13. Regresión No Lineal

Juan Bernal

2024-09-12

El objetivo es encontrar el mejor modelo que relacione la velocidad de los automóviles y las distancias necesarias para detenerse en autos de modelos existentes en 1920 (base de datos car). La ecuación encontrada no sólo deberá ser el mejor modelo obtenido sino también deberá ser el más económico en terminos de la complejidad del modelo.

Parte 1: Análisis de normalidad

1. Accede a los datos de cars en R (data = cars)

```
data = cars
head(data)
```

```
##   speed dist
## 1     4    2
## 2     4   10
## 3     7    4
## 4     7   22
## 5     8   16
## 6     9   10
```

*Prueba normalidad univariada de la velocidad y distancia (prueba con dos de las pruebas vistas en clase)

Prueba de hipótesis:

- H_0 : Los datos provienen de una población normal
- H_1 : Los datos no provienen de una población normal

Regla de decisión: $p - value < \alpha$ se rechaza H_0

```
library(nortest)

ad.test(data$speed)
```

```
##
##  Anderson-Darling normality test
##
## data:  data$speed
## A = 0.26143, p-value = 0.6927
```

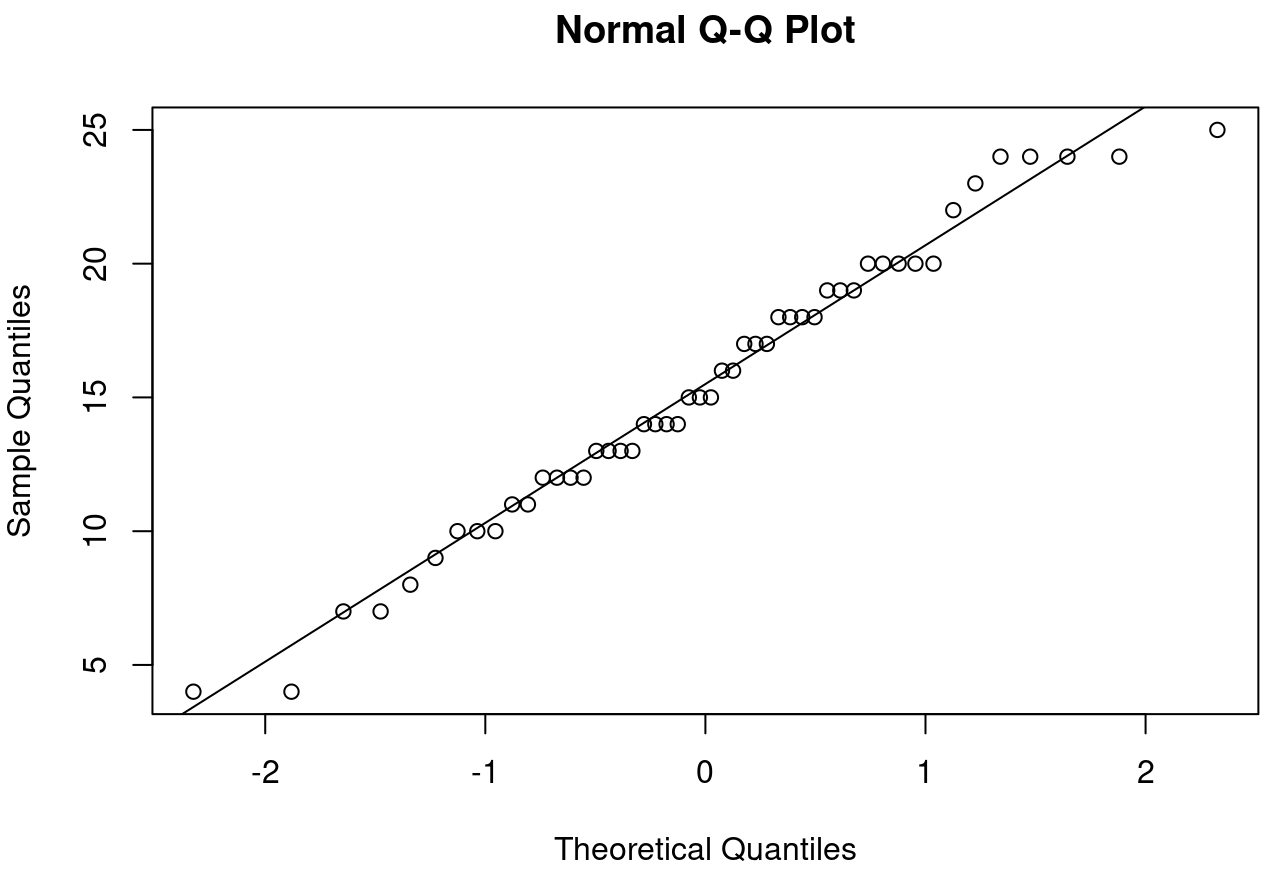
```
ad.test(data$dist)
```

```
##
##  Anderson-Darling normality test
##
## data:  data$dist
## A = 0.74067, p-value = 0.05021
```

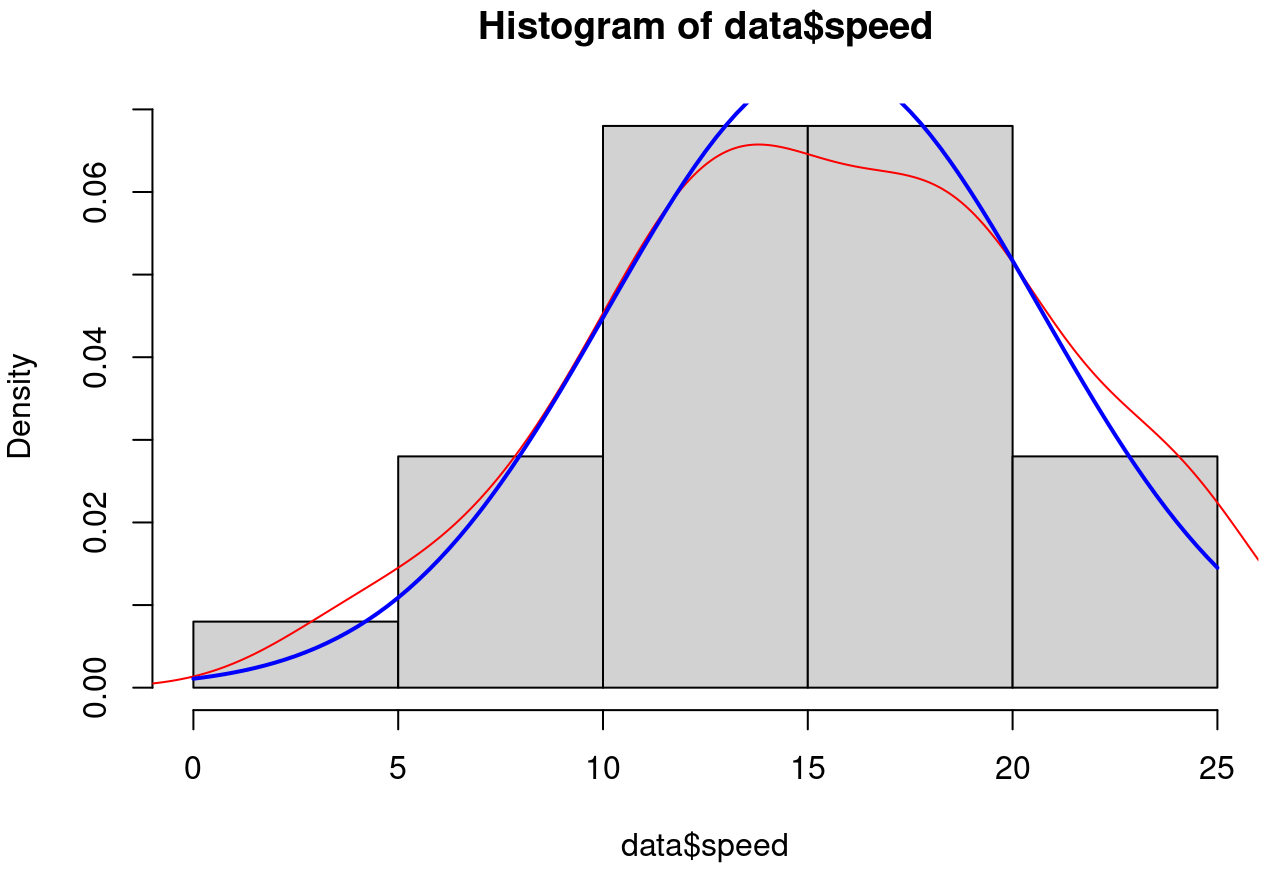
Con un nivel de significancia de 0.03, no tenemos suficiente evidencia para rechazar la hipótesis inicial, por lo que los datos provienen de una población normal, es decir, la distancia y velocidad son normales.

*Realiza gráficos que te ayuden a identificar posibles alejamientos de normalidad:

```
qqnorm(data$speed)
qqline(data$speed)
```



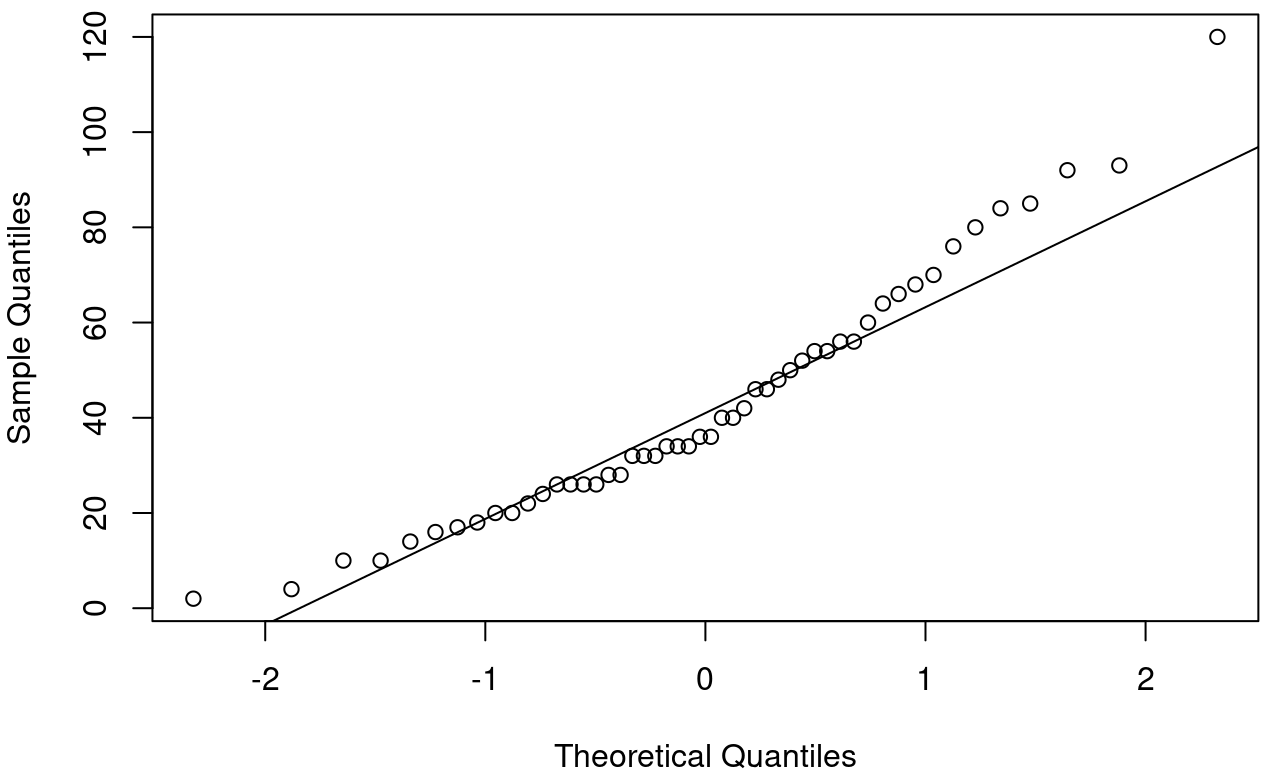

```
hist(data$speed,freq=FALSE)
lines(density(data$speed),col="red")
curve(dnorm(x,mean=mean(data$speed),sd=sd(data$speed)), add=TRUE, col="blue",lwd=2)
```



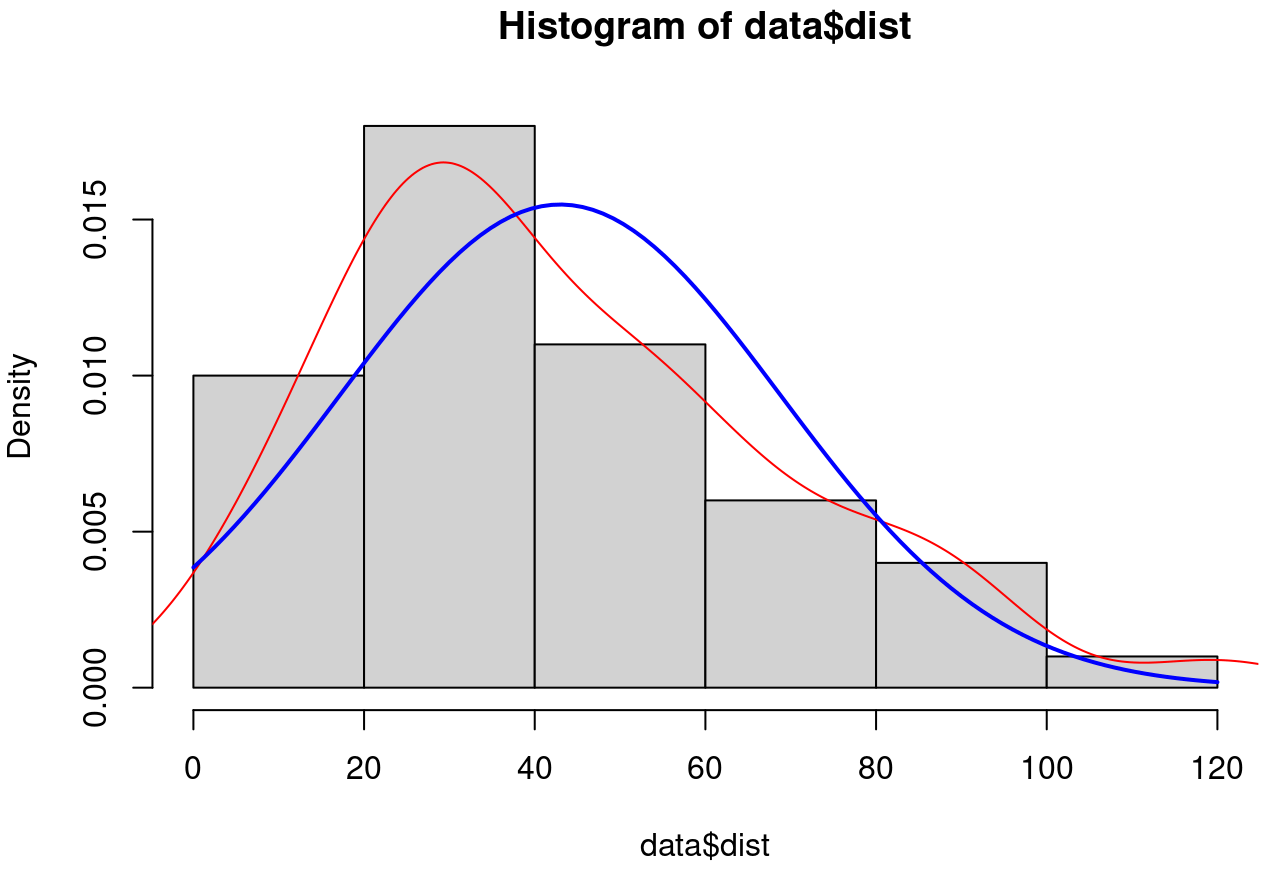

```
par(mfrow=c(1,2))
```

El qqplot de la velocidad demuestra que es normal y no se observa ningún sesgo aparente. Además, el histograma muestra una campana de Gauss y se observa simetría en las frecuencias.

```
qqnorm(data$dist)
qqline(data$dist)
```




```
hist(data$dist,freq=FALSE)
lines(density(data$dist),col="red")
curve(dnorm(x,mean=mean(data$dist),sd=sd(data$dist)), add=TRUE, col="blue",lwd=2)
```




```
par(mfrow=c(1,2))
```

El qqplot revela un sesgo hacia la derecha, y el histograma lo reafirma, al ver que las frecuencias se acumulan a la izquierda.

*Calcula el coeficiente de sesgo y el coeficiente de curtosis (sugerencia: usar la librería e1071, usar: skeness y kurtosis) para cada variable.

```
library(e1071)
cat("La curtosis de los datos de velocidad es ",kurtosis(data$speed), " y el sesgo es", skewness(data$spee
d), '\n')
```

```
## La curtosis de los datos de velocidad es  -0.6730924  y el sesgo es -0.1105533
```

```
cat("La curtosis de los datos de distancia es ",kurtosis(data$dist), " y el sesgo es", skewness(data$dist))
```

```
## La curtosis de los datos de distancia es  0.1193971  y el sesgo es 0.7591268
```

2. Comenta cada gráfico y resultado que hayas obtenido. Emite una conclusión final sobre la normalidad de los datos. Argumenta basándote en todos los análisis

realizados en esta parte. Incluye posibles motivos de alejamiento de normalidad.

Si bien ambas variables aparentan ser normales en las pruebas de hipótesis, notemos que el p-value de la distancia es mucho menor a la de lo velocidad, por lo que existen ciertos valores de α para los cuales la distancia ya no sería normal y la velocidad sí, es decir, la normalidad de la distancia no es tan aparente ni significativa como la normalidad de la velocidad. Apoyémnos en las gráficas, estas demuestran a la velocidad como una campana de Gauss simétrica, mientras que la distancia presenta un sesgo hacia la derecha, explicando porqué el valor numérico del sesgo es cercano a 1.

Parte 2: Regresión lineal

1. Prueba regresión lineal simple entre distancia y velocidad. Usa lm(y~x).

```
rl = lm(data$dist ~ data$speed)
summary(rl)
```

```
##
## Call:
## lm(formula = data$dist ~ data$speed)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -29.069  -9.525  -2.272   9.215  43.201
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -17.5791     6.7584  -2.601   0.0123 *
## data$speed    3.9324     0.4155   9.464 1.49e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 15.38 on 48 degrees of freedom
## Multiple R-squared:  0.6511, Adjusted R-squared:  0.6438
## F-statistic: 89.57 on 1 and 48 DF,  p-value: 1.49e-12
```

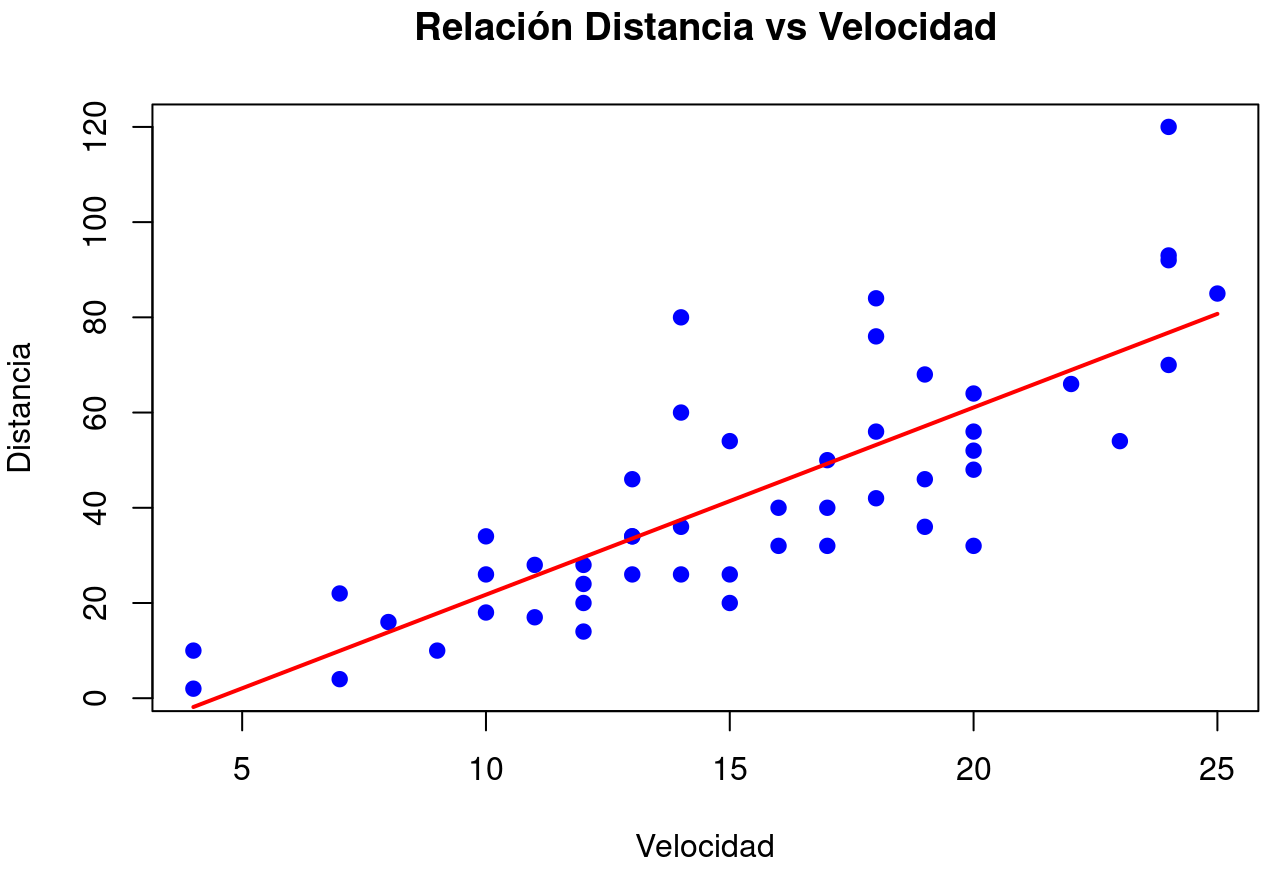
*Escribe el modelo lineal obtenido.

El modelo obtenido donde “Distancia” es la variable dependiente y “Velocidad” es la variable independiente es:

- Distancia = -17.5791 + 3.9324*Velocidad

*Grafica los datos y el modelo (ecuación) que obtuviste.

```
b0 = rl$coefficients[1] # Beta 0
b1 = rl$coefficients[2] # Beta 1

p = function(x){b0 + b1*x}

plot(data$speed, data$dist, col = 'blue', pch = 19, ylab = "Distancia", xlab = "Velocidad", main = "Relació
n Distancia vs Velocidad")
xx = seq(min(data$speed), max(data$speed), 0.01)
lines(xx, p(xx), col = 'red', lwd=2)
```



2. Analiza significancia del modelo: individual, conjunta y coeficiente de determinación. Usa summary(Modelo)

Hipótesis del modelo:

- $H_0 : \beta = 0$ El modelo no es significante
- $H_1 : \beta \neq 0$ El modelo es significante

Hipótesis de parámetros:

- $H_0 : \beta_0 = \beta_1 = 0$ El parámetro β_i no es significante
- $H_1 : \exists \beta_i \neq 0$ El parámetro β_i es significante

```
summary(rl)
```

```
##
## Call:
## lm(formula = data$dist ~ data$speed)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -29.069  -9.525  -2.272   9.215  43.201
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -17.5791     6.7584  -2.601   0.0123 *
## data$speed    3.9324     0.4155   9.464 1.49e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 15.38 on 48 degrees of freedom
## Multiple R-squared:  0.6511, Adjusted R-squared:  0.6438
## F-statistic: 89.57 on 1 and 48 DF,  p-value: 1.49e-12
```

A un nivel de significancia de 0.03, tanto el modelo como sus coeficientes son todos significantes, pues contamos con la suficiente evidencia para rechazar la hipótesis inicial. Además, el modelo explica el 65.11% de la variación de los datos.

3. Analiza validez del modelo.

*Residuos con media cero

Prueba de hipótesis:

- $H_0 : \mu = 0$
- $H_1 : \mu \neq 0$

Regla de decisión: $p - value < \alpha$ se rechaza H_0

```
t.test(rl$residuals)
```

```
##
##  One Sample t-test
##
## data:  rl$residuals
## t = -1.7754e-16, df = 49, p-value = 1
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -4.326  4.326
## sample estimates:
##     mean of x
## -3.821856e-16
```

Dado el p-value obtenido y un nivel de significancia del 0.03, no contamos con la suficiente evidencia para rechazar la hipótesis inicial, por lo que los residuos tienen media cero.

*Normalidad de los residuos

Prueba de hipótesis:

- H_0 : Los datos provienen de una población normal
- H_1 : Los datos no provienen de una población normal

Regla de decisión: $p - value < \alpha$ se rechaza H_0

```
ad.test(rl$residuals)
```

```
##
##  Anderson-Darling normality test
##
## data:  rl$residuals
## A = 0.79406, p-value = 0.0369
```

Dado el nivel de significancia del 0.03 y el p-value obtenido, no contamos con la suficiente evidencia para rechazar la hipótesis inicial, por lo que los residuos provienen de una población normal.

*Homocedasticidad, independencia y linealidad.

Prueba de hipótesis para homocedasticidad:

- H_0 : La varianza de los errores es constante (homocedasticidad)
- H_1 : La varianza de los errores no es constante (heterocedasticidad)

Regla de decisión: $p - value < \alpha$ se rechaza H_0

Prueba de hipótesis para independencia:

- H_0 : Los errores no están correlacionados
- H_1 : Los errores están correlacionados

Prueba de hipótesis para linealidad:

- H_0 : No hay términos omitidos que indican linealidad
- H_1 : Hay una especificación errónea en el modelo que indica no linealidad

Regla de decisión: $p - value < \alpha$ se rechaza H_0

```
library(lmtest)
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
dwtest(rl) # Test de Durbin-Watson para Independencia
```

```
##
##  Durbin-Watson test
##
## data:  rl
## DW = 1.6762, p-value = 0.09522
## alternative hypothesis: true autocorrelation is greater than 0
```

```
bptest(rl) # Test de Breusch-Pagan para Homocedasticidad
```

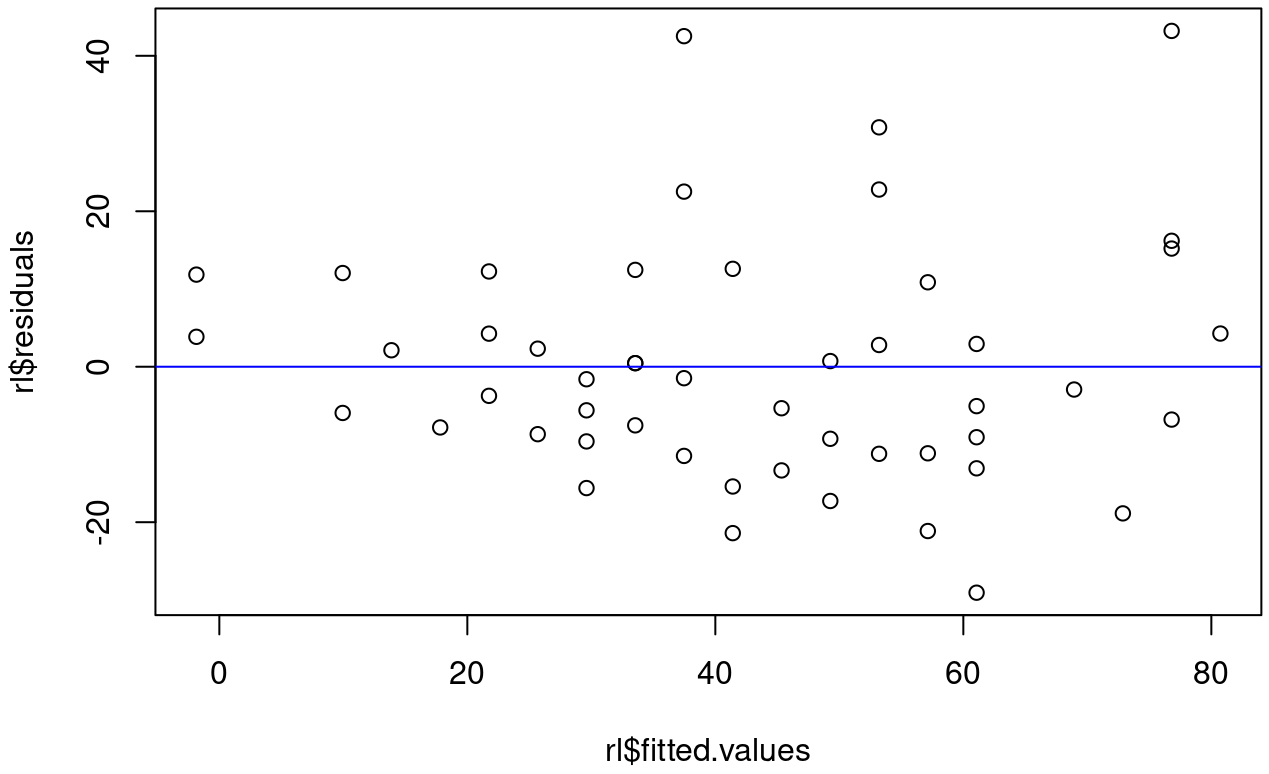
```
##
##  studentized Breusch-Pagan test
##
## data:  rl
## BP = 3.2149, df = 1, p-value = 0.07297
```

```
resettest(rl) # Test para linealidad
```

```
##
##  RESET test
##
## data:  rl
## RESET = 1.5554, df1 = 2, df2 = 46, p-value = 0.222
```

Dado que el nivel de significancia a considerar es de 0.03, no se cuenta con evidencia suficiente para rechazar ninguna de las hipótesis iniciales, por lo que los errores presentan homocedasticidad, independencia y linealidad, cumpliendo con los supuestos de validez de un modelo.

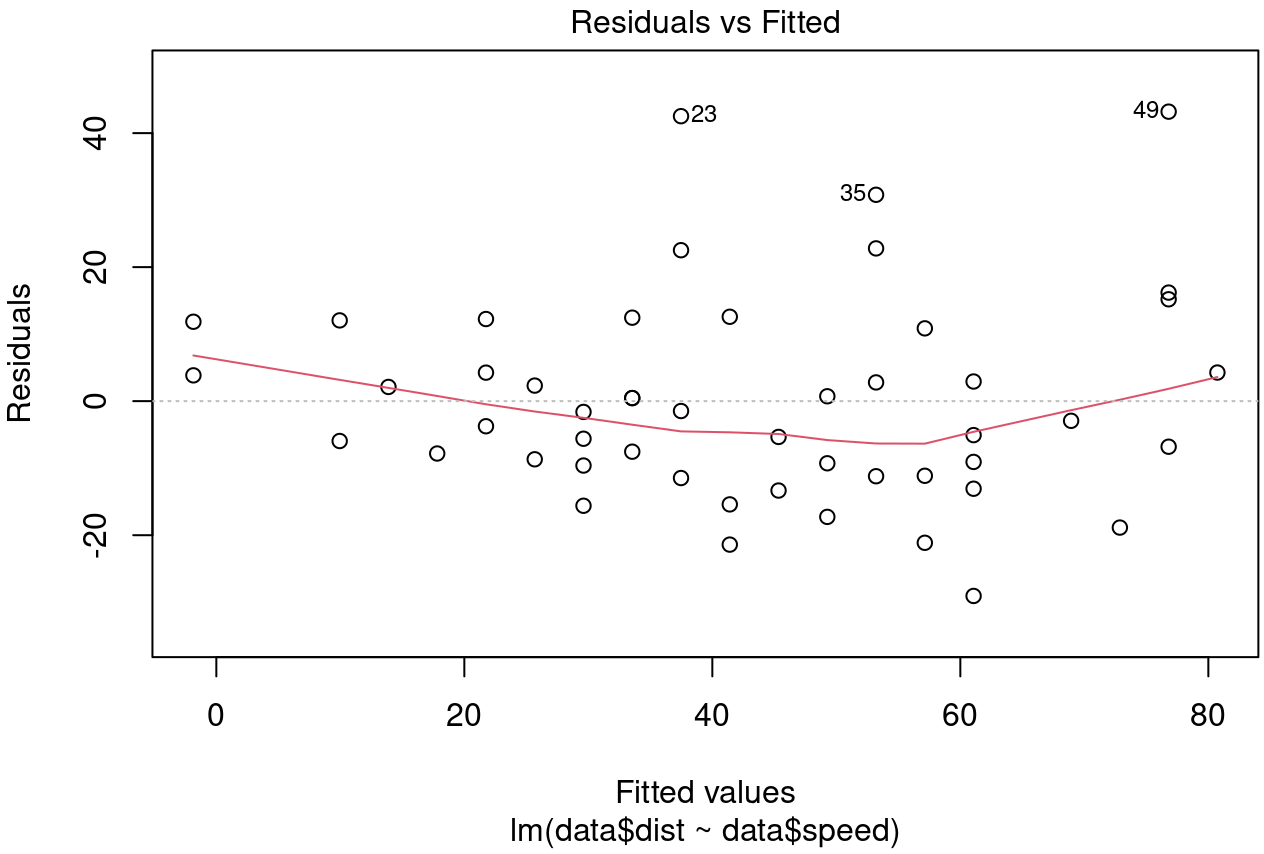
```
plot(rl$fitted.values,rl$residuals)
abline(h=0, col='blue')
```

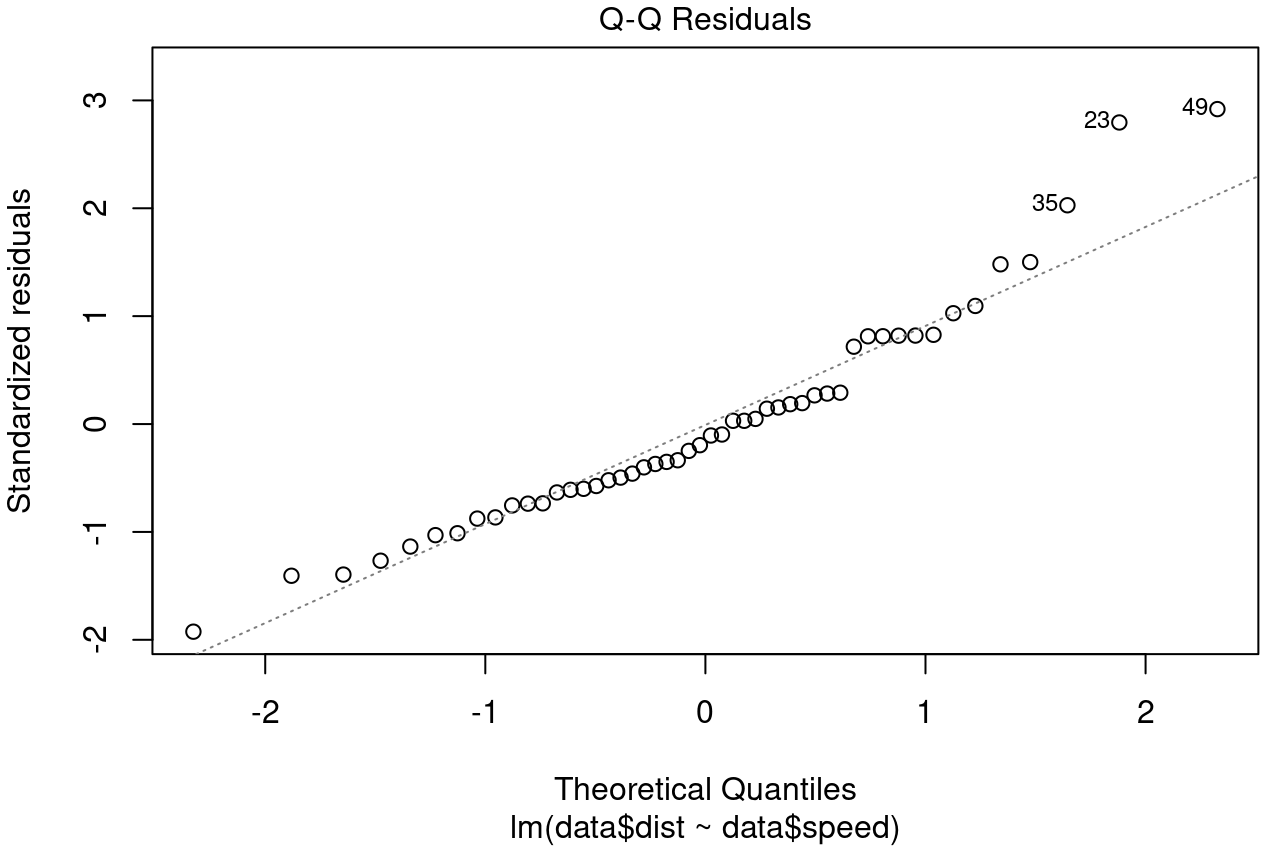


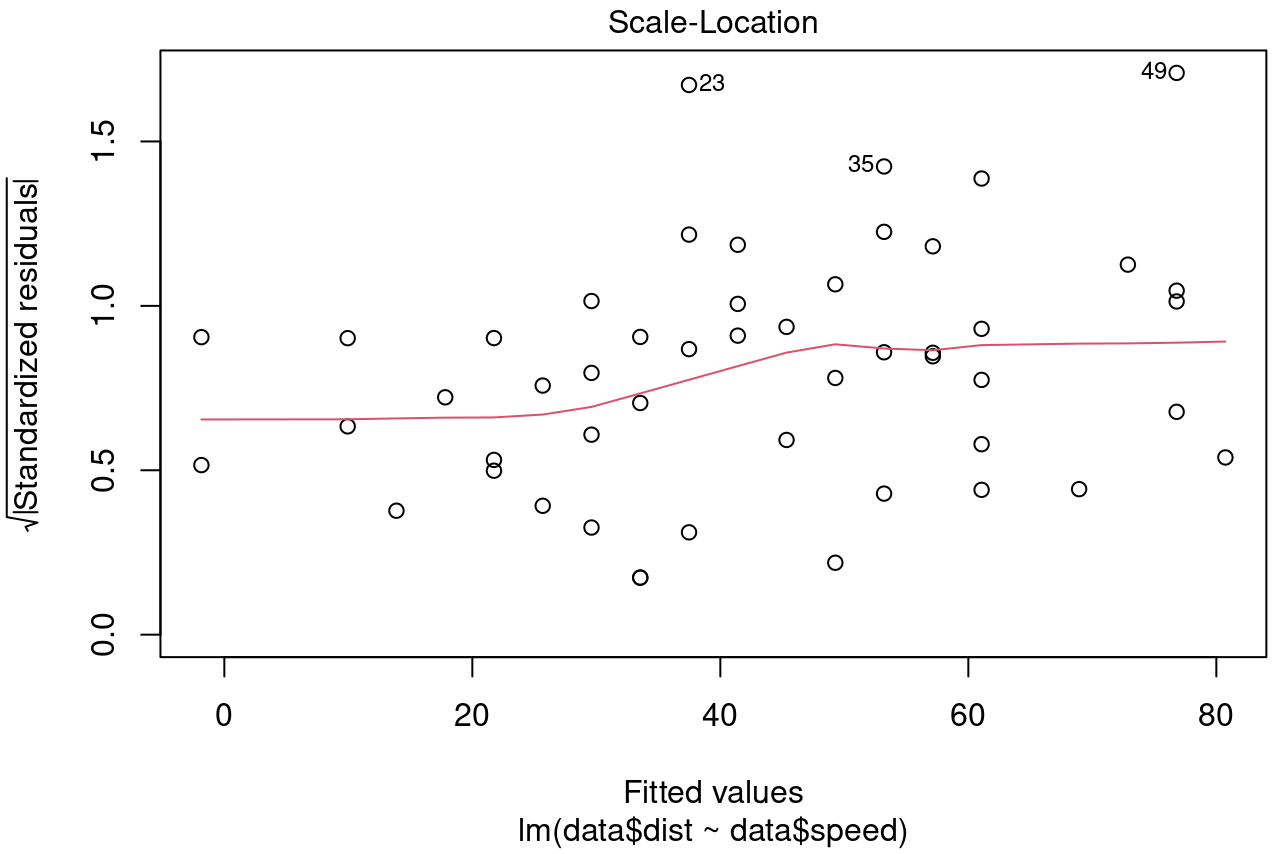
Aquí los errores no demuestran ninguna tendencia aparente, confirmando la homocedasticidad de los mismos.

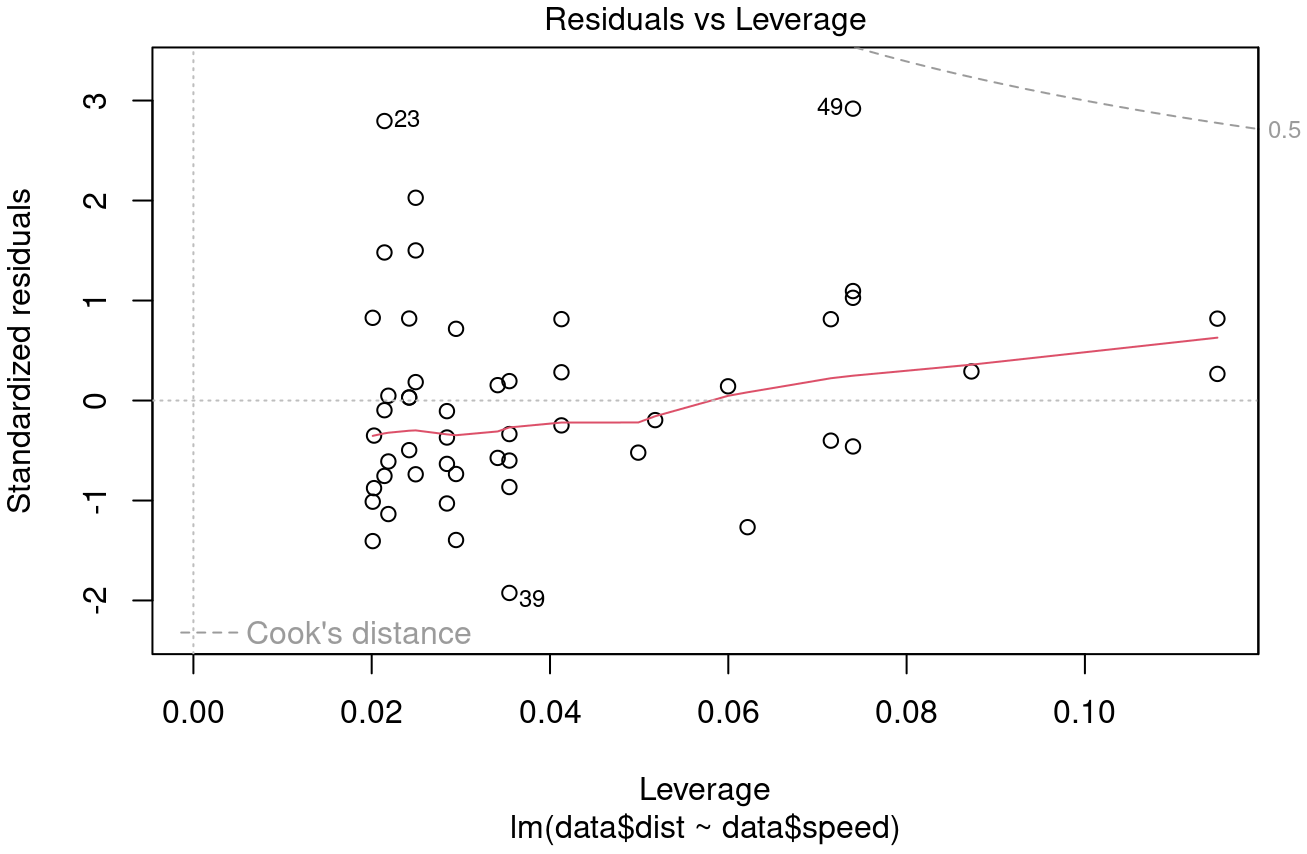
*Usa plot(Modelo) para los gráficos y añade pruebas de hipótesis.

```
plot(rl)
```


Residuals vs Fitted
Residuals
Fitted values
lm(data$dist ~ data$speed)
23
35
49
40
20
0
-20
0
20
40
60
80


Q-Q Residuals
Standardized residuals
Theoretical Quantiles
lm(data$dist ~ data$speed)
23
35
49
3
2
1
0
-1
-2
-2
-1
0
1
2

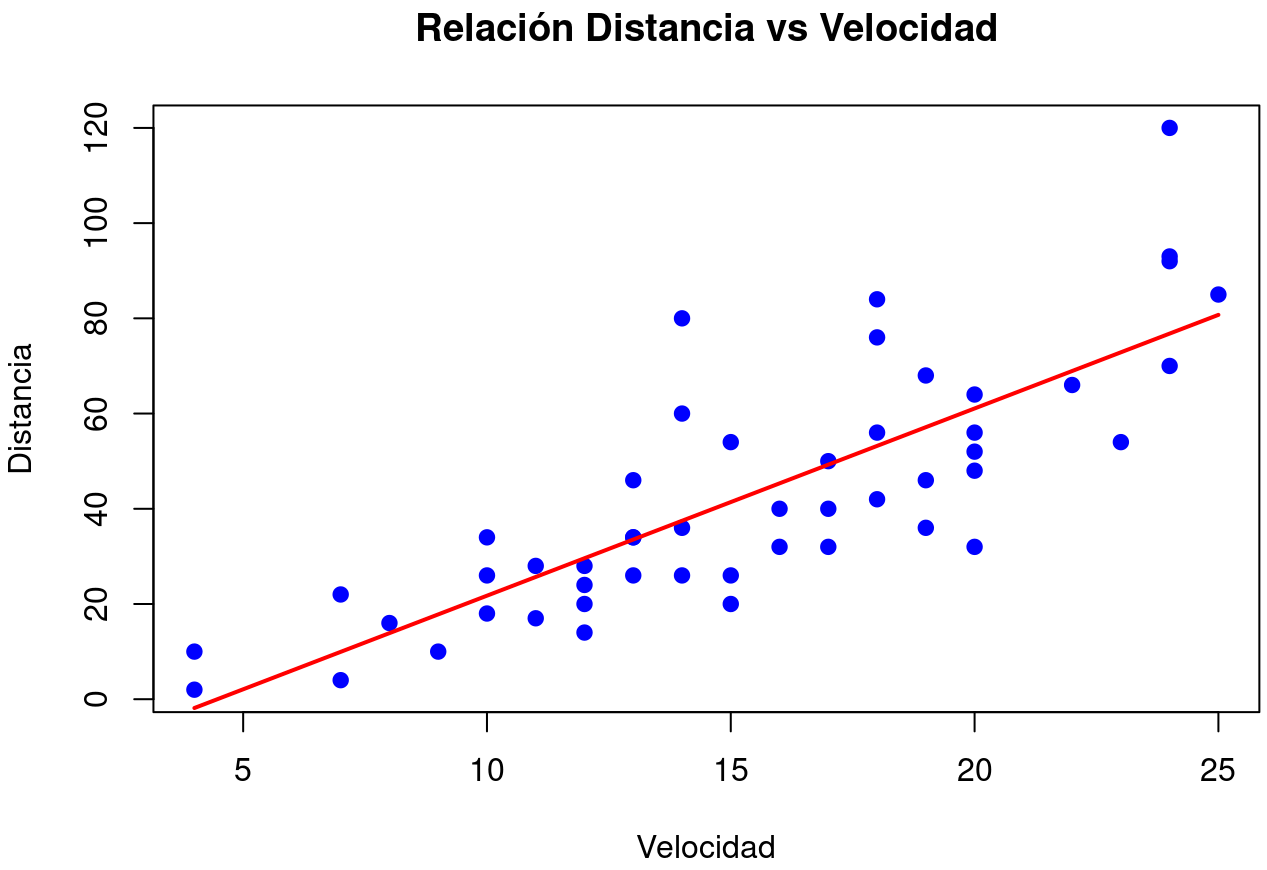

Scale-Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
23
35
49
0
20
40
60
80
Fitted values
lm(data$dist ~ data$speed)


Residuals vs Leverage
Standardized residuals
-2
-1
0
1
2
3
23
49
39
0.5
Cook's distance
0.00
0.02
0.04
0.06
0.08
0.10
Leverage
lm(data$dist ~ data$speed)

Aquí los residuos no aparentan una media cero todo el tiempo, además de que contamos con datos significantes, como el dato 23, 35 y 49, que influyen mucho en el modelo. En el qqplot notamos un sesgo a la derecha, igual que con el análisis de normalidad de la distancia, indicándonos que el modelo es apenas normal.

4. Grafica los datos y el modelo de la distancia en función de la velocidad.

```
b0 = rl$coefficients[1] # Beta 0
b1 = rl$coefficients[2] # Beta 1

p = function(x){b0 + b1*x}

plot(data$speed, data$dist, col = 'blue', pch = 19, ylab = "Distancia", xlab = "Velocidad", main = "Relació
n Distancia vs Velocidad")
xx = seq(min(data$speed), max(data$speed), 0.01)
lines(xx, p(xx), col = 'red', lwd=2)
```



5. Comenta sobre la idoneidad del modelo en función de su significancia y validez.

El modelo apenas es válido, pues no aparenta normalidad con gran significancia, además de que los errores también son “apenas” normales. El modelo tampoco es completamente idoneo, pues solo explica un 65% de la varianza de los datos, si observamos la gráfica podemos notar que la línea recta no se ajusta de manera adecuada a los datos.

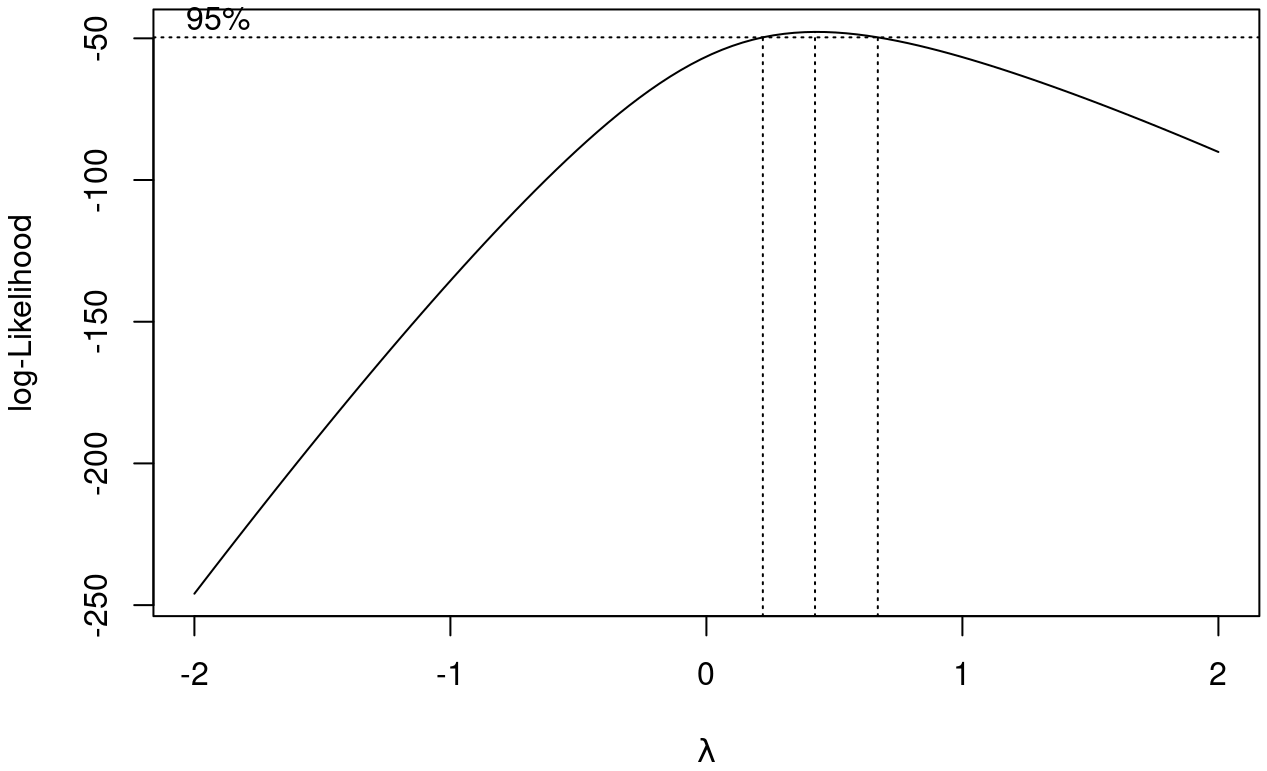
Parte 3: Regresión no lineal

1. Con el objetivo de probar un modelo no lineal que explique la relación entre la distancia y la velocidad, haz una transformación con la base de datos car que te garantice normalidad en ambas variables (ojo: concentrate solo en la variable que tiene más alejamiento de normalidad).

La variable con más alejamiento de la normalidad es la distancia, se analizó anteriormente.

*Encuentra el valor de lambda en la transformación Box-Cox para el modelo lineal: $Y = \beta_0 + \beta_1 X$ donde Y sea la distancia y X la velocidad. Aprovecha que el comando de boxcox en R te da la oportunidad de trabajar con el modelo líneal:

```
library(MASS)
bc = boxcox(lm(data$dist~data$speed))
```



```
l = bc$x[which.max(bc$y)]
cat('El mejor valor de lambda encontrado es ',l)
```

```
## El mejor valor de lambda encontrado es  0.4242424
```

*Define la transformación exacta y el aproximada de acuerdo con el valor de lambda que encontraste en la transformación de Box y Cox. Escribe las ecuaciones de las dos transformaciones encontradas.

x: Distancia

El modelo aproximado queda como $x_1 = \sqrt{x}$, y el modelo exacto queda como $x_2 = \frac{x^{0.4242}-1}{0.4242}$.

```
x1 = sqrt(data$dist) # Modelo 1
x2 = ((data$dist)^l - 1)/l # Modelo 2
```

*Analiza la normalidad de las transformaciones obtenidas. Utiliza como argumento de normalidad:

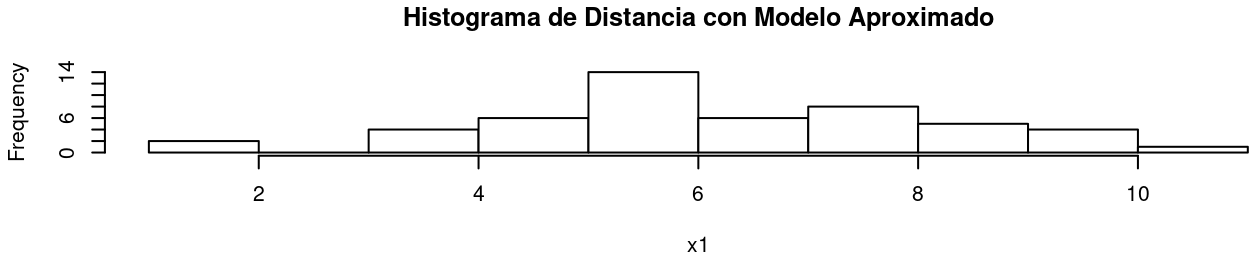
- 1. Compara las medidas: sesgo y curtosis.

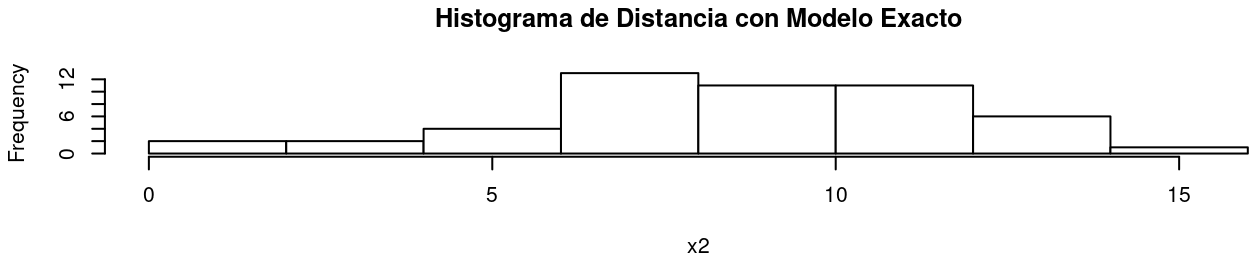
```
m0=round(c(as.numeric(summary(data$dist)),kurtosis(data$dist),skewness(data$dist)),3)
m1=round(c(as.numeric(summary(x1)),kurtosis(x1),skewness(x1)),3)
m2=round(c(as.numeric(summary(x2)),kurtosis(x2),skewness(x2)),3)

m<-as.data.frame(rbind(m0,m1,m2))
row.names(m)=c("Original","Modelo aproximado","Modelo exacto")
names(m)=c("Minimo","Q1","Mediana","Media","Q3","Máximo","Curtosis","Sesgo")
m
```

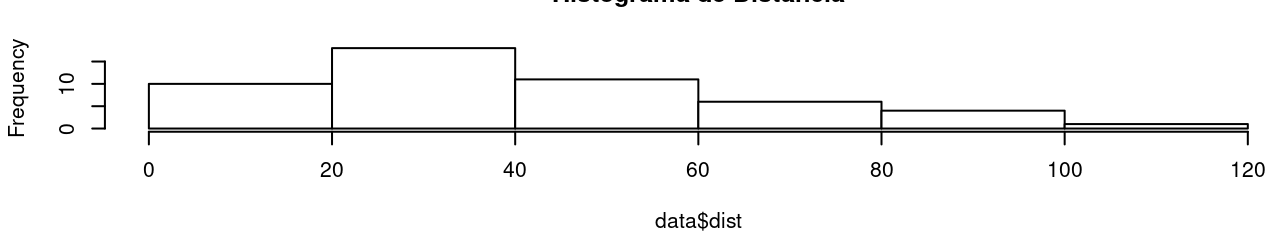
```
##                   Minimo     Q1 Mediana  Media     Q3  Máximo Curtosis  Sesgo
## Original           2.000 26.000  36.000 42.980 56.000 120.000    0.119  0.759
## Modelo aproximado  1.414  5.099   6.000  6.242  7.483  10.954   -0.314 -0.019
## Modelo exacto      0.806  7.033   8.423  8.712 10.646  15.609   -0.187 -0.170
```

- 2. Obten el histograma de los 2 modelos obtenidos (exacto y aproximado) y los datos originales.

```
par(mfrow=c(3,1))
hist(x1,col=0,main="Histograma de Distancia con Modelo Aproximado")
hist(x2,col=0,main="Histograma de Distancia con Modelo Exacto")
hist(data$dist,col=0,main="Histograma de Distancia")
```







Notemos que los datos originales muestran un sesgo a la derecha, mientras que el modelo aproximado y el modelo exacto parecen centrarse más, mas no son normales.

- 3. Realiza algunas pruebas de normalidad para los datos transformados.

Prueba de hipótesis:

- H_0 : Los datos provienen de una población normal
- H_1 : Los datos no provienen de una población normal

Regla de decisión: $p - value < \alpha$ se rechaza H_0

```
D=ad.test(x2)
print("P-value de Distancia con Modelo Exacto")
```

```
## [1] "P-value de Distancia con Modelo Exacto"
```

```
D$p.value
```

```
## [1] 0.9717478
```

Dado que el p-value es mayor a un nivel de significancia de 0.03, no hay suficiente evidencia para rechazar la hipótesis inicial, por lo que la distribución del modelo exacto es normal.

```
D=ad.test(x1)
print("P-value de Distancia con Modelo Aproximado")
```

```
## [1] "P-value de Distancia con Modelo Aproximado"
```

```
D$p.value
```

```
## [1] 0.9731952
```

Dado que el p-value es mayor a un nivel de significancia de 0.03, no hay suficiente evidencia para rechazar la hipótesis inicial, por lo que la distribución del modelo aproximado es normal.

```
D=ad.test(data$dist)
print("P-value de Distancia")
```

```
## [1] "P-value de Distancia"
```

```
D$p.value
```

```
## [1] 0.05021288
```

Dado que el p-value es mayor a un nivel de significancia de 0.03, no hay suficiente evidencia para rechazar la hipótesis inicial, por lo que la distribución del modelo original es normal.

*Detecta anomalías y corrige tu base de datos tranformado (datos atípicos, ceros anámalos, etc): solo en caso de no tener normalidad en las transformaciones. En caso de corrección de los datos por anomalías, vuelve a buscar la para tus nuevos datos.

Dado que todos los modelos son normales, no se corregirá la base de datos.

2. Concluye sobre las dos transformaciones realizadas: Define la mejor transformación de los datos de acuerdo a las características de las dos transformaciones encontradas (exacta o aproximada). Toman en cuenta la normalidad de los datos y la economía del modelo.

Es difícil definir al mejor, dado que todos presentan características diferentes. En los datos originales se obtuvo una curtosis más baja, pero un sesgo muy cercano a 1, mayor a los demás modelos. En el modelo aproximado presenta el menor sesgo de los 3 modelos y la mayor curtosis de los mismos. Y por último, está el modelo exacto, que tiene el 2do menor sesgo de los modelos, y la 2da menor curtosis, sesgo y curtosis equilibradas con valores bajos. Dadas estas características, la mejor transformación sería la del modelo aproximado, ya que aunado a estas características, también obtuvo el p-value más alto en las pruebas de normalidad, además que la curtosis y el sesgo del modelo explican en su totalidad normalidad, pues el sesgo es aproximadamente 0 y la curtosis es menor a 1.

3. Con la mejor transformación (punto 2), realiza la regresión lineal simple entre la mejor transformación (exacta o aproximada) y la variable velocidad:

* Escribe el modelo lineal para la transformación.

```
rl_new = lm(x1~data$speed)
summary(rl_new)
```

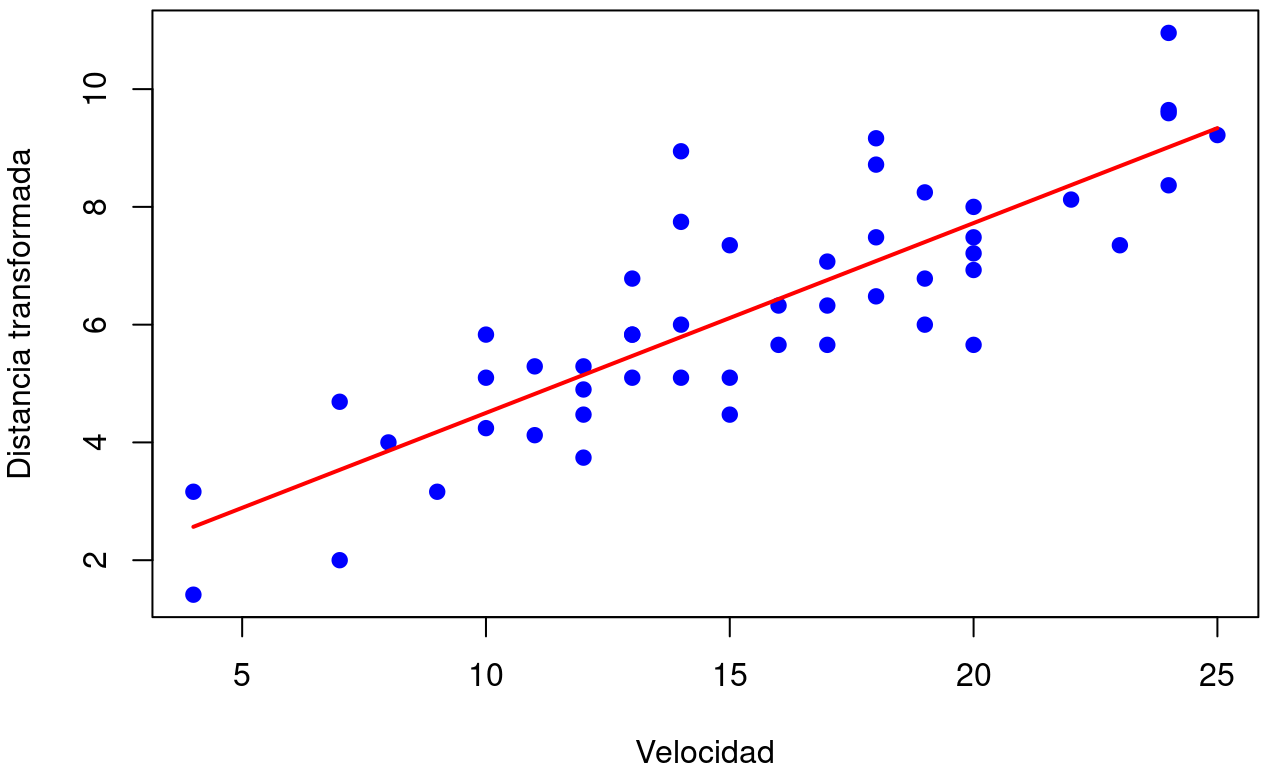
```
##
## Call:
## lm(formula = x1 ~ data$speed)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.0684 -0.6983 -0.1799  0.5909  3.1534
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.27705    0.48444   2.636   0.0113 *
## data$speed   0.32241    0.02978  10.825 1.77e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.102 on 48 degrees of freedom
## Multiple R-squared:  0.7094, Adjusted R-squared:  0.7034
## F-statistic: 117.2 on 1 and 48 DF,  p-value: 1.773e-14
```

El modelo de regresión lineal con la Distancia del “modelo exacto” de BoxCox es:

- Distancia Transformada = 1.27705 + 0.32241*Velocidad

* Grafica los datos y el modelo lineal (ecuación) de la transformación elegida vs velocidad.

```
b0 = rl_new$coefficients[1] # Beta 0
b1 = rl_new$coefficients[2] # Beta 1

p = function(x){b0 + b1*x}

plot(data$speed, x1, col = 'blue', pch = 19, ylab = "Distancia transformada", xlab = "Velocidad", main = "R
elación Distancia vs Velocidad")
xx = seq(min(data$speed), max(data$speed), 0.01)
lines(xx, p(xx), col = 'red', lwd=2)
```



* Analiza significancia del modelo (individual, conjunta y coeficiente de correlación)

Hipótesis del modelo:

- $H_0 : \beta = 0$ El modelo no es significante
- $H_1 : \beta \neq 0$ El modelo es significante

Hipótesis de parámetros:

- $H_0 : \beta_0 = \beta_1 = 0$ El parámetro β_i no es significante
- $H_1 : \exists \beta_i \neq 0$ El parámetro β_i es significante

```
summary(rl_new)
```

```
##
## Call:
## lm(formula = x1 ~ data$speed)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.0684 -0.6983 -0.1799  0.5909  3.1534
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.27705    0.48444   2.636   0.0113 *
## data$speed   0.32241    0.02978  10.825 1.77e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.102 on 48 degrees of freedom
## Multiple R-squared:  0.7094, Adjusted R-squared:  0.7034
## F-statistic: 117.2 on 1 and 48 DF,  p-value: 1.773e-14
```

Dado un nivel de significancia de 0.03, hay suficiente evidencia para rechazar la hipótesis inicial de β_0, β_1 y el modelo, por lo que el modelo es significante y sus coeficientes son significantes.

* Analiza validez del modelo: normalidad de los residuos, homocedasticidad e independencia. Indica si hay candidatos a datos atípicos o influyentes en la regresión. Usa plot(Modelo) para los gráficos y añade pruebas de hipótesis.

####*Residuos con media cero

Prueba de hipótesis:

- $H_0 : \mu = 0$
- $H_1 : \mu \neq 0$

Regla de decisión: $p - value < \alpha$ se rechaza H_0

```
t.test(rl_new$residuals)
```

```
##
##  One Sample t-test
##
## data:  rl_new$residuals
## t = -1.7087e-16, df = 49, p-value = 1
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -0.3100858  0.3100858
## sample estimates:
##     mean of x
## -2.636522e-17
```

Dado el p-value obtenido y un nivel de significancia del 0.03, no contamos con la suficiente evidencia para rechazar la hipótesis inicial, por lo que los residuos tienen media cero.

*Normalidad de los residuos

Prueba de hipótesis:

- H_0 : Los datos provienen de una población normal
- H_1 : Los datos no provienen de una población normal

Regla de decisión: $p - value < \alpha$ se rechaza H_0

```
ad.test(rl_new$residuals)
```

```
##
##  Anderson-Darling normality test
##
## data:  rl_new$residuals
## A = 0.39752, p-value = 0.3551
```

Dado el nivel de significancia del 0.03 y el p-value obtenido, no contamos con la suficiente evidencia para rechazar la hipótesis inicial, por lo que los residuos provienen de una población normal.

*Homocedasticidad, independencia y linealidad.

Prueba de hipótesis para homocedasticidad:

- H_0 : La varianza de los errores es constante (homocedasticidad)
- H_1 : La varianza de los errores no es constante (heterocedasticidad)

Regla de decisión: $p - value < \alpha$ se rechaza H_0

Prueba de hipótesis para independencia:

- H_0 : Los errores no están correlacionados
- H_1 : Los errores están correlacionados

Prueba de hipótesis para linealidad:

- H_0 : No hay términos omitidos que indican linealidad
- H_1 : Hay una especificación errónea en el modelo que indica no linealidad

Regla de decisión: $p - value < \alpha$ se rechaza H_0

```
dwtest(rl_new) # Test de Durbin-Watson para Independencia
```

```
##
##  Durbin-Watson test
##
## data:  rl_new
## DW = 1.9417, p-value = 0.3609
## alternative hypothesis: true autocorrelation is greater than 0
```

```
bptest(rl_new) # Test de Breusch-Pagan para Homocedasticidad
```

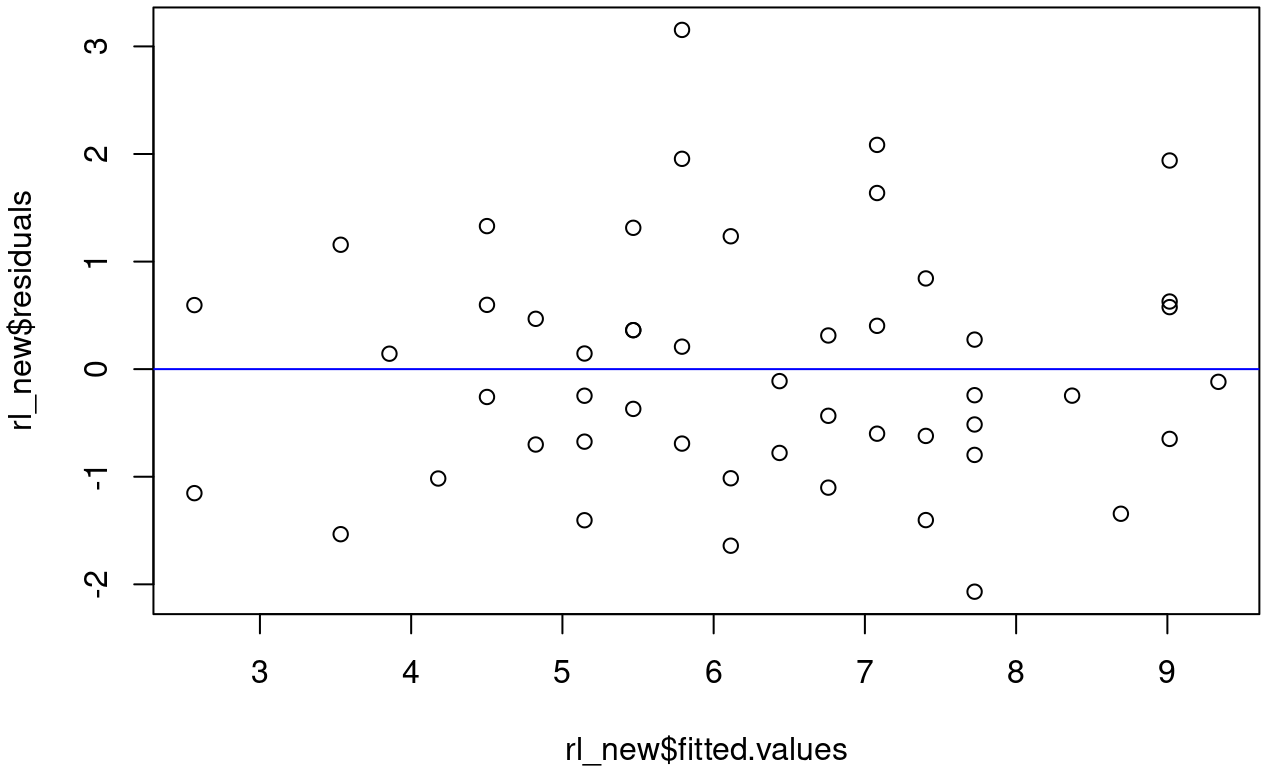
```
##
##  studentized Breusch-Pagan test
##
## data:  rl_new
## BP = 0.011192, df = 1, p-value = 0.9157
```

```
resettest(rl_new) # Test para linealidad
```

```
##
##  RESET test
##
## data:  rl_new
## RESET = 0.47002, df1 = 2, df2 = 46, p-value = 0.628
```

Dado que el nivel de significancia a considerar es de 0.03, no se cuenta con evidencia suficiente para rechazar ninguna de las hipótesis iniciales, por lo que los errores presentan homocedasticidad, independencia y linealidad, cumpliendo con los supuestos de validez de un modelo.

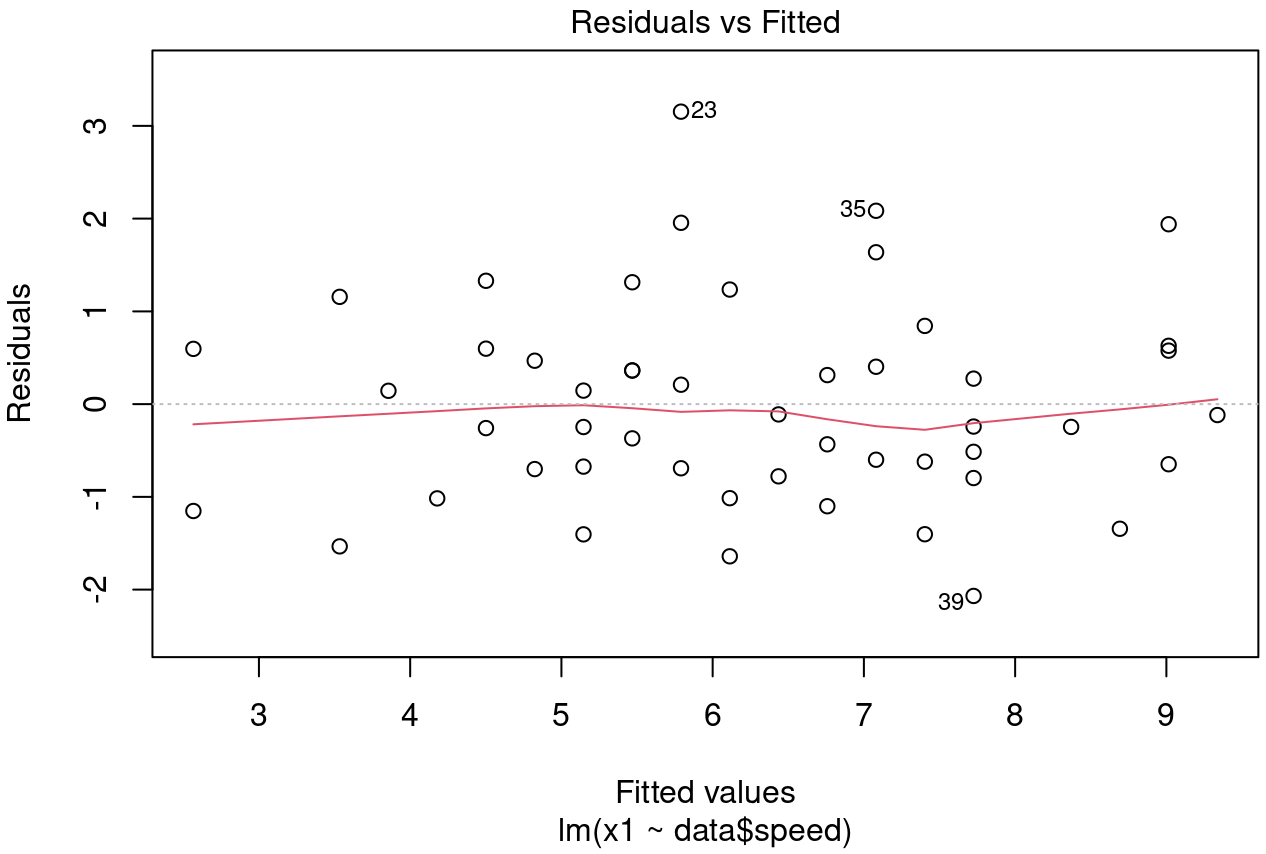
```
plot(rl_new$fitted.values,rl_new$residuals)
abline(h=0, col='blue')
```

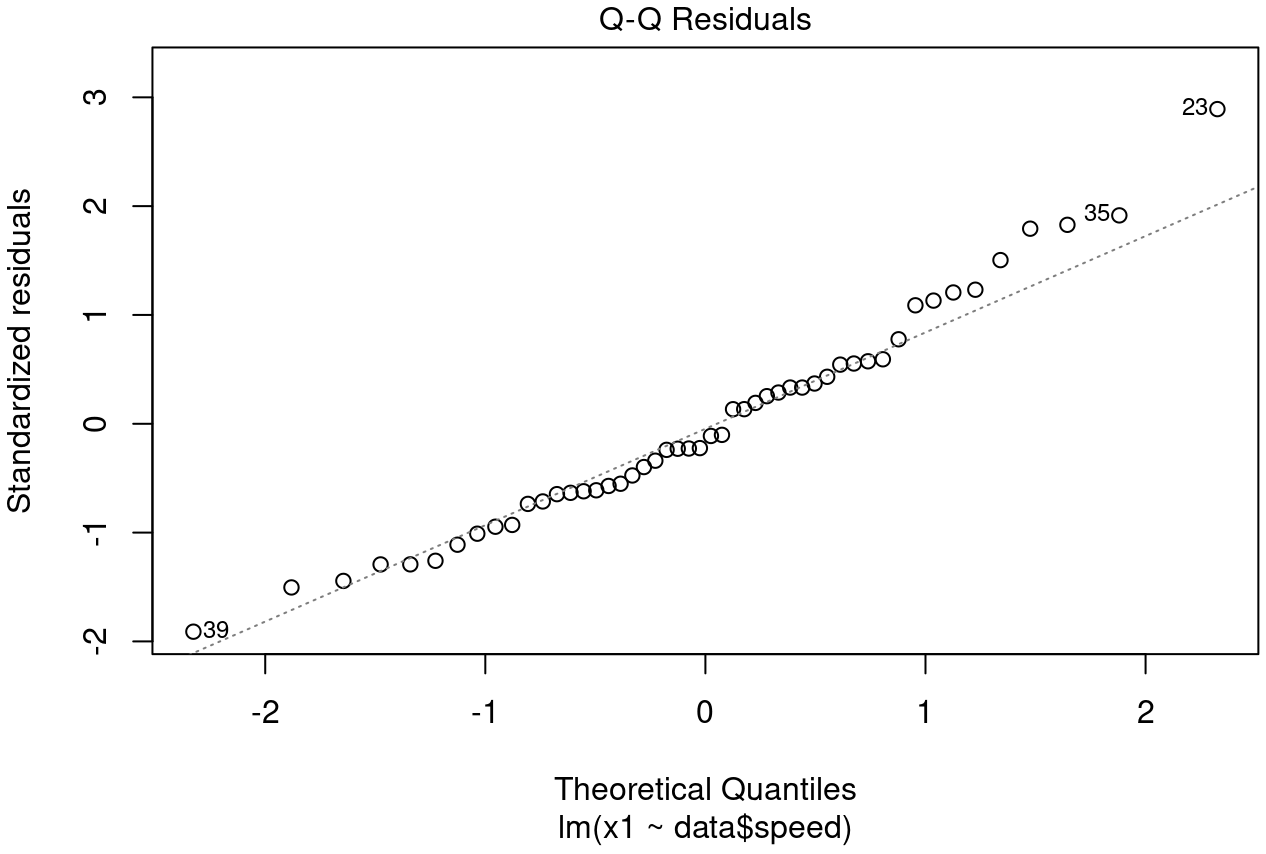


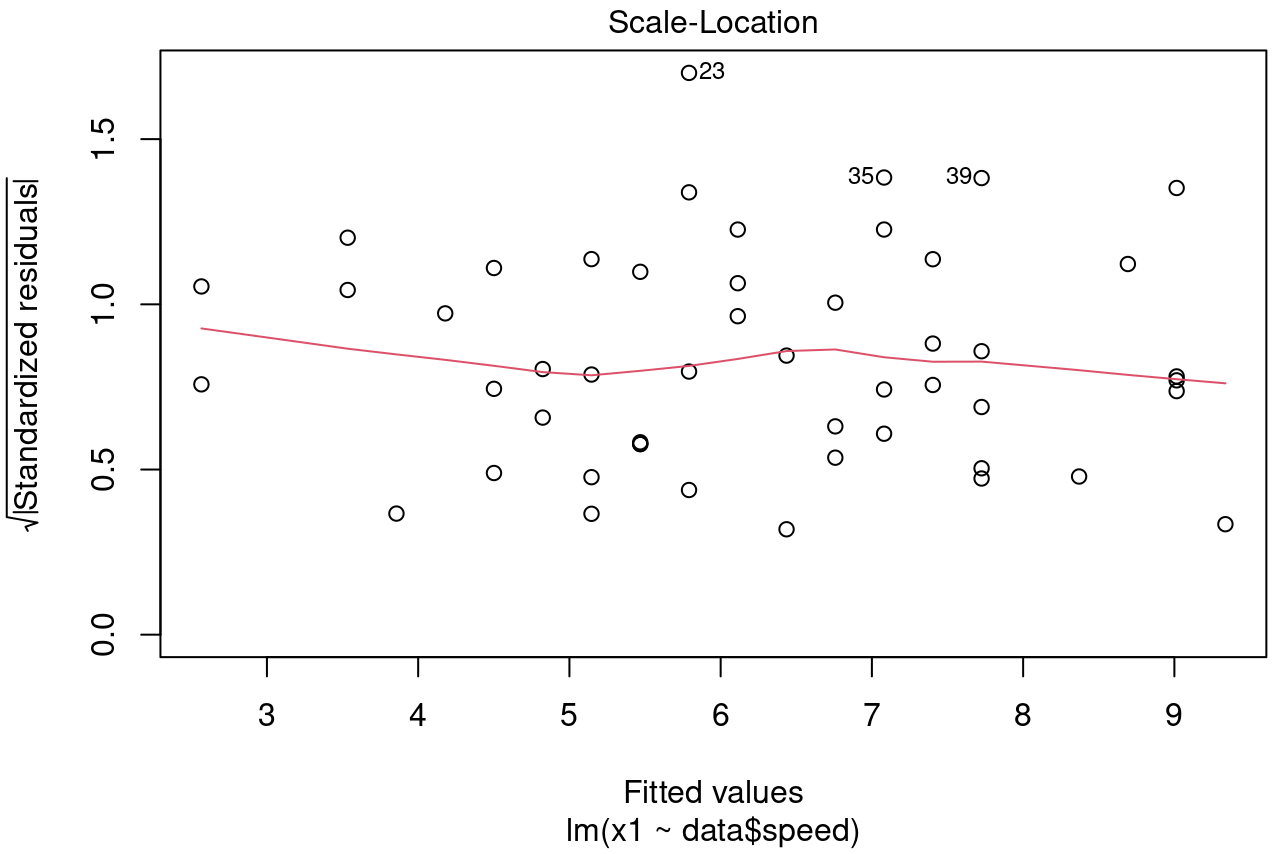
Aquí los errores no demuestran ninguna tendencia aparente, confirmando la homocedasticidad de los mismos.

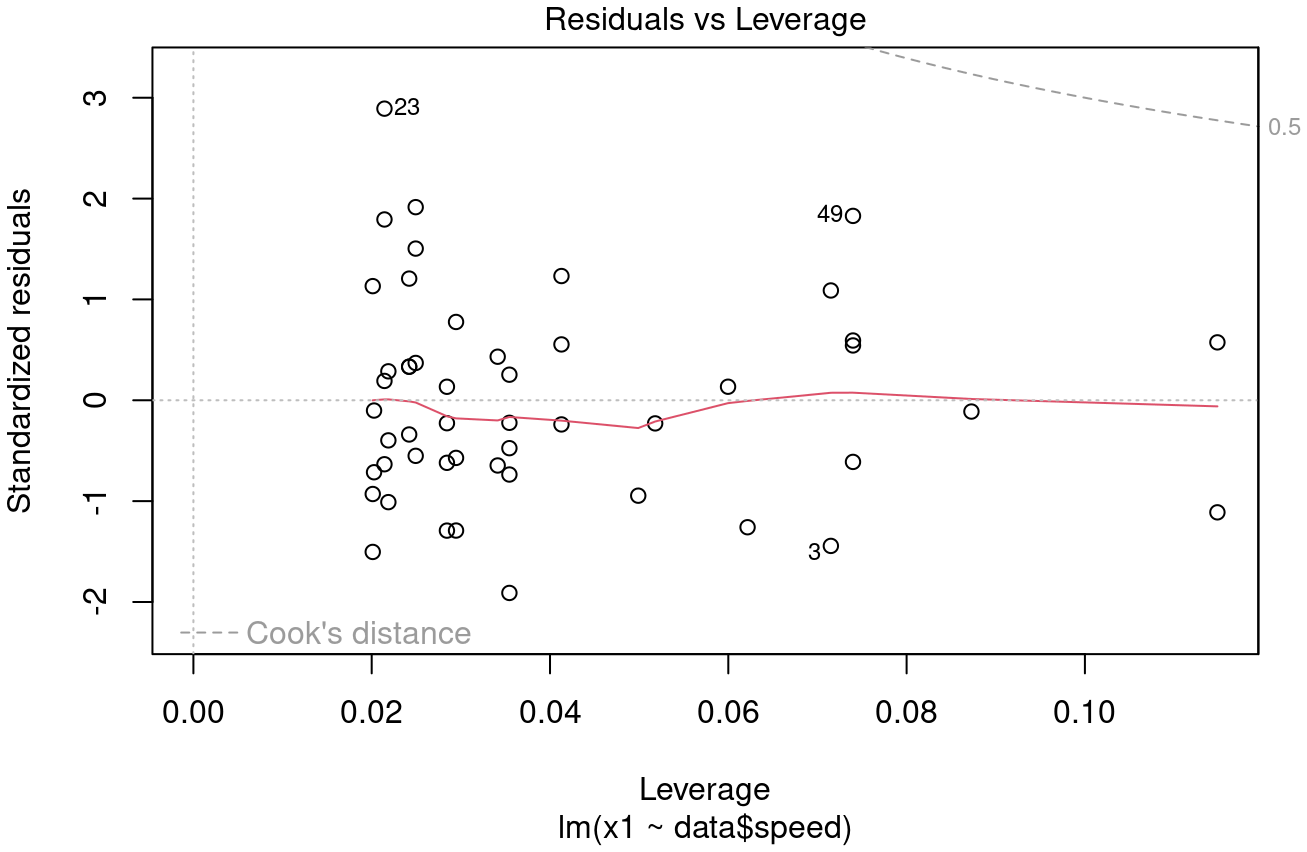
*Usa plot(Modelo) para los gráficos y añade pruebas de hipótesis.

```
plot(rl_new)
```


Residuals vs Fitted
Residuals
Fitted values
lm(x1 ~ data$speed)
23
35
39


Q-Q Residuals
Standardized residuals
Theoretical Quantiles
lm(x1 ~ data$speed)
23
35
39


Scale-Location
√|Standardized residuals|
23
35
39
0.0
0.5
1.0
1.5
3
4
5
6
7
8
9
Fitted values
lm(x1 ~ data$speed)


Residuals vs Leverage
Standardized residuals
23
49
3
0.5
Cook's distance
-2
-1
0
1
2
3
0.00
0.02
0.04
0.06
0.08
0.10
Leverage
lm(x1 ~ data$speed)

A diferencia del modelo original, aquí los residuos aparentan una media cero casi todo el tiempo, aunque seguimos contando con algunos datos significantes, como el dato 23, 35 y 39, que influyen mucho en el modelo. En el qqplot notamos un sesgo a la derecha, pero de menor intensidad que en los datos originales. Esto demuestra que el modelo aproximado se conforma de manera más “Normal” que el modelo original.

* Despeja la distancia del modelo lineal obtenido entre la transformación y la velocidad. Obtendrás el modelo no lineal que relaciona la distancia con la velocidad directamente (y no con su transformación).

Si tenemos que:

- Distancia Transformada = $\sqrt{Distancia}$

Y el modelo de distancia transformada es:

- Distancia Transformada = 1.27705 + 0.32241*Velocidad

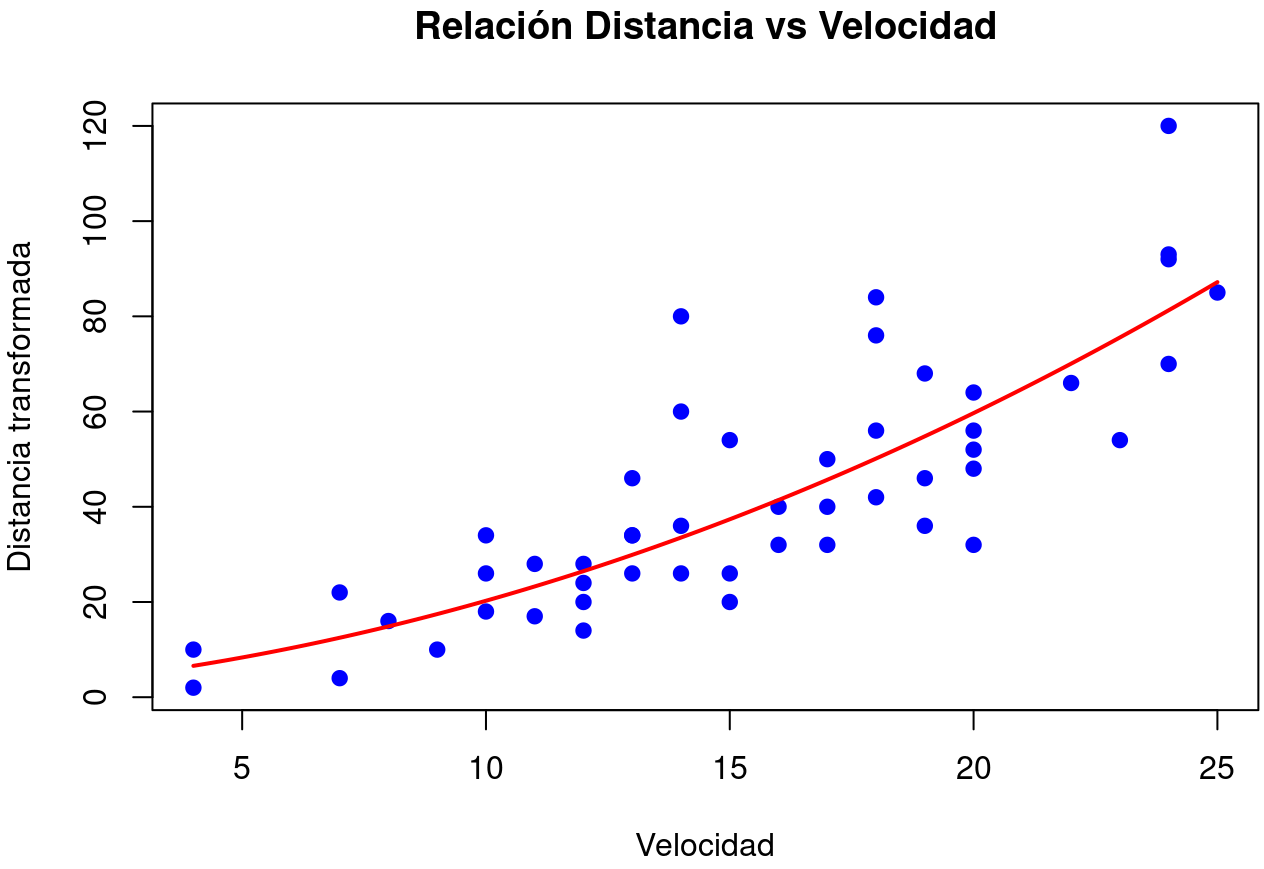
Entonces:

$\sqrt{Distancia}$ = 1.27705 + 0.32241*Velocidad

Distancia = $(1.27705 + 0.32241 Velocidad)^2$

* Grafica los datos y el modelo de la distancia en función de la velocidad.

```
b0 = rl_new$coefficients[1] # Beta 0
b1 = rl_new$coefficients[2] # Beta 1

p = function(x){(b0 + b1*x)**(2)}

plot(data$speed, data$dist, col = 'blue', pch = 19, ylab = "Distancia transformada", xlab = "Velocidad", ma
in = "Relación Distancia vs Velocidad")
xx = seq(min(data$speed), max(data$speed), 0.01)
lines(xx, p(xx), col = 'red', lwd=2)
```



* Comenta sobre la idoneidad del modelo en función de su significancia y validez.

El modelo cumple con los supuestos de normalidad de los datos, así como de homocedasticidad, linealidad e independencia de los errores. Este termina incluso por explicar mayor porcentaje de varianza que el modelo original. Este modelo explica aproximadamente el 71% de la varianza de los datos.

Parte 4: Conclusión

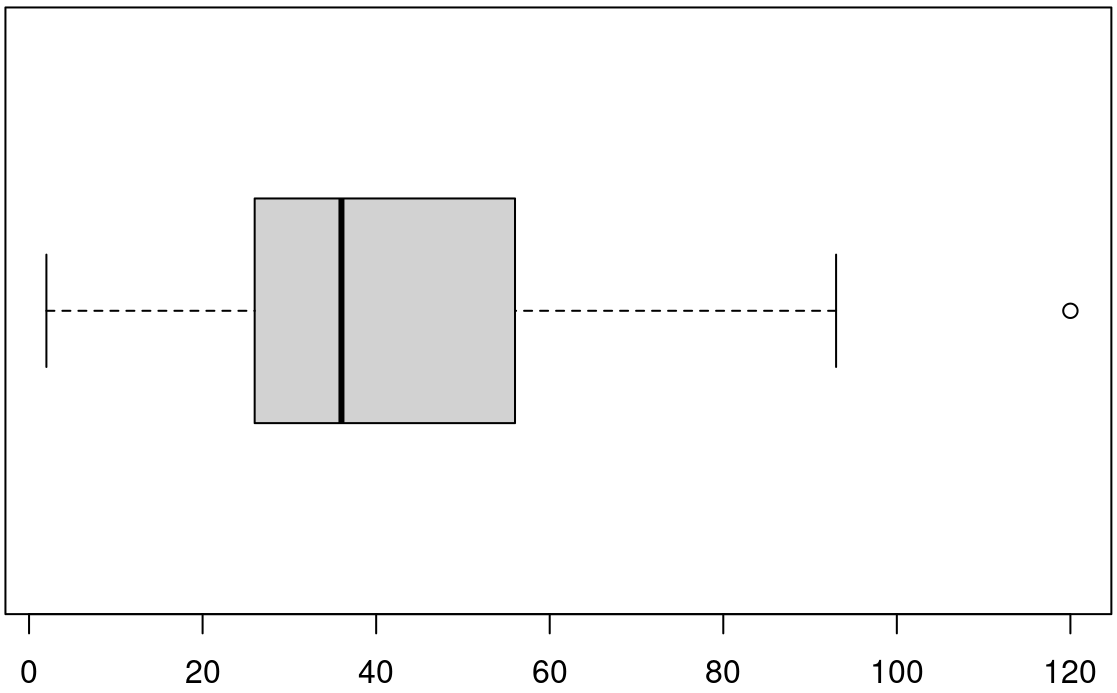
1. Define cuál de los dos modelos analizados (Punto 1 o Punto 2) es el mejor modelo para describir la relación entre la distancia y la velocidad.

Dado los análisis por puntos individuales que se fueron haciendo a lo largo del documento, el modelo que mejor describe la relación entre la distancia y la velocidad es el modelo no lineal, pues explica mejor la variación de los datos que el modelo normal, además que la transformación de la distancia “Normaliza” en mayor medida al modelo.

2. Comenta sobre posibles problemas del modelo elegido (datos atípicos, alejamiento de los supuestos, dificultad de cálculo o interpretación)

Si bien el modelo elegido es normal, sigue habiendo datos significativos y atípicos. Los valores atípicos son datos que se desvían significativamente del resto del conjunto de datos y pueden surgir por errores de medición, eventos raros o variabilidad natural. Para mitigar estos efectos, se podría realizar una detección con boxplot y tratamiento de valores atípicos con los rangos intercuartílicos o rangos de desviaciones estándar alrededor de la media antes de construir el modelo de regresión.

```
boxplot(data$dist, horizontal = TRUE)
```



También se presentan datos que parecen alejarse mucho de la curva de regresion, por lo que debe existir un modelo que se ajuste de mejor manera a los datos y cumpla con los supuestos de normalidad. Aunque, esto podría meterse con la dificultad del cálculo e interpretación del problema, ya que si buscamos la curva que MEJOR se ajuste a los datos, encontraremos un modelo matemático y computacional muy complejo que usará más recursos computacionales y será complejo de entender.