# Business Statistics - Graded Assignment

2024-02-18

## Contents

# 1 Cars: Speed and Stopping Distance

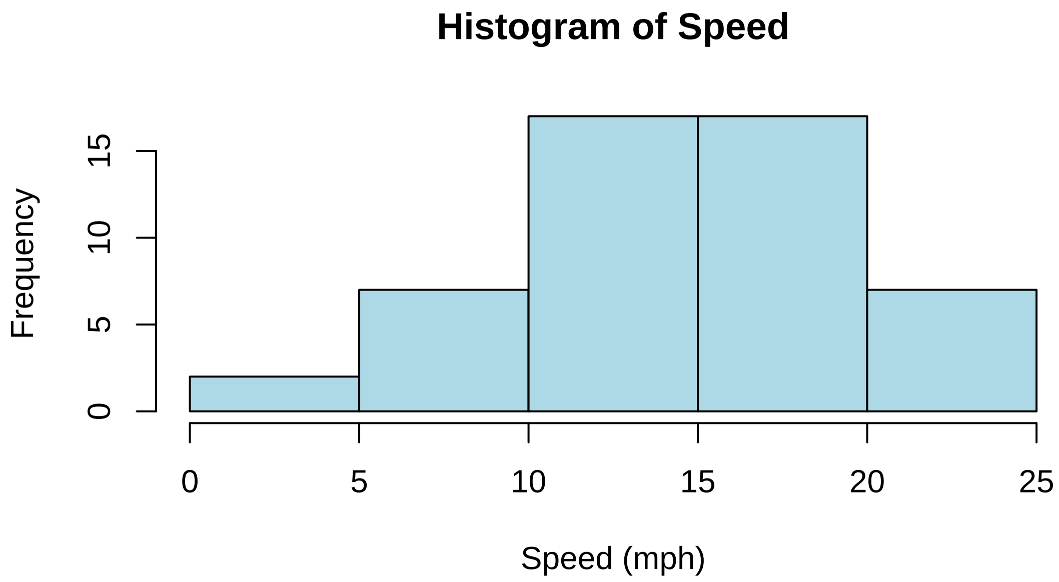
The cars data set provides insights into the speed and stopping distances of cars. It comprises a data frame with 50 rows and 2 variables. The rows correspond to individual cars, while the variables represent speed (speed in mph) and dist (stopping distance in feet). A summary of the first 5 rows of the data set is presented below (R code 7.1):

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

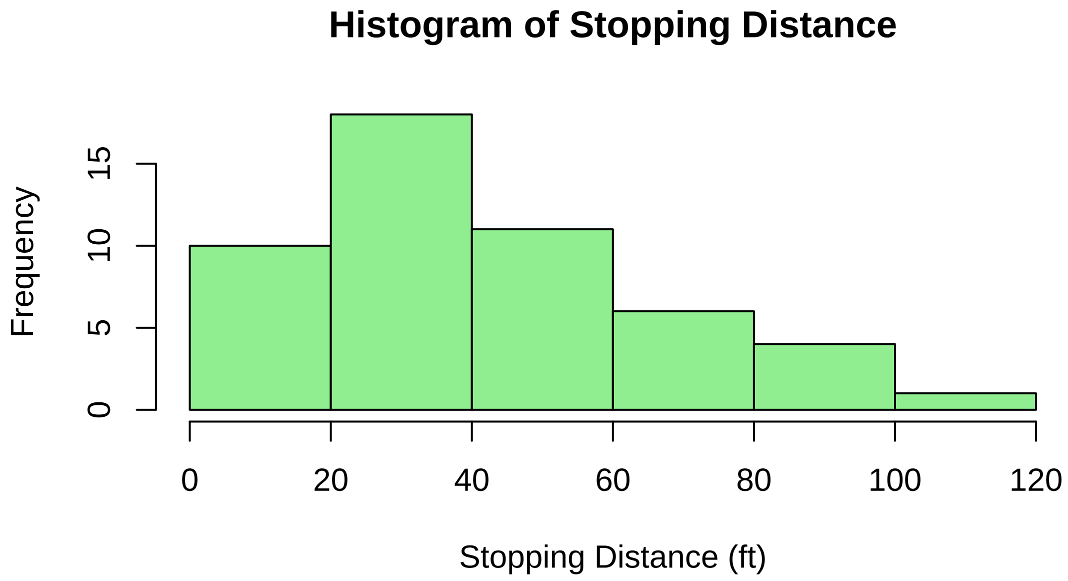
## 1.1 Frequency Distribution

To discern the essential characteristics of the data, we employ histograms to visualise the frequency distribution.

The speed histogram (R code 7.2) appears approximately normally distributed, with a slight left skew (-0.11). The distribution's tail extends towards lower speeds, indicating a prevalence of higher-speed cars. Despite the slight left skew, the mean speed (15.4 mph) closely aligns with the median stopping distance (15 mph). When the distribution is left-skewed, the mean is generally pulled towards the tail because extreme values on the left side of the distribution (lower values) have a more significant influence on the mean than on the median. However, as the speed distribution is slightly left-skewed, the slightly higher mean speed compared to the median speed can be explained by the small size of the dataset and the influence of outliers. However, the distribution's kurtosis of 2.42 suggests a platykurtic distribution, indicating a lower concentration of values around the mean compared to a normal distribution (R code 7.3).



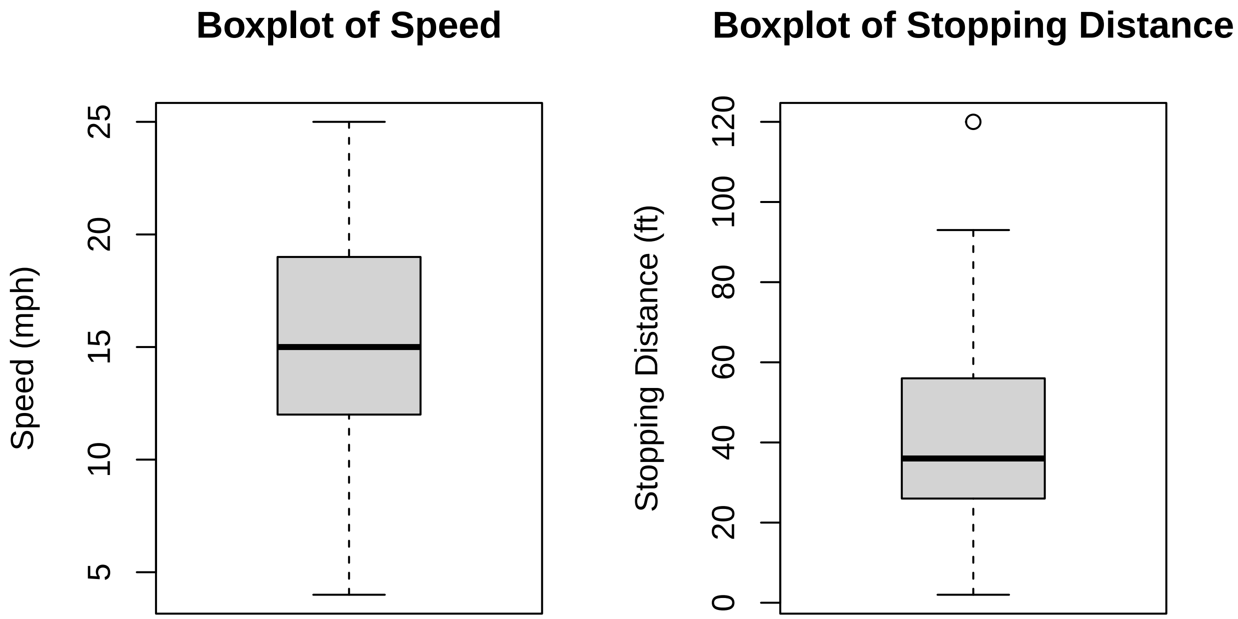
Conversely, the histogram portraying stopping distances (R code 7.4) exhibits a right-skewed distribution (0.78), indicating that most cars have shorter to medium-range stopping distances, with some outliers having longer stopping distances. Here, the mean stopping distance (42.98 ft) surpasses the median stopping distance (36 ft), indicating a positively skewed distribution. The kurtosis value of 3.25 suggests a leptokurtic distribution, indicating a higher concentration of values around the mean than a normal distribution (R code 7.5).



## 1.2 Measures of Position

Box plots can provide a comprehensive overview of the spread of speed and stopping distances among cars (R code 7.6). The first box plot below represents the distribution of car speeds in the data set. The speed's interquartile range (IQR) is 7 mph, with the median indicated by the horizontal line inside the box (15 mph). The "whiskers" extend to the minimum and maximum values within 1.5 times the IQR from the lower and upper quartiles, respectively. Any points outside the whiskers are considered outliers - the absence of outliers indicates a relatively consistent speed distribution among cars.

Like the speed box plot, the stopping distance plot represents the interquartile range (IQR) of 30 ft, with a median stopping distance of 36 ft. The whiskers extend to the minimum and maximum values within 1.5 times the IQR from the lower and upper quartiles, respectively. The presence of an outlier, notably a maximum stopping distance of 120 ft, which lies beyond 1.5 times the IQR from the upper quartile of 101 ft, suggests variability among cars in this regard.



### 1.3 Relationships Between Variables

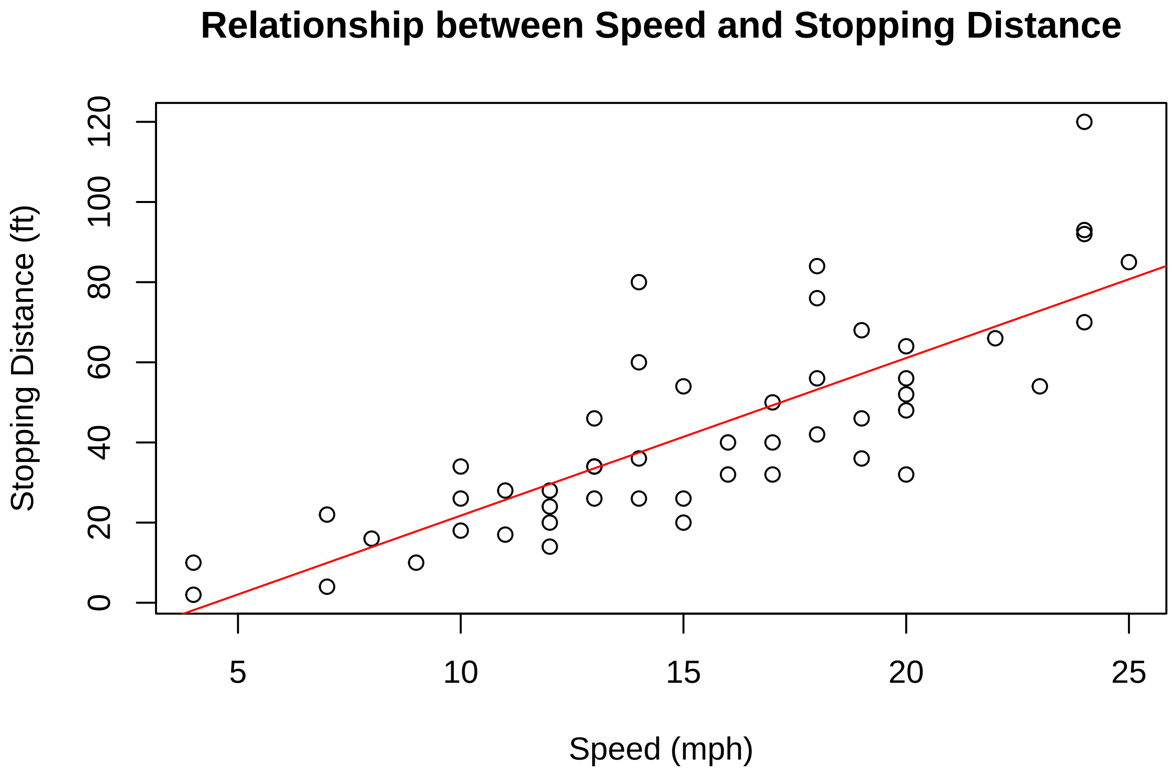
When exploring relationships between two variables, calculating their correlation is a common step in data analysis. When first examining a data set, calculating correlation coefficients can provide insights into potential relationships. This step can help us identify if speed and stopping distance are related and can guide further analysis (R code 7.7).

```
The correlation coefficient between car speed and stopping distance is 0.81
```

As a correlation coefficient quantifies the strength and direction of the linear relationship between two variables, in this case, we can conclude that the correlation coefficient between speed and stopping distance of 0.81 indicates a strong positive linear relationship between the two variables. This suggests that as speed increases, stopping distance also tends to increase.

We can confidently say that the two variables have a strong positive linear association. Knowing the value of one variable allows us to make reasonably accurate predictions about the other variable in many cases. However, it is essential to consider other factors, such as the context of the data, potential outliers, and the possibility of confounding variables when interpreting and making predictions based on correlation coefficients. However, it is essential to remember that correlation does not imply causation - even though the variables are strongly correlated, it does not necessarily mean that one variable causes the other to change.

Subsequently, we fit a simple linear regression model of distance on speed, revealing a relationship between speed and stopping distance (R code 7.8). When plotted on a scatter plot, the data points would tend to fall closely around a straight line sloping upwards from left to right (R code 7.9).



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

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
```

This regression model aims to predict cars’ stopping distance (`dist`) based on their speed (`speed`). From the above output, we can conclude the following:

**Interpretation of Coefficients**:

- **Intercept**: The intercept coefficient (-17.5791) represents the estimated stopping distance when the car’s speed is zero. In this context, it doesn’t have a practical interpretation since vehicles typically don't have a stopping distance when stationary. However, it's included in the model to allow for a more accurate estimation of the relationship between speed and stopping distance.

- **Speed**: The coefficient for speed (3.9324) is the slope that indicates that for every unit increase in speed (in miles per hour), the stopping distance is estimated to increase by approximately 3.9324 feet.

The equation for this linear regression model can be expressed as:

$$\text{dist} = \beta_0 + \beta_1 \times \text{speed}$$

where:

- dist is the predicted stopping distance.
- $\beta_0$ is the intercept, estimated to be $-17.5791$.
- $\beta_1$ is the coefficient for the speed variable, estimated to be 3.9324.
- speed is the speed of the car.

So, substituting the estimated values into the equation, we get:

$$\text{dist} = -17.5791 + 3.9324 \times \text{speed}$$

This equation can be used to predict the stopping distance of a car based on its speed, evaluate how speed changes affect stopping distance, and compare the predicted stopping distances obtained from the model with the actual stopping distances observed in your data set to assess how well the model fits the data and make decisions based on predictions. When using a regression model, it is essential to be cautious about extrapolating beyond the range of the data used to create the model. Extrapolation involves making predictions outside this range, where the model's reliability can significantly decrease, as it is based on the assumption that the established relationship continues in the same manner, which may not hold true in unobserved regions.

**Standard Errors and t-values**:

- The standard error measures the variability of the coefficient estimate. Lower standard errors indicate more precise estimates. In this case, we can conclude that the required stooping distance can vary by 0.4155 ft.
- The t-values measure the significance of each coefficient. In this case, both coefficients have relatively high t-values with p-values less than 0.05, indicating they are statistically significant. Hence, we can reject the null hypothesis and conclude that there is a relationship between speed and stopping distance.

**Significance Levels**:

- The significance codes (`***, **, *`) indicate each coefficient's significance level. Here, both the intercept and the speed coefficient have high significance levels (`***`), suggesting that they are highly likely to differ from zero.

**Model Fit**:

- **Multiple R-squared**: The value of 0.6511 indicates the proportion of variability in the dependent variable (stopping distance) that is explained by the independent variable (speed). In this case, about 65.11% of the variability in stopping distance is explained by the car's speed.

- **Adjusted R-squared**: This is the R-squared value adjusted for the number of predictors in the model. It penalises the addition of unnecessary predictors. Here, the adjusted R-squared value is 0.6438.
- **Residual Standard Error**: It estimates the standard deviation of the errors in the regression model. It measures the typical distance between the observed values and the values predicted by the model. In this case, the actual distance required to stop can deviate from the true regression line by approximately 15.38 feet.

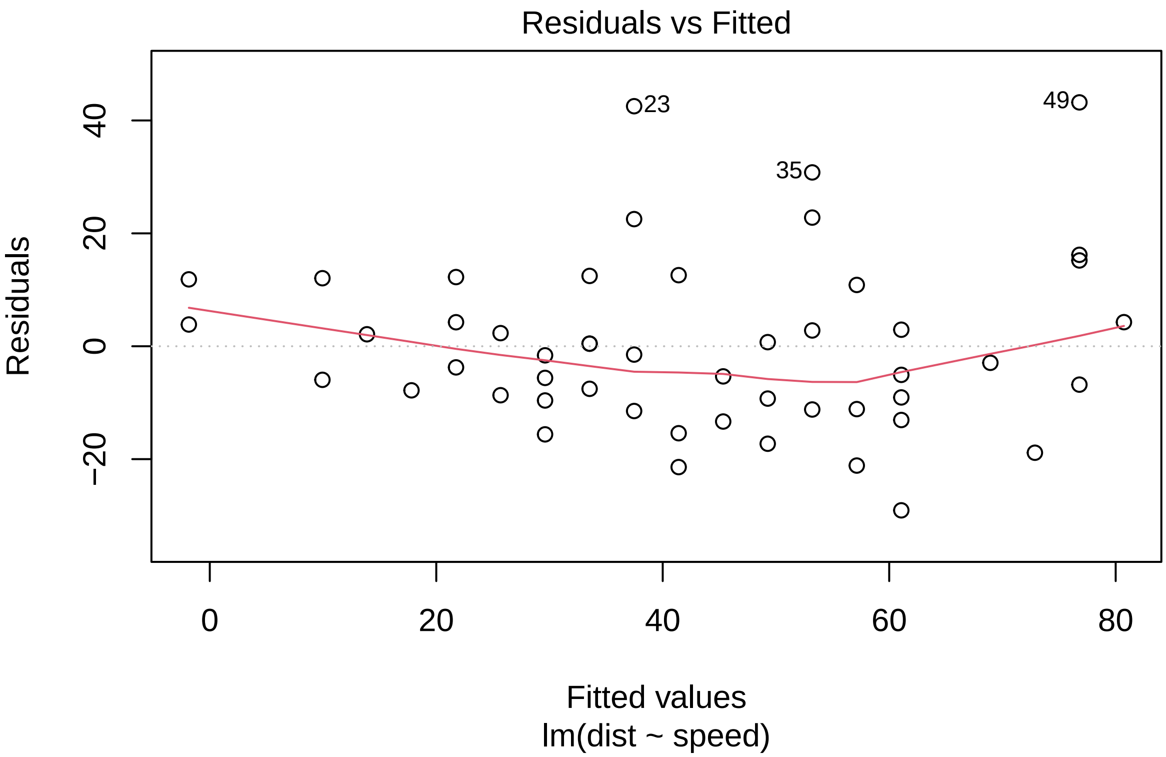
**F-statistic and p-value**:

- The F-statistic tests the overall significance of the regression model. Here, the F-statistic is 89.57 with a very low p-value (less than 0.05, indicating that the regression model as a whole is statistically significant.

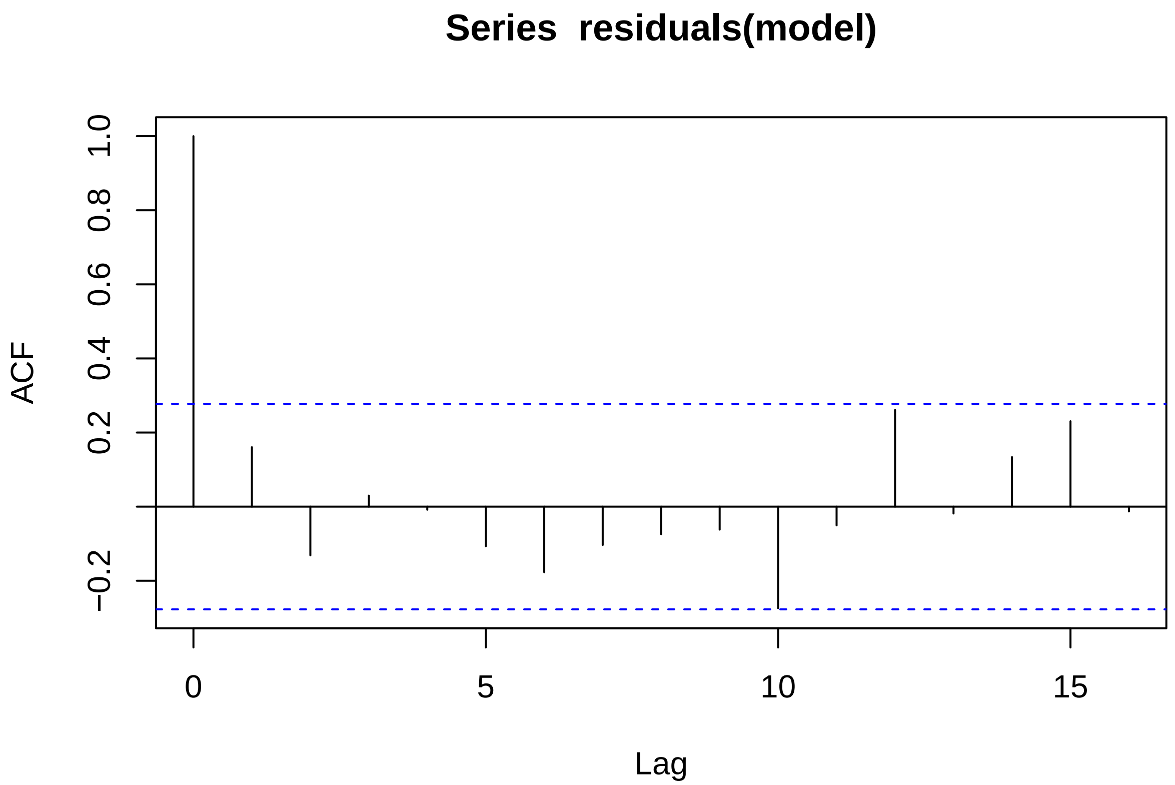
## 1.4 Assessment of Linear Regression Assumptions

The validity of the regression model is assessed through four assumptions:

**Linearity**: The relationship between the independent variable (speed) and the dependent variable (distance) should be linear. The `Residuals vs Fitted` scatter plot indicates a roughly linear relationship between speed and stopping distance, validating this assumption (R code 7.10).



**Independence of Errors**: The residuals should be independent of each other. This means that the error term for one observation should not be correlated with the error term for another observation. This assumption is often checked by examining residual plots or conducting autocorrelation tests. On the Series Residuals plot produced using `acf()` on residuals, we can see that most bars are within the boundary and are close to zero, suggesting that there is no autocorrelation, which indicates a good fit (R code 7.11). However, the first bar crosses the blue line, which means the autocorrelation at that lag is statistically significant.



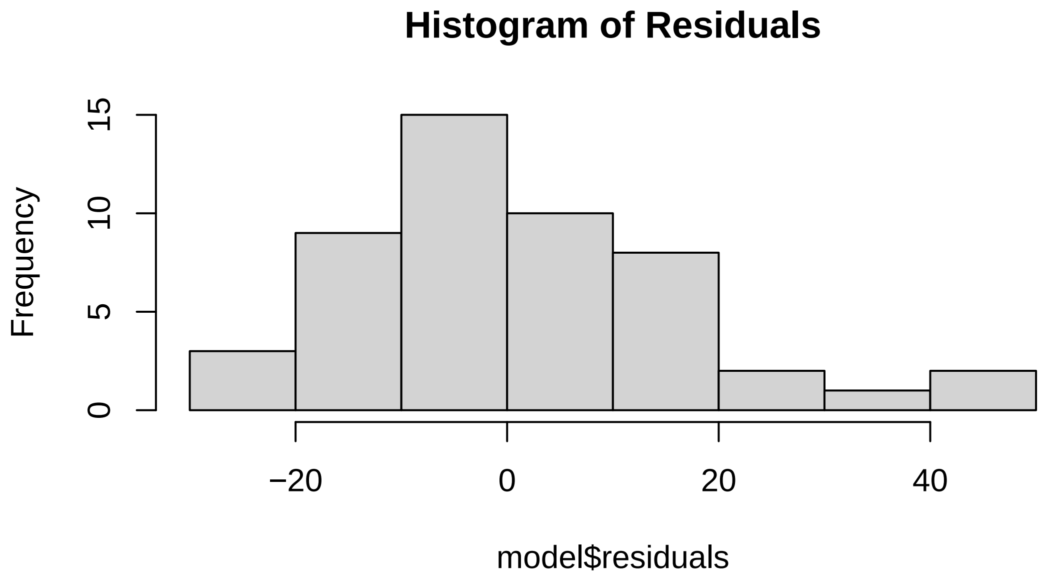
To further assess the independence assumption, we will run the Durbin-Watson test used to detect the presence of autocorrelation in the residuals (R code 7.12):

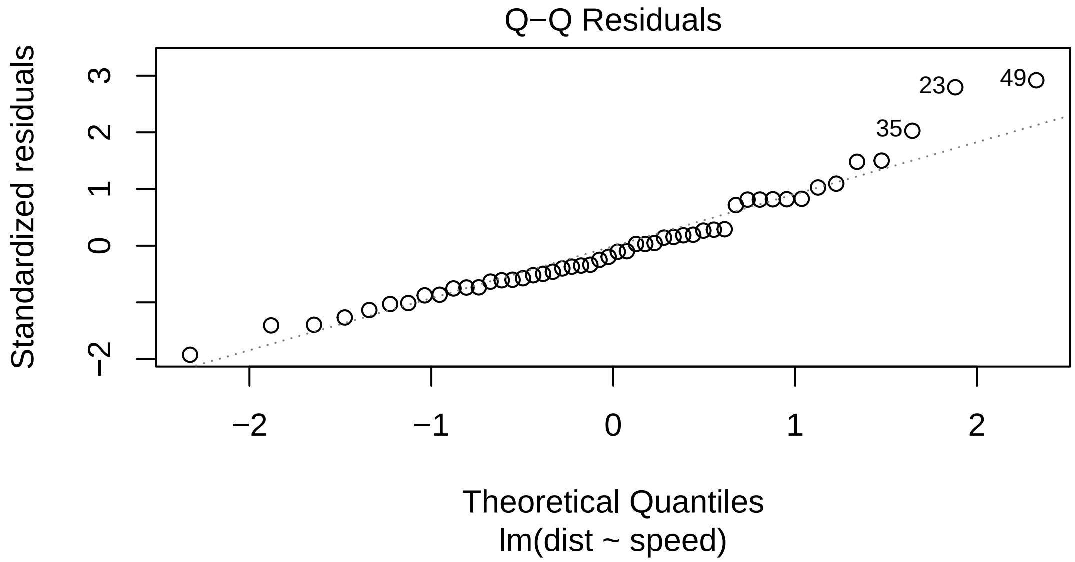
```
	Durbin-Watson test

data:  model
DW = 1.6762, p-value = 0.09522
alternative hypothesis: true autocorrelation is greater than 0
```

The Durbin-Watson statistic of 1.6762 is somewhat below 2, which might suggest the presence of slight positive autocorrelation. However, since the p-value is above this threshold, it suggests that there is not enough statistical evidence to reject the null hypothesis of no autocorrelation at the 5% significance level.

**Normality of Errors**: The residuals should follow a normal distribution. The normality of errors is typically assessed by examining a histogram or a Q-Q plot of the residuals. As the cars data set is too small, it isn't easy to evaluate normality as potential outliers can impact the normal distribution. We can see on the histogram that the data is slightly right-skewed, although it does not appear to depart substantially from a normal distribution (R code 7.13). However, on the Q-Q plot, the majority of the points fall approximately along the reference line, and the endpoints deviate from the straight line, suggesting a heavy-tailed distribution (R code 7.14).





To further assess the normality assumption, we will run the Shapiro-Wilk Normality test (R code 7.15) as the Kolmogorov-Smirnov test's sensitivity varies with the sample size, and it may not detect minor deviations from normality with smaller samples:

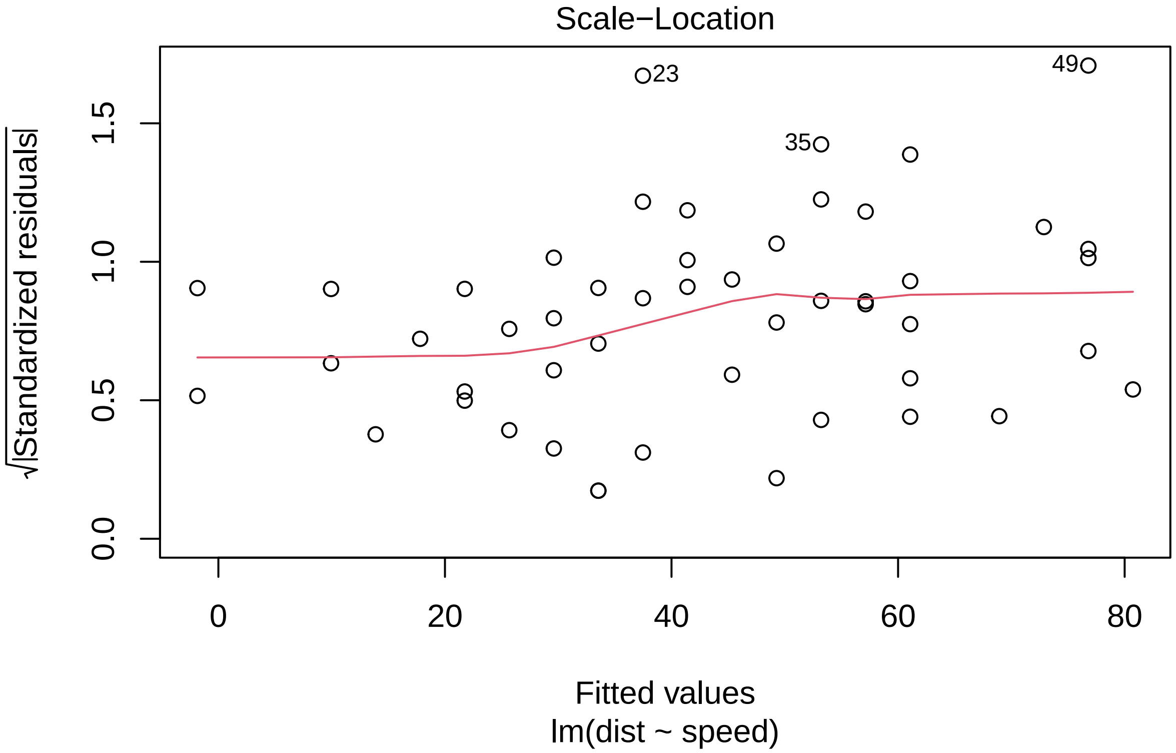
```

	Shapiro-Wilk normality test

data:  model$residuals
W = 0.94509, p-value = 0.02152
```

The W value of 0.94509 suggests that the residuals are fairly close to normal but not perfectly. However, the p-value of 0.02152 suggests that there is statistically significant evidence to reject the null hypothesis of normality at the 5% significance level.

**Equal Variance (Homoscedasticity)**: The variance of the residuals should be constant across all levels of the independent variable. This assumption is important because heteroscedasticity (where the variance of errors differs across levels of the independent variable) can lead to inefficient and biased estimators. The red line in the Scale-Location plot exhibits a slight deviation from horizontal alignment (R code 7.16). There are no apparent indications of heteroscedasticity, as evidenced by the absence of significant deviations from the horizontal line and the lack of a funnel shape in errors, where larger fitted values correspond to larger errors.



We can also conduct the Breusch-Pagan test (R code 7.17) to validate the absence of heteroscedasticity:

```
	studentized Breusch-Pagan test

data:  model
BP = 3.2149, df = 1, p-value = 0.07297
```

Since the BP statistic's p-value is above the common threshold for statistical significance of 0.05, there is not enough evidence to reject the null hypothesis of homoscedasticity. Based on this test, we do not have sufficient evidence to conclude that heteroscedasticity is present in this regression model. Based on the results of the tests and the diagnostic plots, we can see that the assumptions of linearity, homoscedasticity, and independence are met, while the Shapiro-Wilk test for normality rejected the null hypothesis, indicating that the residuals (errors) of the regression model are not normally distributed. However, linear regression can be relatively robust to minor departures from normality, especially when other assumptions are met, and therefore, they may not invalidate the regression analysis results (Schmidt and Finan, 2018).

### 1.5 Fastest and Slowest Cars

To find the minimum speed required to belong to the fastest 10% in the data set, assuming it follows a normal distribution with a mean of 40.0 mph and a standard deviation of 12.1 mph, we need to find the speed value corresponding to the 90th percentile. We can use the qnorm() function in R to calculate it (R code 7.18):

```
Minimum speed required to belong to the fastest 10%: 55.51 mph
```

To find the percentage of cars slower than 30 mph, we calculate the cumulative probability of the normal distribution up to the value of 30 mph (R code 7.19):

```
Percentage of cars slower than 30 mph: 20.43 %
```

From the above outputs, we can conclude that the minimum speed required to belong to the fastest 10% of cars in the data set is 55.51 mph, while the percentage of cars slower than 30 mph is 20.43%.

## 2 Marketing Data: Multiple Regression Models

### 2.1 Correlation Analysis

The `marketing` data set from the `datarium` package consists of four variables and 200 records: advertising expenditure in three media (YouTube, Facebook, and Newspaper) and sales (which is the dependent variable). A summary of the first 5 rows of the data set is presented below (R code 7.20):

```
  youtube facebook newspaper sales
1  276.12    45.36     83.04 26.52
2   53.40    47.16     54.12 12.48
3   20.64    55.08     83.16 11.16
4  181.80    49.56     70.20 22.20
5  216.96    12.96     70.08 15.48
```

Exploring the relationships among variables involves computing correlation coefficients, a standard procedure in data analysis. Therefore, as the first step of the analysis, the `cor()` function is executed to detect a correlation between the variables (R code 7.21):

```
             youtube   facebook  newspaper     sales
youtube   1.00000000 0.05480866 0.05664787 0.7822244
facebook  0.05480866 1.00000000 0.35410375 0.5762226
newspaper 0.05664787 0.35410375 1.00000000 0.2282990
sales     0.78222442 0.57622257 0.22829903 1.0000000
```

Correlation coefficients closer to 1 (positive or negative) indicate stronger relationships, while values closer to 0 indicate weaker ones. In this data set, a strong positive correlation (0.78) exists between YouTube advertising expenditure and sales, suggesting that increases in YouTube advertising are associated with sales increases. The correlation between Facebook advertising and sales is moderate (0.58), implying a less pronounced but still significant positive relationship. However, Newspaper advertising shows a weak positive correlation (0.23) with sales, indicating a lesser impact than digital platforms. Low correlations between different advertising media suggest little multicollinearity, indicating each medium contributes independently to sales.

## 2.2 Exploration of Regression Models

We split the marketing data set into two parts to fit and validate potential linear regression models. The first part, named in-sample, contains the first 150 records and will be used for model training and analysis (R code 7.22). The second part, out-of-sample, consists of the last 50 records and will be used for testing and validating the model's performance on unseen data. Splitting a data set into training and testing sets is crucial for accurately evaluating a model's performance. The training set is used to fit the model, allowing it to learn the underlying patterns in the data. The testing set, which the model has not seen during training, is then used to assess how well the model generalises to new, unseen data. This approach helps mitigate overfitting, ensuring that the model's predictions are robust and reliable when applied to real-world data outside the training data set.

Since we previously explored the correlation between YouTube, Facebook, and newspapers and the sales, we fit three different linear regression models to predict sales using the in-sample data. Each model aims to understand how different combinations of advertising channels impact sales, allowing for the evaluation of their individual and combined effectiveness.

`Model 1` uses YouTube, Facebook and newspaper expenditures as predictors (R code 7.23):

```
Call:
lm(formula = sales ~ youtube + facebook + newspaper, data = in_sample)

Residuals:
     Min       1Q   Median       3Q      Max
-10.2910  -0.8056   0.4106   1.4622   3.2429

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  3.635720   0.444876   8.172 1.32e-13 ***
youtube      0.047000   0.001637  28.713  < 2e-16 ***
facebook     0.179933   0.010282  17.500  < 2e-16 ***
newspaper   -0.001403   0.006752  -0.208    0.836
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

Residual standard error: 2.058 on 146 degrees of freedom
Multiple R-squared:  0.8958,    Adjusted R-squared:  0.8936
F-statistic: 418.2 on 3 and 146 DF,  p-value: < 2.2e-16
```

`Model 2` uses only YouTube and Facebook, excluding newspapers (R code 7.24):

```
Call:
lm(formula = sales ~ youtube + facebook, data = in_sample)

Residuals:
     Min       1Q   Median       3Q      Max
-10.2374  -0.7938   0.4306   1.4357   3.2194

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 3.605173   0.418504   8.614 1.01e-14 ***
youtube     0.046998   0.001632  28.806  < 2e-16 ***
```

```
facebook    0.179140   0.009516  18.825  < 2e-16 ***
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

Residual standard error: 2.052 on 147 degrees of freedom
Multiple R-squared:  0.8957,    Adjusted R-squared:  0.8943
F-statistic: 631.4 on 2 and 147 DF,  p-value: < 2.2e-16
```

`Model 3` is the simplest, using only YouTube expenditure as the predictor (R code 7.25):

```
Call:
lm(formula = sales ~ youtube, data = in_sample)

Residuals:
     Min       1Q   Median       3Q      Max
-10.2880  -2.1005   0.1797   2.2536   8.1968

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 8.532879   0.601011   14.20   <2e-16 ***
youtube     0.049063   0.002996   16.38   <2e-16 ***
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

Residual standard error: 3.776 on 148 degrees of freedom
Multiple R-squared:  0.6444,    Adjusted R-squared:  0.642
F-statistic: 268.1 on 1 and 148 DF,  p-value: < 2.2e-16
```

As we see from the above outputs, `Model 2`, using YouTube and Facebook as predictors, is the best choice based on the results. It has a very high R-squared value of 0.8957 (indicating that the model explains 89.58% of the variance in sales) similar to `Model 1`. However, the coefficient for the newspaper in `Model 1` is not statistically significant, suggesting it may not contribute meaningfully to sales predictions in this model. The exclusion of newspapers from `Model 2` is justified as its coefficient in `Model 1` is not statistically significant (p-value: 0.836), indicating little to no contribution to sales prediction.

As Newspaper advertising shows a weak positive correlation (0.23) with sales, we can use `vif()` function from the `car` package to formally check for multicollinearity in `Model 1` (R code 7.26).

```
  youtube  facebook newspaper
 1.004566  1.165088  1.160856
```

In this case, multicollinearity is not an issue (VIF<5), so we can proceed by removing the newspapers variable, which is non-significant. `Model 3`, with only YouTube as a predictor, has a significantly lower R-squared value (0.6444), indicating a less accurate fit than `Model 2`.

To ensure we have selected the correct model, we can apply the stepwise selection with both directions to the marketing data, including all variables, which removes variables that do not significantly improve the model fit (R code 7.27):

```
Start:  AIC=220.51
sales ~ youtube + facebook + newspaper
```

```
            Df Sum of Sq    RSS    AIC
- newspaper  1       0.2  618.7 218.56
<none>                    618.6 220.51
- facebook   1    1297.5 1916.0 388.11
- youtube    1    3492.8 4111.3 502.63

Step:  AIC=218.56
sales ~ youtube + facebook

            Df Sum of Sq    RSS    AIC
<none>                    618.7 218.56
+ newspaper  1       0.2  618.6 220.51
- facebook   1    1491.6 2110.3 400.60
- youtube    1    3492.6 4111.4 500.63


Call:
lm(formula = sales ~ youtube + facebook, data = in_sample)

Residuals:
     Min       1Q   Median       3Q      Max
-10.2374  -0.7938   0.4306   1.4357   3.2194

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 3.605173   0.418504   8.614 1.01e-14 ***
youtube     0.046998   0.001632  28.806  < 2e-16 ***
facebook    0.179140   0.009516  18.825  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.052 on 147 degrees of freedom
Multiple R-squared:  0.8957,    Adjusted R-squared:  0.8943
F-statistic: 631.4 on 2 and 147 DF,  p-value: < 2.2e-16
```

Based on preliminary analyses, `Model 2`, using YouTube and Facebook predictors, is selected for its simplicity and efficiency without compromising explanatory power.

The equation for this linear regression model can be expressed as:

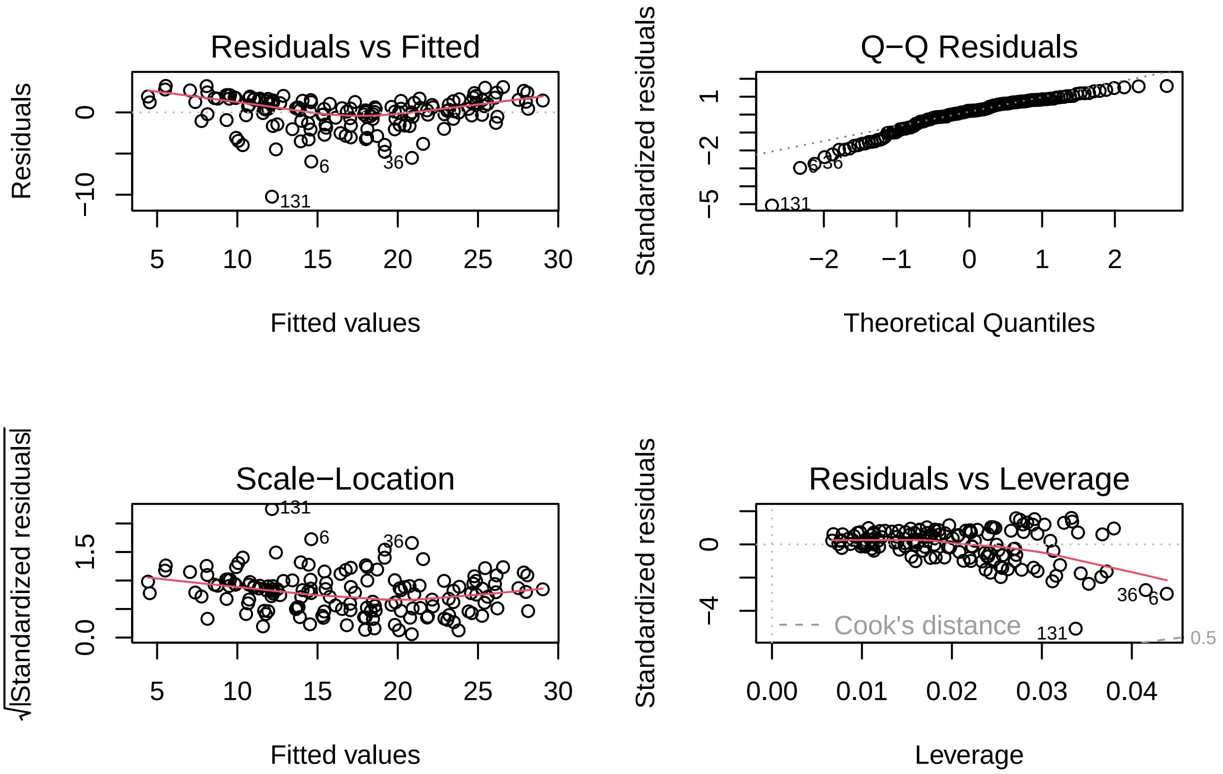
$$\text{sales} = 3.605173 + 0.046998 \times \text{youtube} + 0.179140 \times \text{facebook}$$

Where:

- sales is the predicted sales.
- youtube is the number of YouTube advertising expenditure.
- facebook is the number of Facebook advertising expenditure.

This equation indicates that for each unit increase in YouTube advertising, sales are expected to increase by approximately 0.046998 units. For each unit increase in advertising on Facebook, sales are expected to increase by approximately 0.179140 units. The intercept term represents the expected sales when both YouTube and Facebook advertising expenditures are zero.

However, further diagnostics are necessary to validate the model's assumptions (linearity, independence of errors, normality, and equal variance of errors) and performance (R code 7.28).



The residuals on the `Residuals vs Fitted` plot are mostly randomly scattered around the horizontal line at 0, which is good as it suggests no obvious problems with linearity. However, there seems to be a slight pattern with residuals being more positive at the lower end of fitted values and more negative at the higher end, which could indicate a non-linear relationship that the linear model is not capturing perfectly. There are no clear outliers, which is positive.

The points on the `Q-Q Residuals` plot follow the diagonal line closely, particularly in the centre of the distribution, suggesting that the residuals are approximately normally distributed. However, there are deviations in the tails, particularly the upper tail, which may indicate heavier tails than expected under normality, suggesting the presence of outliers or extreme values. These deviations could be due to left-skewness in the data. Depending on the significance of this deviation from normality and the specific statistical analysis required, we may need to consider data transformation techniques.

The spread of the standardised residuals on the `Scale-Location` plot seems relatively constant across the range of fitted values, which is a good sign of homoscedasticity. No clear pattern indicates increasing or decreasing variance, which suggests that the assumption of equal variance (homoscedasticity) is reasonable for this model.

Most data points on the `Residuals vs. Leverage` plot have low leverage, meaning no single observation unduly influences the model's predictions. However, a few points with higher leverage may warrant further investigation. The Cook's distance lines do not show points that are influential enough to be concerning.

To verify the normality of the model, we will run the Shapiro-Wilk test (R code 7.29):

```
    Shapiro-Wilk normality test
```

```
data:  model2$residuals
W = 0.90479, p-value = 2.486e-08
```

In this case, with such a small p-value, we would typically reject the null hypothesis and conclude that the residuals are not normally distributed. This has implications for the assumptions of the statistical model, indicating that it may not be the best fit for the data or that further investigation is needed.

Overall, the model seems to perform adequately, but there are indications that it could be improved. The slight pattern in the residuals and the deviations in the tails of the `Q-Q` plot suggest that exploring non-linear transformations or adding interaction terms could refine the model.

Considering that we detected some potential issues with the selected model, we will enhance it by including both a linear term for YouTube advertising and its square root term to capture potential non-linear effects of YouTube expenditures on sales (R code 7.30). This model can be helpful if we suspect that simply increasing the advertising budget does not lead to a constant increase in sales and that a more complex relationship might need to be captured. Including both linear and square root terms for YouTube advertising makes the model more flexible in fitting the actual data.[1]

```
Call:
lm(formula = sales ~ youtube + sqrt(youtube) + facebook, data = in_sample)

Residuals:
    Min      1Q  Median      3Q     Max 
-4.8161 -1.0118  0.0436  0.9064  4.4335 

Coefficients:
               Estimate Std. Error t value Pr(>|t|)    
(Intercept)   -3.522093   0.856221  -4.114 6.48e-05 ***
youtube       -0.013881   0.006853  -2.026   0.0446 *  
sqrt(youtube)  1.406386   0.155392   9.051 8.26e-16 ***
facebook       0.188992   0.007720  24.482  < 2e-16 ***
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

Residual standard error: 1.648 on 146 degrees of freedom
Multiple R-squared:  0.9332,	Adjusted R-squared:  0.9318 
F-statistic: 679.9 on 3 and 146 DF,  p-value: < 2.2e-16
```

The equation for the model can be expressed as:

$$\text{sales} = -3.522093 - 0.013881 \times \text{youtube} + 1.406386 \times \sqrt{\text{youtube}} + 0.188992 \times \text{facebook}$$

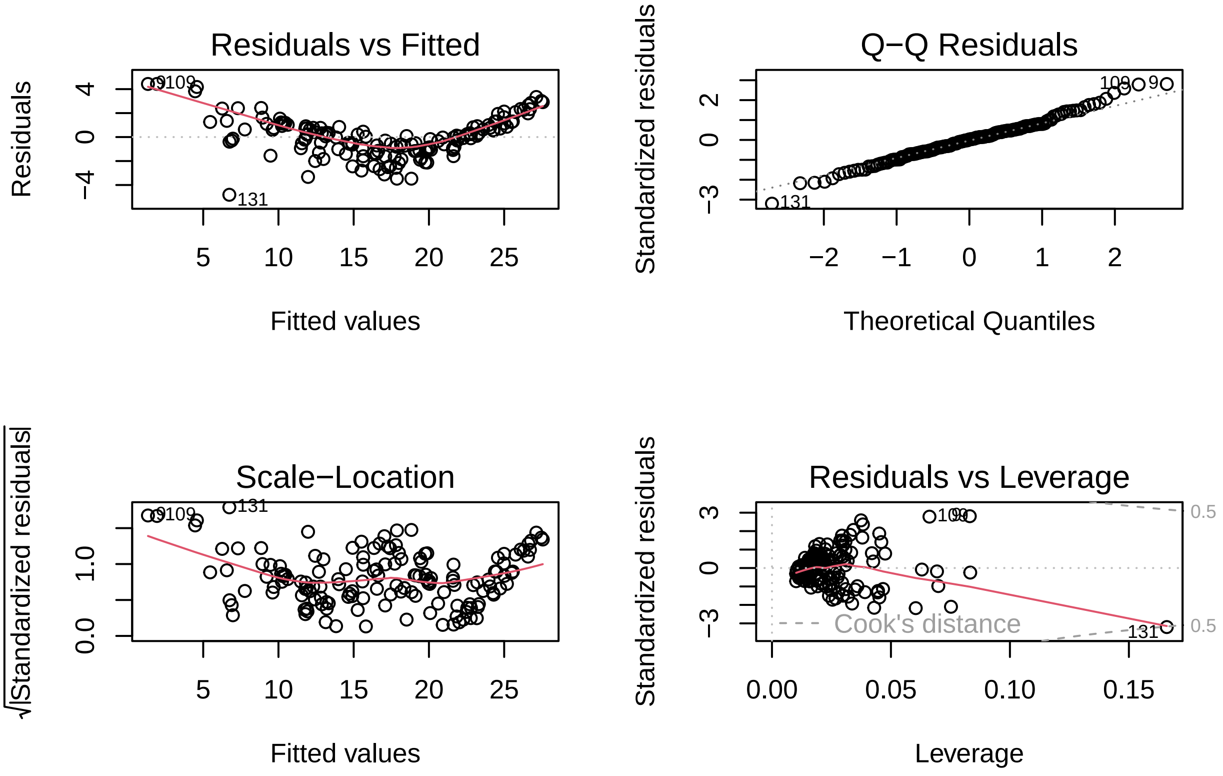
Where:

- sales is the predicted sales.
- youtube is the advertising spending on YouTube.
- facebook is the advertising spending on Facebook.

[1] Adding a square root term for Facebook advertising to capture potential non-linear effects of Facebook expenditures on sales was separately explored, and while it may have exhibited statistical significance, its negligible impact on the multiple R-squared value and the principle of model parsimony justified its exclusion from this regression model, ensuring a more interpretable and efficient analysis.

From the output above, we can see that the squared term for YouTube, is also significant in the transformed model (p-value = 2.5e-09). Including this term and its p-value of less than 0.05 suggest a more complex relationship between YouTube advertising and sales. The transformed model has a lower RSE, which indicates that the predictions are, on average, closer to the actual values. The higher R-squared and Adjusted R-squared values indicate that it explains a higher proportion of the variance in sales.

we will validate this model's assumptions in a similar way as we did with `Model 2` (R code 7.31):



To verify the normality of the model, we will run the Shapiro-Wilk test (R code 7.32):

```
	Shapiro-Wilk normality test

data:  transformed_model$residuals
W = 0.99194, p-value = 0.5582
```

We cannot reject the null hypothesis because the p-value is greater than a common significance level of 0.05. This means that there is not enough evidence to conclude that the residuals depart significantly from a normal distribution. Therefore, the residuals can be considered approximately normally distributed.

To further check the correlation between errors, we run the Durbin-Watson test used to detect autocorrelation in the residuals (errors) of a regression analysis (R code 7.33). Autocorrelation occurs when the residuals are correlated with each other, indicating that there may be some pattern or structure left in the residuals that the model has not captured.

```
The Durbin-Watson test: 2.25
```

The test statistic of the output suggests that there may be some slight positive autocorrelation in the residuals. However, it is not severe enough to invalidate the assumptions of the regression model.

We will also use the studentised Breusch-Pagan test to assess the presence of heteroscedasticity in a regression model (R code 7.34):

```

	studentized Breusch-Pagan test

data:  transformed_model
BP = 43.825, df = 3, p-value = 1.644e-09
```

The small p-value suggests strong evidence against the null hypothesis, indicating statistically significant evidence of heteroscedasticity in the transformed model.

The transformed model fits the data's central part well, with residuals showing approximate normality and no signs of influential points from leverage. The potential issues indicated by the diagnostic plots are the slight non-linearity and heteroscedasticity, as well as the presence of outliers that could distort the model's predictions. It might be beneficial to investigate these outliers further to determine if it is an anomaly or if some aspect of the data is not captured by the model.

While various data transformation techniques offer a robust means to enhance model fit and capture non-linear relationships, such an investigative process demands significant time and methodological consideration. As such, a comprehensive exploration of the optimal data transformation is beyond the scope of this report. The models presented herein, namely `Model 2` and `Transformed Model`, have been selected based on preliminary analyses which suggest their efficacy. Yet, it should be acknowledged that with additional time and resources, further refinement through transformation could yield improvements in model performance.

## 2.3 Sales Predictions

As we previously established, given the similar performance of `Model 1` and `Model 2` and the lack of significance of the newspaper in `Model 1`, `Model 2` is the best for its simplicity and efficiency without losing explanatory power. It effectively balances model complexity and performance, adhering to the principle of parsimony.

Now, we assess how well `Model 2` generalises to new, unseen data. We will try to predict sales for the `out_of_sample` testing set, which the model has not seen during training.

The table below compares actual sales from the out-of-sample testing set against sales predicted by `Model 2` for the same data set (R code 7.35). This variance indicates how far off the predictions are from the actual sales, with a positive percentage indicating an overestimation and a negative percentage indicating an underestimation.

```
    Actual_Sales Predicted_Sales Predicted_vs_Actual_Var
151        19.32           22.42                  16.05%
152        13.92           12.24                 -12.07%
153        19.92           19.76                   -0.8%
154        22.80           21.80                  -4.39%
155        18.72           18.73                   0.05%
156         3.84            6.33                  64.84%
157        18.36           18.25                   -0.6%
158        12.12           12.33                   1.73%
159         8.76           12.20                  39.27%
160        15.48           14.99                  -3.17%
161        17.28           17.22                  -0.35%
```

```
162        15.96           16.13                   1.07%
163        17.88           18.12                   1.34%
164        21.60           20.74                  -3.98%
165        14.28           13.38                   -6.3%
166        14.28           17.56                  22.97%
167         9.60           12.70                  32.29%
168        14.64           16.39                  11.95%
169        20.52           20.83                   1.51%
170        18.00           21.92                  21.78%
171        10.08            8.92                 -11.51%
172        17.40           17.38                  -0.11%
173         9.12            9.03                  -0.99%
174        14.04           14.63                    4.2%
175        13.80           16.88                  22.32%
176        32.40           29.73                  -8.24%
177        24.24           24.11                  -0.54%
178        14.04           14.88                   5.98%
179        14.16           19.70                  39.12%
180        15.12           15.09                   -0.2%
181        12.60           13.00                   3.17%
182        14.64           17.09                  16.73%
183        10.44            8.00                 -23.37%
184        31.44           29.07                  -7.54%
185        21.12           22.50                   6.53%
186        27.12           24.86                  -8.33%
187        12.36           11.92                  -3.56%
188        20.76           20.55                  -1.01%
189        19.08           22.72                  19.08%
190         8.04            7.26                   -9.7%
191        12.96           14.67                  13.19%
192        11.88           10.18                 -14.31%
193         7.08            5.46                 -22.88%
194        23.52           22.04                  -6.29%
195        20.76           19.70                  -5.11%
196         9.12            6.55                 -28.18%
197        11.64            9.97                 -14.35%
198        15.36           15.59                    1.5%
199        30.60           28.63                  -6.44%
200        16.08           18.54                   15.3%
```

We can see that predictions closely align with actual figures in many cases, indicating the model's effectiveness in forecasting sales based on the chosen predictors (YouTube and Facebook). However, there are discrepancies, such as a notable overestimation for record 156 and an underestimation for record 176, suggesting room for model refinement. Overall, `Model 2` demonstrates a good predictive capability, sufficient for informed decision-making in marketing strategies.

Finally, we similarly assess `Model 2`, which includes the square root term of YouTube (a.k.a. the `Transformed Model`) (R code 7.36).

```
    Actual_Sales Predicted_Sales Predicted_vs_Actual_Var
151        19.32           20.77                   7.51%
152        13.92           13.31                  -4.38%
153        19.92           20.13                   1.05%
154        22.80           22.79                  -0.04%
```

```
155        18.72          19.25                   2.83%
156         3.84           2.16                 -43.75%
157        18.36          19.71                   7.35%
158        12.12          13.13                   8.33%
159         8.76           9.92                  13.24%
160        15.48          16.14                   4.26%
161        17.28          17.94                   3.82%
162        15.96          17.43                   9.21%
163        17.88          18.59                   3.97%
164        21.60          21.80                   0.93%
165        14.28          14.54                   1.82%
166        14.28          16.93                  18.56%
167         9.60          11.23                  16.98%
168        14.64          16.37                  11.82%
169        20.52          20.85                   1.61%
170        18.00          20.12                  11.78%
171        10.08           9.17                  -9.03%
172        17.40          18.24                   4.83%
173         9.12           7.53                 -17.43%
174        14.04          15.28                   8.83%
175        13.80          16.52                  19.71%
176        32.40          28.59                 -11.76%
177        24.24          23.47                  -3.18%
178        14.04          15.51                  10.47%
179        14.16          18.02                  27.26%
180        15.12          15.81                   4.56%
181        12.60          13.74                   9.05%
182        14.64          16.84                  15.03%
183        10.44           8.38                 -19.73%
184        31.44          27.57                 -12.31%
185        21.12          21.62                   2.37%
186        27.12          25.35                  -6.53%
187        12.36          12.83                    3.8%
188        20.76          21.10                   1.64%
189        19.08          20.92                   9.64%
190         8.04           5.57                 -30.72%
191        12.96          14.82                  14.35%
192        11.88          11.06                   -6.9%
193         7.08           3.51                 -50.42%
194        23.52          23.12                   -1.7%
195        20.76          20.91                   0.72%
196         9.12           6.20                 -32.02%
197        11.64          10.97                  -5.76%
198        15.36          16.14                   5.08%
199        30.60          27.22                 -11.05%
200        16.08          18.03                  12.13%
```

To compare both models quantitatively, we calculate the mean absolute percentage variance for each (R code 7.37):

```
The mean absolute % variance for Model 2: 11.33
```

```
The mean absolute % variance for the Transformed Model: 10.82
```

The model with the lower mean absolute percentage variance would be the one that, on average, makes predictions closer to the actual values and, therefore, could be considered better in terms of this specific metric. In addition, we should also consider other model diagnostics, such as R-squared, residual standard error, and the results of residual plots, to fully evaluate which model provides the best fit. Thus, we can conclude that the `Transformed Model` can be identified as the better fit for this data set.

---

# 3 Default of Credit Card Clients: Logistic Regression Model

In this task, we will be working with the "Default of Credit Card Clients" data set that contains information about credit card defaults among clients in Taiwan from April 2005 to September 2005. It includes data on 30,000 distinct credit card clients, with each record comprising 24 attributes. These attributes cover a range of information, including default payments, demographic factors, credit data, payment history, and bill statements. The full details regarding the variables can be found here. A summary of the first 5 rows of the data set is presented below (R code 7.38):

```
  ID LIMIT_BAL SEX EDUCATION MARRIAGE AGE PAY_0 PAY_2 PAY_3 PAY_4 PAY_5 PAY_6
1  1     20000   2         2        1  24     2     2    -1    -1    -2    -2
2  2    120000   2         2        2  26    -1     2     0     0     0     2
3  3     90000   2         2        2  34     0     0     0     0     0     0
4  4     50000   2         2        1  37     0     0     0     0     0     0
5  5     50000   1         2        1  57    -1     0    -1     0     0     0
  BILL_AMT1 BILL_AMT2 BILL_AMT3 BILL_AMT4 BILL_AMT5 BILL_AMT6 PAY_AMT1 PAY_AMT2
1      3913      3102       689         0         0         0        0      689
2      2682      1725      2682      3272      3455      3261        0     1000
3     29239     14027     13559     14331     14948     15549     1518     1500
4     46990     48233     49291     28314     28959     29547     2000     2019
5      8617      5670     35835     20940     19146     19131     2000    36681
  PAY_AMT3 PAY_AMT4 PAY_AMT5 PAY_AMT6 default.payment.next.month
1        0        0        0        0                          1
2     1000     1000        0     2000                          1
3     1000     1000     1000     5000                          0
4     1200     1100     1069     1000                          0
5    10000     9000      689      679                          0
```

## 3.1 Fitting Logistic Regression Model

In the context of the "Default of Credit Card Clients" data set, logistic regression is employed to analyse the factors influencing credit card default and predict the likelihood of future defaults based on historical data. This statistical method is particularly suitable for binary outcome variables, like defaulting (coded as 1) or not defaulting (coded as 0) on a credit card payment, as is the case with this data set.

The process begins by dividing the data set into the in-sample (or training) set and the out-of-sample (or testing) set (R code 7.39). This partition facilitates model training on one subset and testing on another, enabling evaluation of the model's predictive performance and generalisability to unseen data.

The logistic regression model, fitted using the in-sample data, assesses the relationship between various independent variables (e.g., demographic factors, payment history) and the dependent binary outcome variable (credit card default) (R code 7.40). By estimating probabilities using a logistic function, which is bounded between 0 and 1, logistic regression is ideal for modelling binary outcomes.

```
The Initial Model (including all variables)
```

```
Call:
glm(formula = default.payment.next.month ~ ., family = "binomial",
    data = in_sample_creditcard)

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -6.260e-01  1.399e-01  -4.474 7.68e-06 ***
ID          -1.281e-06  2.016e-06  -0.635 0.525262
LIMIT_BAL   -9.509e-07  1.829e-07  -5.200 1.99e-07 ***
SEX         -1.338e-01  3.537e-02  -3.783 0.000155 ***
EDUCATION   -1.234e-01  2.428e-02  -5.082 3.73e-07 ***
MARRIAGE    -1.370e-01  3.668e-02  -3.736 0.000187 ***
AGE          8.896e-03  2.059e-03   4.320 1.56e-05 ***
PAY_0        5.815e-01  2.039e-02  28.523  < 2e-16 ***
PAY_2        8.426e-02  2.318e-02   3.634 0.000279 ***
PAY_3        7.108e-02  2.613e-02   2.720 0.006528 **
PAY_4        4.633e-02  2.897e-02   1.600 0.109690
PAY_5        1.207e-02  3.135e-02   0.385 0.700221
PAY_6        5.499e-03  2.578e-02   0.213 0.831097
BILL_AMT1   -5.908e-06  1.288e-06  -4.586 4.53e-06 ***
BILL_AMT2    2.489e-06  1.693e-06   1.470 0.141526
BILL_AMT3    2.171e-06  1.502e-06   1.446 0.148305
BILL_AMT4   -4.092e-07  1.572e-06  -0.260 0.794589
BILL_AMT5    3.959e-07  1.840e-06   0.215 0.829622
BILL_AMT6    4.704e-07  1.442e-06   0.326 0.744267
PAY_AMT1    -1.494e-05  2.757e-06  -5.418 6.02e-08 ***
PAY_AMT2    -9.423e-06  2.372e-06  -3.972 7.12e-05 ***
PAY_AMT3    -1.165e-06  1.930e-06  -0.604 0.545982
PAY_AMT4    -5.797e-06  2.250e-06  -2.577 0.009968 **
PAY_AMT5    -3.910e-06  2.111e-06  -1.853 0.063928 .
PAY_AMT6    -2.369e-06  1.511e-06  -1.568 0.116826
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 23924  on 22499  degrees of freedom
Residual deviance: 20931  on 22475  degrees of freedom
AIC: 20981

Number of Fisher Scoring iterations: 6
```

The initial logistic regression model above includes all variables and estimates each variable’s effect on the log odds of defaulting. Significant predictors of default include `LIMIT_BAL`, `SEX`, `EDUCATION`, `MARRIAGE`, `AGE`, and repayment status from previous months (e.g., `PAY_0`, `PAY_2`, `PAY_3`). Some variables like `PAY_5`, `PAY_6`, `BILL_AMT4`, `BILL_AMT5`, and `BILL_AMT6` show less significance, suggesting potential model simplification without sacrificing predictive power. Based on the above output, a better-fit logistic regression model could be achieved by removing variables that do not significantly contribute to the model, as indicated by their p-values. Hence, a simplified model is derived by removing less significant variables, resulting in improved interpretability without substantial loss of predictive accuracy (R code 7.41).

```
Simplified Model


Call:
glm(formula = default.payment.next.month ~ LIMIT_BAL + SEX +
    EDUCATION + MARRIAGE + AGE + PAY_0 + PAY_2 + PAY_3, family = "binomial",
    data = in_sample_creditcard)

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -7.093e-01  1.359e-01  -5.221 1.78e-07 ***
LIMIT_BAL   -2.020e-06  1.600e-07 -12.629  < 2e-16 ***
SEX         -1.199e-01  3.514e-02  -3.412 0.000645 ***
EDUCATION   -1.421e-01  2.400e-02  -5.921 3.20e-09 ***
MARRIAGE    -1.464e-01  3.640e-02  -4.023 5.76e-05 ***
AGE          9.361e-03  2.046e-03   4.576 4.74e-06 ***
PAY_0        6.064e-01  2.038e-02  29.755  < 2e-16 ***
PAY_2        7.311e-02  2.261e-02   3.233 0.001226 **
PAY_3        9.955e-02  2.085e-02   4.775 1.80e-06 ***
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 23924  on 22499  degrees of freedom
Residual deviance: 21117  on 22491  degrees of freedom
AIC: 21135

Number of Fisher Scoring iterations: 4
```

When comparing the initial and simplified models, we can see the initial model includes more variables, potentially capturing more data nuances but risking overfitting. The simplified model, with fewer variables, offers better interpretability and generalisability. However, the initial model is a slightly better fit to the data, indicated by a lower AIC (20,981 vs. 21,135) and residual deviance. Although the difference is not substantial, we can suggest both models fit the data well. The simplified model converged faster (4 iterations) than the initial model (6 iterations), suggesting a more straightforward optimisation process due to fewer variables.

There are other techniques for variable selection, such as stepwise selection, forward selection, or backward elimination. For this case, we apply the stepwise selection with `both` directions to the initial model with all variables (R code 7.42):

```
The best-fit model chosen with stepwise selection


Call:
glm(formula = default.payment.next.month ~ LIMIT_BAL + SEX +
    EDUCATION + MARRIAGE + AGE + PAY_0 + PAY_2 + PAY_3 + PAY_4 +
    BILL_AMT1 + BILL_AMT2 + BILL_AMT3 + PAY_AMT1 + PAY_AMT2 +
    PAY_AMT4 + PAY_AMT5 + PAY_AMT6, family = "binomial", data = in_sample_creditcard)

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -6.456e-01  1.370e-01  -4.711 2.46e-06 ***
```

```
LIMIT_BAL   -9.589e-07  1.808e-07  -5.302 1.14e-07 ***
SEX         -1.338e-01  3.535e-02  -3.784 0.000154 ***
EDUCATION   -1.248e-01  2.424e-02  -5.148 2.64e-07 ***
MARRIAGE    -1.367e-01  3.666e-02  -3.729 0.000192 ***
AGE          8.889e-03  2.059e-03   4.317 1.58e-05 ***
PAY_0        5.834e-01  2.034e-02  28.684  < 2e-16 ***
PAY_2        8.537e-02  2.307e-02   3.701 0.000215 ***
PAY_3        7.194e-02  2.603e-02   2.763 0.005720 **
PAY_4        5.900e-02  2.292e-02   2.574 0.010052 *
BILL_AMT1   -6.013e-06  1.287e-06  -4.674 2.95e-06 ***
BILL_AMT2    2.501e-06  1.690e-06   1.480 0.138945
BILL_AMT3    2.582e-06  1.165e-06   2.216 0.026716 *
PAY_AMT1    -1.524e-05  2.746e-06  -5.549 2.87e-08 ***
PAY_AMT2    -9.805e-06  2.347e-06  -4.178 2.94e-05 ***
PAY_AMT4    -5.300e-06  1.970e-06  -2.690 0.007135 **
PAY_AMT5    -3.708e-06  1.772e-06  -2.093 0.036371 *
PAY_AMT6    -2.539e-06  1.490e-06  -1.705 0.088229 .
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 23924  on 22499  degrees of freedom
Residual deviance: 20934  on 22482  degrees of freedom
AIC: 20970

Number of Fisher Scoring iterations: 6
```

Employing stepwise selection on the initial model, we got a model with a balance between complexity and fit, featuring significant predictors similar to both the initial and simplified models, such as `LIMIT_BAL`, `SEX`, `EDUCATION`, `MARRIAGE`, `AGE`, `PAY_0`, `PAY_2`, and `PAY_3`. The inclusion of `PAY_4`, `BILL_AMT1`, `BILL_AMT3`, `PAY_AMT1`, `PAY_AMT2`, `PAY_AMT4`, and `PAY_AMT5` indicates these variables also contribute to predicting default, based on their statistical significance. The stepwise model has a lower AIC of 20970, indicating a better fit when considering model complexity. It has a residual deviance of 20934, slightly better than the simplified model (21117) and comparable to the initial model (20931).

## 3.2 Confusion Matrix

Assuming a cut-off point of 0.5 probability with which we categorise a client to have default, we can calculate the values of the confusion matrices (classification tables) used to describe the performance of each classification model on a set of data for which the true values are known (R code 7.43):

```
Initial Model (including all variables)



    FALSE  TRUE
  0 16951   514
  1  3764  1271


Simplified Model
```

```
    FALSE  TRUE
  0 16977   488
  1  3785  1250
```

```
Stepwise Model
```

```
    FALSE  TRUE
  0 16948   517
  1  3762  1273
```

Analysing the confusion matrices for the three logistic regression models — initial, simplified, and stepwise, we can see that the simplified model shows a slight improvement in specificity (reducing false positives), making it slightly more reliable in identifying non-default cases. However, this comes at a slight cost to sensitivity (true positive rate). On the other hand, the stepwise model offers a good balance, with a minor improvement in identifying true defaults without significantly increasing false positives.

The stepwise model balances the initial and simplified models in terms of complexity and fit. Its lower AIC suggests it might offer the best compromise between explanatory power and parsimony among the three models. It is important to remember that the choice of the "best" model could depend on the specific needs and costs associated with false positives (incorrectly predicting a default) versus false negatives (failing to predict a default).

Ultimately, the choice of the model should also consider out-of-sample predictive performance, which can be evaluated through a confusion matrix on out-of-sample data or other cross-validation techniques.

## 3.3 Predictions

Predictions on out-of-sample data indicate the model's ability to classify clients into default and non-default categories. We follow the common practice and use a threshold of 0.5, where probabilities above this threshold are interpreted as predicting a default (coded as 1) and those below as predicting a non-default (coded as 0).

The top 5 rows of the out-of-sample data are displayed below, comparing actual and predicted default class (0 or 1) by the stepwise model (R code 7.44):

```
   Actual Predicted
1       1         1
6       0         0
8       0         0
12      0         0
25      0         0
29      0         0
34      0         0
35      0         0
40      0         0
48      1         0
```

The stepwise model's performance, particularly in terms of sensitivity and specificity, is evaluated through the confusion matrix, which compares the predicted default outcomes against the actual outcomes in the out-of-sample data. It provides a clear breakdown of true positives, false positives, true negatives, and false negatives, allowing for a comprehensive assessment of the model's performance, including its precision, recall, and overall accuracy (R code 7.45):

```
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 5714 1208
         1  185  393

               Accuracy : 0.8143
                 95% CI : (0.8053, 0.823)
    No Information Rate : 0.7865
    P-Value [Acc > NIR] : 1.4e-09

                  Kappa : 0.2791

 Mcnemar's Test P-Value : < 2e-16

            Sensitivity : 0.9686
            Specificity : 0.2455
         Pos Pred Value : 0.8255
         Neg Pred Value : 0.6799
             Prevalence : 0.7865
         Detection Rate : 0.7619
   Detection Prevalence : 0.9229
      Balanced Accuracy : 0.6071

       'Positive' Class : 0
```

The above confusion matrix and statistics for the stepwise model demonstrate that about 81.43% of all predictions made by the model are correct. While this seems high, it is essential to consider this metric in the context of the data set's class distribution. The model is highly sensitive, correctly identifying 96.86% of all 'non-default' cases. This indicates strong performance in detecting the 'non-default' class. However, the model has low specificity, correctly identifying only 24.55% of all 'default' cases. This suggests the model struggles to identify 'default' customers correctly. When the model predicts 'non-default', it is correct 82.55% of the time. When the model predicts 'default', it is correct 67.99% of the time. The Kappa statistic measures the agreement between the predictions and the actual labels, considering the agreement occurring by chance. A Kappa value of 0.2791 indicates fair agreement. The balanced accuracy, which averages sensitivity and specificity, is moderate (0.6071), indicating that the model's overall ability to distinguish between classes is fair but not excellent. We can conclude that while the model is accurate and highly sensitive to the 'non-default' class, its specificity and ability to detect 'default' cases are low. This imbalance could be problematic if the cost of missing a 'default' case is high. The positive predictive value is relatively high, but the negative predictive value is moderate, suggesting room for improvement in accurately predicting 'default' cases.

Further refinements might be necessary to enhance its overall performance and could include exploring interactions between variables or testing non-linear effects to potentially improve model performance. Regularisation methods like Lasso regression could also be explored to enhance the model by penalising large coefficients and helping in feature selection, especially if extending the model to include more predictors. However, this lies beyond the scope of this report.

# 4 Regression Does Not Imply Causation

Regression analysis identifies relationships between a dependent variable and one or more independent variables. It estimates how the dependent variable changes as the independent variables vary but cannot prove causality for several reasons:

1. Regression shows correlation, not causation. For example, ice cream sales and going to a beach increase in summer, but one does not cause the other; both are influenced by temperature.
2. External factors may affect both dependent and independent variables, leading to spurious correlations. For instance, a higher number of hospitals and crime rates in larger cities are both influenced by population size, not by each other.
3. It is difficult to determine the direction of causality; the dependent variable might be influencing the independent variable instead of vice versa. For example, a study might indicate that increased technology use among children is associated with poorer social skills, suggesting that screen time diminishes children's ability to interact effectively. However, reverse causality could be detected: children who naturally have less developed social skills might gravitate more towards technology because it requires less direct social interaction, which could, in turn, lead to even more technology use. Thus, it is unclear whether technology use causes a reduction in social skills or if the initial lower social skills lead to increased use of technology.
4. Excluding relevant variables from the model can lead to inaccurate effect estimates. For example, a study linking study hours to grades might show that more studying improves grades. Yet, without considering students' inherent academic abilities, the study may overestimate the effect of study time. High-ability students might get good grades with less study, while others may study more but achieve less, indicating that academic aptitude is a critical omitted variable in this relationship.
5. Many models assume a linear relationship, which can be misleading if the actual relationship is nonlinear, such as a U-shaped relationship between the amount of sleep one gets and cognitive function. Very little sleep impairs cognition, reducing attention, memory, and reaction times due to insufficient brain function support. Moderate sleep optimises cognitive performance, enhancing memory, decision-making, and problem-solving. However, excessive sleep can again decrease cognitive function, potentially due to health issues linked with oversleeping, such as depression and inflammation.

While regression can highlight patterns, careful interpretation is required to avoid incorrect conclusions about causality.

---

# 5 Leveraging Business Statistics for Post-COVID Economic Recovery [2]

## Introduction

The COVID-19 pandemic shocked global economies unprecedentedly, leading to recessions, unemployment, and various financial challenges. As organisations seek to navigate the post-pandemic landscape, business statistics emerge as an essential tool for economic recovery. This essay explores the application of business statistics in informing policy decisions, fostering business growth, and enhancing labour market resilience.

[2] Word Count: 503 words

## Understanding Economic Health through Data

Business statistics provide a granular view of economic health by tracking indicators such as consumer spending, unemployment rates, and production levels. For example, analysing consumer spending trends can help organisations adapt their operating models and apply forecasting models to locate windows of growth opportunities, drive demand, and reinforce service for existing customers (McKinsey & Company, n.d.).

## Supporting Business Decision-Making

For businesses, statistical analysis enables evidence-based decision-making. By evaluating market trends, consumer behaviour, and supply chain disruptions, companies can adapt their strategies for the post-crisis environment. For instance, retail analytics can inform inventory management, helping companies to optimise stock levels and minimise wastage, reevaluate supply chain disruptions, cost dynamics between countries, and the lack of visibility in lower-tier supply levels, thereby improving profitability (Sneader et al., 2021).

## Enhancing Labour Market Resilience

Labour market statistics are crucial for understanding the pandemic's impact on employment and guiding workforce development. By analysing job loss and growth across different sectors, businesses can identify areas where re-skilling and skill-building are necessary (Bhattacharjee et al., 2021). This approach helps reallocate labour to thriving industries, thus reducing unemployment and enhancing the adaptability of the workforce (Billing et al., 2021).

Moreover, business statistics are crucial in optimising operational efficiency and resource allocation, particularly in a recovering economy. Through predictive analytics and optimisation modelling, businesses can forecast demand, streamline production processes, and optimise supply chain management, improving their competitiveness in the post-crisis marketplace. Additionally, business statistics can inform strategic business decision-making, guiding investments, diversification efforts, and expansion plans in alignment with market dynamics and economic forecasts.

Furthermore, leveraging business statistics can facilitate evidence-based policymaking and strategic planning at both the organisational and governmental levels. By analysing economic indicators, employment data, and financial metrics, policymakers can formulate targeted interventions and stimulus packages to support industries most impacted by the pandemic and other crises.

However, it is essential to recognise the challenges associated with leveraging business statistics for economic recovery. Data quality, accessibility, and privacy concerns pose significant barriers to harnessing the full potential of statistical analysis. Addressing these challenges requires collaborative efforts between governments, businesses, and data stakeholders to enhance data collection, standardisation, and sharing mechanisms while ensuring compliance with regulatory frameworks and ethical guidelines.

## Conclusion

Leveraging business statistics is instrumental in facilitating economic recovery and driving sustainable growth in the long term. By leveraging the power of statistics, businesses can gain actionable insights, optimise decision-making processes, and adapt to changing market dynamics effectively. Moreover, policymakers can leverage business statistics to design targeted interventions and policy measures that support economic resilience and inclusive development. However, realising the full potential of business statistics requires concerted efforts to address data-related challenges and promote data-driven innovation across sectors.

# 6 References

Bhattacharjee, D., Bustamante, F., Curley, A., and Perez, F. (2021) *Navigating the labor mismatch in US logistics and supply chains.* Available at: https://www.mckinsey.com/capabilities/operations/our-insights/navigating-the-labor-mismatch-in-us-logistics-and-supply-chains (Accessed: 9 February 2024).

Billing F., De Smet A., Reich A., and Schaninger B. (2021) *Building workforce skills at scale to thrive during and after – the COVID-19 crisis*, in McKinsey Global Surveys 2021: A year in review. Available at: https://www.mckinsey.com/~/media/mckinsey/featured%20insights/mckinsey%20global%20surveys/mckinsey-global-surveys-2021-a-year-in-review.pdf (Accessed: 9 February 2024).

McKinsey & Company (n.d.) *Emerging consumer trends in a post-COVID-19 world.* Available at: https://www.mckinsey.com/capabilities/growth-marketing-and-sales/our-insights/emerging-consumer-trends-in-a-post-covid-19-world (Accessed: 9 February 2024).

Schmidt, A.F. and Finan, C. (2018) 'Linear regression and the normality assumption', *Journal of Clinical Epidemiology*, 98, pp.146–151. Available at: https://doi.org/10.1016/j.jclinepi.2017.12.006.

Sneader, K. and Singhal, S. (2021) *The next normal arrives: Trends that will define 2021—and beyond.* Available at: https://www.mckinsey.com/featured-insights/leadership/the-next-normal-arrives-trends-that-will-define-2021-and-beyond (Accessed: 9 February 2024).

---

# 7 Appendices

## 7.1 Code dispaying first five rows of the cars data set

```
head(cars, 5)
```

## 7.2 Code plotting speed histogram

```
hist(cars$speed, main = "Histogram of Speed",
     xlab = "Speed (mph)",
     ylab = "Frequency",
     col="lightblue")
```

## 7.3 Code loading measures of shape and distribution of speed and stopping distance

```
library(moments)
summary(cars$speed)
skewness(cars$speed)
kurtosis(cars$speed)
IQR(cars$dist)
```

## 7.4 Code plotting stopping distance histogram

```
hist(cars$dist, main = "Histogram of Stopping Distance",
     xlab = "Stopping Distance (ft)",
     ylab = "Frequency",
     col="lightgreen")
```

## 7.5 Code loading measures of shape and distribution of speed and stopping distance

```
summary(cars$dist)
skewness(cars$dist)
kurtosis(cars$dist)
IQR(cars$dist)
```

## 7.6 Code plotting box plots of speed and stopping distance

```
par(mfrow = c(1, 2))
boxplot(cars$speed, main = "Boxplot of Speed", ylab="Speed (mph)")
boxplot(cars$dist, main = "Boxplot of Stopping Distance", ylab="Stopping Distance (ft)")
```

## 7.7 Code calculating a correlation coefficient between car speed and stopping distance

```
cat("The correlation coefficient between car speed and stopping distance is",
    round(cor(cars$speed, cars$dist), 2))
```

## 7.8 Code fitting a simple linear regression model and printing its summary

```
model <- lm(dist ~ speed, data = cars)
```

```
print(summary(model))
```

## 7.9 Code visualising the relationship between car speed and stopping distance

```
plot(x=cars$speed, y=cars$dist,
     main = "Relationship between Speed and Stopping Distance",
     xlab = "Speed (mph)",
     ylab = "Stopping Distance (ft)")
abline(model, col = "red")
```

## 7.10 Code assessing linearity of the model by plotting ‘Residuals vs Fitted’

```r
plot(model,1)
```

## 7.11 Code assessing independence of errors (autocorrelation) of the model by plotting ‘Series residuals’

```r
acf(residuals(model))
```

## 7.12 Code assessing independence of errors (autocorrelation) of the model by running the Durbin-Watson test

```r
library(lmtest)
dwtest(model)
```

## 7.13 Code plotting ‘Histogram of Residuals’

```r
hist(model$residuals, breaks = 10, main = "Histogram of Residuals")
```

## 7.14 Code plotting ‘Q-Q Residuals’

```r
plot(model, 2)
```

## 7.15 Code running the Shapiro-Wilk normality test

```r
shapiro.test(model$residuals)
```

## 7.16 Code plotting ‘Scale-Location’ to assess homoscedasticity

```r
plot(model, 3)
```

## 7.17 Code running the Breusch-Pagan test

```r
library(lmtest)
lmtest::bptest(model)
```

## 7.18 Code calculating the minimum speed for the fastest 10%

```
# Parameters of the normal distribution
mean_speed <- 40.0  # Mean speed
sd_speed <- 12.1    # Standard deviation of speed

fastest_10_percent_speed <- qnorm(0.90, mean = mean_speed, sd = sd_speed)
cat("Minimum speed required to belong to the fastest 10%:",
    round(fastest_10_percent_speed, 2), "mph\n")
```

## 7.19 Code calculating the percentage of cars slower than 30 mph

```
slower_than_30_percent <- pnorm(30, mean = mean_speed, sd = sd_speed) * 100
cat("Percentage of cars slower than 30 mph:", round(slower_than_30_percent, 2), "%\n")
```

## 7.20 Code loading `datarium` package and first five rows of the `marketing` data set

```
library(datarium)
head(marketing,5)
```

## 7.21 Code calculating correlation between the marketing variables

```
cor(marketing)
```

## 7.22 Code splitting the marketing data set into in-sample (first 150 records) and out-of-sample (last 50 records)

```
in_sample <- marketing[1:150, ]
out_of_sample <- marketing[151:200, ]
```

## 7.23 Code fitting 'Model 1' (YouTube, Facebook and newspaper expenditures as predictors)

```
model1 <- lm(sales ~ youtube + facebook + newspaper, data = in_sample)
summary(model1)
```

## 7.24 Code fitting 'Model 2' (YouTube and Facebook, excluding newspapers expenditures as predictors)

```r
model2 <- lm(sales ~ youtube + facebook, data = in_sample)
summary(model2)
```

## 7.25 Code fitting ‘Model 3’ (only YouTube expenditures as predictors)

```r
model3 <- lm(sales ~ youtube, data = in_sample)
summary(model3)
```

## 7.26 Code executing vif() function from the `car` package to formally check for multicollinearity in ‘Model 1’

```r
library("car")
vif(model1)
```

## 7.27 Code applying the stepwise selection with ‘both’ directions to the marketing data

```r
marketing_stepwise <- step(lm(sales ~ youtube + facebook + newspaper, data = in_sample),
                           direction = "both")
```

```r
summary(marketing_stepwise)
```

## 7.28 Code producing plots on regression analysis assumptions (linearity, independence of errors, normality, and equal variance of errors)

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

## 7.29 Code running the Shapiro-Wilk test to verify the normality of ‘Model 2’

```r
shapiro.test(model2$residuals)
```

## 7.30 Code fitting a transformed ‘Model 2’ including both a linear term for YouTube advertising and its square root term, and dispaying its summary

```r
transformed_model <- lm(formula = sales ~ youtube + sqrt(youtube) + facebook,
                        data = in_sample)
summary(transformed_model)
```

### 7.31 Code producing plots to validate linear regression assumptions for the ‘Transformed Model’

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

### 7.32 Code running the Shapiro-Wilk test to verify the normality of the ‘Transformed Model’

```
shapiro.test(transformed_model$residuals)
```

### 7.33 Code running the Durbin-Watson test to detect autocorrelation in the residuals (errors) of the ‘Transformed Model’

```
library(car)
cat("The Durbin-Watson test:", round(durbinWatsonTest(transformed_model$residuals),2))
```

### 7.34 Code running the Breusch-Pagan test to assess the presence of heteroscedasticity in the ‘Transformed Model’

```
library(lmtest)
bptest(transformed_model)
```

### 7.35 Code comparing actual sales from the out-of-sample testing set against sales predicted by ‘Model 2’ for the same data set

```
predicted_sales <- predict(model2, newdata = out_of_sample)

predicted_results <- data.frame(Actual_Sales = out_of_sample$sales,
                                Predicted_Sales = round(predicted_sales, 2))

# Calculate percentage variance and add % symbol
predicted_results$Predicted_vs_Actual_Var <- paste0(
  round(((predicted_results$Predicted_Sales - predicted_results$Actual_Sales) /
             predicted_results$Actual_Sales) * 100, 2),
  "%")
print(predicted_results)
```

### 7.36 Code comparing actual sales from the out-of-sample testing set against sales predicted by the ’Transformed Model‘ for the same data set

```
predicted_sales_transformed <- predict(transformed_model, newdata = out_of_sample)

predicted_results_transformed <- data.frame(Actual_Sales = out_of_sample$sales,
                                 Predicted_Sales = round(predicted_sales_transformed, 2))

# Calculate percentage variance and add % symbol
predicted_results_transformed$Predicted_vs_Actual_Var <- paste0(
  round((
      (predicted_results_transformed$Predicted_Sales -
           predicted_results_transformed$Actual_Sales) /
       predicted_results_transformed$Actual_Sales) * 100, 2),
  "%")
print(predicted_results_transformed)
```

## 7.37 Code calculating the mean absolute percentage variance for ‘Model 2’ amd ‘Transformed Model’ for the same data set

```
# For the non-transformed model
mean_absolute_variance_non_transformed <- mean(
    abs(predicted_results$Predicted_Sales -
            predicted_results$Actual_Sales) /
        predicted_results$Actual_Sales)

# For the transformed model
mean_absolute_variance_transformed <- mean(
    abs(predicted_results_transformed$Predicted_Sales -
            predicted_results_transformed$Actual_Sales) /
        predicted_results_transformed$Actual_Sales)

# Print the mean absolute percentage variances for comparison
cat("The mean absolute % variance for Model 2:",
    round(mean_absolute_variance_non_transformed*100,2))
cat("The mean absolute % variance for the Transformed Model:",
    round(mean_absolute_variance_transformed*100,2))
```

## 7.38 Code loading the ‘Default of Credit Card Clients’ data set and displaying its first five rows

```
creditcard <- read.csv("https://code.datasciencedojo.com/datasciencedojo/datasets/raw/master/Default%20
                       skip=1)
head(creditcard,5)
```

## 7.39 Code splitting the ‘Default of Credit Card Clients’ data set into in-sample and out-of-sample sets

```
library(dplyr)
library(caret)
set.seed(123) # For reproducibility
in_sample_index_creditcard <- createDataPartition(creditcard$default.payment.next.month,
                                                  p = 0.75, list = FALSE)
in_sample_creditcard <- creditcard[in_sample_index_creditcard, ]
out_of_sample_creditcard <- creditcard[-in_sample_index_creditcard, ]
```

## 7.40 Code fitting an initial logistic regression model using all in-sample data

```
initial_model <- glm(default.payment.next.month ~ ., data = in_sample_creditcard,
                     family = "binomial")
cat("The Initial Model (including all variables)")
summary(initial_model)
```

## 7.41 Code fitting a simplified logistic regression model

```
cat("Simplified Model")
simplified_model <- glm(default.payment.next.month ~
                        LIMIT_BAL + SEX + EDUCATION + MARRIAGE + AGE + PAY_0 + PAY_2 + PAY_3,
                        family = "binomial", data = in_sample_creditcard)
summary(simplified_model)
```

## 7.42 Code fitting a logistic regression model applying the stepwise selection with ‘both’ directions to the initial model with all variables

```
stepwise_model <- step(initial_model, direction = "both")
```

```
cat("The best-fit model chosen with stepwise selection")
summary(stepwise_model)
```

## 7.43 Code producing confusion matrices (classification tables) for the logistics regression models

```
cat("Initial Model (including all variables)")
table(initial_model$y, fitted(initial_model)>=0.5)
cat("Simplified Model")
table(simplified_model$y, fitted(simplified_model)>=0.5)
cat("Stepwise Model")
table(stepwise_model$y, fitted(stepwise_model)>=0.5)
```

## 7.44 Code displaying top 5 rows of the out-of-sample data and comparing actual and predicted default class (0 or 1) by the stepwise model

```
predictions <- predict(stepwise_model, newdata = out_of_sample_creditcard,
                       type = "response")
```

```
comparison_df <- data.frame(Actual = out_of_sample_creditcard$default.payment.next.month,
                            Predicted = ifelse(predictions > 0.5, 1, 0))
head(comparison_df,10)
```

## 7.45 Code producing a confusion matrix (classification table), which compares the predicted default outcomes against the actual outcomes in the out-of-sample data

```
# Convert predicted probabilities to binary outcomes
predicted_classes <- ifelse(predictions > 0.5, 1, 0) # 1 for default and 0 for non-default

actual_outcomes <- out_of_sample_creditcard$default.payment.next.month

# Generate the confusion matrix
library(caret)
confusion_matrix <- confusionMatrix(as.factor(predicted_classes),
                                    as.factor(actual_outcomes))
print(confusion_matrix)
```