CS3DS19 - Data Science Algorithms and Tools

Major Coursework

2021/2022

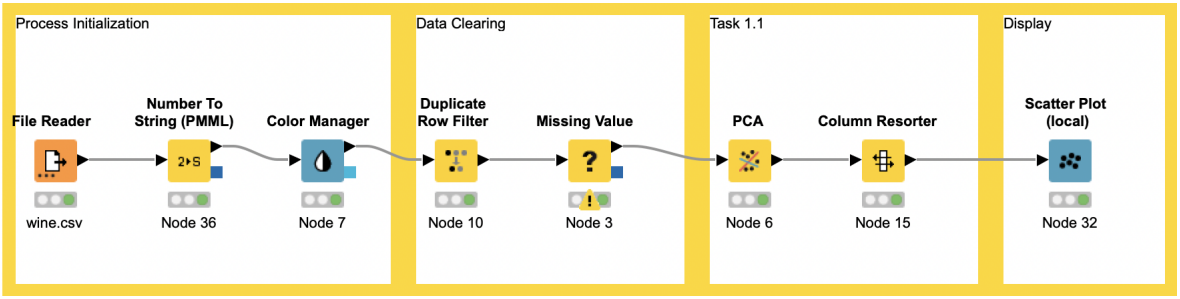
Student ID: 27020363

February 15, 2022

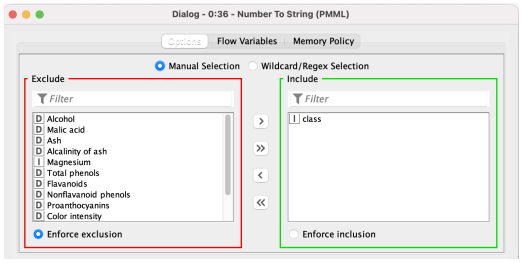
Task 1

Task 1.1 : Generate plot1

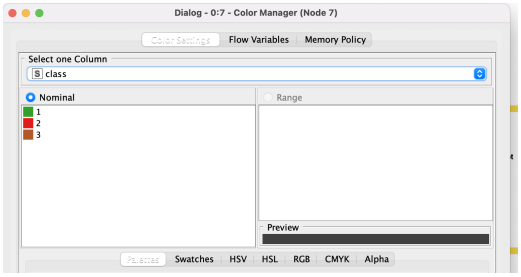
The workflow to generate plot1 is given below



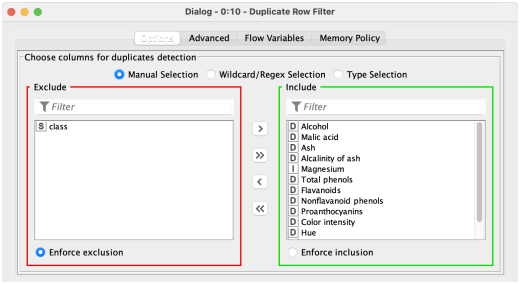
And the configurations of the nodes with descriptions are given below



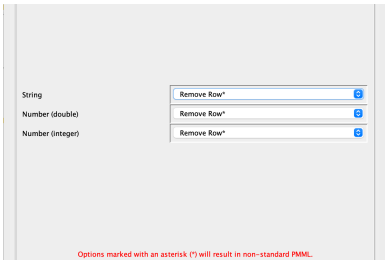
(a) Number to Category(PMML) :This node transforms the 'class' column into a categorical variable



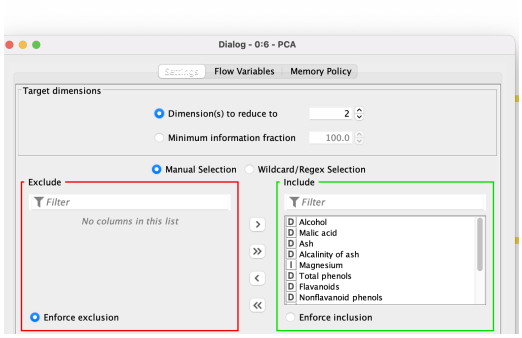
(b) Color Manager: Color Metadata for plot1 on class column



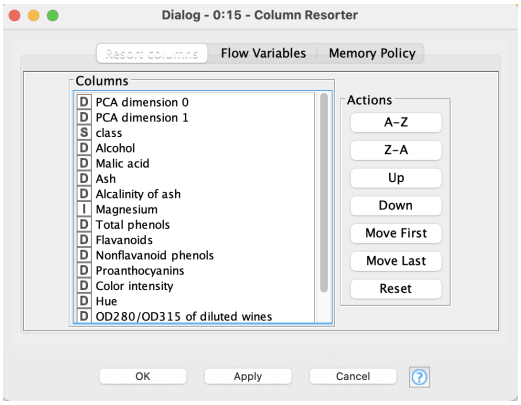
(c) Duplicate Row Filter: Standard data clearing, worth mentioning that we exclude the 'class' variable for obvious reasons



(d) Missing Value: Standard data clearing, removal of row in case some of the variables are empty(not observed)

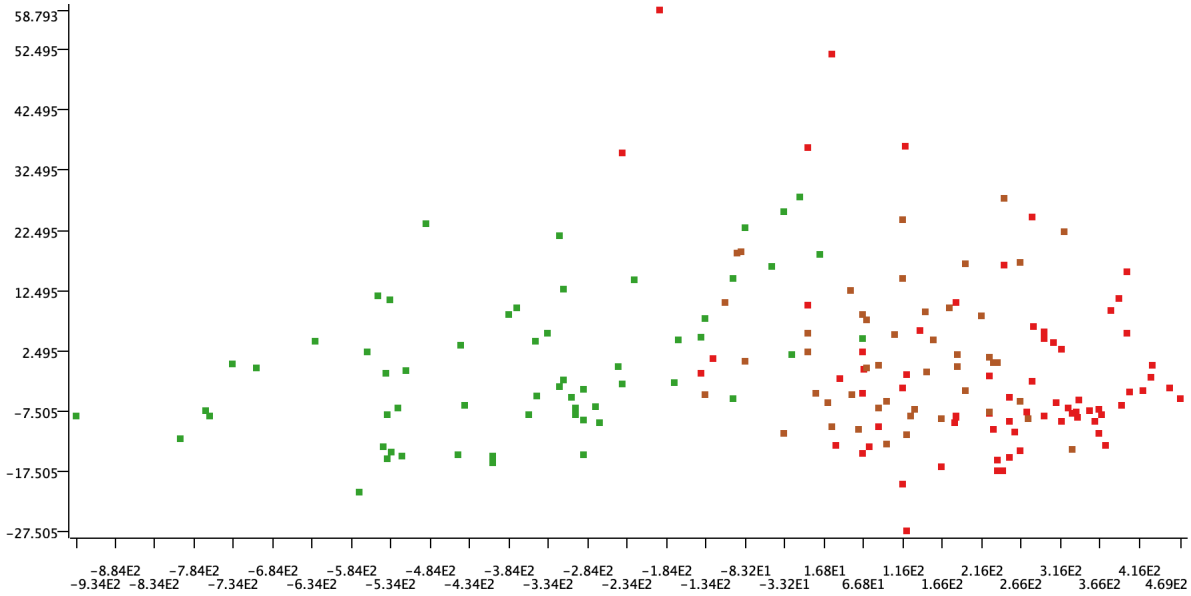


(e) PCA: The Principal Component Analysis node, we request to reduce the dimensionality to 2, worth mentioning that, as 'class' is a string column, is automatically excluded from PCA possible collumns



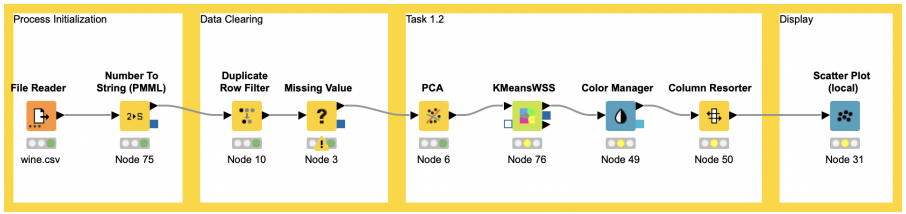
(f) Column Resorter: bring first and second principal components on top(first 2 columns), to be displayed by Scatter Plot Node

Finally, the plot1 is given below

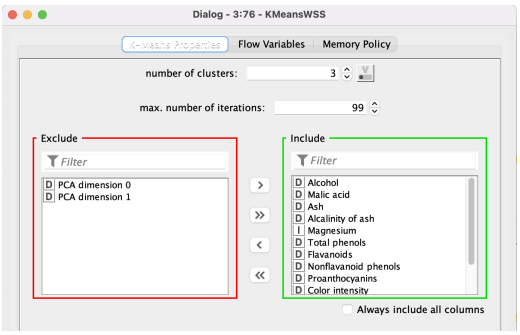


Task 1.2 : Generate plot2

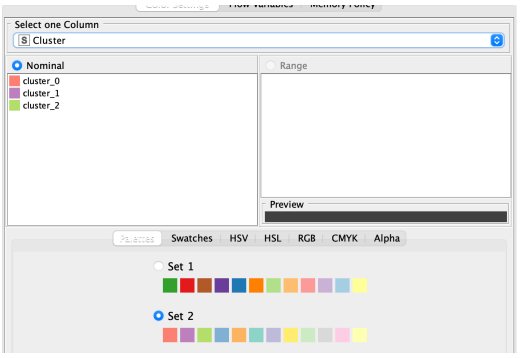
The workflow to generate plot2 is given below



Apart from the clustering algorithm adopted(K-Means), there is no significant changes with the worflow from the previous task, please see task1.1 for an overal review of the workflow. The Newrly introduced nodes configurations, as well as the descriptions are given below

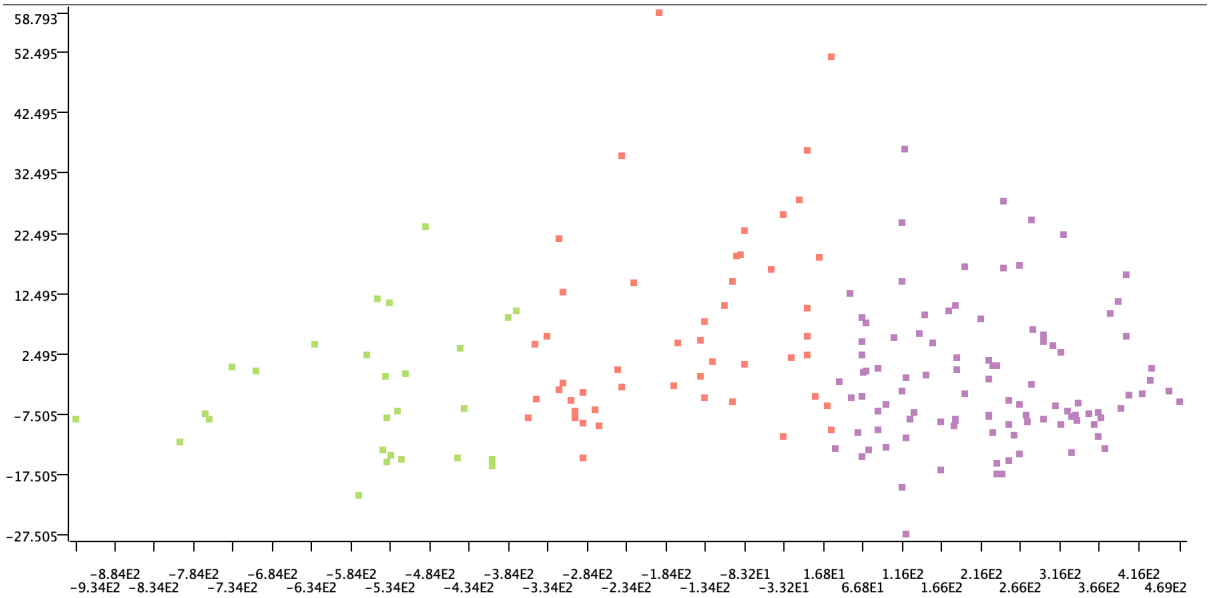


(a) KMeansWSS : this node implements our clustering algorithm of choice, the exact inner-workings to be explained on the next paragraph. A noteworthy configuration is the exclusion of PCA dimensions from the clustering process



(b) Color Manager: After KMeans finishes, we add colour metadata to our 'cluster' column, generated by the aforementioned KMeansWSS Node, to be displayed by the Scatter Plot node. The colour palette is different from the colour palette chosen on Task1.1

Finally, the plot2 is given below



0.0.1 Introduction to K-Means clustering algorithm

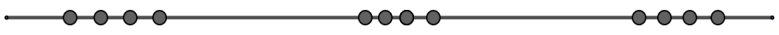
K-Means is a heuristic method for grouping similar items in a form of clusters. K is number of clusters and is pre-determined. By heuristic we mean that the algorithm does not produce absolute optimal solutions, but approximations with an given accuracy. By similarity ,on purely numeric data, we usually mean a geometric distance metric. 3 of the most widely accepted geometric metrics for determining similarity are Euclidean, Manhattan and Minkowski distances.

0.0.2 K-Means Algorithm

K-Means is an repetitive heuristic algorithm that repeats 2 distinct steps to converge into an acceptable solution. As an initialization step, K-Means selects k random points and sets them as the estimated cluster centroids. Then repeats the following 2 steps

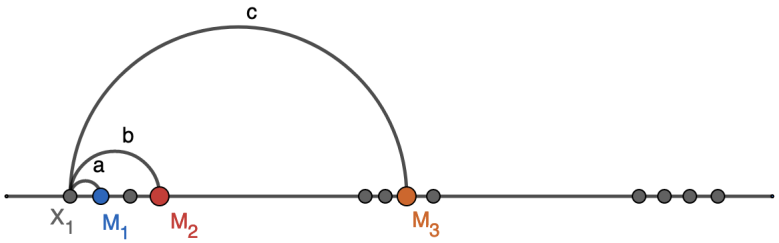
- For every datapoint in set, calculate the distances of the given point from the k estimated cluster centroids
- Assign the datapoint into the closest cluster.
- When the k clusters have been calculated, calculate the cluster's centroids, and set these centroids as the best estimated cluster centroids
- Repeat untill accuracy is acceptable(Accuracy gets calculated with SSE, SSE is explained on Task 1.5)

Let the following fictional 1D Data, $k = 3$



Let initialize $k = 3$ initial random estimated cluster centroids

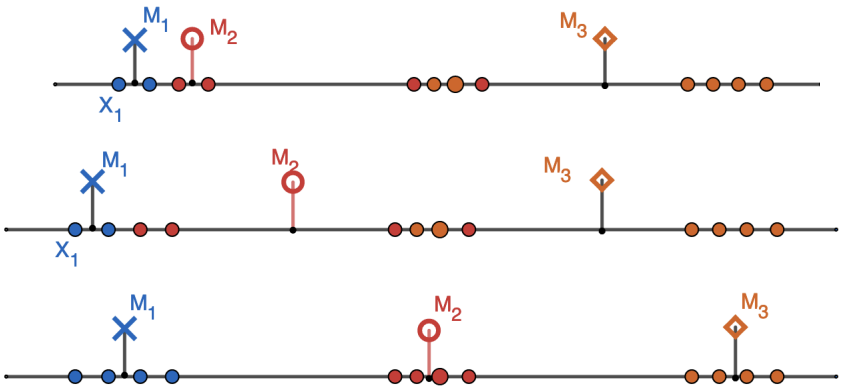
Now, the repetitive process, First , we should calculate the distances of any given point from each of the $k = 3$ estimated cluster centroids, and then assign the given datapoint into the cluster with the closest centroid. Here X_1 will be assigned on the blue cluster(M_1)



After the calculation is finished for all the datapoints, the clustering should look like the following figure

We now calculate the centroids of the clusters. the centroids do not need to be actual datapoints, they can lie on any point on the plane

Finally, after some iterations, we end up with an acceptable solution



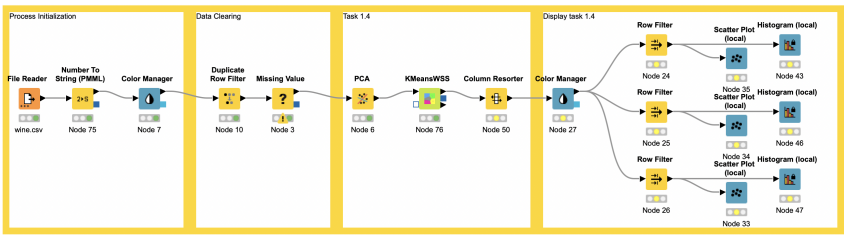
We can quantify the quality of a solution, by using Sum of Squared errors statistic(SSE). K-Means terminates when SSE converges into a value. SEE is explained in detail on Task 1.5

Task 1.3 : Compare plot1 and plot2

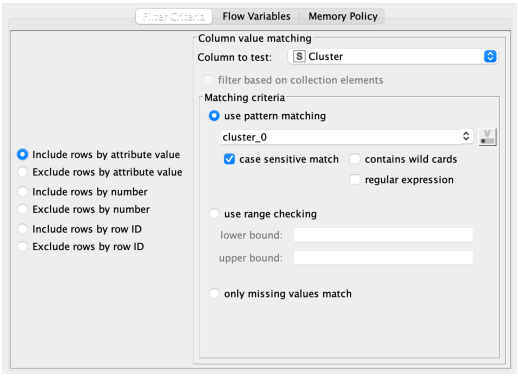
The main point of interest in the figures above, is the fact that, the clustering algorithm(K-Means, k=3) failed to appropriately partition the data, and recognise the classes provided by the dataset, This phenomenon can be explained due to the absence of the standardization process, something that will be explained in the next Task. The exact scale of the problem cannot be appropriately assessed with only those two plots though, we will need to examine the distribution of the classes in each cluster, something that will be done on the next task(Task 1.4)

Task 1.4 : plot3a,plot3b,plot3c

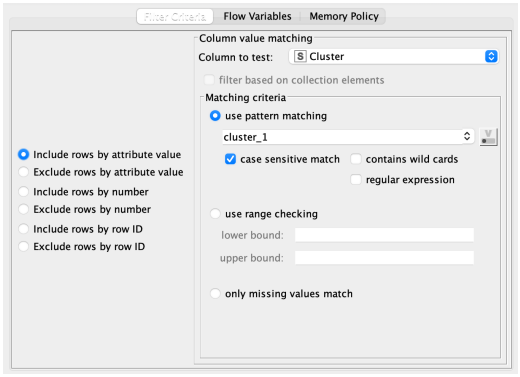
The workflow to generate the requested plots is given below



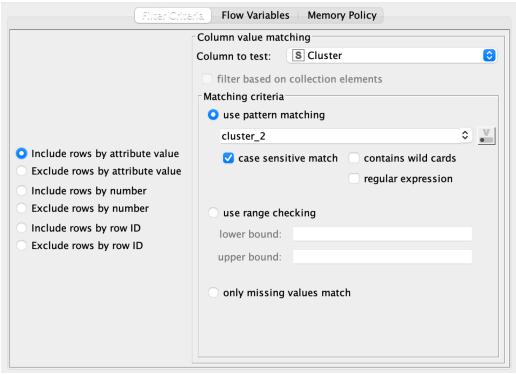
The majority of the workflow is identical to workflows of task1.1 and task1.2, so please refer to those tasks for a detailed explanations of those nodes. The major difference is on the display section. There, we use 3 row filters(one per cluster) to filter the datapoints that were assigned on each cluster. Hence in the first plot, we have all the datapoints assigned to cluster0 and vice versa. The node configurations with the nessesary filters are given below



(a) Row filter: cluster0 filter

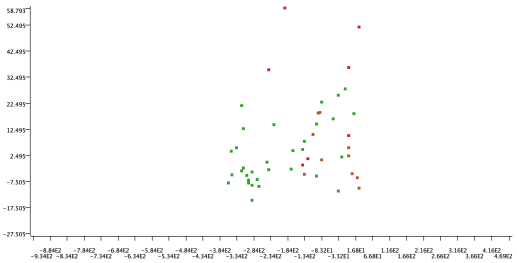


(b) Row filter: cluster1 filter

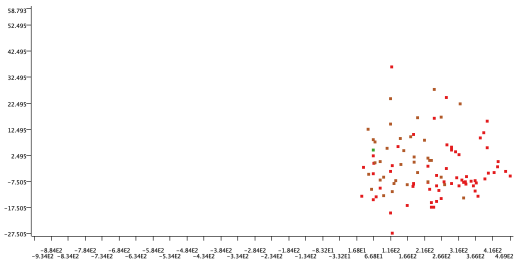


(c) Row filter: cluster2 filter

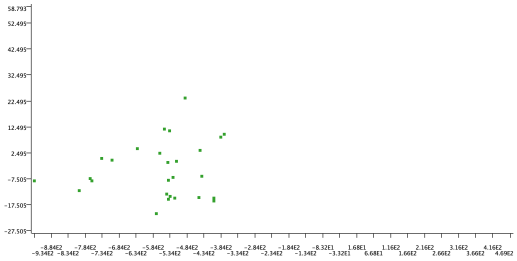
Finally, the generated plots are given in the next figure.



(a) plot3a

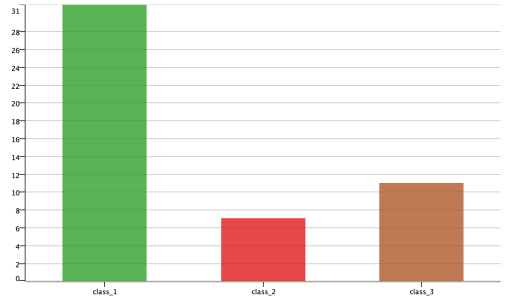


(b) plot3b

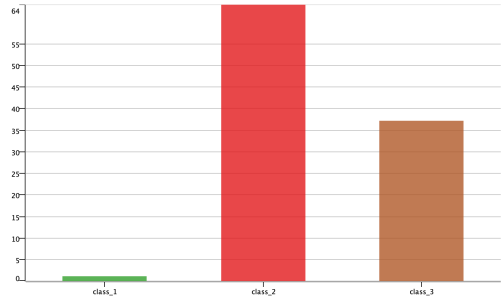


(c) plot3c

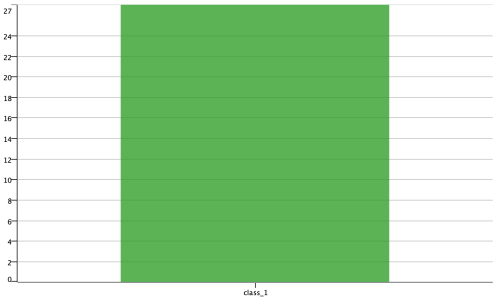
It becomes apparent that the clusters generated are composing an meaningless clustering solution. Under ideal circumstances, each cluster will had a vast majority of each of the classes, with some to none variation due to outliers or errors, Using histograms we can visualize the extend of the issue



(a) plot3a distribution



(b) plot3b distribution

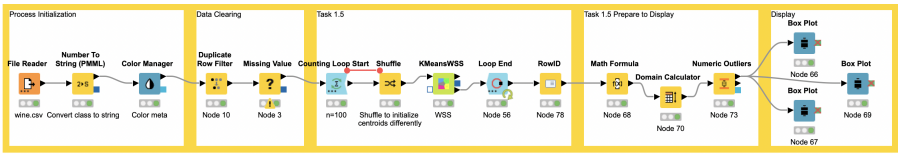


(c) plot3c distribution

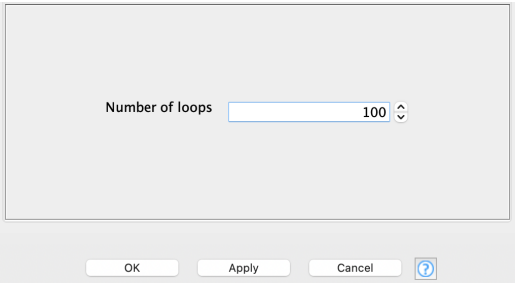
This is the result of the lack of a normalization/standarization process, something that we will explain on the next task.

Task 1.5 : Cluster Validity Measure

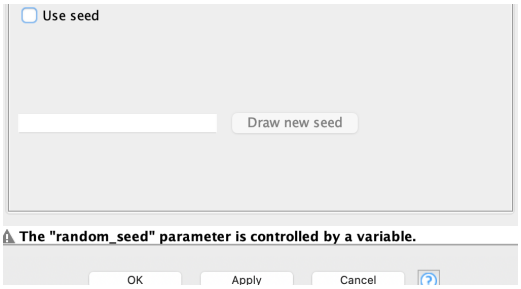
For our cluster validity measure, we will explain and intepret WSS (Within cluster sum of squares) and BSS (Between cluster sum of squares). The workflow for generating the cluster validity measures is given below



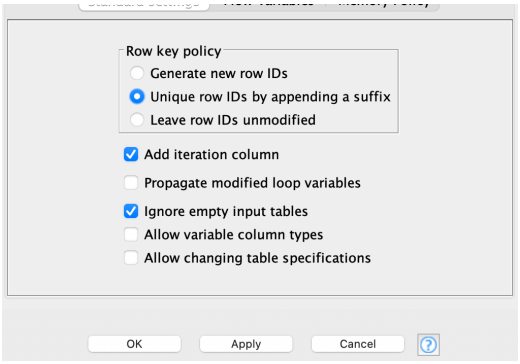
The majority of the workflow is identical to workflows of task1.1 to task1.4, so please refer to those tasks for a detailed explanations of those nodes. The Major Difference is on the repeat process and on the intepretation proccess. The newrly introduced nodes configurations are given below, along with a short explanation



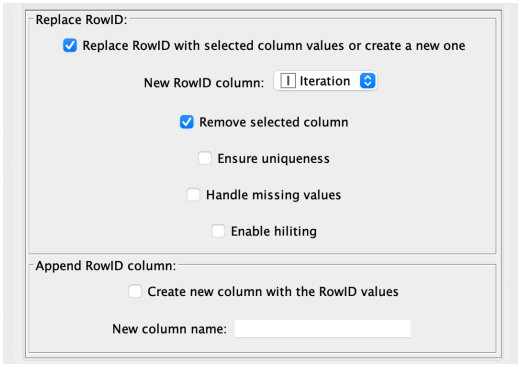
(a) Counting Loop Start: Because of K-Means is an heuristic algorithm, it makes sense to be analysed in a series of runs, we will run for $n = 100$. Thats necessary because of the fact that every heuristic algorithm is vulnerable to be stuck onto a local-minima and produce biased/wrong results [**?**].



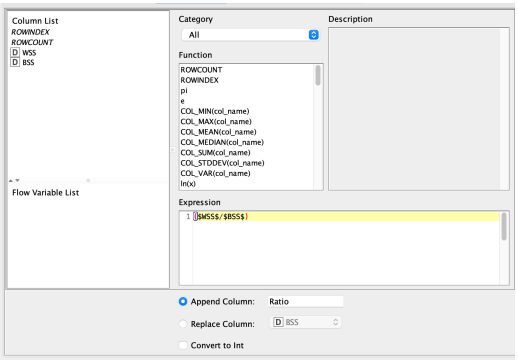
(b) Shuffle: This K-Means implementation, gets initialized by setting the first k elements as centroids. In order to evaluate the behaviour of the algorithm, we need to shuffle our dataset to enforce K-Means to initialized with different centroids [**?**]



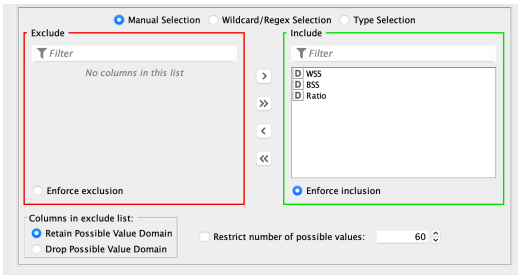
(c) Loop End: The only noteworthy mention here, is the fact that we append a new column, containing the number of iteration



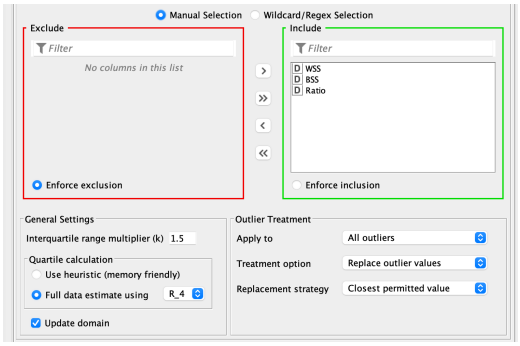
(d) Row ID: Use the number of iterations as RowID column



(e) Math Formula: A nice way to visualize the performance of K-Means, is to calculate the ratio of WSS/BSS, more on that on the 'interpretation' part



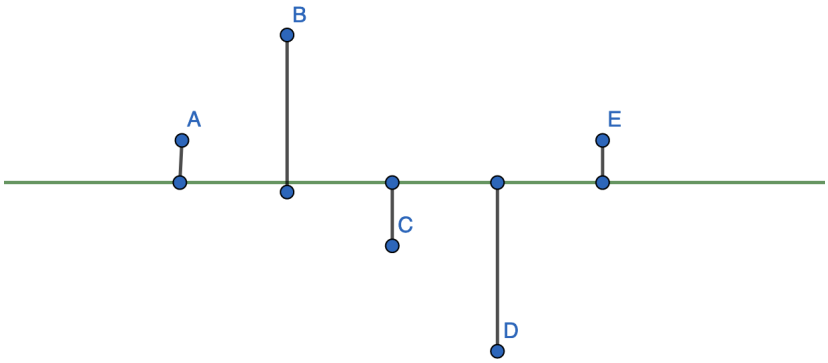
(f) Domain Calculator: As we involve very small numbers, we need to explicitly tell KNIME to keep the ranges (not to round) as much as possible.



(g) Numeric Outliers: In order to extract insightful box plots, we need to eliminate outliers(very bad initializations)

Prerequisites for explaining WSS ans BSS

Before our attempt of explaining WSS and BSS, it is essential to explain SSE (Sum of Square estimate of Errors). SSE is an evaluation metric with a plethora of applications in statistics and predictive analytics[**?**]. It is used to evaluate a model's performance against a training set[**?**]. The logic behind SSE is rather simple in nature. Let X an random variable and a model $y = \bar{X}$



We can evaluate our model's performance, by calculating the Sum of Errors, where 'error' in this case, is the distance of the observed variable from the responce of our model (i.e the mean $\bar{X}$), as follows

$$SE = \sum_{i=1}^{n} X_i - \bar{X}$$

Unfortunately, our evaluation metric has a fatal flaw, it allows for error's to 'cancel out' simply because they are placed beneath our model's line. It can be proven that, for the case of $y = \bar{X}$ SE is always 0.

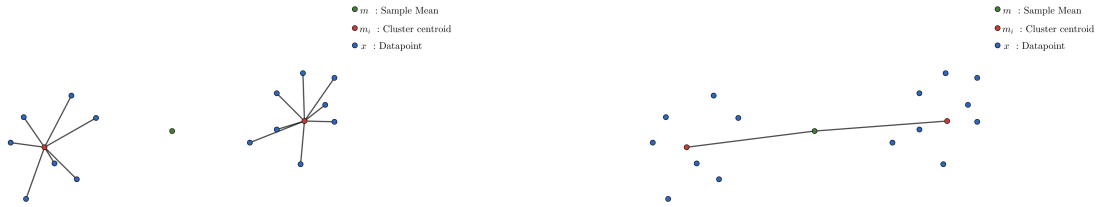
$$SE = \sum_{i=1}^{n} X_i - \bar{X}$$
$$SE = \sum_{i=1}^{n} X_i - \sum_{i=1} n\bar{X}$$
$$SE = \sum_{i=1}^{n} X_i - n\bar{X}$$
$$SE = \sum_{i=1}^{n} X_i - \sum_{i=1}^{n} X_i$$
$$SE = 0$$

For other models(in higher dimensions, where corellation of the involved variables is not 1), it may not be zero, but the fatal flaw remains, by cancelling errors we severely underestimate the errors involved. A more appropriate approach could be to square the distances, that would give us the Sum of Squared Errors (SSE)

$$SE = \sum_{i=1}^{n} (X_i - \bar{X})^2$$

WSS and BSS Inner-Workings

In the world of Clustering models performance evaluation, SSE is translated into two distinct but related ideas[**?**], WSS and BSS



(a) WSS

(b) BSS

WSS

WSS or Within Cluster Sum of Squares is a clustering model performance evaluation function. Our 'model' in this case is the cluster centroid and the error is the distance for each datapoint from the cluster centroid. WSS can be evaluated with the following formula[**?**]

$$WSS = \sum_{i=1}^{k} \sum_{x \in C_i} (x - m_i)^2$$

Where k, the number of clusters involved, C_i the individual cluster and m_i, the given cluster centroid.

BSS

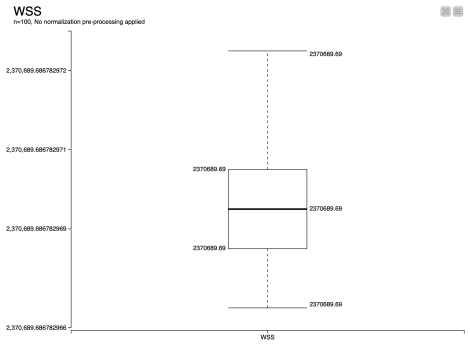
BSS or Between Cluster Sum of Squares is a clustering model performance evaluation function, just like WSS. Our 'model' in this case is the overall data mean and the error is the distance of each cluster's centroid to the overall mean. BSS can be evaluated with the following formula[**?**]

$$BSS = \sum_{i=1}^{k} |C_i|(m - m_i)^2$$

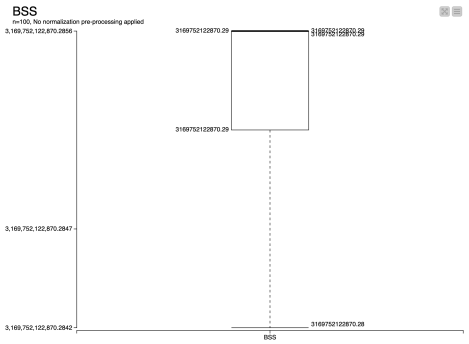
Where k, the number of clusters involved, C_i the individual cluster and m_i, the given cluster centroid.

Interpretation

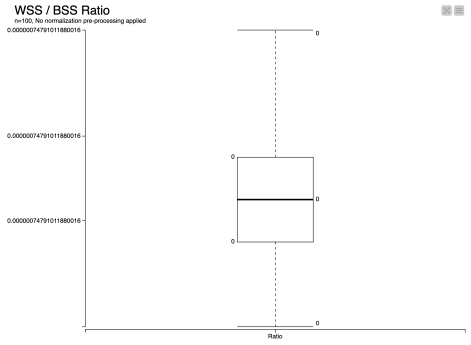
A good clustering solution, would involve well separated clusters, with low WSS to BSS ratio[**?**]. As K-Means is a heuristic based algorithm, we will not analyse WSS and BSS of a single K-Means run, but rather we will analyse their behaviour after a sequence of 100 runs.



(a) WSS



(b) BSS



(c) Ratio

As we can see, The ratio of WSS to BSS is very small, something that indicates well separated clusters(i.e a good clustering solution). However, a single cluster validity measure on its own, is insufficient to provide enough evidence for a meaningful clustering. Taking into consideration our observations from Task 1.3 and Task 1.4, this clustering [**?**]is not meaningful, is full of errors, mainly due to the absence of the normalisation pre-processing step.

Task 2

The Importance of Normalisation

The standard Lorem Ipsum passage, used since the 1500s

”Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.” Section 1.10.32 of ”de Finibus Bonorum et Malorum”, written by Cicero in 45 BC

The Inner-workings of normalisation

The standard Lorem Ipsum passage, used since the 1500s

”Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.” Section 1.10.32 of ”de Finibus Bonorum et Malorum”, written by Cicero in 45 BC

Changes in the workflows

The standard Lorem Ipsum passage, used since the 1500s

”Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.” Section 1.10.32 of ”de Finibus Bonorum et Malorum”, written by Cicero in 45 BC

New Plots

The standard Lorem Ipsum passage, used since the 1500s

”Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.” Section 1.10.32 of ”de Finibus Bonorum et Malorum”, written by Cicero in 45 BC

Compare and Contrast

The standard Lorem Ipsum passage, used since the 1500s

”Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.” Section 1.10.32 of ”de Finibus Bonorum et Malorum”, written by Cicero in 45 BC

References

[1] *Shannon, C., 1948. A Mathematical Theory of Communication. Bell System Technical Journal, 27(3), pp.379-423.*