

# **Estudio de datos biométricos sobre el ciervo volante.**

Olivia Marcos Diaz de Otazu      Irene Canales Giménez  
Samuel Melián Benito

2025-12-13

## **Índice**

<b>Objetivo del estudio</b>	<b>2</b>
<b>Metodología</b>	<b>3</b>
<b>Resultados</b>	<b>4</b>
<b>Conclusiones</b>	<b>14</b>
<b>Anexo:Código R</b>	<b>15</b>

## **Objetivo del estudio**

## Metodología

## Resultados

```
library(tidyverse)
library(summarytools)
library(GGally)
library(gt)
library(flextable)
library(knitr)
library(corrplot)
```

#Primero leemos el dataset y transformamos las provincias en blanco en “Desconocida”

```
datos <- read.table("./datos/CIERVO.txt", fill=TRUE, dec = ",", na.strings =
  ↪ c("", "NA"), header=TRUE );

datos$PROVINCIA[is.na(datos$PROVINCIA)] <- "Desconocida"

datos$SEXO <- factor(datos$SEXO, levels = c("hembra", "macho"))
datos$PROVINCIA <- factor(datos$PROVINCIA, levels = c("Asturias", "Cantabria",
  ↪ "Otras", "Desconocida"))
```

Ahora pasemos a realizar un análisis descriptivo de la anchura de la cabeza de una muestra de 150 ejemplares elegidos de manera aleatoria, reflejando sus principales características

```
set.seed(123) #Semilla para asegurar la aleatoriedad a la hora de elegir la
  ↪ muestra de individuos y así evitar posibles sesgos

indices <- sample(1:nrow(datos), size = 150, replace = FALSE) #Tomamos la muestra
  ↪ aleatoria de 150 individuos

muestra <- datos[indices, ] #Creamos el dataframe de esta nueva muestra de 150
  ↪ individuos
```

Realizamos un análisis descriptivo del tamaño de la cabeza para la muestra de 150 individuos seleccionando los datos relevantes

```
tabla_kb <- muestra |>
  summarise(
    N = n(),
    Media = mean(KB, na.rm = TRUE),
    Q1 = quantile(KB, 0.25, na.rm = TRUE),
    Mediana = median(KB, na.rm = TRUE),
    Q3 = quantile(KB, 0.75, na.rm = TRUE),
    DE = sd(KB, na.rm = TRUE),
```

```

    IQR = IQR(KB, na.rm = TRUE),
    Min = min(KB, na.rm = TRUE),
    Max = max(KB, na.rm = TRUE)
  )
}

tabla_kb

```

	N	Media	Q1	Mediana	Q3	DE	IQR	Min	Max
1	150	12.13833	9	11.2	15.2	3.684269	6.2	7.1	21.2

```

muestra |> ggplot(aes(KB)) +

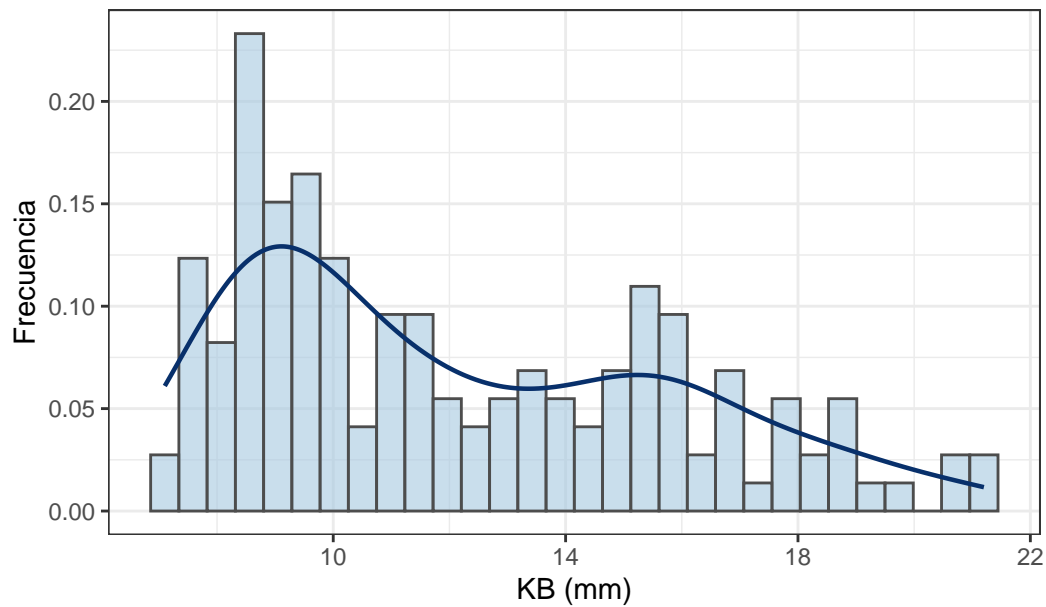
  geom_histogram(
    aes(y = after_stat(density)),
    fill = "#A6C8E0",
    color = "grey30",
    bins = 30,
    alpha = 0.6
  ) +
  geom_density(
    color = "#08306B",
    linewidth = 0.8,
    alpha = 0.5
  ) +

  labs(
    title = "Histograma de KB con curva de densidad",
    x = "KB (mm)",
    y = "Frecuencia"
  ) +

  theme_bw()

```

Histograma de KB con curva de densidad



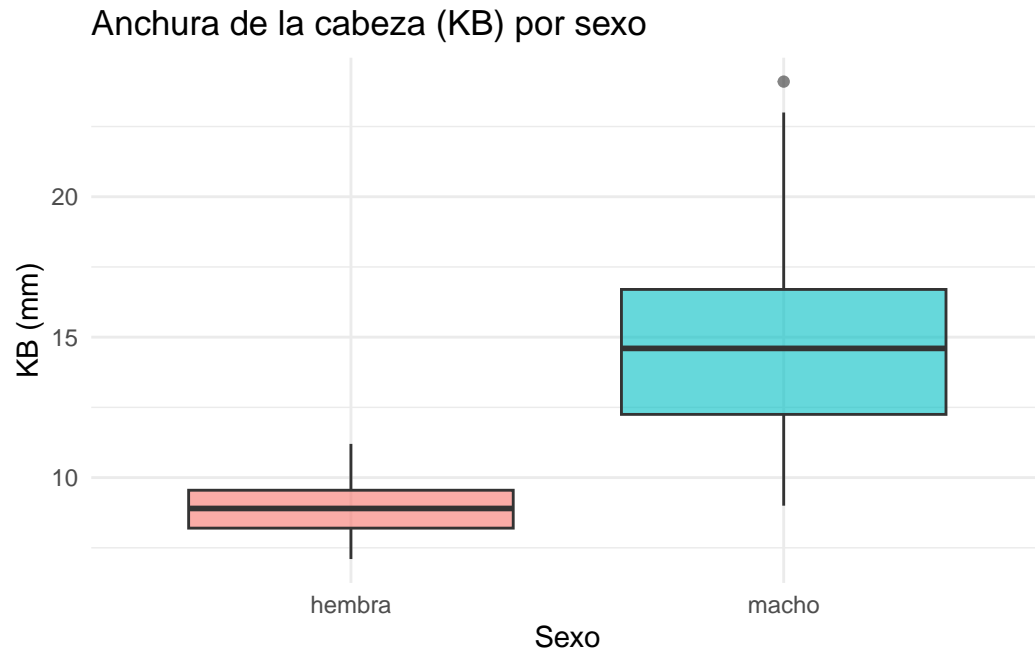
```
tabla_kb_sexo <- datos |>
  group_by(SEX0) |>
  summarise(
    N = n(),
    Media = mean(KB, na.rm = TRUE),
    DE = sd(KB, na.rm = TRUE),
    Mediana = median(KB, na.rm = TRUE),
    IQR = IQR(KB, na.rm = TRUE),
  )
```

```
print(tabla_kb_sexo)
```

# A tibble: 2 x 6

	SEX0	N	Media	DE	Mediana	IQR
	<fct>	<int>	<dbl>	<dbl>	<dbl>	<dbl>
1	hembra	123	8.88	0.913	8.9	1.35
2	macho	142	14.8	3.29	14.6	4.45

```
ggplot(datos, aes(x = SEX0, y = KB, fill = SEX0)) +
  geom_boxplot(alpha = 0.6) +
  labs(
    title = "Anchura de la cabeza (KB) por sexo",
    x = "Sexo",
    y = "KB (mm)"
  ) +
  theme_minimal() +
  theme(legend.position = "none")
```



```
tabla_kb_provincia <- datos |>
  group_by(PROVINCIA) |>
  summarise(
    N = n(),
    Media = mean(KB, na.rm = TRUE),
    DE = sd(KB, na.rm = TRUE),
    Mediana = median(KB, na.rm = TRUE),
    IQR = IQR(KB, na.rm = TRUE),
  )

print(tabla_kb_provincia)
```

```
# A tibble: 4 x 6
  PROVINCIA      N Media  DE Mediana  IQR
  <fct>      <int> <dbl> <dbl>   <dbl> <dbl>
1 Asturias    180  11.9  3.72   10.8  5.75
2 Cantabria    28  13.2  3.67   13.4  6.25
3 Otras       42  11.9  4.53   10.0  3.98
4 Desconocida  15  12.1  3.77   11.2  6.3
```

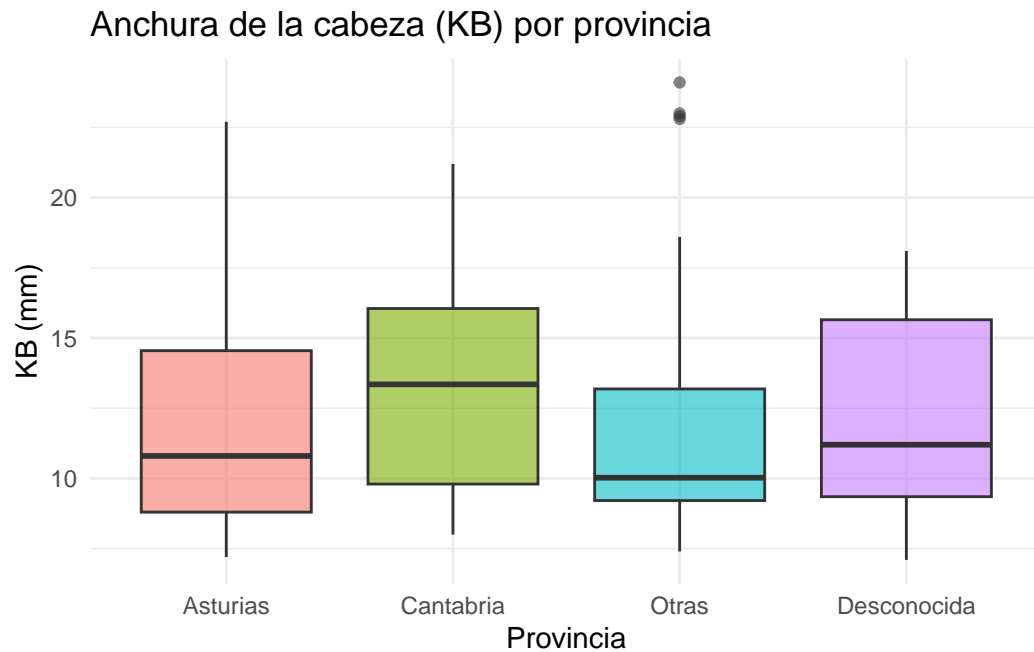
Presentamos ahora un boxplot o diagrama de cajas y bigotes de la anchura de la cabeza para cada provincia

```
ggplot(datos, aes(x = PROVINCIA, y = KB, fill = PROVINCIA)) +
  geom_boxplot(alpha = 0.6) +
  labs(
    title = "Anchura de la cabeza (KB) por provincia",
```

```

  x = "Provincia",
  y = "KB (mm)"
) +
theme_minimal() +
theme(legend.position = "none")

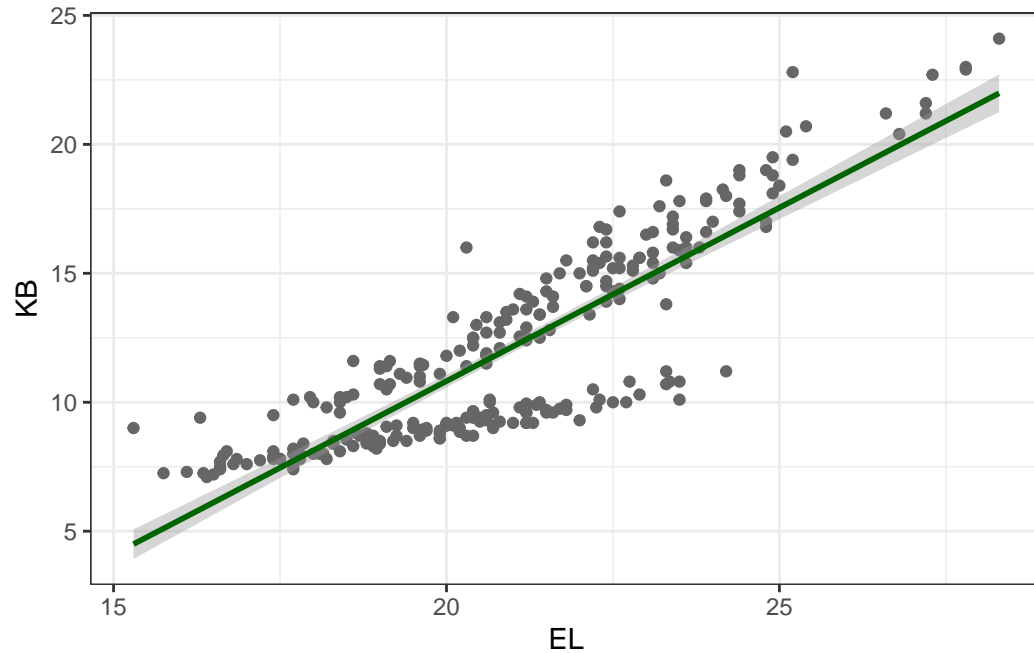
```



```

# Recta de regresión de la variable EL en función de KB
datos |>
  select(KB, EL) |> # Selecciona las columnas 'EL' y 'KB'
  ggplot(aes(x=EL, y = KB)) + # Define las variables para el gráfico
  geom_point(
    color = "grey40",
  ) + # Añade puntos al gráfico
  theme_bw() + # Aplica un tema en blanco y negro
  geom_smooth(
    method = lm,
    se = TRUE,
    color = "darkgreen"
  ) # Añade una línea de regresión lineal sin intervalo de confianza

```



```
# Creación y resumen de un modelo lineal para 'est' en función de 'mg'
modelo <- datos |>
  select(KB, EL) |> # Selecciona las variables 'KB' y 'EL'
  with(lm(KB ~ EL)) # Crea un modelo lineal de 'KB' en función de 'EL'

#Valor de r^2
r2 <- summary(modelo)$r.squared
r2
```

```
[1] 0.7531707
```

```
#Valor de r
r <- sign(coef(modelo)[2]) * sqrt(r2)
r
```

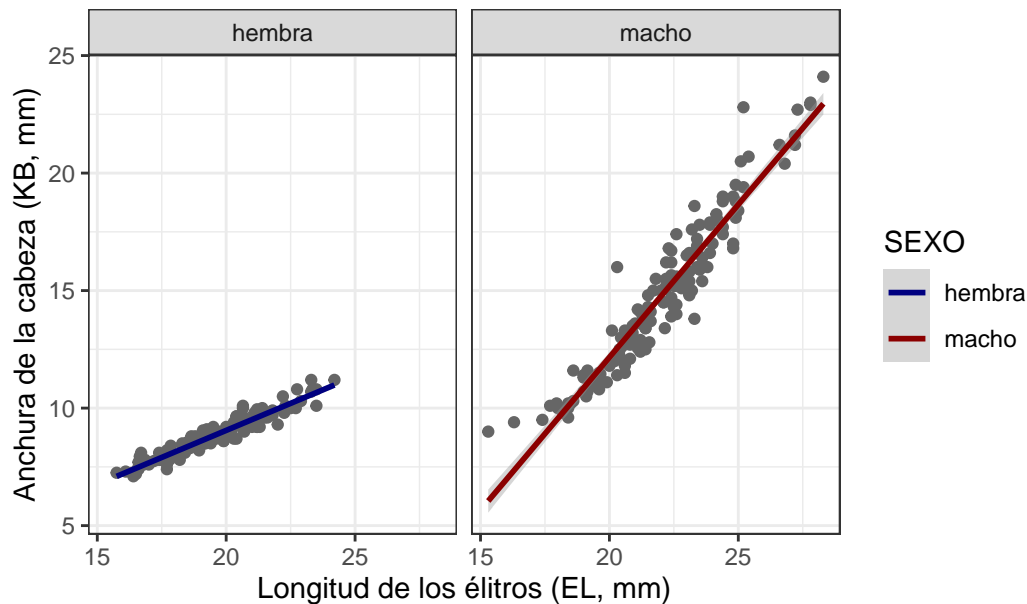
```
EL
0.8678541
```

```
# Recta de regresión de la variable 'est' en función de 'mg' filtrando por tipo
↪ 'C'
datos |>
  # 'filter(tipo == "C")' filtra el dataframe para incluir solo las filas donde
  ↪ la columna 'tipo' es igual a "C".
  ggplot(aes(x = EL, y = KB)) +
  facet_wrap(~ SEXO) +
```

```

geom_point(
  color = "grey40"
) +
labs(
  x = "Longitud de los élitros (EL, mm)",
  y = "Anchura de la cabeza (KB, mm)",
  title = ""
) +
# 'geom_smooth(method = "lm")' añade una línea de ajuste lineal al gráfico.
# ↪ 'method = "lm"' indica que se usa un modelo lineal para el ajuste.
geom_smooth(
  method = "lm",
  aes(color = SEXO),
) +
scale_color_manual(values = c("hembra" = "navy", "macho" = "darkred")) +
theme_bw();

```



```

modelo_hembra <- lm(
  KB ~ EL,
  data = datos,
  subset = SEXO == "hembra"
)
modelo_macho <- lm(
  KB ~ EL,
  data = datos,
  subset = SEXO == "macho"
)

```

```
#Valor de r^2 para la regresion lineal KB ~ EL para hembra y macho por separado
r2_hembra <- summary(modelo_hembra)$r.squared
r2_macho <- summary(modelo_macho)$r.squared

#Valor de r para la regresion lineal KB ~ EL para hembra y macho por separado
r_hembra <- sign(coef(modelo_hembra)[2]) * sqrt(r2_hembra)
r_macho <- sign(coef(modelo_macho)[2]) * sqrt(r2_macho)
```

#Crearemos una tabla que tenga el valor de r y  $r^2$  para la regresión de hembra y lo mismo para la de macho (tabla 2x2)

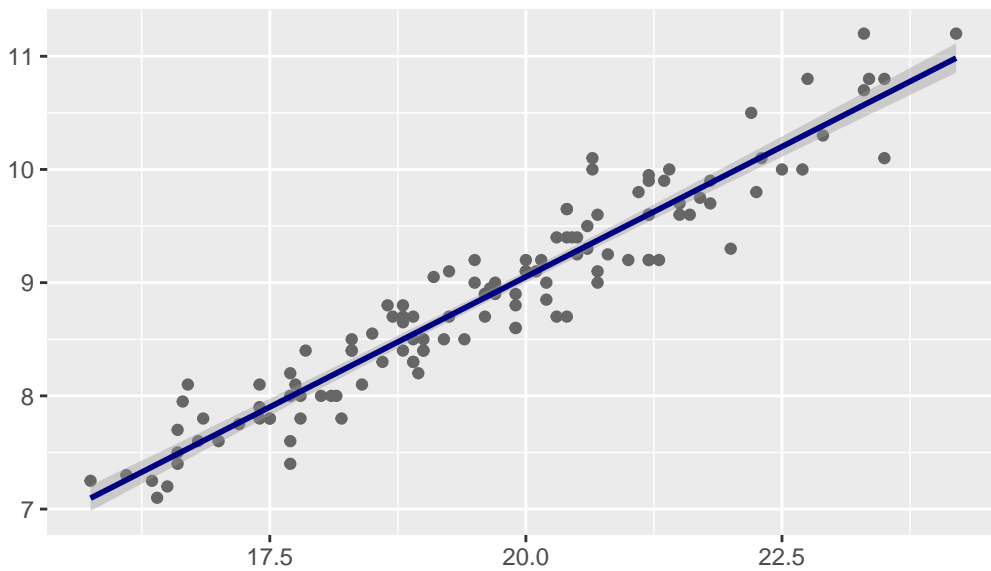
#APARTADO F

```
summary(modelo_hembra)$coefficients
```

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	-0.1499853	0.26186677	-0.5727541	5.678740e-01
EL	0.4600755	0.01327856	34.6479923	1.159924e-64

#TENDREMOS QUE DARLES FORMATO A ESTA Y TODAS LAS TABLAS (EN UNA DE LAS FOTOS DEL GRUPO ESTÁ CÓMO SE PUEDE HACER)

```
datos |>
  filter(SEX0 == "hembra") |>
  ggplot(
    aes(x = EL, y = KB),
  ) +
  geom_point(
    color = "grey40"
  ) +
  geom_smooth(
    method = "lm",
    color = "navy"
  ) +
  labs(
    x = "",
    y = "",
    title = "",
  )
```



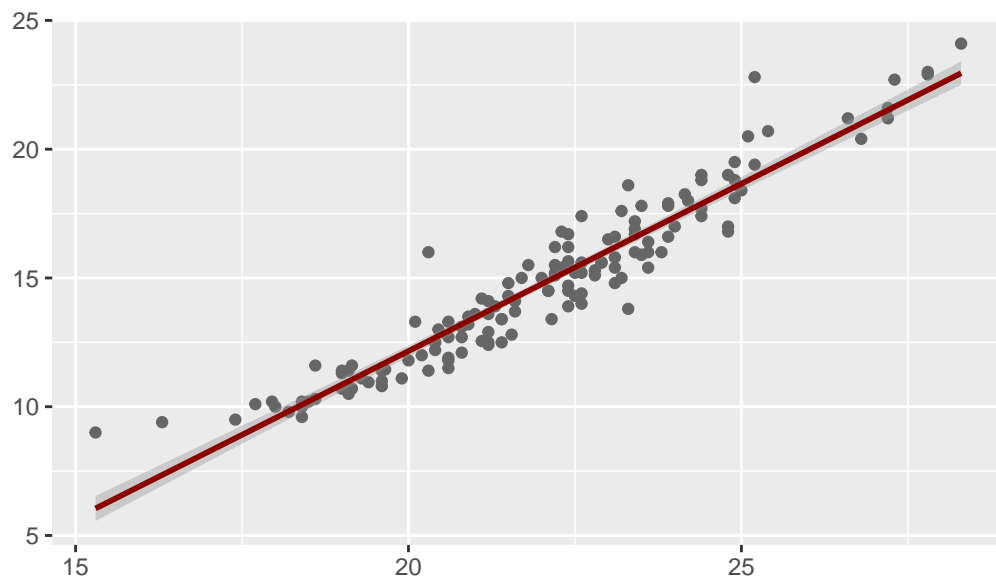
#Nos falta también modificar si queremos los colores de las gráficas, poner bien títulos, ejes...

```
summary(modelo_macho)$coefficients
```

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	-13.843700	0.76487085	-18.09940	1.748192e-38
EL	1.300155	0.03454023	37.64176	4.158379e-75

#Lo mismo para esta tabla

```
datos |>
  filter(SEX0 == "macho") |>
  ggplot(
    aes(x = EL, y = KB),
  ) +
  geom_point(
    color = "grey40"
  ) +
  geom_smooth(
    method = "lm",
    color = "darkred"
  ) +
  labs(
    x = "",
    y = "",
    title = "",
  )
```



## Conclusiones

## **Anexo:Código R**