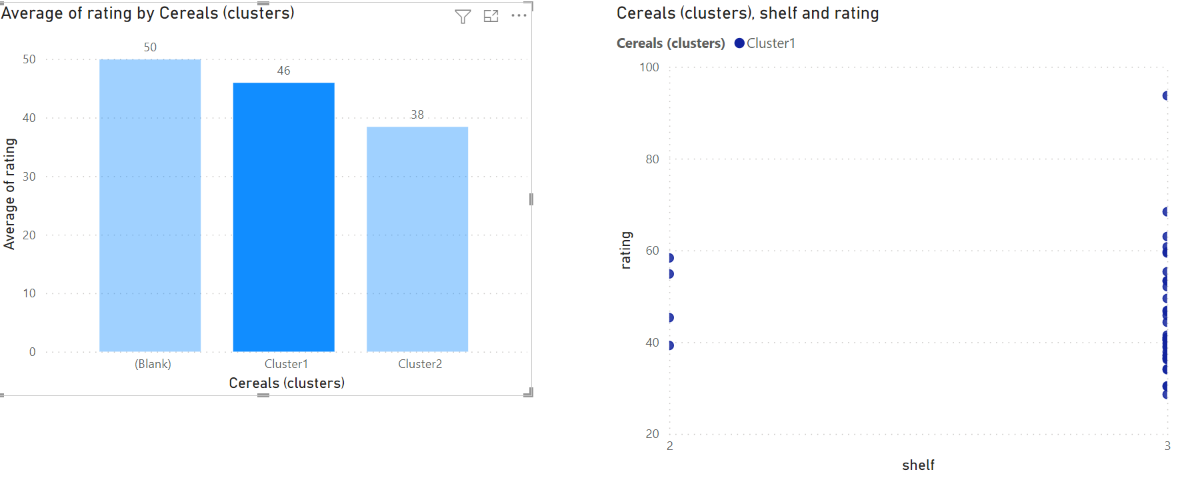
1. Use PowerBI, we temporarily ignore instances with missing values and do the clustering, and the result is as follows.

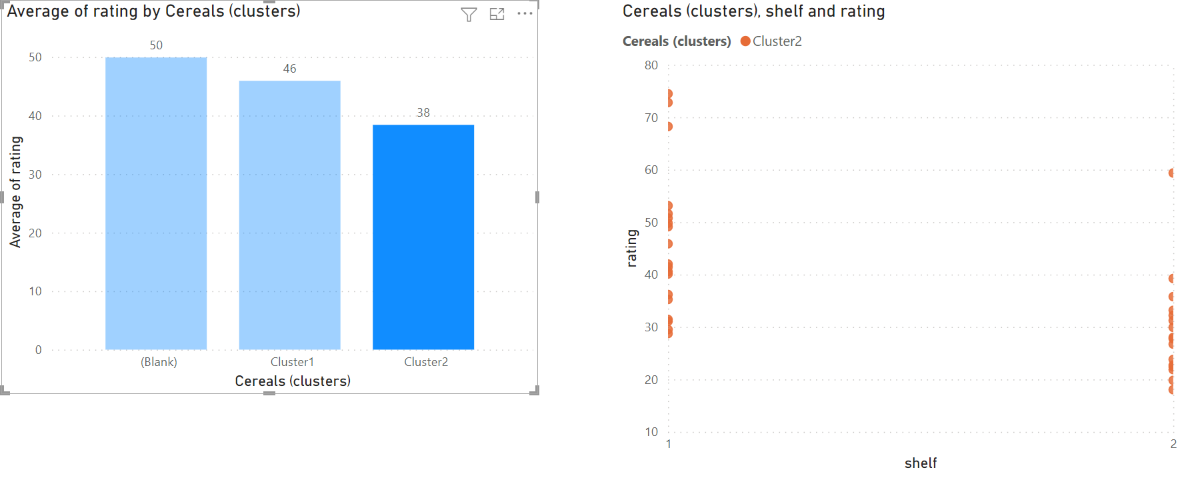




we can see that for cluster1 cereals, they have more protein and fiber, which are more beneficial.

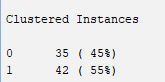


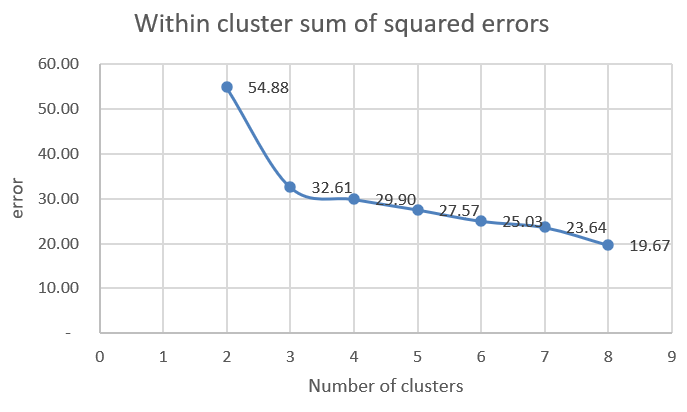


We can see that cereals in cluster 1 tend to have better ratings and mostly are placed on 3rd display shelf. Cluster2.: lower rating and both 1 and 2 shelves have them.

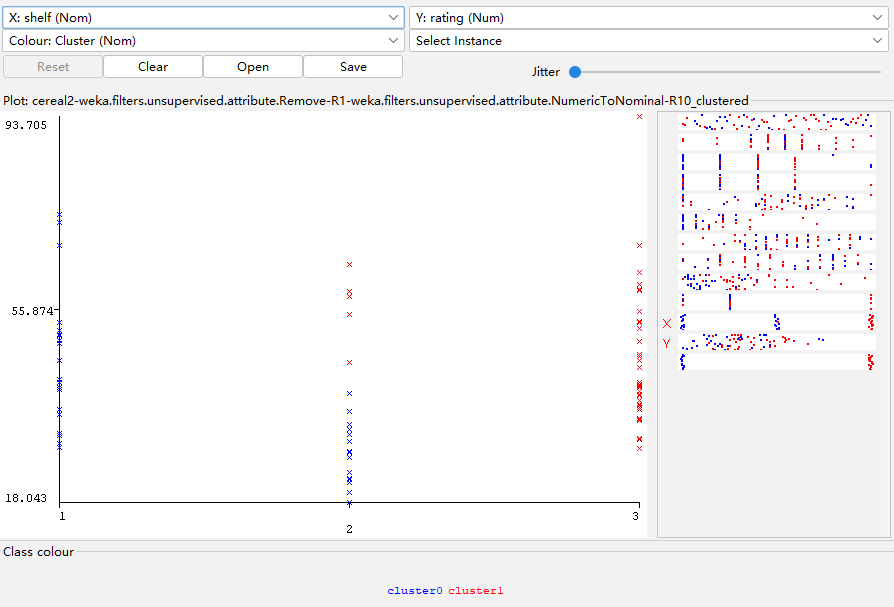
1. Use K-means in weka and we start from k=2 also. The result is as below. (Missing values globally replaced with mean/mode)

Clustered Instances

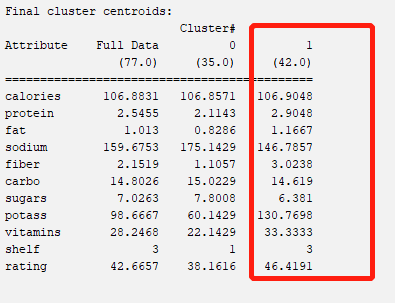




Here k=2 is the best choice.



Similar to what we got in PowerBI.

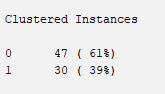


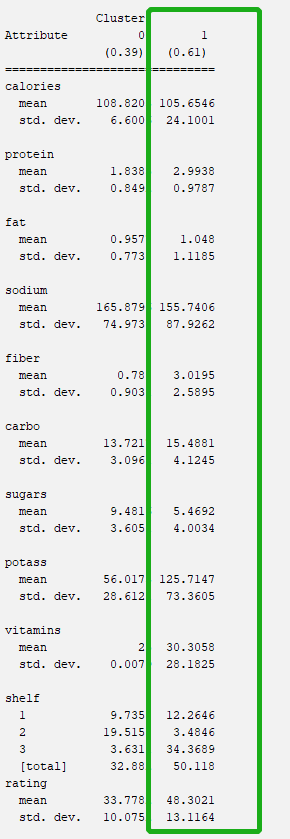
For k=2, we can see centroid cereal in cluster 1 has higher rating.

High-rating cereals are expected to generally have high protein, low sodium, high fiber, low carbohydrates, low sugar, high potassium, high vitamin and tend to be placed on 3rd shelf.

This k separates the cereals well.

Use EM clustering, also apply k=2



Here cereals in cluster 1 with the higher mean rating generally have low calories, high protein, low sodium, high fiber, high carbohydrates, low sugar, high potassium, high vitamin and tend to be placed on 3rd shelf. (I have highlighted difference feature using red color) In general, we get pretty much the same result using two methods.

1. Here we use the clustering result from k-means method, and cluster 1 cereals are healthy cereals. As is explained above, these cereals have high amount in protein/fiber/potassium/vitamin, which are beneficial to students. At the same time, low carbohydrates/sodium/sugar will not cause obesity problem. Also these cereals are popular among customers and are likely to be students’ favorites.

Since we calculate Euclidian distance for measuring similarity between instance, it is important to keep these features at similar scale, otherwise features with larger scale will have greater impact than ones with small scale, and this is against the algorithm. For example, the calories values are much bigger than protein values. Without standardization, the cluster result will be influenced majorly by calories.