Data cleaned:

library(dplyr)

library(ggplot2)

# Remove rows with a year greater than 2024

ford <- ford %>%

filter(year <= 2024)

EDA:

# Histogram of Prices

ggplot(ford, aes(x = price)) +

geom\_histogram(bins = 30, fill = "blue", color = "black") +

ggtitle("Distribution of Prices") +

xlab("Price") +

ylab("Frequency")

# Histogram of Mileage

ggplot(ford, aes(x = mileage)) +

geom\_histogram(bins = 30, fill = "green", color = "black") +

ggtitle("Distribution of Mileage") +

xlab("Mileage") +

ylab("Frequency")

# Histogram of Vehicle Age

ford <- ford %>% mutate(age = 2024 - year) # Calculating age of the vehicles

ggplot(ford, aes(x = age)) +

geom\_histogram(bins = 30, fill = "red", color = "black") +

ggtitle("Distribution of Vehicle Age") +

xlab("Age (years)") +

ylab("Frequency")

# Scatter Plot: Price vs. Mileage

ggplot(ford, aes(x = mileage, y = price)) +

geom\_point(alpha = 0.5) +

geom\_smooth(method = "lm", se = FALSE, color = "blue") +

ggtitle("Price vs. Mileage") +

xlab("Mileage") +

ylab("Price")

# Scatter Plot: Price vs. Age

ggplot(ford, aes(x = age, y = price)) +

geom\_point(alpha = 0.5) +

geom\_smooth(method = "lm", se = FALSE, color = "red") +

ggtitle("Price vs. Age") +

xlab("Age (years)") +

ylab("Price")

Add age column:

current\_year <- 2024

ford\_clean$age <- current\_year - ford\_clean$year

library(MASS)

# Determine best lambda for Box-Cox Transformation

bc <- boxcox(price ~ mileage + age, data = ford\_clean, lambda = seq(-2, 2, by = 0.1))

best\_lambda <- bc$x[which.max(bc$y)]

# Transform the price

ford\_clean$price\_transformed <- (ford\_clean$price^best\_lambda - 1) / best\_lambda

library(glmnet)

> x <- model.matrix(~ mileage + age, data = ford\_clean)

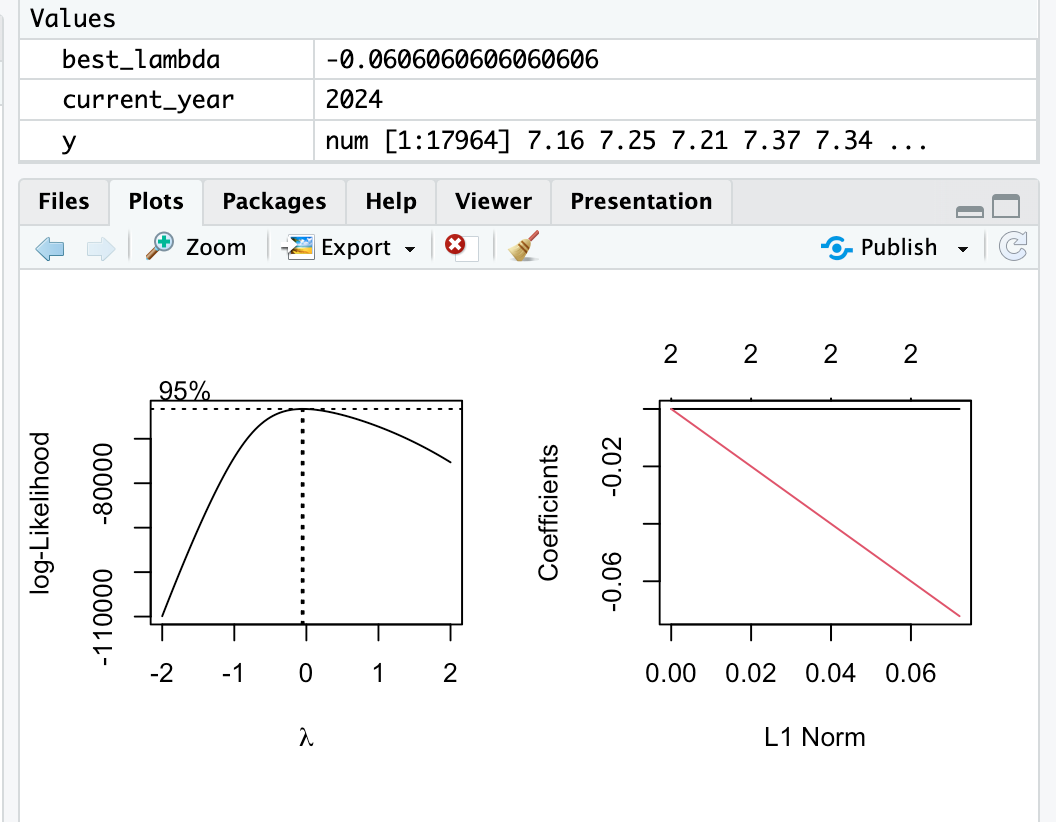
> y <- ford\_clean$price\_transformed

>

> # Fit Ridge regression model

> ridge\_model <- glmnet(x, y, alpha = 0, lambda = 10^seq(4, -2, length = 100))

> plot(ridge\_model)

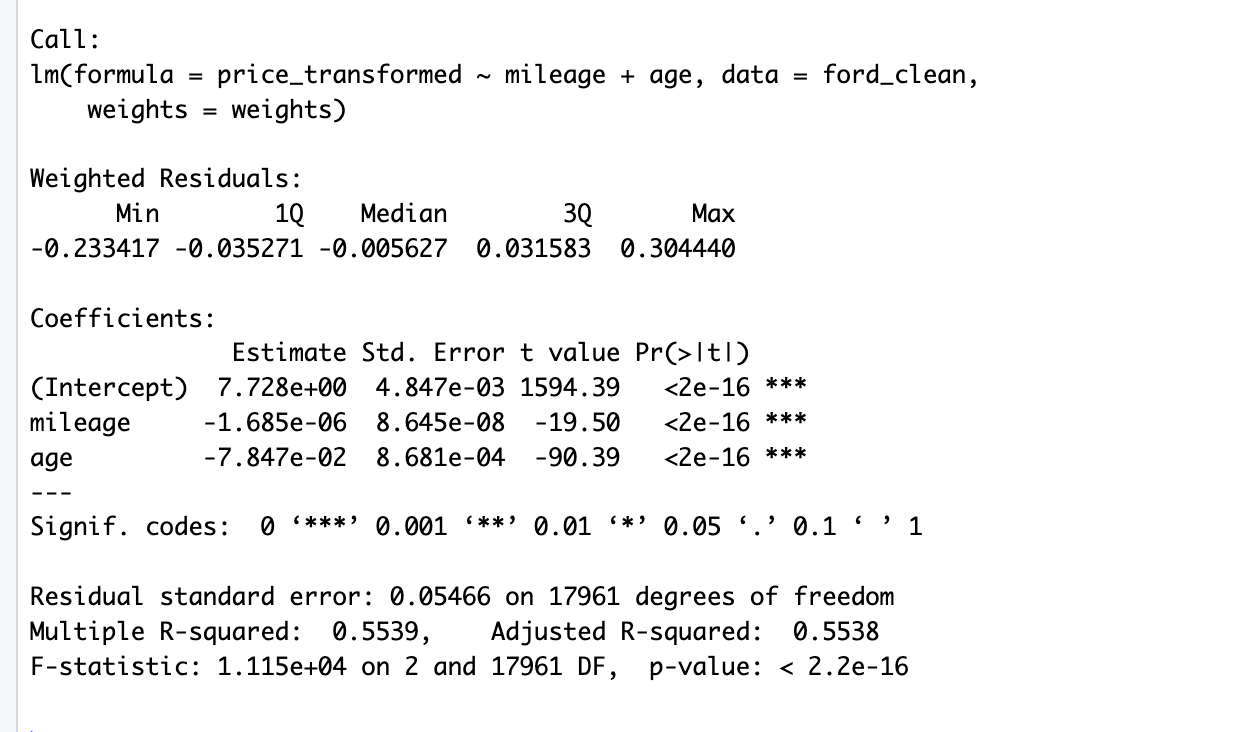


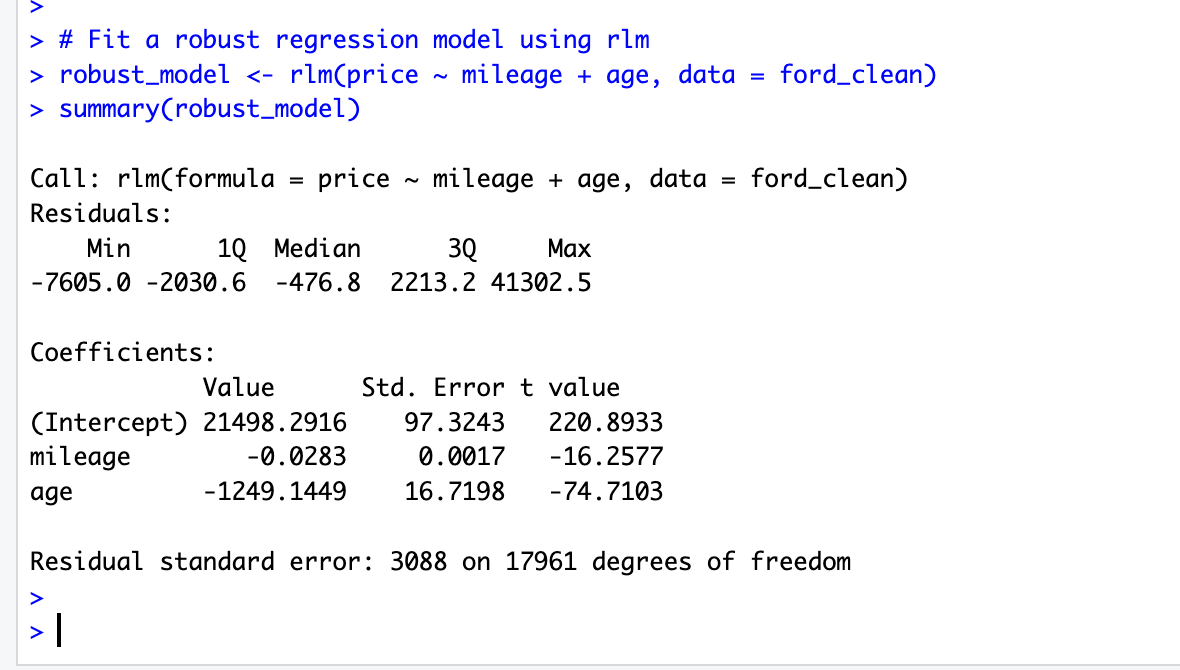
WLS

weights <- 1 / ford\_clean$age

> wls\_model <- lm(price\_transformed ~ mileage + age, data = ford\_clean, weights = weights)

> summary(wls\_model)





# Fit a robust regression model

> robust\_model <- rlm(price ~ mileage + age, data = ford\_clean)

> summary(robust\_model)

Hypothesis testing:

Finding best lambda for ridge:

library(glmnet)

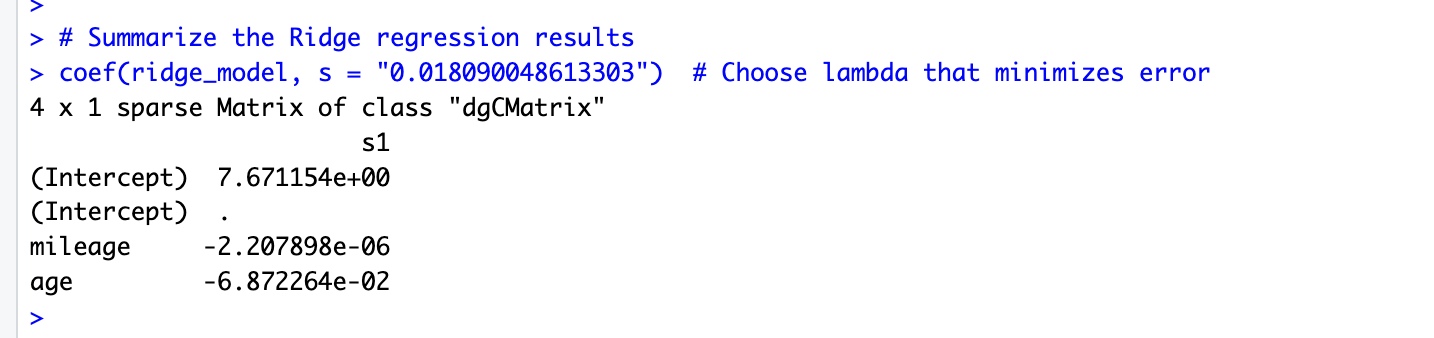
cv\_ridge <- cv.glmnet(x, y, alpha = 0)

best\_lambda\_ridge <- cv\_ridge$lambda.min

ridge\_model <- glmnet(x, y, alpha = 0, lambda = best\_lambda\_ridge)

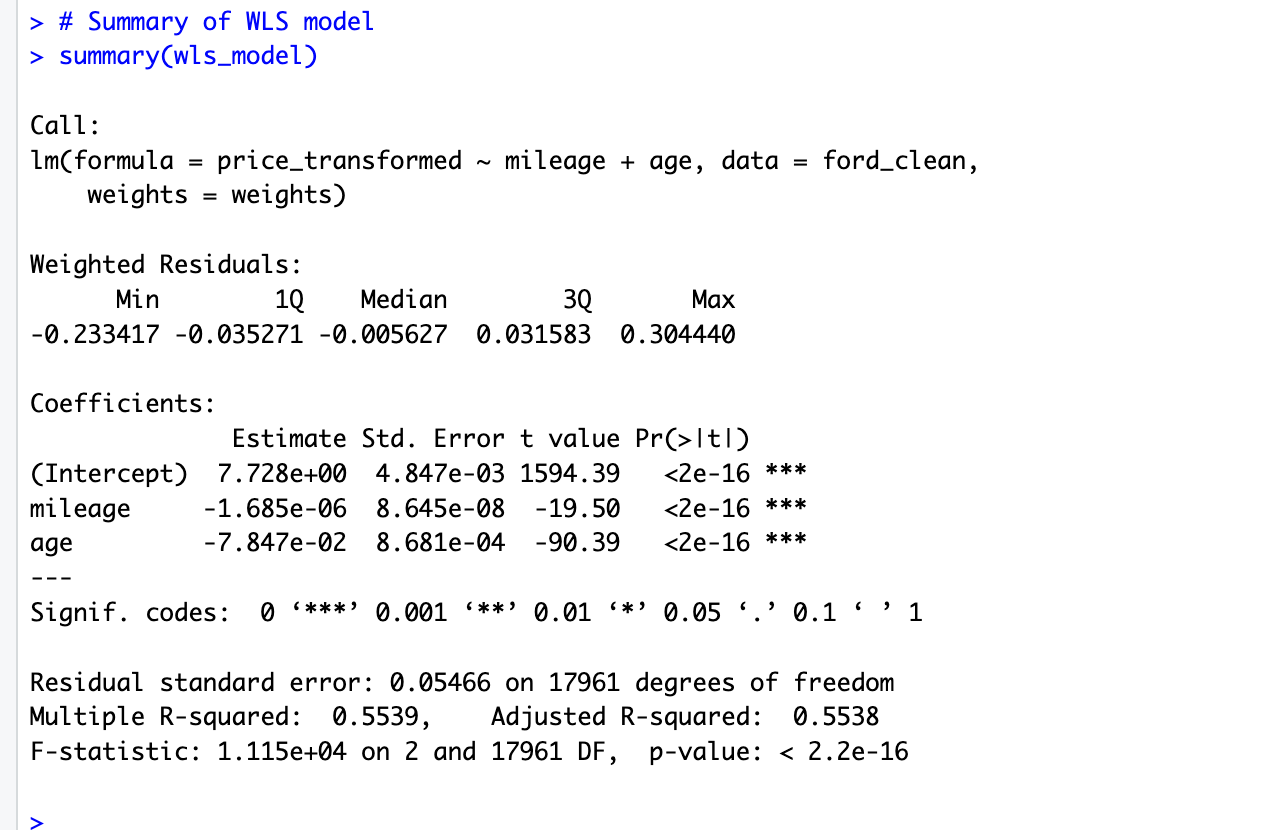
#Ridge regression results

coef(ridge\_model, s = "0.018090048613303")



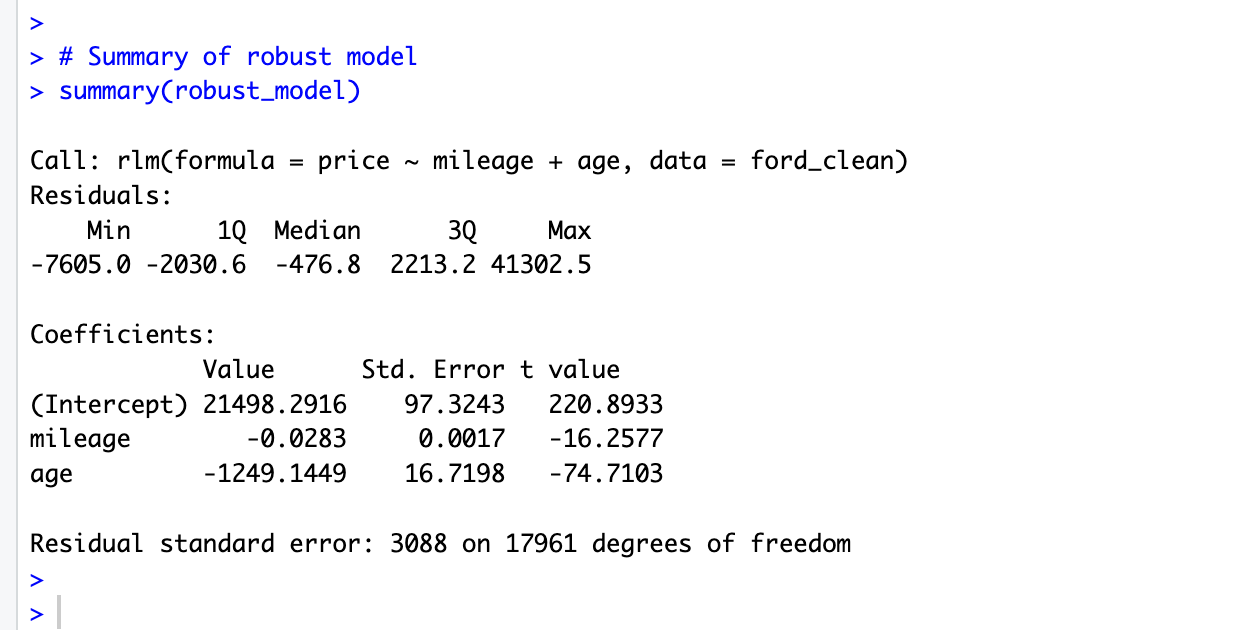
# Summary of WLS model

summary(wls\_model)



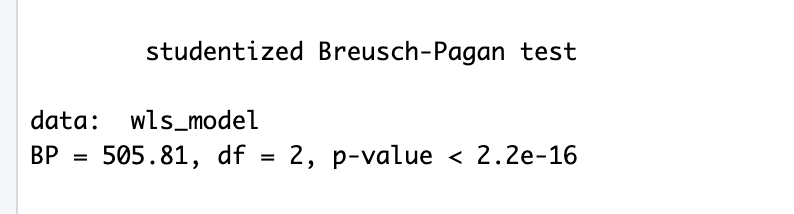
# Summary of robust model

summary(robust\_model)



# Breusch-Pagan test for heteroscedasticity

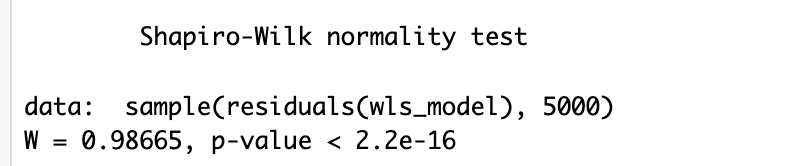
> bptest(wls\_model)



> # Shapiro-Wilk test for normality of residuals

> shapiro.test(residuals(wls\_model))

> shapiro.test(sample(residuals(wls\_model),5000))



# Diagnostic plots

> par(mfrow = c(2, 2))

> plot(wls\_model)

10 fold cross validation

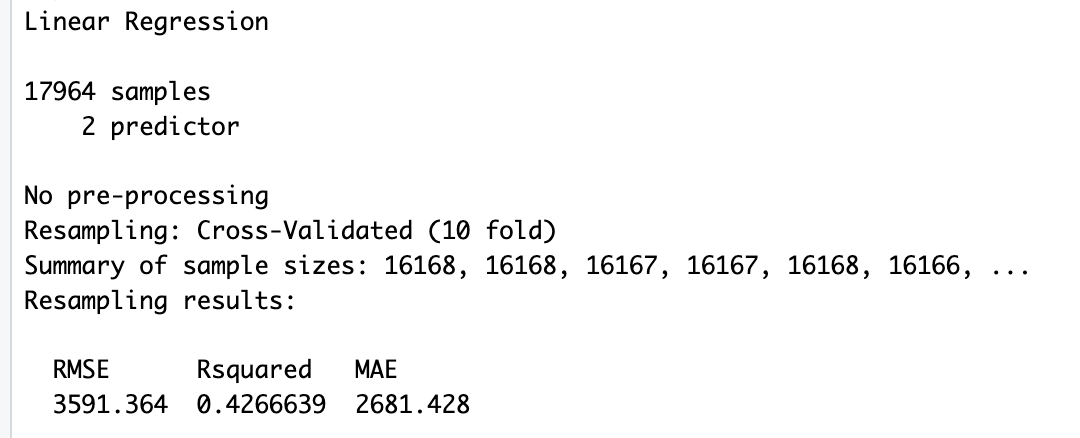
train.control <- trainControl(method="cv", number = 10)

> cv.model <- train(price ~ age + mileage, data = ford\_clean,

+ method = "lm",

+ trControl = train.control)

> print(cv.model)



Cross validation on box cox transformed price

> train.control <- trainControl(method="cv", number = 10)

> cv.model <- train(price\_transformed ~ age + mileage, data = ford\_clean,

+ method = "lm",

+ trControl = train.control)

> print(cv.model)

