# Lessons learned report

# Roberto Leyva

# Use of caret package to split

I didn’t really understand how to use the caret package. I even struggled to install it. The code to split the data frame is simpler and a lot more efficient because the number of lines of code is less.

It is a lot cleaner, more efficient in lines of code, and very straightforward. It is better to use packages instead of loops.

# Use of variable Importance to select features

At first, I thought it was a good idea to select features based on their correlation to the response variable, but I found that making a model with all the variables and then discriminating based on their importance was more practical.

The variable importance is one of the methods to select features for future models and it is very simple to use, so it is highly recommended.

# Add the predicted values to the incomplete data frame

I didn´t know that the names of the columns of the data frames needed to be the same to add the predicted values to the incomplete data frame. I now know the existence of rbind() to add data to another data frame.

Remember the function rbind() if someone wants to add data to from one data frame to another.

# Creating dummies

At first, I tried creating dummies without caret, but the variables were left at the end of the data frame with the response variable in the middle. That was inconvenient if I wanted to use dataframe[,1:10], so I tried using caret and the variables were created in the position they originally were and not in the last position.

It is highly recommended to use caret to make dummies to save time.

# Processing the same way, the complete and incomplete data frames

I discovered that if I want to apply a model trained and tested with the complete data frame, then the incomplete data frame must have the same variables and dummies.

It saves a lot of time if done at the beginning and it is highly recommended if someone is going to use predict().

# Installing packages from the url directly

Some packages didn’t install using the usual command *install.packages(“name of package”, dependencies = TRUE)*, so I went to the packages page of R and copied the url of the package needed and installed it with *install.packages(urlPackage, repos=NULL, type="source")*.

I recommend doing the same the same problem is encountered.

# Managing support and confidence in a market basket analysis

Sometimes when identifying rules of association, the support or confidence may be too high, and no rules would be identified.

If someone encounters this problem, I recommend decreasing support first and then confidence. It also depends on what the clients need, so it is advised to notify the client when the parameters given aren´t identifying any rules of association.