Clustering report

Exploratory tasks

First of all, before going further into the step of the clusterization it is important to analyze every campaign and group of actors for this campaign. So the first step here is to identify those campaign and those group of features which make a campaign existing. We used the GroupBy node to group the table by a feature (column) and take action on it like count the number of each group, mean of the activation in the group, etc...

Table "default" - Rows: 7 | Spec - Columns: 4 | Properties | Flow Variables

Row ID	S type	D activati...	D profit (...	I activati...
Row0	1:st & 2:nd line	0.027	0.255	809
Row1	1:st line	0.026	0.205	4465
Row2	3f2	0.002	-0.028	48
Row3	all order	0.019	0.13	1792
Row4	check	0.019	0.224	1643
Row5	kombination	0.032	0.23	1201
Row6	stege	0.007	-0.192	42

These steps actually make us analyze which campaign sells out the more orders or which one as a better impact on the profit or the other way around, which one was not great and can perhaps be improved.

The next step was to filter every group with less than 50 orders. For this one, I used, as said in the subject, a Rule-Based Row Filter. This node permits us to filter every row who doesn't match with the rules defined in it. I personally choose to do it on the output of the GroupBy node AND on the table of the order. So, for some rules which are very long to write down in the rule-based row filter because they had many features and many things to check at the same time, I used a joiner. The joiner, as he is called, joins two tables and acts exactly like a SQL request with LEFT JOIN or RIGHT JOIN. To be more precise I used the output of the GroupBy to filter the order table with a left join or right join depend on the position input of the tables.
Now that I have filtered the table with the group less than 50 orders, which we consider less impactful it is important to look at the other group by and to check if there is some anomaly in it that we can delete as well. I re-transcript all the rules described in the description of the subject in a rules-based row filter so I can eliminate every order who doesn't fulfill them and it turns out that they were some mistakes.

Table "HistoricalOrdersProfitActivation.xlsx [default]" - Rows: 9828 | Spec - Columns: 33 | Properties | Flow Variables

Row ID	I Year	I Quarter	S Month	I Week o...	I Day of ...	S Day of ...	I D
Row0	2014	1	mars	10	7	fredag	66

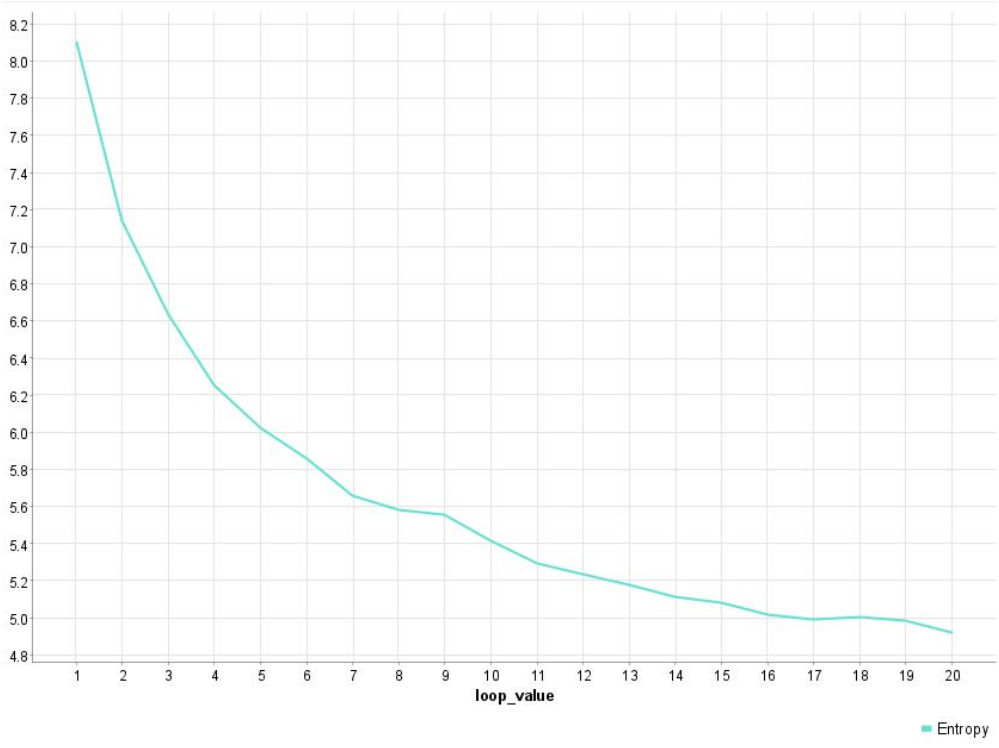
For the last question of the description tasks,

Predictive techniques

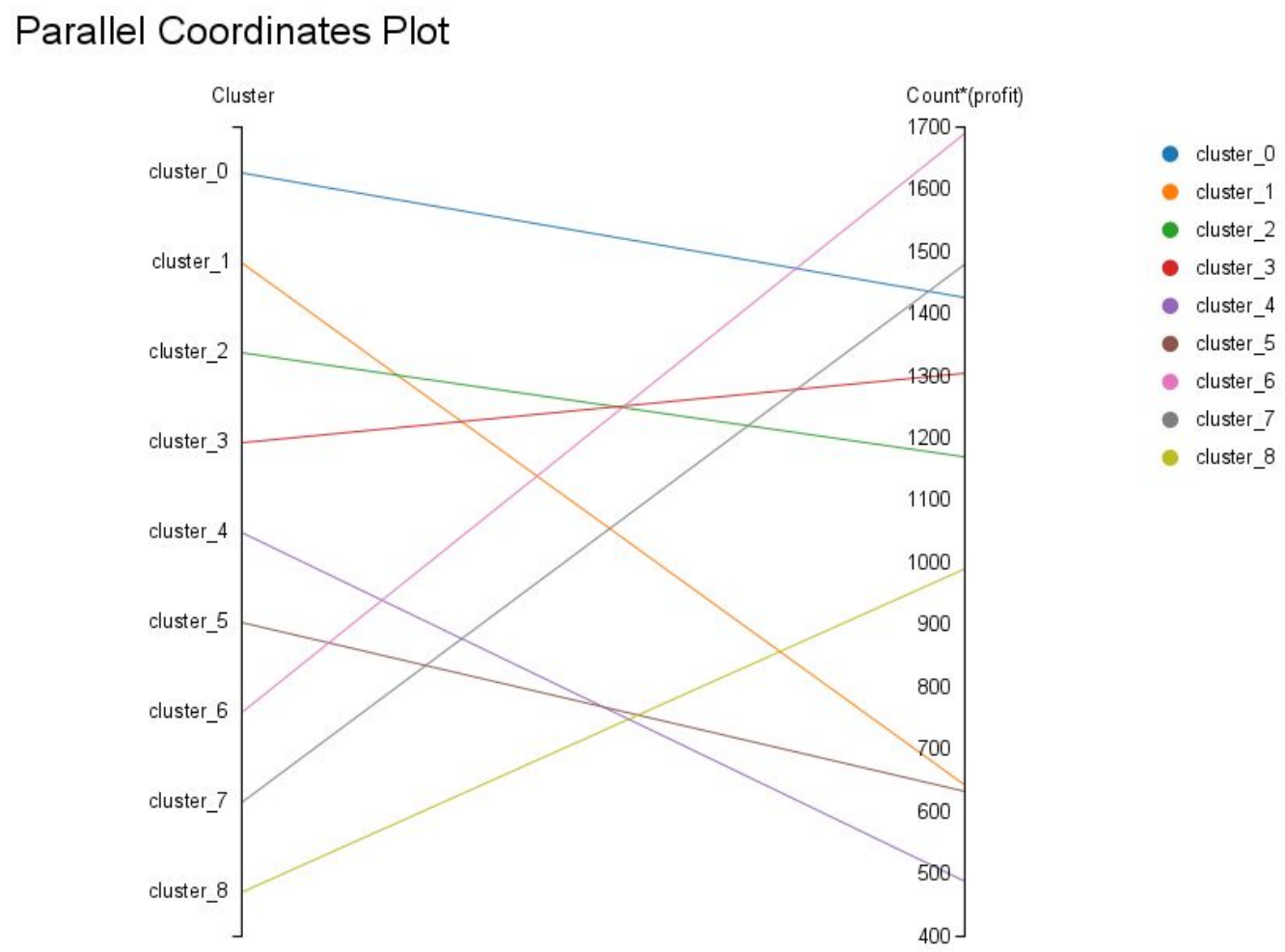
For the predictive part, the importance was not on the prediction but more on how he gets to the point. In my workflow, I added the predictor and the X-Partitioner but I could have just called the Simple Regression Tree learner or the Decision Tree Learner directly after my column filter and see the tree result of that. I add them because with the X-Partitioner I'm sure the result is not on only one part of the data but a set of many parts of the data which makes it more trustful for the next steps. So, after I run them all I compare the result of both regression trees and it appears that they have common features but they didn't use them in the same order, it helps us to know which features are interesting for the decision making of the activation and the profit so it reveals that they have cross path on the decision making.
For the decision tree, I found it more logical in his decision because at first, he uses the reduce price feature and after the month feature. This is logic in some way because I made a prediction on the combined value of activation and profit so for every month there is a different profit and activation so use the month, in the end, is totally normal. I would have used it automatically if I had to compare the profit and activation and make some kind of a prediction.

Descriptive techniques

To resume the question for these tasks we wanted to evaluate the score of the entropy of the cluster we define above. First of all, what is entropy? I struggle a little bit to understand that but I finally get that the entropy it's a measure to evaluate the disorder in the dataset given. Which basically means that we want to compare the disparity in the cluster. So for the K-means, we obtain a curve corresponding to that:

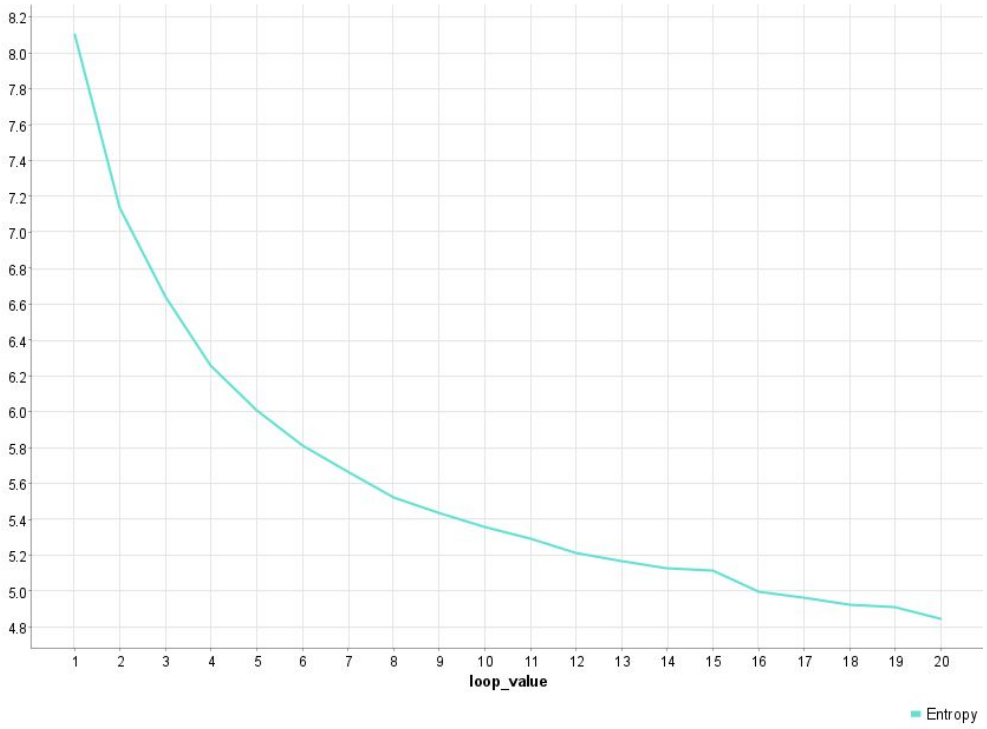


Just for the record, I do not totally agree with the subject where you said that the 8th change corresponds to the sharper bend in the curve. For me, it's the 9th but they are really close so I guess it doesn't change anything really.
The parallel coordinates plot give us this output:



I add a count on the profit but you can see on the submitted workflow that I add a sum on the profit as well. I found interesting to compare more data depending on each cluster.

For the same iteration but with another clustering algorithm, I choose the fuzzy c-means and the result are very different. This time we obtain the 15th change is the sharper bend in the curve, as see below.



Parallel coordinates plot:
Same as above.