CUNEF R - Assessment

Master DS - Leonardo Hansa

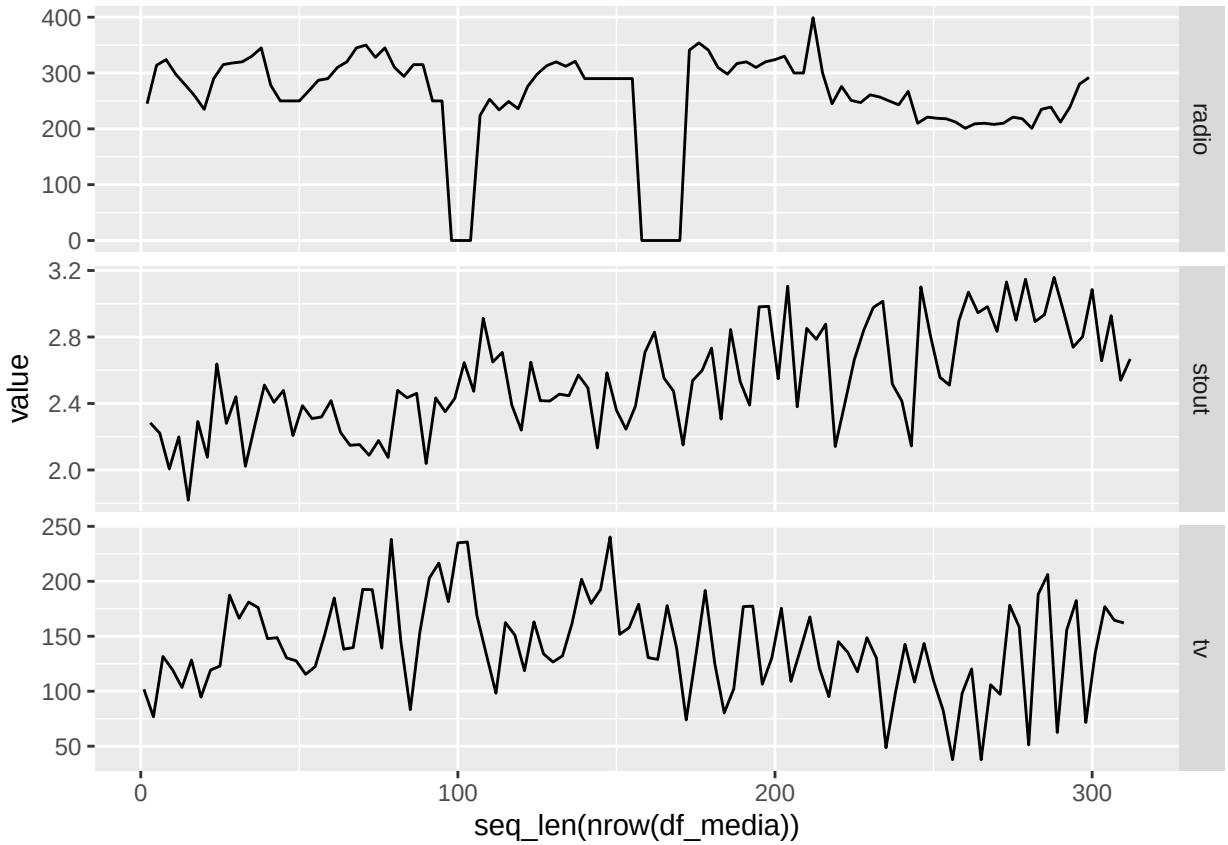
September, 2021

You are preparing a dataset for a posterior Marketing Mix model. The main goal of MMM is quantifying the effect of advertising and promotions on the sales of a product. Here we will be working with anonymous data of a unknown product and its weekly sales during two years.

Provide a code in R for preparing the data, as explained below. The effectiveness of the code, as well as its cleanness, will be considered for the evaluation. Comment everything you consider worth it with `#`.

- Create a data frame called `df_mmm` from the csv file `mktmix.csv`. Use `janitor` for change the column names into something meeting the `tidyverse` guidelines.
- How many columns are there? And rows? What are the classes of `base_price` and `discount`? Try and guess what they mean.
- `newspaper_inserts`'s class is `character`. Change its values so that it can be numeric. All the `NA` values should be 0; the rest of them should be 1.
- How many different values are there in the `website_campaign` column (`NA` doesn't count)? Create new columns for each of these categories, defined as 1 if `website_campaign` equals this category, 0 in other cases. For instance, if `website_campaign` equals `"Google"` on the 10th row, then you will create another column called `Google` that will equal 1 on the 10th row and will equal 0 on the rest.
- Create a line plot with ggplot2 showing the evolution of `new_vol_sales`, which would be the target variable in a model. Since we haven't been provided with dates, you will have to invent an x axis (it can be just numbers from 1 and so on).
- Create a histogram and a boxplot of the same variable. Based on the plots, which is the median of the variable? Calculate it with an R function. Were you close?
- Select only the media investment columns: `tv`, `radio` and `stout`, and create a new data frame just with them. Use this data frame and the provided code for creating a plot with the evolution of these three columns. This should be a plot in an only figure, with a share x axis but different y axis (see the result). For using the provided code, suppose the data frame you created with just the media data is called `df_media`. **Is there anything worth mentioning from the plot?**

```
library(tidyr)
df_media <- df_media %>%
  pivot_longer(everything())
```



- `in_store` is an index of the stock available on stores for selling the product. Create a scatter plot with ggplot2 for comparing the `new_vol_sales` column against `in_store`. Choose carefully which column should be set on the x axis and which on the y, considering that `new_vol_sales` will be the target variable on a model, i.e., the analyst will want to explain this variable based on the rest of the information. **Explain your decision and also comment anything interesting from the plot.** For doing this, think about the relation that could exist between the stock of a product and its sales.

- Create two different versions of the previous plot:

  - Color each dot differently based on the `newspaper_inserts` column (using `as.factor()` here is recommended).
  - Color each dot differently based on the `tv` column.

- Create another column on the data frame indicating whether a discount has been applied or not. You can name `discount_yesno`, for example. The column can be numerical or logical. After that, create another data frame aggregating the original one, for calculating the average base price when there's a discount and where it isn't. Use this data frame for creating a column plot with ggplot. On the x axis you should use the `discount_yesno` values and on the y axis, the average price you calculated. Are the any significant differences? *Remark.* Try not to create new data frames nor overwriting the original one, but chain all the operations with `%>%`, including the plot.

- Create a function that fits a model on this dataset using the provided code. The idea of the function is selecting a subset of columns on the data frame, creating a data frame with this selection, using the data frame on a model and returning a number that will indicate how good the model is. So the input will be a character, that will be the names of the columns, and the output will be this fitness number, provided in the next piece of code. You have been asked to create an auxiliary data frame with the

selection of columns: for this piece of code we are assuming you've called the data frame `df_aux` but you can name it as you want and change the code.

```
# This code fits a linear regression model with all the variables
# in df_aux, using new_vol_sales as target variable
my_model <- lm(new_vol_sales ~ ., data = df_aux)

# Value to be returned by the function, the adjusted R squared.
summary(my_model)$adj.r.squared
```

- You are given three sets of variables. Create a list whose elements will be these three vector. Now, using `map_dbl()` or `sapply()`, call the function you created in the previous exercise for the three cases. Which of the three subsets provide the best model, bearing in mind that the larger the returned number, the better?
  - `c("base_price", "radio", "tv", "stout")`
  - `c("base_price", "in_store", "discount", "radio", "tv", "stout")`
  - `c("in_store", "discount")`
