Final Paper

Kotomi Oda
Kaleb Cervantes
Nikhil Taringonda

Department of Statistics and Biostatistics: California State University: East Bay

STAT 632: Linear and Logistic Regression

Dr. Joshua Kerr

2022-05-04

Which characteristics of Electric Vehicles have a significant impact on increasing their price?

The dataset we used was the "Cheapest Electric Cars" from Kaggle user KOUSTUBHK. This user scraped data from https://ev-database.org/ in August 2021. The dataset contains 180 rows and 11 columns. Some of the columns were stored as strings, but clearly contained numeric substrings that were useful predictors. Some of these strings have the value "-" which indicates a null value. As such, these will be converted to NA when the numeric parts are parsed out. After the data was cleaned, the following figure was able to be shown.

Since we wanted to measure price, the variable `PriceinUK` was chosen to be the response variable. There was also `PriceinGermany`, however since this was also a type of the response we wanted, this was decided to be an alternate response variable. This left the columns `Name`, `Acceleration`, `TopSpeed`, `Range`, `Efficiency`, `FastChargeSpeed`, `Drive`, and `Numberofseas` as most of the predictors.

There is still one remaining predictor. Originally, the data contained the column `Subtitle`. This column contained information about the type of vehicle as well as its battery capacity in kWh. Since all vehicles were of the same type, only the battery capacity component was needed. This numeric variable was stripped and renamed to `battery_capacity`.
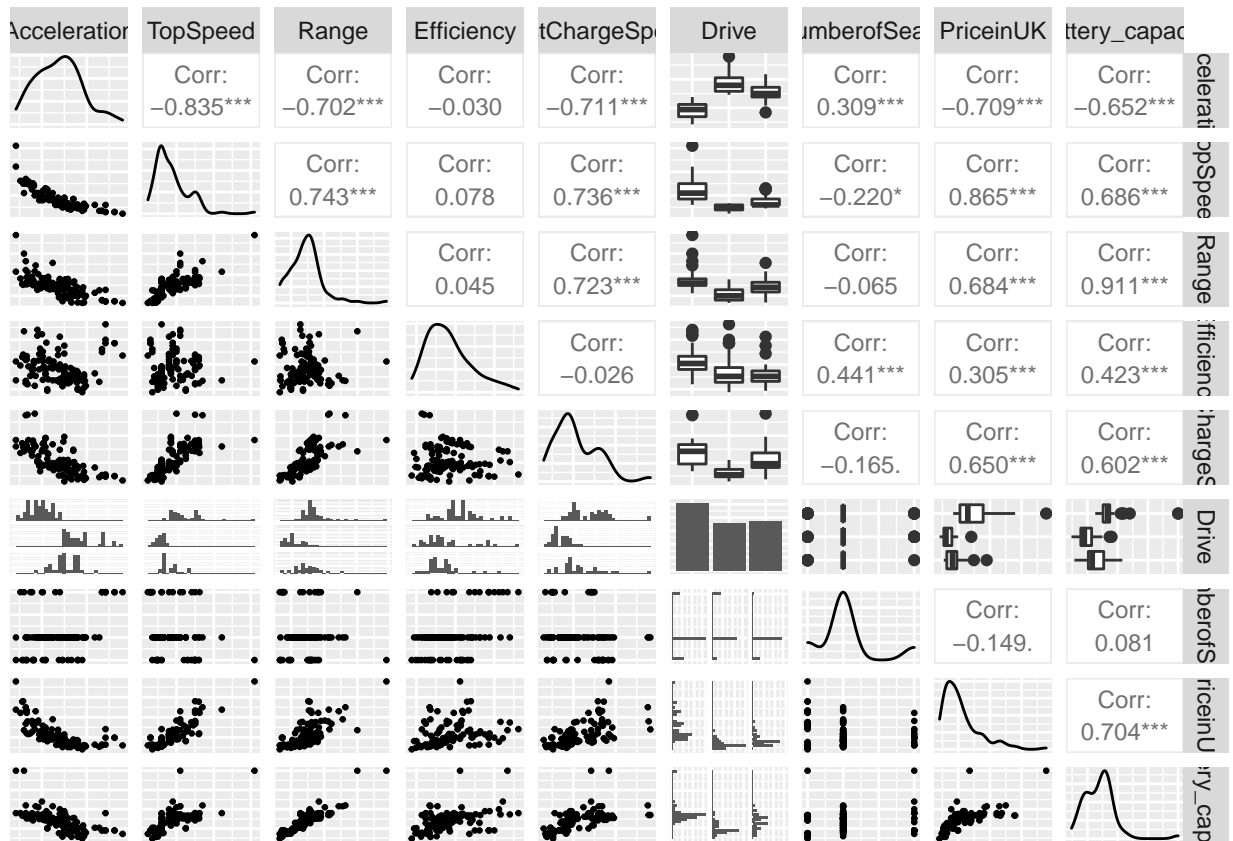

Figure 1: Correlation/Scatterplot/Density Matrix

In order to select the model, we began with a full — and untransformed — additive model. The purpose of this were to recognize which variables introduced a lot of multicolinearity. From this, the scatterplot matrix, and the correlation matrix, we were able to recognize two pairs of variables that had a lot of multicolinearity: `batery_capacity` and `Range`, `Acceleration` and `TopSpeed`. We ended up dropping the variables `battery_capacity` and `Acceleration`. This was because these predictors were less accurate in later parts of the process than `Range` and `TopSpeed` respectively.

After this, there were still possible transformations needed for the predictors. Initial predictions were found

by looking at marginal plots between the predictors and the response. Other transformations of the predictors would be tested and the more accurate predictions would be kept.

There were also possible transformations needed for the response variable. This was done by using a Box-Cox Power Transformation. In this case, we got $\lambda \approx -0.5$, which corresponds to an inverse square root transformation.

After this, there were still some insignificant predictors remaining. In order to choose the significant ones, stepwise selection — with BIC as the metric — was utilized. This resulted in the final model:

$$
\begin{aligned}
\frac{1}{\sqrt{\texttt{PriceinUK}}} = \ & \beta_0 \\
& + \beta_1 \ln \texttt{Range} \\
& + \beta_2 \texttt{TopSpeed} \\
& + \beta_3 \texttt{TopSpeed}^2 \\
& + \beta_4 \texttt{Efficiency} \\
& + \beta_5 \texttt{Efficiency}^2 \\
& + \epsilon
\end{aligned}
$$

A summary of the table can be seen below:

	Estimate	Std. Error	t value	Pr($>$|t|)
(Intercept)	0.01915	0.00154	12.43	3.386e-23
log(Range)	-0.0005637	0.0001617	-3.487	0.0006865
poly(TopSpeed, 2, raw = T)1	-3.424e-05	4.669e-06	-7.334	3.07e-11
poly(TopSpeed, 2, raw = T)2	4.427e-08	9.647e-09	4.589	1.12e-05
poly(Efficiency, 2, raw = T)1	-5.274e-05	1.392e-05	-3.789	0.0002394
poly(Efficiency, 2, raw = T)2	9.774e-08	3.382e-08	2.89	0.00459

Table 2: Final Model Summary

Observations	Residual Std. Error	R^2	Adjusted R^2
124	0.0003877	0.8588	0.8528

From Table 2, we can see that although the magnitude of the coefficients are small, they are significant. This makes sense because the model is linear on an inverse square root scale.

Using this model, we decided to recognize the observations that were both high residual and high leverage. These points were considered outliers.

From figure 2, it appears that the constant variance requirement is mostly satisfied. However, when a studentized Breusch-Pagan test was used, the resulting p-value was $1.086 \cdot 10^{-4}$. This is highly significant and indicates heteroscadicity in the model's residuals.

The next assumption that was checked was normality of residuals. From figure 2, it appears that normality of the residuals is satisfied. In order to check, a Shapiro-Wilk normality test was used. This yielded test statistic $W = 0.97628$ and p-value 0.02778. The p-value is significant for $\alpha = 0.05$, but not for $\alpha = 0.01$. Since the test statistic is also larger than 0.95, the normality condition seems satisfied.
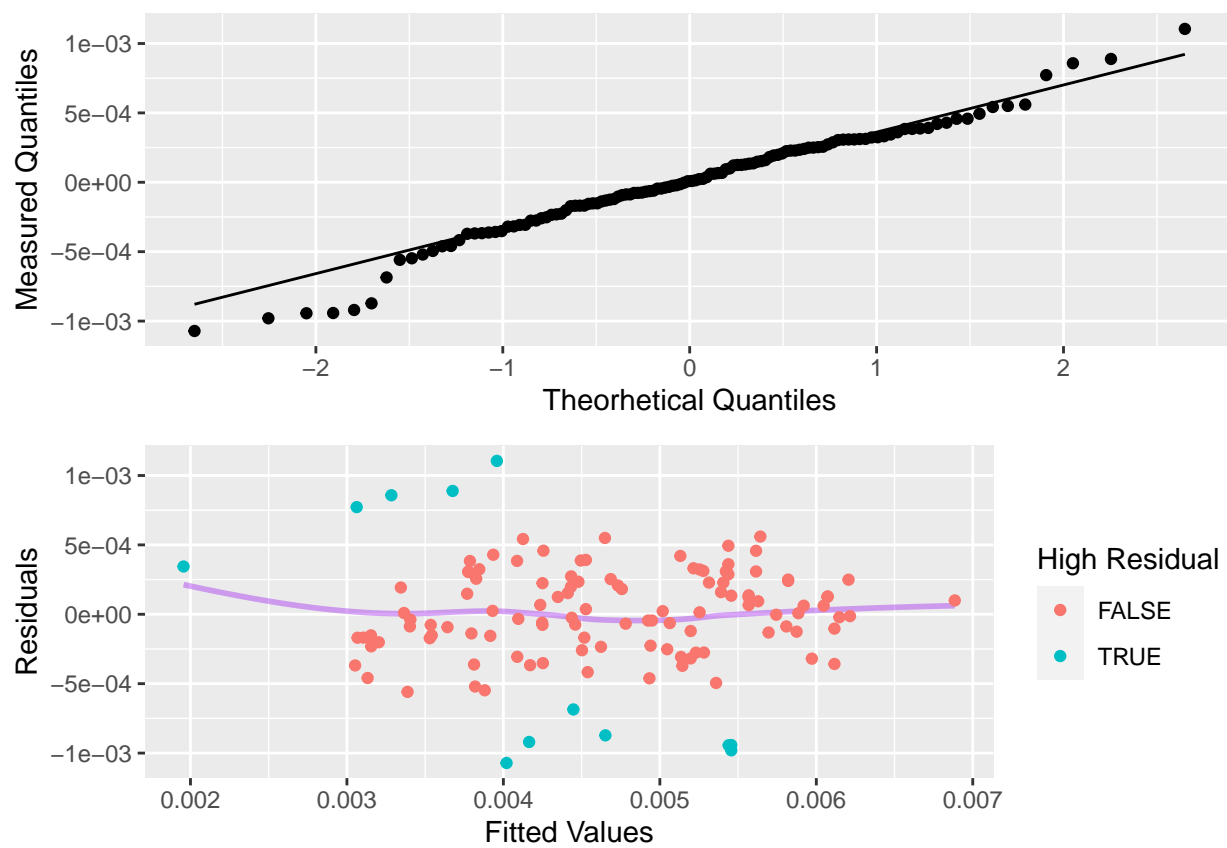
Figure 2: Plots for Regression Assumptions

Table 3: Outliers

Name	PriceinUK
Tesla Roadster	189000
Mercedes EQV 300 Long	70665
Tesla Cybertruck Dual Motor	48000
Tesla Cybertruck Tri Motor	68000

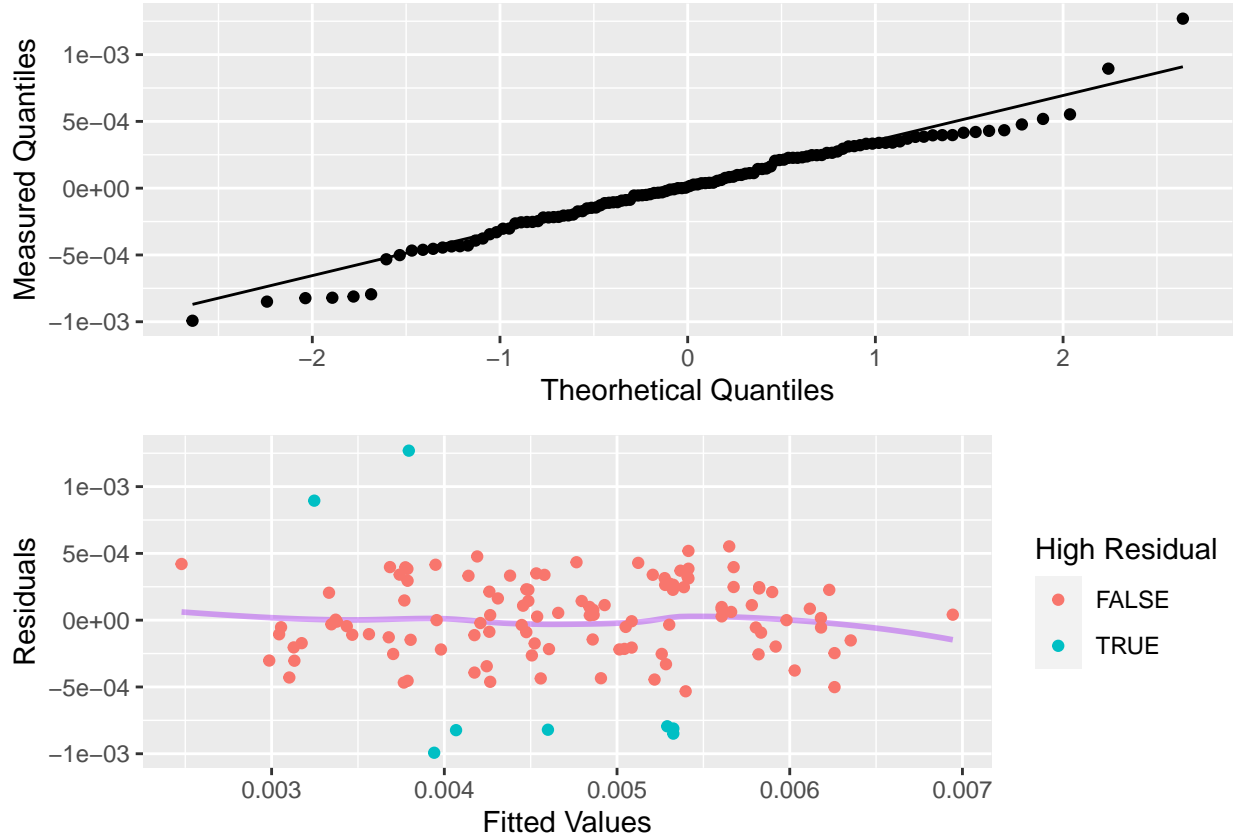We can remodel without the outlier points to see what the results are.

Figure 3: Refitted Diagnostic Plots without Outliers

From the diagnostic plots in figure 3, the results appear similar. It appears that the constant variance requirement is mostly satisfied. However, when a studentized Breusch-Pagan test was used, the resulting p-value was $1.373 \cdot 10^{-4}$. This is slightly higher than in the original model, but is still highly significant and indicates heteroscadicity in the model's residuals.

The next assumption that was checked was normality of residuals. It appears that normality of the residuals is satisfied. In order to check, a Shapiro-Wilk normality test was used. This yielded test statistic $W = 0.97688$ and p-value 0.0328. This is slightly higher than in the original model and appears normal for similar reasons.

So removing the outliers and refitting the models does not seem to have a significant effect on the diagnostics.

Code Appendix

```r
# sets up default settings for code chunks
knitr::opts_chunk$set(
  echo = F,
  message = F,
  warning = F
)

# loads necessary libraries
library(tidyverse)
library(GGally)

# load cleaned dataset and rename subtitle column
full_data <- read_csv("Dataset/cleaned_data.csv") %>%
  mutate(battery_capacity = Subtitle, .keep = "unused")

# drop incomplete observations
data <- drop_na(full_data)

# Plots figure 1
select(data, -Name, -PriceinGermany) %>%
  ggpairs(

    # change font size of correlation stuff
    upper = list(continuous = wrap("cor", size = 3)),

    # changes point size
    lower = list(continuous = wrap("points", size = 0.5)),

    # removes cluttered axis ticks
    axisLabels = "none"

    # I need to figure out how to decrease column font size
  )

# stepwise selection w/ BIC
final_model <- lm(
  1 / sqrt(PriceinUK) ~
    log(Range) +
    poly(TopSpeed, 2, raw = T) +
    poly(Efficiency, 2, raw = T) +
    FastChargeSpeed +
    NumberofSeats +
    Drive,
  data
) %>%
  step(trace = 0, k = nrow(data) %>% log)

# prints model summary in table format
summary(final_model) %>%
  pander::pander(caption = "Final Model Summary")

# indices of high residual points
```

```r
high_res <- rstandard(final_model) %>%
  abs() > 2

normplot <- ggplot(mapping = aes(sample = final_model$residuals)) +
  geom_qq() +
  geom_qq_line() +
  labs(x = "Theorhetical Quantiles", y = "Measured Quantiles")

cvarplot <- ggplot(mapping = aes(final_model$fitted.values, final_model$residuals)) +
  geom_line(stat = "smooth", method = "loess", alpha = 0.4, color = "purple", size = 1) +
  geom_point(aes(color = high_res)) +
  labs(x = "Fitted Values", y = "Residuals", color = "High Residual")

gridExtra::grid.arrange(normplot, cvarplot)

# shapiro.test(final_model$residuals)
# lmtest::bptest(final_model)

# get leverage from model
fm_lev <- hatvalues(final_model)
high_lev <- abs(fm_lev) > 2 * mean(fm_lev)

# print desired observations
filter(
  data,
  high_res,
  high_lev
) %>%
  select(Name, PriceinUK) %>%
  knitr::kable(tabel.envir = "figure", caption = "Outliers")

# subset of data without outliers
no_outlier_data <- filter(data, !(high_res & high_lev))

# refit model
no_outlier_model <- lm(
  1 / sqrt(PriceinUK) ~
    log(Range) +
    poly(TopSpeed, 2, raw = T) +
    poly(Efficiency, 2, raw = T),
  no_outlier_data
)

# new high residual points
high_res2 <- rstandard(no_outlier_model) %>%
  abs() > 2

normplot2 <- ggplot(mapping = aes(sample = no_outlier_model$residuals)) +
  geom_qq() +
  geom_qq_line() +
  labs(x = "Theorhetical Quantiles", y = "Measured Quantiles")

cvarplot2 <- ggplot(
```

```r
  mapping = aes(no_outlier_model$fitted.values, no_outlier_model$residuals)
) +
  geom_line(stat = "smooth", method = "loess", alpha = 0.4, color = "purple", size = 1) +
  geom_point(aes(color = high_res2)) +
  labs(x = "Fitted Values", y = "Residuals", color = "High Residual")

gridExtra::grid.arrange(normplot2, cvarplot2)

# shapiro.test(no_oulier_model$residuals)
# lmtest::bptest(no_oulier_model)
```