

Examen

Consultor Sr.

Parte Práctica

Creado por: Edgar Gerardo Alarcón González



Librerías y funciones de R que serán utilizadas

```
# Propias de R:
library(readxl) # Requiere java
library(dplyr)
library(lubridate)
library(ggplot2)
library(scales)
# Para SQL:
library(sqldf)
# Para Python:
library(reticulate)

# Del GitHub personal del autor:
directorio = "C:/Users/alarc/Documents/Actuaría/GitHub/R_Actuarial/Programación General/"
carpeta = "R + Python + SQL + LaTeX en RStudio/Examen Consultor Sr/Scripts de apoyo/"
archivo = "Funciones_RMD.R"
ruta = paste0(directorio,carpeta,archivo)
source(ruta)
```

Ejercicio 1 - Histórico de transacciones de venta

Usar el archivo Base_txns.xlsx y sigue las instrucciones:

```
carpeta = "R + Python + SQL + LaTeX en RStudio/Examen Consultor Sr/Datos/"
archivo = "Base_txns.xlsx"
ruta = paste0(directorio,carpeta,archivo)
datos <- read_excel(ruta)
```

a) Lee el archivo Base_txns.xlsx y contesta ¿Cuántos clientes y tiendas únicos hay en la base?

```
# Cantidad de clientes únicos:
datos$CURP %>% unique() %>% length()
```

```
## [1] 19
```

```
# Cantidad de tiendas únicas:
datos$Tienda %>% unique() %>% length()
```

```
## [1] 4
```

b) Elabora una tabla por cliente y realiza un conteo de todas sus transacciones y la suma de monto.

```
datos %>% group_by(CURP) %>% summarize(cantidad_transacciones=n(),
                                      monto_total=sum(`Monto en pesos`))->datos_b
```

CURP	cantidad_transacciones	monto_total
CRMB74615	84	2065727
FCC81619	84	1990572
GAJD811230	84	2056378
GDC8657	84	2059190
GLB83411	84	1997628
GMC8657	84	2091895
GODÁ8479	84	1906344
HEMA75324	84	2069923
LÓRÁ81427	84	1966047
LSC74823	84	2318132
MAHL73116	84	1987619
MVC8254	84	2302778
PÉMA771013	84	2141958
PGC811211	84	2156495
RAFH84102	84	1922734
RCC8244	84	2067170
ROMM87314	84	2141586
SÁÁJ87826	84	2008622
SUC87618	84	2116251

Tabla 1: Resumen de tabla `Base_txns.xlsx` procesada por el inciso b).

c) Elabora una tabla por tienda y realiza un conteo de todas sus transacciones y suma de monto.

```
datos %>% group_by(Tienda) %>% summarize(cantidad_transacciones=n(),
                                          monto_total=sum(`Monto en pesos`))>datos_c
```

Tienda	cantidad_transacciones	monto_total
159	421	10384253
175	403	9951553
499	379	9258523
872	393	9772720

Tabla 2: Resumen de tabla `Base_txns.xlsx` procesada por el inciso c).

d) Crea una nueva columna que se llame periodo que contenga el año y el mes de la fecha de operación (si fecha = 25/02/2022 entonces periodo = 202202). Crea una tabla por periodo y realiza un conteo de todas las transacciones y suma de monto.

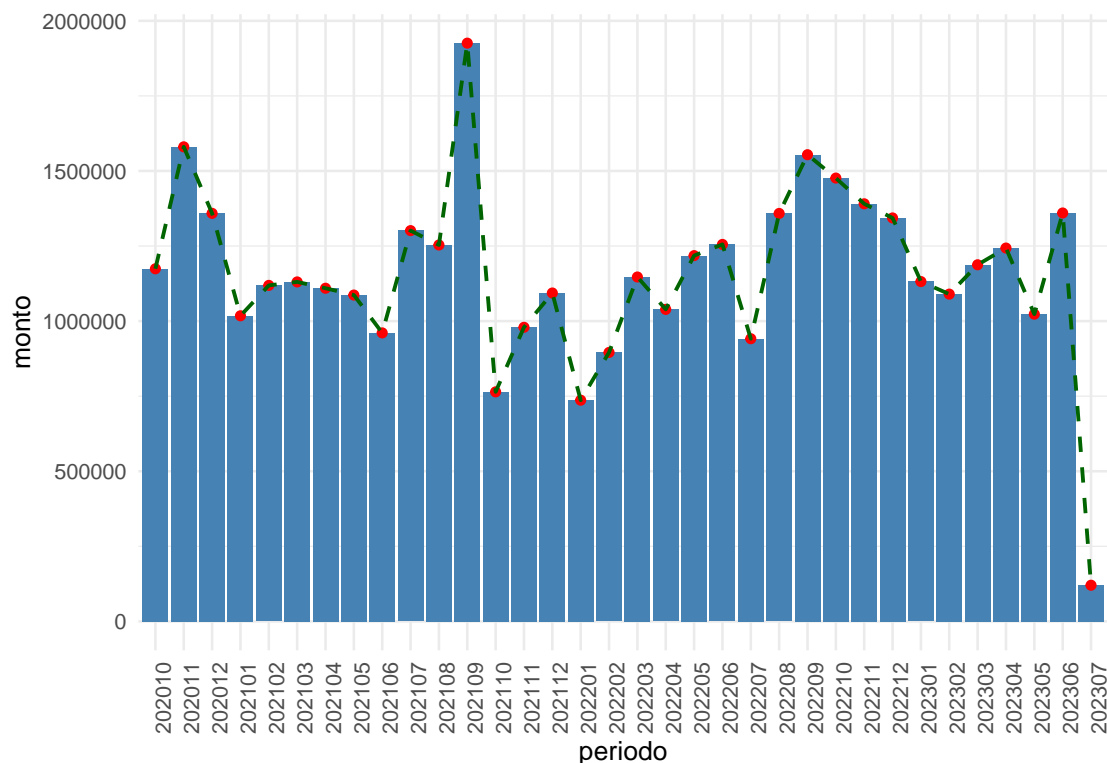
[illegible]

Nombre	Ap paterno	Ap materno	Fecha de operación	Tienda	periodo
Arturo	Hernández	Martín	2020-12-29	159	202012
Débora	García	Jiménez	2022-10-03	159	202210
Lázaro	Martínez	Hernández	2022-01-18	499	202201
Ángela	López	Ruiz	2023-06-03	499	202306

Tabla 3: Fragmento de tabla `Base_txns.xlsx` procesada por el inciso d).

- e) Grafica uno de los resultados del inciso anterior, en donde el eje x = periodo y en y tendrás Monto o transacciones. Ejemplo en la imagen 1.

```
datos %>%
  group_by(periodo) %>% summarize(monto=sum(`Monto en pesos`)) %>%
  ggplot(aes(x=periodo, y=monto)) +
  geom_bar(stat="identity", fill="steelblue")+
  geom_point(col="red")+
  geom_line(mapping = aes(x=as.numeric(periodo)),linetype="dashed", color="darkgreen", size=0.75)+
  theme_minimal()+
  theme(axis.text.x = element_text(angle = 90))
```



- f) Exporta los resultados del inciso c en un archivo plano con extensión txt delimitado por un pipe.

```
carpeta = "R + Python + SQL + LaTeX en RStudio/Examen Consultor Sr/Resultados/"
archivo = "resultados_inciso_c.txt"
ruta = paste0(directorio,carpeta,archivo)
write.table(file = ruta,x = datos_c,sep = "|")
```

Ejercicio 2 - Procesa dos bases con código SQL

Utiliza lenguaje SQL para procesar las bases Base_txns.xlsx y Tiendas.xlsx, puedes utilizar sqldf en R, pandasql en Python o alguna otra librería. Sigue las siguientes indicaciones:

- a) Utiliza cualquier función para leer los archivos Base_txns.xlsx y Tiendas.xlsx.

```
carpeta = "R + Python + SQL + LaTeX en RStudio/Examen Consultor Sr/Datos/"
archivo = "Base_txns.xlsx"
ruta = paste0(directorio, carpeta, archivo)
base_txns <- read_excel(ruta)
archivo = "Tiendas.xlsx"
ruta = paste0(directorio, carpeta, archivo)
tiendas <- read_excel(ruta)
```

- b) Cruza las tablas usando como llave la tienda.

```
sqldf('
  SELECT *
  FROM base_txns t1 JOIN tiendas t2 ON t1.Tienda = t2.ID_TIENDA
  ') -> query_b
```

Nombre	Ap paterno	Ap materno	Fecha de nacimiento
Arturo	Hernández	Martín	1975-03-23 18:00:00
Débora	García	Jiménez	1981-12-29 18:00:00
Lázaro	Martínez	Hernández	1973-11-05 18:00:00
Ángela	López	Ruiz	1981-04-26 18:00:00

CURP	Monto en pesos	Fecha de operación	Tienda
HEMA75324	22379	2020-12-28 18:00:00	159
GAJD811230	14124	2022-10-02 19:00:00	159
MAHL73116	24482	2022-01-17 18:00:00	499
LÓRÁ81427	48015	2023-06-02 19:00:00	499

ID_TIENDA	longitud	latitud	CVE_INEGI
159	-99.13	19.38	09005
159	-99.13	19.38	09005
499	-99.06	19.34	09006
499	-99.06	19.34	09006

cve_ent	nom_ent	cve_mun	nom_mun
9	Ciudad De Mexico	5	Gustavo A. Madero
9	Ciudad De Mexico	5	Gustavo A. Madero
9	Ciudad De Mexico	6	Iztacalco
9	Ciudad De Mexico	6	Iztacalco

Tabla 4: Fragmento del cruce tablas procesada por el inciso b).

c) Excluye las transacciones realizadas por Milton Rodríguez Muñoz.

```
sqldf('
SELECT *
FROM base_txns t1 JOIN tiendas t2 ON t1.Tienda = t2.ID_TIENDA
WHERE Nombre NOT LIKE "Milton" AND
      [Ap paterno] NOT LIKE "Rodríguez" AND
      [Ap materno] NOT LIKE "Muñoz"
') -> query_c
```

d) Calcula el monto y numero de transacciones acumuladas por municipio y guárdalo en una tabla.

```
sqldf('
SELECT cve_mun, COUNT(cve_mun) as cantidad , sum([Monto en pesos]) as monto
FROM base_txns t1 JOIN tiendas t2 ON t1.Tienda = t2.ID_TIENDA
GROUP BY cve_mun
') -> query_d
```

cve_mun	cantidad	monto
5	421	10384253
6	379	9258523
51	403	9951553
114	393	9772720

Tabla 5: Resumen del curce de tablas procesado por el inciso d). La columna 'cantidad' expresa el número de transacciones acumuladas, y la columna 'monto' el total acumulado del monto en ese municipio.

Ejercicio 3 - Minería con Python

Usar el archivo `Nombres.txt` y sigue las siguientes instrucciones:

- a) Lee el archivo, toma en cuenta el formato txt, el encoding UTF-8 y la delimitación por un pipe.

```
import pandas as pd
directorio = "C://Users/alarc/Documents/Actuaría/GitHub/R_Actuarial/Programación General/"
carpeta = "R + Python + SQL + LaTeX en RStudio/Examen Consultor Sr/Datos/"
archivo = "Nombres.txt"
ruta = directorio + carpeta + archivo
# Se guarda en el objeto "datos"
datos = pd.read_csv(ruta, sep="|", encoding="UTF-8"))
```

Nombre	Apellido	Edad	Saldo	Id	CP
,Cristina	FLOREZ	28	5009.344	7065	356
,Cesar	ACUÑA	34	20278.343	758	7534
,Julio	JUÁREZ	45	18900.643	954	45980
,Fernanda	GONZÁLEZ	28	10000.346	184	4590

Tabla 6: Primeras observaciones `Nombres.txt` importada con código Python.

- b) Limpia los campos Nombre y Apellido (quita espacios, acentos y caracteres especiales).

Asumiremos que también se deben quitar las mayúsculas con el objetivo de tener monotonía.

```
# Nombre
import string
simbolos = str.maketrans('', '', string.punctuation)
datos["Nombre"] = datos["Nombre"].str.translate(simbolos)
datos["Nombre"] = datos["Nombre"].str.replace(" ", "")
datos["Nombre"] = datos["Nombre"].str.lower()
import unicode
datos["Nombre"] = datos["Nombre"].apply(unicode.unidecode)

# Apellido
datos["Apellido"] = datos["Apellido"].str.translate(simbolos)
datos["Apellido"] = datos["Apellido"].str.replace(" ", "")
datos["Apellido"] = datos["Apellido"].str.lower()
datos["Apellido"] = datos["Apellido"].apply(unicode.unidecode)
```

Nombre	Apellido	Edad	Saldo	Id	CP
cristina	florez	28	5009.344	7065	356
cesar	acuna	34	20278.343	758	7534
julio	juarez	45	18900.643	954	45980
fernanda	gonzalez	28	10000.346	184	4590

Tabla 7: Primeras observaciones de tabla `Nombres.txt` procesada por el inciso b).

c) Crea un campo que se llame First que contenga la primera letra del apellido.

```
datos["First"] = datos["Apellido"].str[0]
```

Nombre	Apellido	Edad	Saldo	Id	CP	First
cristina	florez	28	5009.344	7065	356	f
cesar	acuna	34	20278.343	758	7534	a
julio	juarez	45	18900.643	954	45980	j
fernanda	gonzalez	28	10000.346	184	4590	g

Tabla 8: Primeras observaciones de tabla `Nombres.txt` procesada por el inciso c).

d) Usando el campo First, filtra las personas que su nombre inicia con consonante.

```
datos_consonantes = datos[~(datos.First.isin(["a","e","i","o","u"]))]
```

Nombre	Apellido	Edad	Saldo	Id	CP	First
cristina	florez	28	5009.344	7065	356	f
julio	juarez	45	18900.643	954	45980	j
fernanda	gonzalez	28	10000.346	184	4590	g
cristina	selvas	29	4000000.745	20014	5456	s

Tabla 9: Primeras observaciones de tabla `Nombres.txt` procesada por el inciso d).

e) Agrupa la tabla anterior por los campos 'Nombre', 'Apellido' y 'Id', y suma el saldo.

```
datos_agrupado = datos_consonantes.groupby(["Nombre", "Apellido", "Id"]).sum()[["Saldo"]].reset_index()
```

Nombre	Apellido	Id	Saldo
ana	hernandez	6987	1250.7010
angel	florez	45045	1352379.3680
berenice	hernandez	34874	742.9239
cristina	florez	7065	15590.0320

Tabla 10: Primeras observaciones de tabla `Nombres.txt` procesada por el inciso e).

f) Exporta los resultados de la tabla agrupada en un archivo Excel con el nombre `Nombres_agrupado.xlsx`.

```
import openpyxl
carpeta = "R + Python + SQL + LaTeX en RStudio/Examen Consultor Sr/Resultados/"
archivo = "Nombres_agrupado.xlsx"
ruta = directorio + carpeta + archivo
writer = pd.ExcelWriter(ruta)
datos_agrupado.to_excel(writer)
writer.save()
```