K-Means Clustering

**Overview of the Process: The process itself has a simple structure: reading the CSV file with the data, selecting attributes (columns) that we will use in our clustering (income, spending score), normalizing the data to make all the attributes’ weights similar, clustering itself, and visualization to look at how clusters look like in our model.**

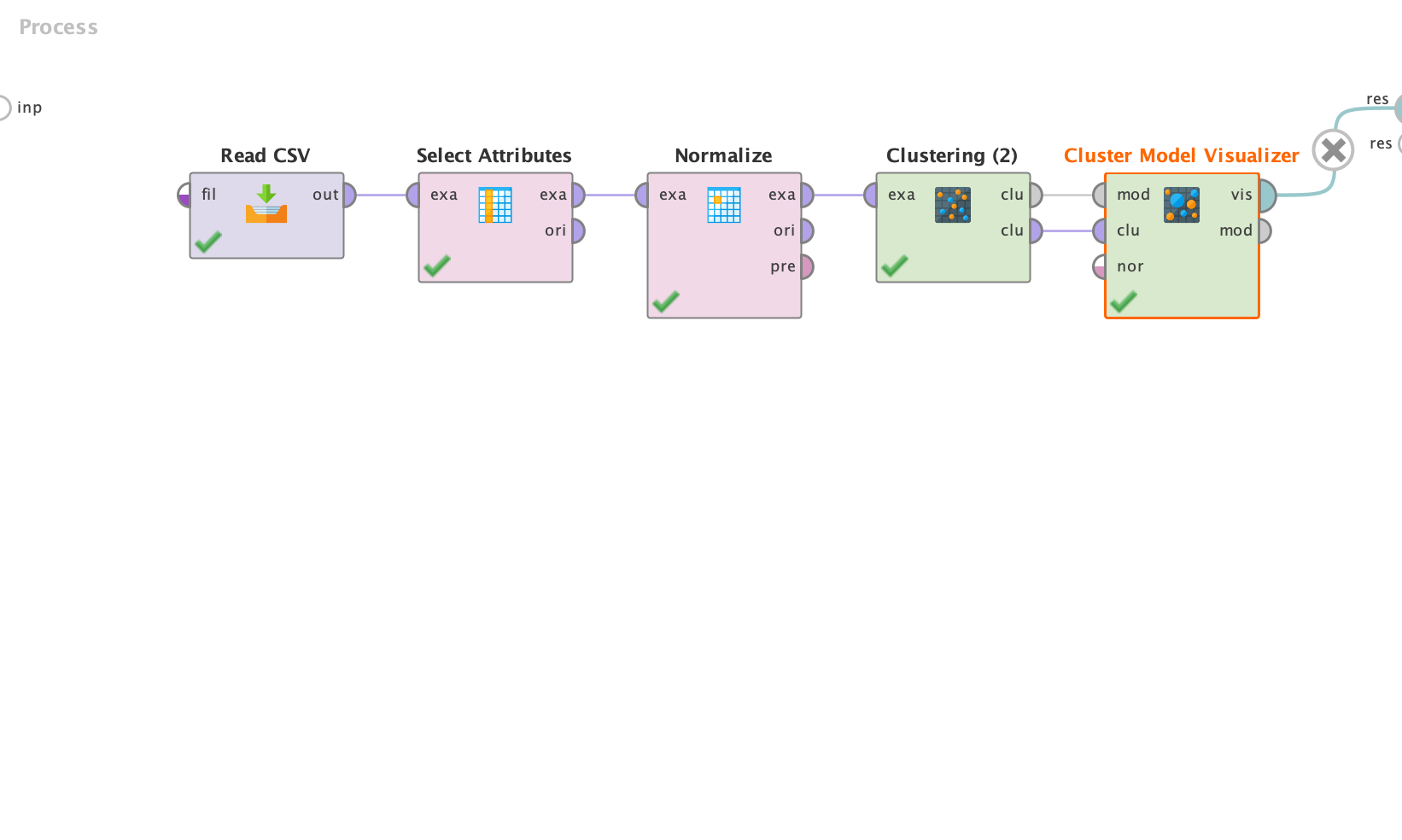
****

Figure 1. Scheme of RapidMiner K-means clustering process.

**Number of Clusters: Before we look at the scatter plot of the points where the y-axis is spending score and x-axis is annual income, we cannot know how many clusters we want to make. However, as the data is located in 5 nearly round areas, we can choose to use 5 clusters. It is also relevant for our purpose of increasing the mall profit while targeting the specific actions to different audiences.**

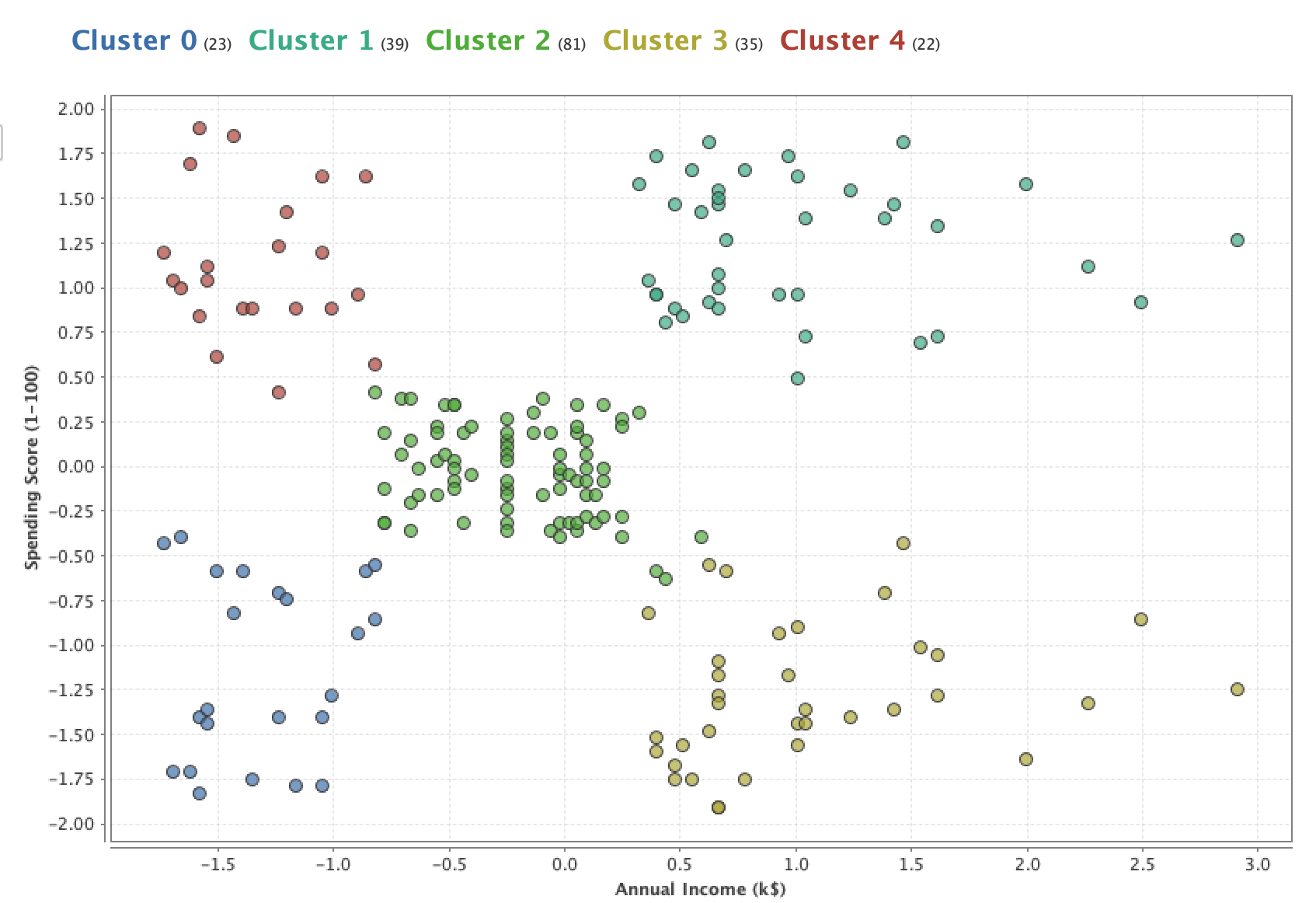
****

Figure 2. Scatter plot of points in 5 clusters (Euclidian distance metric, Z-transformation).

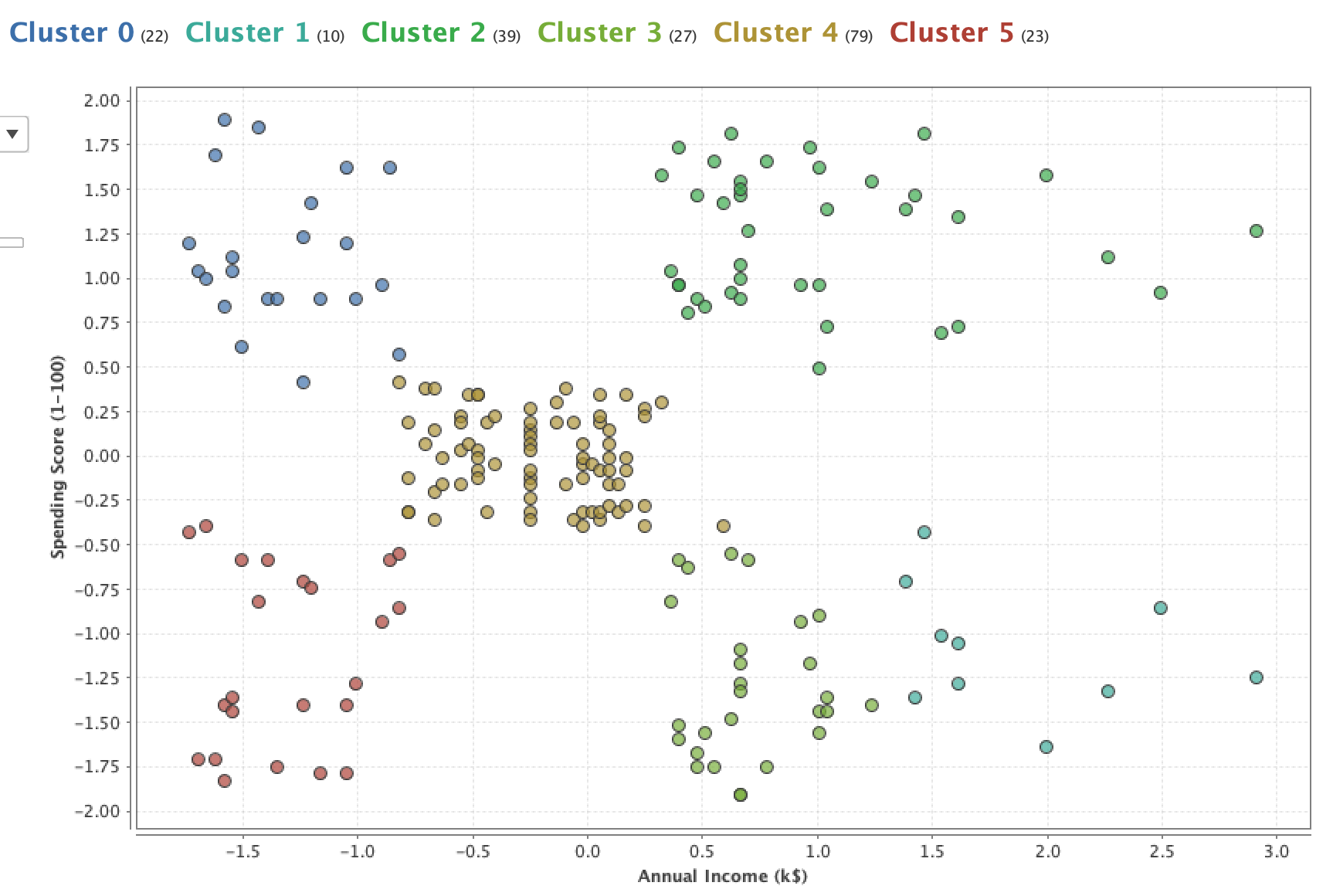
****

Figure 3. Scatter plot of points in 6 clusters (Euclidian distance metric, Z-transformation).

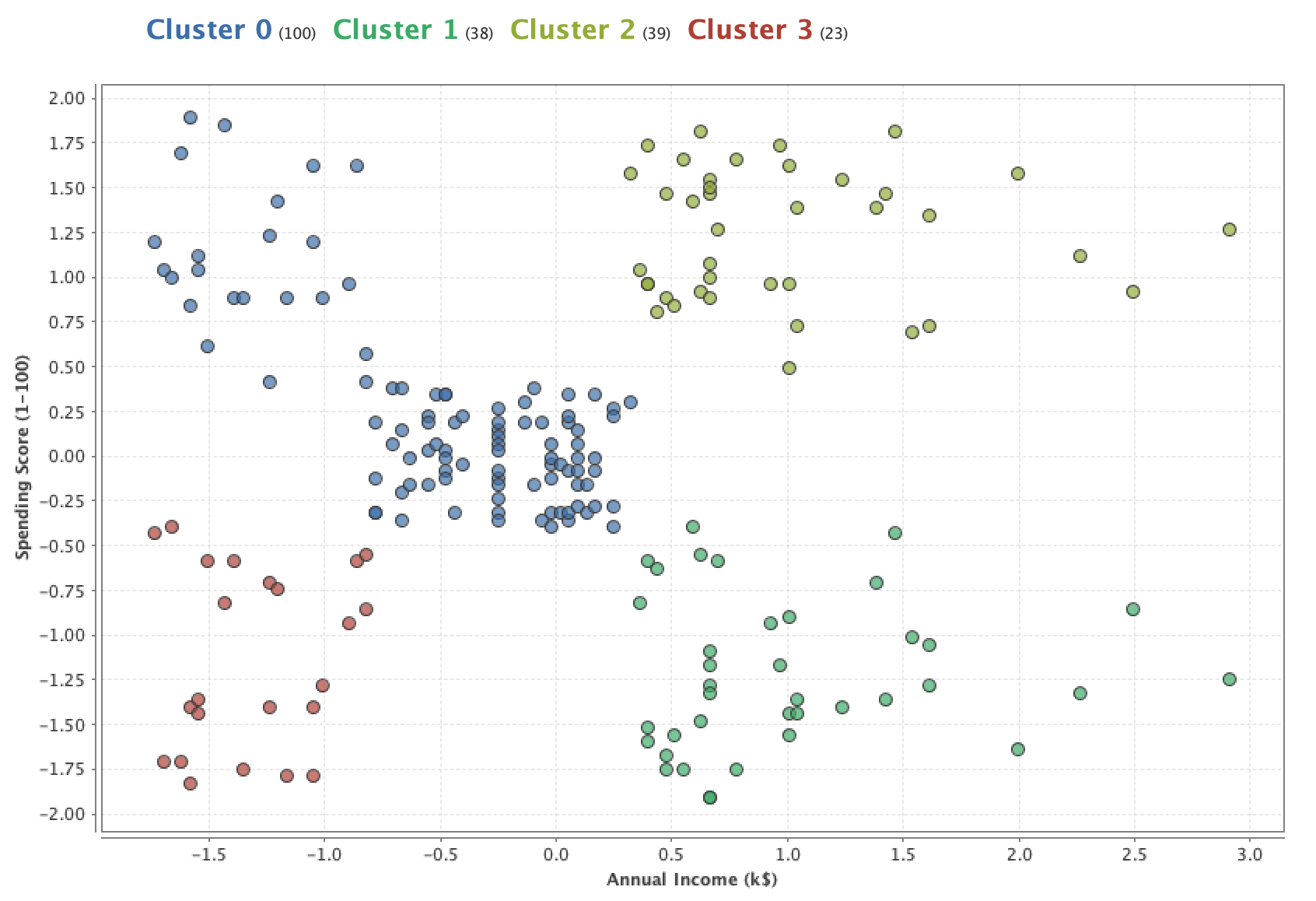
****

Figure 4. Scatter plot of points in 4 clusters (Euclidian distance metric, Z-transformation).

Even though we tried to use 4 and 6 clusters instead of 5 exactly, 5 clusters give us the best clustering that suites our purposes.

**Normalization Technique: We need to normalize the data because the two features we use do not have the same metric, in which they are measured. Therefore, the clustering can be incorrect without proper normalization. We have two possible alternatives here: z-transformation and range transformation (usually the range is 0 – 1).**

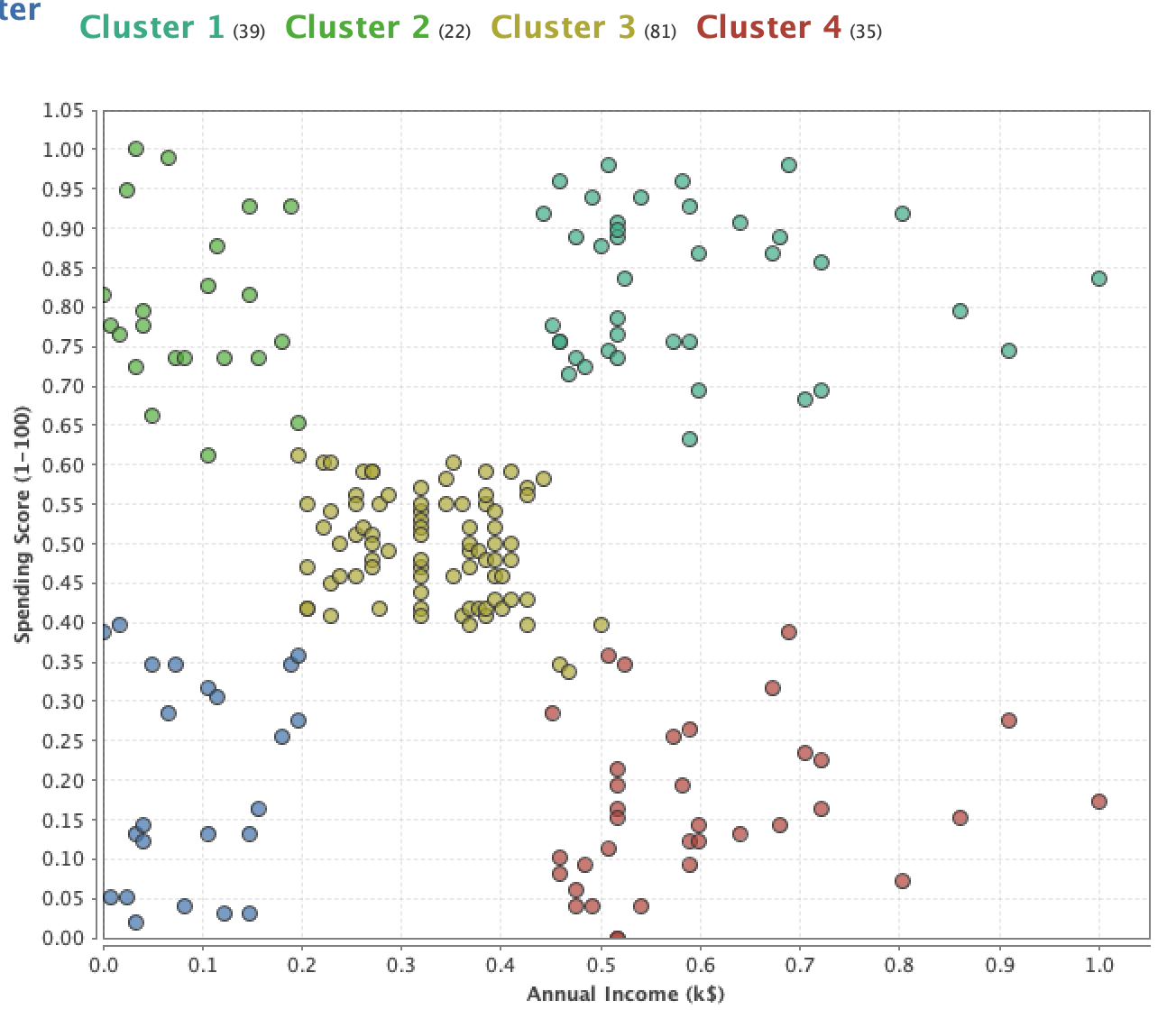
****

Figure 5. Scatter plot of points in 5 clusters (Euclidian distance metric, range transformation (0-1 range)).

It seems more useful to use z-transformation because some points in the scatter plot with range normalization are not fully visible (they are a bit off the axis). It is more visually appealing to use the z-transformation technique, yet in terms of normalization quality, both methods are the same.

**Distance Metric** Distance metric is another critical parameter for our k-means clustering. We tried various distance metrics: cosine similarity, Euclidian, and Manhattan. As we can see, Euclidean and Manhattan metrics give us the same results! Cosine similarity metric, however, gives us strange clustering, which is not what we want to obtain. Thus, we can use Euclidean or Manhattan distance metric (we will use Euclidian) as they give us the results we want to have; we have clearly defined groups of customers.

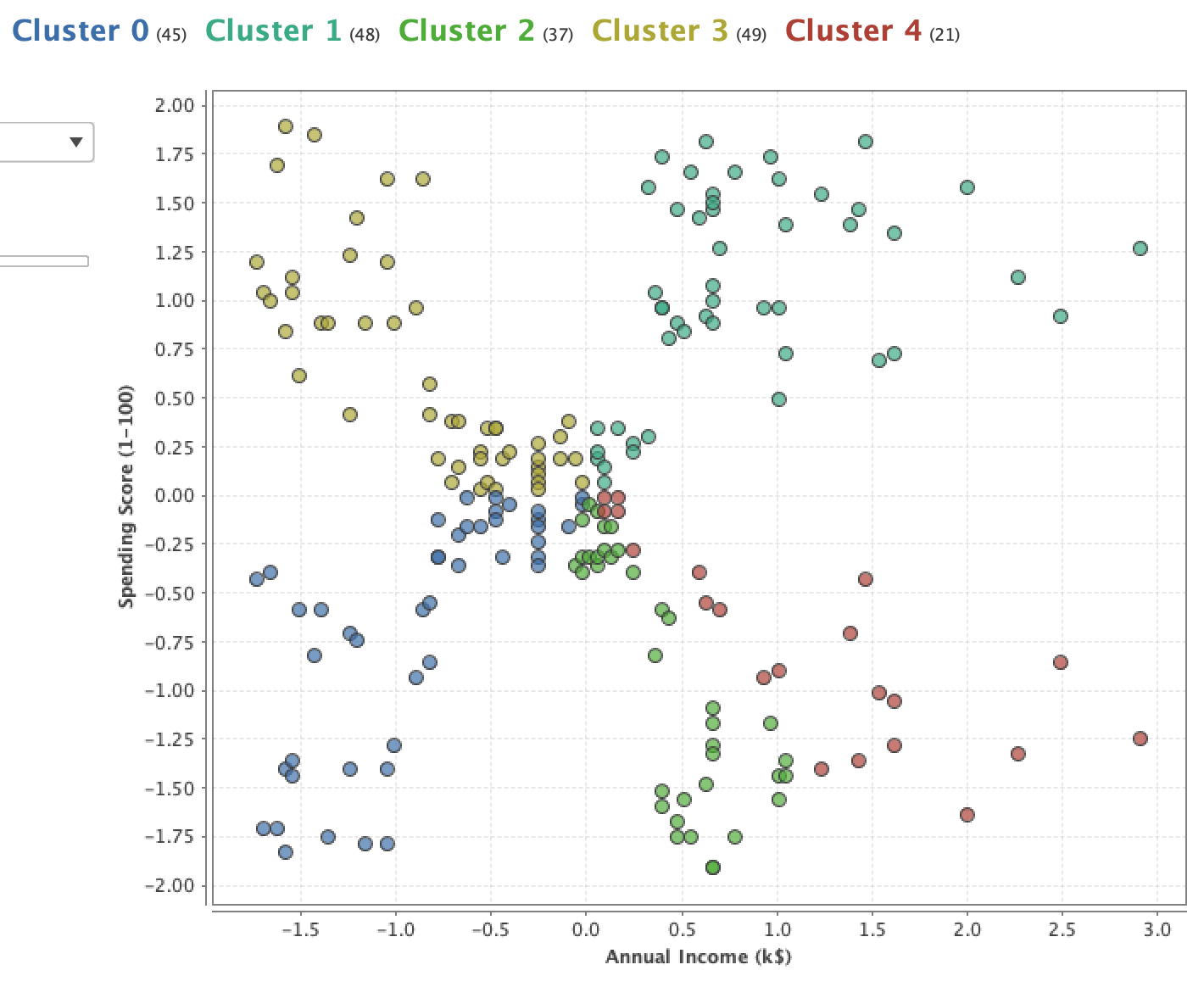
****

Figure 6. Scatter plot of points in 5 clusters (Cosine similarity distance metric, z-transformation).

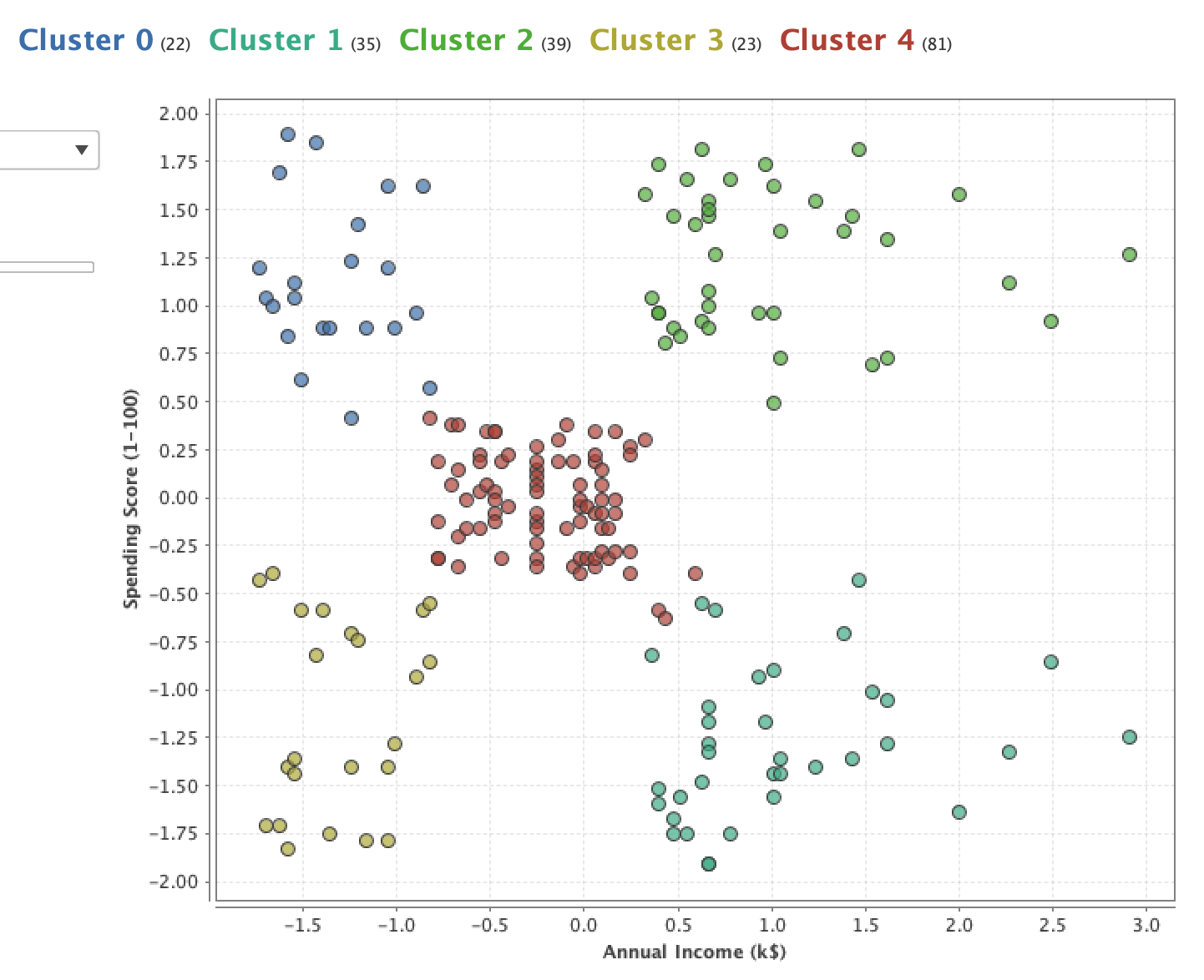
****

Figure 7. Scatter plot of points in 5 clusters (Manhattan distance metric, z-transformation).

**Results: The result of our clustering is a set of 5 clusters that describe different categories of customers. We have customers, which has low income/low spending score, high income/high spending score, high income/low spending score, low income/high spending score, medium income/medium spending score.**

**The obtained clusters can help us, as entrepreneurs, to develop an effective marketing strategy to benefit every customer and the mall itself. For example, we can target a different approach to every audience:**

* **low income/low spending score: give discounts because these people cannot afford to buy things in our mall,**
* **high income/high spending score: send a thank-you note for being an active buyer,**
* **high income/low spending score: do a survey to ask what these people need but cannot buy in our mall,**
* **low income/high spending score: send thank-you discounts,**
* **medium income/medium spending score: action depends on the current economic situation.**

**Therefore, we can spend less money on discounts as we do not give them to wealthy customers, while also appreciating them with the thank-you notes.**

Naïve Bayes Classifier

**Overview of the Process: The classification process is fully developed within Google Sheets. Firstly, we need to solve the problem with continuous variables required for the classifier (thalach and oldpeak). As for Google Sheets, it is quite complicated to build Gaussian Naïve Bayes Classifier (due to difficulties in computations and overall complexity), we define the ranges for these variables based on the distributions obtained on the histograms of this columns. Then, we build tables for prior probabilities of all features (cp, thalach, and oldpeak) and classes (0 or 1). After that, we must obtain conditional probabilities with the help of pivot tables. Subsequently, we define the formula to calculate the posteriors of both classes for each record, compare them to find the highest probability, and assign the final label to the class with it (highest chance). As we have a small dataset, we decided to use all of it for training; however, to train the model, we used the other (similar) dataset with the same columns from the other university.**

**Results of the Analysis: To train the model, we used various combinations of parameters (cp + thalach + oldpeak, cp + thalach, cp + oldpeak). For the best model (with three features), we also checked its accuracy if being used for training set to see that this model has nearly the same accuracy for both training and test datasets, which means it is not overfitting or underfitting.**

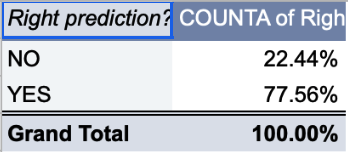
****

Figure 8. Accuracy of 3 features model used on the training dataset. (YES means the predicted label is the same as the right label)

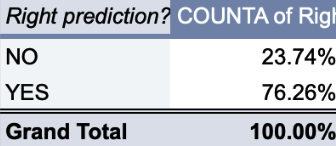
****

Figure 9.Accuracy of 3 features model on the test dataset.

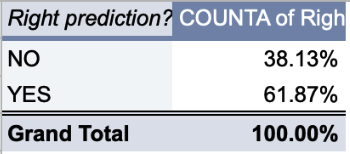
****

Figure 10. Accuracy of cp + oldpeak model on the test dataset.

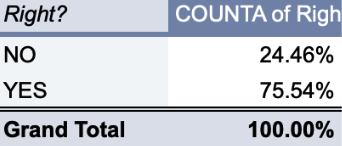
****

Figure 11. Accuracy of cp + thalach model on test dataset.

As we can see from the obtained results, the best model for accuracy is three features (cp + thalach + oldpeak) model. One interesting detail here is that even though we used ranges for continuous variables, we did not lose relevant data as if we do the Naïve Bayes classifier with tools that allow us to work with numerical values (such as Python) efficiently we obtain the accuracy of 80%, which is quite a near number to our result here, in Google Sheets. The accuracy of 77.56% is excellent for Naïve Bayes Classifier. It is rarely used for this kind of task (medical classification) as here we can encounter a lot of other factors affecting the label.

**Selected Features: In the results, we already mentioned that three features model is the best model in terms of accuracy score; therefore, we use cp, thalach, and oldpeak features to predict the label.**

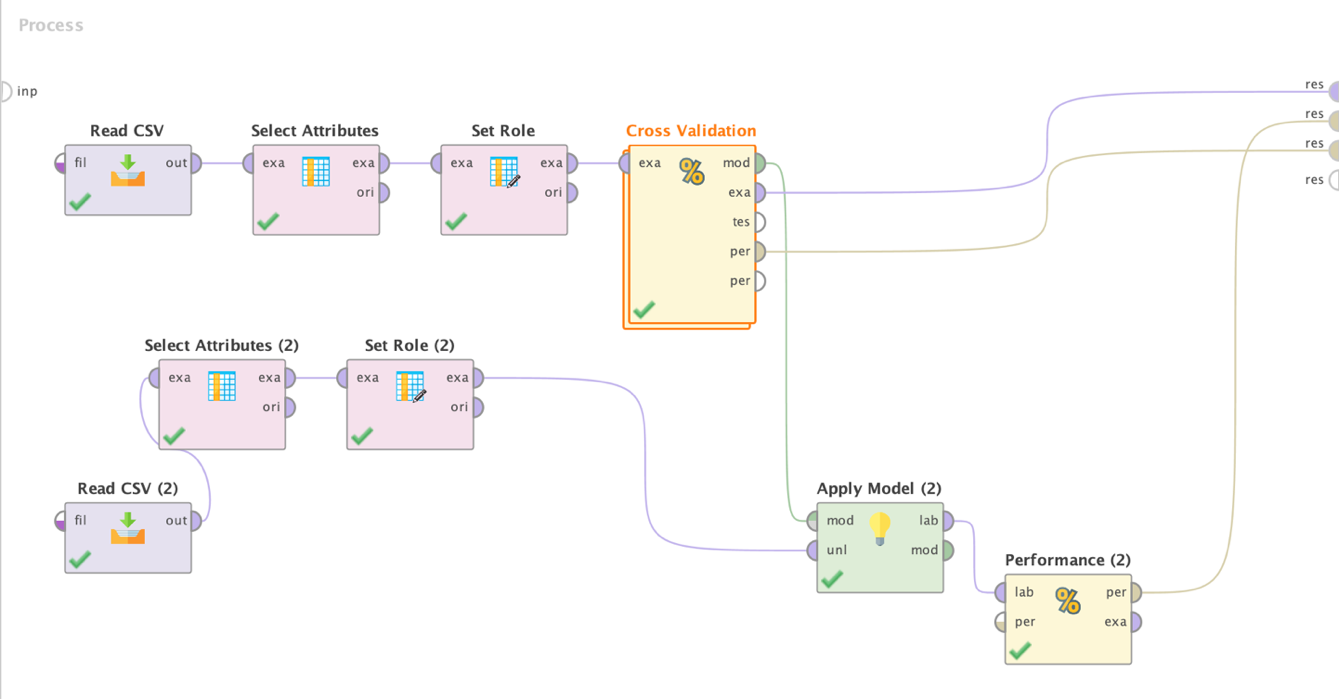
**Rapid Miner Process overview: To look at how different tools can provide us the results of classification, we developed the same classification process in RapidMiner. Even though it is a bit different (due to cross-validation: when we train and test the model multiple times on different parts of the dataset to obtain the model that will be learned using all records yet will be able to analyze in terms of accuracy, which is especially useful for small datasets such as this one (only 303 records) as in cross-validation all records are used for training, so no data is missed), however, we continue to perform additional training with the data obtained in another university. The process looks like this:**

Figure 12. RapidMiner Process for Naive Bayes Classification.

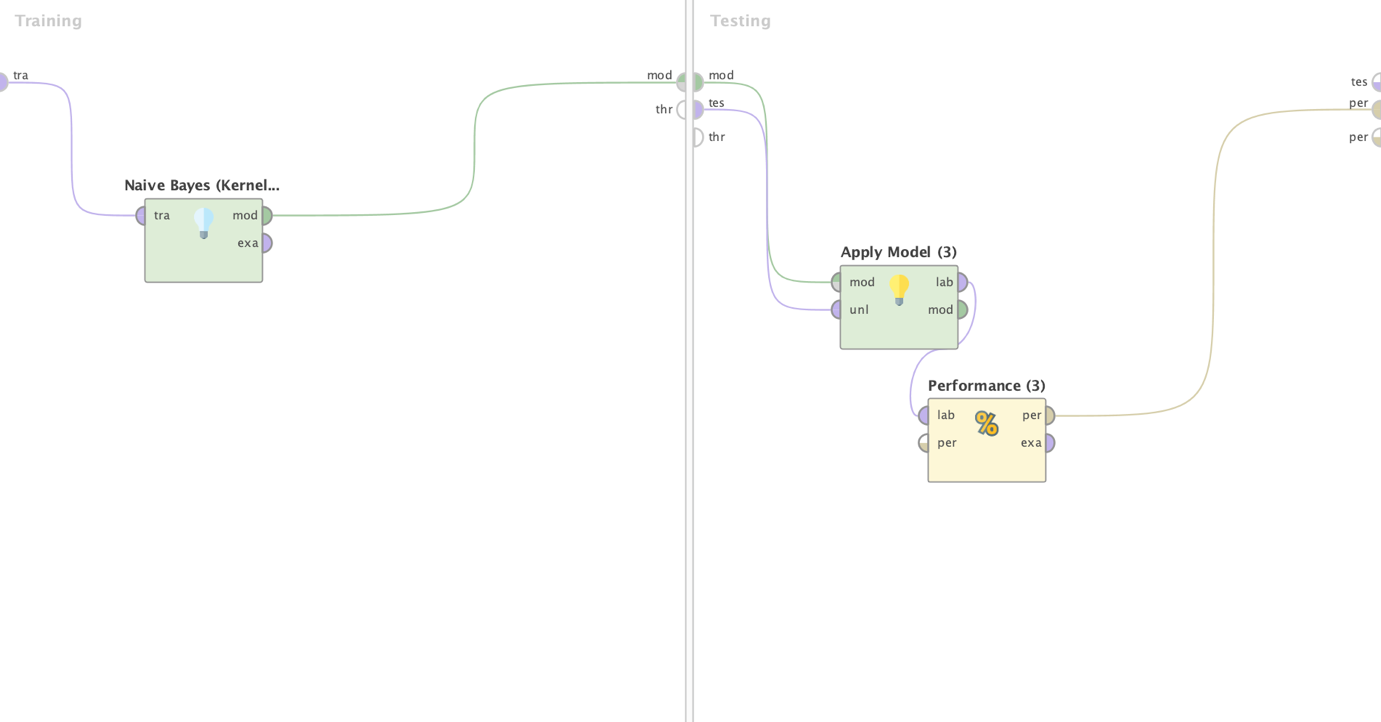


Figure 13. The Part of RapidMiner Process in Cross-Validation Block.

**Comparison of RapidMiner Process to Google Sheets Process: The accuracy of the RapidMiner process on the test data is 77,7%, which is slightly higher than the accuracy of the Google Sheets' model. It is most possibly related to the cross-validation used in the RapidMiner process as cross-validation typically rases the accuracy by a couple per cents.**

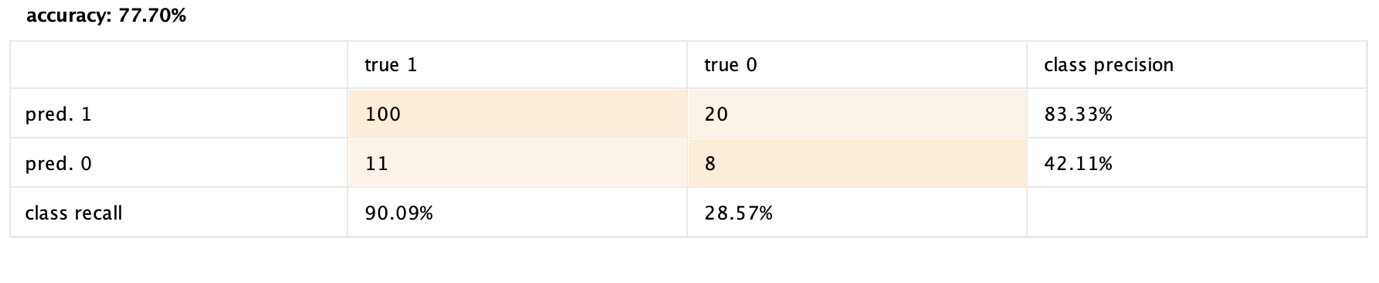
****

Figure 14. Accuracy of the model developed in the RapidMiner on the test data.

The conclusion here is that both tools can be used to perform Naïve Bayes Classification; however, the RapidMiner process is much easier to set up than to create Google Sheets formula.