Regresión no lineal

Catherine Rojas

2024-09-10

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

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

```r
# Cargar el dataset de cars
data(cars)

# Ver las primeras filas del dataset
head(cars)
```

```
##   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

```r
# Cargar las librerías necesarias
library(nortest)
library(tseries)
```

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

## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```r
# Pruebas de normalidad

# Shapiro-Wilk Test
shapiro_test_speed <- shapiro.test(cars$speed)
shapiro_test_dist <- shapiro.test(cars$dist)
```

```r
# Anderson-Darling Test
ad_test_speed <- ad.test(cars$speed)
ad_test_dist <- ad.test(cars$dist)

# Jarque-Bera Test
jb_test_speed <- jarque.bera.test(cars$speed)
jb_test_dist <- jarque.bera.test(cars$dist)

# Crear una tabla con los resultados
resultados <- data.frame(
  Prueba = c("Shapiro-Wilk", "Shapiro-Wilk", "Anderson-Darling",
"Anderson-Darling", "Jarque-Bera", "Jarque-Bera"),
  Variable = c("Velocidad", "Distancia", "Velocidad", "Distancia",
"Velocidad", "Distancia"),
  Estadístico = c(shapiro_test_speed$statistic,
shapiro_test_dist$statistic,
                  ad_test_speed$statistic, ad_test_dist$statistic,
                  jb_test_speed$statistic, jb_test_dist$statistic),
  `Valor P` = c(shapiro_test_speed$p.value, shapiro_test_dist$p.value,
                ad_test_speed$p.value, ad_test_dist$p.value,
                jb_test_speed$p.value, jb_test_dist$p.value)
)

# Mostrar la tabla de resultados
resultados

##             Prueba  Variable Estadístico    Valor.P
## 1     Shapiro-Wilk Velocidad   0.9776489 0.45763191
## 2     Shapiro-Wilk Distancia   0.9514385 0.03909968
## 3 Anderson-Darling Velocidad   0.2614262 0.69265915
## 4 Anderson-Darling Distancia   0.7406694 0.05021288
## 5      Jarque-Bera Velocidad   0.8021706 0.66959293
## 6      Jarque-Bera Distancia   5.2304897 0.07314987
```

Observaciones

- Velocidad: Los resultados de todas las pruebas indican que la variable Velocidad puede considerarse normalmente distribuida.

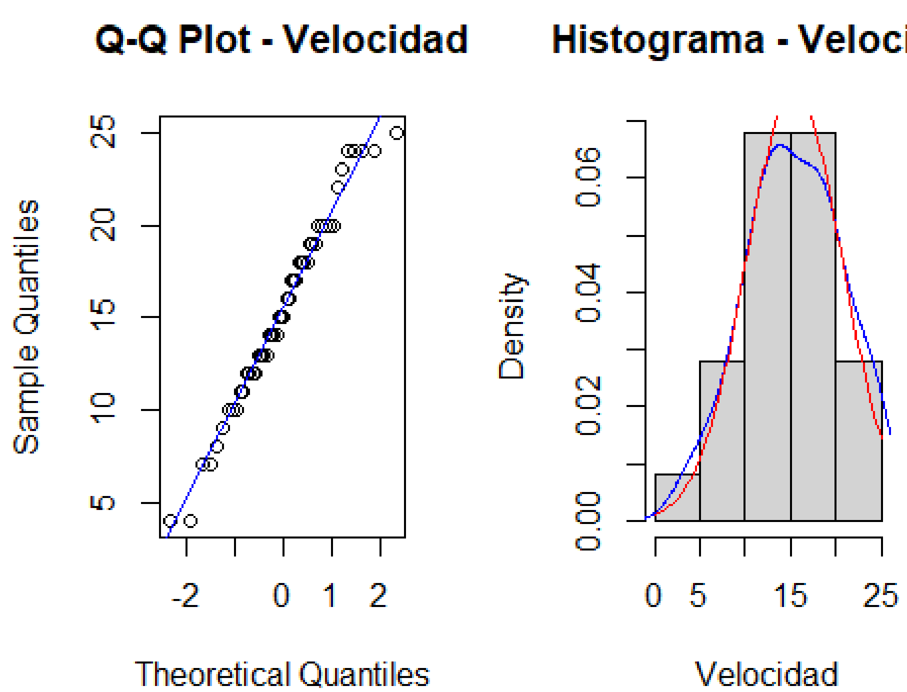
- Distancia: Parece que Distancia no sigue estrictamente una distribución normal, aunque el resultado no es concluyente en todas las pruebas. La prueba de Shapiro-Wilk rechaza la normalidad para Distancia, mientras que las otras pruebas están en un punto de indecisión.

Para un modelo de regresión, esto sugiere que se podría necesitar transformar la variable Distancia si la normalidad es un requisito importante para los residuos del modelo.

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

QQPlot: qqnorm(datos) y qqline(datos) para cada variable Realiza el histograma y su distribución teórica de probabilidad

```
# Dividir la ventana gráfica en dos gráficos: 1 fila, 2 columnas
par(mfrow = c(1, 2))

# Q-Q plot para la variable speed
qqnorm(cars$speed, main = "Q-Q Plot - Velocidad")
qqline(cars$speed, col = "blue")

# Histograma de la variable de velocidad (speed) con curva normal superpuesta
hist(cars$speed, probability = TRUE, main = "Histograma - Velocidad", xlab = "Velocidad")
lines(density(cars$speed), col = "blue")
curve(dnorm(x, mean = mean(cars$speed), sd = sd(cars$speed)), add = TRUE, col = "red")
```

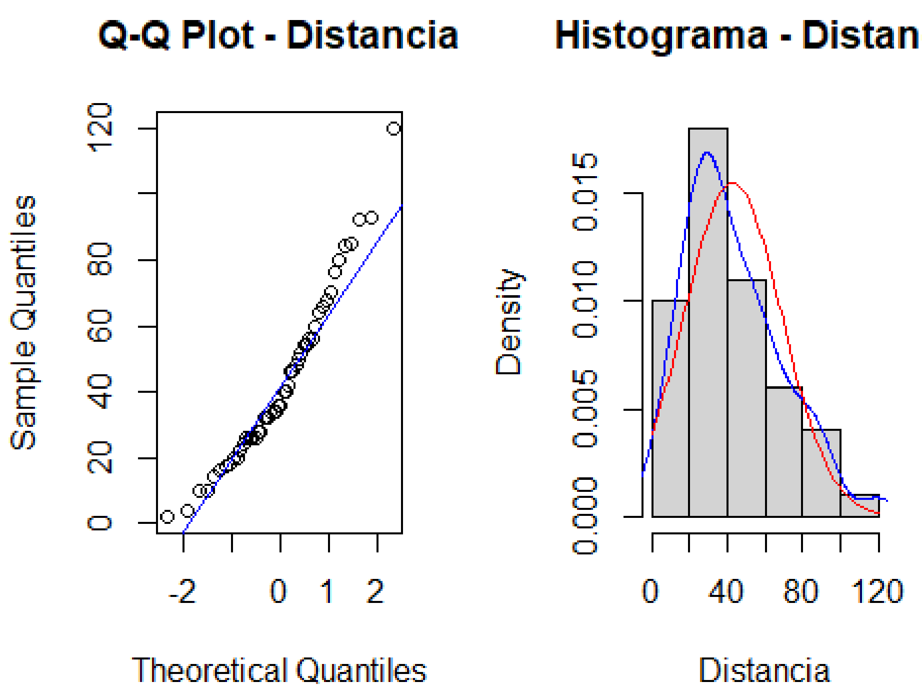



```
# Restablecer la configuración de la ventana gráfica (opcional)
par(mfrow = c(1, 1))
```

Interpretación

- Normalidad: La variable Velocidad parece aproximarse bastante bien a una distribución normal, aunque con algunas desviaciones en los extremos. Los resultados son coherentes con las pruebas de normalidad.
- Desviación en los extremos: Las desviaciones observadas en los extremos tanto en el Q-Q plot como en el histograma no son drásticas, pero pueden influir en análisis más avanzados, especialmente si se desea un ajuste estricto a la normalidad.

```
par(mfrow=c(1, 2)) # Gráficos en una sola fila

# Q-Q plot para la variable dist
qqnorm(cars$dist, main = "Q-Q Plot - Distancia")
qqline(cars$dist, col = "blue")

# Histograma de la variable de distancia (dist) con curva normal
superpuesta
hist(cars$dist, probability = TRUE, main = "Histograma - Distancia", xlab
= "Distancia")
lines(density(cars$dist), col = "blue")
curve(dnorm(x, mean = mean(cars$dist), sd = sd(cars$dist)), add = TRUE,
col = "red")
```



Interpretación

- Normalidad: La variable Distancia no sigue una distribución completamente normal. Aunque la parte central de la distribución está relativamente bien alineada con una curva normal. Los valores extremos y la asimetría positiva indican que la distribución se desvía de la normalidad, especialmente en las colas.

Para lograr un ajuste más cercano a la normalidad, se podría considerar transformar los datos utilizando una transformación logarítmica, sqrt o de Box-Cox para corregir la asimetría y las colas largas.

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

```
library(e1071)

## Warning: package 'e1071' was built under R version 4.3.3

# Calcular el coeficiente de sesgo (skewness) para la variable velocidad
(speed)
sesgo_speed <- skewness(cars$speed)

# Calcular el coeficiente de sesgo (skewness) para la variable distancia
(dist)
sesgo_dist <- skewness(cars$dist)

# Calcular el coeficiente de curtosis (kurtosis) para la variable
velocidad (speed)
curtosis_speed <- kurtosis(cars$speed)

# Calcular el coeficiente de curtosis (kurtosis) para la variable
distancia (dist)
curtosis_dist <- kurtosis(cars$dist)

# Crear una tabla con los resultados de sesgo y curtosis
resultados_sesgo_curtosis <- data.frame(
  Variable = c("Velocidad", "Distancia"),
  Sesgo = c(sesgo_speed, sesgo_dist),
  Curtosis = c(curtosis_speed, curtosis_dist)
)

resultados_sesgo_curtosis

##    Variable      Sesgo   Curtosis
## 1 Velocidad -0.1105533 -0.6730924
## 2 Distancia  0.7591268  0.1193971
```

Observaciones

- Velocidad tiene una distribución que es casi simétrica y ligeramente dispersa, lo que la hace bastante cercana a una distribución normal.

- Distancia muestra una asimetría positiva moderada (sesgo hacia la derecha) y una forma que es cercana a una distribución normal en términos de curtosis, aunque con valores extremos a la derecha.

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

- La variable **Velocidad** puede considerarse normal para la mayoría de los propósitos estadísticos. Las ligeras desviaciones en los extremos no afectan significativamente el análisis global, por lo que no es necesario realizar ninguna transformación para corregir la normalidad. Por otro lado,la variable **Distancia** no sigue una distribución normal, principalmente debido a la asimetría positiva y los valores extremos en las distancias más altas. Estos datos sugieren que existen factores que están haciendo que las distancias más grandes se comporten de manera diferente a lo esperado en una distribución normal.

- Posibles motivos de alejamiento de la normalidad: **Valores extremos (outliers):** En la variable Distancia, es probable que existan algunos valores extremos que correspondan a automóviles con características muy diferentes. Estos valores afectan la normalidad, especialmente en las colas de la distribución.

Heterogeneidad en los datos: Los datos de frenado pueden haber sido recogidos bajo diferentes condiciones, lo que introduce variabilidad no controlada que puede alejar los datos de una distribución normal.

Relación no lineal: Podría existir una relación no lineal entre la velocidad y la distancia de frenado que no ha sido capturada en este análisis inicial, lo que afectaría la normalidad de los residuos y de los datos mismos.

Parte 2: Regresión lineal

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

```r
# Realizar la regresión lineal simple (y = distancia, x = velocidad)
modelo_lineal <- lm(dist ~ speed, data = cars)

# Mostrar el resumen del modelo
summary(modelo_lineal)

##
## Call:
## lm(formula = dist ~ speed, data = cars)
```

```
##
## 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 *
## 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
```

```
# Obtener los coeficientes del modelo lineal
coeficientes <- coef(modelo_lineal)
coeficientes
```

```
## (Intercept)       speed
##  -17.579095    3.932409
```

Observaciones * El modelo de regresión lineal simple muestra que la velocidad tiene un impacto significativo en la distancia de frenado. Aproximadamente el 65.11% de la variabilidad en la distancia de frenado puede explicarse por la velocidad.

- El coeficiente de la velocidad es altamente significativo, lo que refuerza la relación positiva entre estas dos variables. Sin embargo, el error estándar residual de 15.38 sugiere que hay cierta variabilidad en los datos que el modelo no captura, lo que podría indicar la presencia de otros factores que afectan la distancia de frenado.

Escribe el modelo lineal obtenido.

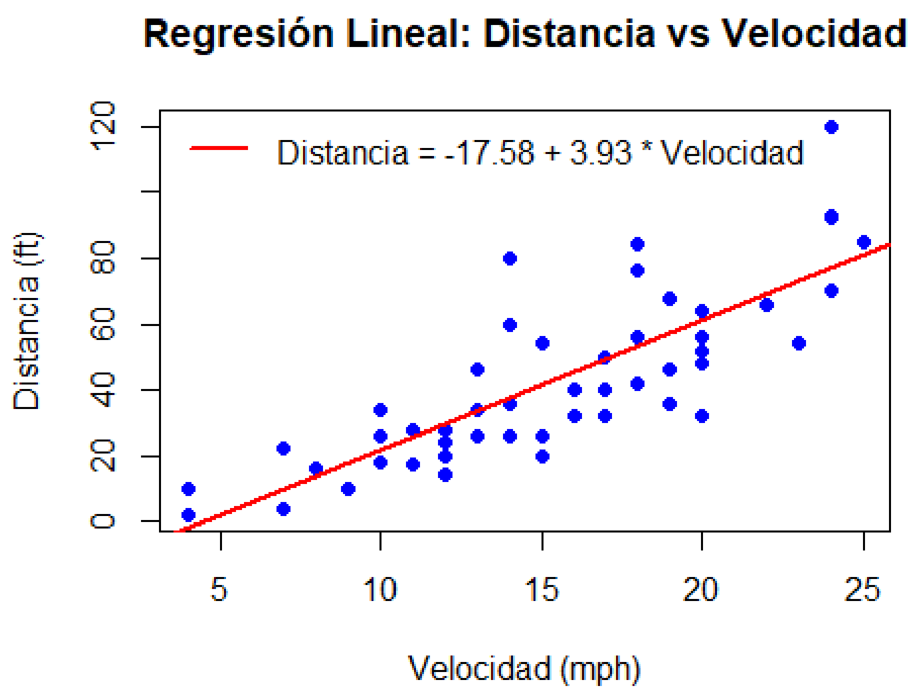
```
# Escribir el modelo lineal obtenido
cat("El modelo lineal obtenido es: Distancia = ", round(coeficientes[1],
2),
    "+", round(coeficientes[2], 2), "* Velocidad\n")
```

```
## El modelo lineal obtenido es: Distancia =  -17.58 + 3.93 * Velocidad
```

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

```
# Crear la gráfica
plot(cars$speed, cars$dist,
     main = "Regresión Lineal: Distancia vs Velocidad",
     xlab = "Velocidad (mph)",
     ylab = "Distancia (ft)",
     pch = 16, col = "blue")

# Agregar la línea del modelo a la gráfica
```

```r
abline(modelo_lineal, col = "red", lwd = 2)

# Mostrar la ecuación del modelo en la gráfica
legend("topleft", legend = paste("Distancia =", round(coeficientes[1], 2),
                                  "+", round(coeficientes[2], 2), "* Velocidad"),
       col = "red", lwd = 2, bty = "n")
```



Interpretación

- Este modelo muestra que existe una relación lineal positiva entre la velocidad y la distancia de frenado. Los autos que van a mayor velocidad requieren más distancia para detenerse, y el incremento es de aproximadamente 3.93 pies por cada milla por hora de incremento en la velocidad.

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

De acuerdo con lo resultados obtenidos con anterioridad:

Individualmente, la velocidad es un predictor altamente significativo. **Conjuntamente**, el modelo es significativo, y la relación entre la velocidad y la distancia de frenado es estadísticamente importante. El R^2 muestra que la velocidad explica un porcentaje considerable de la variabilidad en la distancia de frenado, lo que hace que el modelo sea adecuado para predecir la distancia en función de la velocidad.

Analiza validez del modelo.

Para todas las hipótesis se utiliza un alpha de 0.05.

Residuos con media cero

$$H_0\text{:La media de los residuos es cero } (\mu = 0)$$

$$H_1\text{:La media de los residuos no es cero } (\mu \neq 0)$$

```r
# Obtener los residuos del modelo
residuos <- residuals(modelo_lineal)

# Prueba t para la media de los residuos
t_test_residuos <- t.test(residuos, mu = 0)
t_test_residuos

##
##  One Sample t-test
##
## data:  residuos
## t = 1.0315e-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
## 2.220446e-16

# Calcular la media de los residuos
media_residuos <- mean(residuos)
cat("Media de los residuos:", media_residuos, "\n")

## Media de los residuos: 2.220446e-16
```

Interpretación

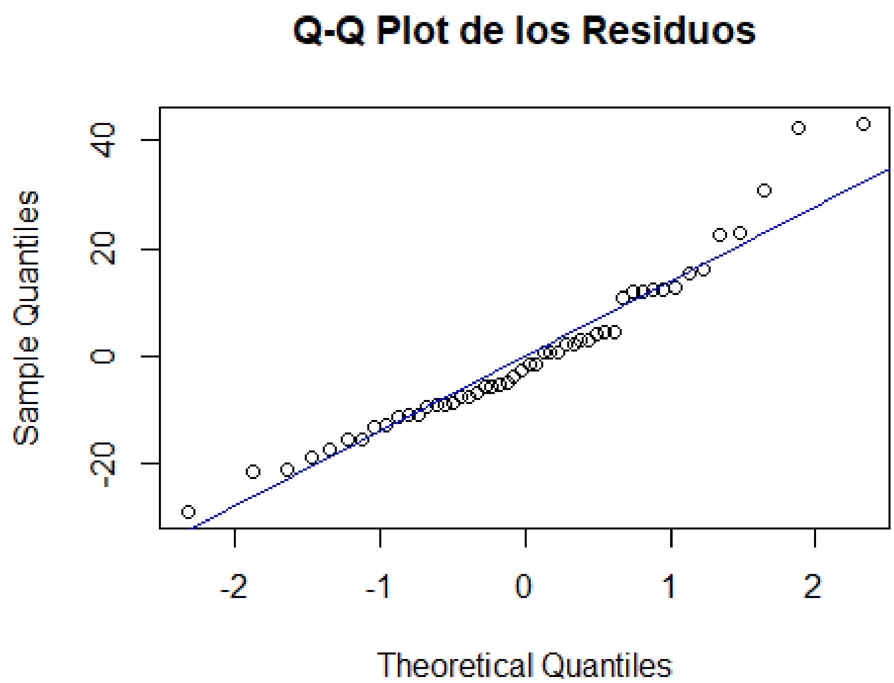
- La prueba t muestra que no podemos rechazar la hipótesis nula de que la media de los residuos es cero. La media de los residuos es prácticamente cero, lo que es un buen indicador de que el modelo está adecuadamente ajustado en este aspecto.

Normalidad de los residuos

$$H_0\text{:Los residuos siguen una distribución normal}$$

$$H_1\text{:Los residuos no siguen una distribución normal}$$

```r
# Gráfico Q-Q para los residuos
qqnorm(residuos, main = "Q-Q Plot de los Residuos")
qqline(residuos, col = "blue")
```




```
# Anderson-Darling Test para los residuos
ad_test_residuos <- ad.test(residuos)
ad_test_residuos

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

Interpretación

- El Q-Q plot muestra que los residuos son cercanos a la normalidad, pero con algunas desviaciones en los extremos. Esto indica que los valores extremos (outliers o colas) podrían no seguir una distribución normal perfecta. A pesar de estas pequeñas desviaciones, el modelo en general parece cumplir con el supuesto de normalidad de los residuos

- Dado que el p-value es menor que $\alpha = 0.05$, rechazamos la hipótesis nula H_0. Esto indica que los residuos no siguen una distribución normal.

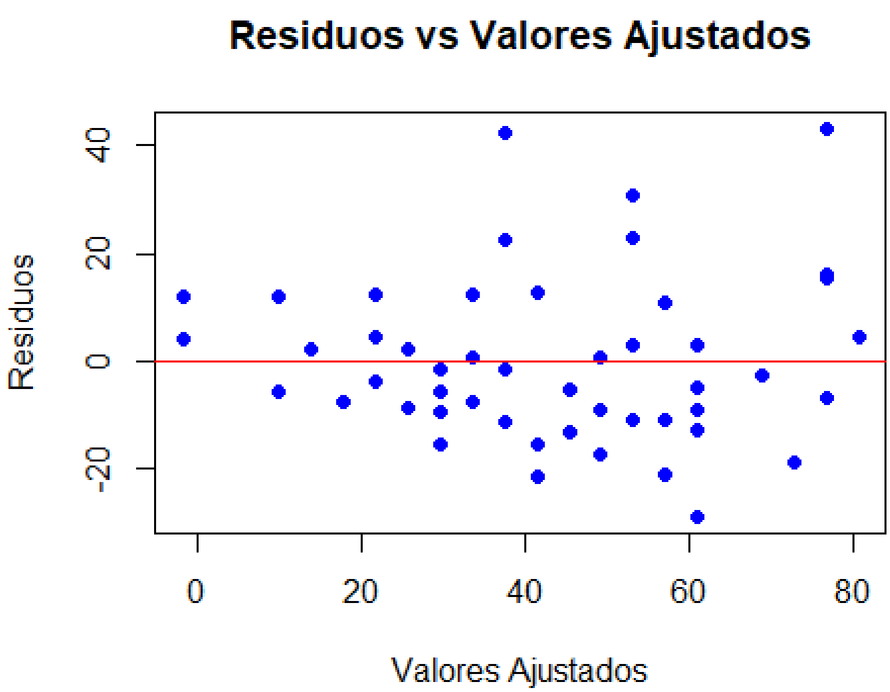
Homocedasticidad, independencia y linealidad.

Homocedasticidad y Linealidad

H_0:Los residuos tienen una varianza constante (homocedasticidad)

H_1:Los residuos no tienen una varianza constante (heterocedasticidad)

```r
# Obtener los valores ajustados y los residuos
fitted_values <- fitted(modelo_lineal)
residuos <- residuals(modelo_lineal)

# Graficar los residuos vs valores ajustados
plot(fitted_values, residuos,
     main = "Residuos vs Valores Ajustados",
     xlab = "Valores Ajustados",
     ylab = "Residuos",
     pch = 16, col = "blue")

# Añadir una línea horizontal en y = 0
abline(h = 0, col = "red")
```



Interpretación * Los residuos parecen dispersarse de manera relativamente aleatoria alrededor de la línea horizontal en $y = 0$, lo que es una buena señal de que no hay un patrón claro de heterocedasticidad visual. Sin embargo, los puntos parecen un poco más dispersos a valores ajustados más altos, por lo que podría haber alguna leve variabilidad en la varianza de los residuos.

```r
library(lmtest)

## Warning: package 'lmtest' was built under R version 4.3.3

## Loading required package: zoo

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

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

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

# Prueba de Breusch-Pagan (homocedasticidad)
bptest(modelo_lineal)

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

# Prueba de Goldfeld-Quandt (homocedasticidad)
gqtest(modelo_lineal)

##
##  Goldfeld-Quandt test
##
## data:  modelo_lineal
## GQ = 1.5512, df1 = 23, df2 = 23, p-value = 0.1498
## alternative hypothesis: variance increases from segment 1 to 2
```

Observaciones

- Para ambas pruebas realizadas, el p-value es mayor que 0.05, no podemos rechazar la hipótesis nula, lo que sugiere que no hay evidencia fuerte de heterocedasticidad en los residuos.

```
# Prueba de Ramsey RESET (linealidad)
resettest(modelo_lineal)

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

Observaciones

- El modelo parece estar correctamente especificado y no muestra problemas de linealidad, de acuerdo con el test RESET.

Independencia

$$H_0\text{:No hay autocorrelación en los residuos}$$

$$H_1\text{:Hay autocorrelación en los residuos}$$

```r
library(lmtest)

# Prueba de Durbin-Watson para verificar independencia de los residuos
dw_test <- dwtest(modelo_lineal)
dw_test
```

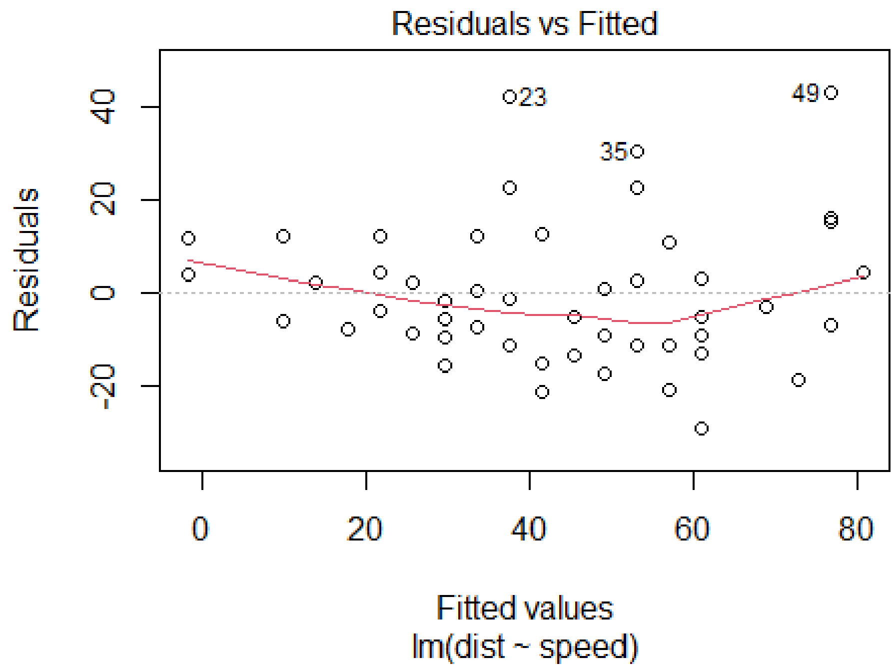
```
##
##  Durbin-Watson test
##
## data:  modelo_lineal
## DW = 1.6762, p-value = 0.09522
## alternative hypothesis: true autocorrelation is greater than 0
```

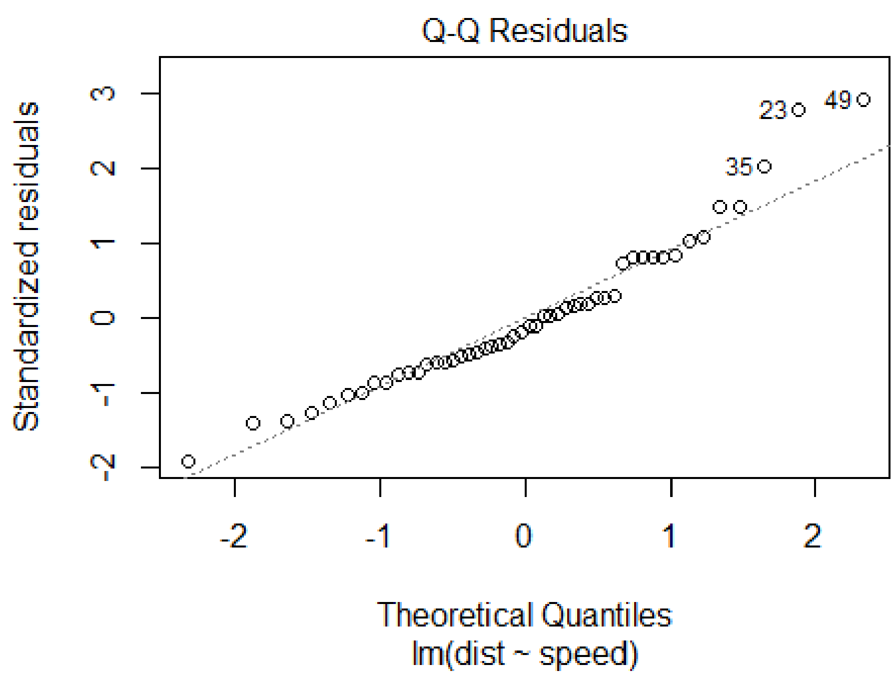
Observación * No se encontró evidencia significativa de autocorrelación positiva en los residuos del modelo. Esto sugiere que el supuesto de independencia de los residuos se mantiene. El valor de DW cerca de 2 también refuerza la idea de que no hay un patrón claro de autocorrelación en los residuos.

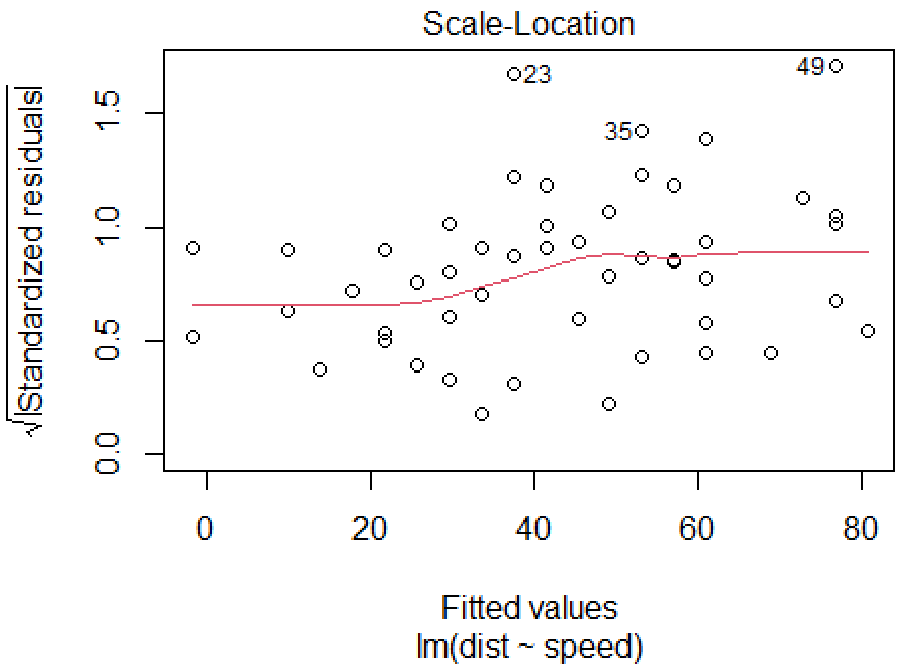
Resultados * El supuesto de que los residuos tienen una media cero se cumple. * El supuesto de normalidad de los residuos no se cumple. * El supuesto de linealidad se cumple. * El supuesto de homocedasticidad se cumple. * El supuesto de independencia de los residuos se cumple.

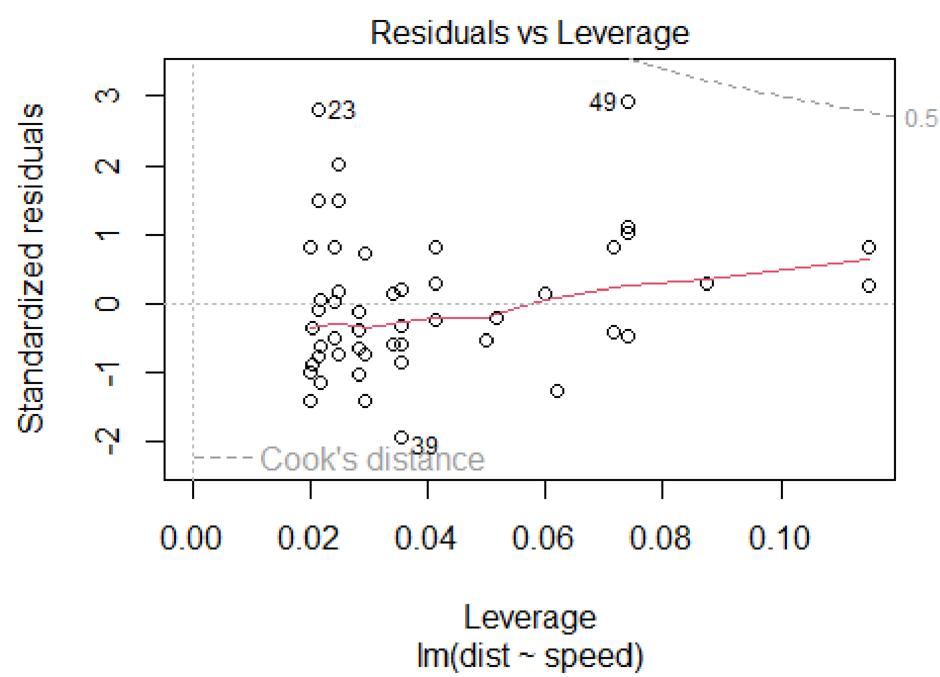
Usa plot(Modelo) para los gráficos y añade pruebas de hipótesis.

```r
# Generar los gráficos de diagnóstico del modelo
plot(modelo_lineal)
```


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


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





* Linealidad:

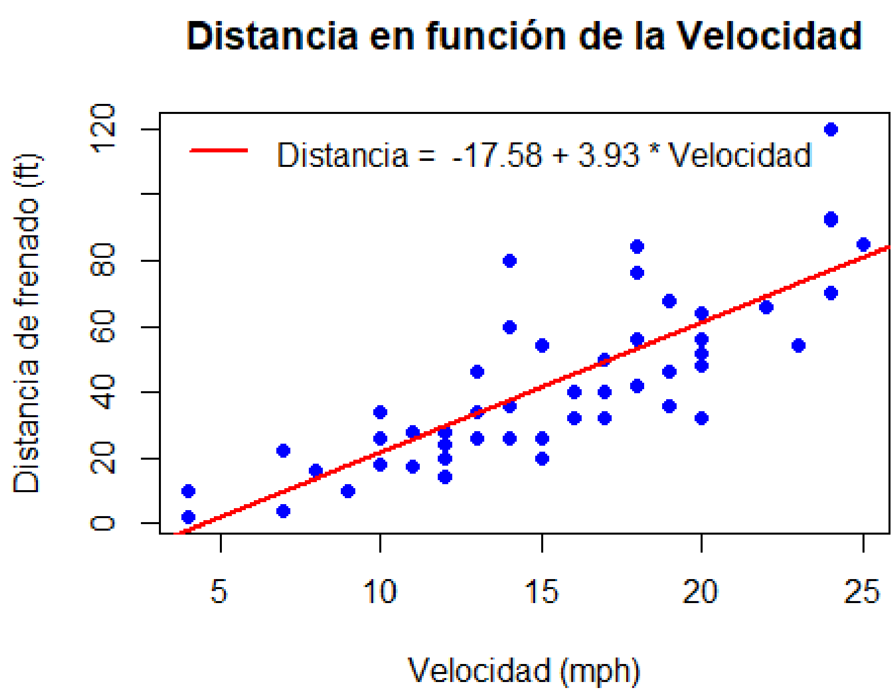
Puede haber una ligera violación de la linealidad, como sugiere la curva suave en el gráfico de residuos vs valores ajustados.

- Normalidad: Los residuos son aproximadamente normales, aunque hay algunas desviaciones en los extremos.

- Homocedasticidad: La varianza parece ser constante en general, aunque hay una ligera tendencia en los valores ajustados más altos.
- Puntos influyentes: Las observaciones 23, 35, 49 y 39 parecen tener un impacto significativo en el modelo y podrían ser puntos atípicos influyentes. Sería prudente investigar estos puntos más a fondo para ver si deberían ser eliminados o si el modelo debe ajustarse para mitigar su influencia.

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

```
# Crear la gráfica de dispersión de los datos
plot(cars$speed, cars$dist,
     main = "Distancia en función de la Velocidad",
     xlab = "Velocidad (mph)",
     ylab = "Distancia de frenado (ft)",
     pch = 16, col = "blue")

# Agregar la línea del modelo de regresión
abline(modelo_lineal, col = "red", lwd = 2)

# Añadir una leyenda con la ecuación del modelo
legend("topleft", legend = paste("Distancia = ",
round(coef(modelo_lineal)[1], 2),
                                 "+", round(coef(modelo_lineal)[2], 2),
"* Velocidad"),
       col = "red", lwd = 2, bty = "n")
```



Observaciones

- Como en interpretaciones anteriores, la relación entre la velocidad y la distancia de frenado es aproximadamente lineal y positiva, lo que significa que al aumentar la velocidad, la distancia de frenado también aumenta. La línea de regresión ajustada refleja esta relación, aunque hay cierta dispersión de los puntos alrededor de la línea.

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

- Significancia del coeficiente de velocidad: El valor p asociado al coeficiente de la velocidad es extremadamente bajo (1.49e-12) lo que indica que la variable velocidad es altamente significativa para explicar la variabilidad en la distancia. Esto significa que hay una relación clara entre ambas variables y que la velocidad contribuye de manera importante a predecir la distancia de frenado.

- F-Statistic: El valor de F=89.57 y su valor p asociado es extremadamente bajo, lo que refuerza la significancia global del modelo y su capacidad para explicar la variabilidad en la distancia.

- Validez del modelo: El $R^2 = 0.6511$ indica que aproximadamente el 65.11% de la variabilidad en la distancia puede explicarse por la velocidad. Si bien esto indica que el modelo tiene una buena capacidad de predicción, aún queda un 35% de la variabilidad sin explicar, lo que sugiere que hay otros factores o variables no incluidas en el modelo que pueden estar influyendo en la distancia de frenado.

- El modelo lineal es estadísticamente significativo y es relativamente simple, lo que lo hace útil para una comprensión inicial de la relación entre la velocidad y la distancia de frenado. Sin embargo, tiene algunas limitaciones en cuanto a su capacidad para capturar todos los aspectos de la relación (como lo sugiere el intercepto negativo y el valor moderado de R^2.

Parte 3: Regresión no lineal

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).

Encuentra el valor de λ 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:

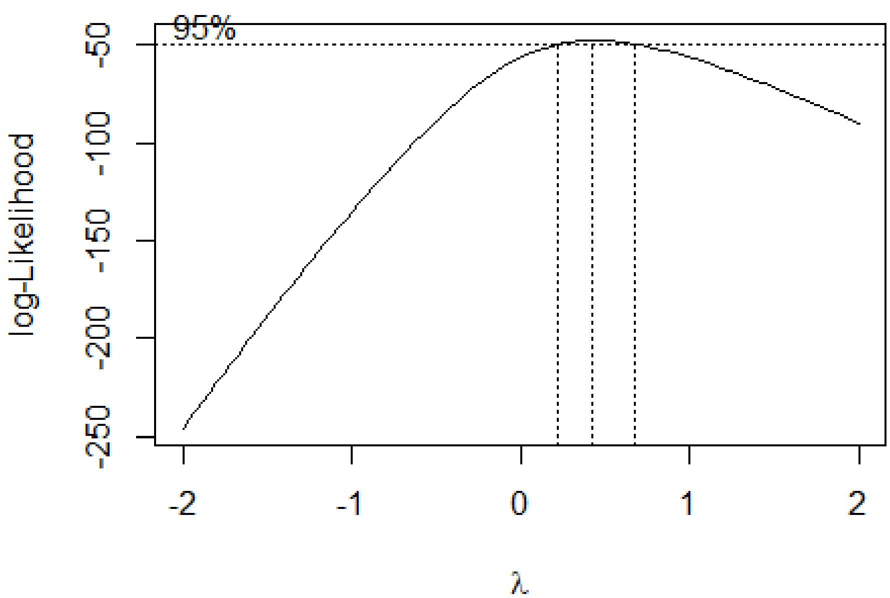
Utiliza: boxcox(lm(Distancia~Velocidad)) si la variable con más alejamiento de normalidad es la distancia

Utiliza: boxcox(lm(Velocidad~Distancia)) si la variable con más alejamiento de normalidad es la velocidad

La transformación se hará sobre la variable que usas como dependiente en el comando lm(y~x)

Dado que la variable Distancia es la que está más alejada de la normalidad, procederemos a utilizar la transformación de Box-Cox sobre esta variable para intentar mejorar la normalidad y ajustar mejor el modelo no lineal.

```
library(MASS)

# Aplicar la transformación Box-Cox para encontrar el valor óptimo de lambda
modelo_boxcox <- boxcox(lm(dist ~ speed, data = cars))
```




```
# Extraer el valor óptimo de lambda
lambda_optimo <- modelo_boxcox$x[which.max(modelo_boxcox$y)]
lambda_optimo

## [1] 0.4242424
```

Observación

- El valor del λ es el exponente que debe aplicarse a la variable Distancia para transformarla en una forma más cercana a una distribución normal. Si este hubiera sido cercano a 1, no se habría necesitado transformación (la variable sería aproximadamente normal).

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.

- El valor óptimo de λ encontrado es **0.424**.
- Definimos las dos transformaciones: **exacta** y **aproximada** para la variable **Distancia**.

Transformación Exacta:

$$Y_{\text{exacta}} = \frac{Y^{\lambda} - 1}{\lambda}$$

Donde: - Y es la variable original (en este caso, **Distancia**), - λ es el parámetro de Box-Cox (en este caso, $\lambda = 0.424$.

La transformación exacta de Box-Cox para la variable **Distancia** sería:

$$Y_{\text{exacta}} = \frac{Y^{0.424} - 1}{0.424}$$

Transformación Aproximada:

$$Y_{\text{aproximada}} \approx \sqrt{(Y)}$$

```
# Transformación exacta de Box-Cox
cars$dist_transformada_exacta <- (cars$dist^lambda_optimo - 1) /
lambda_optimo

# Ver las primeras filas del dataset con las nuevas transformaciones
head(cars)

##   speed dist dist_transformada_exacta
## 1     4    2                 0.805831
## 2     4   10                 3.903635
## 3     7    4                 1.887150
## 4     7   22                 6.390651
## 5     8   16                 5.285168
## 6     9   10                 3.903635

# Ajustar el modelo lineal con la transformación exacta
modelo_exacto <- lm(dist_transformada_exacta ~ speed, data = cars)
summary(modelo_exacto)

##
## Call:
```

```
## lm(formula = dist_transformada_exacta ~ speed, data = cars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.0926 -1.0444 -0.3055  0.7999  4.7520
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.08227    0.73856   1.465    0.149
## speed        0.49541    0.04541  10.910 1.35e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.681 on 48 degrees of freedom
## Multiple R-squared:  0.7126, Adjusted R-squared:  0.7066
## F-statistic:   119 on 1 and 48 DF,  p-value: 1.354e-14
```

Observaciones * El modelo con la transformación Box-Cox ha mejorado el ajuste en comparación con el modelo lineal sin transformación (el R^2 ha pasado de 0.65 a 0.71).

- La transformación ha permitido que el modelo capture mejor la relación entre la velocidad y la distancia, como lo refleja el aumento en R^2.

```r
# Transformación aproximada
cars$dist_transformada_aproximada <- sqrt(cars$dist)

# Ver las primeras filas del dataset con las nuevas transformaciones
head(cars)
```

```
##   speed dist dist_transformada_exacta dist_transformada_aproximada
## 1     4    2                 0.805831                     1.414214
## 2     4   10                 3.903635                     3.162278
## 3     7    4                 1.887150                     2.000000
## 4     7   22                 6.390651                     4.690416
## 5     8   16                 5.285168                     4.000000
## 6     9   10                 3.903635                     3.162278
```

```r
# Ajustar el modelo lineal con la transformación sqrt
modelo_aproximado <- lm(dist_transformada_aproximada ~ speed, data =
cars)
summary(modelo_aproximado)
```

```
##
## Call:
## lm(formula = dist_transformada_aproximada ~ speed, data = cars)
##
## 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 *  
## 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
```

Observaciones * El modelo ajustado utilizando la transformación de raíz cuadrada tiene un buen ajuste a los datos, con un R^2del 70.94%, lo que indica que la velocidad es un predictor fuerte de la distancia.

- Los coeficientes del modelo son significativos, y el modelo en su conjunto es altamente significativo.

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

Compara las medidas: sesgo y curtosis.

```r
# Calcular el sesgo y la curtosis de cada variable: original, exacta y sqrt
sesgo_original <- skewness(cars$dist)
curtosis_original <- kurtosis(cars$dist)

sesgo_exacta <- skewness(cars$dist_transformada_exacta)
curtosis_exacta <- kurtosis(cars$dist_transformada_exacta)

sesgo_sqrt <- skewness(cars$dist_transformada_aproximada)
curtosis_sqrt <- kurtosis(cars$dist_transformada_aproximada)

# Crear una tabla con los resultados
resultados <- data.frame(
  Transformacion = c("Original", "Box-Cox Exacta", "Sqrt"),
  Sesgo = c(sesgo_original, sesgo_exacta, sesgo_sqrt),
  Curtosis = c(curtosis_original, curtosis_exacta, curtosis_sqrt)
)

resultados
```

```
##   Transformacion       Sesgo   Curtosis
## 1       Original  0.75912678  0.1193971
## 2 Box-Cox Exacta -0.17016185 -0.1868840
## 3           Sqrt -0.01902765 -0.3144682
```

Interpretación

- Box-Cox Exacta ha mejorado considerablemente la simetría de la distribución (sesgo cercano a 0) y ha mantenido colas ligeras.

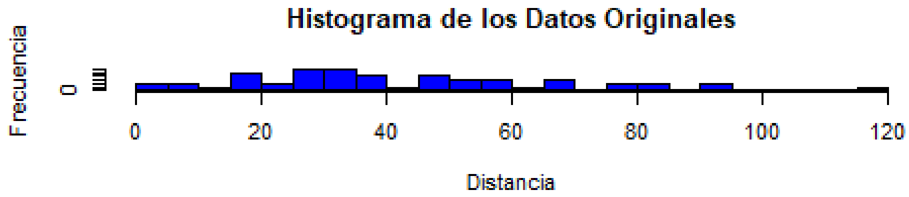
*La transformación Raíz Cuadrada ha logrado un sesgo prácticamente neutro (−0.019), lo que indica que ha tenido éxito en hacer la distribución más simétrica.

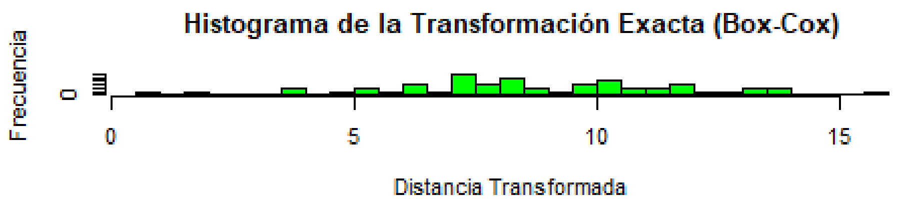
- En términos de curtosis, ambas transformaciones (Box-Cox y Raíz Cuadrada) han logrado reducir la curtosis de los datos originales, acercando la distribución a una forma más normal.

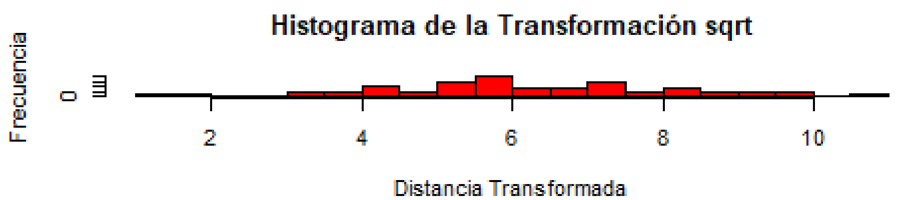
Si el objetivo es minimizar el sesgo, la transformación de Raíz Cuadrada parece ser la mejor opción debido a su sesgo casi nulo. Sin embargo, si se prefiere una distribución más cercana a la normalidad en términos generales, la transformación Box-Cox también es una buena opción.

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

```r
# Crear los histogramas
par(mfrow = c(3, 1))  # Dividir el área gráfica en 3 filas

# Histograma de los datos originales
hist(cars$dist, breaks = 30, col = "blue", main = "Histograma de los Datos Originales",
     xlab = "Distancia", ylab = "Frecuencia")

# Histograma de los datos transformados con Box-Cox
hist(cars$dist_transformada_exacta, breaks = 30, col = "green", main = "Histograma de la Transformación Exacta (Box-Cox)",
     xlab = "Distancia Transformada", ylab = "Frecuencia")

# Histograma de los datos transformados con sqrt
hist(cars$dist_transformada_aproximada, breaks = 30, col = "red", main = "Histograma de la Transformación sqrt",
     xlab = "Distancia Transformada", ylab = "Frecuencia")
```








```r
# Ajustar el espacio entre los gráficos
par(mfrow = c(1, 1))
```

Observaciones * La transformación Box-Cox ha mejorado la simetría de la distribución y ha mantenido una mejor dispersión de los valores en comparación con los datos originales.

- La transformación de raíz cuadrada ha corregido el sesgo casi por completo, logrando una distribución casi simétrica.

- En términos de simetría y normalización de la distribución, ambas transformaciones han mejorado sustancialmente los datos originales, pero la transformación de raíz cuadrada ha producido una distribución con menos sesgo, mientras que la Box-Cox es más equilibrada en términos de simetría y curtosis.

Realiza algunas pruebas de normalidad para los datos transformados.

```r
# Prueba de normalidad Shapiro-Wilk para los datos originales, Box-Cox y sqrt
shapiro_original <- shapiro.test(cars$dist)$p.value
shapiro_boxcox <- shapiro.test(cars$dist_transformada_exacta)$p.value
shapiro_sqrt <- shapiro.test(cars$dist_transformada_aproximada)$p.value

# Prueba Anderson-Darling para los datos originales, Box-Cox y sqrt
ad_original <- ad.test(cars$dist)$p.value
ad_boxcox <- ad.test(cars$dist_transformada_exacta)$p.value
ad_sqrt <- ad.test(cars$dist_transformada_aproximada)$p.value
```

```r
# Prueba Jarque-Bera para los datos originales, Box-Cox y sqrt
jb_original <- jarque.bera.test(cars$dist)$p.value
jb_boxcox <- jarque.bera.test(cars$dist_transformada_exacta)$p.value
jb_sqrt <- jarque.bera.test(cars$dist_transformada_aproximada)$p.value

# Crear una tabla con los resultados
resultados <- data.frame(
  Test = c("Shapiro-Wilk", "Anderson-Darling", "Jarque-Bera"),
  `Original (p-value)` = c(shapiro_original, ad_original, jb_original),
  `Box-Cox (p-value)` = c(shapiro_boxcox, ad_boxcox, jb_boxcox),
  `sqrt (p-value)` = c(shapiro_sqrt, ad_sqrt, jb_sqrt)
)

resultados

##               Test Original..p.value. Box.Cox..p.value. sqrt..p.value.
## 1     Shapiro-Wilk         0.03909968         0.9772686      0.9941205
## 2 Anderson-Darling         0.05021288         0.9717478      0.9731952
## 3      Jarque-Bera         0.07314987         0.8750981      0.9561497
```

Interpretación

- La transformación de Box-Cox es claramente la mejor opción para acercar los datos a una distribución normal, con valores p altos en todas las pruebas.
- Transformación sqrt: La transformación sqrt también mejora la normalidad de los datos de manera sustancial, con valores p incluso mayores que los de la transformación Box-Cox, especialmente en la prueba de Shapiro-Wilk.
- La transformación sqrt parece ofrecer una mejor normalización en los datos según las pruebas aplicadas, con valores p más altos en todos los casos.

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 lambda para tus nuevos datos.

- Este paso se omite, ya que se logro llegar a la normalidad con la transformaciones.

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.

- La transformación de raíz cuadrada (sqrt) parece ser la mejor opción. Ofrece una mejor simetría (sesgo casi nulo), resultados de normalidad ligeramente superiores, y un modelo con menor error estándar residual. Por lo tanto, la transformación de raíz cuadrada es la más adecuada para este conjunto de datos.

- Desde el punto de vista de la economía del modelo, la transformación de raíz cuadrada es preferible debido a su simplicidad y facilidad de interpretación. Es una transformación bien entendida, no requiere ajustes de parámetros adicionales, y permite un análisis más transparente sin comprometer demasiado la calidad del ajuste.

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

```
# Realizar la regresión lineal simple entre la distancia transformada
sqrt y la velocidad
modelo_aproximado <- lm(dist_transformada_aproximada ~ speed, data =
cars)

# Mostrar el resumen del modelo
summary(modelo_aproximado)

##
## Call:
## lm(formula = dist_transformada_aproximada ~ speed, data = cars)
##
## 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 *
## 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
```

Escribe el modelo lineal para la transformación.

```
# Obtener los coeficientes del modelo lineal
coeficientes_aprox <- coef(modelo_aproximado)
coeficientes_aprox

## (Intercept)       speed
##   1.2770502   0.3224125

# Mostrar el modelo lineal obtenido
modelo_aprox <- paste0("Distancia Transformada (sqrt) = ",
```

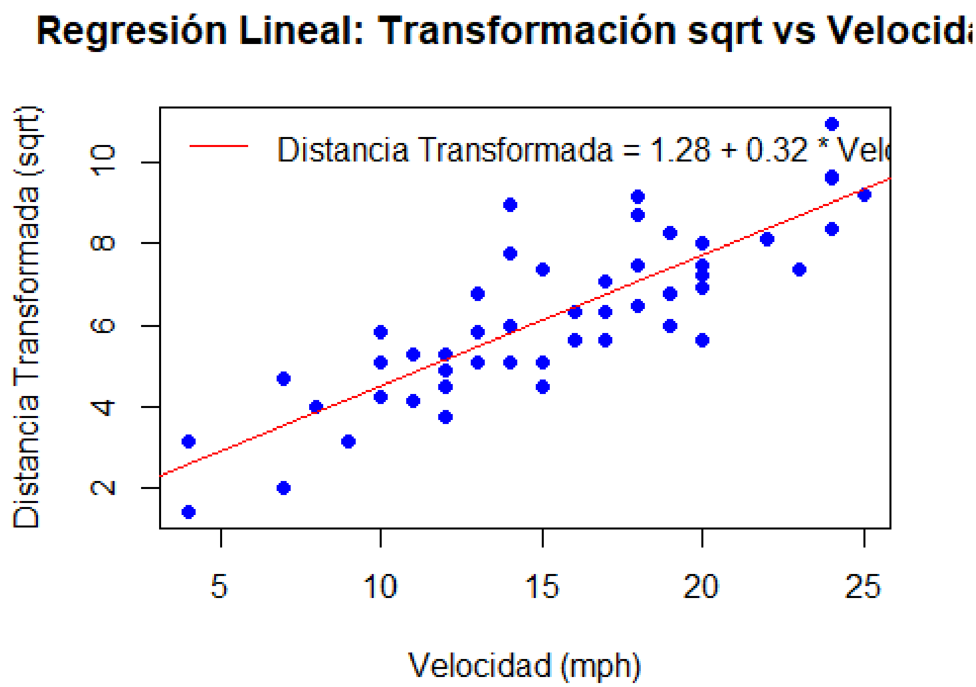
```
round(coeficientes_aprox[1], 4),
                        " + ", round(coeficientes_aprox[2], 4), " *
Velocidad")
cat("Modelo lineal obtenido:\n", modelo_aprox, "\n")
```

```
## Modelo lineal obtenido:
##  Distancia Transformada (sqrt) = 1.2771 + 0.3224 * Velocidad
```

Interpretación *El modelo muestra que la velocidad tiene un impacto significativo en la distancia de frenado transformada, y que esta relación es positiva y casi linealcon la aplicación de una transformación de raíz cuadrada para estabilizar los datos y mejorar la normalidad.

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

```
# Graficar los datos transformados y la línea de regresión
plot(cars$speed, cars$dist_transformada_aproximada, main = "Regresión
Lineal: Transformación sqrt vs Velocidad",
     xlab = "Velocidad (mph)", ylab = "Distancia Transformada (sqrt)",
pch = 16, col = "blue")

# Añadir la línea de regresión
abline(modelo_aproximado, col = "red")

# Añadir la ecuación del modelo en el gráfico
legend("topleft", legend = paste0("Distancia Transformada = ",
round(coeficientes_aprox[1], 2), " + ",
                                  round(coeficientes_aprox[2], 2), " *
Velocidad"),
       col = "red", lty = 1, bty = "n")
```



Interpretación * La gráfica sugiere que existe una relación lineal clara entre la velocidad y la distancia transformada por raíz cuadrada. El modelo lineal es un ajuste adecuado, y aunque hay algo de variabilidad en los datos, la mayoría de los puntos siguen la tendencia marcada por la línea de regresión.

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

- Significancia individual: Tanto el intercepto como la velocidad son significativos, con la velocidad siendo el factor más importante.
- Significancia conjunta: El modelo en su conjunto es altamente significativo, con un F-estadístico muy alto y un p-valor extremadamente pequeño.
- Coeficiente de correlación: Existe una correlación fuerte entre la velocidad y la distancia transformada.

En general, el modelo es confiable y muestra una relación clara entre la velocidad y la distancia transformada.

Analiza validez del modelo: normalidad de los residuos, homocedasticidad e independencia.

Para todas las hipótesis se utiliza un alpha de 0.05.

Residuos con media cero

$$H_0\text{:La media de los residuos es cero } (\mu = 0)$$

$$H_1\text{:La media de los residuos no es cero } (\mu \neq 0)$$

```r
# Obtener los residuos del modelo
residuos_aprox <- residuals(modelo_aproximado)

# Prueba t para la media de los residuos
t_test_residuos <- t.test(residuos_aprox, mu = 0)
t_test_residuos

##
##  One Sample t-test
##
## data:  residuos_aprox
## t = 3.9571e-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
## 6.105969e-17

# Calcular la media de los residuos
media_residuos <- mean(residuos_aprox)
cat("Media de los residuos:", media_residuos, "\n")

## Media de los residuos: 6.105969e-17
```

Interpretación

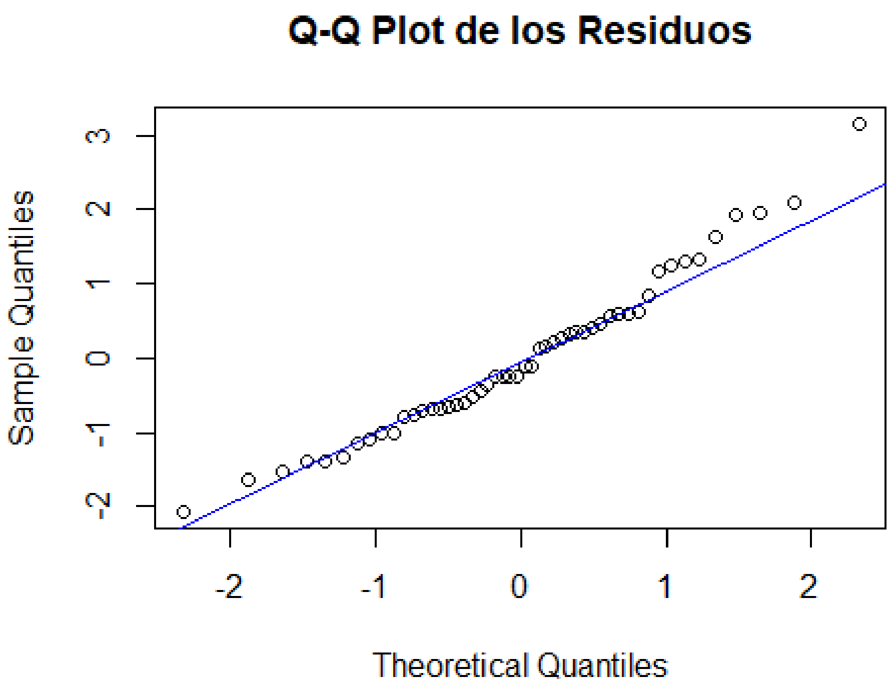
- La prueba confirma que la media de los residuos del modelo es prácticamente cero, lo que es un buen indicador de que el modelo es adecuado y no presenta sesgos significativos en los residuos

Normalidad de los residuos

$$H_0\text{:Los residuos siguen una distribución normal}$$

$$H_1\text{:Los residuos no siguen una distribución normal}$$

```r
# Gráfico Q-Q para los residuos
qqnorm(residuos_aprox, main = "Q-Q Plot de los Residuos")
qqline(residuos_aprox, col = "blue")
```




```r
# Anderson-Darling Test para los residuos
ad_test_residuos <- ad.test(residuos_aprox)
ad_test_residuos

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

Interpretación

- El p-value de 0.3551 indica que no hay suficiente evidencia para rechazar la hipótesis nula de que los residuos siguen una distribución normal. Esto sugiere que los residuos son normalmente distribuidos de acuerdo con este test.

Homocedasticidad, independencia y linealidad.

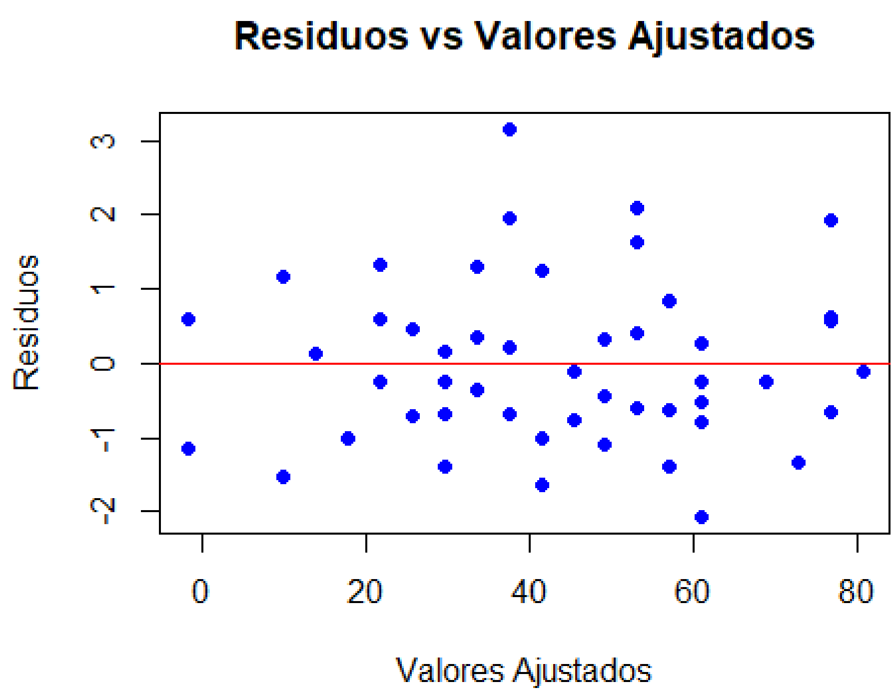
Homocedasticidad y Linealidad

H_0:Los residuos tienen una varianza constante (homocedasticidad)

H_1:Los residuos no tienen una varianza constante (heterocedasticidad)

```r
# Graficar los residuos vs valores ajustados
plot(fitted_values, residuos_aprox,
     main = "Residuos vs Valores Ajustados",
     xlab = "Valores Ajustados",
```

```r
     ylab = "Residuos",
     pch = 16, col = "blue")

# Añadir una línea horizontal en y = 0
abline(h = 0, col = "red")
```



Interpretación

- El gráfico sugiere que las suposiciones de linealidad y homocedasticidad se cumplen razonablemente bien en el modelo ajustado, lo que apoya la validez del modelo en cuanto a estas suposiciones.

```r
# Prueba de Breusch-Pagan (homocedasticidad)
bptest(modelo_aproximado)

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

# Prueba de Goldfeld-Quandt (homocedasticidad)
gqtest(modelo_aproximado)

##
##  Goldfeld-Quandt test
##
## data:  modelo_aproximado
```

```
## GQ = 0.83457, df1 = 23, df2 = 23, p-value = 0.6659
## alternative hypothesis: variance increases from segment 1 to 2
```

Observaciones

- Ambas pruebas, la de Breusch-Pagan y la de Goldfeld-Quandt, sugieren que no hay problemas de heterocedasticidad en el modelo. Esto implica que la varianza de los residuos es constante y que el modelo cumple adecuadamente con este supuesto.

```r
# Prueba de Ramsey RESET (linealidad)
resettest(modelo_aproximado)
```

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

Observaciones

- El valor p es considerablemente mayor que $\alpha = 0.05$, lo que significa que no se rechaza la hipótesis nula. No hay evidencia de errores de especificación en el modelo.

Independencia

$$H_0\text{:No hay autocorrelación en los residuos}$$

$$H_1\text{:Hay autocorrelación en los residuos}$$

```r
library(lmtest)

# Prueba de Durbin-Watson para verificar independencia de los residuos
dw_test <- dwtest(modelo_aproximado)
dw_test
```

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

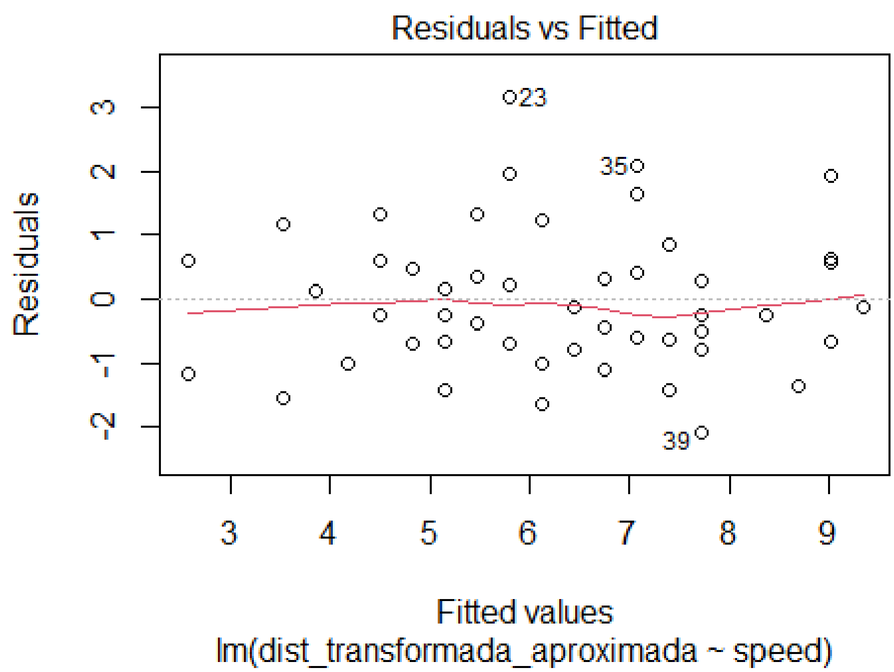
Observación

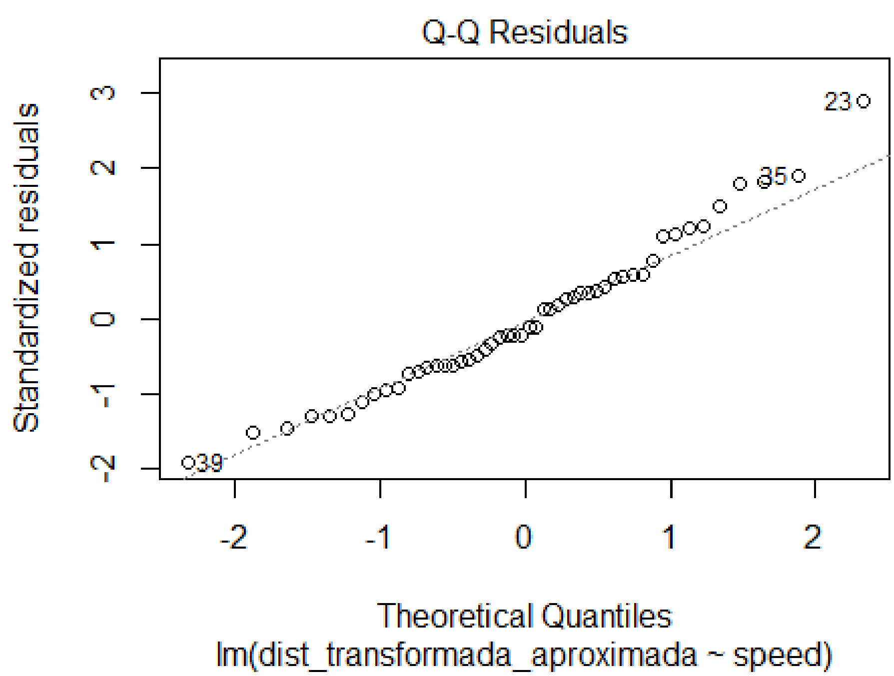
- No se encontró evidencia de autocorrelación en los residuos, lo que indica que los residuos son independientes y el modelo es adecuado en este aspecto.

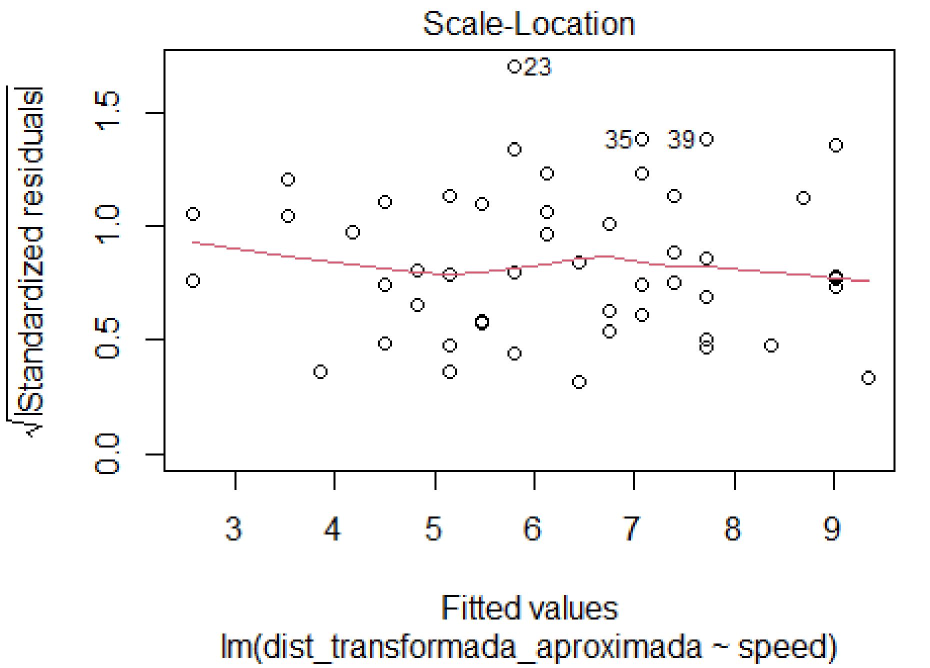
Resultados * Se cumplen todos los supuestos.

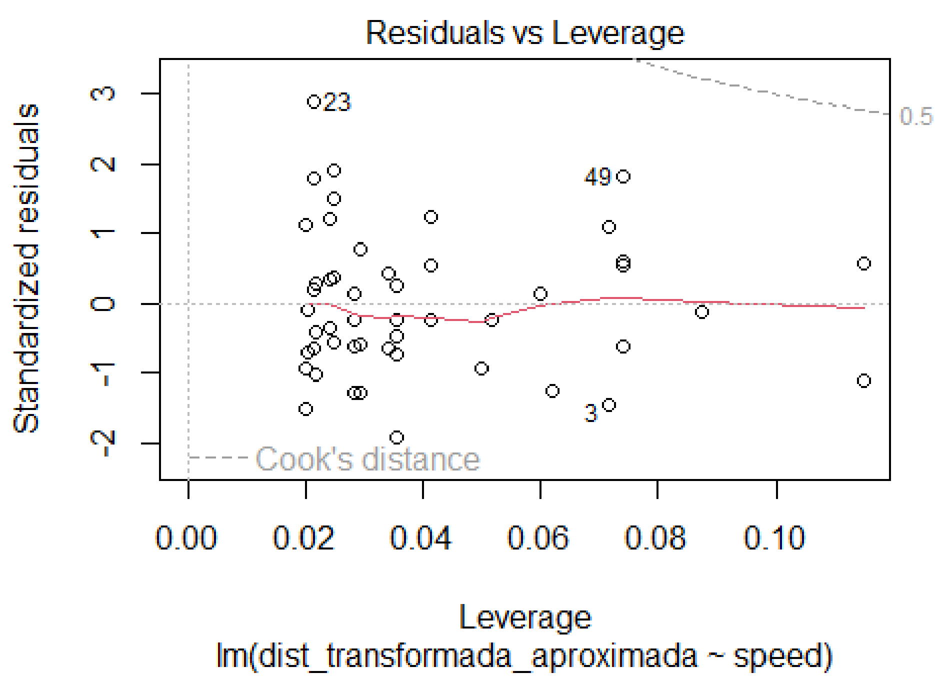
Usa plot(Modelo) para los gráficos y añade pruebas de hipótesis.

```r
# Generar los gráficos de diagnóstico del modelo
plot(modelo_aproximado)
```









Interpretación

- Los gráficos de diagnóstico respaldan que el modelo ajustado cumple con los supuestos de la regresión lineal.

- El gráfico Q-Q muestra que la mayoría de los puntos siguen la línea teórica, con algunas desviaciones menores en los extremos

- El gráfico de residuos vs valores ajustados no muestra un patrón claro y los residuos parecen distribuirse aleatoriamente alrededor de 0.

- El gráfico de residuos vs valores ajustados no presenta una tendencia no lineal clara, lo que sugiere que la relación entre las variables es lineal.

Indica si hay candidatos a datos atípicos o influyentes en la regresión.

- En el gráfico de **Residuos vs Leverage**, se observan algunos puntos etiquetados como **23** y **49** que se destacan de los demás, lo que indica que podrían tener un mayor **apalancamiento** (leverage). Estos puntos se encuentran más alejados del centro, al igual que el **3** lo que los convierte en **posibles candidatos a ser datos atípicos o influyentes**.

- En el gráfico de **Gráfico de Q-Q Plot**, el punto 23 parece estar alejado de la línea de los cuartiles teóricos, lo que sugiere que este valor podría no ajustarse bien a la distribución normal de los residuos. El punto 35 también está un poco alejado, pero menos pronunciado.

Si bien estos puntos podrían ser **datos atípicos o influyentes**, no parece que afecten significativamente el modelo en términos de ajuste general.

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).

Para despejar la distancia del modelo lineal obtenido, consideramos que la transformación aplicada a la distancia fue una transformación raíz cuadrada

$$\sqrt{\text{Distancia}}.$$

El modelo lineal obtenido para la distancia transformada (raíz cuadrada) fue:

$$\text{Distancia Transformada} = 1.28 + 0.32 \cdot \text{Velocidad}$$

Ahora, para obtener la distancia sin la transformación, aplicamos el inverso de la transformación, es decir, elevamos ambos lados al cuadrado.

$$\sqrt{\text{Distancia}} = 1.28 + 0.32 \cdot \text{Velocidad}$$

Elevamos ambos lados al cuadrado para despejar la distancia:

$$\text{Distancia} = (1.28 + 0.32 \cdot \text{Velocidad})^2$$

Este es el modelo no lineal que relaciona la distancia directamente con la velocidad.

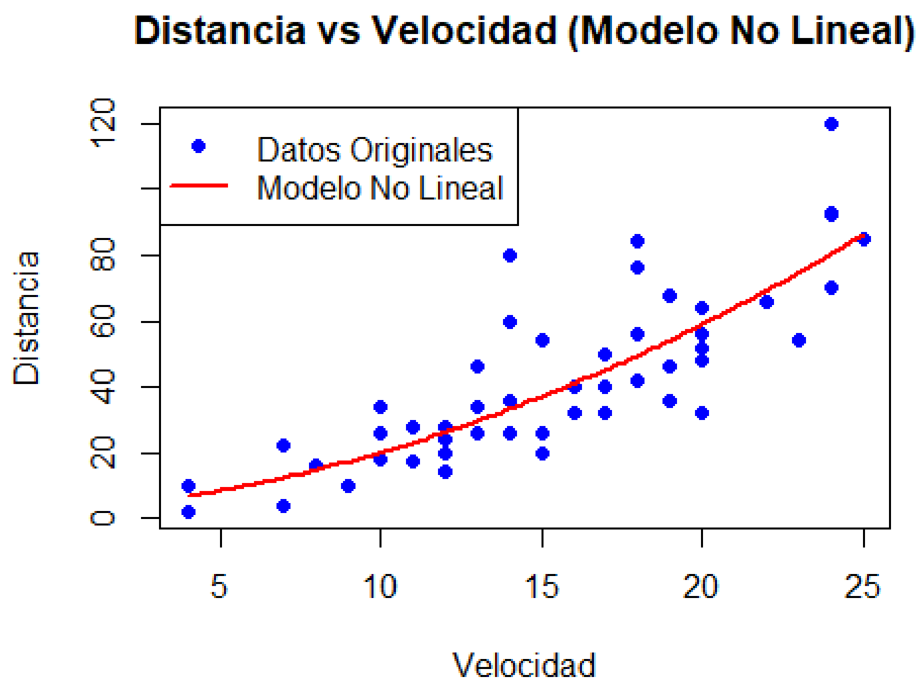
Por lo tanto, la **ecuación no lineal** es:

$$\text{Distancia} = (1.28 + 0.32 \cdot \text{Velocidad})^2$$

* Este modelo captura mejor la relación no lineal que podría existir entre la velocidad y la distancia, reflejando la tendencia acelerada de las distancias a medida que aumenta la velocidad.

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

```r
# Datos originales
velocidad <- cars$speed
distancia <- cars$dist

# Función del modelo no lineal obtenido
modelo_no_lineal <- function(velocidad) {
  (1.28 + 0.32 * velocidad)^2
}

# Crear una secuencia de valores de velocidad para graficar la curva del
modelo
velocidad_modelo <- seq(min(velocidad), max(velocidad), length.out = 100)
distancia_modelo <- modelo_no_lineal(velocidad_modelo)

# Graficar los datos originales
plot(velocidad, distancia, main = "Distancia vs Velocidad (Modelo No
Lineal)",
     xlab = "Velocidad", ylab = "Distancia", pch = 16, col = "blue")

# Graficar la curva del modelo no lineal
lines(velocidad_modelo, distancia_modelo, col = "red", lwd = 2)

# Añadir una leyenda
legend("topleft", legend = c("Datos Originales", "Modelo No Lineal"),
       col = c("blue", "red"), pch = c(16, NA), lty = c(NA, 1), lwd =
c(NA, 2))
```



Interpretación * La gráfica muestra la relación entre la velocidad y la distancia de los datos originales del conjunto de datos cars. El modelo no lineal parece describir adecuadamente la relación entre velocidad y distancia, capturando la aceleración de la distancia a medida que la velocidad aumenta.

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

- La idoneidad del modelo entre la transformación de la distancia (raíz cuadrada) y la velocidad:

- El modelo es adecuado en términos de significancia estadística, bondad de ajuste y validez de los supuestos, pues las pruebas en el análisis confirman que no hay violaciones importantes de los supuestos, lo que refuerza la validez del modelo. Además, la transformación aplicada (raíz cuadrada) ha ayudado a mejorar la linealidad entre la velocidad y la distancia transformada, permitiendo una mejor interpretación de los resultados. Por lo tanto, el modelo es idóneo para explicar la relación entre velocidad y distancia, y es válido desde el punto de vista estadístico.

Parte 4: Conclusión

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.

- El modelo del Punto 2 es claramente el mejor para describir la relación entre la distancia y la velocidad.

- Mejor ajuste: El R^2 del modelo del Punto 2 es superior, lo que indica que explica mejor la variabilidad en la distancia.
- Menor error: El error estándar residual es mucho menor en el modelo del Punto 2, lo que sugiere predicciones más precisas.
- Cumplimiento de los supuestos: Los diagnósticos muestran que el modelo con la distancia transformada (raíz cuadrada) cumple mejor con los supuestos de la regresión lineal.
- Validez del modelo: El modelo transformado permite capturar mejor la relación no lineal subyacente entre la velocidad y la distancia.

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

- Aunque el modelo con la transformación raíz cuadrada de la distancia es el mejor en términos de ajuste y cumplimiento de los supuestos, también tiene algunos posibles problemas que deben considerarse:

Datos atípicos e influyentes:

En los gráficos de residuos vs valores ajustados y el gráfico de leverage, se pueden observar algunos puntos que podrían ser datos influyentes o atípicos. En particular, los puntos 23, 35 y 39 han sido identificados en los gráficos como posibles candidatos. Estos datos podrían estar influyendo en los coeficientes de regresión. Estos puntos deben investigarse más a fondo con métricas como la distancia de Cook, ya que si tienen un alto leverage y gran distancia de Cook, podrían estar afectando negativamente al modelo.

Complejidad adicional:

El modelo no lineal derivado de la transformación agrega una capa de complejidad en el cálculo y validación del modelo.

Supuestos:

Aunque la transformación mejora los supuestos, sigue habiendo ligeras preocupaciones con la normalidad de los residuos y la posibilidad de heterocedasticidad, ya que siempre es posible que algunos residuos no sigan perfectamente la normalidad, como se observó en algunos gráficos Q-Q, donde se detectan ligeros desvíos en los extremos, así como las variaciones en los residuos parecen estar más dispersas a mayores valores ajustados, lo que podría indicar una leve heterocedasticidad.