



# TRABAJO FINAL: GESTIÓN DE DATOS GRUPO 5

Aldo Ortega  
Jorge Gonzales  
Fabrizio Berrios  
Jorge Aybar

# Evolución de Muertes por Clínica

```
# Plot yearly proportion of deaths at the two clinics
# .... YOUR CODE FOR TASK 3 ....

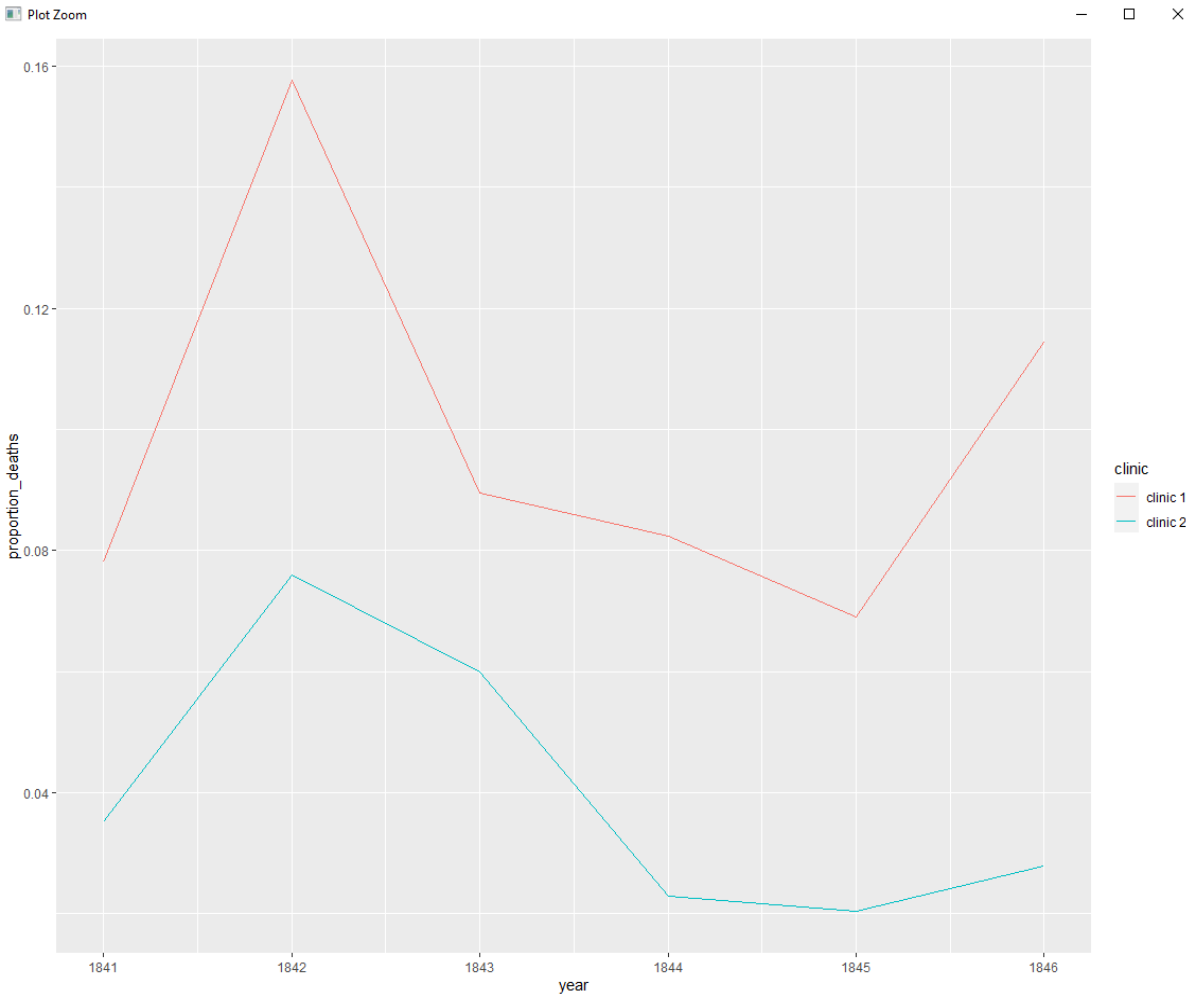
ggplot(yearly,aes(x=year,y=proportion_deaths,color=clinic)) + geom_line()
```

Evolución de Muertes de neonatos por clínica por año y proporción

# A tibble: 98 x 4

	date	births	deaths	proportion_deaths
	<date>	<dbl>	<dbl>	<dbl>
1	1841-01-01	254	37	0.146
2	1841-02-01	239	18	0.0753
3	1841-03-01	277	12	0.0433
4	1841-04-01	255	4	0.0157
5	1841-05-01	255	2	0.00784
6	1841-06-01	200	10	0.05
7	1841-07-01	190	16	0.0842
8	1841-08-01	222	3	0.0135
9	1841-09-01	213	4	0.0188
10	1841-10-01	236	26	0.110

# ... with 88 more rows



## Evolución de Muertes por mes y prueba de hipótesis del Dr. Semmelweis.

El Doctor Semmelweis hizo su recomendación y aseveración el **01-06-1847**, se observa que la proporción de muertes si se redujo

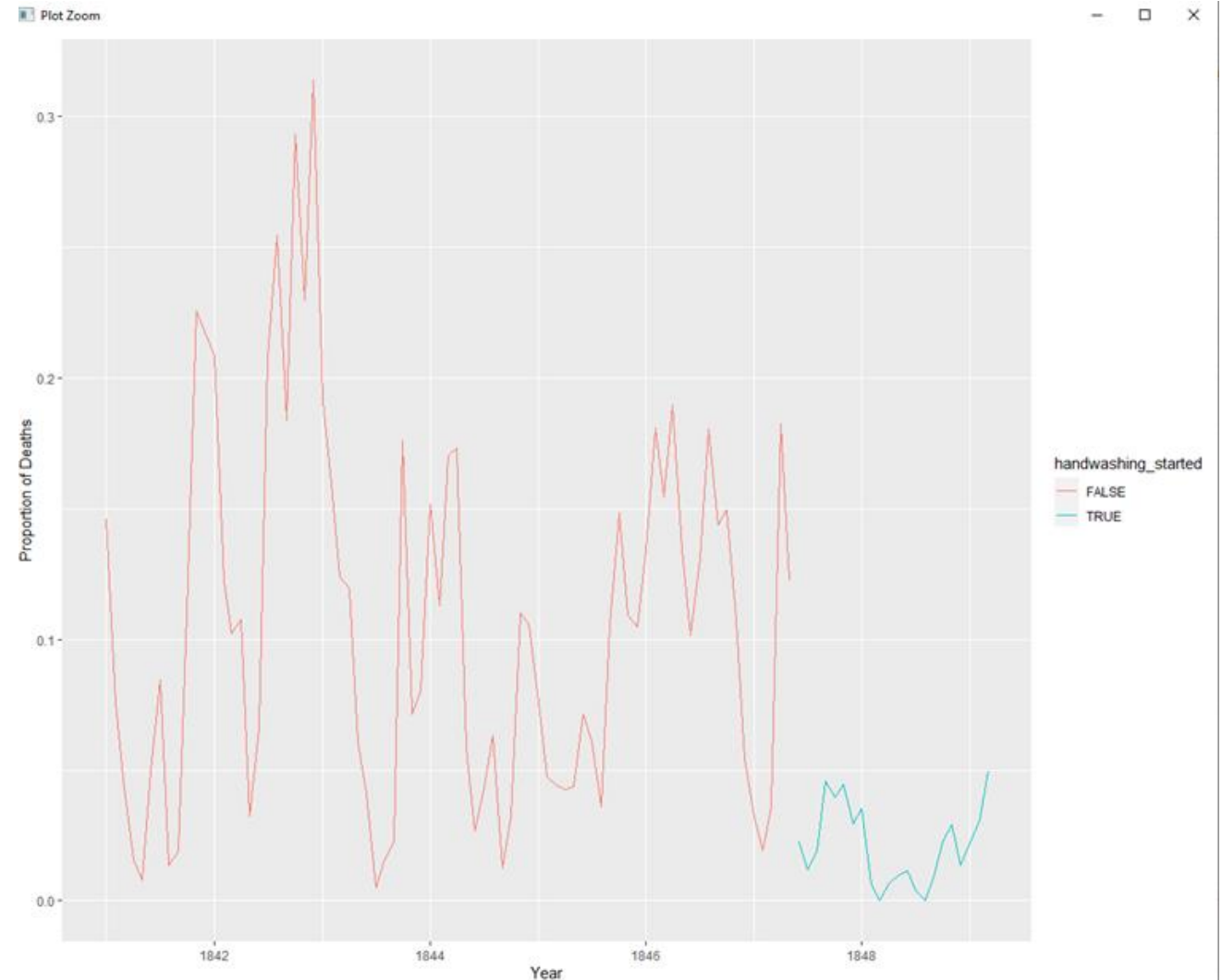
```
A tibble: 2 x 2
  handwashing_started mean_proportion_deaths
  <lgl>               <dbl>
1 FALSE              0.105
2 TRUE               0.0211
```

```
2 TRUE 0.0211
> # Calculating a 95% Confidence interval using t.test
> test_result <- t.test(proportion_deaths ~ handwashing_started, data = monthly)
> test_result

Welch Two Sample t-test

data: proportion_deaths by handwashing_started
t = 9.6101, df = 92.435, p-value = 1.445e-15
alternative hypothesis: true difference in means between group FALSE and group TRUE is not equal to 0
95 percent confidence interval:
 0.06660662 0.10130659
sample estimates:
mean in group FALSE mean in group TRUE
 0.10504998         0.02109338
```

```
> # The data Semmelweis collected points to that:
> doctors_should_wash_their_hands <- TRUE
> doctors_should_wash_their_hands
[1] TRUE
>
```

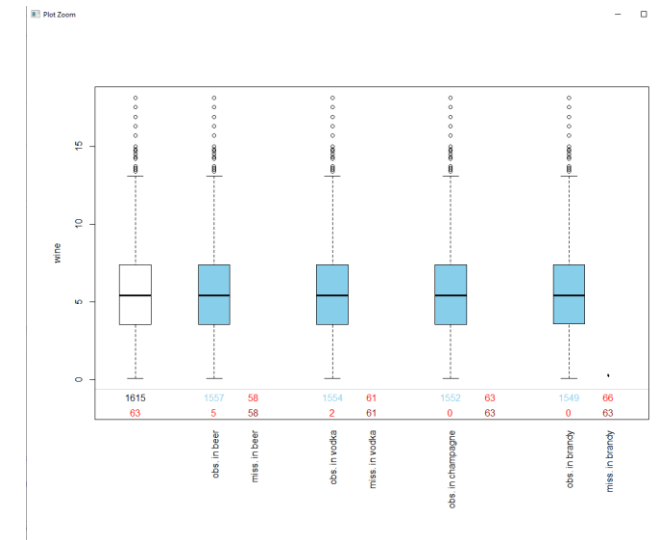
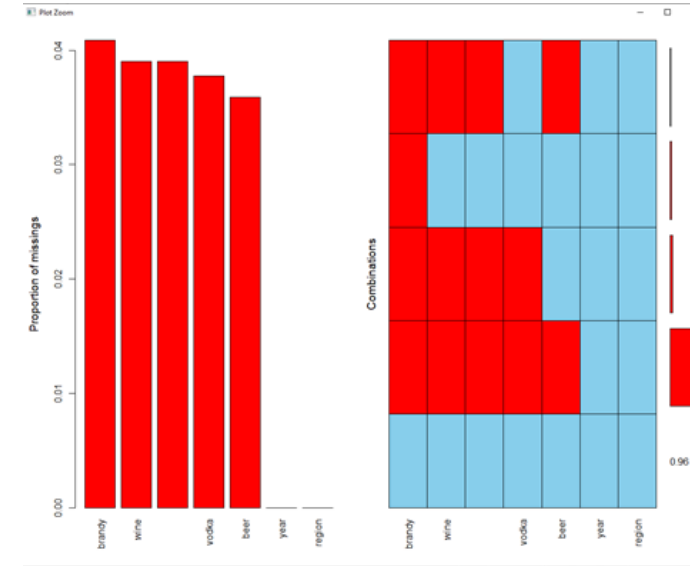


# Caso 2 : Análisis del Consumo de Alcohol en Rusia

## Proporción de Datos Perdidos

- 1 % de datos faltantes: trivial (el método de imputación no tiene mayor impacto)
- 1 a 10 % de datos faltantes: manejable (requiere un método “simple”)
- 10 a 20 % de datos faltantes: requiere métodos sofisticados (puede requerir método “propio”)
- Más del 20 % de datos faltantes: interpretación perjudicial (ya se perdió “demasiado”)

```
> # install.packages("VIM")
> library(VIM)
> # Mostrar cuales columnas tienen valores perdidos
> cidx_perd <- which(colsums(is.na(alcohol))!=0)
> cidx_perd
      wine      beer    vodka champagne    brandy
        3         4         5         6         7
>
> # Cantidad de valores perdidos en las columnas
> nperdidos <- colsums(is.na(alcohol[,cidx_perd]))
> nperdidos
      wine      beer    vodka champagne    brandy
        63        58        61         63        66
>
> # Porcentaje de valores perdidos en las columnas
> pperdidos <- 100*nperdidos/ndatos
> pperdidos
      wine      beer    vodka champagne    brandy
  3.900929  3.591331  3.777090  3.900929  4.086687
> |
```



# Caso 2 : Análisis del Consumo de Alcohol en Rusia

## Evolución de Ventas en Saint Petersburg

```
##San Petersburg Alcohol Sales

ventasSaintPetersburg <- alcohol %>% filter(region == "Saint Petersburg") %>% group_by(year) %>% select(region,wine,beer,vodka,champagne,brandy)
ventasSaintPetersburg

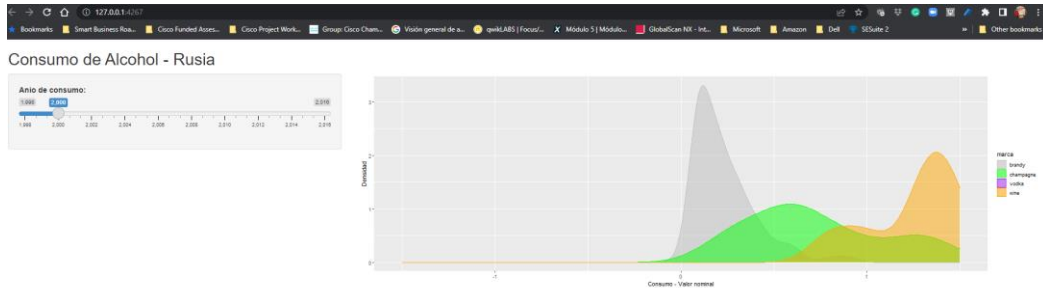
View(ventasSaintPetersburg)

#Evolución de Ventas Saint Petersburg

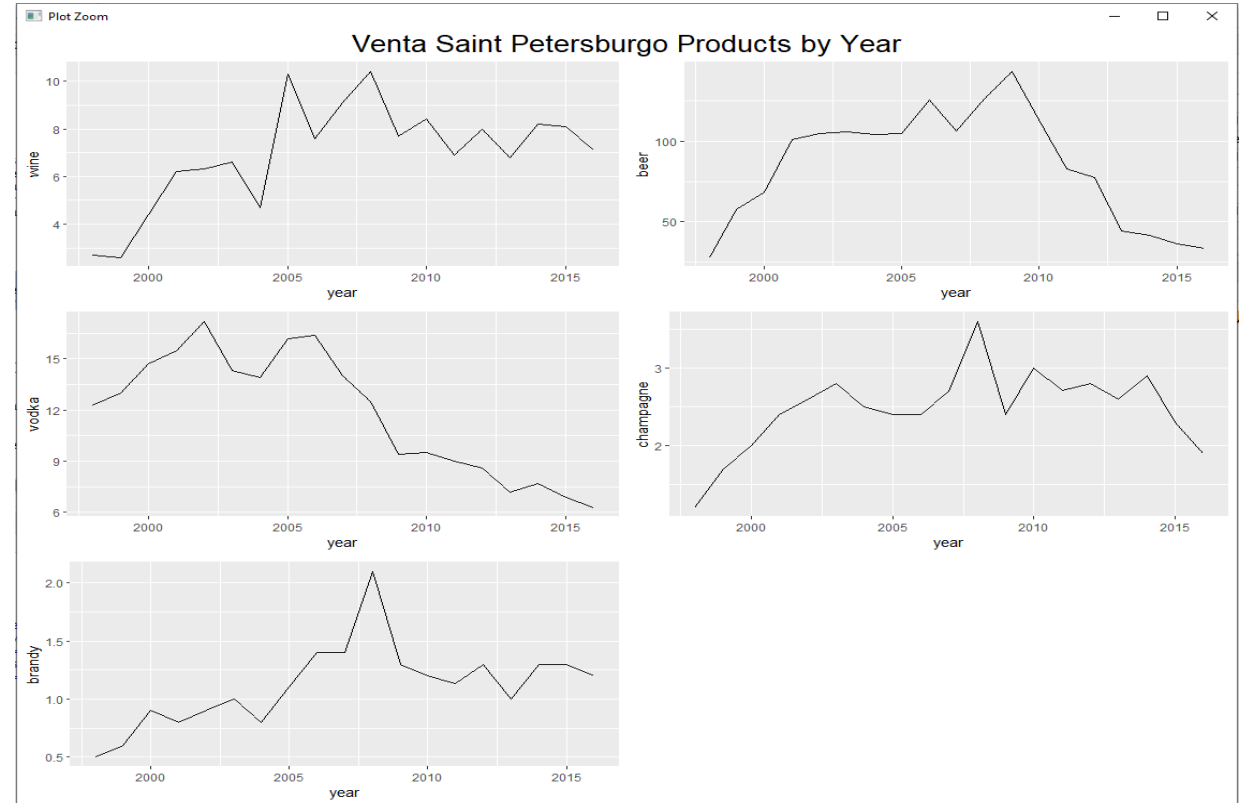
wine <- ggplot(ventasSaintPetersburg,aes(x=year,y= wine )) + geom_line()
beer <- ggplot(ventasSaintPetersburg,aes(x=year,y= beer )) + geom_line()
vodka <- ggplot(ventasSaintPetersburg,aes(x=year,y= vodka )) + geom_line()
champagne <- ggplot(ventasSaintPetersburg,aes(x=year,y= champagne )) + geom_line()
brandy <- ggplot(ventasSaintPetersburg,aes(x=year,y= brandy )) + geom_line()

# install.packages('ggpubr')
library(ggpubr)

final_plot <- annotate_figure(
  ggarrange(wine, beer, vodka, champagne,brandy, ncol=2, nrow=3),
  top = text_grob("Venta Saint Petersburg Products by Year", size = 20))
final_plot
```

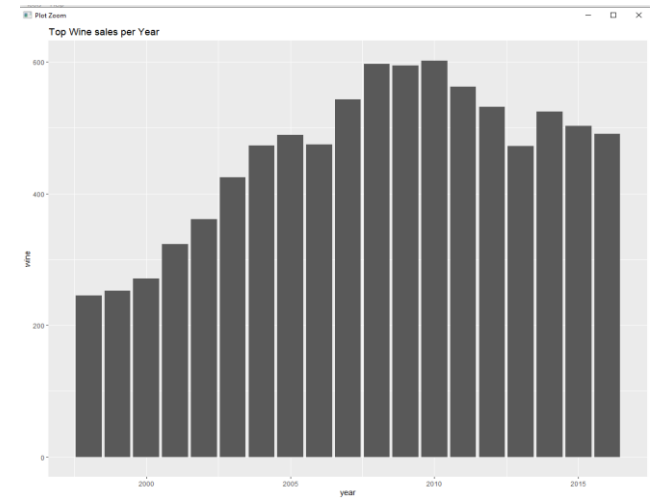
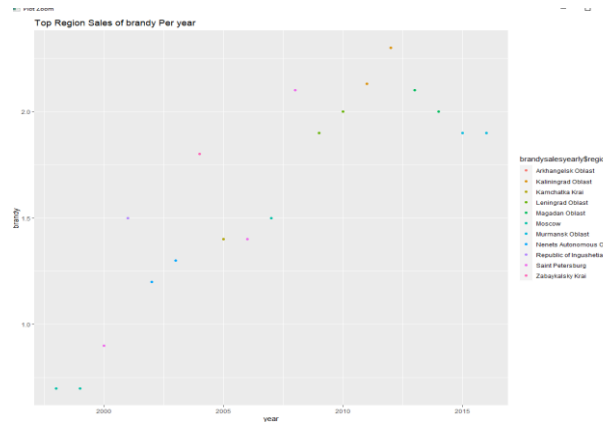
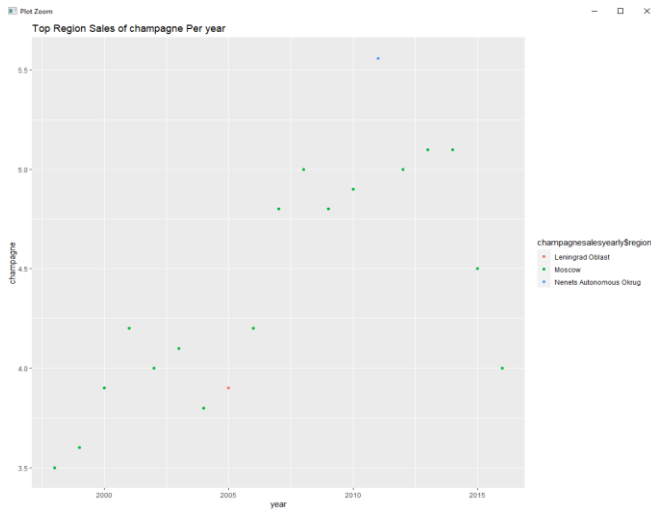
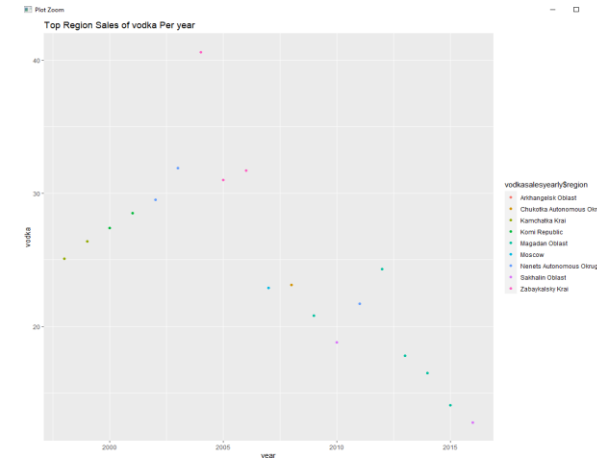
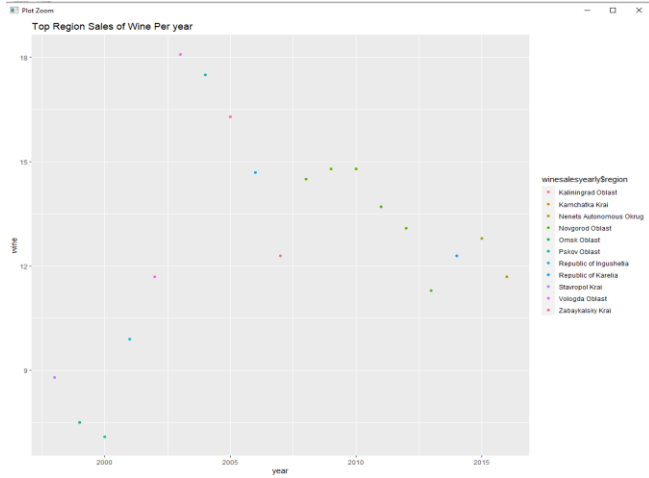


Shiny app view



## Caso 2 : Análisis del Consumo de Alcohol en Rusia

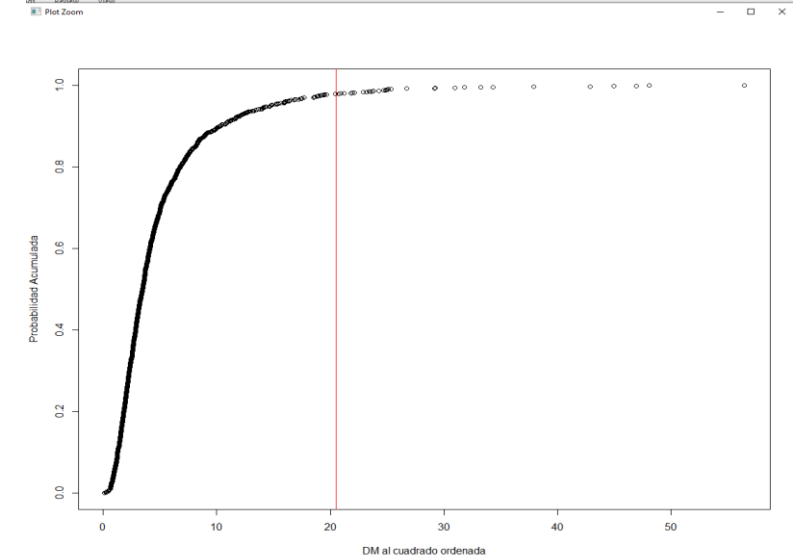
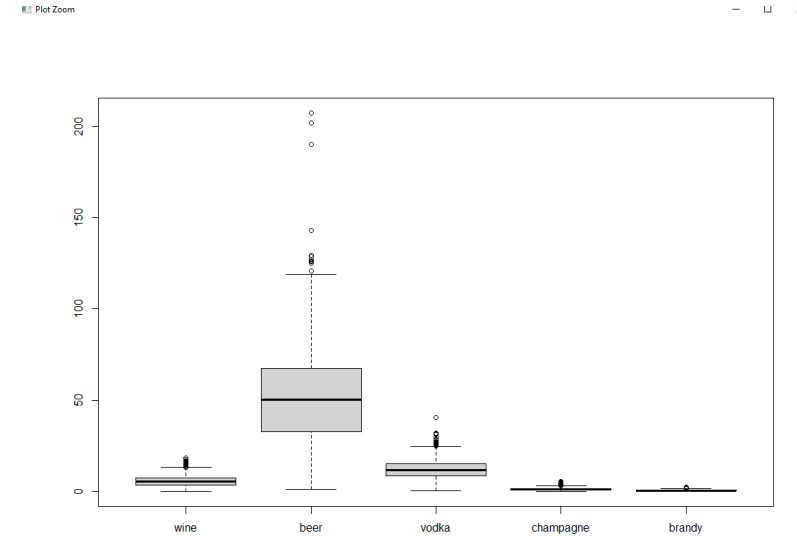
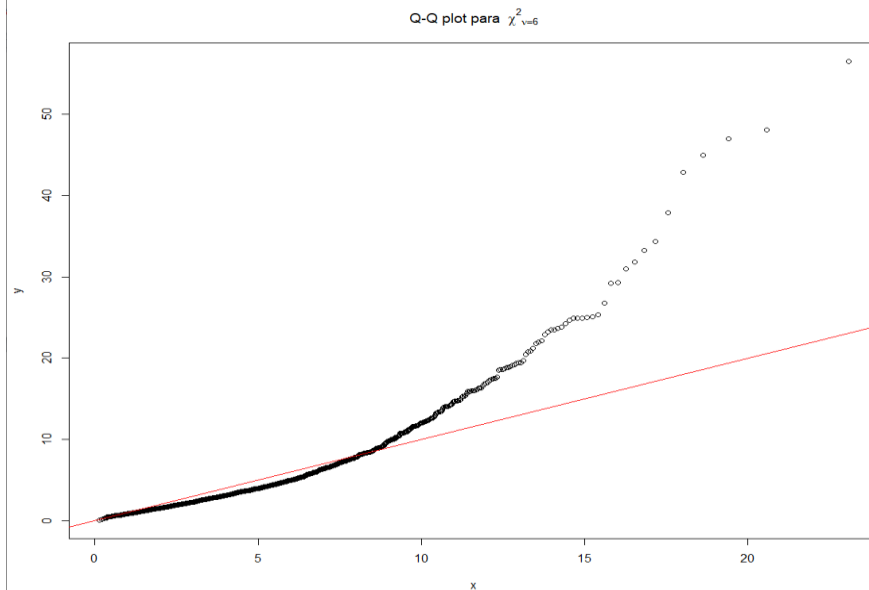
# Top N Sales per Región by Year for Wine, Beer, Vodka, Champagne and Brandy



# Caso 2 : Análisis del Consumo de Alcohol en Rusia

## Analisis de Outliers: Q-Q and Ojiva X<sup>2</sup>

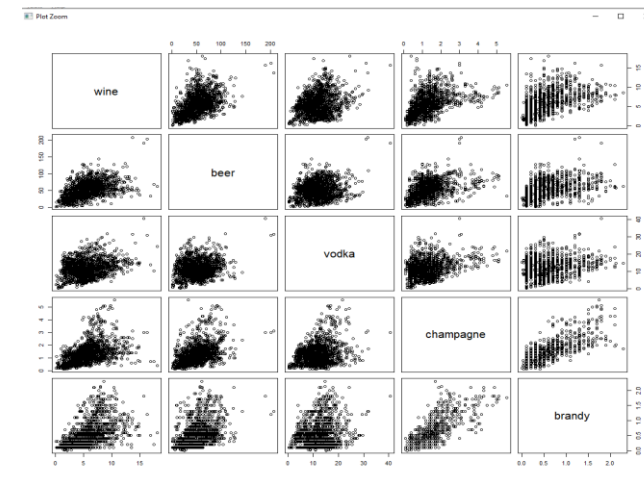
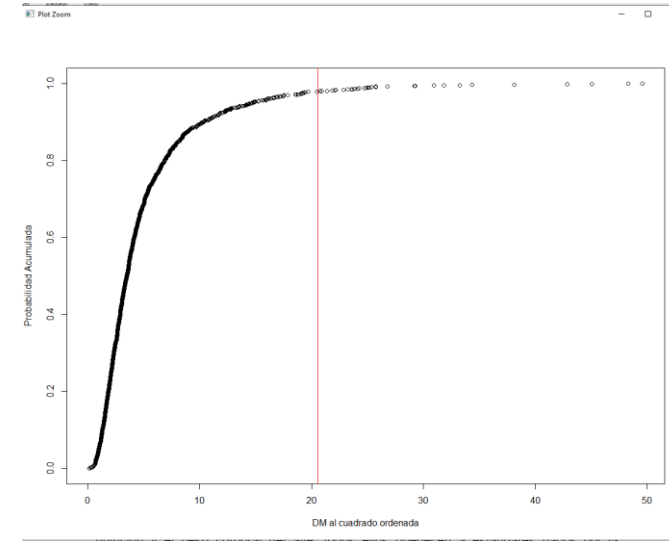
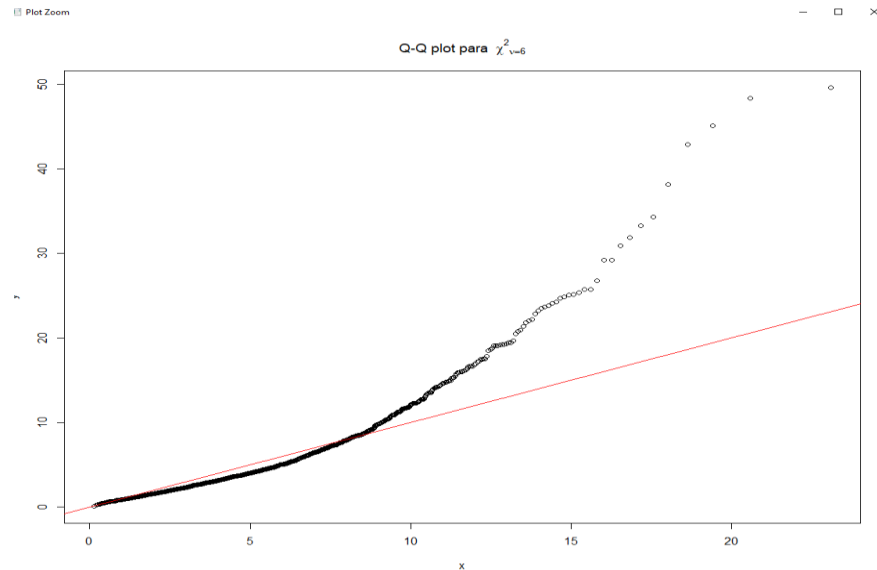
```
# Distribución Chi-Cuadrado: Punto de Corte
p <- 1-0.001
dof = ncol(dfa)
k <- (qchisq(p, dof))
print("el valor de k es: ")
1] "el valor de k es: "
k
1] 20.51501
idx_outliers <- which(dm2 > k)
idx_outliers
[1] 200 261 282 364 411 419 446 449 501 505 515 523 541 587 596 604 668 690 771 852 996 1009
23] 1014 1077 1095 1098 1158 1173 1176 1239 1254 1257 1339
dfa[idx_outliers,] # Registros con valores atípicos
A tibble: 33 x 5
  wine beer vodka champagne brandy
<dbl> <dbl> <dbl> <dbl> <dbl>
1 6.3 61.8 18.8 3.9 0.6
2 9.9 9.5 4 1.3 1.5
3 5 78.9 19.7 4.2 0.7
4 4.8 85.7 20.8 4 0.9
5 9.2 109 31.9 1.7 1.3
6 18.1 61.7 24.6 0.4 0.6
7 6.2 89.1 23 4.1 0.9
8 9.2 109 31.9 1.7 1.3
9 14.2 85.1 26.7 0.6 0.7
0 15.7 190. 40.6 3 1.8
... with 23 more rows
```



# Caso 2 : Análisis del Consumo de Alcohol en Rusia

## Analisis de Outliers: Q-Q and Ojiva $\chi^2$ – Data Cleaned

```
1 15.7 190. 40.0 3 1.0
> dfa_clean <- dfa[-idx_excluido, ]
>
> # Distancia de Mahalanobis
> dm2 <- mahalnobis(dfa_clean, colMeans(dfa_clean), cov(dfa_clean))
> plot(sort(dm2), ppoints(nrow(dfa_clean)), xlab="DM al cuadrado ordenada",
+       ylab="Probabilidad Acumulada")
> abline(v = qchisq(p,dof), col = "red")
>
> idx_outliers <- which(dm2 > k)
> idx_outliers
[1] 200 261 282 364 411 419 446 449 501 514 522 540 586 595 603 667 689 770 851 995 1008 1013
[23] 1076 1094 1097 1157 1172 1175 1238 1253 1256 1338
>
> # QQ-plot dfa_clean:
> x <- qchisq(ppoints(nrow(dfa_clean)), dof)
> y <- dm2
> qqplot(x, y, main=expression("Q-Q plot para"~{\chi^2}[nu==6]))
> abline(0, 1, col="red")
>
```



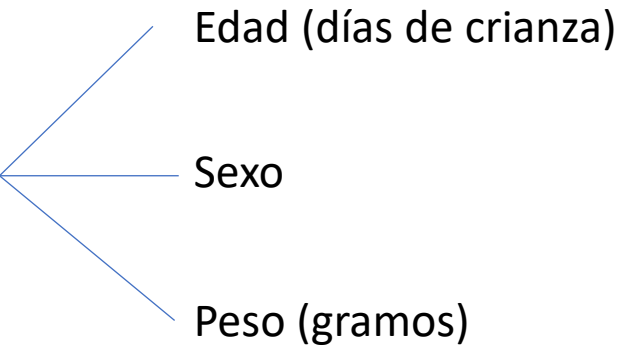


# Caso 3: Control de Pesos en la Crianza de Aves

Grupo Santa Elena



Su principal actividad es la crianza de pollos y la comercialización de la carne en kilogramos.



El proceso de crianza de pollos tiene la **misión** de **maximizar el peso corporal del ave con eficiencia** (días y usos de recursos) cumpliendo los estándares de calidad establecidos



## Cartilla de Registro de Control de Crianza

ID_GALPON	SEXO	DIA07	DIA14	DIA21	DIA28	DIA35	DIA38	DIA40	DIA42
20201-2002-11-H	H	183	466	853			2220	2350	2480
20201-2002-11-M	M	178	478	960			2670	2790	2960
20201-2002-12-H	H	183	468	892			2270	2390	2460
20201-2002-12-M	M	183	468	892			2270	2390	2460
20201-2003-01-H	H	173	456	800	1300	1750	2000		2300
20201-2003-01-M	M	173	455	825	1400	2024		2100	
20201-2003-02-H	H	171	443	791	1280	1500			
20201-2003-02-M	M	171	443	791	1280	2068	2300		2640
20201-2003-03-H	H	172	453	808	1280	1771	2080	2150	2380

- Las aves se agrupan en galpones y se clasifican por sexo
- El control del peso es semanal y en los últimos días se acorta a dos días.

# Caso 3: Control de Pesos en la Crianza de Aves

## Procedimiento de Aplicación de MongoDB & Python en el caso



# Caso 3: Control de Pesos en la Crianza de Aves

## Base de datos en Atlas Mongo DB

The screenshot displays the MongoDB Atlas web interface. In the left sidebar, the 'Database' section is selected, showing a list of databases under 'Project 0'. The 'BD\_GRANJAS' database is highlighted, and the 'pesos' collection is visible below it. The main panel shows the 'Find' tab for the 'pesos' collection. At the top, it displays statistics: STORAGE SIZE: 160KB, TOTAL DOCUMENTS: 1235, and INDEXES TOTAL SIZE: 96KB. Below this, there's a filter bar with a placeholder '{ field: 'value' }' and buttons for 'Apply' and 'Reset'. The query results section shows '1-20 OF MANY' results. A sample document is displayed, with fields: `_id: 0`, `ID_GALPON: "20201-2002-11-H"`, and `SEXO: "H"`. Navigation buttons for 'PREVIOUS' and 'NEXT' are at the bottom of the results list.

- Se evidenció la carga de data en Atlas Mongo DB.

# Caso 3: Control de Pesos en la Crianza de Aves

## Codificación Python

```
✓ [1] import pymongo
```

```
✓ [2] import csv  
import pandas as pd
```

```
✓ [3] client = pymongo.MongoClient("mongodb://jgonza
```

```
[5] db = client['BD_GRANJAS']  
list(db.list_collection_names())  
  
['pesos']
```

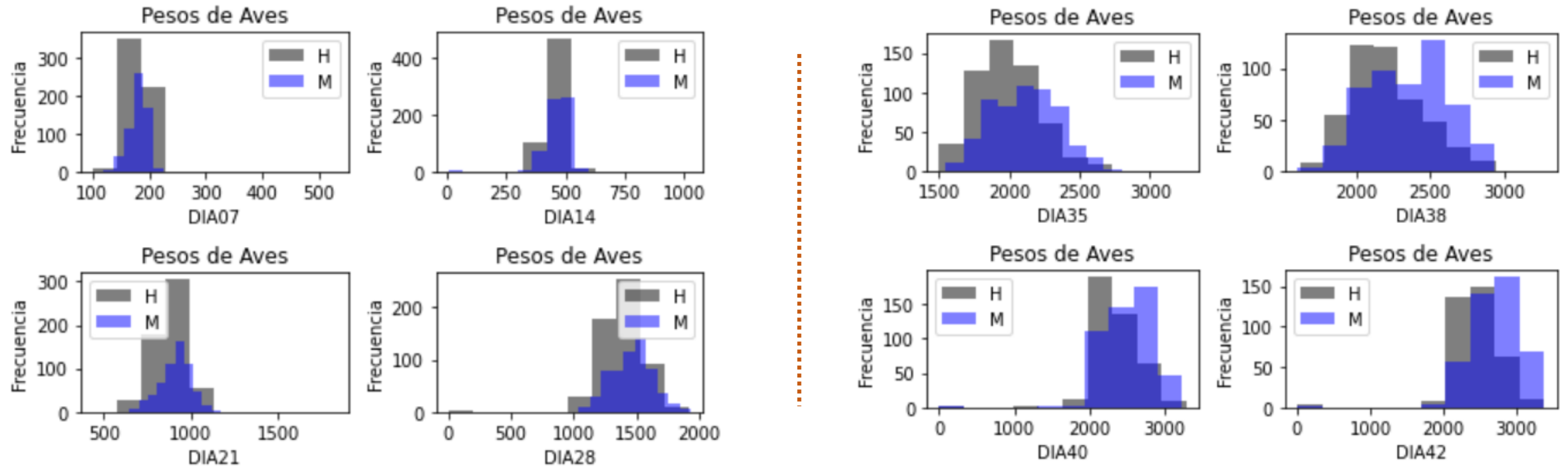
```
▶ #Recuperando documentos de la colección  
mycol = db["pesos"]  
#Comprobando  
for x in mycol.find().limit(5):  
    print(x)
```

```
▶ #Recuperando PESOS de las aves Machos  
myquery = { "SEXO": "M" }  
mydoc = mycol.find(myquery)  
data_list = []  
for x in mydoc:  
    data_list.append(x)  
df_machos = pd.DataFrame(data_list)  
  
#Recuperando PESOS de las aves Hembras  
myquery = { "SEXO": "H" }  
mydoc = mycol.find(myquery) #.limit(5)  
data_list = []  
for x in mydoc:  
    data_list.append(x)  
df_hembras = pd.DataFrame(data_list)  
print(df_hembras.describe())
```

[https://colab.research.google.com/drive/1-XuGPiT2Hlv\\_KU48hFXHGWNnO6S6Mv14?usp=sharing](https://colab.research.google.com/drive/1-XuGPiT2Hlv_KU48hFXHGWNnO6S6Mv14?usp=sharing)

# Caso3: Control de Pesos en la Crianza de Aves

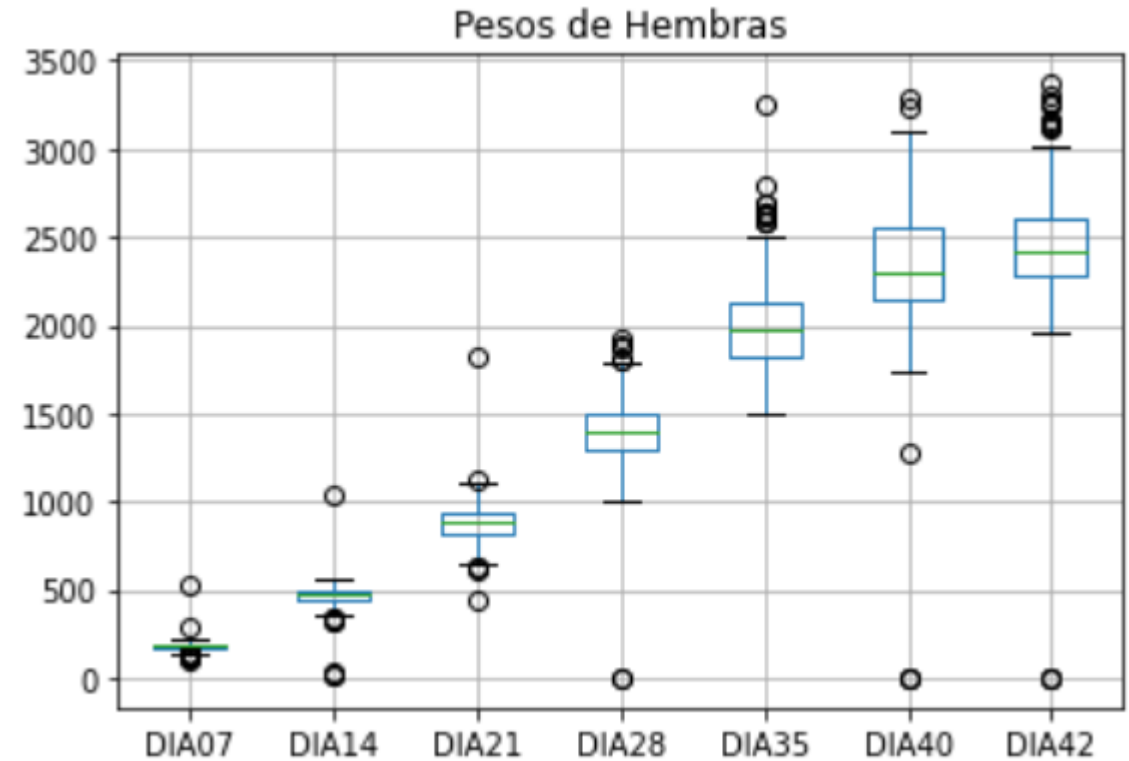
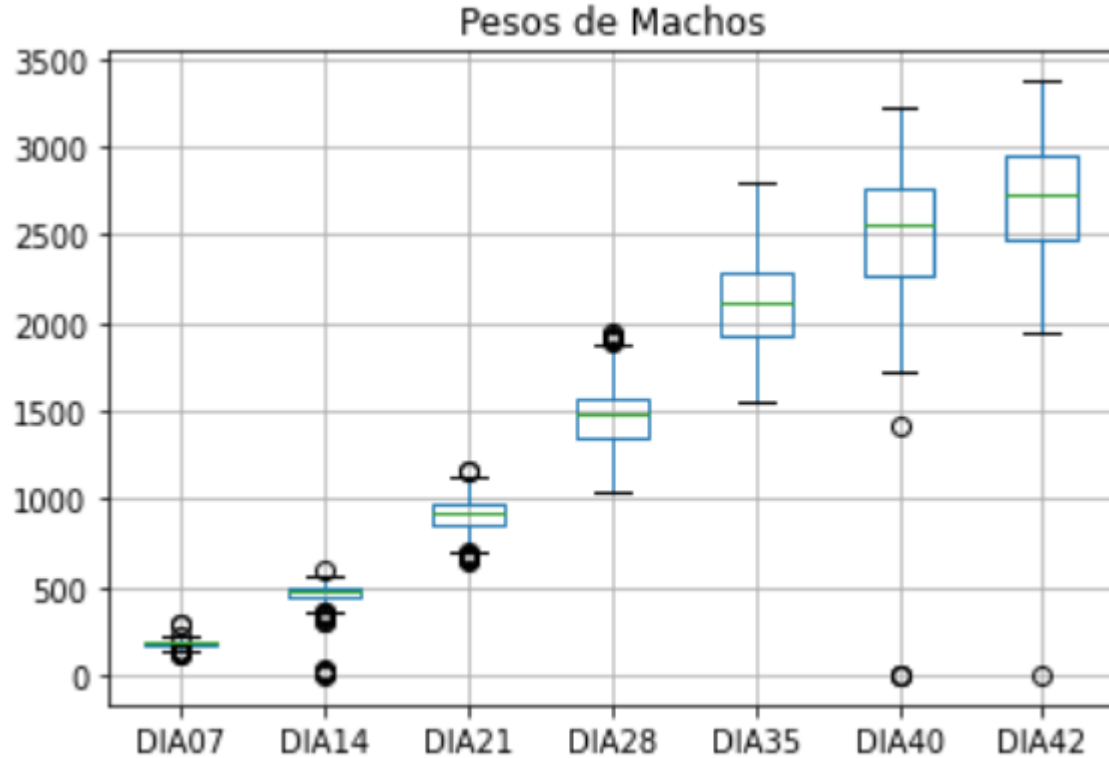
## Análisis Exploratorio: Distribución



- Se observa que en los diferentes controles semanales el peso de las aves machos tienen una mejor distribución normal en comparación con el de las hembras.
- Se evidencia que en las primeras semanas el peso de las aves hembras son ligeramente mayor que el de los machos pero en las últimas semanas se invierte el patrón.

# Caso 3: Control de Pesos en la Crianza de Aves

## Análisis Exploratorio: Outliers



- La data de pesos de las hembras presentan mayor cantidad de outliers que el de los machos.

# Contexto de la data – Venta de autos usados

- Model – Modelos de la marca BMW
- Year – Año de fabricación
- Price – Precio de venta
- Transmission – Tipo de transmisión del vehículo
- Mileage – Millas recorridas
- fuelType – Tipo de combustible
- tax – Impuesto anual
- mpg – Millas por galón
- engineSize – Tamaño del motor en centímetros cúbicos



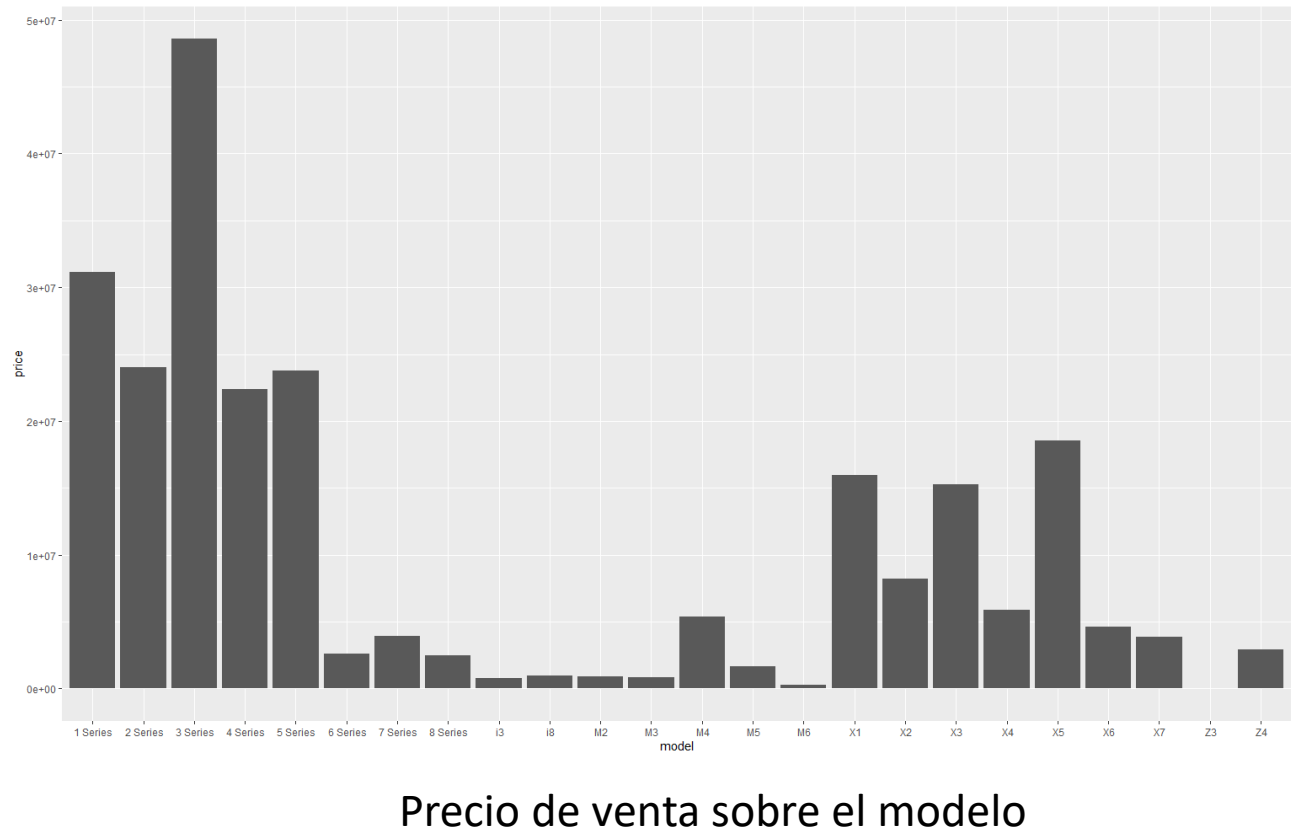
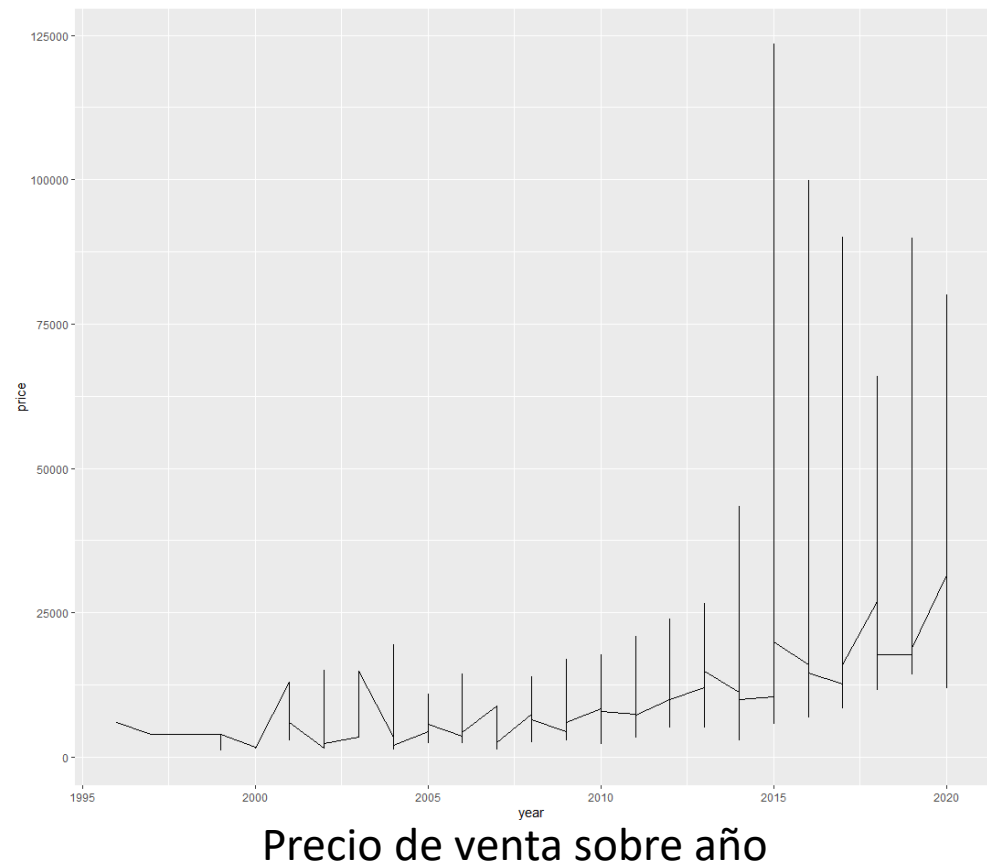
**2020**

# Ejemplo del database

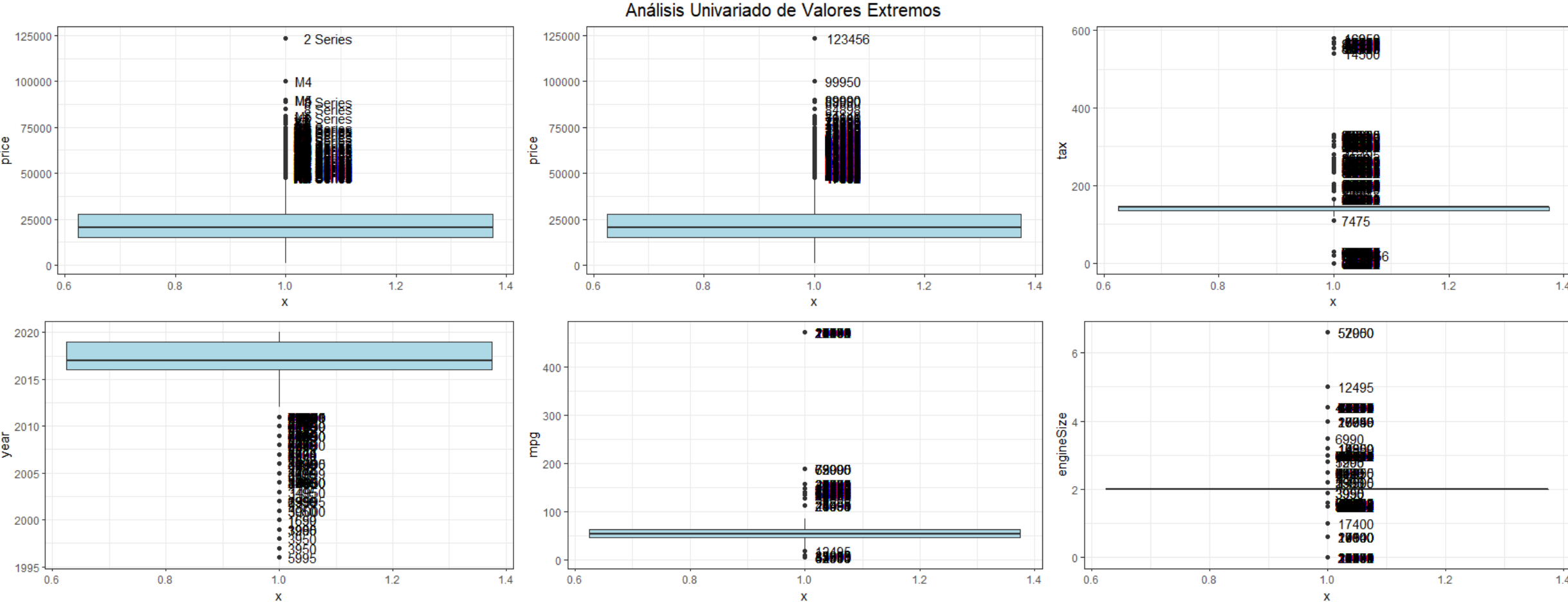
	model	year	price	transmission	mileage	fuelType	tax	mpg	engineSize
1	5 Series	2014	11200	Automatic	67068	Diesel	125	57.6	2.0
2	6 Series	2018	27000	Automatic	14827	Petrol	145	42.8	2.0
3	5 Series	2016	16000	Automatic	62794	Diesel	160	51.4	3.0
4	1 Series	2017	12750	Automatic	26676	Diesel	145	72.4	1.5
5	7 Series	2014	14500	Automatic	39554	Diesel	160	50.4	3.0
6	5 Series	2016	14900	Automatic	35309	Diesel	125	60.1	2.0
7	5 Series	2017	16000	Automatic	38538	Diesel	125	60.1	2.0
8	2 Series	2018	16250	Manual	10401	Petrol	145	52.3	1.5
9	4 Series	2017	14250	Manual	42668	Diesel	30	62.8	2.0
10	5 Series	2016	14250	Automatic	36099	Diesel	20	68.9	2.0
11	X3	2017	15500	Manual	74907	Diesel	145	52.3	2.0



# Análisis exploratorio



# Análisis Univariado de Valores Extremos



# Gracias Totales!!!!

