Multi Agent Restaurant

2nd Delivery, AIAD 2019/2020
4MIEIC03, Group 35

Eduardo Silva - up201603135@fe.up.pt | Joana Ramos - up201605017@fe.up.pt | Pedro Gonçalves - up201604643@fe.up.pt

Problem Description

In the first part of this project, we simulated a restaurant, populated by a number of agents of three kinds: customers, waiters and kitchen.
The customers placed individual orders and their goal was to have a satisfying meal, while the waiters had as a main objective the maximization of the amount of cash received in tips.
In this second part of the project, our aim is to analyse how certain factors influenced two main **dependent** variables: a customer’s final mood (good or bad), a **classification** problem, and the value of a waiter’s tip after serving a customer, a **linear regression** problem.
Both of these variables are affected by other (**independent**) variables , such as the customer’s initial mood, the characteristics of the meal ordered (preparation time, quality and availability) and the waiter’s strategy regarding information sharing.

Experiments

As a first experiment, we collected information from several executions of our application regarding the tip a waiter received when delivering a meal to the respective customer and the various variables that had some sort of influence in this decision.

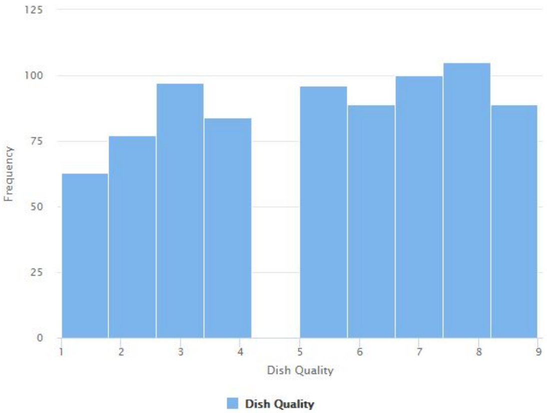
Finally, with the help of RapidMiner, we split the data gathered in a training and an example set, generated a linear regression model based on the later and applied it to the training one, evaluating the produced outputs.

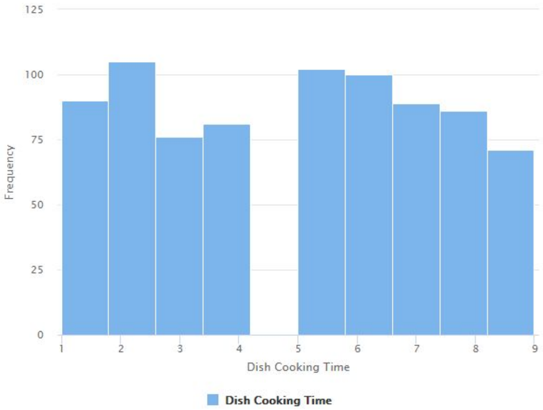
In the second experiment, most of the process was identical to the first one except for a few key details:

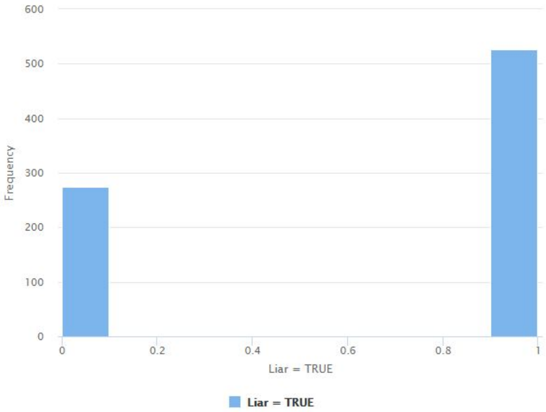
- The data gathered was related to a customer’s final mood after its meal being delivered (and the variables the affected this).
- The model applied to our training set was based on a decision tree and not linear regression.

Statistical data

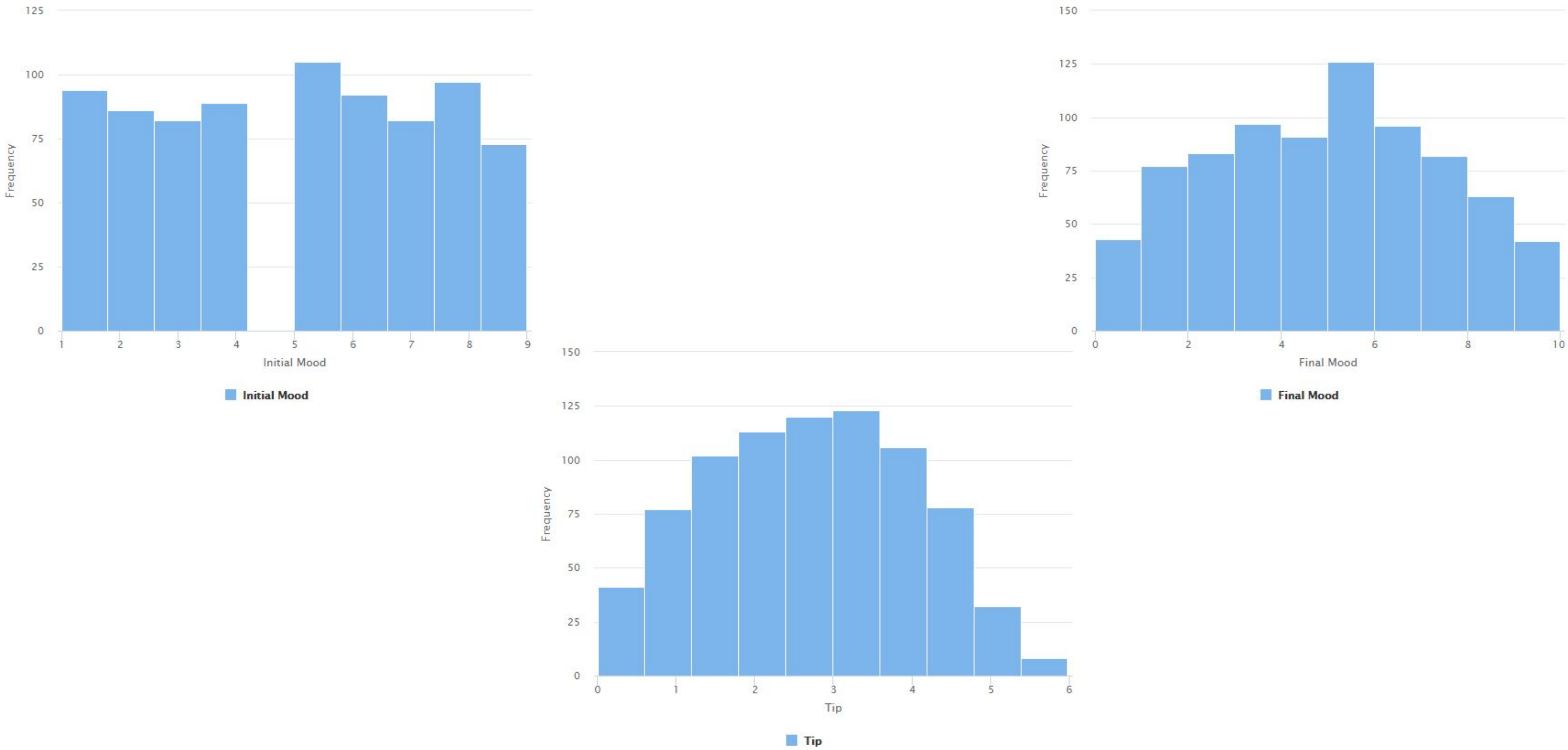
The following graphs represent the value distribution of the various variables from the data gathered from our application post processing in RapidMiner.







Statistical data



Data Analysis

In both experiments we used some operators to provide additional information, namely performance operators (of the regression and classification type) and correlation matrices.

Regarding the first operator mentioned, we chose to calculate the absolute (default and normalized) and relative error.

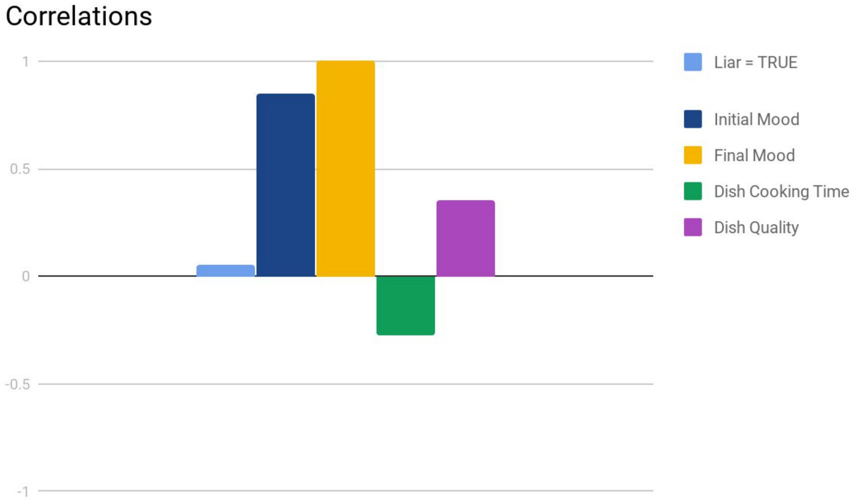
The correlation matrix was used to obtain the individual correlation between the tip and the other attributes.

This said, the most relevant data obtained was the following:

Data Analysis

Tips / regression experiment:

- Absolute error: 0.246 +/- 0.140
- Relative error: 12.77% +/- 25.99%
- Normalized absolute error: 0.228

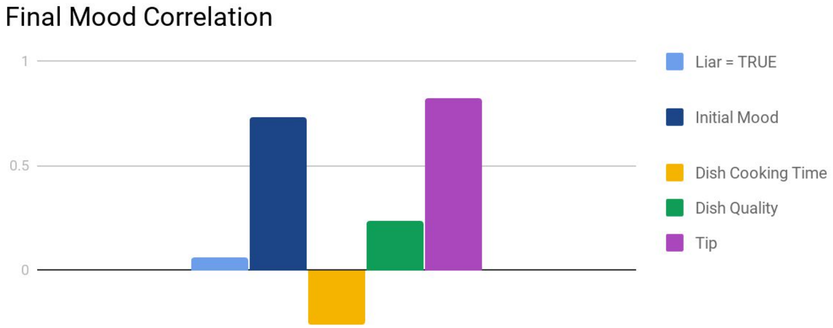


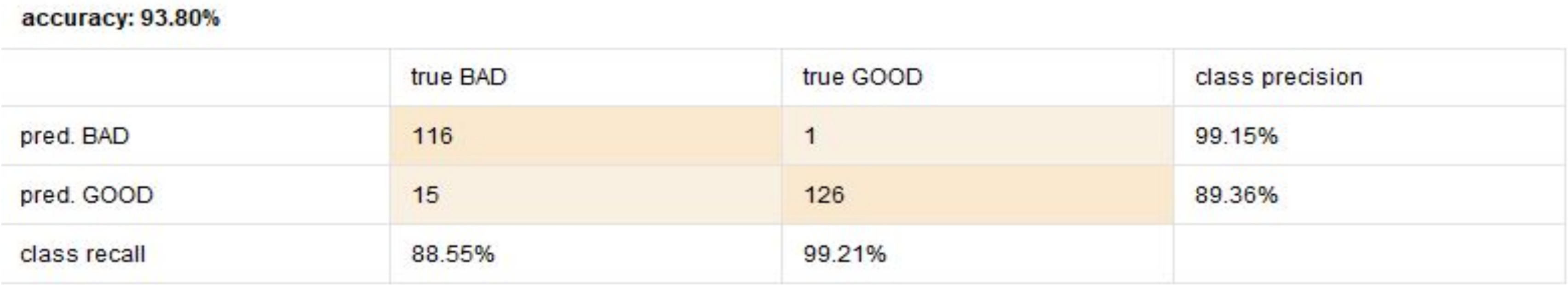
As we can see, the predictions were fairly accurate, with a low absolute error. This can be explained due to the fact that the way a tip is calculated follows a relatively direct formula based on the other attributes, with a small randomness mixed in.

We can also see that the attributes with the most impact are the initial and final mood of the customer.

Data Analysis

Final mood / classification experiment:





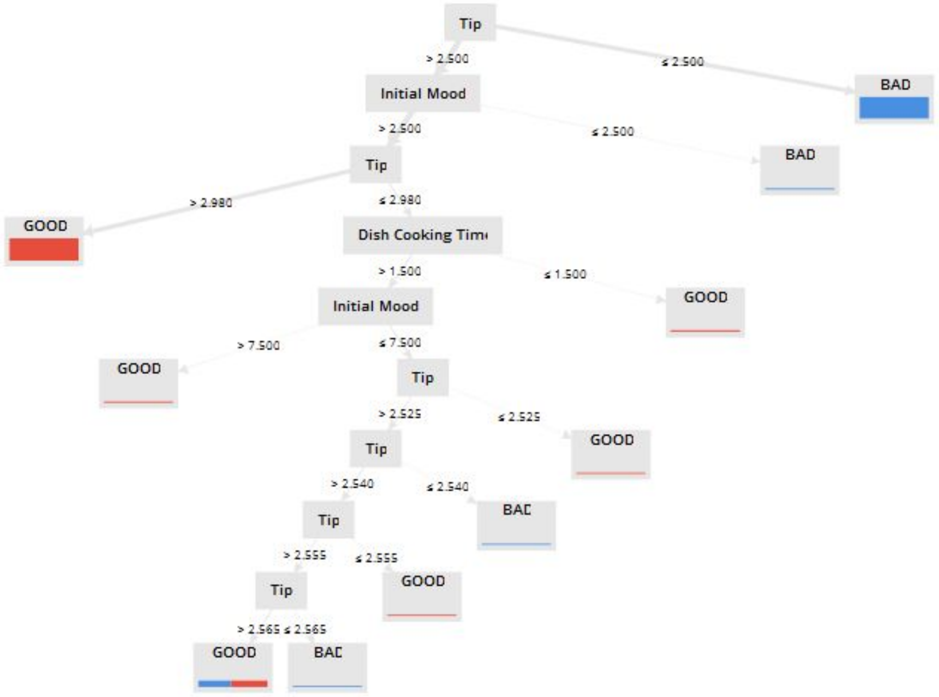
accuracy: 93.80%

	true BAD	true GOOD	class precision
pred. BAD	116	1	99.15%
pred. GOOD	15	126	89.36%
class recall	88.55%	99.21%	

As before, the model’s predictions were fairly accurate, mainly due to the same factor: the way a customer’s final mood is calculated is also fairly straight forward, with low levels of randomness affecting it.

Data Analysis

Final mood / classification experiment decision tree:



Conclusion

As expected, the models generated in both experiments were very accurate in their predictions since in both cases the level of randomness involved in our application didn’t influence the outcome by a large margin, being closer to a regular formula.

In future work we would like to increase this influence on the variables being tested so as to better mimic real life scenarios.

RapidMiner processes

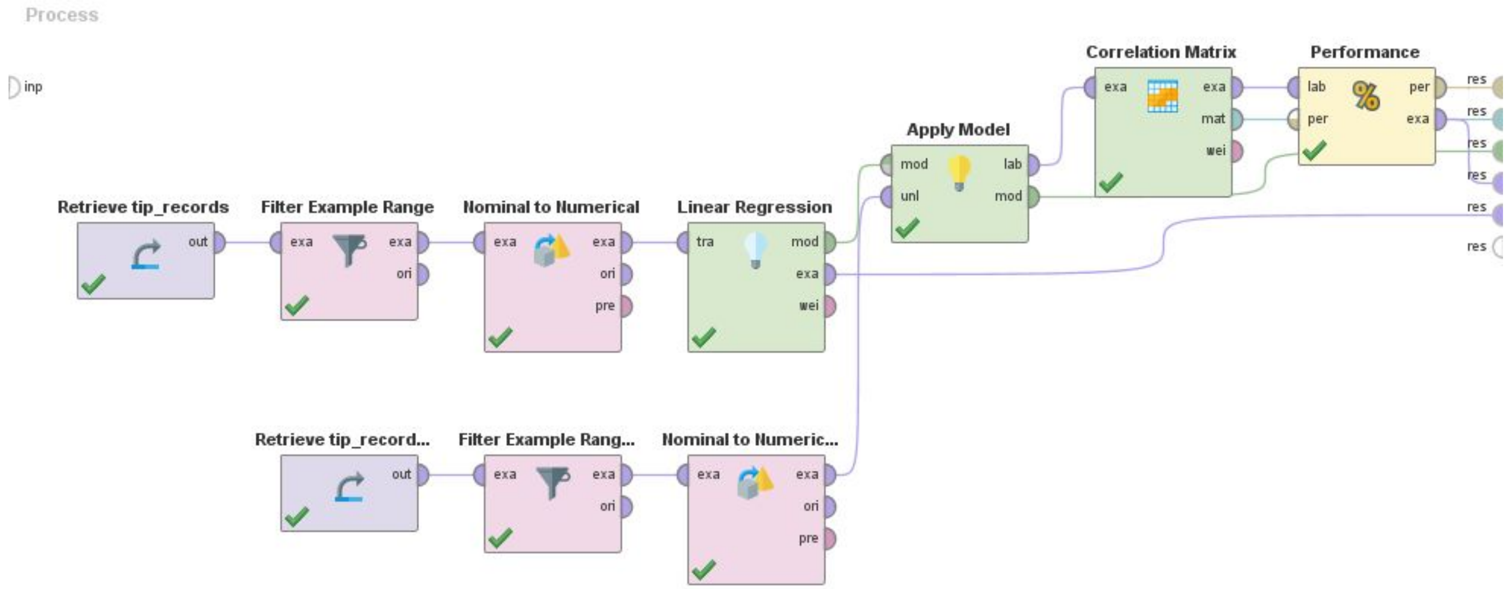
In both processes, we first filtered the data to create a training and an example set and proceeded to convert in each of them the binomial attribute “Liar” to a numerical (-1 and 1) one.

Next we created the desired model (linear regression or decision tree) with the example set and applied it to the testing set.

Finally, we then connected its output to a correlation matrix and a performance operator in order to show additional information.

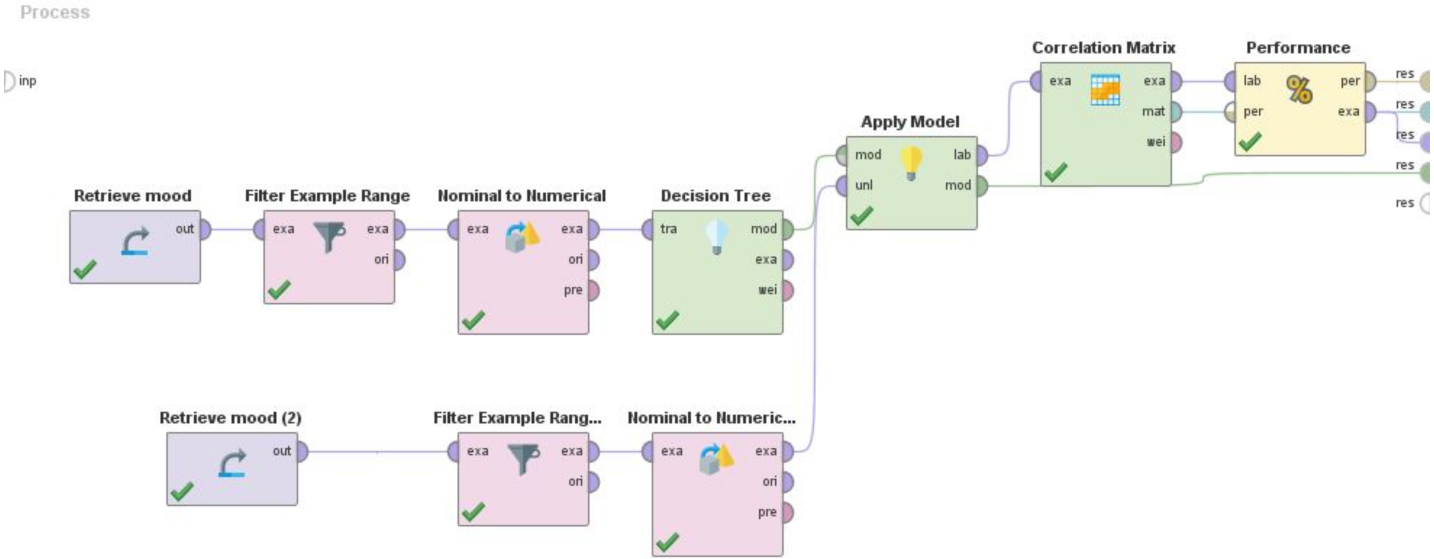
RapidMiner processes

Tips / Linear regression process



RapidMiner processes

Mood / Decision tree process



Other experiments

Besides the experiments explained in detail previously, we also applied a decision tree model to the tip amount problem to see how it would compare against the regression model.

As expected, the absolute error was greater than on the regression model but, surprisingly, only by a very small margin. The difference between the two was of 0.039.

Observations

Some interesting details observed in these experiments were the average tip a waiter got, which was of 2.809 +/- 1.291, so not whole much, and the correlation between the type of waiter (liar or not) and the tip he got. Since the absolute value of this was the same for either type, we can conclude that there is no clear benefit in either one of them in the short term.