# Electric Vehicle Pricing

Eric Wilson; Min Thiha Myo; Kevin Maldonado

12/7/2021

## Introduction

Electric Vehicles are rising in popularity amongst consumers in todays day and age. With climate change playing a big factor, several nations have made a consorted effort in investing in 'green technologies' which include electric vehicles. The development and improvement of battery technology has made the electric vehicle a viable and perhaps a more preferable choice for consumers. Some of the advantages of electric vehicles include saving money on gas, environmentally friendly, low maintenance cost. Local governments are also offering incentives to buy an electric vehicle. As there is with any new technology, there are challenges that electric vehicles face. Charging station scarcity and battery range are 2 big hurdles that owners will have to face. Despite the challenges that owners and manufactures might have to overcome, the rise of ownership of electric vehicles is undeniable and will only increase heading into the future.

Our project will take data from various electric vehicles that are available in the market and from that data we will perform a regression analysis to predict the price of electric vehicles based on different variables such as battery size, acceleration, range and efficiency.

## Data Preparation

### Data Source

Our data can be found through this link: https://www.kaggle.com/kkhandekar/cheapest-electric-cars

##Loading Dataset Our EV dataset contains 180 observations (rows) and 12 variables (columns).

```
ev <- read_xlsx('Electric Vehicle Data/Cheapestelectriccars-EVDatabase.xlsx',
                sheet = 'Cheapestelectriccars- UTF8')
```

### Data Cleanup

We now go into our data and start our clean up process. We start by converting several variables to numeric. Battery Size, PriceinUK, Acceleration, Top Speed, Range, Efficiency, Fast charge speed, and Price in Germany all get converted to numeric. Along with this conversion, we perform some regex (regular expression) to our data so that we can perform a successful conversion to numeric. Our 'Drive' variable will be converted to a factor. Note that since our data set comes from european sources the units will be in kilometers.

```
# Convert Character Vectors into useable numeric vectors
ev$BatterySize <- as.numeric(gsub(".* ([0-9]{2,3}[.]*[0-9]*) kWh", "\\1", ev$Subtitle))
libra_strip <- gsub(".([0-9]{2,3},[0-9]{3})", "\\1", ev$PriceinUK) # strip the libra symbol
ev$PriceinUK <- as.numeric(gsub(",", "", libra_strip))
ev$Acceleration <- as.numeric(gsub(" sec", "", ev$Acceleration))
ev$TopSpeed <- as.numeric(gsub(" km/h", "", ev$TopSpeed))
ev$Range <- as.numeric(gsub(" km", "", ev$Range))
ev$Efficiency <- as.numeric(gsub(" Wh/km", "", ev$Efficiency))
ev$FastChargeSpeed <- as.numeric(gsub(" km/h", "", ev$FastChargeSpeed))
ev$Drive <- as.factor(ev$Drive)
ev$PriceinGermany <- as.numeric(ev$PriceinGermany)
ev <- ev[,-2] #removing subtitle column
```

We then check for any missing values in our data set and remove them.

```
# Limit ev to complete records, German Prices
ev <- ev[complete.cases(cbind(ev$FastChargeSpeed, ev$PriceinGermany)),]
attach(ev)
```

## Exploratory Data Analysis

Let's to a look at our data set now that we have removed missing values and cleaned it up.

```
head(ev)
```

```
## # A tibble: 6 x 11
##   Name           Acceleration TopSpeed Range Efficiency FastChargeSpeed Drive
##   <chr>                 <dbl>    <dbl> <dbl>      <dbl>           <dbl> <fct>
## 1 Opel Ampera-e           7.3      150   335        173             210 Front W~
## 2 Nissan Leaf             7.9      144   220        164             230 Front W~
## 3 Porsche Tayca~          2.8      260   390        215             860 All Whe~
## 4 Nissan e-NV20~         14        123   165        218             170 Front W~
## 5 Volkswagen ID~          8.9      160   275        164             260 Rear Wh~
## 6 BMW iX3                 6.8      180   385        192             520 Rear Wh~
## # ... with 4 more variables: NumberofSeats <dbl>, PriceinGermany <dbl>,
## #   PriceinUK <dbl>, BatterySize <dbl>
```

We have data available for EV prices in Germany (in euros) and prices in the UK (pound sterling ), but for the purpose of this report we will only be using prices in Germany as our response variable since it provides more data points.

### Summary of Variables

```
#summary of Battery Size
summary(BatterySize)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   23.80   45.00   67.25   66.73   78.05  200.00 
```

```
#summary of EV prices in Germany
summary(PriceinGermany)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   20490   38750   50890   59645   65096  215000
```

```
#summary of Top speed
summary(TopSpeed)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   123.0   150.0   160.0   177.7   200.0   410.0
```

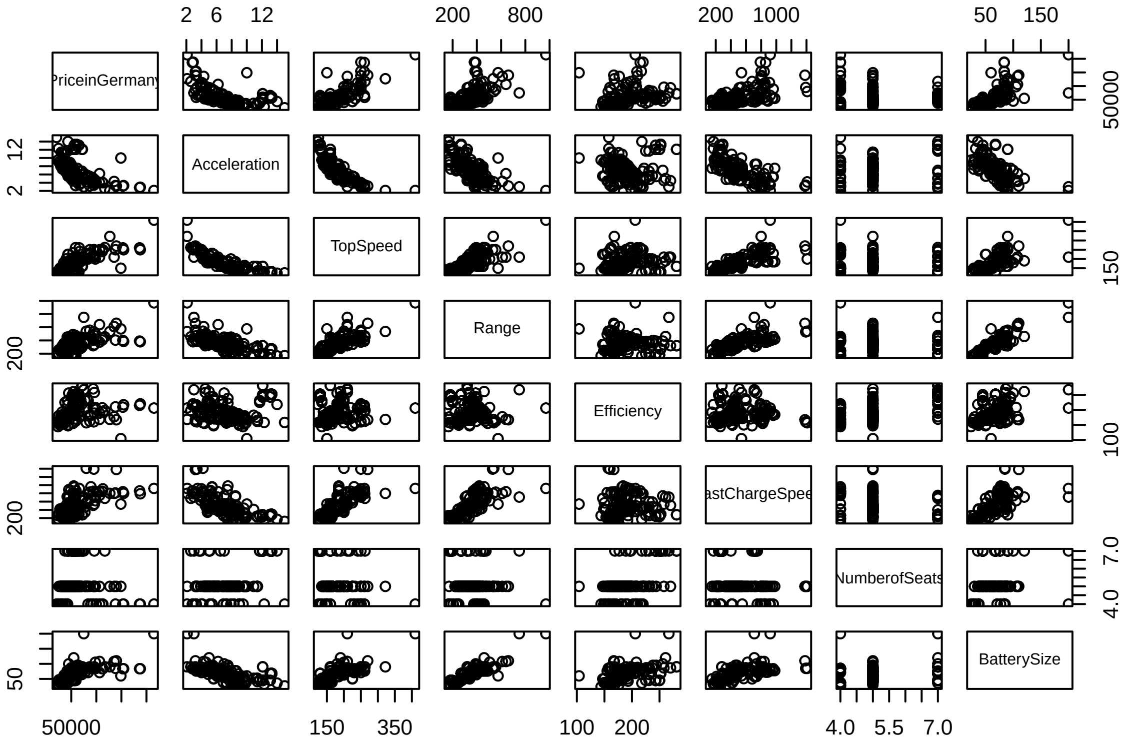
```
#summary of Battery Range
summary(Range)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   165.0   270.0   340.0   344.1   400.0   970.0
```

The electric vehicles in our data set range in price from €20,490 to €215,000 with an average price of €59,645. Battery size ranges from 23.8kWh to 200kWh with an average battery size of 66.73kWh The battery range ranges from 165km to 970km Top speed ranges from 123km/h to 410km/h with an average top speed of 177.7km/h

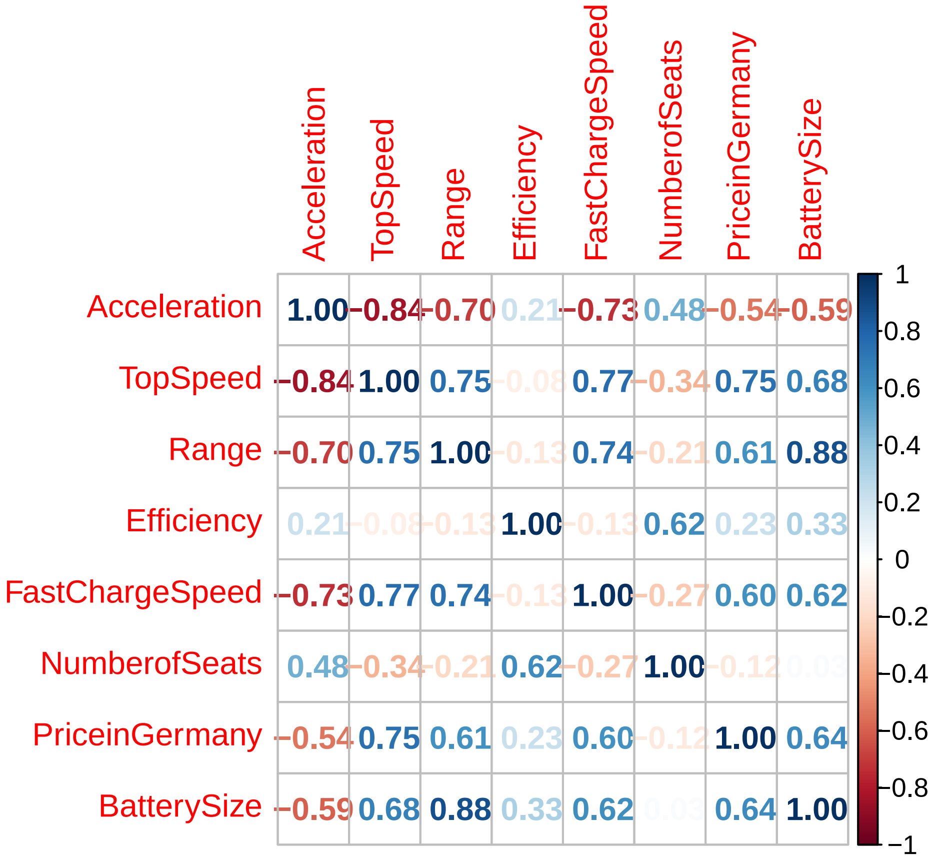
## Correlation

```
ev_numeric <- ev[, -c(1,7,10 )] #Creating numeric data frame for correlation plots and removing price i
pairs(PriceinGermany~., ev_numeric)
```



This scatter plot matrices shows the correlation that exists between variables. Starting from the top left, the variables move downward diagonally and is an axis label for the plots shown. With these scatterplots we can see that there is some correlation between PriceinGermany and top speed, range, fast charging speed and battery size. These plots might be a little difficult to read so lets do a different correlation plot.

```
corrplot(cor(ev_numeric), method = "number")
```



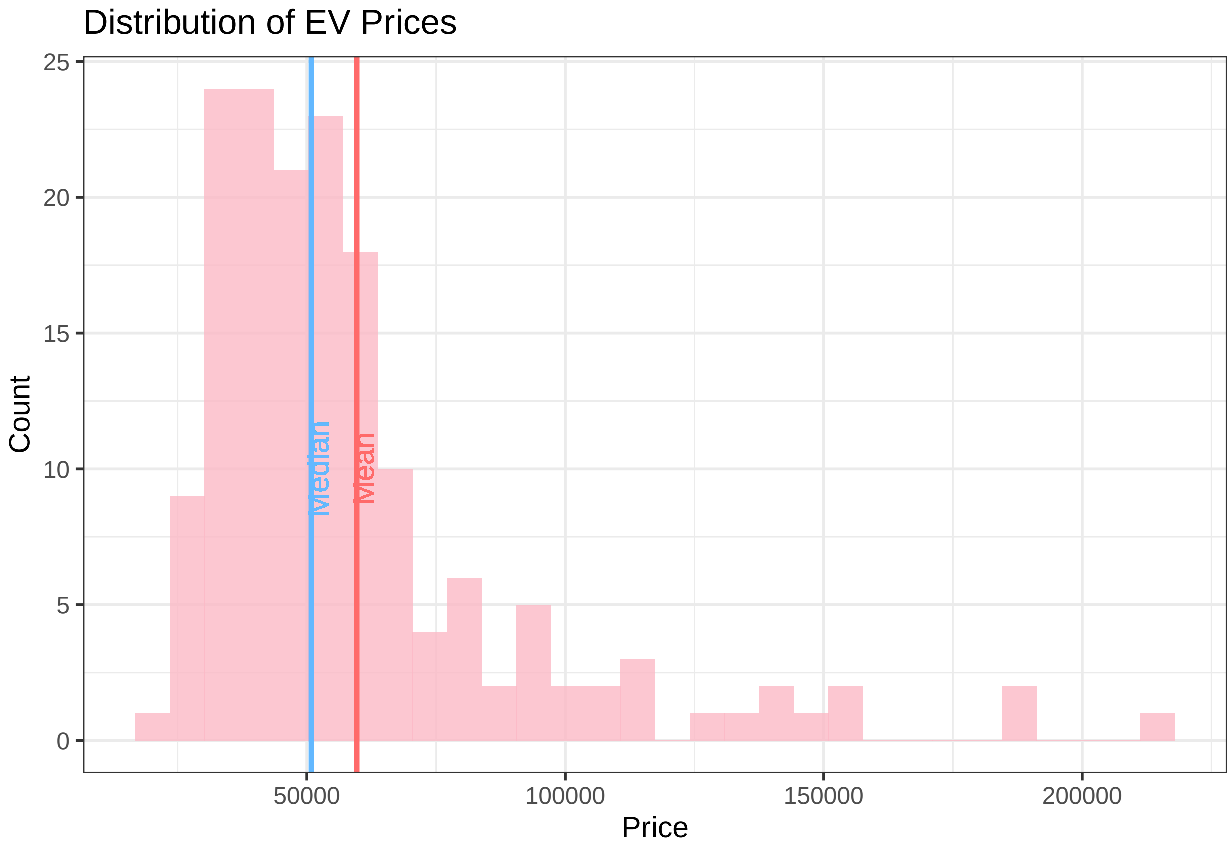
This correlation plot is also a correlation matrix. We us the argument "method = number" to show the coefficients in a number with different colors that describe their respective correlation. Confirming with what we saw in the scatterplot matrix, we can easily identify the correlation between PriceinGermany and top speed, range, fast charge speed and battery size.

### Distribution of EV Prices

Taking a more in depth look in EV prices, we will graph the distribution of the prices as well as mark the median and mean

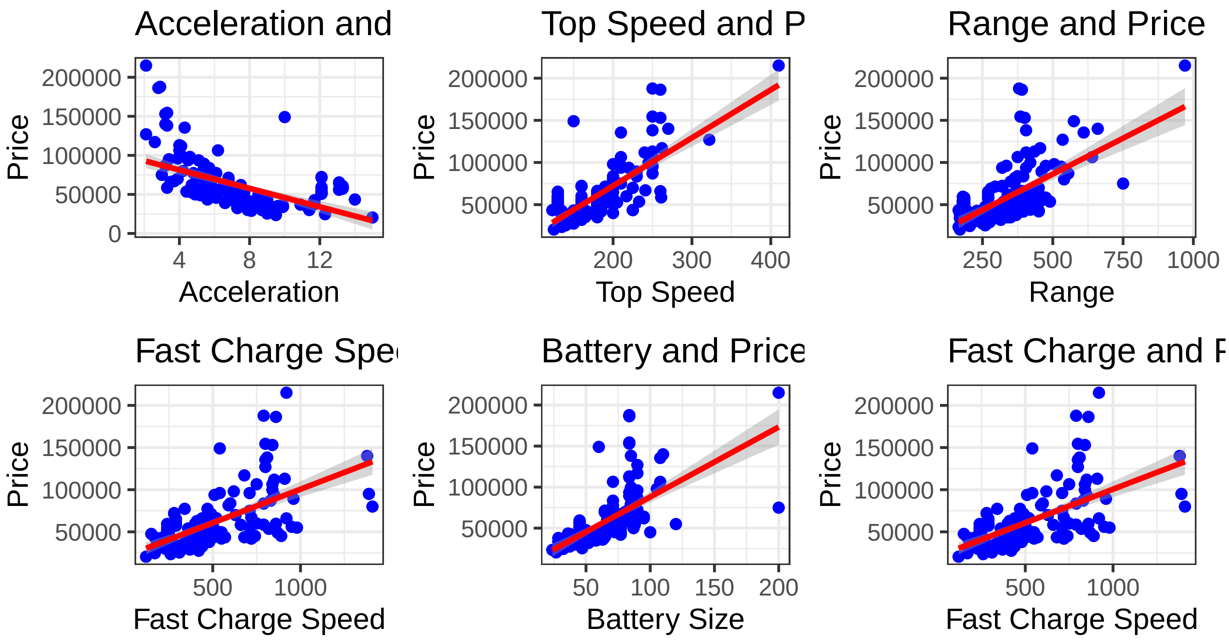
```
ggplot(ev_numeric, aes(x = PriceinGermany)) +
  geom_histogram(alpha = 0.8,fill = "#fcb9c6") +
  geom_vline(aes(xintercept = mean(PriceinGermany)),col='indianred1',size=1)+ #Vertical Line for Mean
  geom_text(aes(x=mean(PriceinGermany) + 1250, label="Mean", y=10), colour="indianred1", angle=90) +
  geom_vline(aes(xintercept = median(PriceinGermany)),col='steelblue1',size=1)+ #Vertical Line for Medi
  geom_text(aes(x=median(PriceinGermany) + 1250, label="Median", y=10), colour="steelblue1", angle=90) 
  labs(x= 'Price',y = 'Count', title = paste("Distribution of EV Prices")) +
  theme_bw()
```

```
## 'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.
```



Now we will look at the relationship price has with acceleration, top speed, range and efficiency

```
## `geom_smooth()` using formula 'y ~ x'
## `geom_smooth()` using formula 'y ~ x'
## `geom_smooth()` using formula 'y ~ x'
## `geom_smooth()` using formula 'y ~ x'
## `geom_smooth()` using formula 'y ~ x'
## `geom_smooth()` using formula 'y ~ x'
```



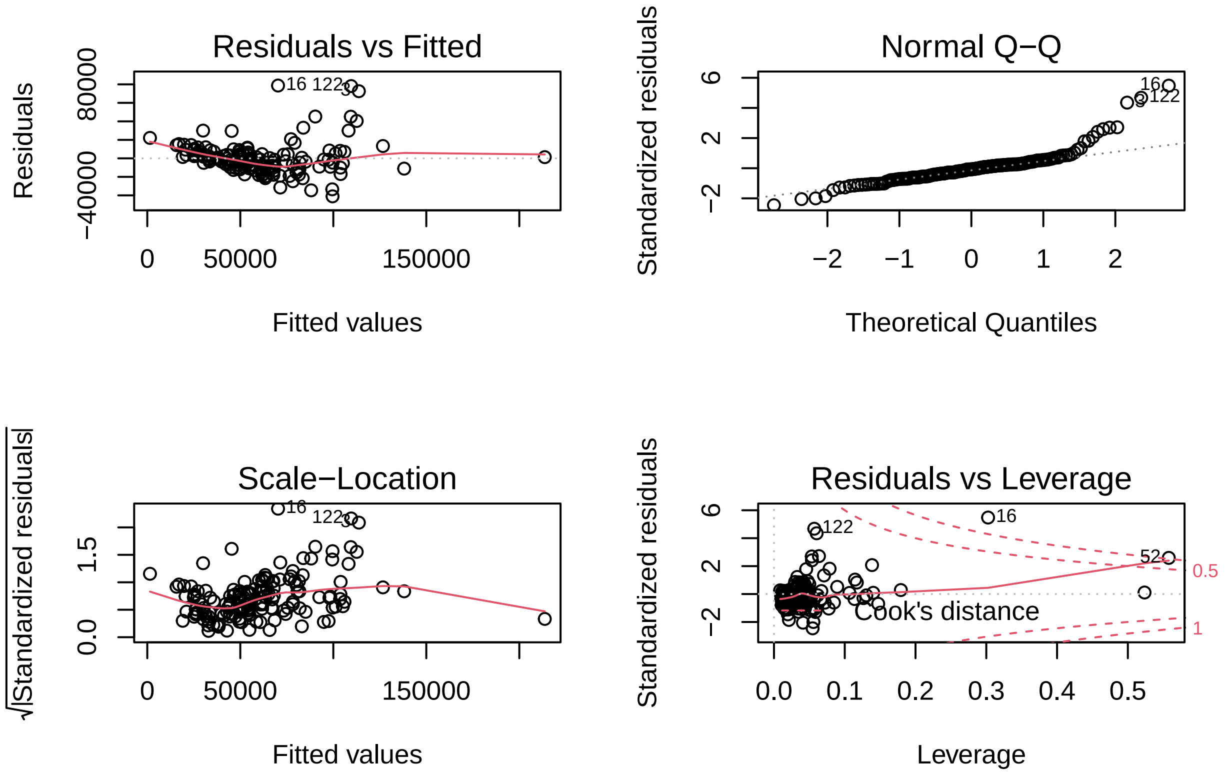
## Model Selection and Evaluation

Let's fit our data into the simplest model and perform residual analysis

```
model1 <- lm(PriceinGermany ~ ., data= ev_numeric)
summary(model1)
```

```
##
## Call:
## lm(formula = PriceinGermany ~ ., data = ev_numeric)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -41048  -9454  -1264   4493  78697
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -2.358e+05  2.510e+04  -9.394  < 2e-16 ***
## Acceleration      2.034e+03  9.816e+02   2.072   0.0399 *
## TopSpeed          5.398e+02  6.838e+01   7.893 4.86e-13 ***
## Range             4.547e+02  7.441e+01   6.110 7.64e-09 ***
## Efficiency        1.093e+03  1.410e+02   7.747 1.12e-12 ***
## FastChargeSpeed -2.057e+00  1.002e+01  -0.205   0.8376
## NumberofSeats    -9.750e+03  2.241e+03  -4.351 2.44e-05 ***
## BatterySize     -2.006e+03  3.582e+02  -5.602 9.34e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17200 on 156 degrees of freedom
## Multiple R-squared:  0.7331, Adjusted R-squared:  0.7211
## F-statistic:  61.2 on 7 and 156 DF,  p-value: < 2.2e-16
```

```
par(mfrow=c(2,2))
plot(model1)
```

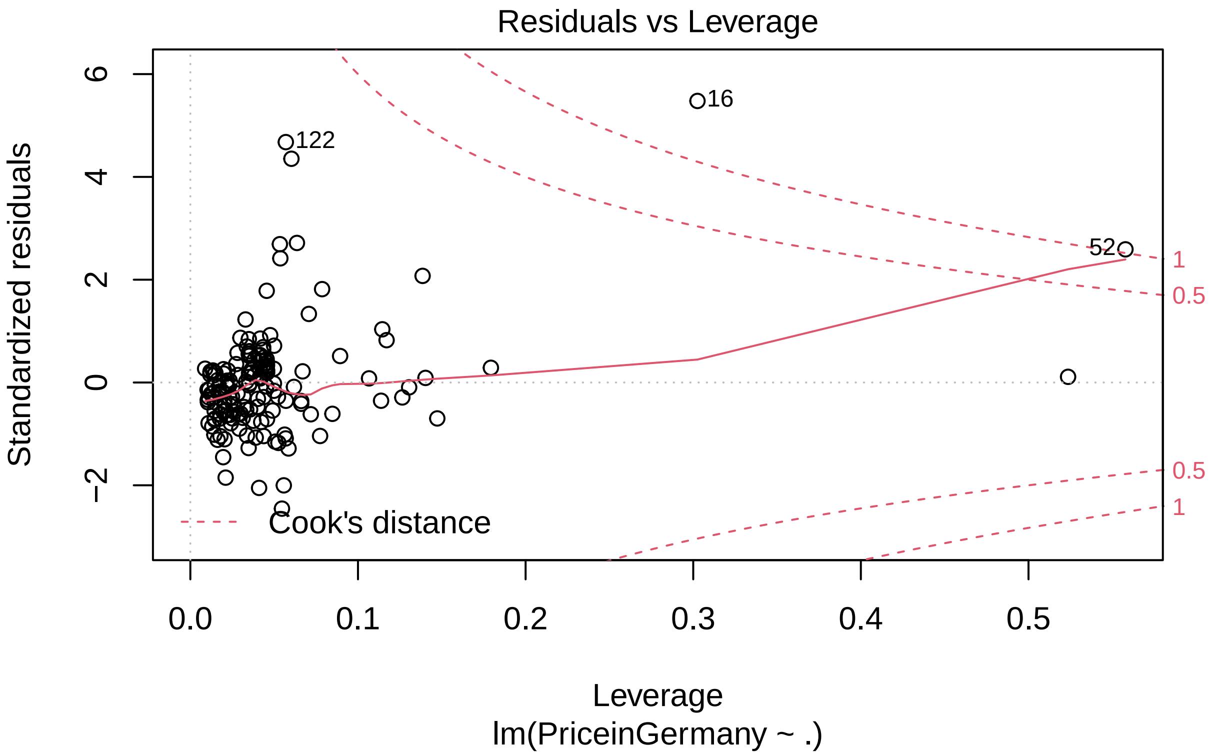


Our base model is highly significant with a good Adjusted R-squared of 0.7211. Every variable except Acceleration and FastChargeSpeed are significant. However, looking at the residual plots, we can see that there is a cosine pattern in the residuals. The normal qq plot also diverges away from the line.

## Leverage points

From our Price distribution, we can observe that there are already outliers in our dataset. However, we'd like to see how it impacts our model and if those outliers are leverage points. To do this, we utilize the Outliers vs Leverage Plot.

```
plot(model1, which = 5)
```



Let’s investigate row 16, 52, and 122

```
ev[c(16,52,122),]
```

```
## # A tibble: 3 x 11
##   Name             Acceleration TopSpeed Range Efficiency FastChargeSpeed Drive
##   <chr>                   <dbl>    <dbl> <dbl>      <dbl>           <dbl> <fct>
## 1 Lightyear One            10        150   575        104             540 All Wh~
## 2 Tesla Cybertru~           3        210   750        267             710 All Wh~
## 3 Porsche Taycan~           2.9      250   380        220             790 All Wh~
## # ... with 4 more variables: NumberofSeats <dbl>, PriceinGermany <dbl>,
## #   PriceinUK <dbl>, BatterySize <dbl>
```

```
ev_numeric <- ev_numeric[-16,]
```

All Three Vehicles are premium vehicles with a higher price point and better performance than other vehicles. We do not want to remove data points unless necessary to avoid overfitting. However one vehicle stands out more than others which is ‘Lightyear One’. It’s price point is very high at 149000 EUD, however the stats do not follow the trend in our models. This is because Lightyear One is a concept Solar Vehicle. Thus, we will consider the vehicle to be a Leverage Point and remove the row.

## Checking Multicollinearity

One of the assumptions of a linear regression model is that there is little to no Multicollinearity among the variables. This means Observations are independent of each other. Here we will fit all the variables into a linear regression model and analyze the results.

From looking at the summary of the model, we can observe that ‘FastChargeSpeed’ coefficients have the wrong sign. From our plots, we see that Price increases as ‘FastChargeSpeed’ increases. From the pairs plots, we can also observe that there is an obvious linear relationship between TopSpeed, Acceleration and Fast Charge Speed.

Thus, we will use a measure called VIF (Variance Inflation Factor) to further investigate

```
vif(model1)
```

```
##    Acceleration        TopSpeed           Range      Efficiency FastChargeSpeed 
##        4.675994        4.687021       41.221009       12.076341        3.369230 
##   NumberofSeats     BatterySize 
##        2.438362       42.062331
```

As a general rule of thumb, a VIF of 10 or greater is a cause for concern. We have three variables that have very concerning VIF scores.

## Transformation

It is imperative that we transform are variables so that we could produce a model that fits the assumptions of confidence intervals, normality and constant variance. We’ll go ahead and use Power Transform to analyze the recommended transformations.

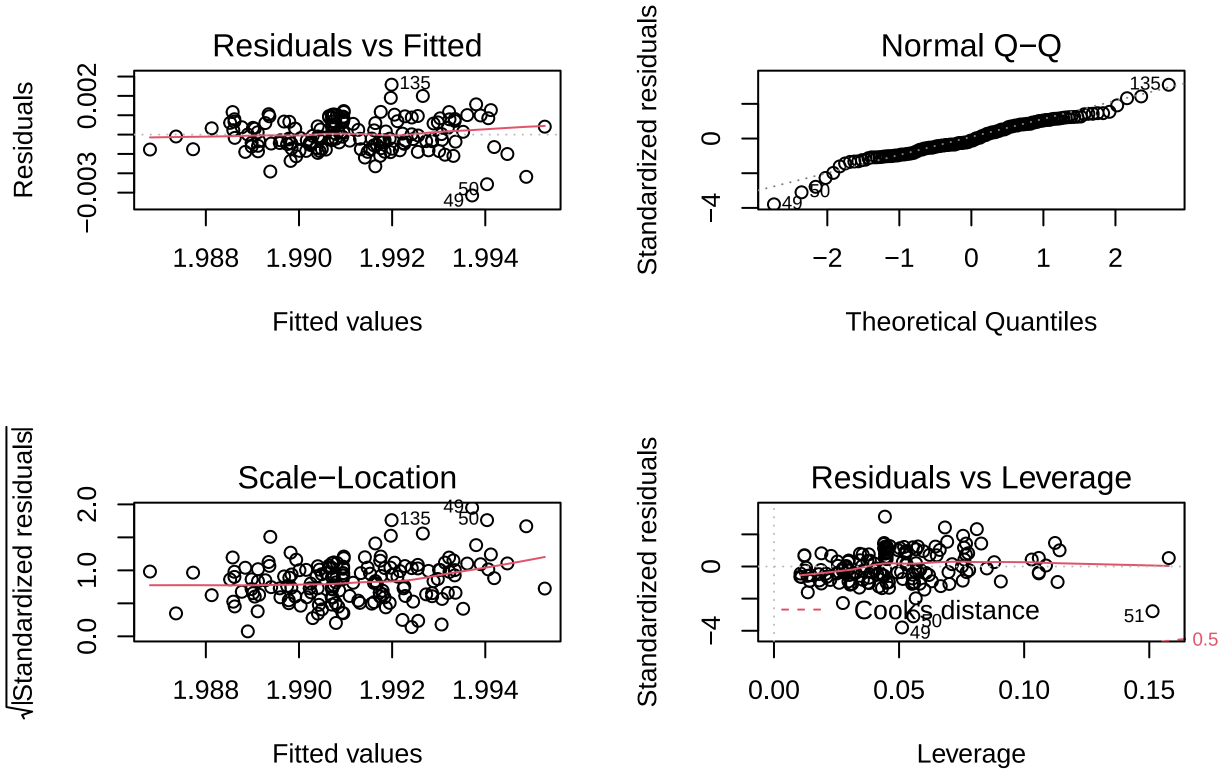
```
summary(powerTransform(cbind(PriceinGermany, BatterySize, Acceleration, TopSpeed, Range, Efficiency, Fa
```

```
## bcPower Transformations to Multinormality 
##                 Est Power Rounded Pwr Wald Lwr Bnd Wald Upr Bnd
## PriceinGermany    -0.7166        -0.5      -0.9467      -0.4864
## BatterySize        0.0024         0.0      -0.0035       0.0082
## Acceleration       0.3696         0.5       0.1239       0.6152
## TopSpeed          -0.9921        -1.0      -1.4089      -0.5752
## Range              0.0037         0.0      -0.0029       0.0103
## Efficiency         0.0087         0.0      -0.0121       0.0294
## FastChargeSpeed   -0.0310         0.0      -0.2356       0.1737
## NumberofSeats     -0.8567        -1.0      -1.6196      -0.0938
## 
## Likelihood ratio test that transformation parameters are equal to 0
##  (all log transformations)
##                                           LRT df       pval
## LR test, lambda = (0 0 0 0 0 0 0 0) 65.46494  8 3.9051e-11
## 
## Likelihood ratio test that no transformations are needed
##                                           LRT df       pval
## LR test, lambda = (1 1 1 1 1 1 1 1) 1675.886  8 < 2.22e-16
```

```
p1 <- powerTransform(PriceinGermany ~   BatterySize + Acceleration + TopSpeed + Range + Efficiency + Fa
```

Strong evidence that $\lambda = 0$ for variable Battery Size, Range, Efficiency, and Fast Charged Speed so take the $log()$ of those variable to get closer to multivariate normality.

```
model2 <- lm(bcPower(PriceinGermany, p1$roundlam) ~ log(BatterySize) + I(Acceleration^0.5) + I(TopSpeed

par(mfrow=c(2,2))
plot(model2)
```



## Variable Selection

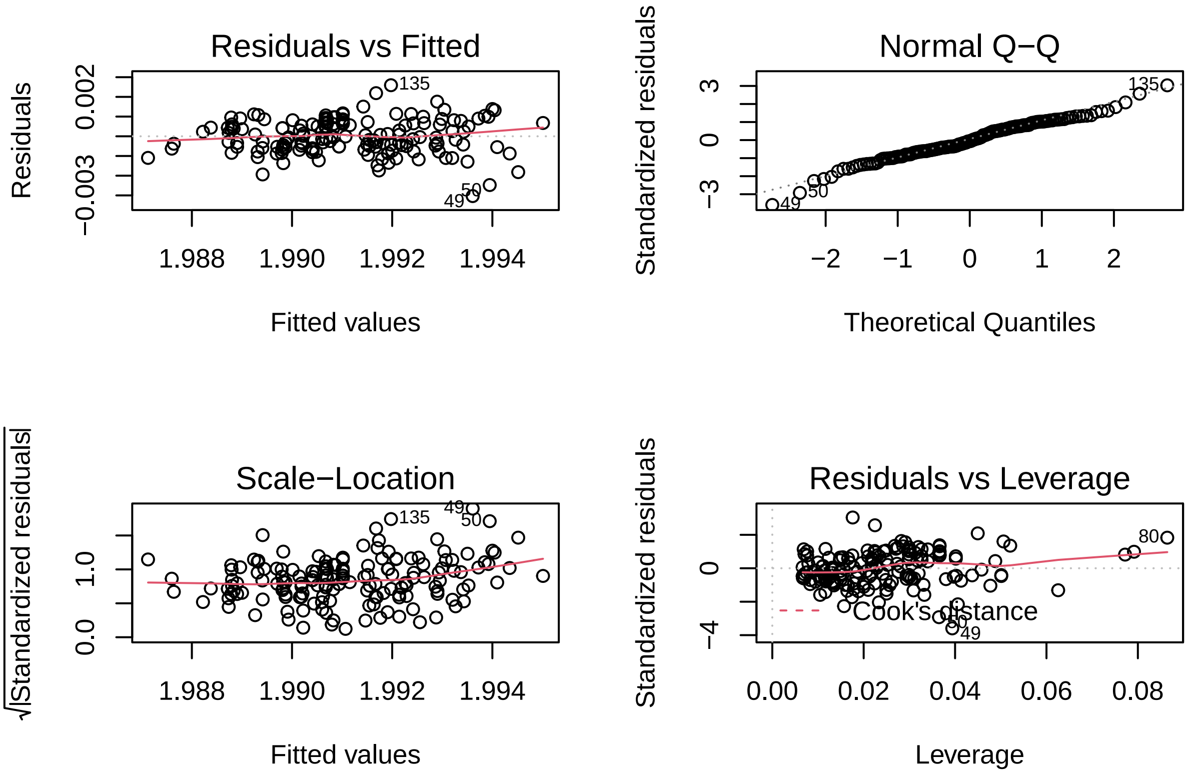
We've talked about our dataset having multi-colinearity before. To tackle that problem we will try a few Variable Selection methods. We will be using sub-setting and analyzing the recommendations. While using the regsubsets() function, we will particularly focus on Adjusted R-squared, CP values and BIC, as we cannot use $R^2$ or RSS since higher variables will win everytime.

```
library(leaps)

best2 <- regsubsets(bcPower(PriceinGermany, p1$roundlam) ~ log(BatterySize) + I(Acceleration^0.5) + I(T

with(summary(best2), data.frame(adjr2,bic,cp,rss, outmat))
```

```
##                adjr2        bic         cp          rss log.BatterySize.
## 1  ( 1 ) 0.5808465 -132.55323 141.358975 0.0002175607                *
## 1  ( 2 ) 0.4294332  -82.28517 249.859422 0.0002961514
## 2  ( 1 ) 0.7586573 -218.45419  14.868318 0.0001244903
## 2  ( 2 ) 0.7144000 -191.00921  46.385403 0.0001473193
## 3  ( 1 ) 0.7696946 -222.01270   7.983222 0.0001180545
## 3  ( 2 ) 0.7657600 -219.25150  10.767654 0.0001200714                *
## 4  ( 1 ) 0.7721409 -219.68803   7.237815 0.0001160659                *
## 4  ( 2 ) 0.7706551 -218.62856   8.282722 0.0001168228                *
## 5  ( 1 ) 0.7755692 -218.10023   5.828076 0.0001135961                *
## 5  ( 2 ) 0.7737800 -216.80596   7.078291 0.0001145017                *
## 6  ( 1 ) 0.7767596 -214.91487   6.002646 0.0001122739                *
## 6  ( 2 ) 0.7742251 -213.07478   7.762377 0.0001135485                *
## 7  ( 1 ) 0.7753231 -209.82391   8.000000 0.0001122720                *
##          I.Acceleration.0.5. I.TopSpeed..1. log.Range. log.Efficiency.
## 1  ( 1 )
## 1  ( 2 )                                  *
## 2  ( 1 )                                  *                          *
## 2  ( 2 )                   *                                         *
## 3  ( 1 )                                  *                          *
## 3  ( 2 )                                  *          *
## 4  ( 1 )                                  *          *               *
## 4  ( 2 )                                  *          *
## 5  ( 1 )                                  *          *               *
## 5  ( 2 )                   *              *          *               *
## 6  ( 1 )                   *              *          *               *
## 6  ( 2 )                                  *          *               *
## 7  ( 1 )                   *              *          *               *
##          log.FastChargeSpeed. I.NumberofSeats..1.
## 1  ( 1 )
## 1  ( 2 )
## 2  ( 1 )
## 2  ( 2 )
## 3  ( 1 )                    *
## 3  ( 2 )
## 4  ( 1 )
## 4  ( 2 )                    *
## 5  ( 1 )                    *
## 5  ( 2 )
## 6  ( 1 )                    *
## 6  ( 2 )                    *                   *
## 7  ( 1 )                    *                   *
```

Based on our results the 3(1) row looks the best. Note: The higher R^2 the better. The lower CP, the better. The lower BIC the better. We'll fit and test out the data accordingly

```
model5 <- lm(bcPower(PriceinGermany, p1$roundlam) ~ I(TopSpeed^-1) + log(Efficiency) + log(FastChargeSpe

par(mfrow=c(2,2))
plot(model5)
```



Our model looks very well. The residuals look to have a constant variance. The model follows normallity in our Normal QQ plot. From the Residual vs Leverage Point, we can see that there are no more bad leverage points. Our R squared value is 0.783. Our BIC value is -231.38. Our CP value is 89.41. Let's do some ANOVA testing and check our VIF again.

```
AIC(model5)
```

```
## [1] -1831.939
```

```
vif(model5)
```

```
##       I(TopSpeed^-1)      log(Efficiency) log(FastChargeSpeed)
##             3.298808             1.020711             3.261492
```

```
anova(model2,model5)
```

```
## Analysis of Variance Table
##
## Model 1: bcPower(PriceinGermany, p1$roundlam) ~ log(BatterySize) + I(Acceleration^0.5) +
##     I(TopSpeed^-1) + log(Range) + log(Efficiency) + log(FastChargeSpeed) +
##     I(NumberofSeats^-1)
## Model 2: bcPower(PriceinGermany, p1$roundlam) ~ I(TopSpeed^-1) + log(Efficiency) +
##     log(FastChargeSpeed)
##   Res.Df        RSS Df   Sum of Sq      F   Pr(>F)
## 1    155 0.00011227
## 2    159 0.00011805 -4 -5.7825e-06 1.9958 0.09784 .
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

From our ANOVA test, the p-value is not significant which infers that there is no difference between the model before transformation and after transformation.

## Model Validation

Let’s split our data into Training and Testing Sets by using the caret library. We’ll train the model again using K-fold Cross Validation and predict the test dataset. We will then plot the actual values and predicted values and get our post-prediction Metrics.

```
library(caret)
```

```
## Loading required package: lattice
```

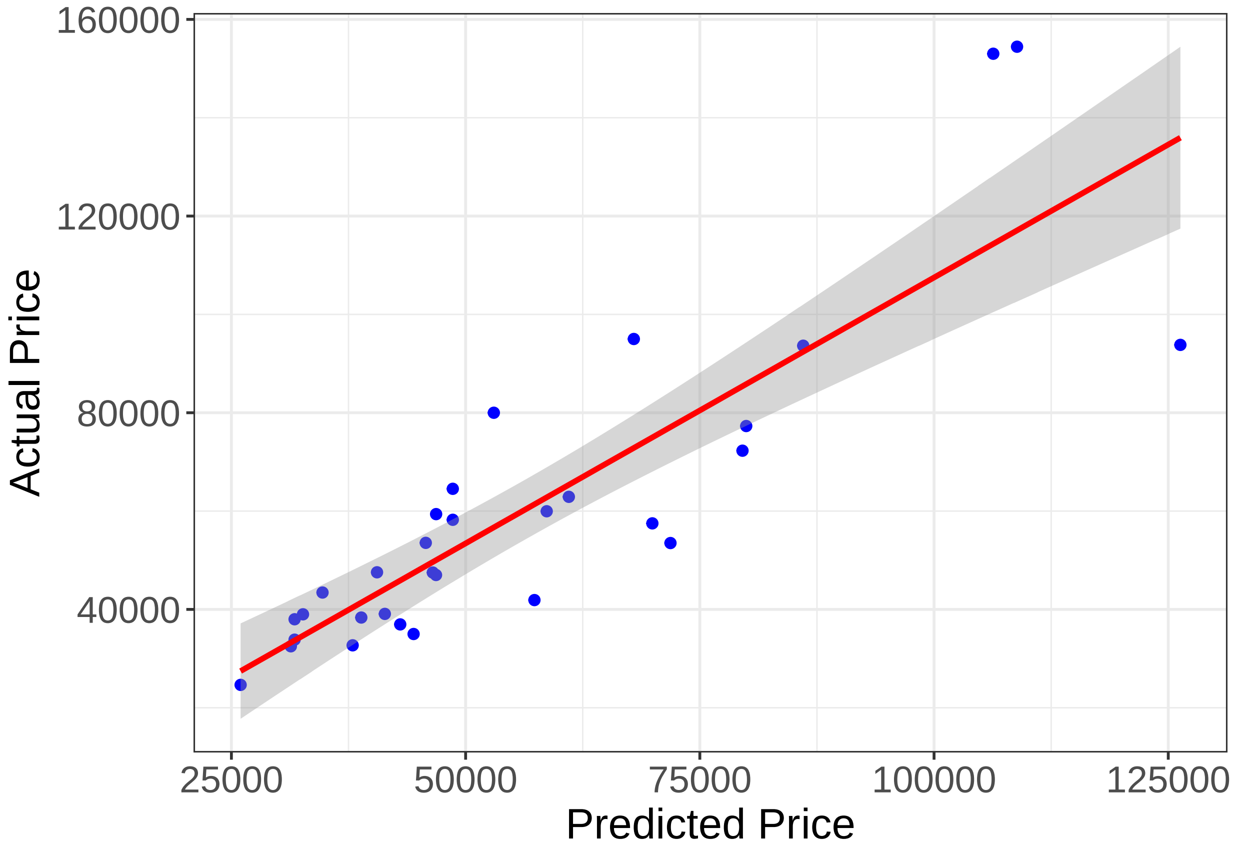
```
##
## Attaching package: ’caret’
```

```
## The following object is masked from ’package:purrr’:
##
##     lift
```

```
trainIndex <- createDataPartition(a$PriceinGermany, p = .8,
                                  list = FALSE,
                                  times = 1)

train <- a[trainIndex,]
test <- a[-trainIndex,]

formula = PriceinGermany ~ .
fitControl <- trainControl(method="cv",number = 10) #Hyper Parameters

lrmodel = train(formula, data = train,
                   method = "lm",trControl = fitControl,metric="RMSE")
```

```
ggplot(prediction, aes(x=pred,y=PriceinGermany))+
  geom_point(color = "blue")+

  stat_smooth(aes(x=pred,y=PriceinGermany),method="lm", color="red")+
  theme_bw()+
  theme(axis.title = element_text(size=16),axis.text = element_text(size=14))+
  xlab("Predicted Price")+
  ylab("Actual Price")
```

```
## ‘geom_smooth()‘ using formula ’y ~ x’
```



```
## [1] "Our prediction R2 score : 0.730632171502426"

## [1] "Our prediction RMSE score : 16619.6873400301"

## [1] "Our prediction R2 score : 11265.7441376102"

## [1] "Our prediction Error score : 0.196326481762398"
```

### Conclusion:

The purpose of this project is to build a suitable model that will accurately predict the Price of Electronic cars, as well as see the variable importance among the variables.

The final model that we have developed is:

The steps we have taken are as follows: *Remove Leverage Points* Transform our Variables * Perform variable Selection

After using those 3 techniques, Our model satisfies the assumptions of a linear regression model with decent model metrics.

1. Our model follows linearity.
2. Our predictor variables are independent of each other.
3. Our model follows Homoscedasticity and the variance of residuals are constant.
4. Our models is normally distributed.

Therefore, we can utilize this models for different purposes: if you are an electronic car manufacturer, you can set a good price point for your vehicle compared to the rest of the industry. And if you're a consumer, you can use our model to see a good price point for the stats that you'd wish.

## Limitation and Future Improvements

The biggest limitation that we have is that we do not have an accurate and detailed classification of the electric vehicles. Our datasets contains concept vehicles as well as those in production and those that are outdated. There is also a variable in the vehicle class: the dataset contains both trucks, sport cars, and sedans. There is also the topic of luxury: some vehicles are priced more than the other due to their interior furnishings instead of the variables themselves.