DATA 605 - HW Assignment11

Soumya Ghosh

November 10, 2019

Libraries

```r
library(ggplot2)
library(car)
library(gvlma)
library(ggResidpanel)
```

One-Factor Linear Regression

1. Using the "cars" dataset in R, build a linear model for stopping distance as a function of speed and replicate the analysis of your textbook chapter 3 (visualization, quality evaluation of the model, and residual analysis.)

Objective

In this exercise we will build a simple linear regression model which will fit car stopping distance as a function of speed. So here Speed of the car is the **Predictor Variable** and car stoppng distance is the **Response Variable**. We will use built in car dataset in R for this exercise.

Below is a preview if the data set -

```r
cars_df <- cars
head(cars_df)
```

```
##   speed dist
## 1     4    2
## 2     4   10
## 3     7    4
## 4     7   22
## 5     8   16
## 6     9   10
```

```r
dim(cars_df)
```

```
## [1] 50  2
```

Here is the statistical summary of the data -

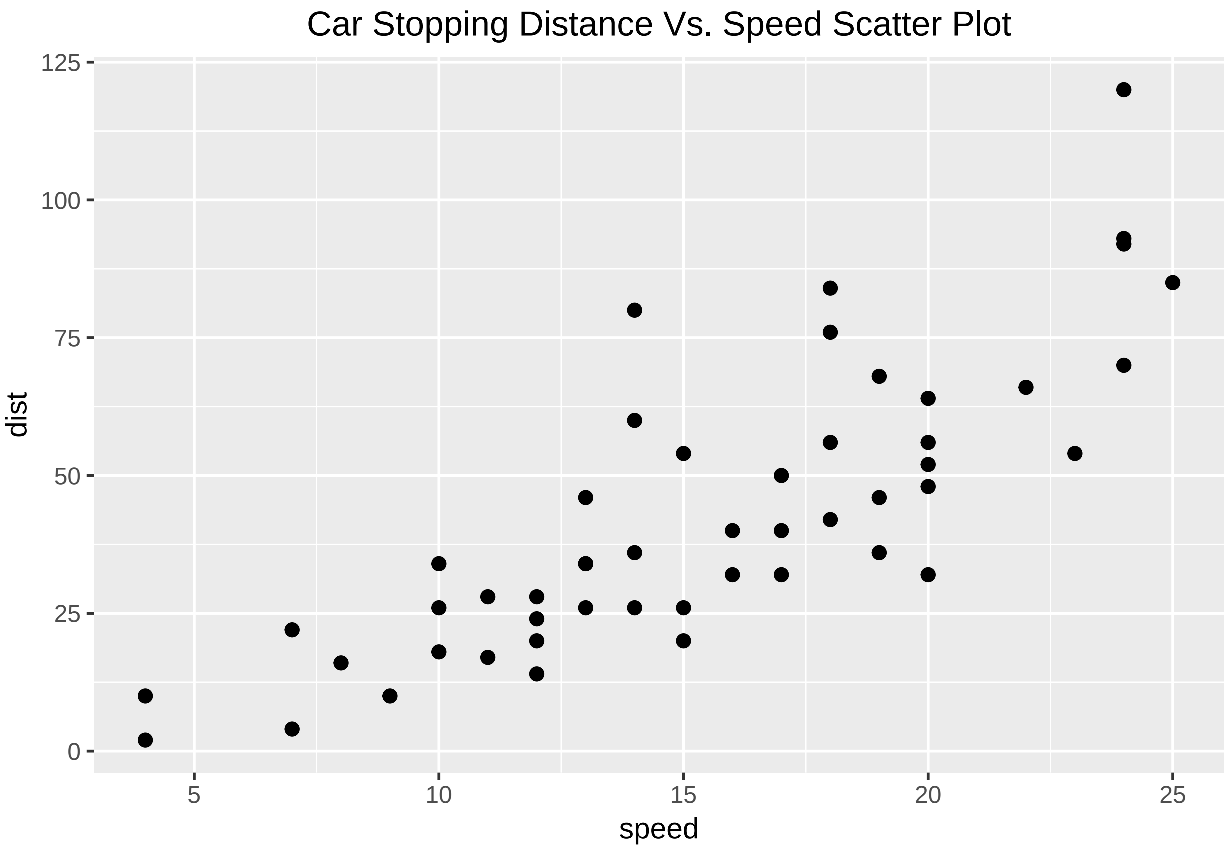
```r
summary(cars_df)
```

```
##      speed           dist       
##  Min.   : 4.0   Min.   :  2.00  
##  1st Qu.:12.0   1st Qu.: 26.00  
##  Median :15.0   Median : 36.00  
##  Mean   :15.4   Mean   : 42.98  
##  3rd Qu.:19.0   3rd Qu.: 56.00  
##  Max.   :25.0   Max.   :120.00  
```

Visualize the Data

The first step in this one-factor modeling process is to determine whether or not it looks as though a linear relationship exists between the predictor and the output value. Let's inspect through a scatter plot if there is any apparent linear relationship between the Preditor and Response variable -

```
ggplot(cars_df, aes(x=speed, y=dist)) +
  geom_point(size=2) +
  ggtitle("Car Stopping Distance Vs. Speed Scatter Plot") +
  theme(plot.title = element_text(hjust = 0.5))
```



The plot above shows that stopping distance tends to increase with the pseed of the car. If we superimpose a straight line on the scatter plot, we can observe a roughly linear relationship between the predictor and response variable although the relationship is not perfectly linear.

The next step is to develop a Regression model which will help us quantify the degree of linearity between these two variables.

Linear Model Function

Regression models is used to predict a system's behavior by extrapolating from previously measured output values when the system is tested with known input parameter values. Below is the mathemtical representation of a simple linear regression model as a straight line -

$y = \beta_0 + \beta_1 x$, where β_0 is the y intercept of the line and β_1 is the slope.

I am going to use R's lm() function to generate a linear model. For the one factor model, R computes the values of β_0 and β_1 using te method of least squares which finds the line that most closely fits the measured data by minimizing the distance between the line and the data points.

Model Coefficients

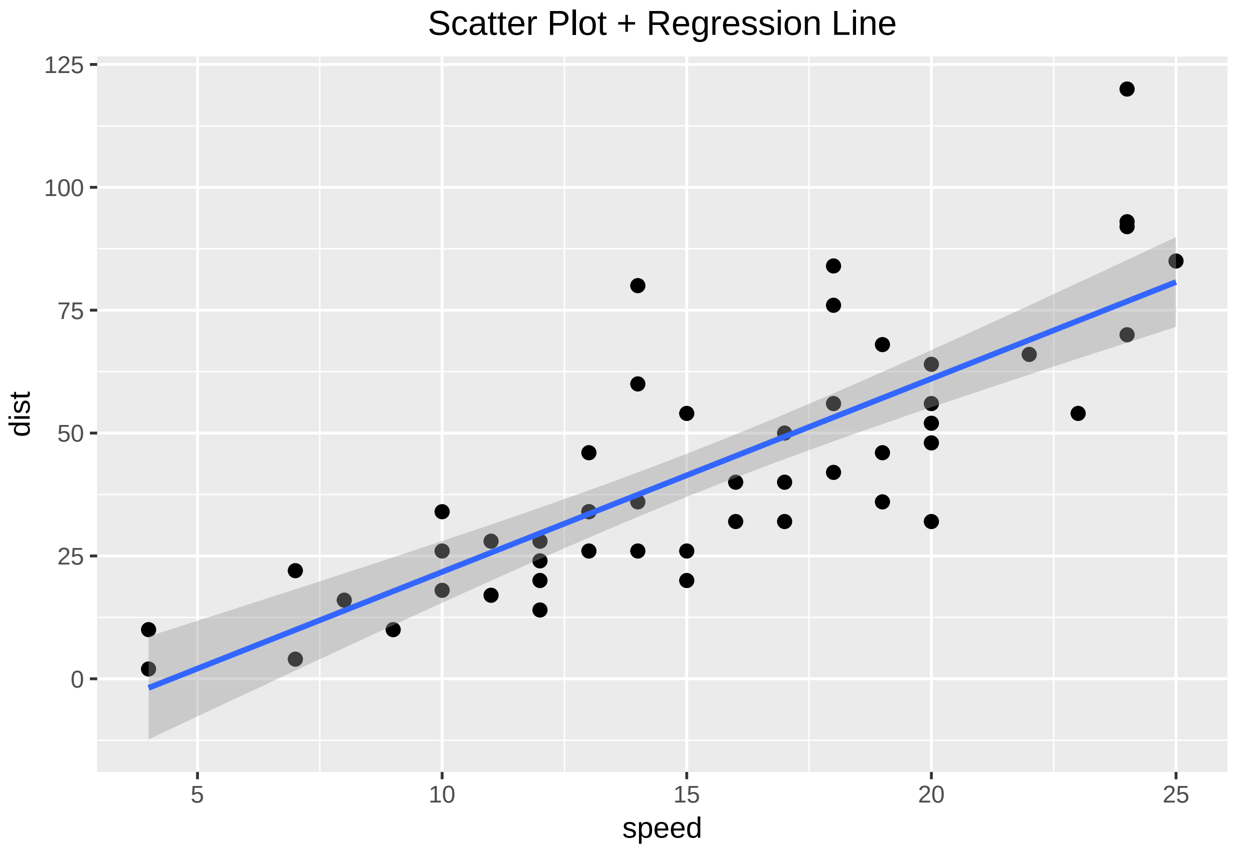
```
cars_model <- lm(dist ~ speed, cars_df)

cars_model
```

```
##
## Call:
## lm(formula = dist ~ speed, data = cars_df)
##
## Coefficients:
## (Intercept)        speed
##     -17.579        3.932
```

Here the y-intercept β_0=-17.579 and slope β_1=3.932.

Model Visualization

```
ggplot(cars_df, aes(x=speed, y=dist)) +
  geom_point(size=2) +
  geom_smooth(method=lm) +
  ggtitle("Scatter Plot + Regression Line") +
  theme(plot.title = element_text(hjust = 0.5))
```



Quality Evaluation of the Model

Using the function summary() below, some additional details can be extracted to understand how well the model fits the data -

Model Coefficients Summary

```
summary(cars_model)
```

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

Model Interpretation

1. Regression Model

The final regression model is :

$\hat{dist} = -17.579 + 3.932 * speed$

For each additional increase in the miles per hour, the model expects an increase of 3.9 feet in stopping distance.

2. Residuals

The **Residuals** are the differences between the actual measured values and the corresponding values on the fitted regression line. Each data point's residual is the distance that the individual data point is above (positive residual) or below (negative residual) the regression line.

If the line is a good fit with the data, we would expect residual values that are normally distributed around a mean of zero. With Minimum and Maximum roughly the same magnitued and 1st and 3rd quartile values also roughly the same magnitude. For this model, the residual values are little off from what we would expect for Gaussian-distributed numbers.

3. Coefficient Std. Error

The Std. Error column shows the statistical standard error for each of the coefficients. For a good model, we typically would like to see a standard error that is at least five to ten times smaller than the corresponding coefficient.

Here the Std. Error for the speed is 0.4155 which 9.46 (3.9324/0.4155 = 9.46) times smaller than the Co-efficient value. This ratio means that there is some variability in the slope estimate, β_1.

The ratio for the intercept estimate, β_0 is only 2.6 (-17.5791/6.7584=-2.6). This smaller ratio indicates significant variability for the intercept co-efficient.

4. Residuals Std. Error

The Residual standard error is a measure of the total variation in the residual values. If the residuals are distributed normally, the first and third quantiles of the previous residuals should be about 1.5 times this standard error.

The number of degrees of freedom is the total number of measurements or observations used to generate the model, minus the number of coefficients in the model. This example had 50 unique rows in the data frame, corresponding to 50 independent measurements. We used this data to produce a regression model with two coefficients: the slope and the intercept. Thus, we are left with (50 - 2 = 48) degrees of freedom.

5. Multiple R^2

The Multiple R-squared value is a number between 0 and 1. It is a statistical measure of how well the model describes the measured data.In this model, multiple R^2 is 0.6511, which means that the model's least-squares line accounts for approximately 65% of the variation in the stopping distance.

6. p-Value

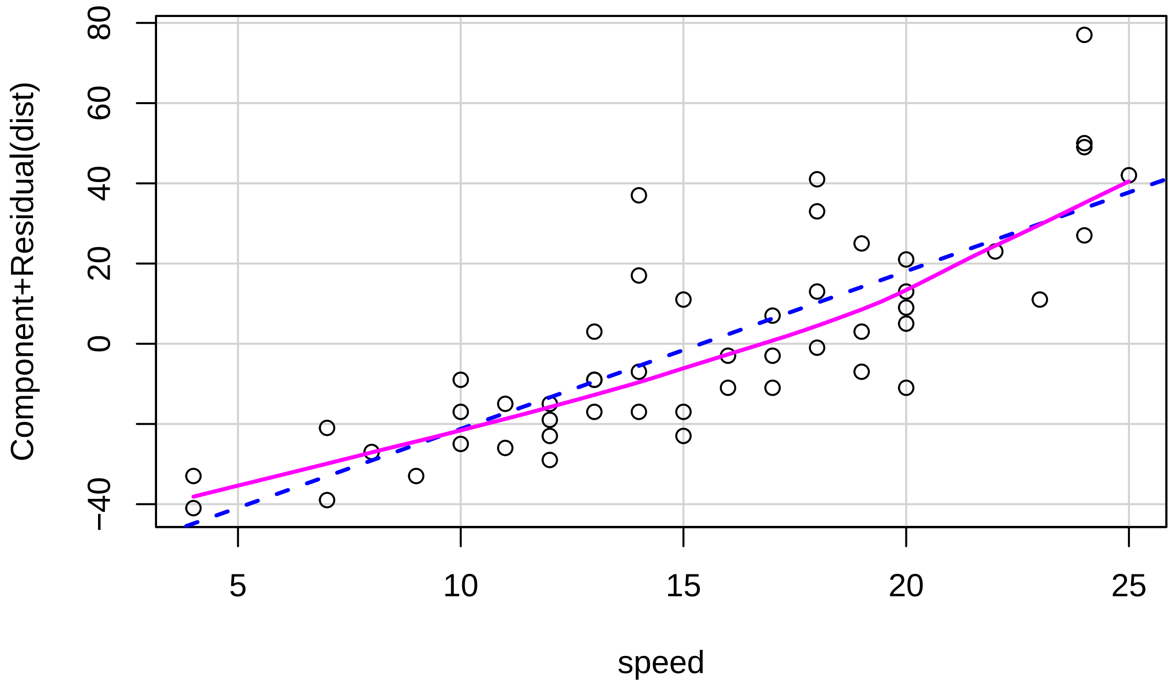
The speed's p-value is near zero and Y-intercept's p-value is ~1%, which means that there is very little chance that they are not relevant to the model.

Model Diagnostics

Linearity Test

This test is to check the degree of linear relationship between the variables - Speed and Stopiing Distance. The Component+Residual plot shows some deviation from a linear relationship.

```
crPlots(cars_model, smooth = list(span=0.75))
```

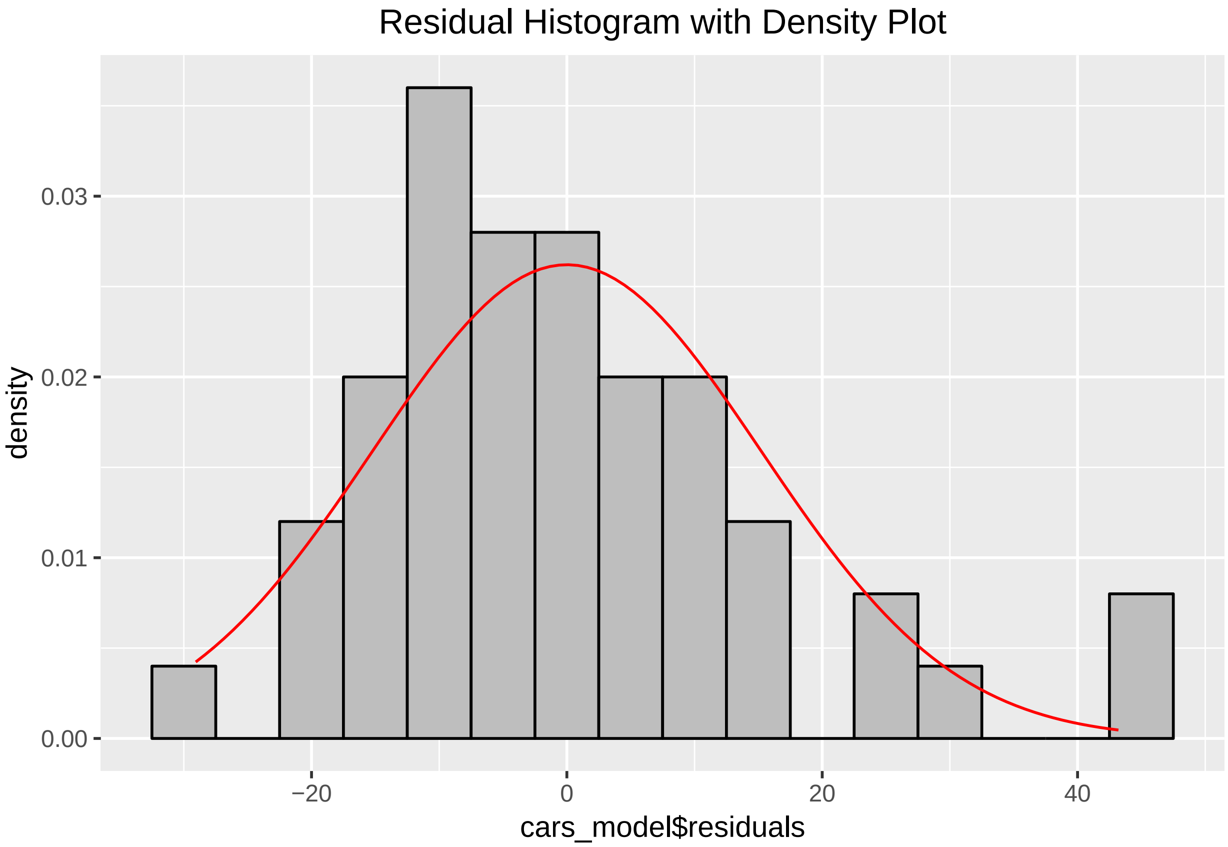


Normality Test

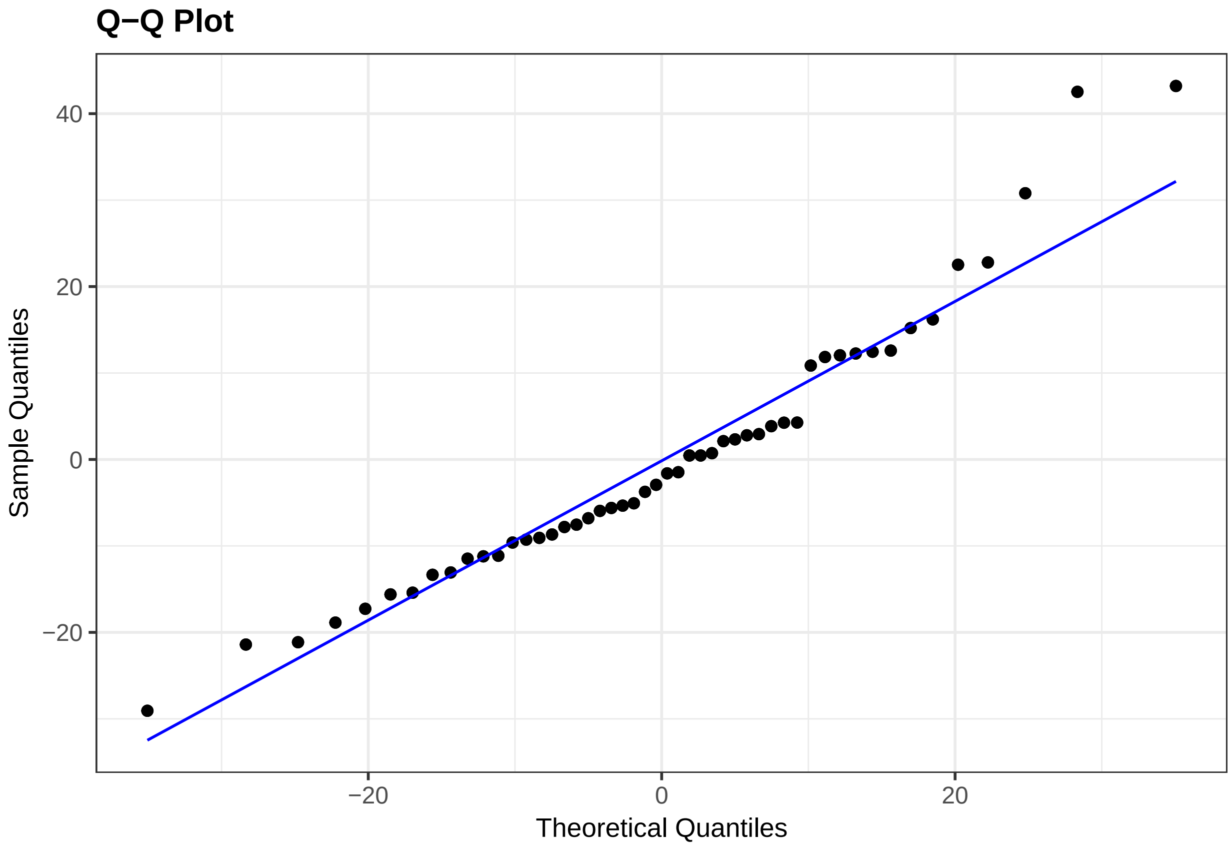
This checks if the residuals of the model follow a Normal Distribution. Another test of the residuals uses the quantile-versus-quantile, or Q-Q, plot. if the model fits the data well, we would expect the residuals to be normally (Gaussian) distributed around a mean of zero.

Per the Residual Histogram and Q-Q plots below, the residuals are not normally distributed. There are some outliers on the left-side of the distribution.

```
## Resudual Histogarm Plot
p <- ggplot(cars_model) +
  geom_histogram(aes(x=cars_model$residuals, y=..density..),
                  binwidth = 5, fill = "grey", color = "black")

xn <- seq(min(cars_model$residuals), max(cars_model$residuals), length.out = 100)
yn <- dnorm(xn, mean(cars_model$residuals), sd(cars_model$residuals))

df <- with(cars_model, data.frame(x=xn, y=yn))

p + geom_line(data = df, aes(x = xn, y = yn), color = "red")+
  ggtitle("Residual Histogram with Density Plot") +
  theme(plot.title = element_text(hjust = 0.5))
```



```
# QQ Plot using resid_panel function from ggResidpanel package
resid_panel(cars_model, plots=c('qq'))
```

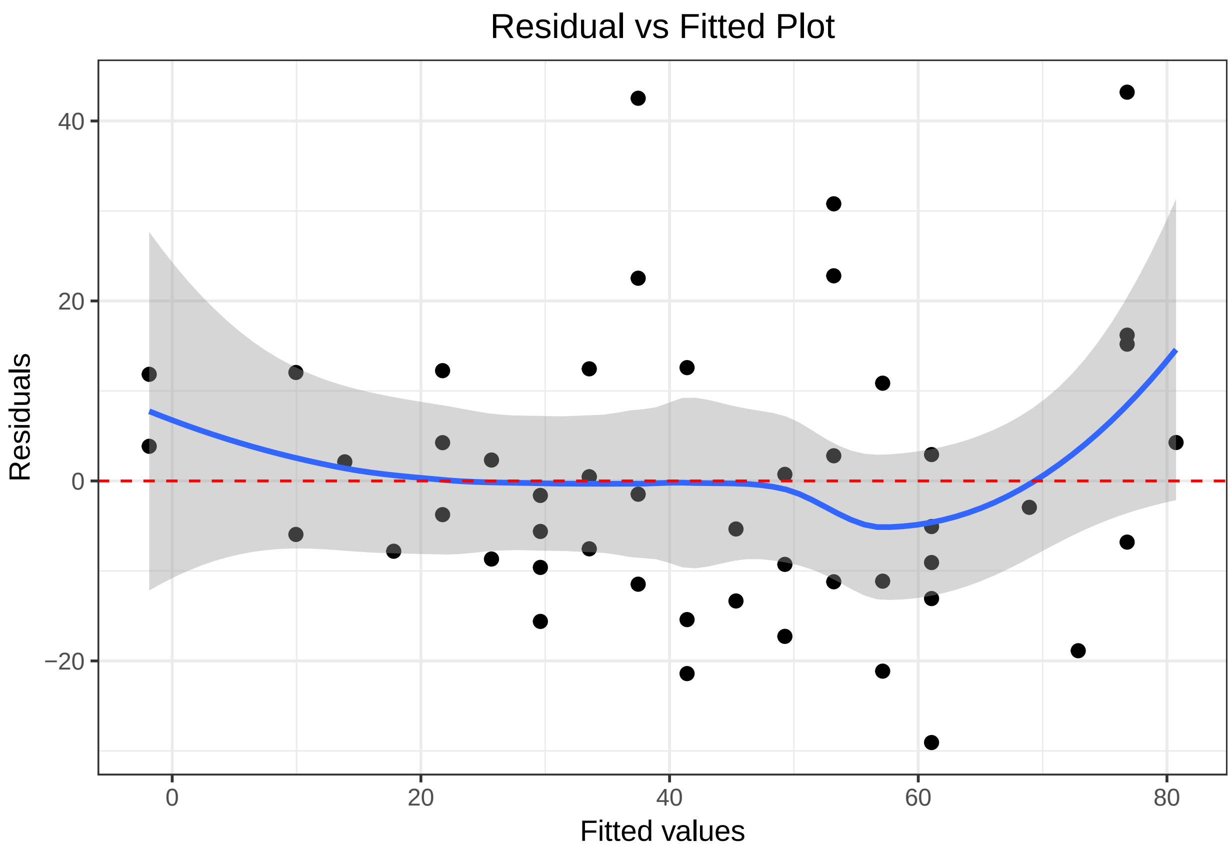


Homoscedasticity Test

This test is conducted to check the constant cariability of residuals.

- Based on the scatter plot, the residuals do show some small deviation in variability.
- The Non-constant Variance Score Test has a p-value of <.05, which means that we reject the null hypothesis of homoscedasticity.

```
# Residual Vs. Fitted Plot
ggplot(cars_model, aes(.fitted, .resid))+
    geom_point(size =2) +
    stat_smooth(method="loess")+
    geom_hline(yintercept=0, col="red", linetype="dashed") +
    xlab("Fitted values")+ylab("Residuals") +
    ggtitle("Residual vs Fitted Plot")+theme_bw() +
    theme(plot.title = element_text(hjust = 0.5))
```



```r
# Non-Constant Variance Test
ncvTest(cars_model)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 4.650233, Df = 1, p = 0.031049
```

Independence Test

This test ensures that data is from a completely random sample and not from a Time Series etc.

The function **durbinWatsonTest()** from car package verifies if the residuals from a linear model are correlated or not:

The null hypothesis (H_0) is that there is no correlation among residuals, i.e., they are independent. The alternative hypothesis (H_a) is that residuals are autocorrelated.

The Durbin Watson test's p-value is >.05. Therefore, we fail to reject the null hypothesis of independence (no autocorrelation).

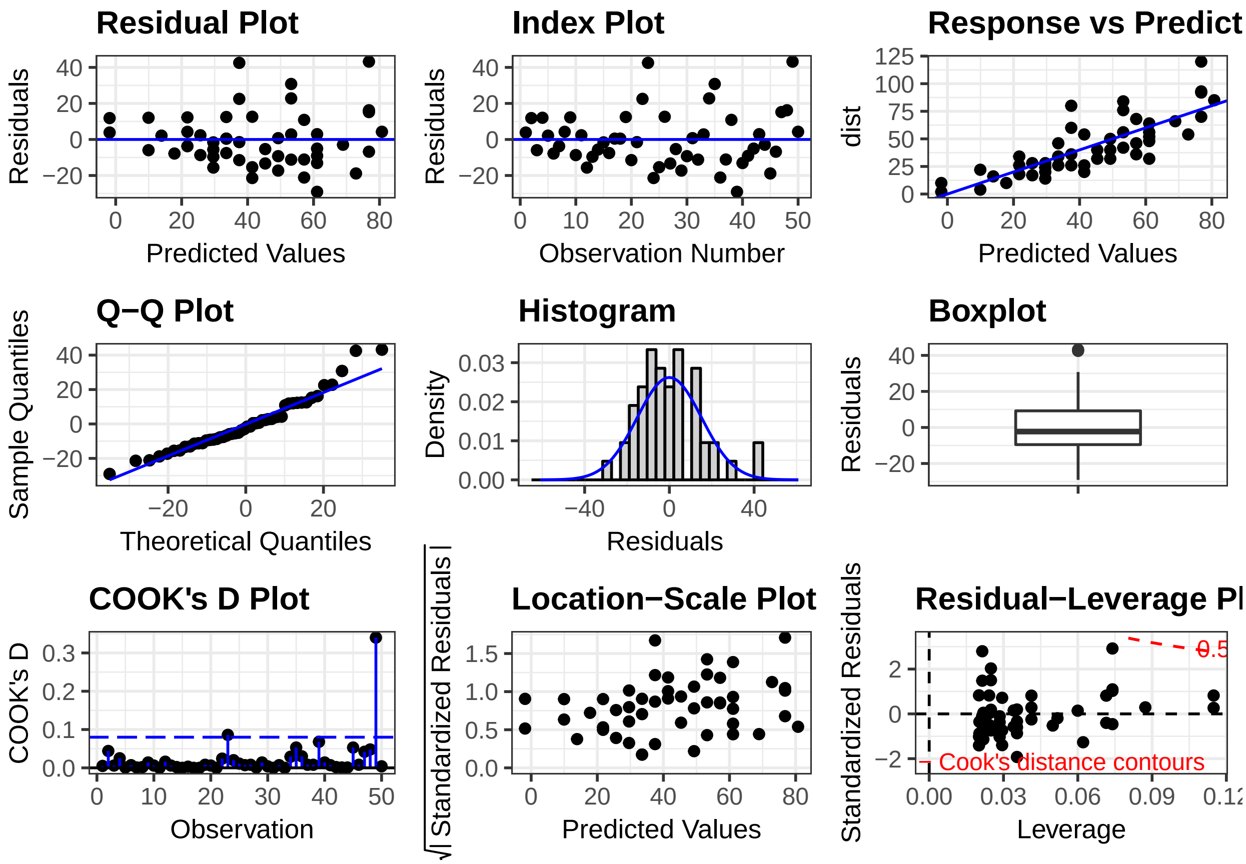
```r
durbinWatsonTest(cars_model)
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1       0.1604322      1.676225    0.14
##  Alternative hypothesis: rho != 0
```

Residial Analysis Summary

Below is a summary of the Residual Analysis using the **resid_panel** function of the **ggResidpanel** package.

```
resid_panel(cars_model, plots='all')
```



Conclusion

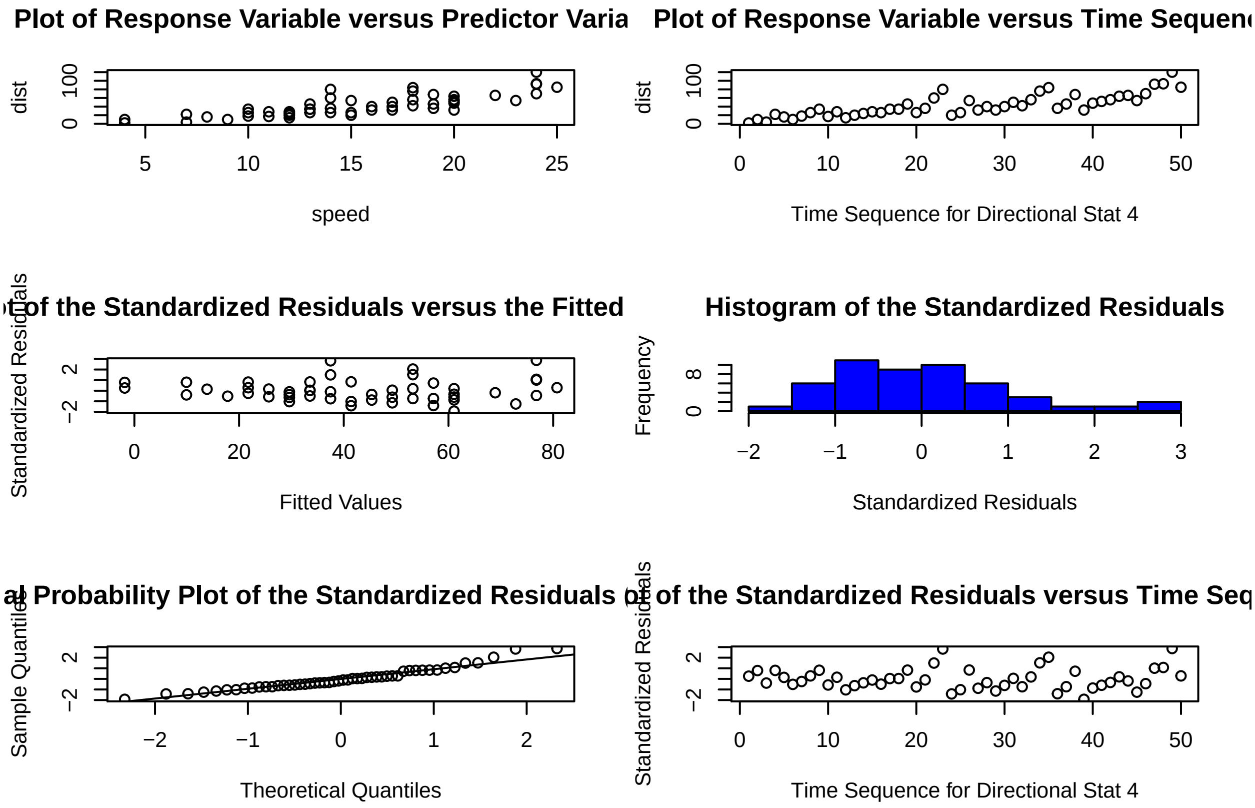
Based upon above Model Diagnostics, it can be concluded that the speed alone is not a very good predictor of stopping distance of a car. Also based on the output of the gvlma function on cars_model and the corresponding plot, we can find that the conditions for linear regression have not been met.

```
gvlma(cars_model)
```

```
##
## Call:
## lm(formula = dist ~ speed, data = cars_df)
##
## Coefficients:
## (Intercept)        speed
##     -17.579        3.932
##
##
## ASSESSMENT OF THE LINEAR MODEL ASSUMPTIONS
## USING THE GLOBAL TEST ON 4 DEGREES-OF-FREEDOM:
## Level of Significance =  0.05
```

```
##
## Call:
##  gvlma(x = cars_model)
##
##                      Value  p-value                   Decision
## Global Stat        15.801 0.003298 Assumptions NOT satisfied!
## Skewness            6.528 0.010621 Assumptions NOT satisfied!
## Kurtosis            1.661 0.197449    Assumptions acceptable.
## Link Function       2.329 0.126998    Assumptions acceptable.
## Heteroscedasticity  5.283 0.021530 Assumptions NOT satisfied!
```

```r
plot(gvlma(cars_model))
```



References:

Global Validation of Linear Model Assumptions

An Introduction to ggResidpanel