Regression

Jovanni Ochoa

February 10 2023

Linear regression works by taking inputs from data and trying to show some linear relationship between them. This also means that each predictor or piece of data works independently of the other prediction values. Linear regression is good at showing the strength/relationship between two pieces of data or variables. It does have a lot of weaknesses though such as hidden variables. Just because something appears to correlate doesn't mean that it's the cause. It could be for other reasons entirely. There are also interaction effects that correlate with the target and predictor, as well as, interaction effects that show a synergy with predictors.

Data cleaning

First we must load in the data

```
Autos <- read.csv(file = 'desktop/autos.csv')
```

Now, we check if there are any nulls in the columns

```
null_counts <- colSums(is.na(Autos))
total_null_count <- sum(null_counts)
```

Next, we clean the data for any outliers of the columns we want to use.

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

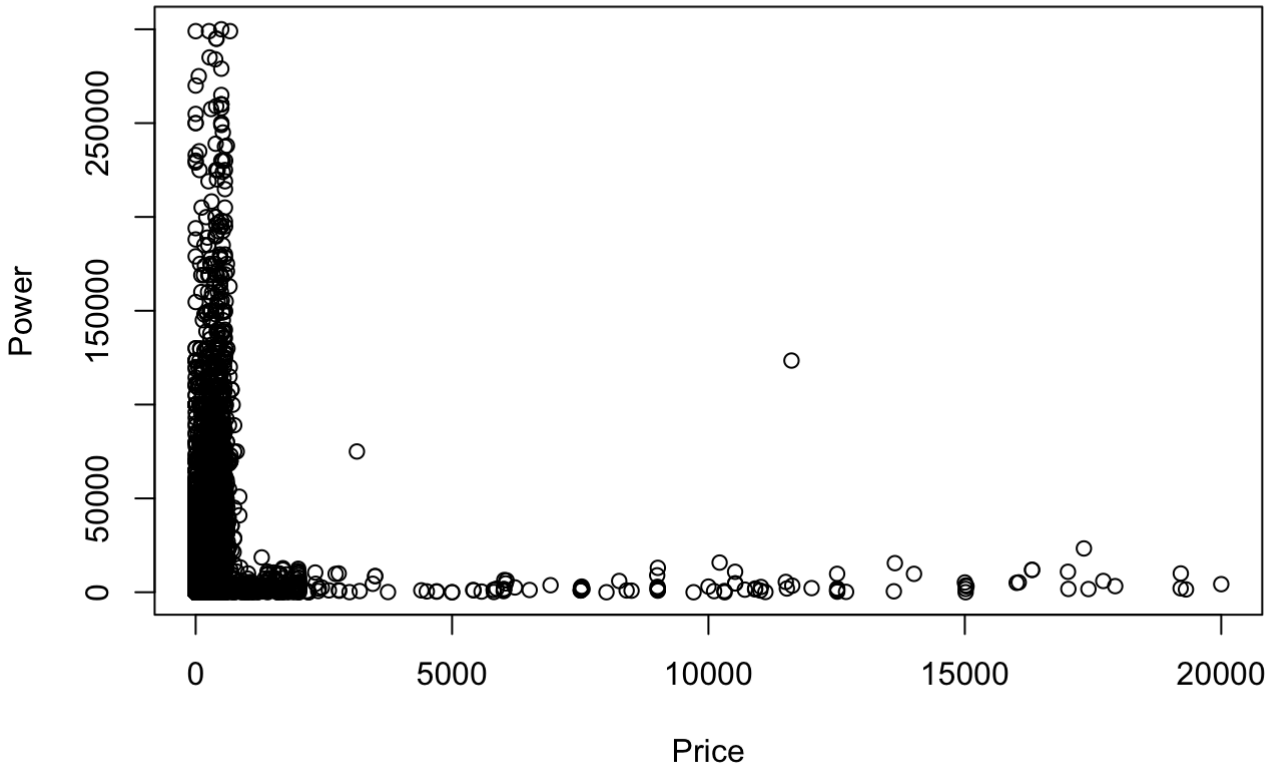
```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
remove_outliers <- function(df) {
  # Filter for columns "price", "kilometer", and "powerPS"
  df <- df %>% select(price, kilometer, powerPS)

  # Find the Q1 and Q3 values for each column with numeric data
  Q1 <- quantile(df, 0.25, na.rm = TRUE)
  Q3 <- quantile(df, 0.75, na.rm = TRUE)
  IQR <- Q3 - Q1 # Calculate the interquartile range

  # Remove rows that have values less than Q1 - 1.5 * IQR or greater than Q3 + 1.5 * IQR
  df <- df[complete.cases(df), ] # Remove rows with missing values
  df <- df[rowSums(df < (Q1 - 1.5 * IQR) | df > (Q3 + 1.5 * IQR)) == 0, ]

  return(df)
}

clean_data <- remove_outliers(Autos)
```

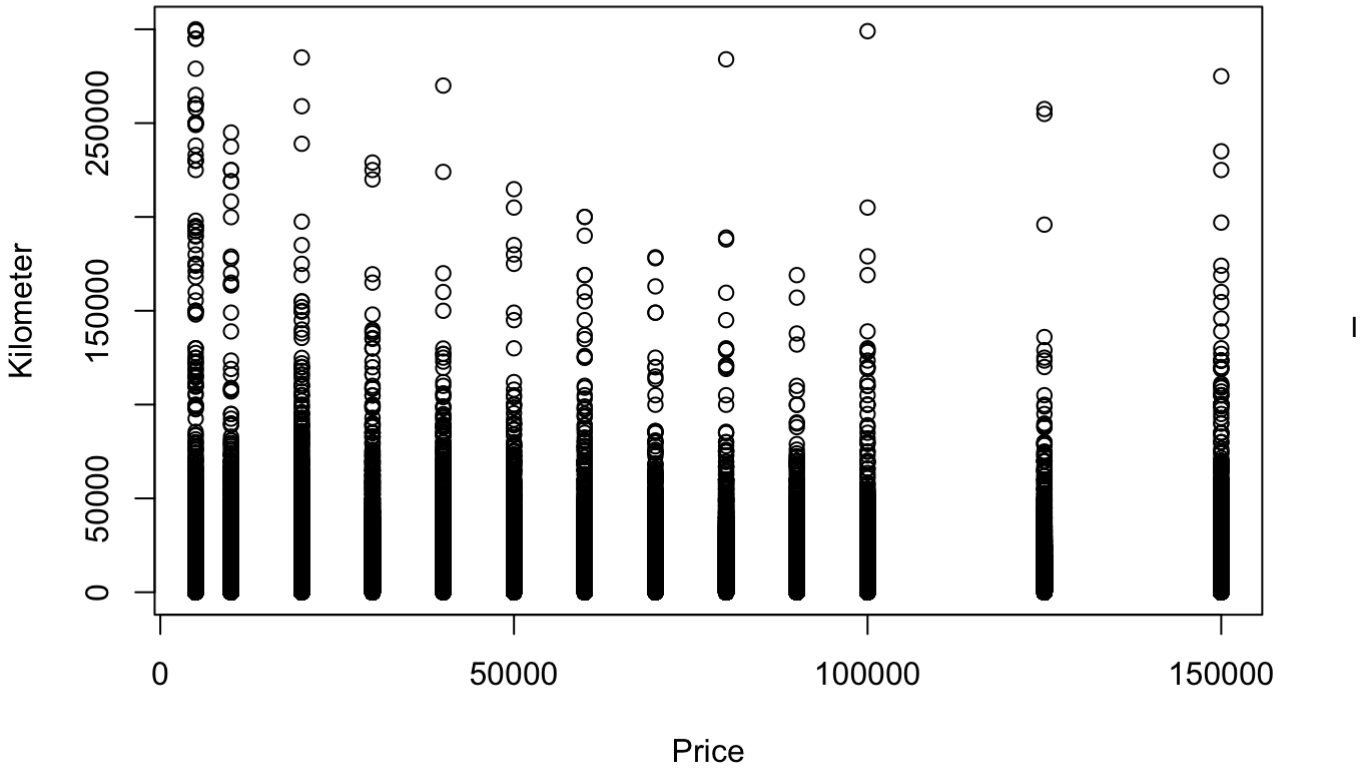
Plotting

We can start plotting the data

```
#a.
plot(clean_data$powerPS, clean_data$price,
     xlab="Price", ylab="Power")
```




```
plot(clean_data$kilometer, clean_data$price,
     xlab="Price", ylab="Kilometer")
```



notice from the plots that maybe my filter is messed up because it still looks like bad data to me.

```
#a. Divide into 80/20 train/test

# Set the seed for reproducibility
set.seed(1234)

i <- sample(1:nrow(clean_data), nrow(clean_data)*0.8, replace=FALSE)
train <- clean_data[i,]
test <- clean_data[-i,]
```

Here we try to explore the data. Maybe get a sence for what's wrong.

```
#b. Use at least 5 R functions for data exploration, using the training data

data(clean_data)
```

```
## Warning in data(clean_data): data set 'clean_data' not found
```

```
dim(clean_data)
```

```
## [1] 371406      3
```

```
head(clean_data)
```

	price <int>	kilometer <int>	powerPS <int>
1	480	150000	0
2	18300	125000	190
3	9800	125000	163
4	1500	150000	75
5	3600	90000	69
6	650	150000	102

6 rows

```
nrow(clean_data)
```

```
## [1] 371406
```

```
summary(clean_data$price)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0    1150    2950    5721    7200  300000
```

```
summary(clean_data$kilometer)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    5000  125000  150000  125634  150000  150000
```

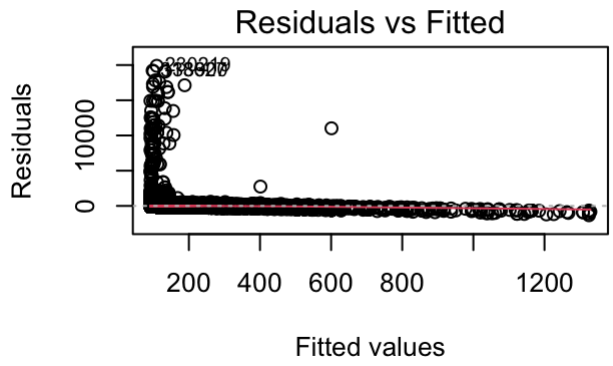
##Simple linear regression we create our first informative graphs

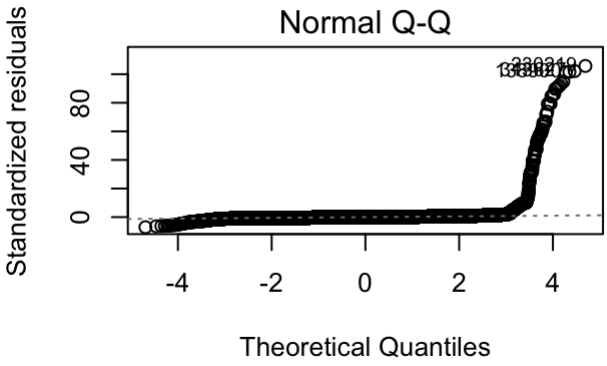
```
#c. Create at least 2 informative graphs, using the training data
lm1 <- lm(clean_data$power ~ clean_data$price, data=train)
summary(lm1)
```

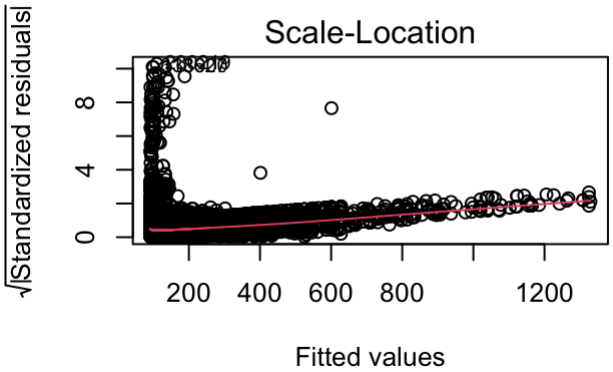
```
##
## Call:
## lm(formula = clean_data$power ~ clean_data$price, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1325.0   -36.6    -4.7    28.5 19890.4
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       9.189e+01  3.697e-01   248.5   <2e-16 ***
## clean_data$price 4.124e-03  3.560e-05   115.8   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 188.1 on 371404 degrees of freedom
## Multiple R-squared:  0.03487,    Adjusted R-squared:  0.03487
## F-statistic: 1.342e+04 on 1 and 371404 DF,  p-value: < 2.2e-16
```

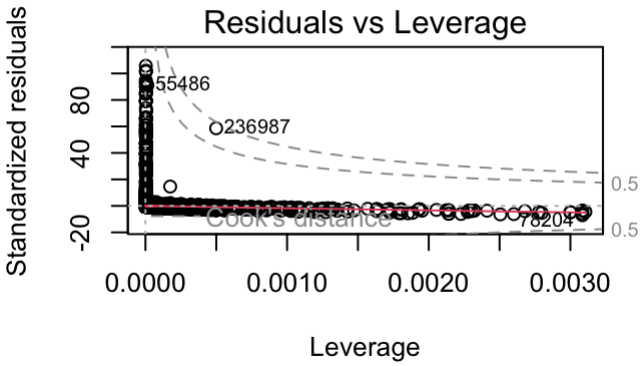
This data shows me that my data in the datafram is not good data. This can be infered by the low r squared values and the large differences in max and min even after adjustment to reduce outliers. interestingly enough, there is a low p value meaning that this data is statistically significant. Since the null value in this case is too low however, it signifies that the null value is incompatible with the data collected.

```
par(mfrow=c(2,2))
plot(lm1)
```









Write a thurough summary:

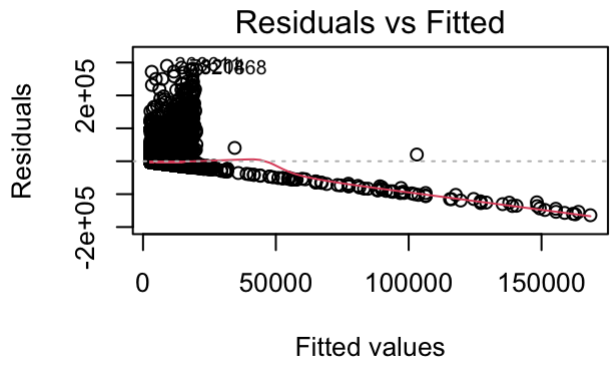
For residual versus fitted, I do not see a distinct pattern meaning that this is bad data. The theoretical quantities for my data are concerning meaning that it wasn't able to accurately predict any meaningful data. Because the x axis keeps getting narrower and narrower the red fit line is almost horizontal and shows a slight angle. The residual leverage shows that maybe some other type of model is better suited for the data since we can see both the cooks distance the the upper graphs.

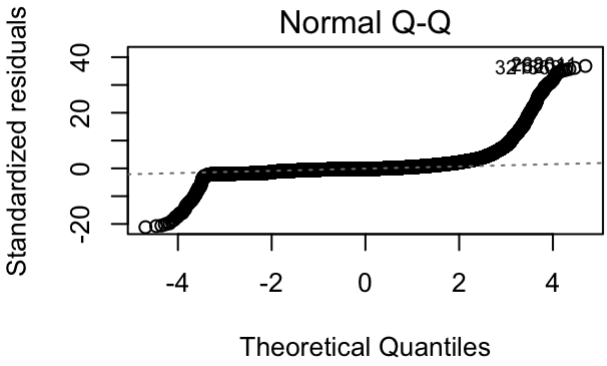
```
#f. Build a multiple linear regression model (multiple predictors), output the summary a
nd
#residual plots.
lm2 <- lm(clean_data$price ~ clean_data$kilometer + clean_data$powerPS, data=train)
summary(lm2)
```

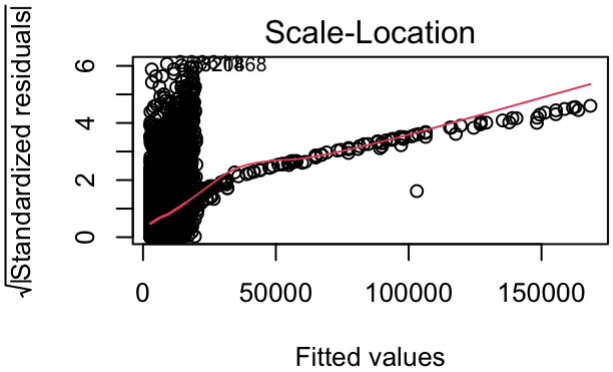
```
##
## Call:
## lm(formula = clean_data$price ~ clean_data$kilometer + clean_data$powerPS,
##     data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -164100   -2869   -1642    1369  290006
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)            1.494e+04  4.327e+01   345.3   <2e-16 ***
## clean_data$kilometer -8.101e-02  3.222e-04  -251.4   <2e-16 ***
## clean_data$powerPS    8.280e+00  6.748e-02   122.7   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7872 on 371403 degrees of freedom
## Multiple R-squared:  0.1753, Adjusted R-squared:  0.1753
## F-statistic: 3.946e+04 on 2 and 371403 DF,  p-value: < 2.2e-16
```

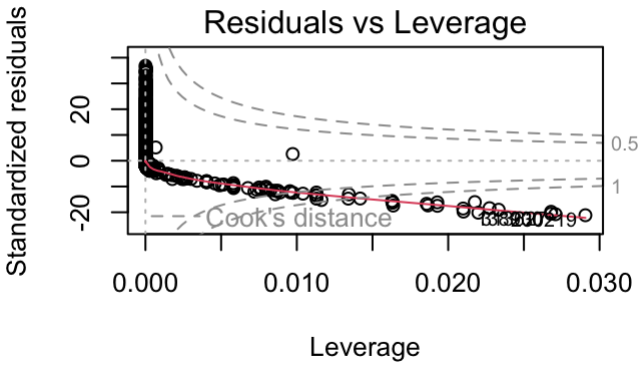
This data shows that there is a very slight linear trend. This is evident through the r squared value. Since it's 0.1753, we can conclude that there is somewhat of a trend. This might not be a good or perfect trend, but it shows there is a slight correlation. This makes sense with the data since theres more than price and power to cause a cars price to differ from another car. The degree of freedom appears to be ok, but not ideal especially with a big database like this. The p value again shows that this may be cause for concern in terms of looking at the data.

```
par(mfrow=c(2,2))
plot(lm2)
```









g. Build a third linear regression model using a different combination of predictors, interaction effects, polynomial regression, or any combination to try to improve the results. Output the summary and residual plots.

```
lm3 <- lm(price ~ poly(kilometer, 2)*powerPS, data=train)
summary(lm3)
```

```
##
## Call:
## lm(formula = price ~ poly(kilometer, 2) * powerPS, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -299071   -2910   -1651    1398  288008
##
## Coefficients:
##                               Estimate Std. Error  t value Pr(>|t|)
## (Intercept)                  4.753e+03  1.664e+01  285.645  < 2e-16 ***
## poly(kilometer, 2)1         -1.404e+06  9.035e+03 -155.426  < 2e-16 ***
## poly(kilometer, 2)2         -5.084e+04  8.692e+03   -5.848 4.97e-09 ***
## powerPS                      8.208e+00  7.499e-02  109.463  < 2e-16 ***
## poly(kilometer, 2)1:powerPS -3.173e+03  3.948e+01  -80.355  < 2e-16 ***
## poly(kilometer, 2)2:powerPS -1.593e+03  3.483e+01  -45.720  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7741 on 297118 degrees of freedom
## Multiple R-squared:  0.198,  Adjusted R-squared:  0.198
## F-statistic: 1.467e+04 on 5 and 297118 DF,  p-value: < 2.2e-16
```

We can see that this data is a lot better than the previous one by trying to do a multiple linear regression model with interaction effects and polynomial terms. THis is evident from the R-squared being greater than any of the previous versions. Again, this is not an ideal r-squared value, but it is better than the rest and shows an ever so slight trend.

Conclusion

h. Write a paragraph or more comparing the results. Indicate which model is better and why you think that is the case. So far, the last model worked better. This is probably because there are multiple factors that go into setting the price. and adding up all of the factors together will give a better understanding and a better trend for predicting price. Since I only looked at individual things it's difficult to gather any accurate information doing so; until you add everything up, only low correlations will matter to make up the whole instead of random data yet determining that may be difficult to dissern.

i. Using your 3 models, predict and evaluate on the test data using metrics correlation and mse. Compare the results and indicate why you think these results happened.

The MSE shows us the difference of data assumming the experiment were to be carried out again. When looking at that data summary for Ml3, we see that the standard error for powerPS is close to 0, meaning that reproducing the data is easy to do. Every other data poing however, would suggest otherwise. I think these results happened because using a polynomial set allows you to have more freeding, and therefore allows for better graphs when the data is non-linear which appears to be the case here.