CUNEF ETL - Exercises

Leonardo Hansa

October, 2021

Section 3. General tools

Exercise 1. Using `ls` and `cd` in your computer explore your folders and the contents of the folder you use for the Master's degree. *Remark.* If you use Windows, it is recommendable downloading Git bash for Windows.

Exercise 2. Create in your local Desktop a file called `testing-bash.ter` and write in it the text `"Holi!"`. Use `echo` and `>` for this. Then create a folder in your Documents folder named `carpeta_prueba`. Move the previous file inside this folder, and check with `ls` you've done it properly. Now use `cat` for checking the text is actually written in the file. Finally, remove the folder `carpeta_prueba` with `rm`.

Exercise 3. Run a docker container from the image `lhansa/cunefark:0.1.0`, as shown in the Canvas notes. Explore the container with `ls` and `cd`. Create a Git repository in GitHub and link a folder inside this container to that repo.

Section 4. Extract

Plain text

Exercise 1. With pandas, from the `hipotecas_lectura` file read only the data from the second semester of 2020, and including somehow the names of the columns.

Exercise 2. With readr, from the `hipotecas_lectura` file read only the data from the first semester of 2020, and including somehow the names of the columns.

Exercise 3. Try your best reading the original file downloaded from the INE's site, `hipotecas_numero_ine.csv`. Remark: you may need to specify that the encoding is `"ISO-8859-2"`.

Excel

Exercise 1. Read with Python and `pandas` the second sheet of the `ejemplos_lecturas.xlsx` file.

SQL

Exercise 1. In R, download all the rows from `IndexPrice` in `indexKaggle.sqlite` whose region is United States or Europe, from 2019 until the end of the period.

Exercise 2. In R and Python, from the `indexKaggle.sqlite` download a table containing all the close prices and volume since 2007 until 2010 whose currency is dollars or euros.

Exercise 3. With R or Python, use the `elections2016.sqlite` database for extracting some data. We want a table that includes all the adjusted polls for Trump and Clinton in the Ohio and Pennsylvania states, along with the final results, order from the newest poll to the oldest (considering only the `enddate` column). The final table will have the next columns:

- (From the `Polls` table) state, enddate, grade, samplesize, adjpoll_clinton, adjpoll_trump.
- (From the `Results` table) electoral_votes, clinton, trump.

Exercise 4. In the **Pets** database, check if there any owner with more than one pet.

Exercise 5. Calculate the income per day considering all the procedures.

Exercise 6. Using `strftime()`, calculate the income per month considering only the transactions done by owners from the largest city in the database (the largest city is the one with a larger number of owners).

Section 5. Transform

Missing values

Exercise 1. Finish replacing the `NA` in the `df_simulated` data frame using the column known distributions. For column `V5` use a normal distribution with a mean and a variance you consider appropriate.

Exercise 2. Given the next vector, replace every `NA` value with the previous non `NA` value.

```
set.seed(5678)
vector_letters <- sample(letters, 50, TRUE)
vector_letters[sample(seq_len(50), 25)] <- NA
```

Exercise 3. Replace all the `NA` of the column `V5` in the `df_simulated` data frame using the moving average method –with a period longer than 1.

Exercise 4. Build a function for scaling the `iris` dataset with the *min-max* approach and scale all the numeric columns.

Exercise 5. For the data frame `iris` build new functions `setosa`, `versicolor` and `virginica`. `setosa` will equal 1 if `Species == "setosa"` and 0 elsewhere, and so on.

Dates and time series

Exercise 1. Extract from the Pets database the daily number of procedures from the `ProceduresHistory` table.

Exercise 2. In that table you extracted in the previous exercise, create a new column that equals 1 if the date is a Sunday; 0, elsewhere. For knowing when a date is Sunday, you can use something like `format(a_date, format = "%u")`, which output the weekday number (7 for Sundays). **Remark.** The column must be of type `Date`.

Exercise 3. During February 4th 2016 there was a peak, a very extreme value. Create a column with a dummy variable indicating that date.

Exercise 4. Level variables can be useful when modelling, for indicating whether the average during a period was higher than during other period. Create two level variables (1s and 0s), one for each semester.

Exercise 5. Let's go now with something independent from the previous data. Imagine we have a data frame like the one created from the next code. The first column indicates the beginning and end of each week of 2021, but in a terrible format. Create a new column with only the first date of each week, but with the format `"yyyy-mm-dd"`.

```r
library(dplyr)
crear_dias <- function(ini, fin) {
  format(seq(as.Date(ini), as.Date(fin), by = 7),
         format = "%d/%m/%Y")
}

fechas_horribles <- paste(
  crear_dias("2020-12-28", "2021-12-27"),
  crear_dias("2021-01-03", "2022-01-02"),
  sep = " - "
)

df <- tibble(
  semana = fechas_horribles,
  metrica = runif(length(fechas_horribles))
)
```

Regular expressions

Exercise 1. Translate the regex operations in R into Python. *Remark.* For replacing, use the `re.sub()` method.

Exercise 2. Translate the regex Python operations into R.

Master in DS Introduction to programming

Section 6. Load

Exercise 1. Repeat with R all the process shown in Python for the `indexKaggle.sqlite` database.

Section 7. APIs

Exercise 1. Select three subgroups within the INE data base and retrieve the IPC for these subgroups. Make the request within a loop. Somehow, manage to get a table similar to the original one. It can be done in Python or R.

fecha	alimentos	bebidas_no_alcoholicas	bebidas_alcoholicas	alquiler
202108	...	...	...	...
202107	...	...	...	...
202106	...	...	...	...

Exercise 2. Think on a company you're interested in and extract the following data. Check the documentation.

- its market capitalization.
- its PE ratio.

- its total revenue.