

# Tarea A1

María Sánchez Paniagua

## Ejercicios iniciales

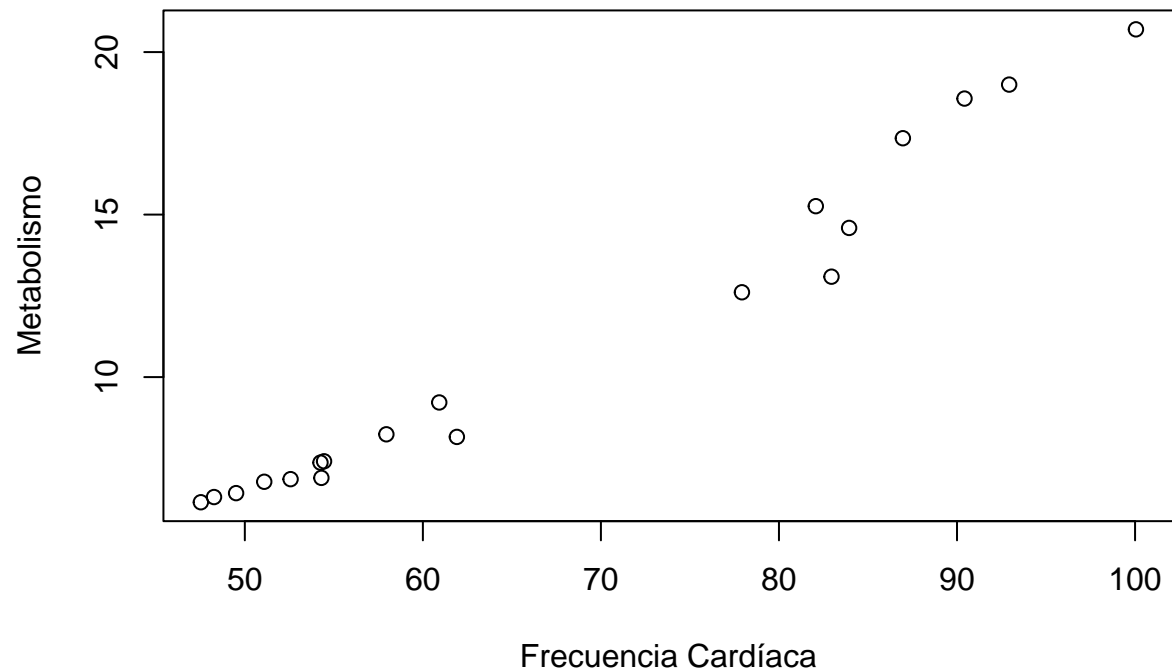
1. Con los datos de los buitres leonados:

```
heartbpm <- c(47.53, 48.27, 49.51, 51.09, 52.57, 54.30,  
54.25, 54.45, 57.95, 60.92, 61.91, 77.92,  
82.07, 82.95, 83.94, 86.96, 90.42, 92.93, 100.05)  
metabol <- c(6.15, 6.31, 6.43, 6.78, 6.86, 6.90, 7.37, 7.41,  
8.24, 9.22, 8.16, 12.61, 15.26, 13.09, 14.59,  
17.35, 18.57, 19.00, 20.70)  
vulture <- data.frame(heartbpm, metabol)  
rm(heartbpm, metabol)  
attach(vulture)
```

- (a) Dibujar la nube de puntos con la instrucción plot().

```
plot(vulture$heartbpm, vulture$metabol, xlab = "Frecuencia Cardíaca", ylab = "Metabolismo", main = "Relación entre Frecuencia Cardíaca y Metabolismo")
```

## Relación entre Frecuencia Cardíaca y Metabolismo

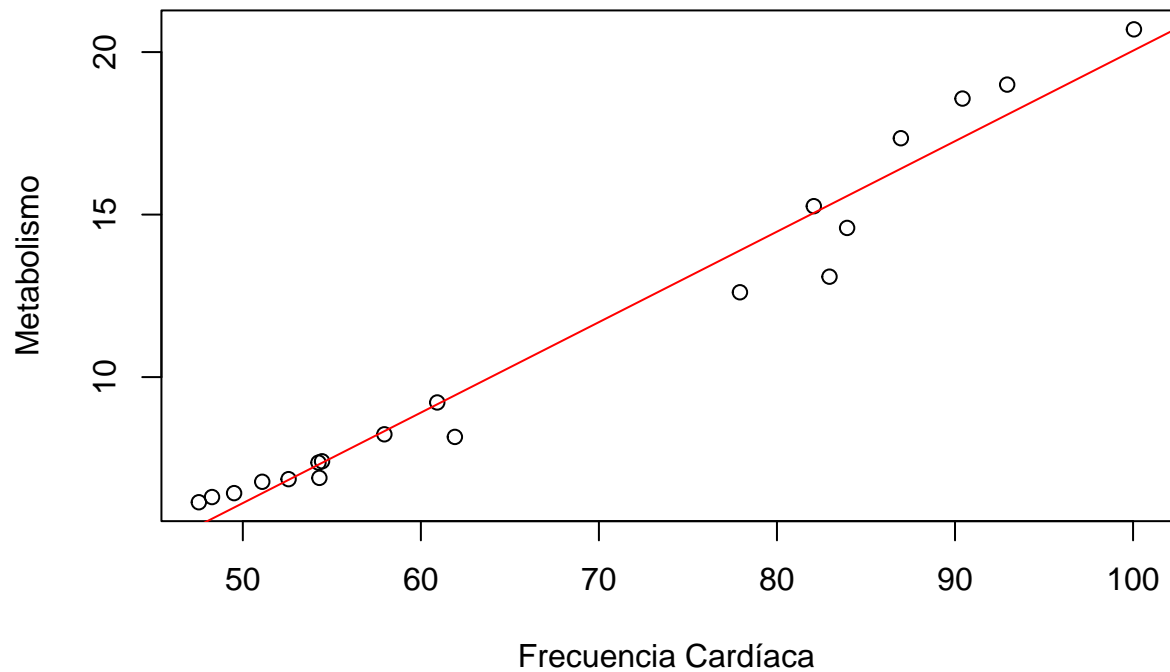


(b) Añadir la recta de regresión con la instrucción `abline()`.

```
plot.new()
plot(vulture$heartbpm, vulture$metabol, xlab = "Frecuencia Cardíaca", ylab = "Metabolismo", main = "Relación entre Frecuencia Cardíaca y Metabolismo")

model <- lm(metabol ~ heartbpm, data = vulture)
abline(model, col = "red")
```

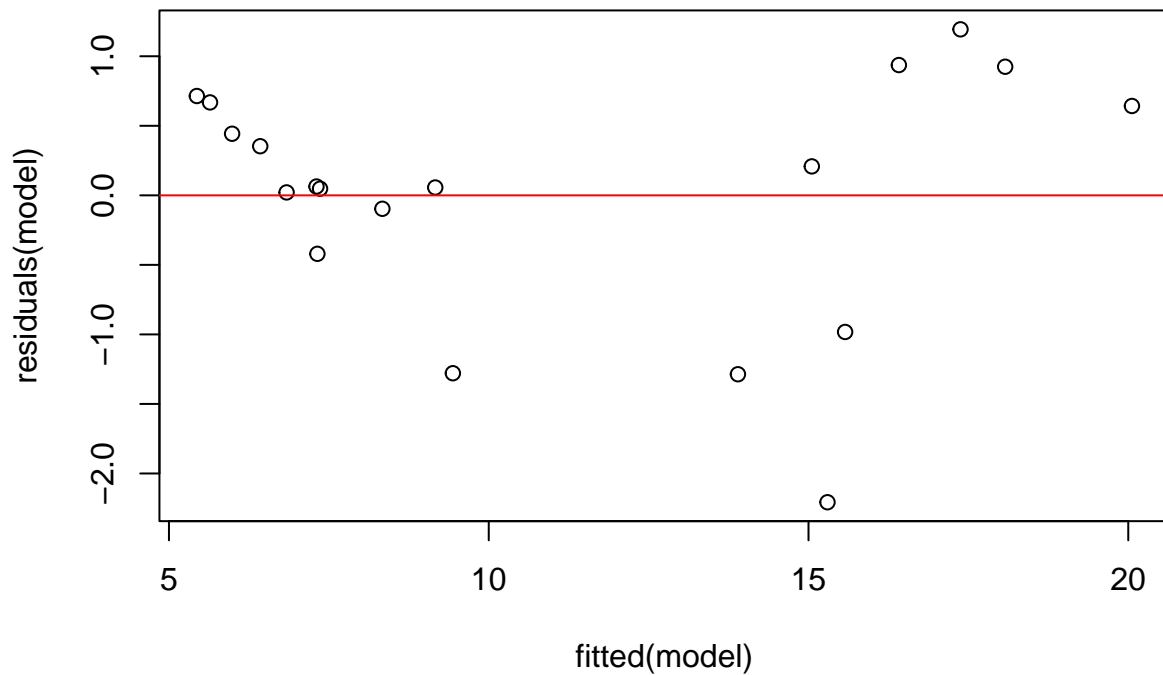
## Relación entre Frecuencia Cardíaca y Metabolismo



(c) Dibujar el gráfico de residuos vs. valores ajustados con la función `plot()` aplicada al objeto `lm`.

```
plot(residuals(model) ~ fitted(model), main = "Residuos vs. Valores Ajustados")  
abline(h = 0, col = "red")
```

## Residuos vs. Valores Ajustados



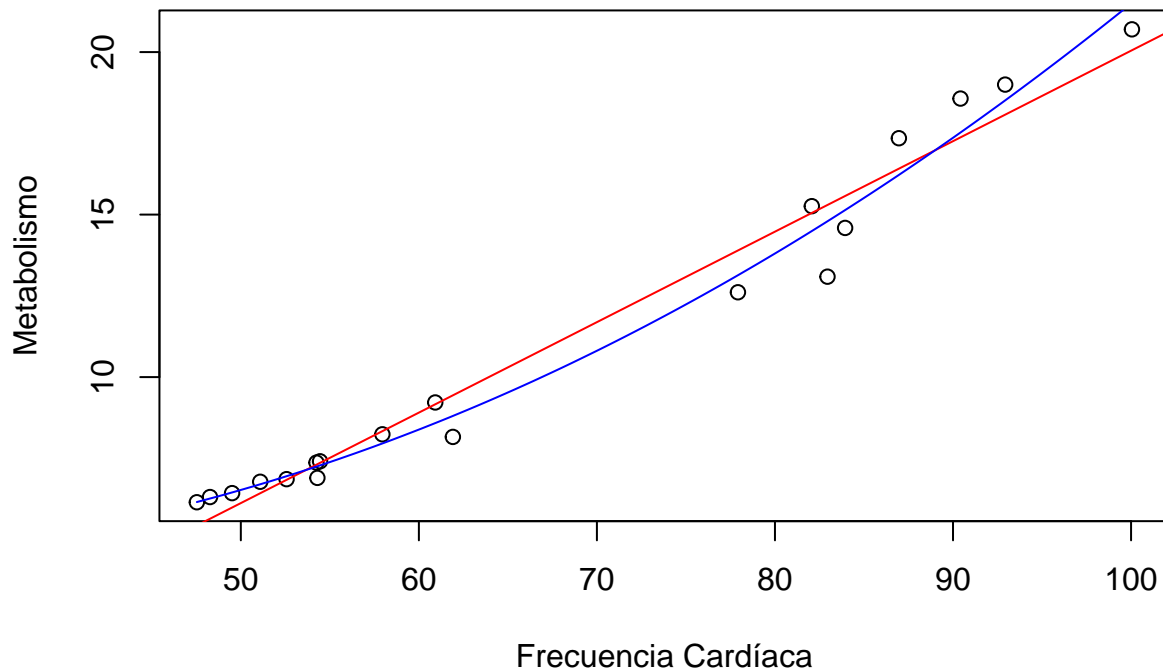
(d) Añadir al gráfico de dispersión del apartado (a) la curva de regresión parabólica. Nota: Esto último se puede hacer con la instrucción `lines()` o con la instrucción `curve()` , `add = TRUE`).

```
plot(vulture$heartbpm, vulture$metabol, xlab = "Frecuencia Cardíaca", ylab = "Metabolismo", main = "Relación entre Frecuencia Cardíaca y Metabolismo")

# Añadir la recta de regresión y la curva de regresión parabólica
model <- lm(metabol ~ heartbpm, data = vulture)
abline(model, col = "red")

model_quad <- lm(metabol ~ heartbpm + I(heartbpm^2), data = vulture)
curve(predict(model_quad, newdata = data.frame(heartbpm = x)), add = TRUE, col = "blue")
```

## Relación entre Frecuencia Cardíaca y Metabolismo



2. Con los datos de la tensión arterial sistólica y la edad de los 69 pacientes que podemos encontrar en la web de [www.fisterra.com](http://www.fisterra.com)

<https://www.fisterra.com/formacion/metodologia-investigacion/tecnicas-regresion-regresion-lineal-simple/> calcular los coeficientes de regresión de la recta mínimo cuadrática.

```
library(RCurl)
```

```
## Warning: package 'RCurl' was built under R version 4.3.2
```

```
library(XML)
```

```
## Warning: package 'XML' was built under R version 4.3.2
```

```
url <- "https://www.fisterra.com/formacion/metodologia-investigacion/tecnicas-regresion-regresion-lineal-simple/"
html_content <- getURL(url)
tables <- readHTMLTable(html_content)
datos <- tables[[2]]
```

```
# Habría que reestructurar el dataframe
```

```
#modelo <- lm(tension_sistolica ~ edad, data = datos)
#summary(modelo)
```

## Ejercicios del Libro de Faraway

1. (Ejercicio 1 cap. 1 pág. 12) The dataset teengamb concerns a study of teenage gambling in Britain. Make a numerical and graphical summary of the data, commenting on any features that you find interesting. Limit the output you present to a quantity that a busy reader would find sufficient to get a basic understanding of the data.

```
#chooseCRANmirror()
library(faraway)
```

```
## Warning: package 'faraway' was built under R version 4.3.3
```

```
data(teengamb, package = "faraway")

# Resumen numérico
summary(teengamb)
```

```
##      sex      status      income      verbal
##  Min.   :0.0000   Min.   :18.00   Min.   : 0.600   Min.   : 1.00
## 1st Qu.:0.0000   1st Qu.:28.00   1st Qu.: 2.000   1st Qu.: 6.00
## Median :0.0000   Median :43.00   Median : 3.250   Median : 7.00
## Mean   :0.4043   Mean   :45.23   Mean   : 4.642   Mean   : 6.66
## 3rd Qu.:1.0000   3rd Qu.:61.50   3rd Qu.: 6.210   3rd Qu.: 8.00
## Max.   :1.0000   Max.   :75.00   Max.   :15.000   Max.   :10.00
##      gamble
##  Min.   : 0.0
## 1st Qu.: 1.1
## Median : 6.0
## Mean   :19.3
## 3rd Qu.:19.4
## Max.   :156.0
```

```
# Medias de las variables numéricas
mean(teengamb$gamble)
```

```
## [1] 19.30106
```

```
mean(teengamb$income)
```

```
## [1] 4.641915
```

```
mean(teengamb$verbal)
```

```
## [1] 6.659574
```

```
# Desviaciones estándar
sd(teengamb$gamble)
```

```
## [1] 31.51587
```

```
sd(teengamb$income)
```

```
## [1] 3.551371
```

```
sd(teengamb$verbal)
```

```
## [1] 1.856558
```

```
# Gráficos
```

```
# Histograma de la cantidad apostada
```

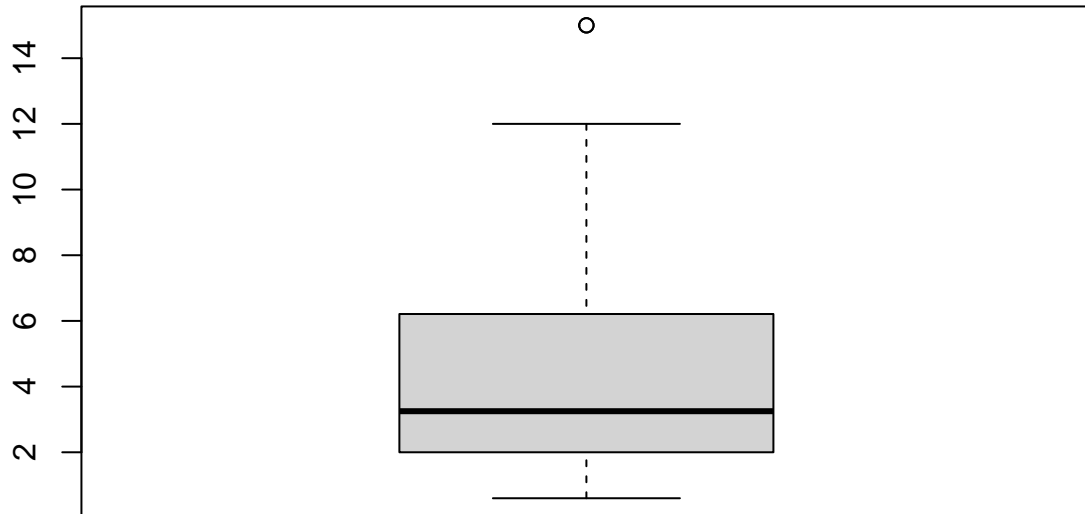
```
hist(teengamb$gamble, main="Histograma")
```



```
# Diagrama de caja de ingresos
```

```
boxplot(teengamb$income, main="Diagrama de Cajas")
```

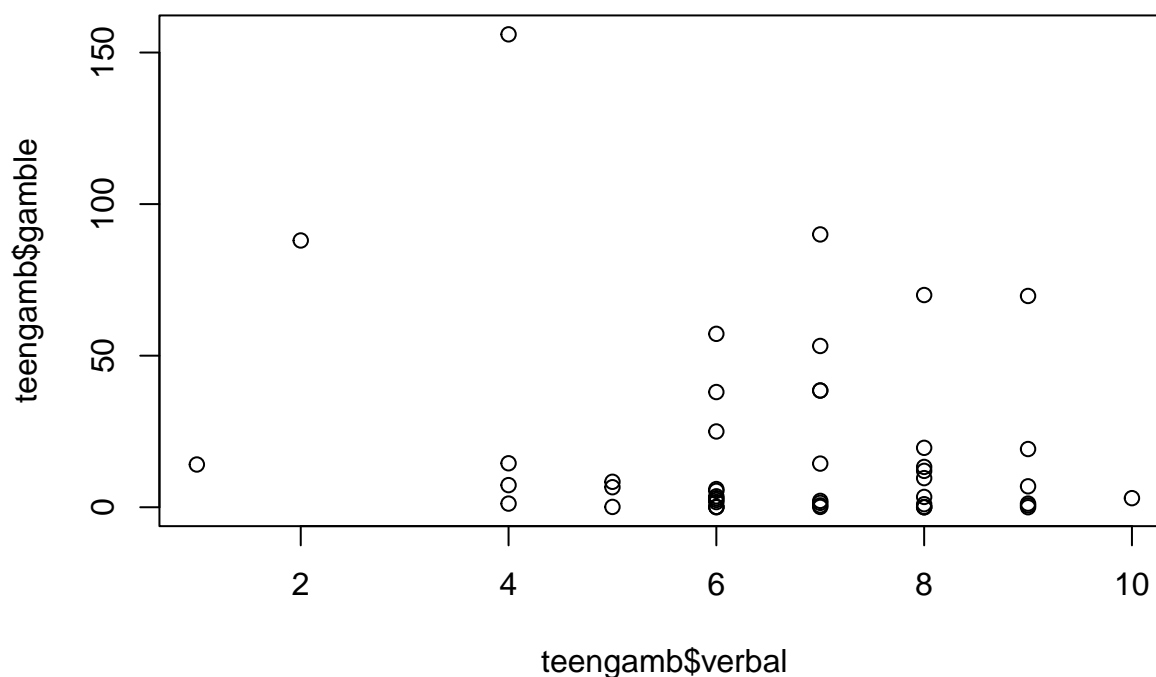
## Diagrama de Cajas



```
# Gráfico de dispersión entre cantidad apostada y verbal  
plot(teengamb$verbal, teengamb$gamble, main="Gráfico de dispersión")
```



## Gráfico de dispersión



2. (Ejercicio 2 pág. 12) The dataset `uswages` is drawn as a sample from the Current Population Survey in 1988. Make a numerical and graphical summary of the data as in the previous question.

```
# Carga del dataset uswages
data(uswages, package = "faraway")

# Resumen numérico
summary(uswages)
```

```
##      wage      educ      exper      race
## Min.   : 50.39  Min.   : 0.00  Min.   : -2.00  Min.   :0.000
## 1st Qu.: 308.64 1st Qu.:12.00  1st Qu.:  8.00  1st Qu.:0.000
## Median : 522.32 Median :12.00  Median :15.00  Median :0.000
## Mean   : 608.12 Mean   :13.11  Mean   :18.41  Mean   :0.078
## 3rd Qu.: 783.48 3rd Qu.:16.00  3rd Qu.:27.00  3rd Qu.:0.000
## Max.   :7716.05 Max.   :18.00  Max.   :59.00  Max.   :1.000
##      smsa      ne      mw      so
## Min.   :0.000  Min.   :0.000  Min.   :0.0000  Min.   :0.0000
## 1st Qu.:1.000  1st Qu.:0.000  1st Qu.:0.0000  1st Qu.:0.0000
## Median :1.000  Median :0.000  Median :0.0000  Median :0.0000
## Mean   :0.756  Mean   :0.229  Mean   :0.2485  Mean   :0.3125
## 3rd Qu.:1.000  3rd Qu.:0.000  3rd Qu.:0.0000  3rd Qu.:1.0000
## Max.   :1.000  Max.   :1.000  Max.   :1.0000  Max.   :1.0000
##      we      pt
## Min.   :0.00  Min.   :0.0000
```

```
## 1st Qu.:0.00 1st Qu.:0.0000
## Median :0.00 Median :0.0000
## Mean :0.21 Mean :0.0925
## 3rd Qu.:0.00 3rd Qu.:0.0000
## Max. :1.00 Max. :1.0000
```

```
# Medias de las variables numéricas relevantes
mean(uswages$wage)
```

```
## [1] 608.1179
```

```
mean(uswages$education)
```

```
## Warning in mean.default(uswages$education): argument is not numeric or logical:
## returning NA
```

```
## [1] NA
```

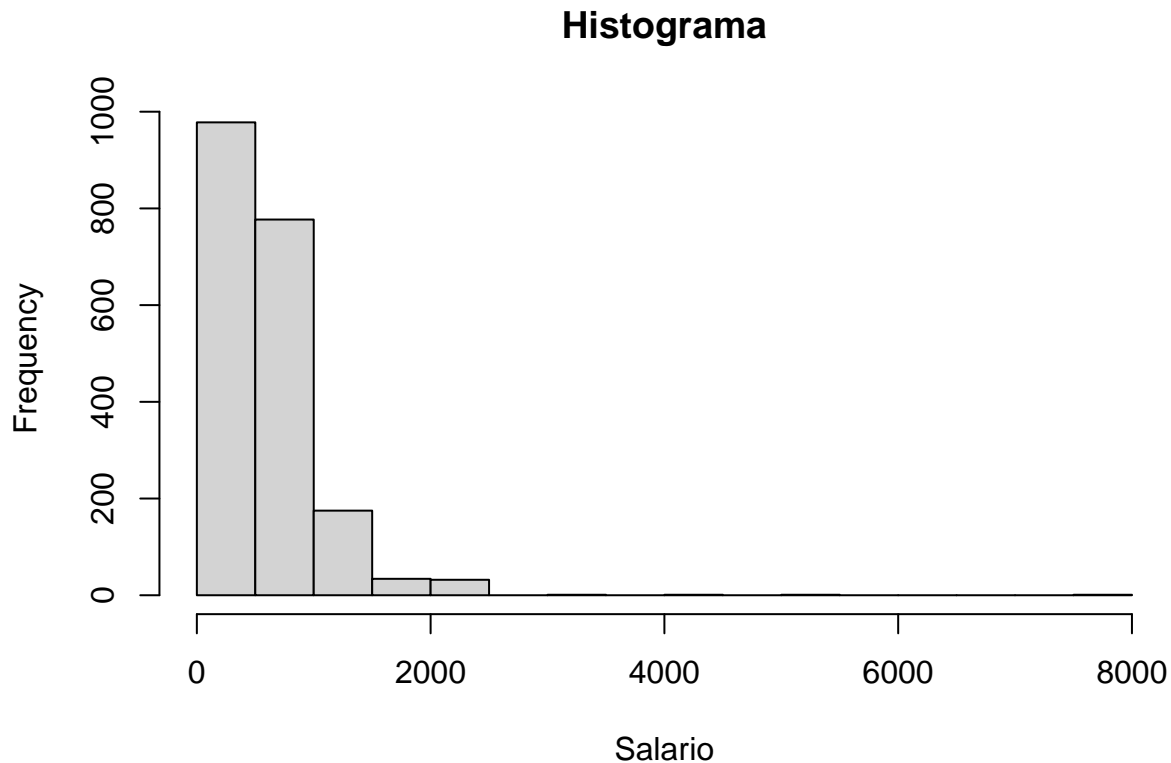
```
# Desviaciones estándar
sd(uswages$wage)
```

```
## [1] 459.8326
```

```
sd(uswages$education)
```

```
## [1] NA
```

```
# Gráficos
# Histograma de salarios
hist(uswages$wage, main="Histograma", xlab="Salario")
```



## Ejercicios del libro de Carmona

### 3. (Ejercicio 1.3 del Capítulo 1 página 24)

Consideremos el problema de tráfico planteado en el apartado 1.2 de este capítulo, con la variable independiente densidad y la variable dependiente raíz cuadrada de la velocidad. Con los datos proporcionados en la tabla 1.1

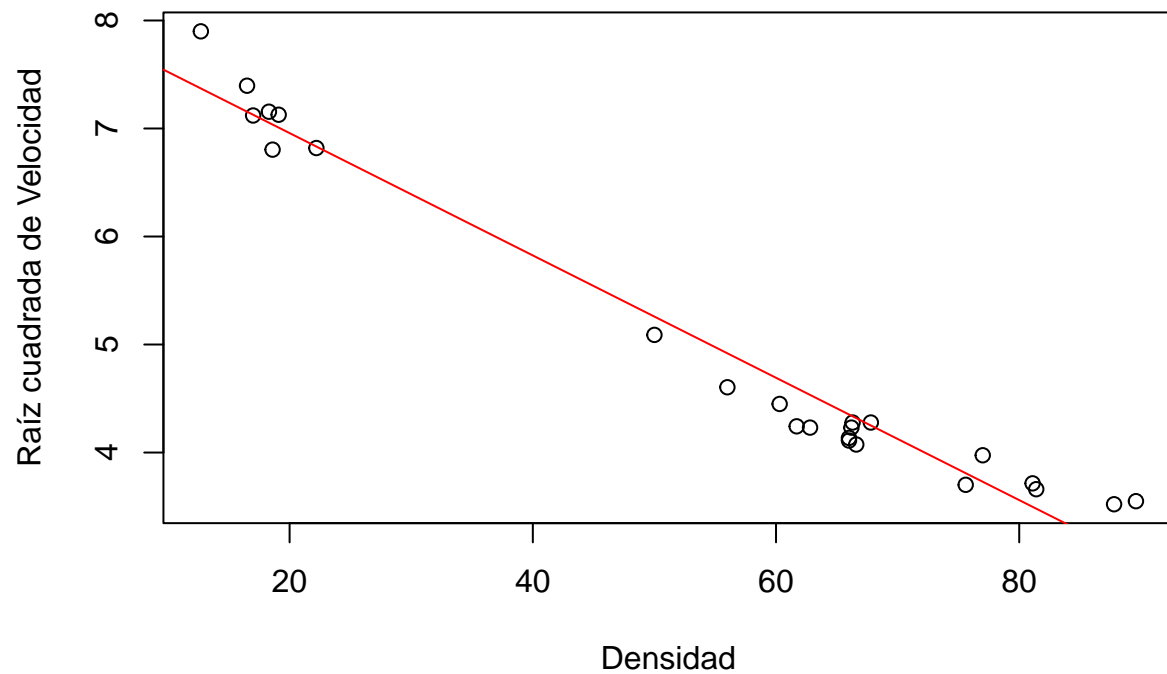
```
dens <- c(12.7,17.0,66.0,50.0,87.8,81.4,75.6,66.2,81.1,62.8,77.0,89.6,
18.3,19.1,16.5,22.2,18.6,66.0,60.3,56.0,66.3,61.7,66.6,67.8)
vel <- c(62.4,50.7,17.1,25.9,12.4,13.4,13.7,17.9,13.8,17.9,15.8,12.6,
51.2,50.8,54.7,46.5,46.3,16.9,19.8,21.2,18.3,18.0,16.6,18.3)
rvel <- sqrt(vel)
```

realizar el siguiente proceso:

- Dibujar la nube de puntos y la recta que pasa por los puntos (12.7,62.4) y (87.8,12.4). Dibujar el gráfico de los residuos con la densidad y el gráfico con las predicciones. Calcular la suma de cuadrados de los residuos.

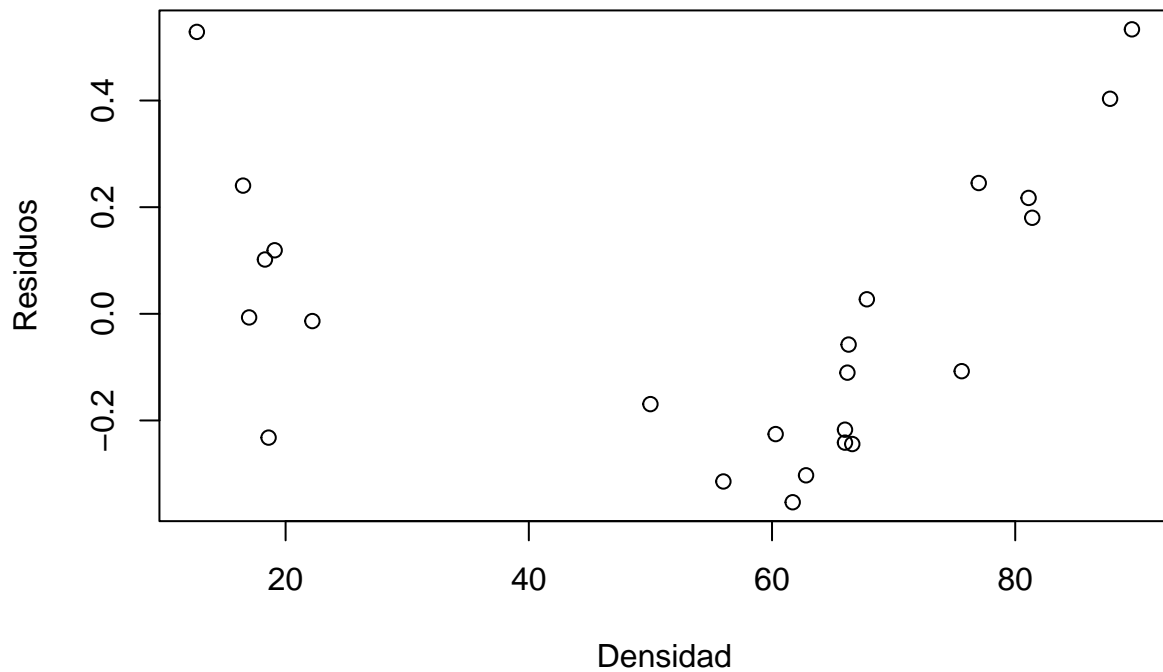
```
plot(dens, rvel, main = "Nube de Puntos y Recta Específica", xlab = "Densidad", ylab = "Raíz cuadrada de la velocidad")
abline(lm(rvel ~ dens), col = "red") # Ajusta y dibuja una recta de regresión lineal
```

## Nube de Puntos y Recta Específica



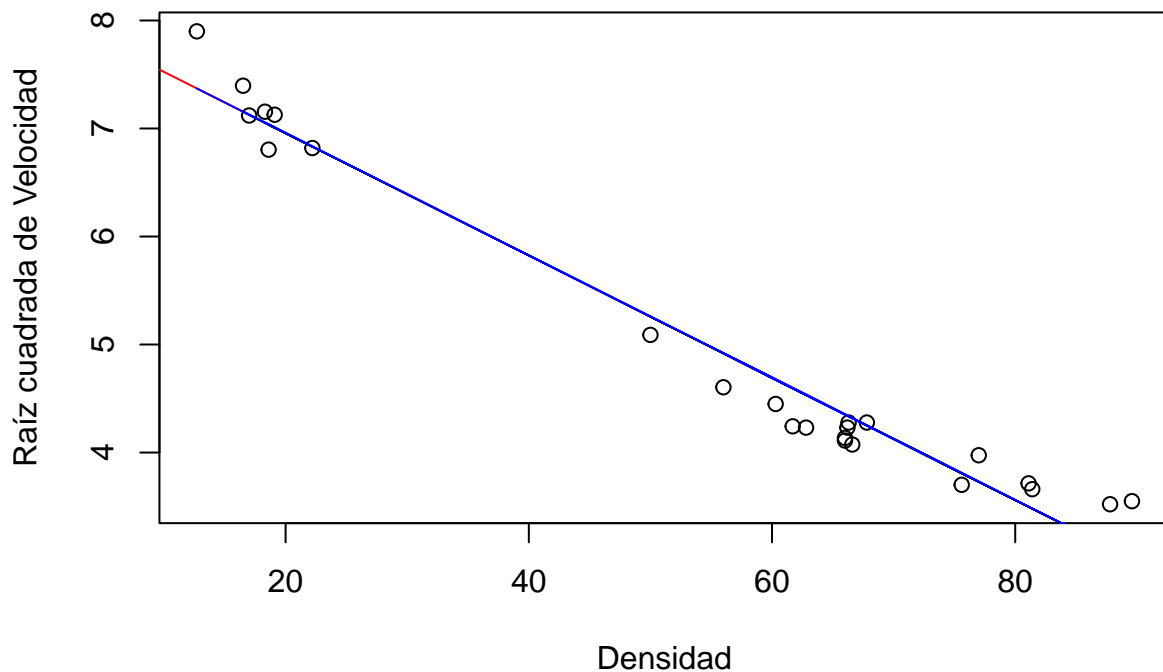
```
# Dibujar el gráfico de los residuos
model <- lm(rvel ~ dens)
residuos <- residuals(model)
plot(dens, residuos, main = "Gráfico de Residuos con Densidad", xlab = "Densidad", ylab = "Residuos")
```

## Gráfico de Residuos con Densidad



```
# Dibujar el gráfico con las predicciones
newdata <- data.frame(dens = dens)
predicciones <- predict(model, newdata)
plot(dens, rvel, main = "Nube de Puntos y Recta Específica", xlab = "Densidad", ylab = "Raíz cuadrada de")
abline(lm(rvel ~ dens), col = "red") # Recta de regresión
lines(dens, predicciones, col = "blue") # Línea de predicciones
```

## Nube de Puntos y Recta Específica



```
# Calcular la suma de cuadrados de los residuos
suma_cuadrados_residuos <- sum(residuos^2)
cat("Suma de cuadrados de los residuos:", suma_cuadrados_residuos, "\n")
```

```
## Suma de cuadrados de los residuos: 1.591218
```

- (b) Hallar la recta de regresión simple. Dibujar el gráfico de los residuos con la densidad y el gráfico con las predicciones. Calcular la suma de cuadrados de los residuos.

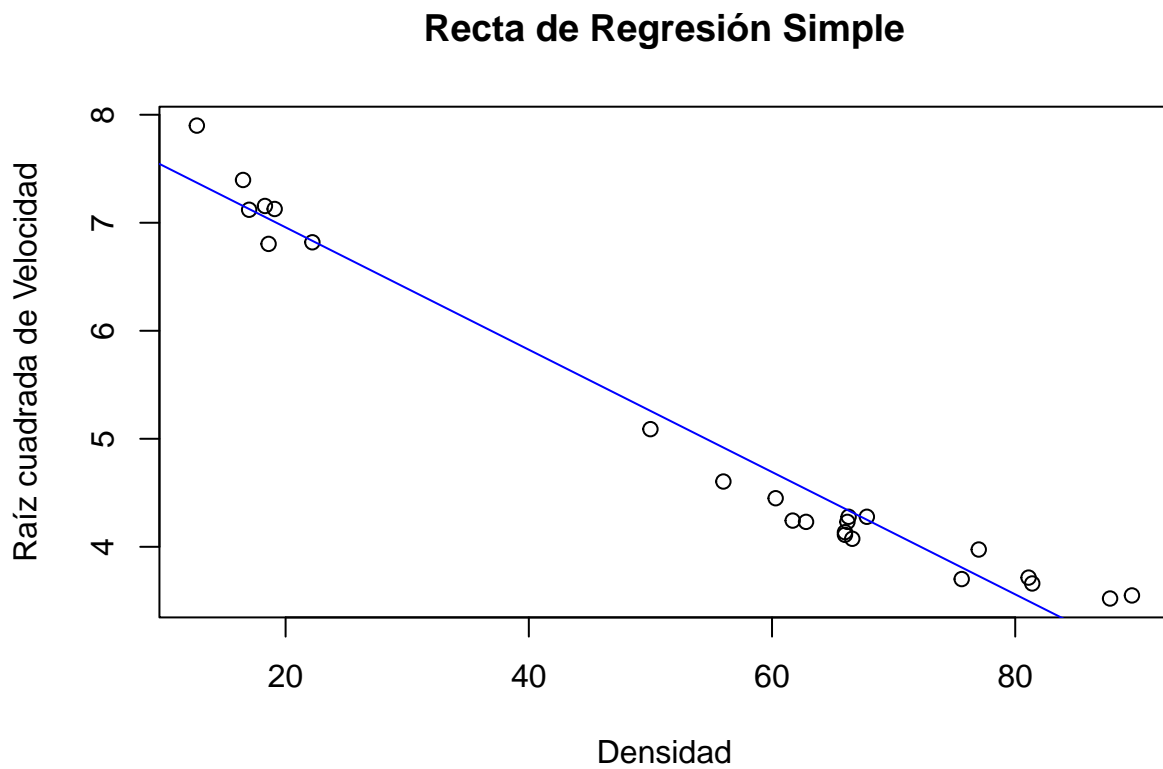
La recta de regresión simple:

```
model_simple <- lm(rvel ~ dens)
summary(model_simple)
```

```
##
## Call:
## lm(formula = rvel ~ dens)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.35337 -0.22722 -0.03566  0.18942  0.53349
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)  8.089813   0.130629   61.93   <2e-16 ***
## dens        -0.056626   0.002177  -26.01   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2689 on 22 degrees of freedom
## Multiple R-squared:  0.9685, Adjusted R-squared:  0.9671
## F-statistic: 676.4 on 1 and 22 DF,  p-value: < 2.2e-16
```

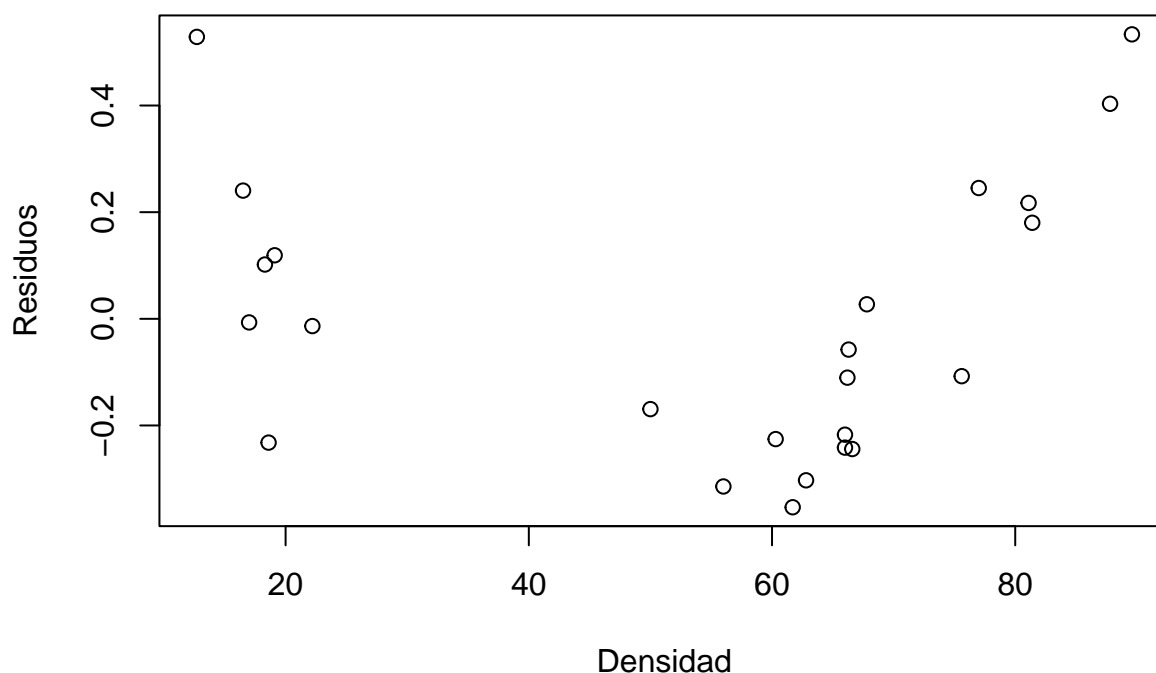
```
plot(dens, rvel, main = "Recta de Regresión Simple", xlab = "Densidad", ylab = "Raíz cuadrada de Velocidad")
abline(model_simple, col = "blue")
```



Para los residuos:

```
residuos_simple <- resid(model_simple)
plot(dens, residuos_simple, main = "Residuos de la Regresión Simple", xlab = "Densidad", ylab = "Residuos")
```

## Residuos de la Regresión Simple



La suma de cuadrados de los residuos:

```
sum(residuos_simple^2)
```

```
## [1] 1.591218
```

(c) Mejorar el modelo anterior considerando una regresión parabólica. Dibujar el gráfico de los residuos con la densidad y el gráfico con las predicciones. Calcular la suma de cuadrados de los residuos.

```
mparabolic <- lm(rvel ~ dens + I(dens^2))
summary(mparabolic)
```

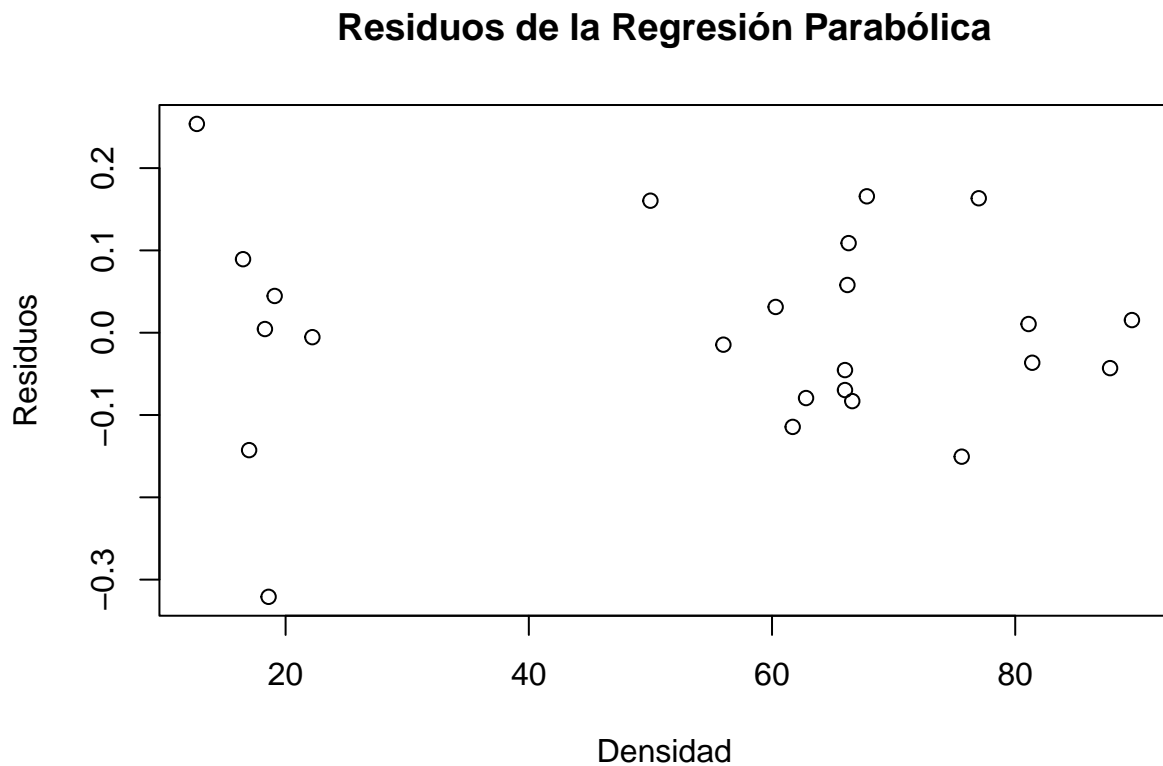
```
##
## Call:
## lm(formula = rvel ~ dens + I(dens^2))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.32089 -0.07209 -0.00047  0.06582  0.25368
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  8.881e+00  1.118e-01  79.469  < 2e-16 ***
## dens        -1.035e-01  5.567e-03 -18.593 1.60e-14 ***
## I(dens^2)     4.893e-04  5.705e-05   8.576 2.65e-08 ***
```



```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1297 on 21 degrees of freedom
## Multiple R-squared:  0.993, Adjusted R-squared:  0.9923
## F-statistic: 1490 on 2 and 21 DF,  p-value: < 2.2e-16
```

Para los residuos:

```
residuosp <- resid(mparabolic)
plot(dens, residuosp, main = "Residuos de la Regresión Parabólica", xlab = "Densidad", ylab = "Residuos")
```



La suma de cuadrados de los residuos:

```
sum(residuosp^2)
```

```
## [1] 0.3534143
```

(d) Calcular la capacidad de la carretera o punto de máximo flujo. Recordar que  $\text{flujo} = \text{vel} \times \text{densidad}$ .

```
flujo <- vel * dens
max_flujo <- max(flujo)
max_flujo_dens <- dens[which.max(flujo)]
max_flujo
```

```
## [1] 1295
```

```
max_flujo_dens
```

```
## [1] 50
```

#### 4. (Ejercicio 1.4 del Capítulo 1 página 24)

La siguiente tabla contiene los mejores tiempos conseguidos en algunas pruebas de velocidad en atletismo en los Juegos Olímpicos de Atlanta:

```
distancia <- c(100, 200, 400, 800, 1500, 5000, 10000, 42195)
tiempo_hombres <- c(9.84, 19.32, 43.19, 102.58, 215.78, 787.96, 1627.34, 7956.00)
tiempo_mujeres <- c(10.94, 22.12, 48.25, 117.73, 240.83, 899.88, 1861.63, 8765.00)
```

Si tomamos como variable regresora o independiente la distancia (metros) y como variable respuesta o dependiente el tiempo (segundos):

- (a) Calcular la recta de regresión simple con los datos de los hombres y dibujarla. Dibujar el gráfico de los residuos con la distancia y el gráfico con las predicciones. Calcular la suma de cuadrados de los residuos y el  $R^2$

```
model_hombres <- lm(tiempo_hombres ~ distancia)
summary(model_hombres)
```

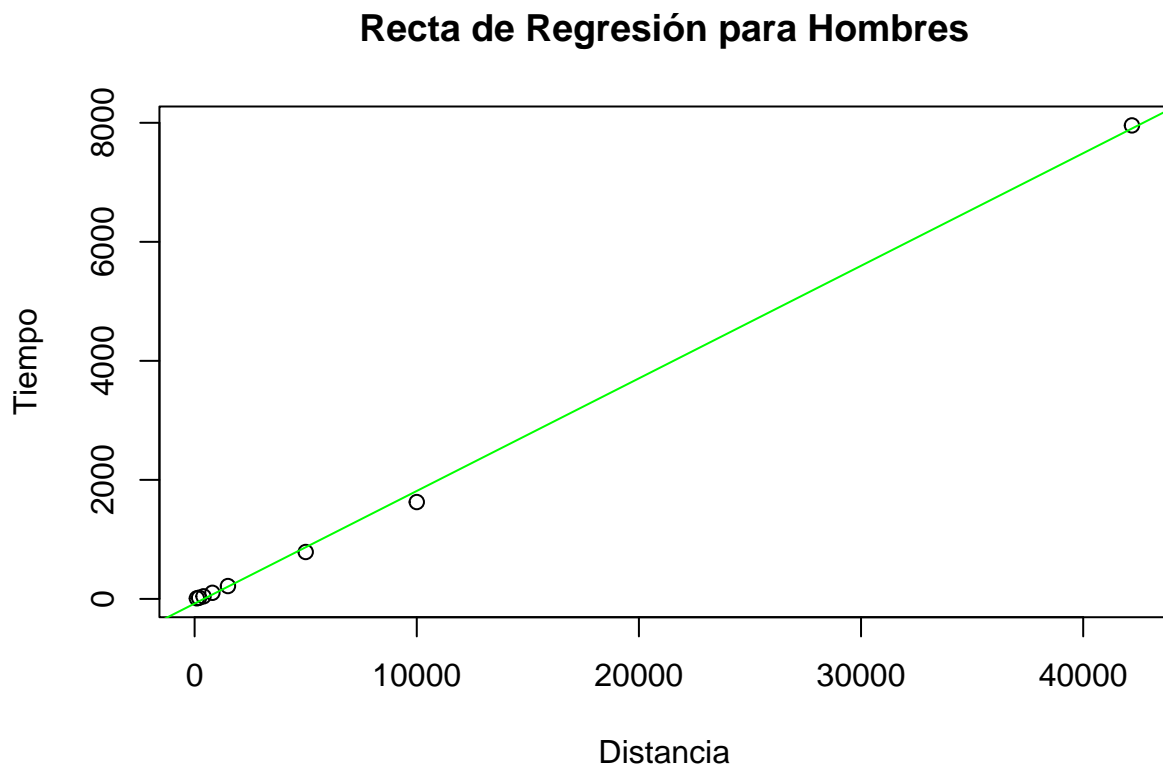
```
##
## Call:
## lm(formula = tiempo_hombres ~ distancia)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -186.25  -12.27   37.58   53.76   69.13
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -78.204273  38.806368  -2.015   0.0905 .
## distancia    0.189179   0.002512  75.296 3.69e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 95.86 on 6 degrees of freedom
## Multiple R-squared:  0.9989, Adjusted R-squared:  0.9988
## F-statistic: 5670 on 1 and 6 DF, p-value: 3.694e-10
```

```
distancia <- c(100, 200, 400, 800, 1500, 5000, 10000, 42195)
tiempo_hombres <- c(9.84, 19.32, 43.19, 102.58, 215.78, 787.96, 1627.34, 7956.00)

model_hombres <- lm(tiempo_hombres ~ distancia)
summary(model_hombres)
```

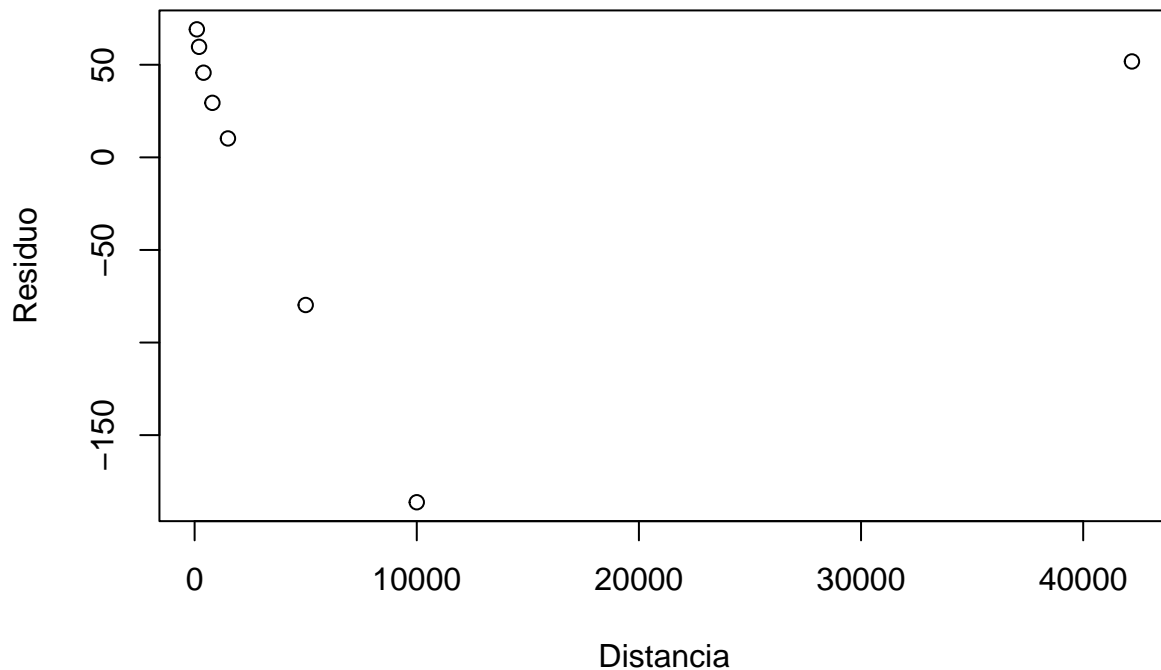
```
##
## Call:
## lm(formula = tiempo_hombres ~ distancia)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -186.25  -12.27   37.58   53.76   69.13
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -78.204273  38.806368  -2.015  0.0905 .
## distancia    0.189179   0.002512  75.296 3.69e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 95.86 on 6 degrees of freedom
## Multiple R-squared:  0.9989, Adjusted R-squared:  0.9988
## F-statistic: 5670 on 1 and 6 DF, p-value: 3.694e-10
```

```
plot(distancia, tiempo_hombres, main = "Recta de Regresión para Hombres", xlab = "Distancia", ylab = "Tiempo", col = "green")
abline(model_hombres, col = "green")
```



```
residuos_hombres <- resid(model_hombres)
plot(distancia, residuos_hombres, main = "Residuos de la Regresión para Hombres", xlab = "Distancia", ylab = "Residuos", col = "green")
```

## Residuos de la Regresión para Hombres



```
suma_cuadrados_residuos_hombres <- sum(residuos_hombres^2)
```

```
# Continuación de tu código para hombres
```

```
# Cálculo de la suma de cuadrados de los residuos
```

```
suma_cuadrados_residuos_hombres <- sum(residuos_hombres^2)
```

```
print(suma_cuadrados_residuos_hombres)
```

```
## [1] 55130.19
```

```
# Cálculo del R^2
```

```
r2_hombres <- summary(model_hombres)$r.squared
```

```
print(r2_hombres)
```

```
## [1] 0.9989428
```

(b) Repetir el apartado anterior utilizando los logaritmos de las variables tiempo y distancia.

```
model_hombres_log <- lm(log(tiempo_hombres) ~ log(distancia))
```

```
summary(model_hombres_log)
```

```
##
```

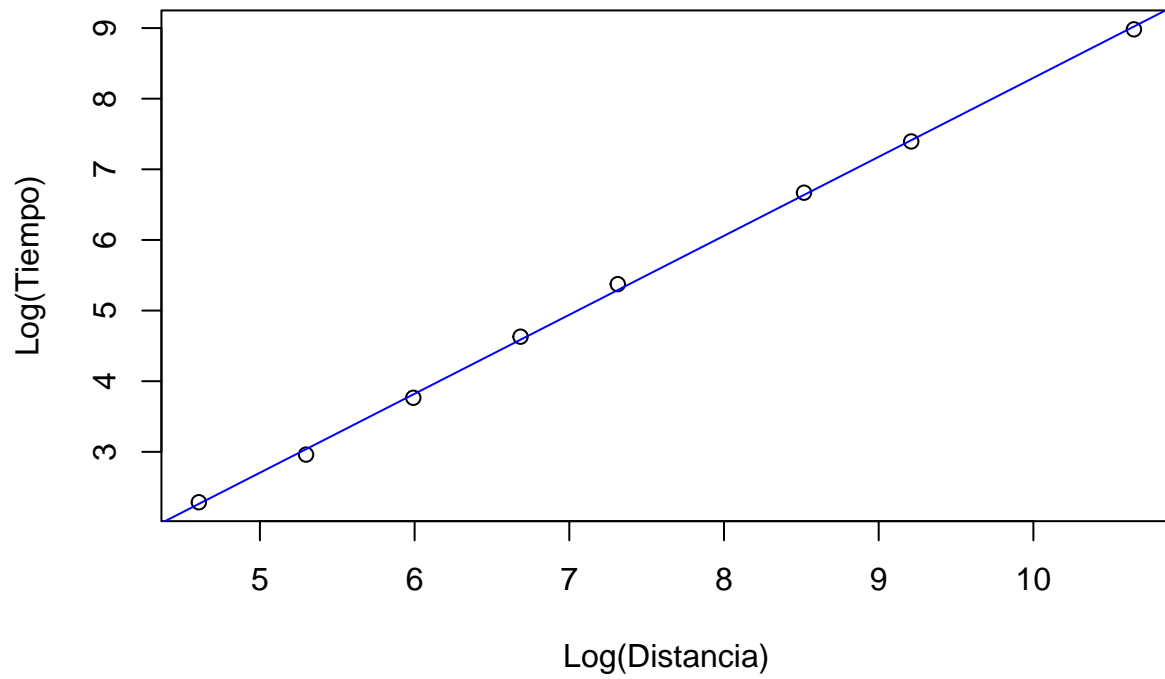
```
## Call:
## lm(formula = log(tiempo_hombres) ~ log(distancia))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.076918 -0.041917  0.003033  0.034958  0.083369
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -2.88591    0.08067  -35.77 3.18e-08 ***
## log(distancia)  1.11808    0.01071  104.43 5.20e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.05836 on 6 degrees of freedom
## Multiple R-squared:  0.9995, Adjusted R-squared:  0.9994
## F-statistic: 1.091e+04 on 1 and 6 DF,  p-value: 5.197e-11
```

```
# Regresión con logaritmos para hombres
summary(model_hombres_log)
```

```
##
## Call:
## lm(formula = log(tiempo_hombres) ~ log(distancia))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.076918 -0.041917  0.003033  0.034958  0.083369
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -2.88591    0.08067  -35.77 3.18e-08 ***
## log(distancia)  1.11808    0.01071  104.43 5.20e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.05836 on 6 degrees of freedom
## Multiple R-squared:  0.9995, Adjusted R-squared:  0.9994
## F-statistic: 1.091e+04 on 1 and 6 DF,  p-value: 5.197e-11
```

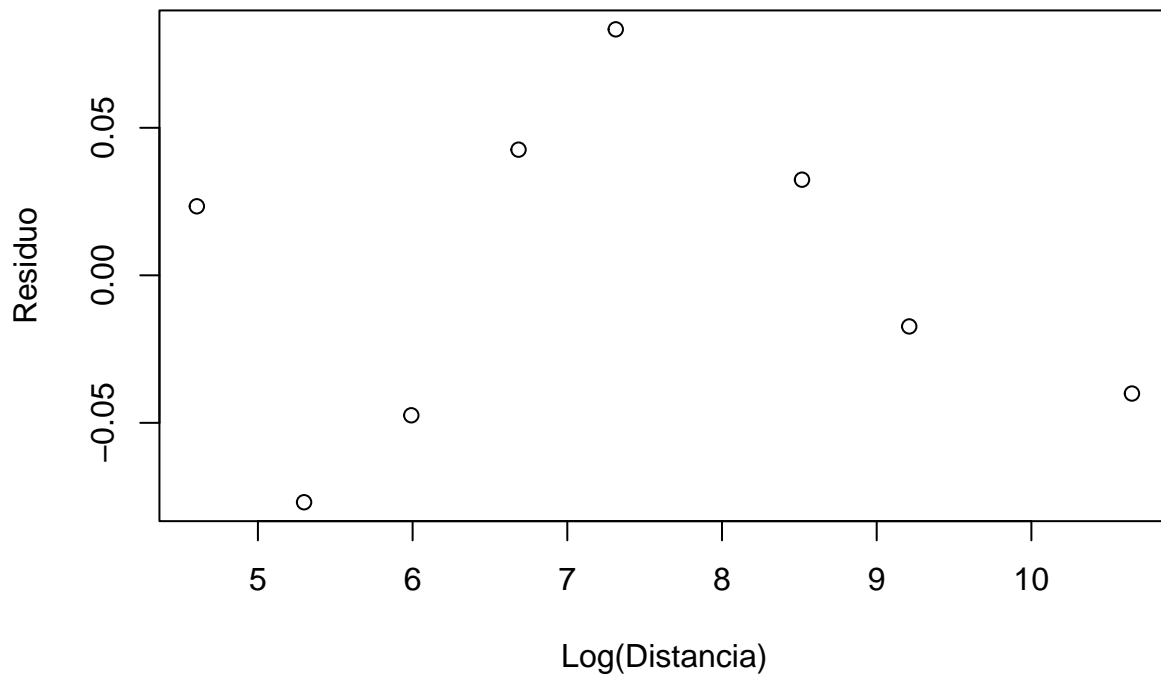
```
plot(log(distancia), log(tiempo_hombres), main = "Recta de Regresión Logarítmica para Hombres", xlab =
abline(model_hombres_log, col = "blue")
```

## Recta de Regresión Logarítmica para Hombres



```
# Residuos de la regresión logarítmica  
residuos_hombres_log <- resid(model_hombres_log)  
plot(log(distancia), residuos_hombres_log, main = "Residuos de la Regresión Logarítmica para Hombres",
```

## Residuos de la Regresión Logarítmica para Hombres



```
# Suma de cuadrados de los residuos para el modelo logarítmico
suma_cuadrados_residuos_hombres_log <- sum(residuos_hombres_log^2)
print(suma_cuadrados_residuos_hombres_log)
```

```
## [1] 0.02043599
```

```
# R^2 para el modelo logarítmico
r2_hombres_log <- summary(model_hombres_log)$r.squared
print(r2_hombres_log)
```

```
## [1] 0.9994501
```

(c) Repetir los dos apartados anteriores utilizando los datos de las mujeres.

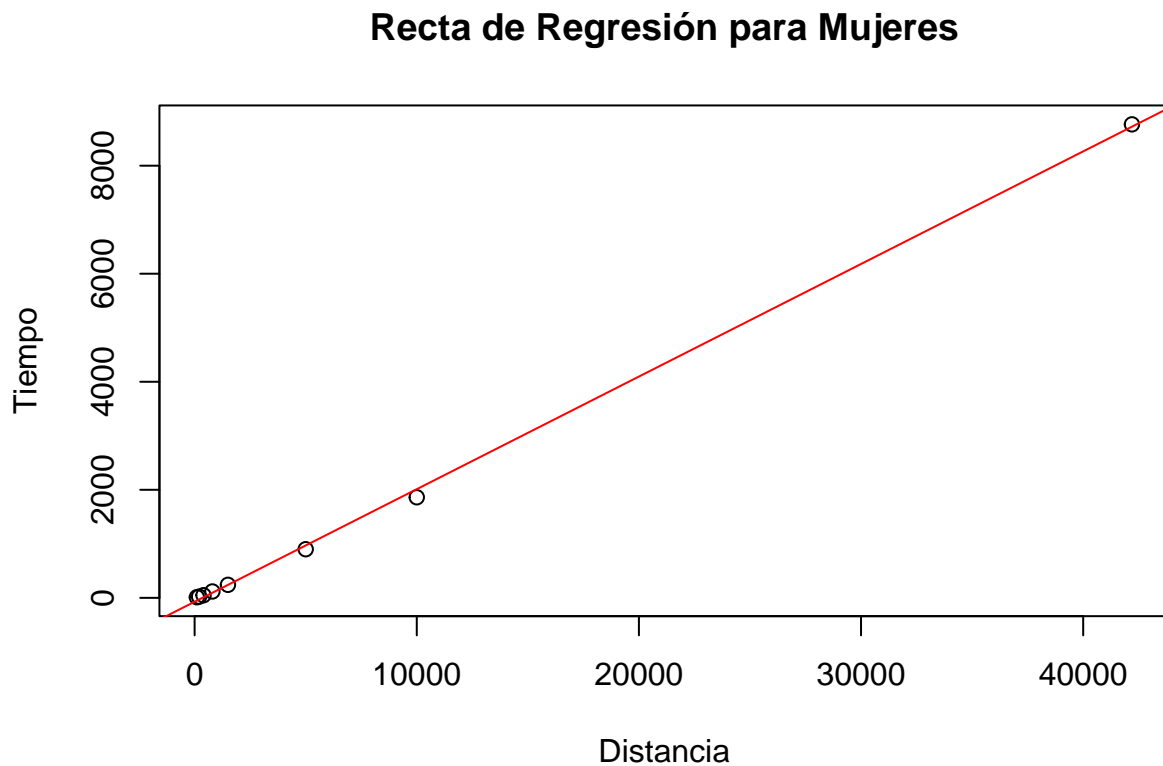
```
# Modelo lineal para mujeres
model_mujeres <- lm(tiempo_mujeres ~ distancia)
summary(model_mujeres)
```

```
##
## Call:
## lm(formula = tiempo_mujeres ~ distancia)
##
## Residuals:
```

```
##      Min      1Q  Median      3Q      Max
## -150.17 -16.89   30.48   45.22   62.64
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -72.545862  32.184144  -2.254   0.0651 .
## distancia    0.208435   0.002084 100.030 6.73e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 79.5 on 6 degrees of freedom
## Multiple R-squared:  0.9994, Adjusted R-squared:  0.9993
## F-statistic: 1.001e+04 on 1 and 6 DF, p-value: 6.727e-11
```

```
# Gráfico de la recta de regresión para mujeres
```

```
plot(distancia, tiempo_mujeres, main = "Recta de Regresión para Mujeres", xlab = "Distancia", ylab = "Tiempo")
abline(model_mujeres, col = "red")
```

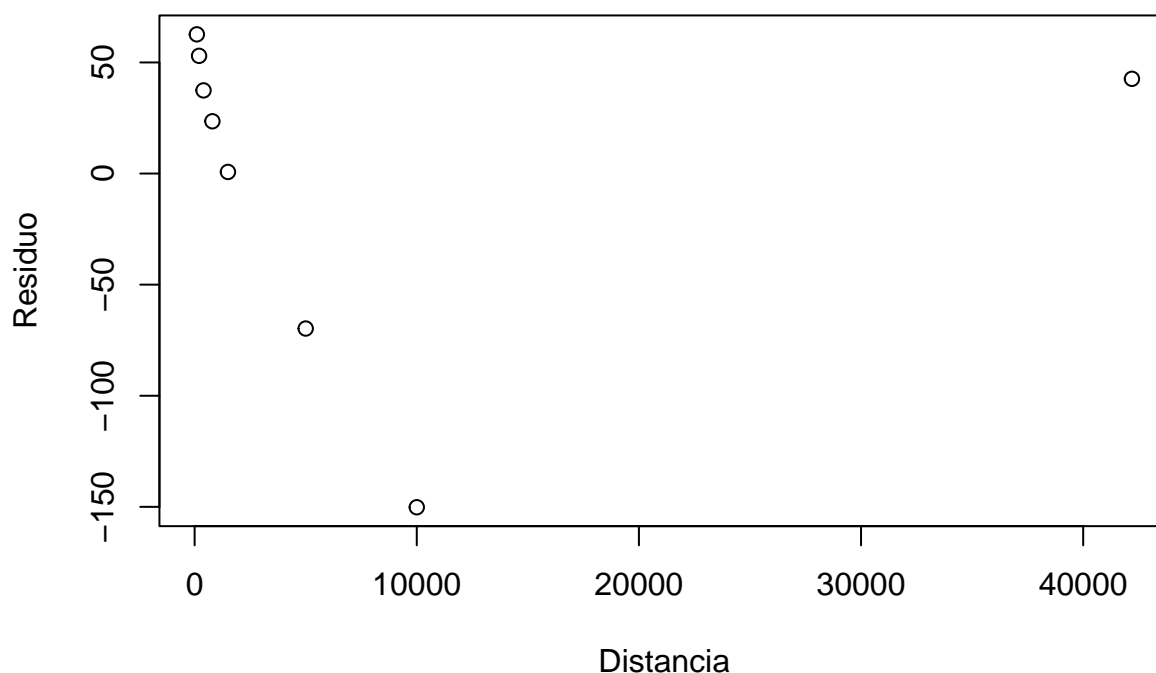


```
# Residuos para mujeres
```

```
residuos_mujeres <- resid(model_mujeres)
plot(distancia, residuos_mujeres, main = "Residuos de la Regresión para Mujeres", xlab = "Distancia", ylab = "Residuos")
```



## Residuos de la Regresión para Mujeres



```
# Suma de cuadrados de los residuos y R^2 para mujeres
suma_cuadrados_residuos_mujeres <- sum(residuos_mujeres^2)
print(suma_cuadrados_residuos_mujeres)
```

```
## [1] 37919.92
```

```
r2_mujeres <- summary(model_mujeres)$r.squared
print(r2_mujeres)
```

```
## [1] 0.9994007
```

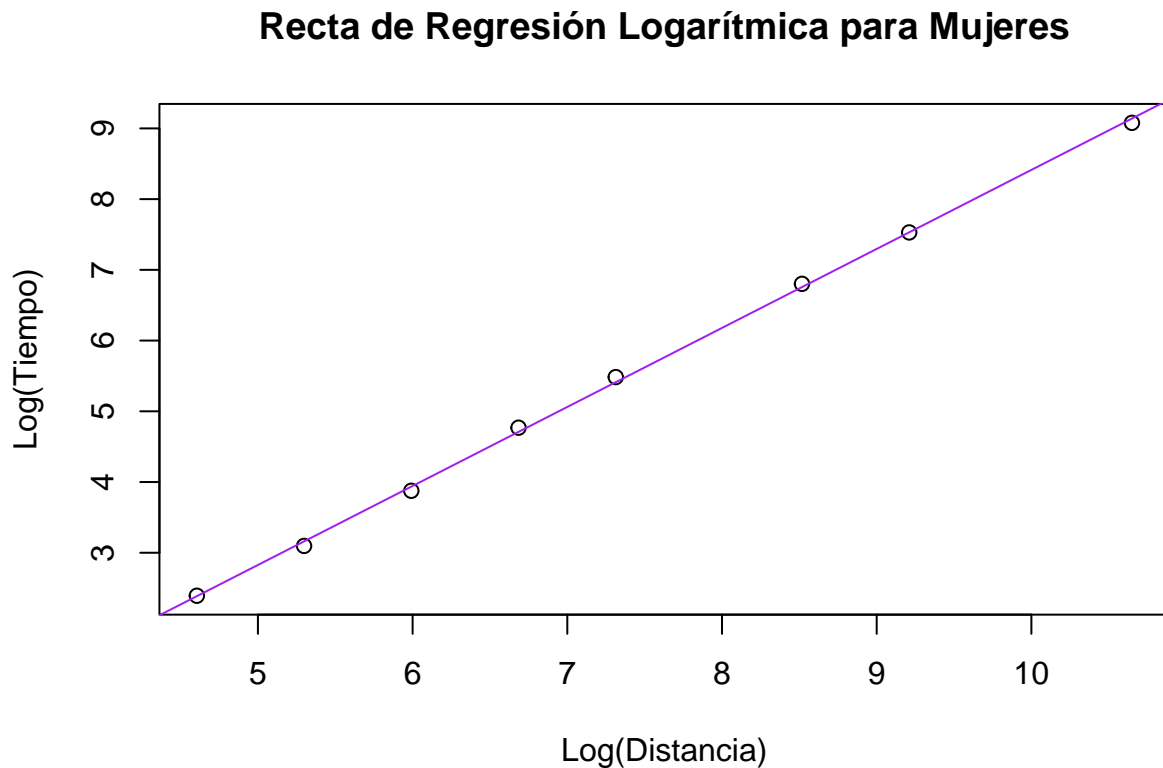
```
# Modelo logarítmico para mujeres
model_mujeres_log <- lm(log(tiempo_mujeres) ~ log(distancia))
summary(model_mujeres_log)
```

```
##
## Call:
## lm(formula = log(tiempo_mujeres) ~ log(distancia))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.06380 -0.05890  0.00246  0.04921  0.07275
##
## Coefficients:
```

```
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -2.75902    0.08392  -32.88 5.27e-08 ***
## log(distancia) 1.11721    0.01114  100.30 6.62e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06071 on 6 degrees of freedom
## Multiple R-squared:  0.9994, Adjusted R-squared:  0.9993
## F-statistic: 1.006e+04 on 1 and 6 DF, p-value: 6.618e-11
```

```
# Gráfico de la recta de regresión logarítmica para mujeres
```

```
plot(log(distancia), log(tiempo_mujeres), main = "Recta de Regresión Logarítmica para Mujeres", xlab = "Log(Distancia)", ylab = "Log(Tiempo)", col = "purple")
```

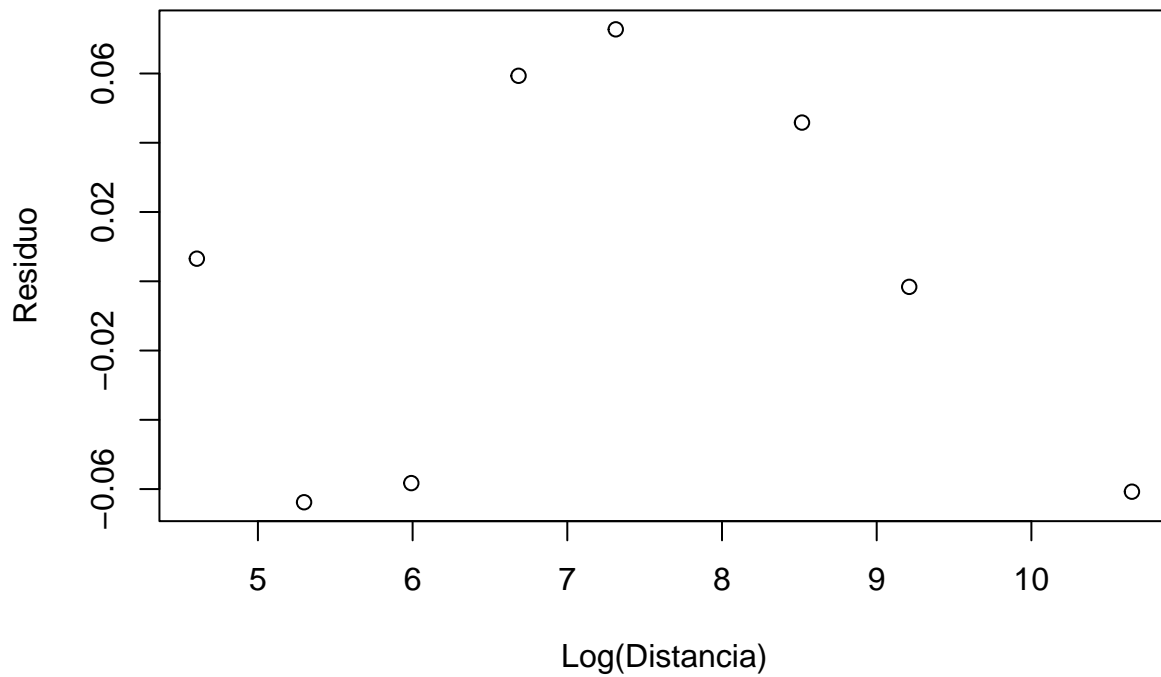


```
# Residuos del modelo logarítmico para mujeres
```

```
residuos_mujeres_log <- resid(model_mujeres_log)
```

```
plot(log(distancia), residuos_mujeres_log, main = "Residuos de la Regresión Logarítmica para Mujeres", xlab = "Log(Distancia)", ylab = "Residuos", col = "purple")
```

## Residuos de la Regresión Logarítmica para Mujeres



```
# Suma de cuadrados de los residuos y R^2 para el modelo logarítmico de mujeres  
suma_cuadrados_residuos_mujeres_log <- sum(residuos_mujeres_log^2)  
print(suma_cuadrados_residuos_mujeres_log)
```

```
## [1] 0.02211633
```

```
r2_mujeres_log <- summary(model_mujeres_log)$r.squared  
print(r2_mujeres_log)
```

```
## [1] 0.999404
```