**Purpose**:

This is to illustrate how to do CP3403/CP5634 data mining assignment with an example.

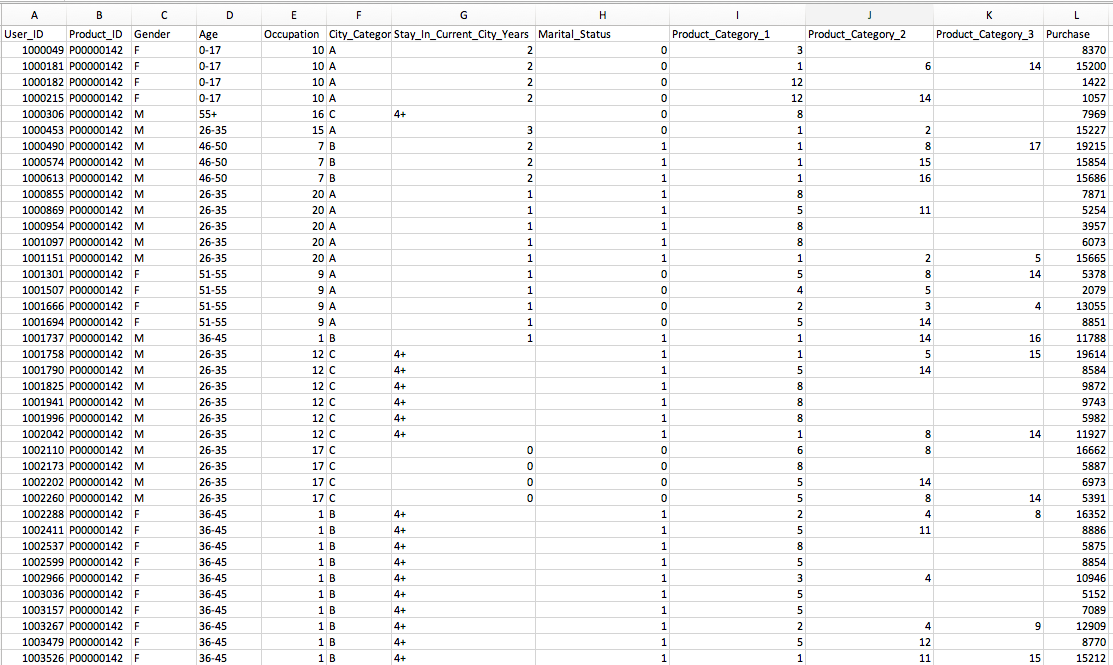
**Dataset**: BackFriday.csv (https://www.kaggle.com/mehdidag/black-friday)

The dataset is a sample of the transactions made in a retail store. The store wants to know better the customer purchase behaviour against different products.

Dataset of 550 000 observations about the black Friday in a retail store, it contains different kinds of variables either numerical or categorical. It contains missing values.

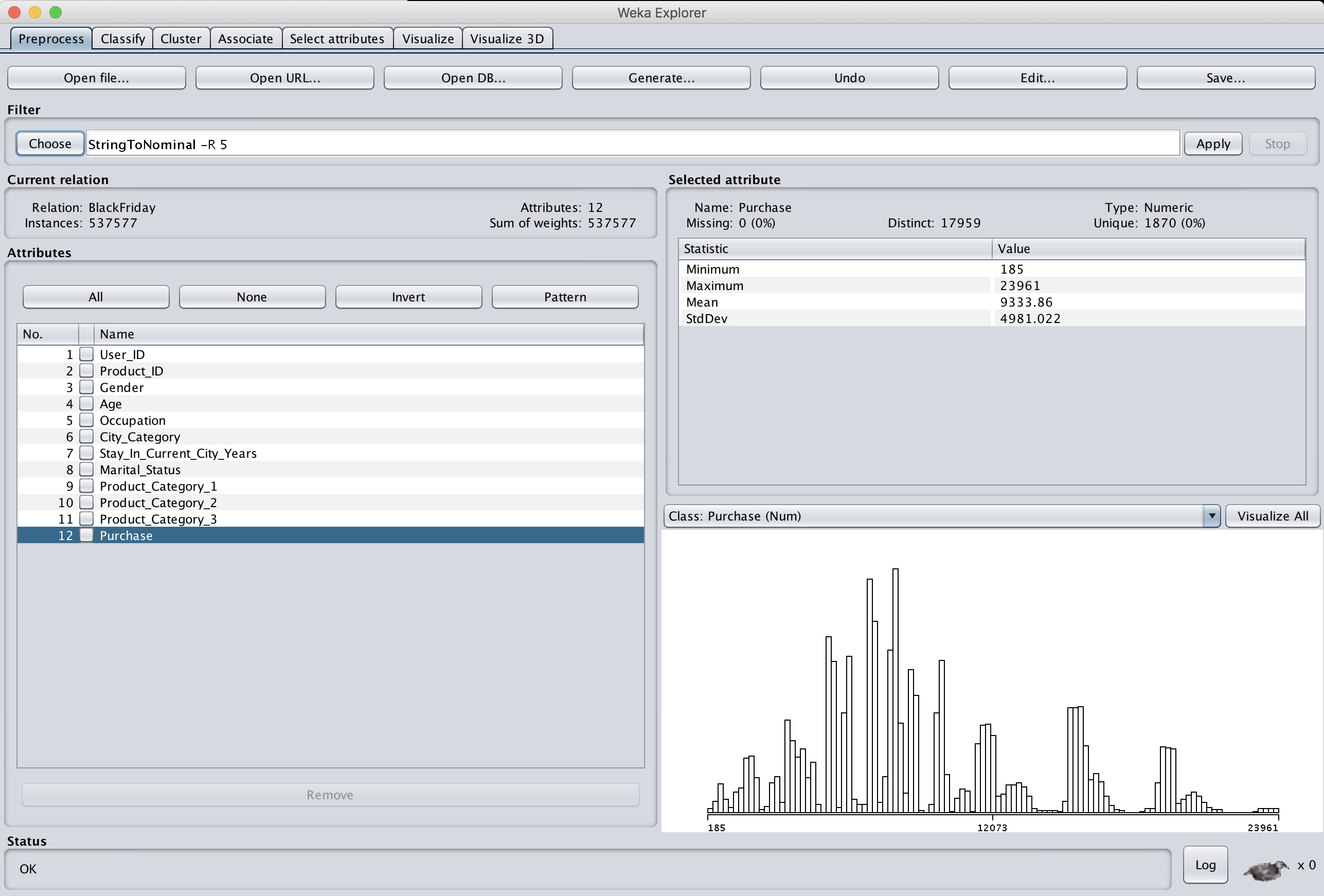
Columns are: User\_ID, Product\_ID, Gender, Age, Occupation, City-Category, Stay\_In\_Current\_City\_Years, Marital\_Status, Product\_Category\_1, Product\_Category\_2, Product\_Category\_3, Purchase (in dollars)

Some attributes are nominal, some are numeric, and one is a string type.



**WEKA**:

Once you load the data into Weka, it will look like below. It has 537,577 distinct instances, 5891 distinct users, two gender types, 7 age groups, 21 occupation types, 3 city categories, 5 distinct stay\_in\_current\_city\_years, 2 marital status, 3 attributes for product categories, and purchase in dollar.



**Before you start:**

The first thing you have to do before mining data, is to understand what the dataset is about. What each attribute is about, and its data type and distribution etc. Once you fully understand the dataset then you can build some scenarios (what patterns you are interested in). So please spend some time to understand what the dataset is about.

**Preprocessing**:

Since the dataset includes various data types, not all data mining tools are active and not directly applicable to the dataset. Instead, it needs preprocessing before data mining.

Well, you can decide what kind of preprocessing this data needs, but it depends on what kind of data mining you want to do, what patterns you would like to mine. So even though two students use the same dataset for the assignment: one is doing clustering whilst the other is doing association rules mining, they will end up with different preprocessing processes!

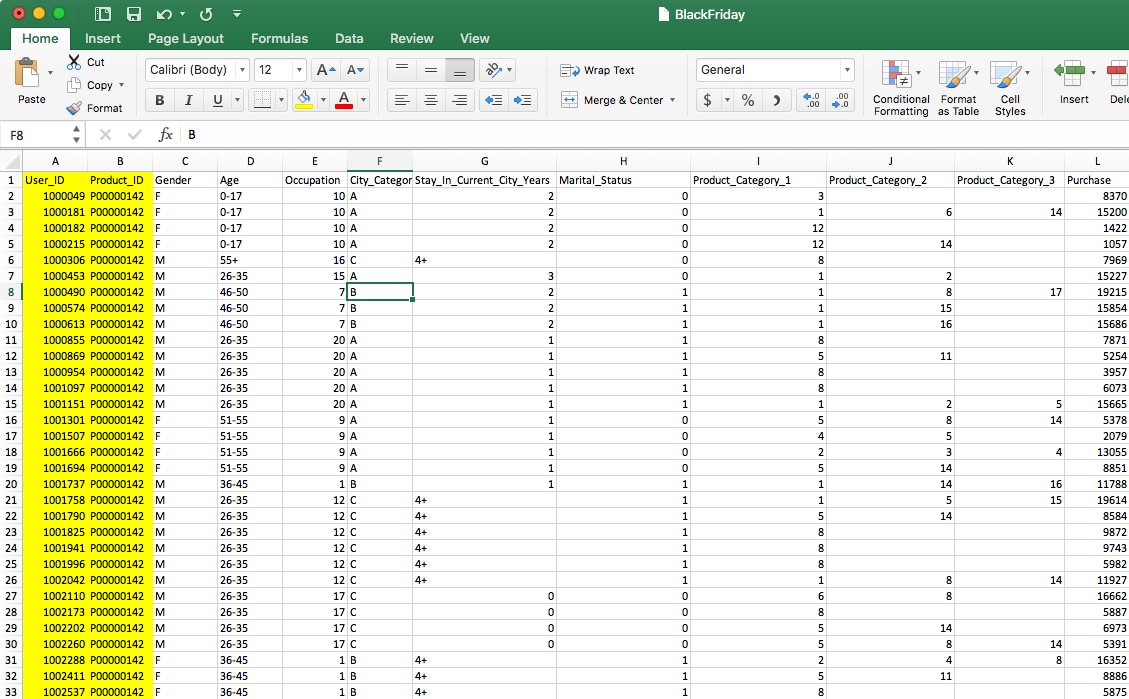
**Possible Tasks**:

Now let’s examine what data mining we could do with this particular dataset. Actually, you can do various data mining tasks with this dataset, but here we examine few options.

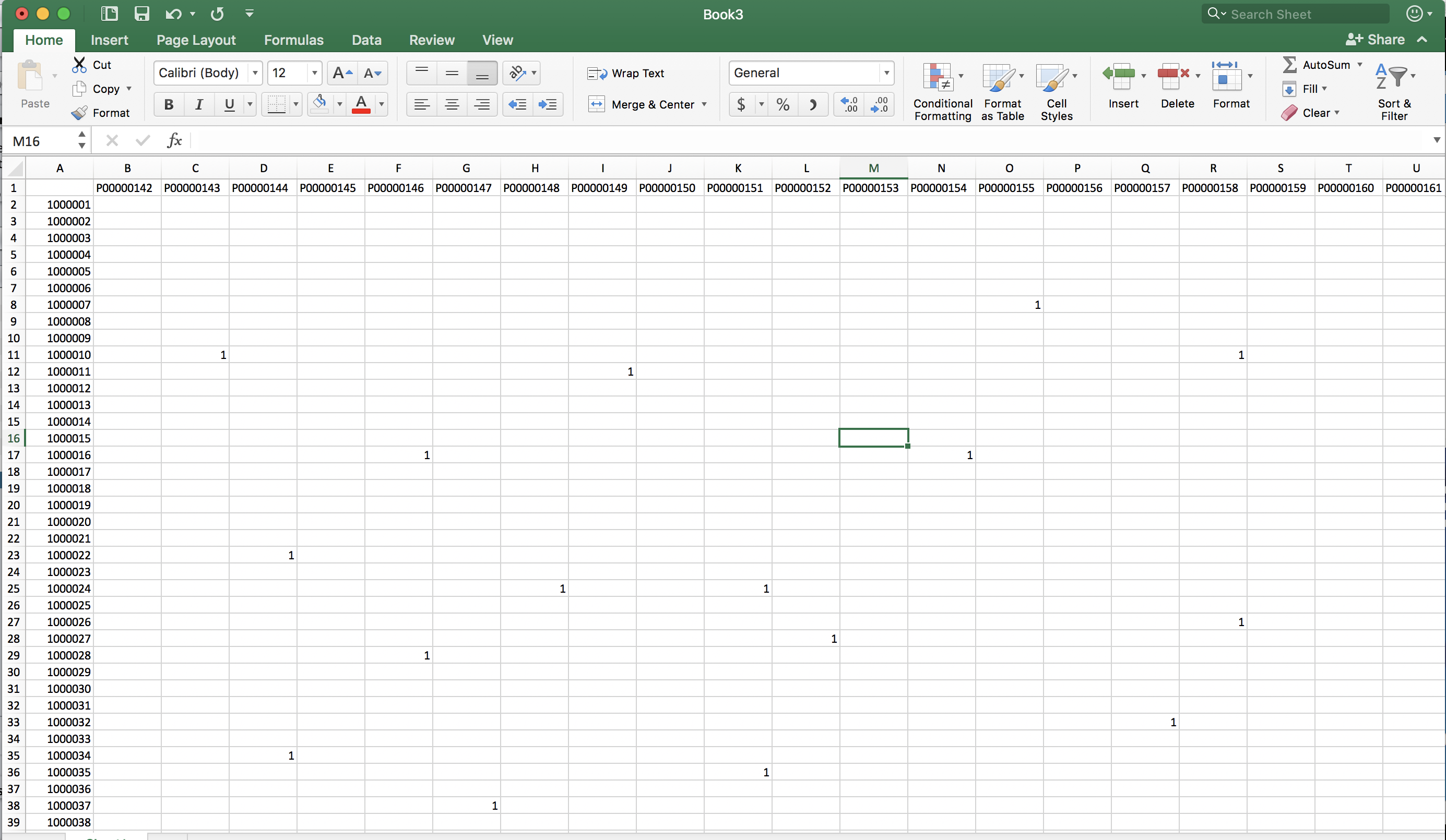
* Association Rules Mining

1. First of all, you could be interested in finding associations among items (products) since this dataset is about retail purchase transactions, and it would be interesting to find some associative patterns such as: customers who purchase ‘P0000142’ and ‘P0000152’, also purchase ‘P0000213’ with 90% of confidence. This kind of pattern could be used for promoting sales, marketing, etc.

In this case, we are only interested in the first two columns.



Now, you need preprocessing to transform these two columns to a transactional table for association rules mining something like below.

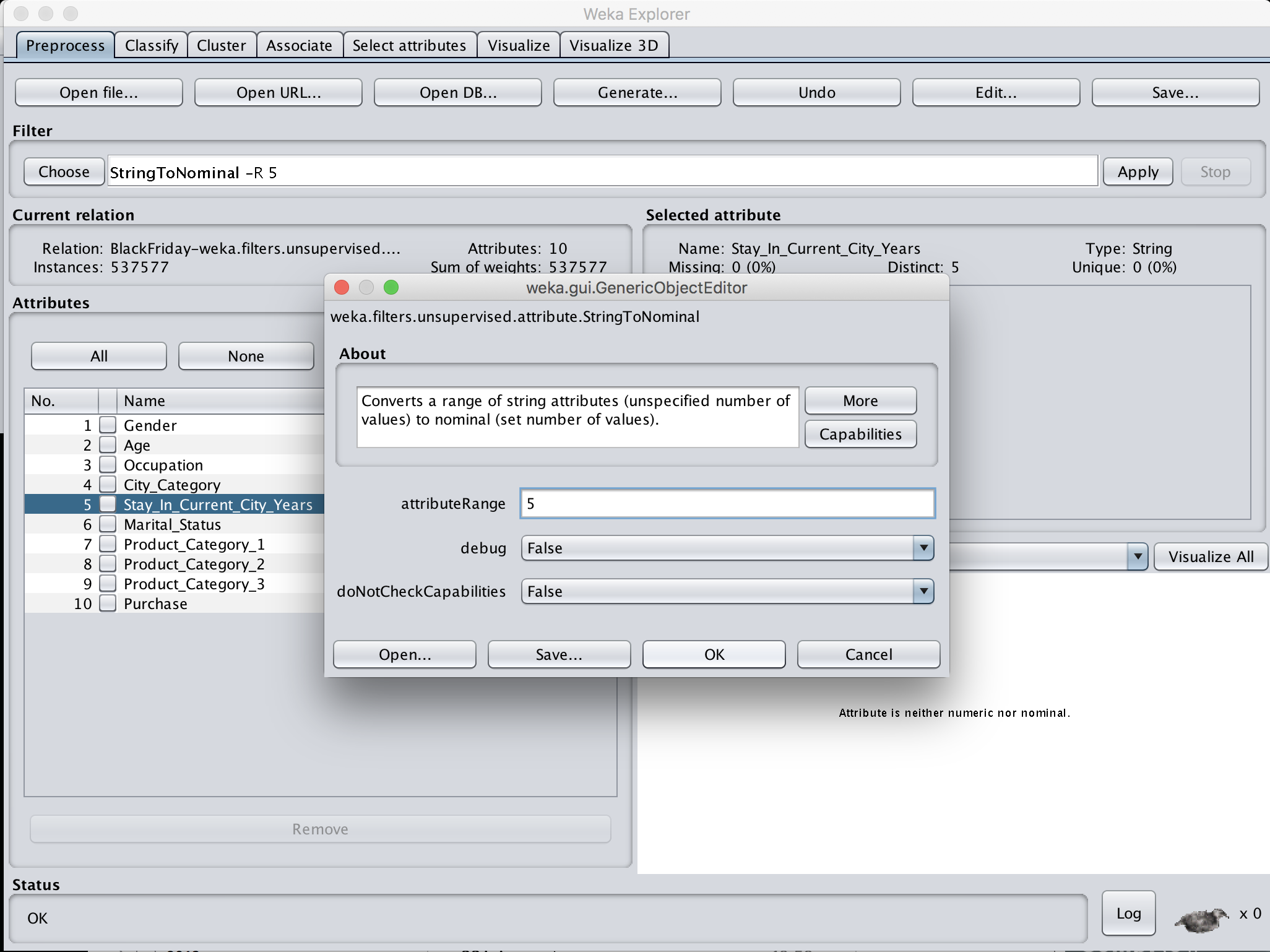


where columns are for products whilst rows are for users. Also, “1” represents that the user purchased the product. (You have to figure this out how to transform the original data to this format. Weka does not do this for you automatically, you need to use some other ways to transform such as your own Python code, spreadsheet VisualBasic, or some other ways…).

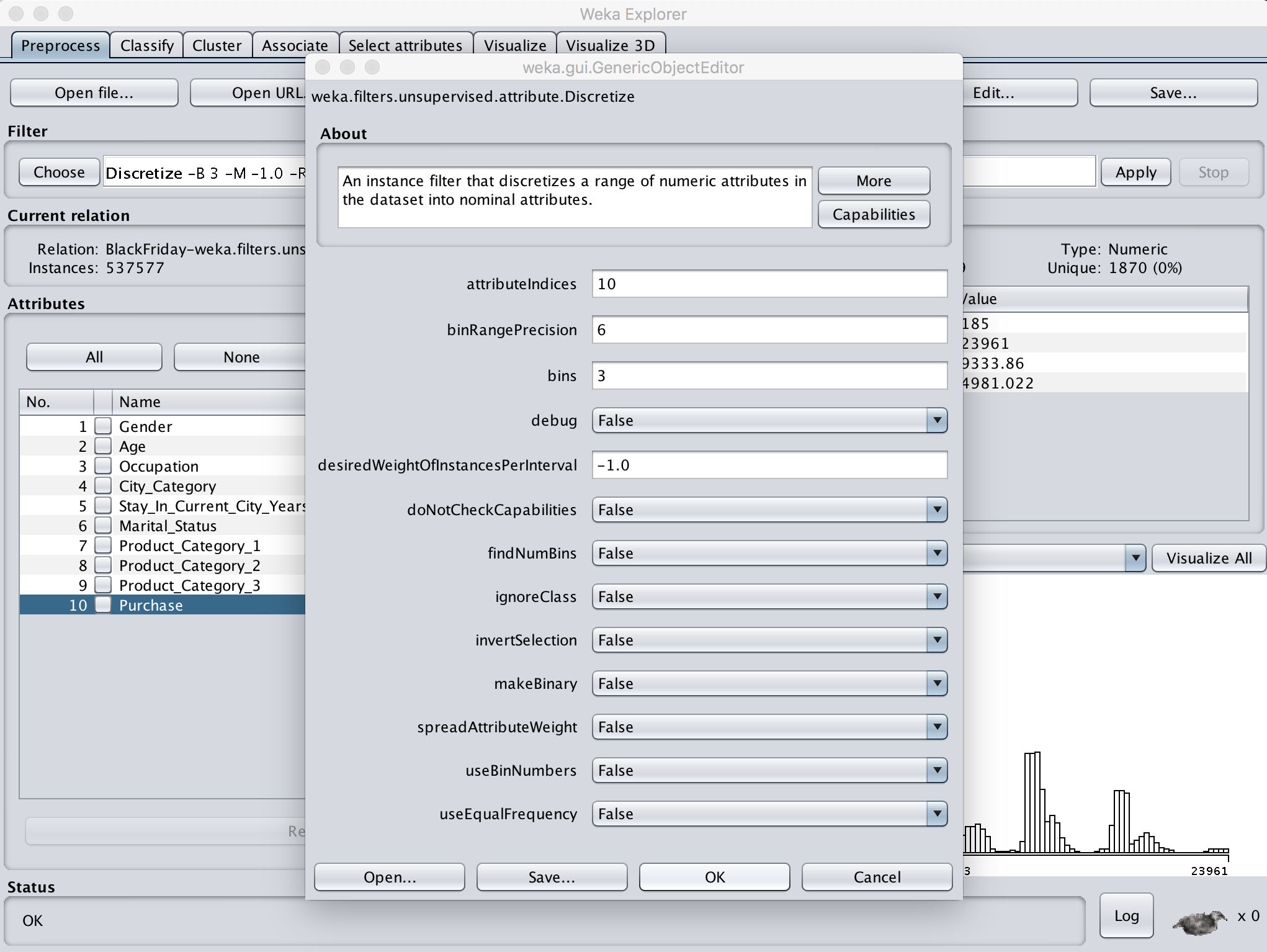
Once you have this transactional table, then you should be able to find some associative patterns using Apriori in Weka, and you can report some interesting associative patterns and describe your preprocessing in detail).

1. Since the price is about the total dollars the user spend, you might be interested in what are associative patterns for this price. As a manager of this store, you would like to find out what are associative patterns for high spendings. You can find associative patterns for the price class. For instance, Marital\_Status = F and Occupation = 2 implies Purchase dollars = High with 90% of confidence.

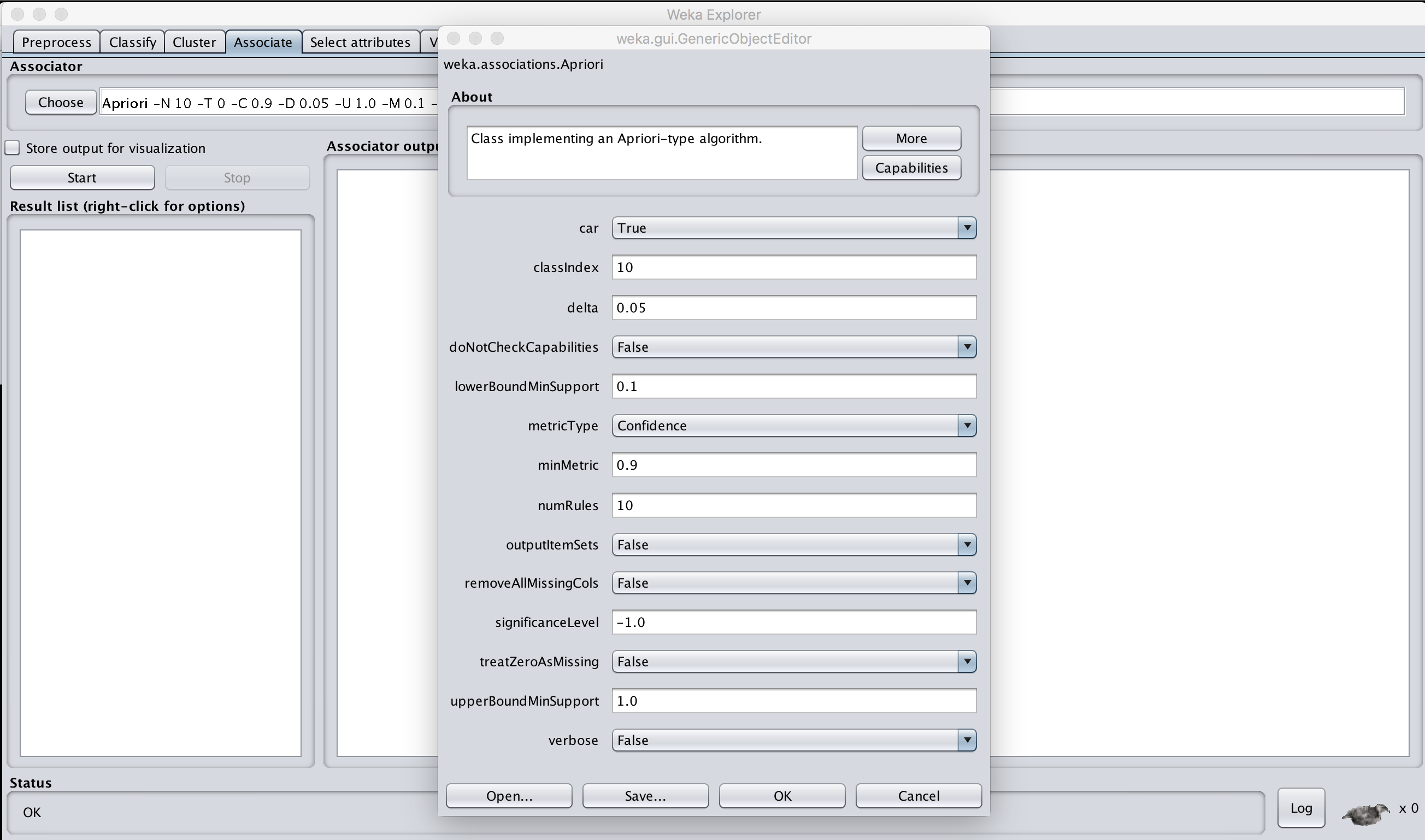
* To do this, you need to load the raw dataset into Weka.
* In this case, you are not interested in User\_ID and Product\_ID, so remove them.
* As the raw dataset contains numeric attributes and also a string type, apriori is not directly applicable but inactive.
* Convert numeric data types (Occupation, Marital Status, Product\_Category\_1, Product\_Category\_2, Product\_Category\_3) to nominal using NumericToNominal filter.
* Convert string data type (Stay\_In\_Current\_City\_Years) to nominal using StringToNominal filter.



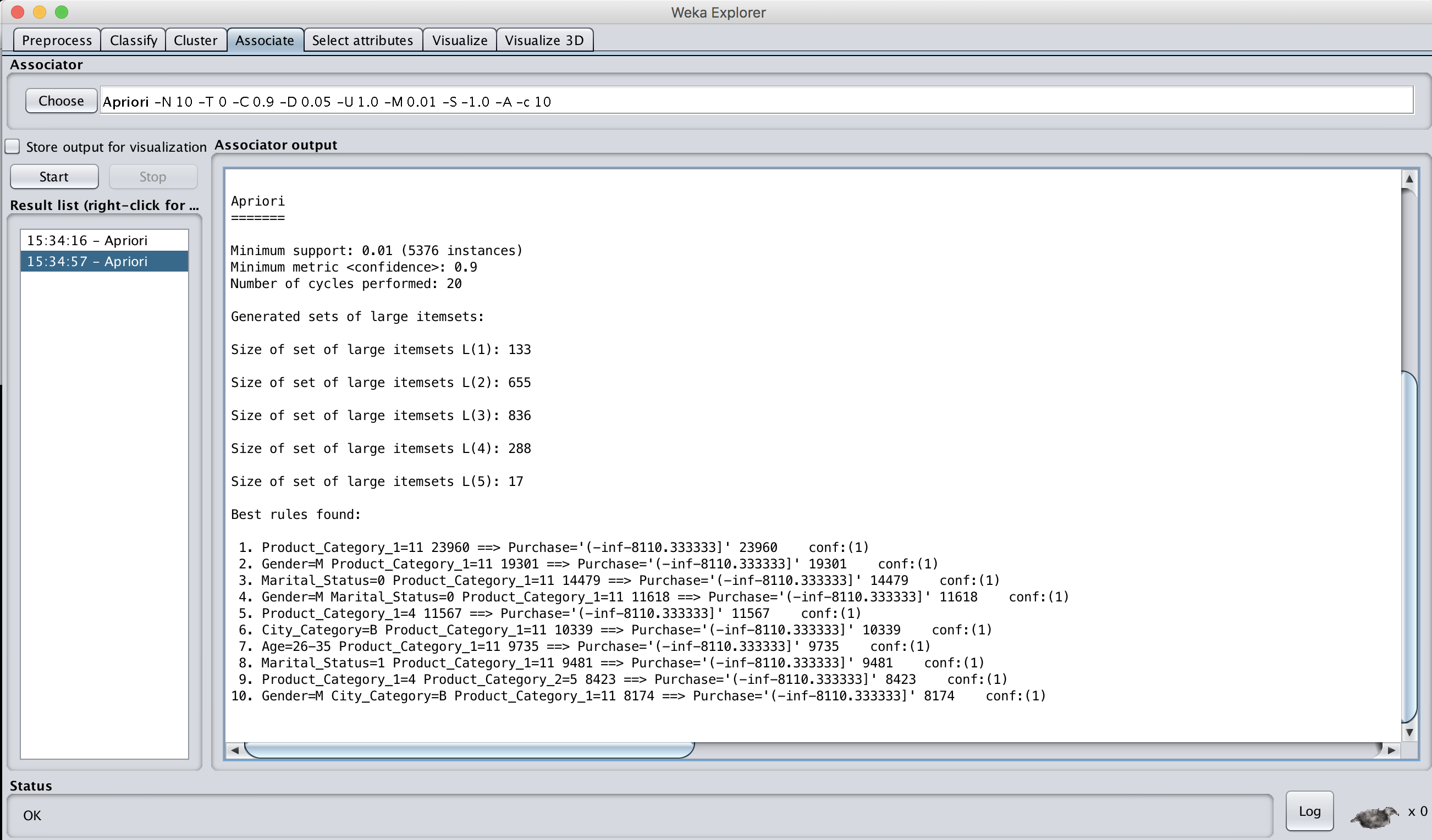
* + Now all the attributes except Purchase are nominal.
  + Now you have to decide how to discretise Purchase attribute for your analysis. Well, in this case, let’s divide Purchase (in dollars) into three groups: low spending; medium spending; high spending.
  + So choose Discretize filter with 3 bins and apply it to Purchase.



* + Now, Purchase has three groups: -8110: low spending; 8110-16035: medium spending; 16035-; high spending).
  + Now Apriori is active and applicable to this preprocessed dataset.
  + Now click Apriori and “car=True” with “classIndex=10” to find associative patterns for Purchase.

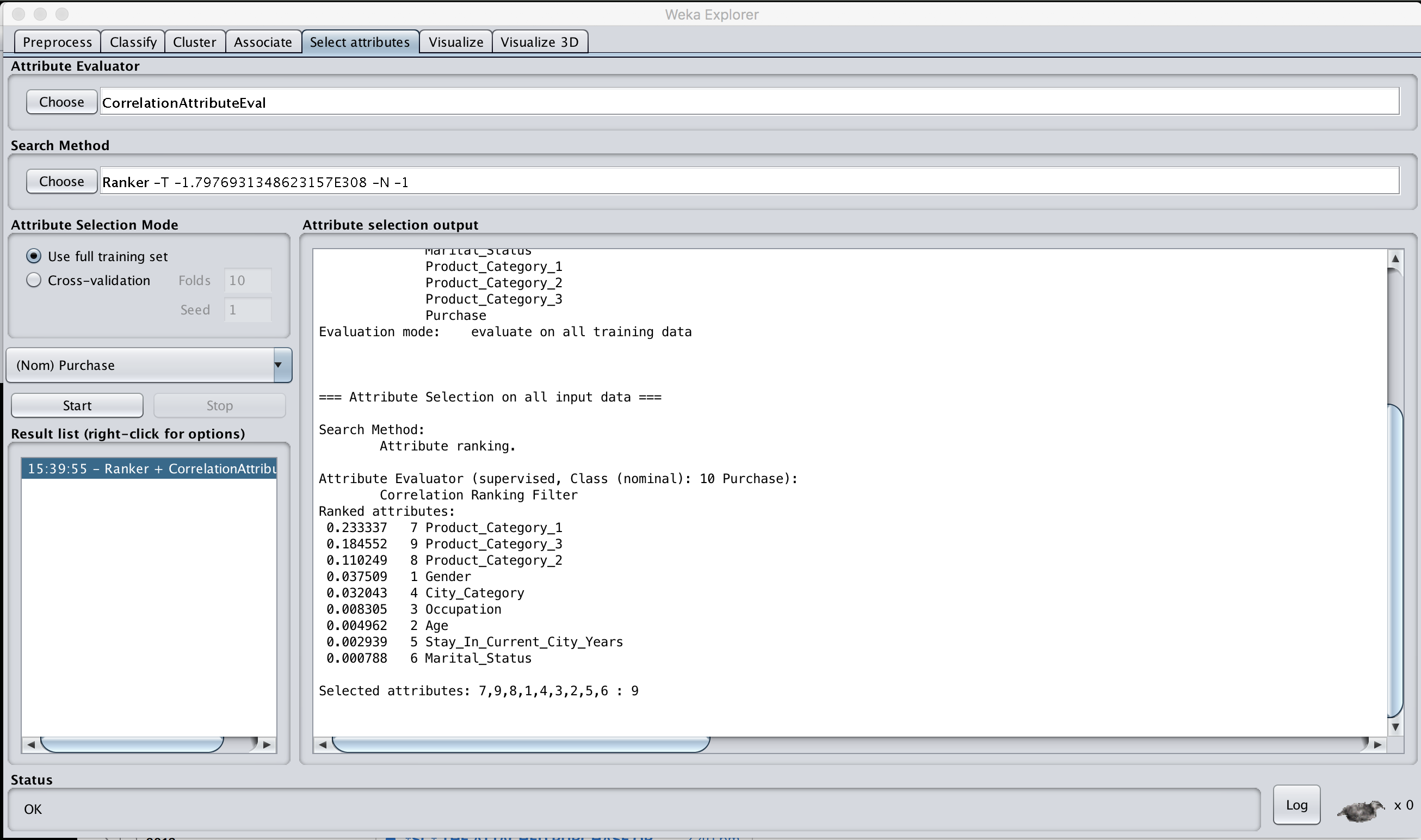


* + You can now explore with different minSupport to find association rules.



* Correlation

In some cases, you might be interested in correlation analysis in particular w.r.t. Purchase. Let’s imagine that you do the same preprocessing steps we did for Association Rules Mining. Then, go to “Select attributes” tab, and choose “CorrelationAttributeEval” which analyses correlation.



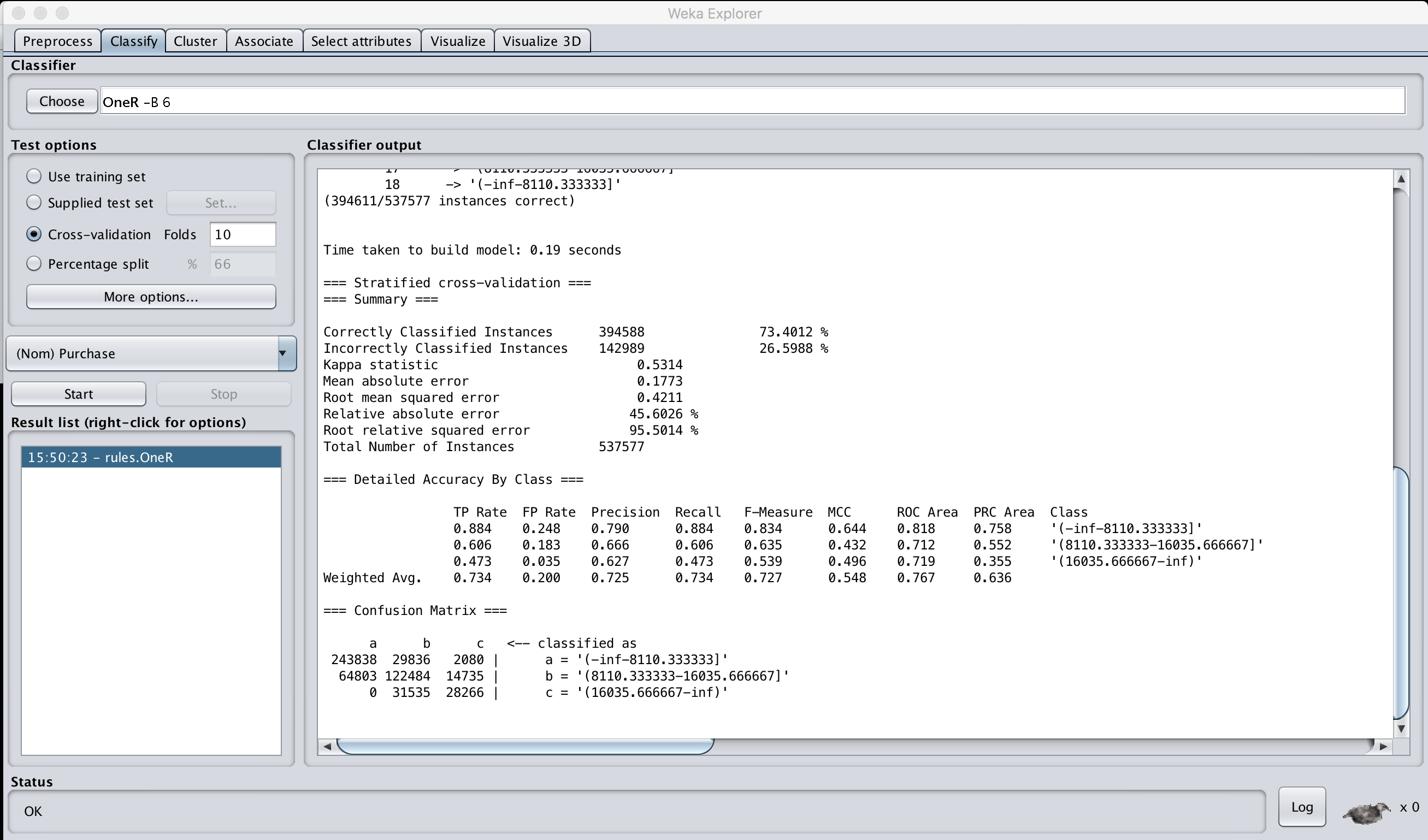
The result shows that Product\_Category\_1 is the most correlated to Purchase, whilst Marital\_Status is least correlated to Purchase. That is, whether or not a customer is married is not indicative or is not correlated to dollars he/she spends.

* Classification (Regression)

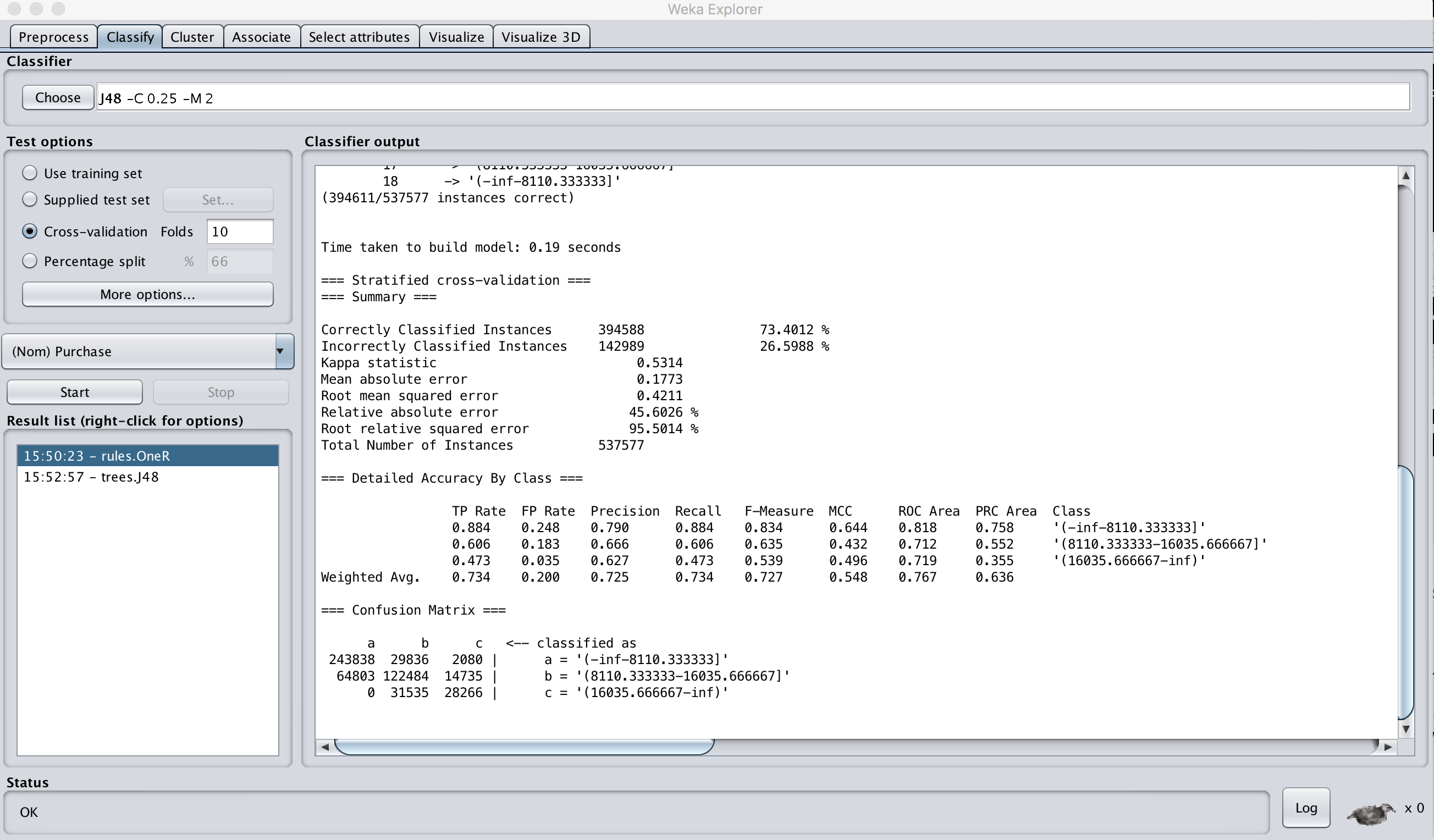
You want to predict/classify customers to one of low/medium/high spending customers. Please note that you can have more Purchase classes if you want, such as 5 groups: very low, low, medium, high, very high.

You can apply various classification methods to this preprocessed data.

OneR has around 73% accuracy with 0.19 second as below:



Decision tree (J48) has around 73% accuracy with 33 secs. Thus, no point using this slow J48 over OneR.



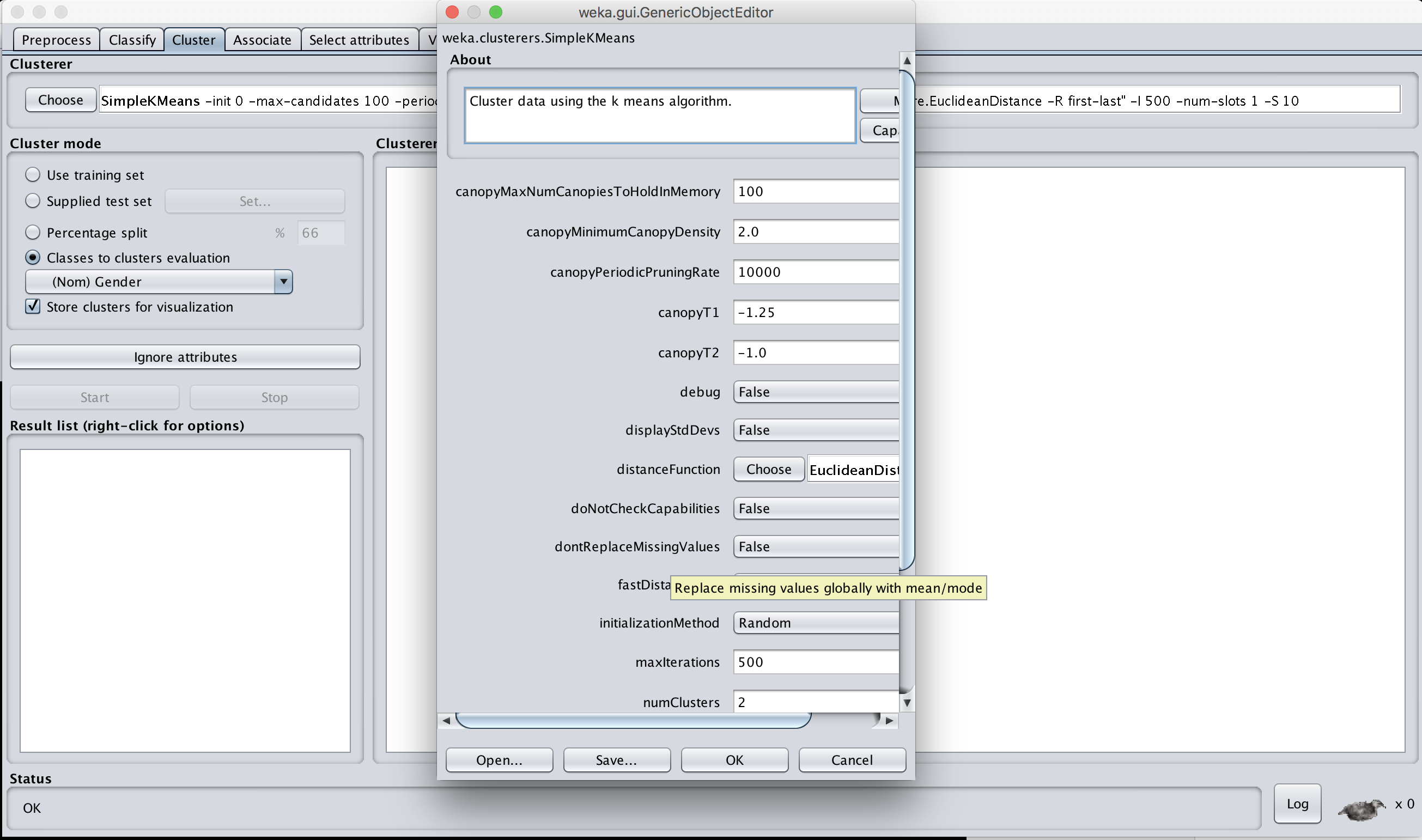
You can apply NaiveBayes, MultilayerPerceptron (Neural Network), SMO (Support Vector Machine), IBK (k nearest neighbour).

You can do some classification tasks as illustrated above (might be also with different number of class labels) for your assignment.

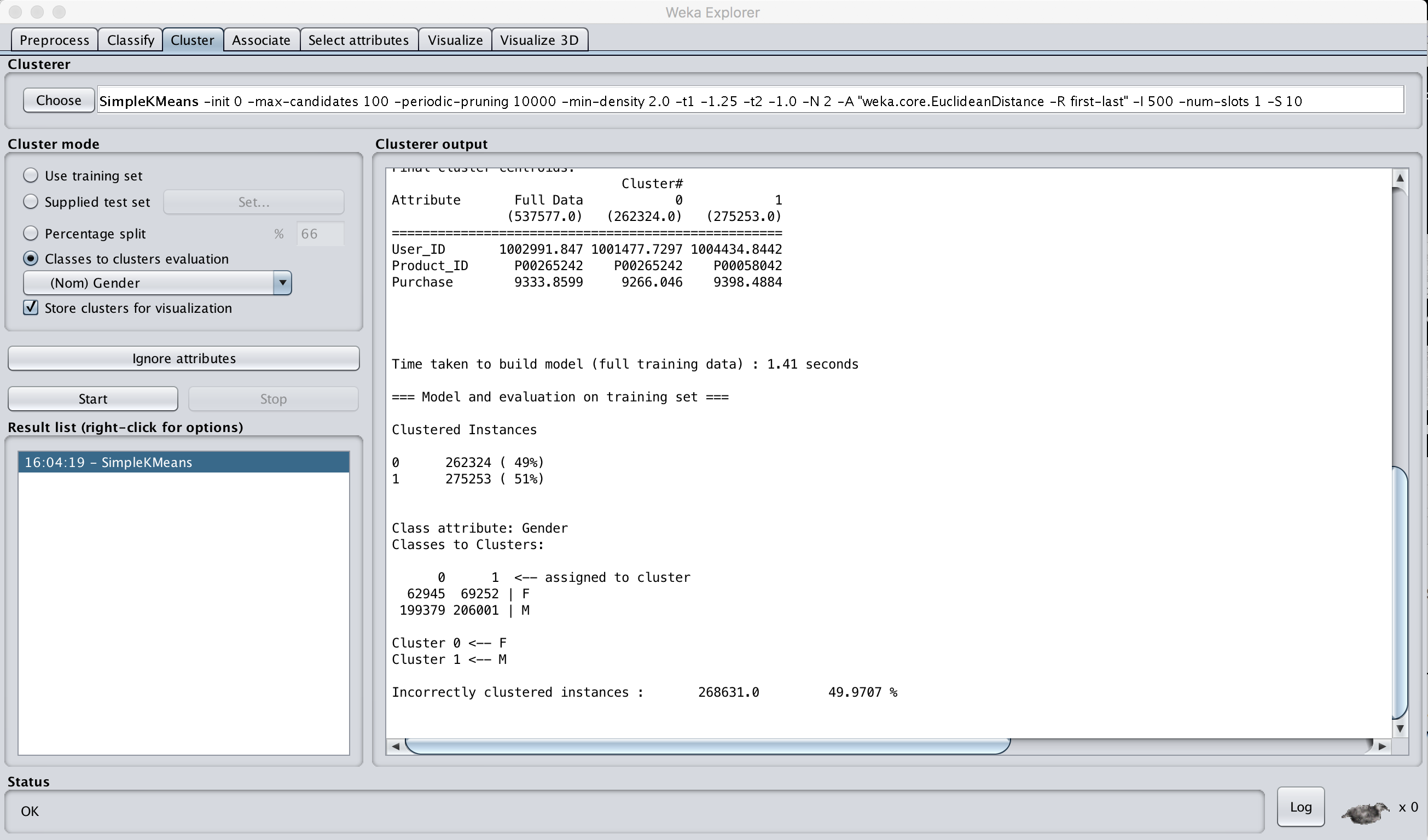
* Clustering

Also, you can do various clustering tasks to find some interesting groups.

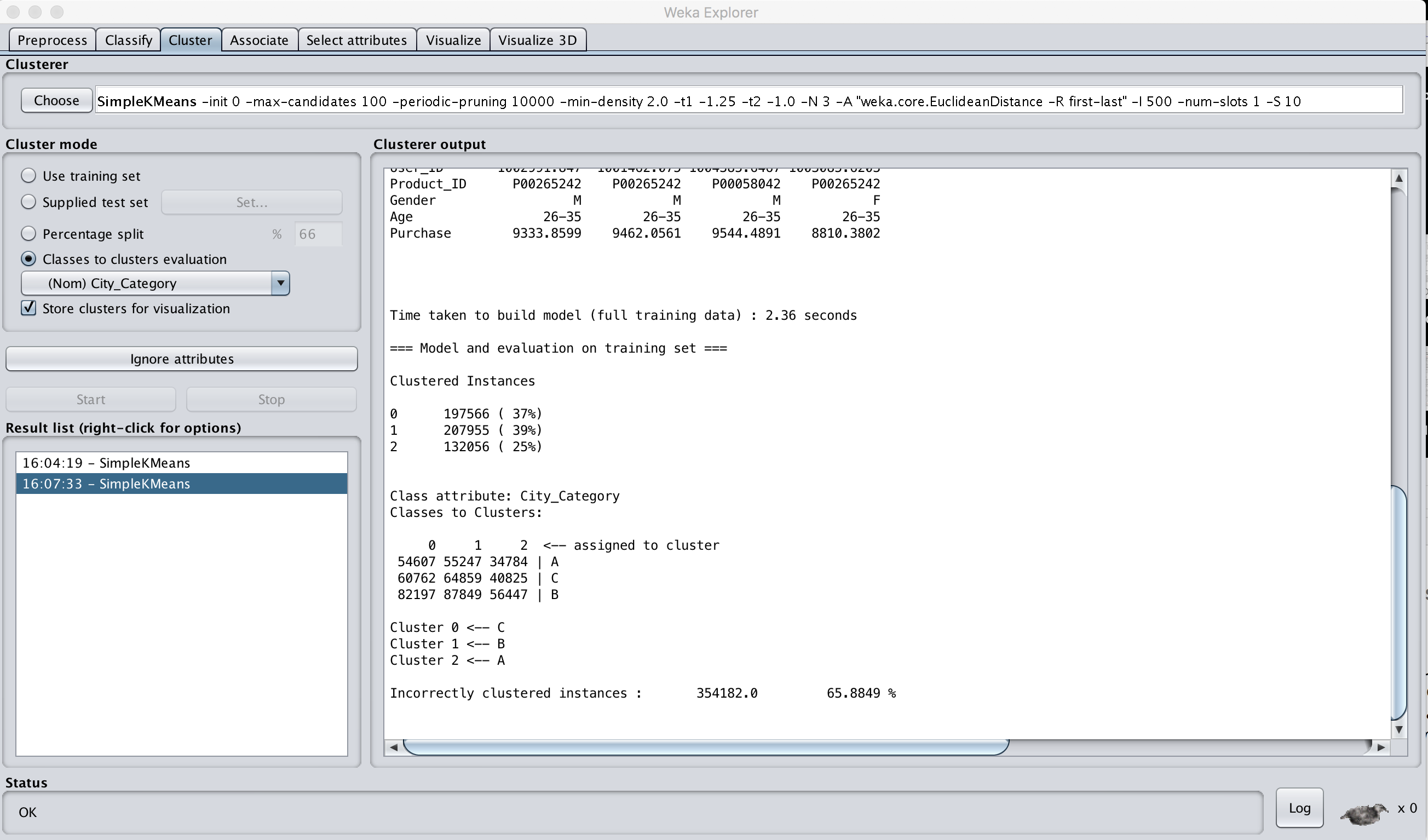
* + Load the raw dataset into Weka.
  + Since Purchase (in dollar) is numeric, you might want to analyse Purchase to see any meaning groups you can find with it, and compare clustered Purchase groups with Gender.



* + Apply k-means with (k=2 since the target attribute Gender has two groups), and evaluate the clustering result with Gender.



* + Now you have around 49% incorrect and 51% correct. This could mean that spending in dollars (Purchase) is not coincided with Gender types.
  + This time compare the clustering with City\_Category (3 types). So apply k-means (k=3) to Purchase and compare the result with City\_Category.



* + Now, you have 65% matches which is better than Gender. This could mean that City categories are more related to Purchase than Gender is to.