

TRABAJO PRÁCTICO R
LABORATORIO PARA EL ANÁLISIS DE DATOS
Segundo Semestre 2025

Leonardo Rossi

Introducción

En este trabajo vamos a realizar 3 ejercicios de análisis de datos con el lenguaje *R*, el propósito es evaluar el contenido del curso optativo *Laboratorio para el Análisis de Datos Económicos y Financieros* de la licenciatura en economía de la universidad Torcuato Di Tella

Lo primero que hicimos en el código fue asegurarnos su reproducibilidad ya que en algunos incisos realizamos simulaciones, utilizamos el comando `set.seed()` para poder garantizar esta pseudo-aleatorización.

Luego de esto cargamos las librerías que usamos para los 3 ejercicios, estas fueron

```
library(tidyverse)
library(scales)
library(readr)
library(dplyr)
```

Ejercicio 1: Análisis de Datos

Para este ejercicio teníamos más libertad acerca del tema a investigar. El objetivo era elegir un dataset y Hacernos tres preguntas y contestarlas con las herramientas de tidyverse.

Yo decidí hacer una investigación en el ámbito del deporte, más específicamente, sobre futbol. Las tres preguntas que me planteé fueron:

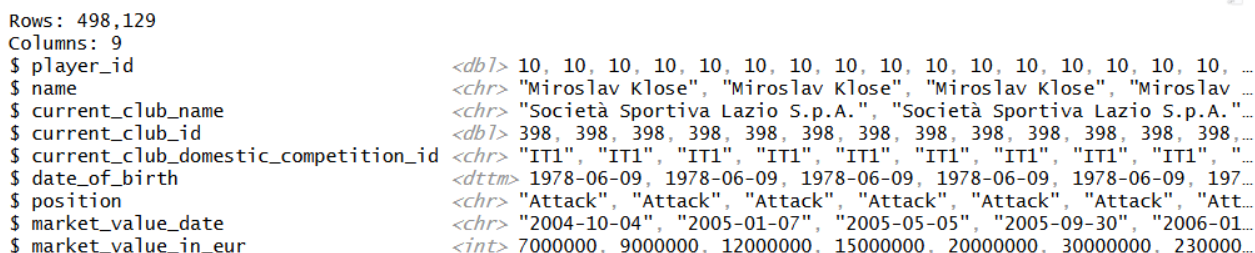
1. ¿Cuál es la relación entre la edad de un jugador y el valor de mercado?
2. ¿Cómo se distribuye el valor de mercado según la posición del jugador?
3. ¿Cuáles son las 10 ligas con mayor valor de mercado total?

Para esto, trabajamos sobre una base de datos de Kaggle llamada *Football Data from Transfermarkt*, que está compuesta por varios archivos CSV con información de jugadores, clubes, valores de mercado y demás. De todos los archivos que componen este dataset, vamos a trabajar con 4 que consideré los esenciales para esta investigación, que contienen la información que mencioné previamente

Lo primero que hicimos fue cargar las bases de datos, inspeccionarlas y limpiarlas. Buscábamos revisar que las variables estuviesen bien categorizadas y que no hubiesen NAs en los datos que nos pudiesen complicar el análisis. Para limpiar las bases de datos las redefinimos y seleccionamos únicamente las variables que nos resultaban relevantes para el análisis. Adicionalmente renombramos algunas variables para evitar que más adelante se generen confusiones.

Primera Pregunta

La primera pregunta que queríamos resolver era cuál era la relación entre la edad de un jugador y su valor de mercado. Para esto vamos a tener que hacer un `left_join()` entre los dataframes de jugadores y de valuaciones. Este es el glimpse de nuestro dataframe después del join



```
Rows: 498,129
Columns: 9
$ player_id      <dbl> 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, ...
$ name          <chr> "Miroslav Klose", "Miroslav Klose", "Miroslav Klose", "Miroslav ...
$ current_club_name <chr> "Società Sportiva Lazio S.p.A.", "Società Sportiva Lazio S.p.A."...
$ current_club_id <dbl> 398, 398, 398, 398, 398, 398, 398, 398, 398, 398, 398, 398, 398, ...
$ current_club_domestic_competition_id <chr> "IT1", "IT1", "IT1", "IT1", "IT1", "IT1", "IT1", "IT1", "IT1", "IT1", ...
$ date_of_birth  <dtm> 1978-06-09, 1978-06-09, 1978-06-09, 1978-06-09, 1978-06-09, 197...
$ position      <chr> "Attack", "Attack", "Attack", "Attack", "Attack", "Attack", "Att...
$ market_value_date <chr> "2004-10-04", "2005-01-07", "2005-05-05", "2005-09-30", "2006-01...
$ market_value_in_eur <int> 7000000, 9000000, 12000000, 15000000, 20000000, 30000000, 230000...
```

Figura 1: Joined Dataframe Glimpse

Lo que notamos es que no tenemos una variable que represente la edad de los jugadores, sino que por un lado tenemos una variable que representa la fecha de nacimiento de cada jugador, y por otro lado tenemos una variable que nos da la fecha en la que se realizó una valuación de mercado en particular.

Como cada jugador tiene varias observaciones, ya que se realizan múltiples valuaciones de mercado por año, durante muchos años, queremos calcular la edad para cada jugador al momento en que se realizó cada valuación

de mercado en particular. Esto nos permite incluir en nuestro análisis a jugadores que ya estén retirados y nos ayuda a mantener coherencia temporal.

```
df_player_valuations <- df_player_valuations %>%
  mutate(
    age_at_valuation = floor(time_length(interval(date_of_birth,
                                                    market_value_date), "years"))
  )

df_player_valuations <- df_player_valuations %>%
  filter(!is.na(age_at_valuation) & !is.na(market_value_in_eur))
```

Esto lo que hace es generar un intervalo entre la fecha de nacimiento y la fecha de valuación de mercado, codificar ese intervalo como un intervalo de tiempo en años y redondearlo para tener únicamente números enteros.

Ahora podemos agrupar los datos por edad y calcular el promedio de valor de mercado, una forma de ver esto gráficamente es usando *ggplot*, realizando el gráfico obtenemos este resultado.

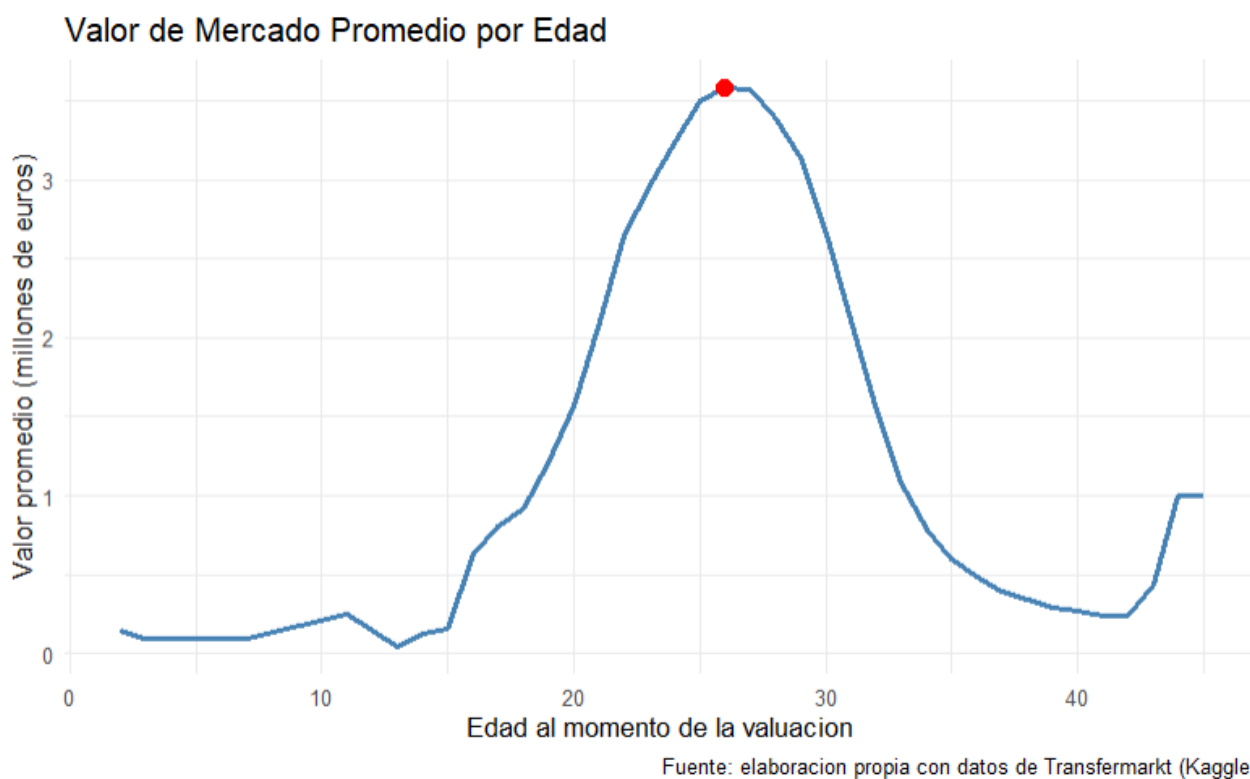


Figura 2: Valor de Mercado Promedio por Edad

Lo que podemos ver es que el valor de mercado promedio crece de forma acelerada durante los primeros años de carrera, alcanzando su punto máximo alrededor de los 25 años, momento en que los jugadores combinan experiencia y potencial de crecimiento. A partir de esa edad, el valor tiende a descender gradualmente, reflejando el envejecimiento deportivo y la pérdida de proyección futura. El leve aumento a edades avanzadas se explica por la presencia de algunos futbolistas excepcionales que prolongan su carrera más allá de lo habitual.

Segunda Pregunta

Para esta pregunta queríamos explorar la relación entre el valor de mercado y la posición de jugador. Como vemos en el dataframe original teníamos dos categorías posicionales: *position* y *subposition*. Si quisiéramos hacer el gráfico tomando en cuenta las subposiciones serían demasiadas como para tener algo legible y ordenado. Así que vamos a trabajar tomando solo las posiciones generales, es decir, portero, defensa, medio y delantero

Vamos a analizar esta información usando un boxplot para ver las medias, los cuartiles, los extremos y los outliers de nuestra data. Como hay mucho desvío en los valores de mercado (desde 100.000 hasta 180.000.000) vamos a usar una escala logarítmica, para que no se vea tan aplastada la información.

Como hay algunos jugadores que no tienen posición disponible, filtramos el dataframe para que no hayan NAs en la posición y para que la valuación de mercado sea mayor a 0. Una vez hecho esto obtenemos el siguiente resultado

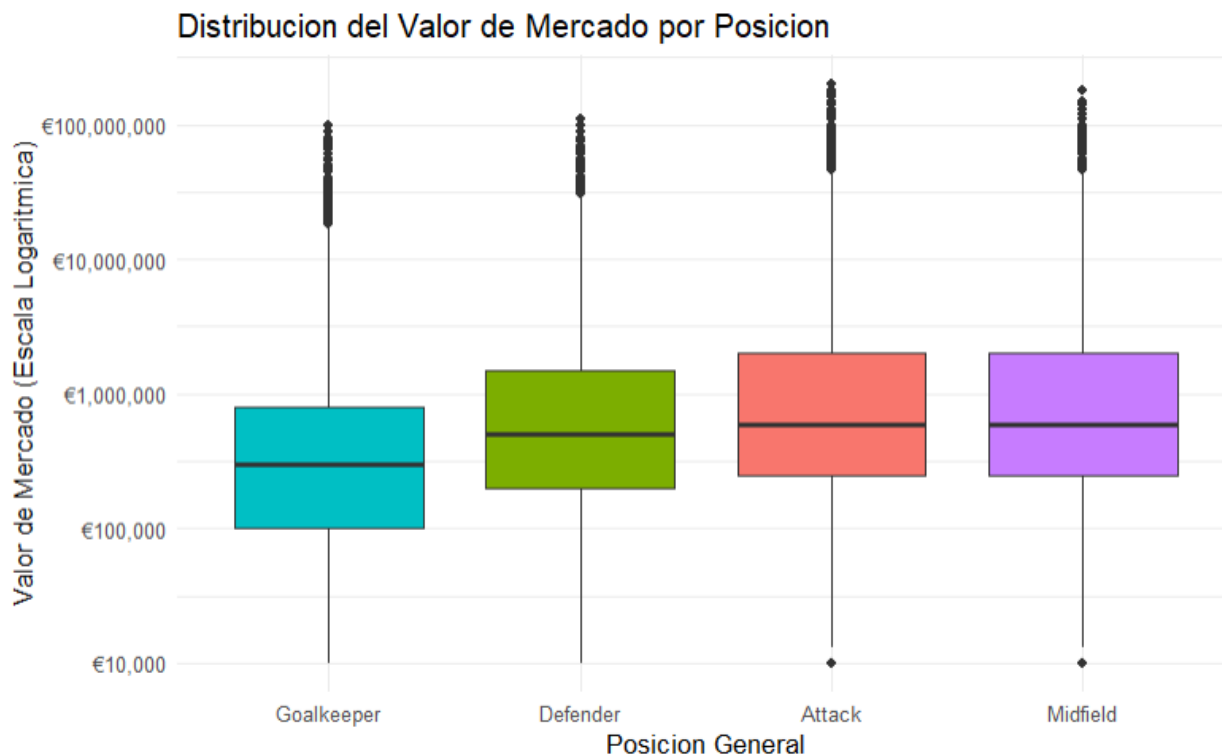


Figura 3: Distribución de valor de mercado por posición

Lo que podemos ver es que los porteros son los que en promedio tienen menor valor de mercado y que los mediocampistas son por poco la posición que mayor valor de mercado tiene en promedio, sin embargo, los delanteros parecieran tener una mayor concentración de outliers. Por eso es que después vemos en delanteros valores de hasta 180 o 200 millones de euros.

Tercera Pregunta

Por últimos queremos saber cuáles son las 10 ligas con mayor valor de promedio total. Lo más probable es que el top 5 sean las que popularmente se conocen como el "Big 5" que son las 5 ligas de mayor popularidad o visibilidad a nivel mundial, estas son:

1. Premier League (Inglaterra)
2. La Liga (España)
3. Serie A (Italia)
4. Bundesliga (Alemania)
5. Ligue 1 (Francia)

Queremos ver si podemos corroborar esto con los datos que tenemos y cuáles serán las 5 ligas cuyos jugadores tengan una mayor valuación de mercado total por fuera de este top 5.

Lo primero que tenemos que hacer es la base de datos, podemos utilizar el dataframe que usamos para las preguntas anteriores, y les hacemos un `join()` con las bases de datos de clubes y competiciones.

```
df_completo <- df_player_valuations %>%
```

```
# Unimos el df de clubes
```

```
left_join(clubs_clean, by = c("current_club_id" = "club_id")) %>%
```

.

```
# Unimos el df de competiciones
```

```
left_join(competitions_clean, by = c("domestic_competition_id" = "competition_id"))
```

Ahora, si agrupamos las ligas por valor de mercado vemos que vamos a sumar varias observaciones para el mismo jugador, y adicionalmente vamos a mezclar jugadores recientes con jugadores retirados, lo que no tendría mucho sentido económicamente. Para tomar una "foto" de la temporada más reciente, vamos a usar la columna que de *market_value_date*. En lugar de filtrar por jugadores *.activos*, vamos a filtrar por valoraciones que ocurrieron en el año más reciente. El código para esta agrupación será

```
top_10_ligas_recientes <- df_completo %>%
```

```
# 1. Nos aseguramos que la fecha este bien definida
mutate(market_value_date = ymd(market_value_date)) %>%
```

```
# 2. Filtramos para quedarnos solo con el 'year' mas reciente.
# Solo va a incluir las filas donde el 'year' de market_value_date
# sea igual al 'year' MAXIMO encontrado en toda la columna.
```

```
filter(year(market_value_date) == max(year(market_value_date),
na.rm = TRUE)) %>%
```

```
# 3. Nos aseguramos de tener solo un valor por jugador por cada 'year'
group_by(player_id) %>%
```

```
# 4. Desagrupamos para poder agrupar de nuevo
slice_max(order_by = market_value_date, n = 1, with_ties = FALSE) %>%
ungroup() %>%
```

```
# 5. Agrupamos por el nombre de la liga
group_by(comp_name) %>%
```

```
# 6. Sumamos el valor de mercado
summarise(
  valor_total_liga = sum(market_value_in_eur, na.rm = TRUE)
) %>%
```

```
# 7. Limpiamos ligas sin nombre o sin valor
filter(valor_total_liga > 0 & !is.na(comp_name)) %>%
```

```
# 8. Ordenamos y tomamos el top 10
arrange(desc(valor_total_liga)) %>%
head(10)
```

Esto soluciona automáticamente el problema de los jugadores retirados, ya que no tendrían ninguna valoración en el último año. Graficando obtenemos:

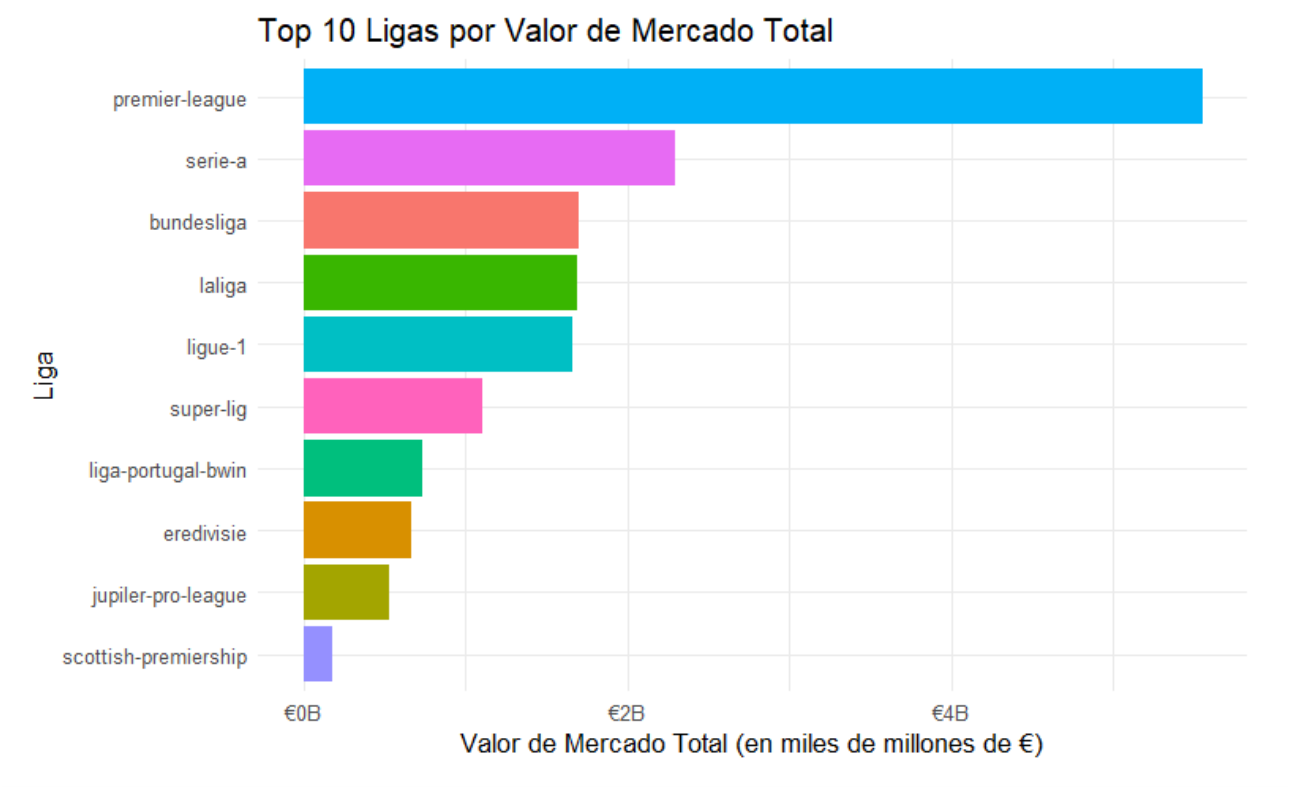


Figura 4: Top 10 Ligas por Mercado de Valor Total

Vemos que efectivamente las 5 ligas con mayores valores de mercado son las que esperábamos que fueran, el resto de los lugares los ocupan las ligas de: Turquía, Portugal, Países Bajos, Bélgica, Escocia.

Ejercicio 2: Análisis Econométrico

Para este ejercicio trabajaremos con el dataset *gapminder*, que contiene información sobre ingresos, esperanza de vida y otros indicadores socioeconómicos de distintos países a lo largo del tiempo. El objetivo es practicar herramientas de regresión, análisis de correlaciones y visualización de datos en R.

Parte 1: Ingreso por persona

Empezamos cargando los datos de *gapminder* y filtrándolos para Argentina, posteriormente hacemos uso del comando *glimpse()* para revisar que esté todo bien definido

Inciso 1

Primero queremos graficar la evolución en el tiempo del ingreso por persona. Usando *ggplot()* obtenemos el siguiente gráfico:



Figura 5: Evolución del ingreso por persona en Argentina

El gráfico muestra que el ingreso por persona en Argentina tuvo una tendencia creciente a largo plazo entre 1960 y 2010, aunque con fuertes altibajos. Se observan varios períodos de crisis y recuperación, especialmente hacia fines de los 70, los 80 y principios de los 2000.

Tras la crisis de 2001, el ingreso crece con fuerza, alcanzando su nivel más alto al final del período. En conjunto, el gráfico refleja una economía que logra crecer, pero con una alta volatilidad e inestabilidad macroeconómica.

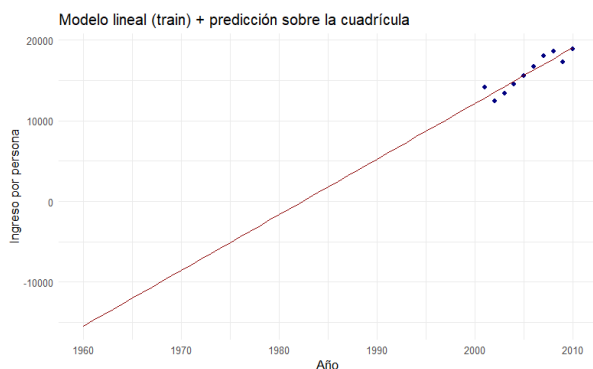
Inciso 2

Ahora queremos hacer un modelo de tratamiento y control (train-test, donde el entrenamiento son los últimos 10 años de los que tenemos datos para la argentina. Vamos a realizar regresiones de distintos modelos sobre la variable *income_per_person*. Empezamos con un modelo lineal

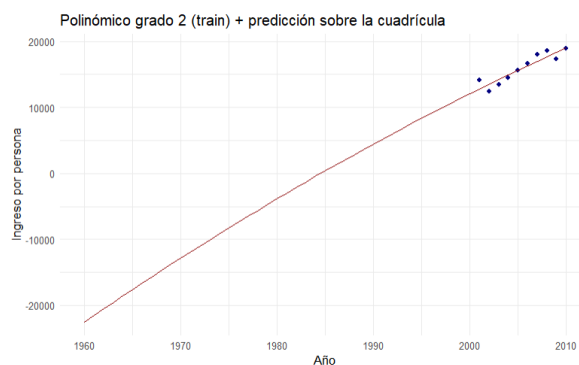
$$y = \beta_0 + \beta_1 t + \varepsilon$$

Lo primero que hacemos es separar la base de datos del train y del test. Una vez que tenemos los datos, fitteamos el modelo con los datos del train y predecimos los resultados del modelo con los datos de test. Estas predicciones son las que utilizamos para graficar.

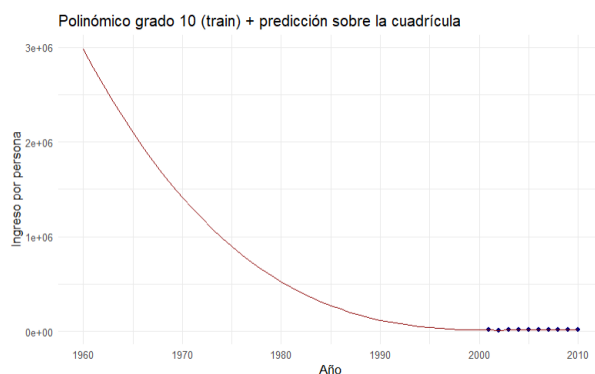
Luego repetimos los mismos pasos para un modelo polinómico de grado 2 y un modelo polinómico de grado 10. Los graficos de estos 3 modelos son:



(a) Regresión del ingreso por persona bajo modelo lineal



(b) Regresión del ingreso por persona bajo modelo polinómico de grado 2



(c) Regresión del ingreso por persona bajo modelo polinómico de grado 10

Figura 6: Regresiones del ingreso por persona en modelo de tratamiento y control.

Lo que podemos ver es que el modelo lineal captura la tendencia general pero no las variaciones. El polinómico de grado 2 mejora el ajuste al permitir curvatura. Finalmente el modelo polinómico de grado 10 puede sobreajustar: se adapta demasiado a los datos de entrenamiento pero predice mal el test.

Inciso 3

Vamos a elegir 4 países sudamericanos distintos de argentina para analizar sus correlaciones respecto al ingreso por persona. En este caso usaremos:

1. Brasil
2. Chile
3. Colombia
4. Uruguay

Empezamos creando un vector que incluya los nombres de los países con los que vamos a trabajar y creamos la matriz de correlaciones usando el comando `cor()`

```
matriz_cor <- gapminder %>%
  filter(country %in% paises) %>%
  arrange(year) %>%
  select(country, year, income_per_person) %>%
  pivot_wider(names_from = country, values_from = income_per_person) %>%
```



```
select(-year) %>%  
cor(use = "pairwise.complete.obs")
```

Esto nos da como resultado la siguiente matriz

	Argentina	Brasil	Chile	Colombia	Uruguay
Argentina	1.0000000	0.7951110	0.7650413	0.8073113	0.8291831
Brasil	0.7951110	1.0000000	0.7717164	0.9600286	0.8713498
Chile	0.7650413	0.7717164	1.0000000	0.8962301	0.9407932
Colombia	0.8073113	0.9600286	0.8962301	1.0000000	0.9504906
Uruguay	0.8291831	0.8713498	0.9407932	0.9504906	1.0000000

Cuadro 1: Matriz de Correlaciones entre los ingresos por persona en sudamérica

Ahora queremos una matriz de correlaciones entre las variaciones porcentuales anuales de dichos ingresos. Para esto tenemos que calcular primero el crecimiento interanual

$$\frac{Y_t - Y_{t-1}}{Y_{t-1}} \times 100$$

Vamos a hacerlo agrupando los datos por país y usando el comando *lag()*

```
mat_crec <- gapminder %>%  
  filter(country%in%países) %>%  
  arrange(country, year) %>%  
  mutate(  
    growth = (income_per_person -  
              lag(income_per_person)) / lag(income_per_person) * 100) %>%  
  select(country, year, growth) %>%  
  pivot_wider(names_from=country, values_from = growth) %>%  
  select(-year) %>%  
  cor(use="pairwise.complete.obs")
```

Esto nos da la siguiente matriz de correlaciones como resultado

	Argentina	Brasil	Chile	Colombia	Uruguay
Argentina	1.0000000	0.2720522	0.1691989	0.3996597	0.5127562
Brasil	0.2720522	1.0000000	0.8262895	0.9649538	0.7580258
Chile	0.1691989	0.8262895	1.0000000	0.8835025	0.7630475
Colombia	0.3996597	0.9649538	0.8835025	1.0000000	0.7766619
Uruguay	0.5127562	0.7580258	0.7630475	0.7766619	1.0000000

Cuadro 2: Matriz de Correlaciones entre las variaciones interanuales de los ingresos

En la primera matriz (niveles), todas las correlaciones son muy altas (cercanas a 1), porque los ingresos de todos los países tienden a crecer en el tiempo y comparten una tendencia común. Esto muestra tendencias similares de largo plazo. En la segunda matriz (variaciones), las correlaciones son bastante menores, ya que las fluctuaciones anuales dependen de shocks específicos de cada país (crisis, políticas, tipo de cambio, etc.). Aquí se observa mayor independencia o asincronía entre economías.

Parte 2: Esperanza de vida

En esta parte del ejercicio vamos a trabajar para un año en particular, usando todos los países del dataset. Vamos a utilizar el año 2010 que es el más reciente entre los datos. El filtro para los datos será:

```
data_2010 <- gapminder %>%  
  filter(year == "2010")
```

Después de usar el comando *glimpse()* en los datos podemos ver que hay que hacer un par de correcciones. Primero, hay valores en las variables de *life_expectancy* que aparecen como -999 estos son valores faltantes que también vamos a querer excluir de nuestros análisis. Segundo, la columna de *life_expectancy_female* está categorizada como carácter y no de manera numérica.

Inciso 5

Lo primero que nos pide esta parte del ejercicio es graficar *life_expectancy* frente a *life_expectancy_female*. Usando *ggplot()* conseguimos esto fácilmente

```
ggplot(data_filter , aes(x = life_expectancy_female , y = life_expectancy)) +  
  geom_point() +  
  labs(  
    x = "Esperanza de vida femenina",  
    y = "Esperanza de vida total",  
    title = "Relacion entre esperanza de vida total y femenina (2010)"  
  ) +  
  theme_minimal()
```

El resultado será el siguiente gráfico:

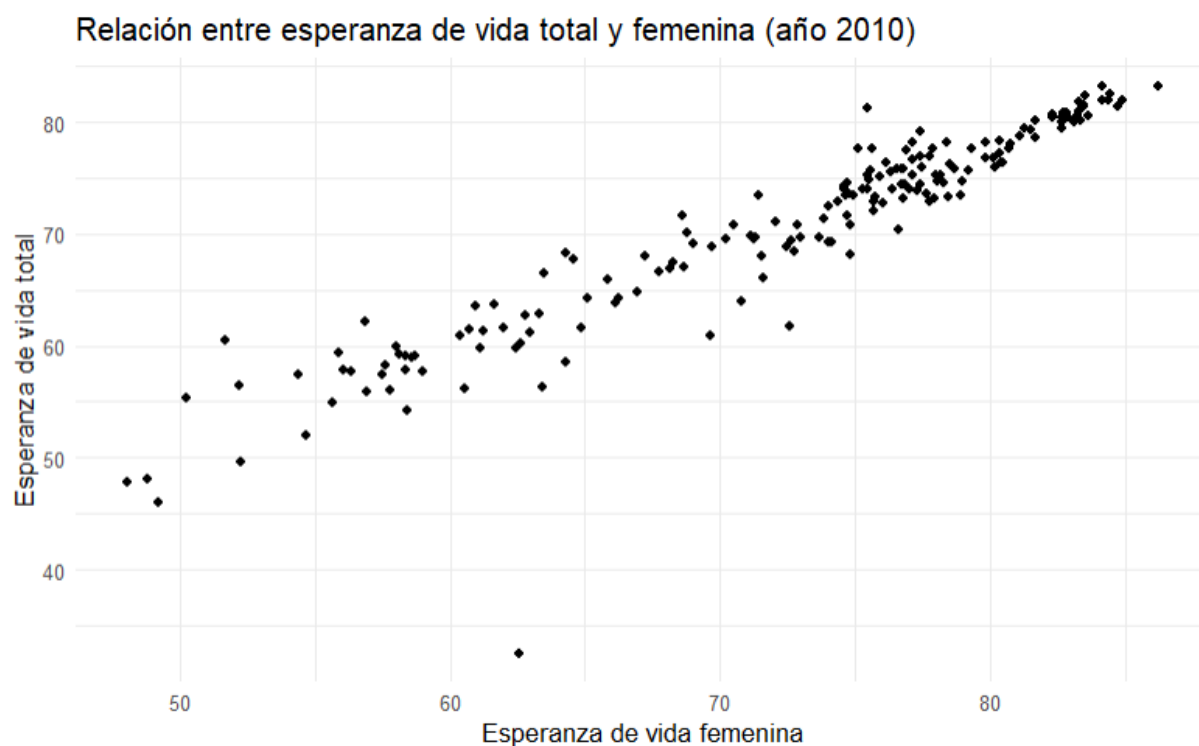


Figura 7: Relación entre esperanza de vida total y esperanza de vida femenina

El gráfico evidencia una fuerte relación positiva entre la esperanza de vida total y la esperanza de vida femenina. En general, las mujeres presentan una mayor esperanza de vida en todos los países analizado. Pareciera además que entre mayor es la esperanza de vida total, más correlación tiene con la esperanza de vida femenina

Inciso 6

Para este inciso queremos estimar una regresión simple de las variables *life_expectancy* como variable independiente usando como variable explicativa *life_expectancy_female*. Al correr la regresión con el comando *lm()* y hacer un summary de la misma obtenemos los siguientes resultados

```
Call:
lm(formula = life_expectancy ~ life_expectancy_female, data = data_filter)

Residuals:
    Min       1Q   Median       3Q      Max
-29.1941  -0.9869   0.2541   1.2947   8.6041

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)      5.2225     1.8477   2.827  0.00523 **
life_expectancy_female  0.9037     0.0254  35.580 < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.176 on 182 degrees of freedom
Multiple R-squared:  0.8743,    Adjusted R-squared:  0.8736
F-statistic: 1266 on 1 and 182 DF,  p-value: < 2.2e-16
```

Figura 8: Summary de la regresión lineal de life expectancy sobre life expectancy female

Vemos que el R^2 es 0,8743. Este coeficiente elevado implica que la esperanza de vida femenina es un muy buen predictor de la esperanza de vida total. Esto tiene sentido, ya que la esperanza de vida total es una combinación ponderada entre la femenina y la masculina.

Inciso 7

Ahora vamos a realizar un contraste de las siguientes hipótesis:

$$\begin{cases} H_0 : life_expectancy_female = life_expectancy_male \\ H_1 : life_expectancy_female > life_expectancy_male \end{cases}$$

La mejor manera de trabajar con estas hipótesis es con la diferencia entre las expectativas de vida, ya que son dos variables que están emparejadas por país. Hacemos primero la prueba t :

```
t_test <- t.test(data_filter$life_expectancy_female,
                 data_filter$life_expectancy,
                 alternative = "greater",
                 paired=TRUE)

t_test
```

Como el p-value es muy pequeño tenemos suficiente evidencia estadística para rechazar la hipótesis nula. En otras palabras, tenemos suficiente evidencia para decir que la esperanza de vida femenina es mayor a la esperanza de vida total

Inciso 8

Ahora queremos hacer una regresión múltiple de *life_expectancy* sobre las variables *life_expectancy_female* e *income_per_person*. Primero corremos la regresión lineal simple:

```
Call:
lm(formula = life_expectancy ~ life_expectancy_female + income_per_person,
    data = data_filter)

Residuals:
    Min       1Q   Median       3Q      Max
-29.0815  -1.0202   0.2511   1.2641   8.1949

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    7.403e+00  2.111e+00   3.508  0.00057 ***
life_expectancy_female 8.663e-01  3.096e-02  27.980 < 2e-16 ***
income_per_person   3.202e-05  1.542e-05   2.077  0.03919 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.148 on 181 degrees of freedom
Multiple R-squared:  0.8772,    Adjusted R-squared:  0.8759
F-statistic: 646.6 on 2 and 181 DF,  p-value: < 2.2e-16
```

Figura 9: Summary de la regresión lineal de life expectancy sobre life expectancy female e income per person

Podemos ver que al incluir el ingreso por persona, el R^2 aumenta solo muy ligeramente, esto nos dice que el ingreso también influye, pero una vez controlado por el nivel de vida de las mujeres, su aporte adicional es pequeño. Por tanto, sí mejora ligeramente el ajuste incluir *income_per_person*, pero no cambia sustancialmente la relación base.

Inciso 9

El objetivo de este inciso es, partiendo de excluir la esperanza de vida de las mujeres, que ya vimos que tiene un gran aporte al poder explicativo de la esperanza de vida total, construir una regresión con 3 variables explicativas que nos permita explicar la variable *life_expectancy*.

Las 3 variables que elegiremos son:

1. *income_per_person*: nos puede ayudar a representar el nivel de desarrollo económico individual. Esperamos un coeficiente positivo
2. *child_mortality*: nos ayuda a representar características de la calidad del sistema de salud.
3. *is_oecd*: nos ayuda a representar características del desarrollo institucional

Corremos la regresión:

```
Call:
lm(formula = life_expectancy ~ income_per_person + child_mortality +
    is_oecd, data = data_filter)

Residuals:
    Min       1Q   Median       3Q      Max
-14.9540  -1.6160   0.6176   2.1165   8.9800

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    7.565e+01  6.023e-01 125.610 < 2e-16 ***
income_per_person 5.701e-05  1.961e-05   2.907  0.004112 **
child_mortality  -1.708e-01  8.216e-03 -20.784 < 2e-16 ***
is_oecdTRUE      3.131e+00  8.968e-01   3.492  0.000604 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.821 on 180 degrees of freedom
Multiple R-squared:  0.8201,    Adjusted R-squared:  0.8171
F-statistic: 273.5 on 3 and 180 DF,  p-value: < 2.2e-16
```

Figura 10: Nueva regresión lineal

Esto nos dice que estas 3 variables tienen el signo de coeficiente que esperábamos y nos ayudan a explicar una parte importante de la variable de *life_expectancy*, sin embargo el R^2 sigue siendo menor al que teníamos cuando usábamos como única variable explicativa la esperanza de vida femenina.

Ejercicio 3: Simulación Demanda con Preferencias Cobb-Douglas

En este ejercicio estudiaremos la demanda de dos bienes de consumo por parte de los hogares bajo preferencias Cobb–Douglas y cómo reaccionan ante cambios en ingresos y precios.

Para el setup de este problema consideramos dos bienes, x_1 y x_2 , con precios $p_1 > 0$ y $p_2 > 0$. Cada hogar tiene ingreso $Y > 0$ y preferencias representables por una utilidad Cobb-Douglas, esto es:

$$U(x_1, x_2) = x_1^{\alpha_1} x_2^{\alpha_2}$$

Donde $\alpha_1, \alpha_2 \in (0, 1)$ y $\alpha_1 + \alpha_2 = 1$. El hogar elige: (x_1, x_2) sujeto a la restricción presupuestaria habitual $p_1 x_1 + p_2 x_2 \leq Y$

Inciso 1

Primero queremos crear una función que genere el ingreso mensual de un hogar Y que siga una distribución χ_k^2 con k grados de libertad para un n número de hogares

```
simular_ingreso <- function(n, k){  
  ingresos <- rchisq(n, df=k)  
  return(ingresos)  
}
```

Si elegimos un k muy chico los ingresos serán muy desiguales, es decir tendremos una función con mucha densidad en las colas. En cambio, si tenemos un k más elevado la distribución se vuelve más simétrica y los ingresos serán más parejos.

Inciso 2

Ahora, una vez definidos los ingresos de los hogares queremos definir la función de la demanda de los hogares así como la utilidad indirecta. Buscaremos resultados de la forma

$$x_1^* = \frac{\alpha_1 Y}{p_1}, \quad x_2^* = \frac{\alpha_2 Y}{p_2} \tag{1}$$

$$U^* = (x_1^*)^{\alpha_1} (x_2^*)^{\alpha_2} \tag{2}$$

Definimos la función:

```
demanda_cd <- function(Y, p1, p2, alpha1, alpha2){  
  # Demandas  
  x1_opt <- (alpha1*Y)/p1  
  x2_opt <- (alpha2*Y)/p2  
  
  # Utilidad indirecta  
  U_opt <- (x1_opt)^(alpha1)*(x2_opt)^(alpha2)  
  
  return(data.frame(x1_opt, x2_opt, U_opt))  
}
```

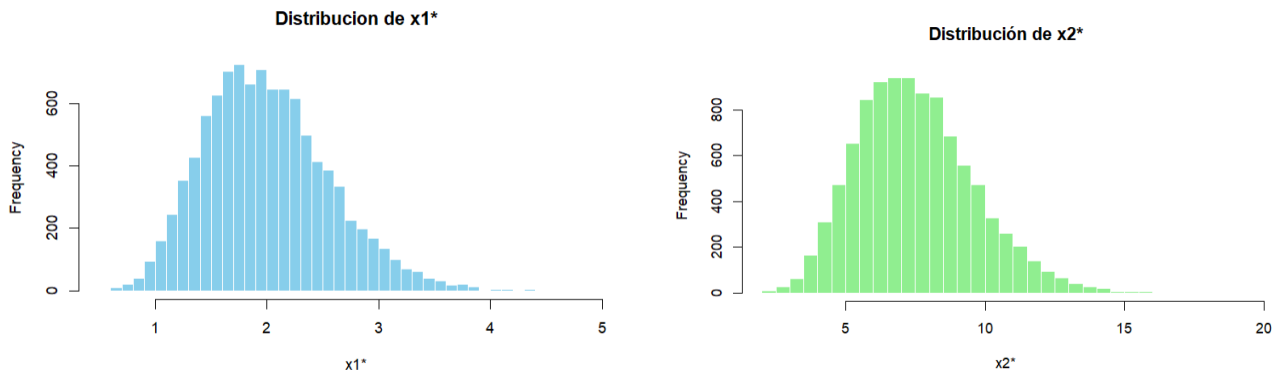
Inciso 3

Vamos a simular 10.000 hogares utilizando las funciones anteriormente definidas.

```
# Definimos los parametros  
n <- 10000  
k <- 25  
p1 <- 5  
p2 <- 2  
alpha1 <- 0.4  
alpha2 <- 0.6
```

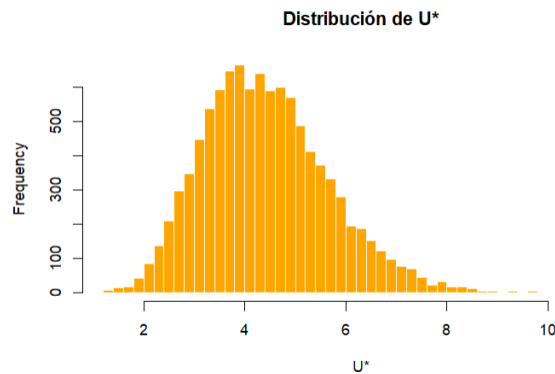
```
Y <- simular_ingreso(n,k)
demandas <- demanda_cd(Y,p1,p2,alpha1,alpha2)
```

Una vez hecha la simulación podemos presentar las distribuciones de los consumos óptimos y de la utilidad óptima como histogramas



(a) Histograma de la densidad del bien 1

(b) Histograma de la densidad consumo del bien 2



(c) Histograma de la densidad de la utilidad óptima

Figura 11: Distribuciones empíricas de los resultados de la simulación.

Ahora presentamos las estadísticas de las distribuciones

<code>x1_opt</code>	<code>x2_opt</code>	<code>U_opt</code>
Min. :0.5443	Min. : 2.041	Min. : 1.203
1st Qu.:1.5903	1st Qu.: 5.964	1st Qu.: 3.515
Median :1.9514	Median : 7.318	Median : 4.313
Mean :1.9995	Mean : 7.498	Mean : 4.419
3rd Qu.:2.3450	3rd Qu.: 8.794	3rd Qu.: 5.183
Max. :5.0812	Max. :19.054	Max. :11.230

Figura 12: Medias y cuartiles de las distribuciones de la simulación

Inciso 4

Para este inciso vamos a querer definir una función que nos devuelve la probabilidad de que la demanda óptima de alguno de los dos bienes (definido por nosotros), sea menor a un umbral arbitrario $c > 0$. Empezamos definiendo la función:

```
prob_bajo_consumo <- function(demandas, j, c){
  # Primero revisamos que j este bien definido
  .
```

```
if(j==1){
  vector_demanda <- demandas$x1_opt
} else if(j==2){
  vector_demanda <- demandas$x2_opt
} else{
  stop("El índice debe ser 1 o 2")
}

# Comparamos el vector de demandas con el parametro c
condicion <- vector_demanda < c
# Calculamos la media de la condicion
probabilidad <- mean(condicion)

return(probabilidad)
}
```

Inciso 5

Queremos simular un shock en el precio del bien 1. Lo primero que tenemos que hacer es definir el nuevo precio que sea 20 % más que el precio original.

Lo primero que hicimos fue repetir las simulaciones del inciso 3 con este nuevo precio. Una vez iterados los ingresos de los hogares ante el shock en el precio, calculamos los consumos óptimos de los bienes y la utilidad óptima post-shock

Lo que podemos ver en la figura 13 es que ahora hay una mayor concentración de hogares que compran menos, esto lo vemos en una mayor densidad cerca del 0 en el histograma rojo. Lo otro que queremos ver es que pasa con la utilidad indirecta, esto no está en el gráfico pero lo calculamos

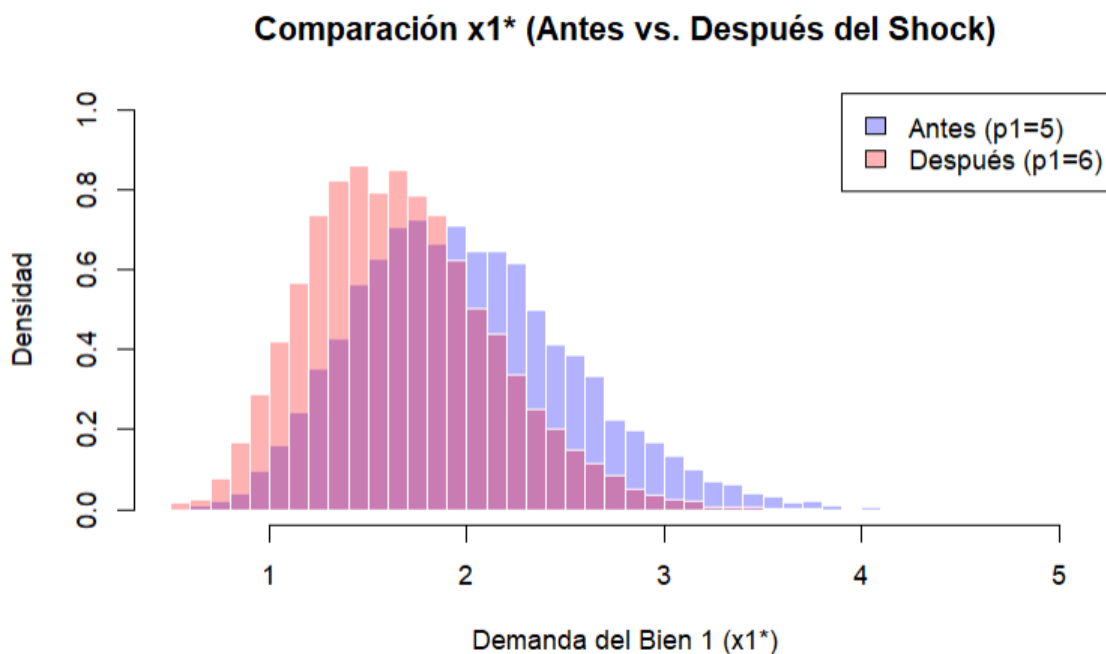
Min	1st Qu.	Median	Mean	3rd Qu.	Max
0.4536	1.3253	1.6262	1.6662	1.9542	4.2343

Cuadro 3: Medias y cuartiles de los óptimos post-shock

Podemos ver que x_1^* bajó con la suba del precio. Esto tiene sentido y era el resultado que esperábamos.

Inciso 6

Para este inciso vamos a presentar un histograma que represente las densidades de x_1^* pre y post shock.

Figura 13: Densidades de x_1^* pre y post shock

El cálculo de las utilidades promedio fue el siguiente

```
# Calculamos los dos promedios
utilidad_antes <- mean(demandas$U_opt)
utilidad_despues <- mean(demandas_shock$U_opt)

# Creamos un dataframe para presentarlo
tabla_utilidad <- data.frame(
  Utilidad_Promedio = c(utilidad_antes, utilidad_despues),
  row.names = c("Antes-del-Shock-(p1=5)", "Despues-del-Shock-(p1=6)")
)

tabla_utilidad
```

El resultado fue la siguiente tabla

	Utilidad Promedio
Antes del Shock ($p_1 = 5$)	4.419142
Después del Shock ($p_1 = 6$)	4.108331

Cuadro 4: Utilidades promedio antes y después del shock de p_1

Lo que podemos ver es que el aumento del precio disminuye la utilidad promedio de los consumidores

Pregunta 7

En este inciso los hogares tienen preferencias heterogéneas, donde cada hogar toma $\alpha_1 \sim \text{Beta}(a, b)$ y $\alpha_2 = 1 - \alpha_1$. Lo primero que tenemos que hacer es redefinir las preferencias de los hogares.

Como la media de una distribución Beta es $\frac{a}{a+b}$ algo lógico sería tomar $a = 4$ y $b = 6$ ya que nos va a dar, en promedio los mismos alfa y beta que tomamos en la simulación del caso homogéneo (0,4 y 0,6).

Usamos el comando `rbeta()` para simular los ponderadores de los hogares

```
a <- 4
b <- 6
```

```
alpha1.het <- rbeta(n,a,b) # usamos el n del caso sin heterogeneidad
alpha2.het <- 1-alpha1.het
```

Una vez definidos los ponderadores, podemos recalcular las demandas pre y post shock para los hogares heterogéneos.

Nuestro objetivo es comparar este escenario con el caso base de preferencias sin heterogeneidad. La mejor forma de hacerlo es analizando como esta dispersión en las preferencias al consumo y a la sensibilidad sobre el shock.

Tenemos calcular la desviación estándar para nuestros 4 escenarios (pre y post shock, distintas preferencias) y presentarla en una tabla para analizar posteriormente.

El cálculo de estos desvíos nos dio de resultado la siguiente tabla

	Caso Base	Caso Shock
1. Preferencias Homogéneas	0.5617531	0.4681276
2. Preferencias Heterogéneas	0.9546750	0.7955625

Cuadro 5: Desvíos estándar pre y post shock para hogares con heterogeneidad

Podemos ver que en el caso base, ante distintos α_1 introducir heterogeneidad en las preferencias hace que el consumo en la población sea, lógicamente, más diverso y disperso. Ya no todos reaccionan igual ante el mismo ingreso.

Adicionalmente, podemos ver que

$$\begin{cases} \text{caída del desvío en homogeneidad} &= 0,0944 \\ \text{caída del desvío en heterogeneidad} &= 0,1592 \end{cases}$$

Es decir que ante la presencia de heterogeneidad en las preferencias, los consumidores son mucho más sensibles al shock del aumento del precio.

Para terminar, vamos a querer comparar el gráfico que hicimos en el inciso anterior pero ahora con preferencias heterogéneas

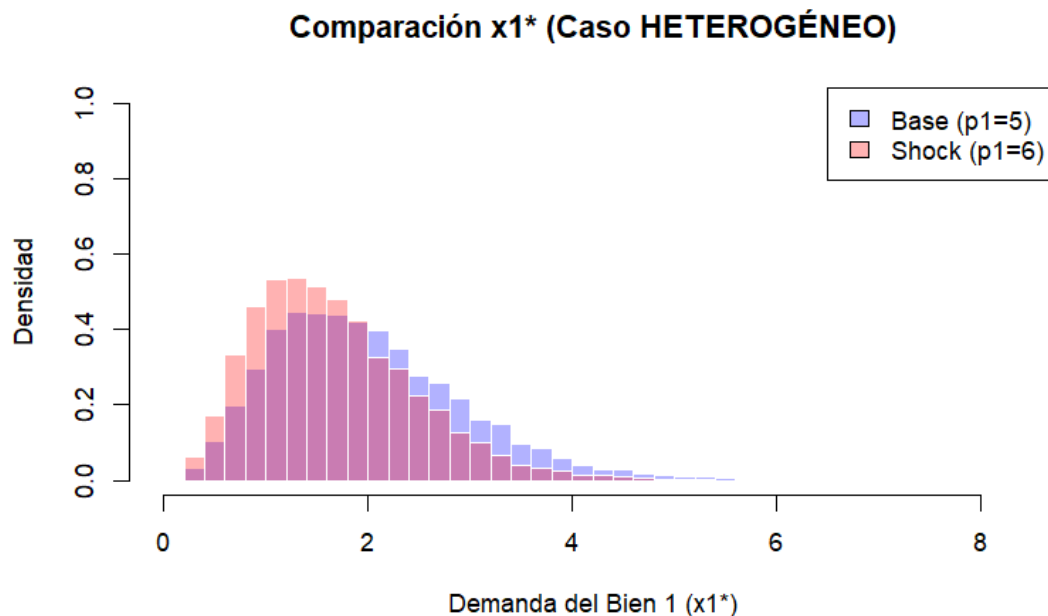


Figura 14: Densidades de x_1^* pre y post shock bajo hogares heterogéneos

Podemos ver que las distribuciones son parecidas al caso de preferencias homogéneas pero mucho más anchas y planas.