Multiple Regression Project

Kabin Devkota, Nishan Khanal and Udita Bista

4/15/2025

Introduction

This analysis focuses on a data set from Kaggle that describes the nutrition facts for McDonald's Menu. This dataset provides a nutrition analysis of every menu item on the US McDonald's menu, including breakfast, beef burgers, chicken and fish sandwiches, fries, salads, soda, coffee and tea, milkshakes, and desserts. The data for this analysis consists of response variable $y = Calories$ and following explanatory variables:

$Category$

$item$

$ServingSize$

$TotalFat$

$TotalFat(\%DailyValue)$

$SaturatedFat$

$SaturatedFat(\%DailyValue)$

$TransFat$

$Cholesterol$

$Cholesterol(\%DailyValue)$

$Sodium$

$Sodium(\%DailyValue)$

$Carbohydrates$

$Carbohydrates(\%DailyValue)$

$DietaryFiber$

$DietaryFiber(\%DailyValue)$

$Sugars$

$Protein$

$VitaminA(\%DailyValue)$

$VitaminC(\%DailyValue)$

$Calcium(\%DailyValue)$

$Iron(\%DailyValue)$

```r
# library(ggplot2)
library(lmtest, pos=4)
```

```
## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```r
library(corrplot)
```

```
## corrplot 0.95 loaded
```

```r
# reading the data from the csv file which has some values containing comma enclosed by double quotes
mcdonalds = read.csv("Mcdonalds_menu.csv",header=TRUE,quote="\"",sep=",")
head(mcdonalds)
```

```
##    Category                             Item   Serving.Size Calories
## 1 Breakfast                     Egg McMuffin 4.8 oz (136 g)      300
## 2 Breakfast                Egg White Delight 4.8 oz (135 g)      250
## 3 Breakfast                 Sausage McMuffin 3.9 oz (111 g)      370
## 4 Breakfast        Sausage McMuffin with Egg 5.7 oz (161 g)      450
## 5 Breakfast Sausage McMuffin with Egg Whites 5.7 oz (161 g)      400
## 6 Breakfast             Steak & Egg McMuffin 6.5 oz (185 g)      430
##   Calories.from.Fat Total.Fat Total.Fat....Daily.Value. Saturated.Fat
## 1               120        13                        20             5
## 2                70         8                        12             3
## 3               200        23                        35             8
## 4               250        28                        43            10
## 5               210        23                        35             8
## 6               210        23                        36             9
##   Saturated.Fat....Daily.Value. Trans.Fat Cholesterol
## 1                            25         0         260
## 2                            15         0          25
## 3                            42         0          45
## 4                            52         0         285
## 5                            42         0          50
## 6                            46         1         300
##   Cholesterol....Daily.Value. Sodium Sodium....Daily.Value. Carbohydrates
## 1                          87    750                     31            31
## 2                           8    770                     32            30
## 3                          15    780                     33            29
## 4                          95    860                     36            30
## 5                          16    880                     37            30
## 6                         100    960                     40            31
##   Carbohydrates....Daily.Value. Dietary.Fiber Dietary.Fiber....Daily.Value.
## 1                            10             4                            17
## 2                            10             4                            17
## 3                            10             4                            17
## 4                            10             4                            17
## 5                            10             4                            17
## 6                            10             4                            18
##   Sugars Protein Vitamin.A....Daily.Value. Vitamin.C....Daily.Value.
## 1      3      17                        10                         0
## 2      3      18                         6                         0
## 3      2      14                         8                         0
## 4      2      21                        15                         0
## 5      2      21                         6                         0
## 6      3      26                        15                         2
##   Calcium....Daily.Value. Iron....Daily.Value.
## 1                      25                   15
## 2                      25                    8
## 3                      25                   10
## 4                      30                   15
## 5                      25                   10
## 6                      30                   20
```

```r
# getting the column names
colnames(mcdonalds)
```

```
##  [1] "Category"                      "Item"
##  [3] "Serving.Size"                  "Calories"
##  [5] "Calories.from.Fat"             "Total.Fat"
##  [7] "Total.Fat....Daily.Value."     "Saturated.Fat"
##  [9] "Saturated.Fat....Daily.Value." "Trans.Fat"
## [11] "Cholesterol"                   "Cholesterol....Daily.Value."
## [13] "Sodium"                        "Sodium....Daily.Value."
## [15] "Carbohydrates"                 "Carbohydrates....Daily.Value."
## [17] "Dietary.Fiber"                 "Dietary.Fiber....Daily.Value."
## [19] "Sugars"                        "Protein"
## [21] "Vitamin.A....Daily.Value."     "Vitamin.C....Daily.Value."
## [23] "Calcium....Daily.Value."       "Iron....Daily.Value."
```

```r
# cleaning column names: replace spaces, %, parentheses, etc.
colnames(mcdonalds) <- make.names(colnames(mcdonalds))
```

Out of these 22 explanatory variables, some of the variables like “category”, “Total.Fat. . . .Daily.Value.”, “Cholesterol. . . .Daily.Value.”, etc are either redundant because there already exist other columns that account for those values or irrelevant for prediction of calories in food items.Therefore, we are dropping them from the analysis to avoid multicollinearity and improve model efficiency.

```r
# dropping the columns that is irrelevant or redundant for the analysis
cols_to_drop <- c(
  "Category",
  "Calories.from.Fat",
  "Saturated.Fat",
  "Trans.Fat",
  "Total.Fat....Daily.Value.",
  "Saturated.Fat....Daily.Value.",
  "Cholesterol....Daily.Value.",
  "Sodium....Daily.Value.",
  "Carbohydrates....Daily.Value.",
  "Sugars",
  "Dietary.Fiber....Daily.Value."
)
mcdonalds <- mcdonalds[, !(names(mcdonalds) %in% cols_to_drop)]
```

Now, we are converting the % Daily Value columns to absolute values. The daily values are based on a 2000 calorie diet, and the conversion is done using the following formula:

$$\text{Absolute Value} = \left(\frac{\text{Percentage Daily Value}}{100}\right) \times \text{Daily Value}$$

where the daily values are as follows:

```
# Daily values (units must match dataset)
daily_values <- c(
  "Vitamin.A" = 900,    # mcg RAE
  "Vitamin.C" = 90,     # mg
  "Calcium" = 1300,     # mg
  "Iron" = 18           # mg
)

# Map of columns to their associated nutrients
conversion_map <- list(
  "Vitamin.A....Daily.Value." = "Vitamin.A",
  "Vitamin.C....Daily.Value." = "Vitamin.C",
  "Calcium....Daily.Value." = "Calcium",
  "Iron....Daily.Value." = "Iron"
)

# For each %DV column, calculate the absolute value and overwrite the same column with absolute value
for (dv_col in names(conversion_map)) {
  nutrient <- conversion_map[[dv_col]]

  if (dv_col %in% names(mcdonalds)) {
    # Create a new column name or overwrite the existing one
    new_col <- nutrient  # Replace the DV column with nutrient name only
    mcdonalds[[new_col]] <- (mcdonalds[[dv_col]] / 100) * daily_values[[nutrient]]
  }
}
```

The values in the serving size column contain both numbers and units (e.g., “15 oz”). In this step, we are extracting only the numeric part and discarding the unit, so “15 oz” becomes just 15.

```
# Extract numeric part before "oz" or "fl oz" from the Serving Size column
mcdonalds$Serving.Size.Oz <- as.numeric(sub("([0-9.]+)\\s*(fl\\s*)?oz.*", "\\1", mcdonalds$Serving.Size
```

```
## Warning: NAs introduced by coercion
```

```
head(mcdonalds)
```

```
##                                Item   Serving.Size Calories Total.Fat
## 1                     Egg McMuffin 4.8 oz (136 g)      300        13
## 2                Egg White Delight 4.8 oz (135 g)      250         8
## 3                 Sausage McMuffin 3.9 oz (111 g)      370        23
## 4        Sausage McMuffin with Egg 5.7 oz (161 g)      450        28
## 5 Sausage McMuffin with Egg Whites 5.7 oz (161 g)      400        23
## 6             Steak & Egg McMuffin 6.5 oz (185 g)      430        23
##   Cholesterol Sodium Carbohydrates Dietary.Fiber Protein
## 1         260    750            31             4      17
## 2          25    770            30             4      18
## 3          45    780            29             4      14
## 4         285    860            30             4      21
## 5          50    880            30             4      21
## 6         300    960            31             4      26
##   Vitamin.A....Daily.Value. Vitamin.C....Daily.Value. Calcium....Daily.Value.
## 1                        10                         0                      25
## 2                         6                         0                      25
## 3                         8                         0                      25
## 4                        15                         0                      30
## 5                         6                         0                      25
## 6                        15                         2                      30
##   Iron....Daily.Value. Vitamin.A Vitamin.C Calcium Iron Serving.Size.Oz
## 1                   15        90       0.0     325 2.70             4.8
## 2                    8        54       0.0     325 1.44             4.8
## 3                   10        72       0.0     325 1.80             3.9
## 4                   15       135       0.0     390 2.70             5.7
## 5                   10        54       0.0     325 1.80             5.7
## 6                   20       135       1.8     390 3.60             6.5
```

```
# dropping the data points with null values that would be a hindrance for the analysis
mcdonalds <- na.omit(mcdonalds)

head(mcdonalds)
```

```
##                               Item   Serving.Size Calories Total.Fat
## 1                     Egg McMuffin 4.8 oz (136 g)      300        13
## 2                Egg White Delight 4.8 oz (135 g)      250         8
## 3                 Sausage McMuffin 3.9 oz (111 g)      370        23
## 4        Sausage McMuffin with Egg 5.7 oz (161 g)      450        28
## 5 Sausage McMuffin with Egg Whites 5.7 oz (161 g)      400        23
## 6             Steak & Egg McMuffin 6.5 oz (185 g)      430        23
##   Cholesterol Sodium Carbohydrates Dietary.Fiber Protein
## 1         260    750            31             4      17
## 2          25    770            30             4      18
## 3          45    780            29             4      14
## 4         285    860            30             4      21
## 5          50    880            30             4      21
## 6         300    960            31             4      26
##   Vitamin.A....Daily.Value. Vitamin.C....Daily.Value. Calcium....Daily.Value.
## 1                        10                         0                      25
## 2                         6                         0                      25
## 3                         8                         0                      25
## 4                        15                         0                      30
## 5                         6                         0                      25
## 6                        15                         2                      30
##   Iron....Daily.Value. Vitamin.A Vitamin.C Calcium Iron Serving.Size.Oz
## 1                   15        90       0.0     325 2.70             4.8
## 2                    8        54       0.0     325 1.44             4.8
## 3                   10        72       0.0     325 1.80             3.9
## 4                   15       135       0.0     390 2.70             5.7
## 5                   10        54       0.0     325 1.80             5.7
## 6                   20       135       1.8     390 3.60             6.5
```

```
# Identify all columns to normalize (exclude Item and Serving Size Oz)
cols_to_normalize <- setdiff(names(mcdonalds), c("Item", "Serving.Size", "Serving.Size.Oz"))

# Convert values to per oz
mcdonalds[cols_to_normalize] <- lapply(mcdonalds[cols_to_normalize], function(col) col / mcdonalds$Servi

# dropping the remaining columns containing daily value percentages and serving size
cols_to_drop <- c(
  "Vitamin.A....Daily.Value.",
  "Vitamin.C....Daily.Value.",
  "Calcium....Daily.Value.",
  "Iron....Daily.Value.",
  "Serving.Size"
)
mcdonalds <- mcdonalds[, !(names(mcdonalds) %in% cols_to_drop)]


# saving all the column names except "Item" and "Serving Size Oz"
col_names <- setdiff(names(mcdonalds), c("Item","Serving.Size.Oz"))
```

```
col_names
```

```
##  [1] "Calories"      "Total.Fat"     "Cholesterol"   "Sodium"
##  [5] "Carbohydrates" "Dietary.Fiber" "Protein"       "Vitamin.A"
##  [9] "Vitamin.C"     "Calcium"       "Iron"
```

After the completion of the data preprocessing part, our data set has been narrowed down to one response variable *calories* and ten explanatory variables:

1. x1 = $TotalFat$
2. x2 = $Cholestrol$
3. x3 = $Sodium$
4. x4 = $Carbohydrates$
5. x5 = $DietaryFiber$
6. x6 = $Protein$
7. x7 = $VitaminA$
8. x8 = $VitaminC$
9. x9 = $Calcium$
10. x10 = $Iron$

```
# creating new names: y for Calories, x1, x2, ... for the rest
new_names <- c("y", paste0("x", seq_along(col_names[-which(col_names == "Calories")])))

# creating a named vector to rename the columns
name_map <- setNames(new_names, c("Calories", col_names[col_names != "Calories"]))

# renaming the columns
names(mcdonalds)[names(mcdonalds) %in% names(name_map)] <- name_map[names(mcdonalds)[names(mcdonalds) %


# saving all the column names except "Item" and "Serving Size Oz"
col_names <- setdiff(names(mcdonalds), c("Item","Serving.Size.Oz"))
```

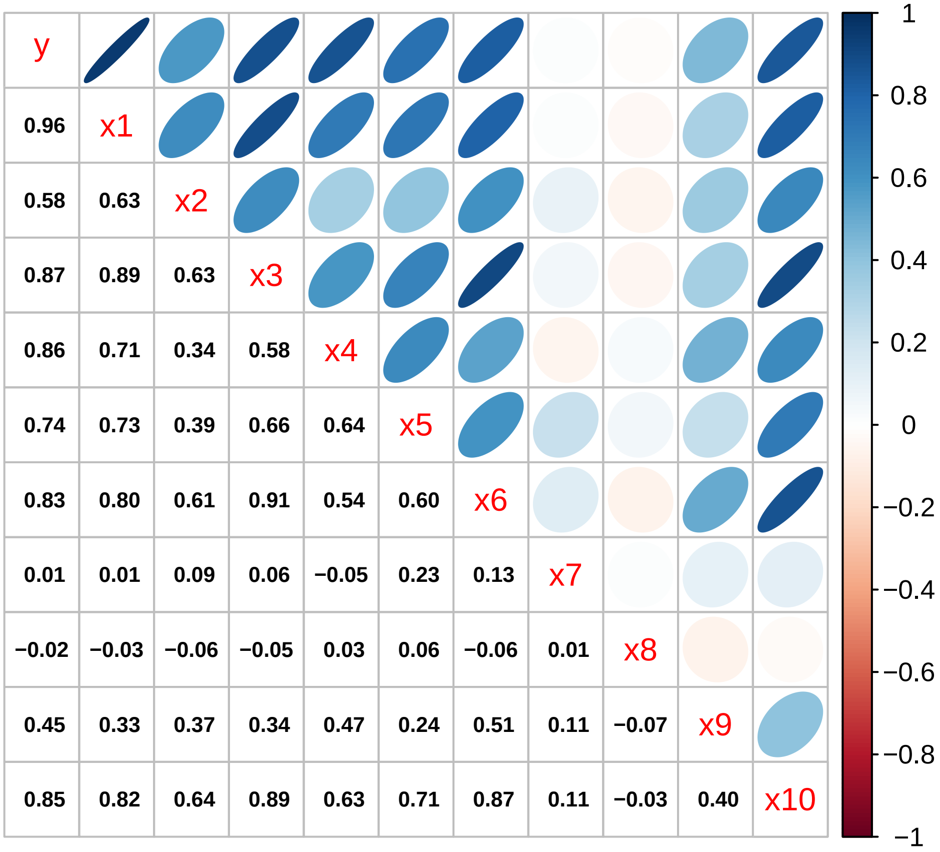
Correlation Coefficients

Since we are looking at linear relationships between the outcome variable (calories) with each explanatory variable, it may be of interest to determine the correlation coefficients between the outcome variable with each explanatory variable.

```
corr_matrix <- cor(mcdonalds[,col_names], use="everything")
round(corr_matrix, 3)
```

```
##          y     x1     x2     x3     x4    x5     x6     x7     x8     x9    x10
## y    1.000  0.960  0.579  0.872  0.862 0.745  0.826  0.012 -0.019  0.448  0.848
## x1   0.960  1.000  0.628  0.890  0.709 0.726  0.802  0.010 -0.035  0.326  0.823
## x2   0.579  0.628  1.000  0.626  0.336 0.394  0.607  0.092 -0.056  0.367  0.642
## x3   0.872  0.890  0.626  1.000  0.581 0.664  0.906  0.056 -0.047  0.339  0.890
## x4   0.862  0.709  0.336  0.581  1.000 0.638  0.538 -0.050  0.033  0.470  0.631
## x5   0.745  0.726  0.394  0.664  0.638 1.000  0.599  0.229  0.057  0.237  0.709
## x6   0.826  0.802  0.607  0.906  0.538 0.599  1.000  0.134 -0.063  0.510  0.868
## x7   0.012  0.010  0.092  0.056 -0.050 0.229  0.134  1.000  0.013  0.107  0.114
## x8  -0.019 -0.035 -0.056 -0.047  0.033 0.057 -0.063  0.013  1.000 -0.067 -0.025
## x9   0.448  0.326  0.367  0.339  0.470 0.237  0.510  0.107 -0.067  1.000  0.403
## x10  0.848  0.823  0.642  0.890  0.631 0.709  0.868  0.114 -0.025  0.403  1.000
```

```
corrplot.mixed(corr_matrix, lower.col = "black", number.cex = .7, upper = "ellipse")
```



Multiple Regression

At this point we have interest in building a model for calories using some combination of the explanatory variables. Using multiple regression one initially build a model with all of the possible explanatory variables. Below is some R output for this Multiple Linear Regression (MLR) analysis.

General Form for a Multiple Regression Model

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \cdots + \beta_{14} X_{14}$$

```
model.1 <- lm(y~x1+x2+x3+x4+x5+x6+x7+x8+x9+x10, data=mcdonalds)

summary(model.1)
```

```
##
## Call:
## lm(formula = y ~ x1 + x2 + x3 + x4 + x5 + x6 + x7 + x8 + x9 +
##     x10, data = mcdonalds)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.49593 -0.29807  0.04731  0.31380  2.39670
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.120801   0.084412  -1.431  0.15368
## x1           9.066437   0.061215 148.109  < 2e-16 ***
## x2          -0.014724   0.005112  -2.880  0.00433 **
## x3           0.002323   0.001659   1.400  0.16282
## x4           4.031077   0.024589 163.937  < 2e-16 ***
## x5          -0.719929   0.275351  -2.615  0.00949 **
## x6           3.776797   0.077650  48.639  < 2e-16 ***
## x7           0.001713   0.001739   0.985  0.32571
## x8          -0.002740   0.004894  -0.560  0.57608
## x9          -0.001219   0.003086  -0.395  0.69325
## x10          0.922310   0.455764   2.024  0.04409 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6007 on 245 degrees of freedom
## Multiple R-squared:  0.9996, Adjusted R-squared:  0.9996
## F-statistic: 6.146e+04 on 10 and 245 DF,  p-value: < 2.2e-16
```

Equation of the Model with all of the Explanatory Variables

(Note: This is referred to at the Full Model)

$$Y = -0.120 + 9.066X_1 - 0.014X_2 + 0.002X_3 + 4.031X_4 + \cdots + 0.922X_{10}$$

Coefficient of Determination

Interpretation: **99.96** % of the variability in calories is accounted for in this model, (i.e., is accounted for in the model between calories and the thirteen explanatory variables).

Test for the Significance of the Model:

Ho: None of the explanatory variables is a linear predictor of calories (i.e., the model is not significant or is not useful in predicting the response)

Ha: At least one of the explanatory variables is a significant linear predictor of calories (i.e., the model is significant or at least some portion of the model is useful in predicting the response)

Test statistic: F* = 6.146e+04

P-value: < 2.2e-16

Conclusion: Reject Ho in favor of Ha. There is sufficient evidence to conclude that at least one of the explanatory variables is a significant linear predictor of calories (i.e., the model is significant or at least some portion of the model is useful in predicting the response)

Further Analysis

Since at least one of the independent variables is significant, we do further analysis to determine which one(s) is/are significant.

Test for an Individual Predictor in this Model:

Ho: With x1, x2, x3, x4, x5, x6, x7, x8, and x10 in the model, x9 is not a linear predictor of y

Ha: With x1, x2, x3, x4, x5, x6, x7, x8, and x10 in the model, x9 is a significant linear predictor of y

Test statistic: t* = -0.395

P-value: 0.69325

Conclusion: Fail to reject Ho. There is insufficient evidence to conclude that with x1, x2, x3, x4, x5, x6, x7, x8, and x10 in the model, x9 is not a linear predictor of y

In the above model, x9 has the largest p-value and thus is the least significant. We could remove it from the model and rerun the analysis. Then we could test for significance of another independent variable. We could continue this process until only significant variables are left. This method for identifying the best model is referred to as **Backward Selection**.

Some selected output for the **Backward Selection** procedure:

```r
model.2 <- lm(y~x1+x2+x3+x4+x5+x6+x7+x8+x10, data=mcdonalds)

summary(model.2)
```

```
##
## Call:
## lm(formula = y ~ x1 + x2 + x3 + x4 + x5 + x6 + x7 + x8 + x10,
##     data = mcdonalds)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.48249 -0.30917  0.04845  0.30838  2.38516
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.125508   0.083422  -1.504  0.13374
## x1           9.071362   0.059827 151.625  < 2e-16 ***
## x2          -0.015260   0.004920  -3.102  0.00215 **
## x3           0.002475   0.001611   1.536  0.12593
## x4           4.026415   0.021533 186.989  < 2e-16 ***
## x5          -0.709413   0.273590  -2.593  0.01008 *
## x6           3.761373   0.066997  56.142  < 2e-16 ***
## x7           0.001667   0.001733   0.962  0.33681
## x8          -0.002618   0.004875  -0.537  0.59181
## x10          0.936844   0.453496   2.066  0.03989 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5997 on 246 degrees of freedom
## Multiple R-squared:  0.9996, Adjusted R-squared:  0.9996
## F-statistic: 6.852e+04 on 9 and 246 DF,  p-value: < 2.2e-16
```

```
model.3 <- lm(y~x1+x2+x3+x4+x5+x6+x7+x10, data=mcdonalds)

summary(model.3)
```

```
##
## Call:
## lm(formula = y ~ x1 + x2 + x3 + x4 + x5 + x6 + x7 + x10, data = mcdonalds)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.47965 -0.30666  0.05207  0.30998  2.39673
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.128464   0.083120  -1.546  0.12350
## x1           9.072878   0.059675 152.039  < 2e-16 ***
## x2          -0.015233   0.004913  -3.101  0.00215 **
## x3           0.002468   0.001609   1.534  0.12642
## x4           4.025910   0.021481 187.415  < 2e-16 ***
## x5          -0.722892   0.272043  -2.657  0.00839 **
## x6           3.763033   0.066829  56.308  < 2e-16 ***
## x7           0.001671   0.001730   0.966  0.33515
## x10          0.935256   0.452832   2.065  0.03993 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5988 on 247 degrees of freedom
## Multiple R-squared:  0.9996, Adjusted R-squared:  0.9996
## F-statistic: 7.731e+04 on 8 and 247 DF,  p-value: < 2.2e-16
```

```
model.4 <- lm(y~x1+x2+x3+x4+x5+x6+x10, data=mcdonalds)

summary(model.4)
```

```
##
## Call:
## lm(formula = y ~ x1 + x2 + x3 + x4 + x5 + x6 + x10, data = mcdonalds)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.49076 -0.29629  0.05666  0.30846  2.36800
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.109634   0.080790  -1.357  0.17601
## x1           9.064833   0.059082 153.427  < 2e-16 ***
## x2          -0.014810   0.004892  -3.027  0.00273 **
## x3           0.002292   0.001599   1.434  0.15293
## x4           4.022544   0.021194 189.799  < 2e-16 ***
## x5          -0.630767   0.254732  -2.476  0.01395 *
## x6           3.775812   0.065497  57.648  < 2e-16 ***
## x10          0.935771   0.452770   2.067  0.03979 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 0.5987 on 248 degrees of freedom
## Multiple R-squared:  0.9996, Adjusted R-squared:  0.9996
## F-statistic: 8.838e+04 on 7 and 248 DF,  p-value: < 2.2e-16
```

```r
model.5 <- lm(y~x1+x2+x4+x5+x6+x10, data=mcdonalds)

summary(model.5)
```

```
##
## Call:
## lm(formula = y ~ x1 + x2 + x4 + x5 + x6 + x10, data = mcdonalds)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.54519 -0.28767  0.03628  0.30153  2.51553
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.116425   0.080822  -1.441  0.15098
## x1           9.112407   0.048987 186.018  < 2e-16 ***
## x2          -0.015342   0.004889  -3.138  0.00190 **
## x4           4.015530   0.020665 194.315  < 2e-16 ***
## x5          -0.642730   0.255134  -2.519  0.01239 *
## x6           3.822813   0.056822  67.277  < 2e-16 ***
## x10          1.166167   0.424186   2.749  0.00641 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6 on 249 degrees of freedom
## Multiple R-squared:  0.9996, Adjusted R-squared:  0.9996
## F-statistic: 1.027e+05 on 6 and 249 DF,  p-value: < 2.2e-16
```

Final Model using Backward Selection:

$$Y = -0.116 + 9.112X_1 - 0.015X_2 + 4.015X_4 - 0.642X_5 + 3.822X_6 + 1.166X_{10}$$

Coefficient of Determination

(Assessing the fit of the model) 99.96 % of the variability in the calories is accounted for in this multiple linear regression model.

Prediction:

For a sample that has a total fat (x1) of 4, a cholesterol (x2) of 6, a carbohydrate (x4) of 8.2, a dietary fiber (x5) of 0.4, a protein (x6) of 3.4, and a iron (x10) of 0.54, we predict the calories (y) to be 82.0.

Estimation:

For samples that have a total fat (x1) of 4, a cholesterol (x2) of 6, a carbohydrate (x4) of 8.2, a dietary fiber (x5) of 0.4, a protein (x6) of 3.4, and a iron (x10) of 0.54, we predict the calories (y) to be 82.0.

Interpretation of Partial Slopes (B-weights or Coefficients):

For a fixed total fat (x1), a fixed cholesterol (x2), a fixed carbohydrates (x4), a fixed dietary fiber (x5), and a fixed protein (x6), as the iron increases by 1, the calories increases by 1.166.

For a fixed total fat (x1), a fixed cholesterol (x2), a fixed carbohydrates (x4), a fixed dietary fiber (x5), and a fixed iron (x10), as the protein increases by 1, the calories increases by 3.822.

For a fixed total fat (x1), a fixed cholesterol (x2), a fixed carbohydrates (x4), a fixed protein (x6), and a fixed iron (x10), as the dietary fiber increases by 1, the calories decreases by 0.642.

For a fixed total fat (x1), a fixed cholesterol (x2), a fixed dietary fiber (x5), a fixed protein (x6), and a fixed iron (x10), as the carbohydrates increases by 1, the calories increases by 4.015.

For a fixed total fat (x1), a fixed carbohydrates (x4), a fixed dietary fiber (x5), a fixed protein (x6), and a fixed iron (x10), as the cholesterol increases by 1, the calories decreases by 0.015.

For a fixed cholesterol (x2), a fixed carbohydrates (x4), a fixed dietary fiber (x5), a fixed protein (x6) and a fixed iron (x10), as the total fat increases by 1, the calories increases by 9.112.

Confidence Intervals for the Coefficients

```r
confint(model.5, level=0.95)
```

```
##                   2.5 %       97.5 %
## (Intercept) -0.27560643  0.042756083
## x1           9.01592622  9.208888169
## x2          -0.02497068 -0.005713673
## x4           3.97482982  4.056231109
## x5          -1.14522649 -0.140233825
## x6           3.71090073  3.934725261
## x10          0.33071590  2.001617540
```

We are 95% confident that for a fixed total fat (x1), a fixed cholesterol (x2), a fixed carbohydrates (x4), a fixed dietary fiber (x5), and a fixed protein (x6), as the iron increases by 1, the calories increases between 0.331 and 2.001.

We are 95% confident that for a fixed total fat (x1), a fixed cholesterol (x2), a fixed carbohydrates (x4), a fixed dietary fiber (x5), and a fixed iron (x10), as the protein increases by 1, the calories increases between 3.711 and 3.935.

We are 95% confident that for a fixed total fat (x1), a fixed cholesterol (x2), a fixed carbohydrates (x4), a fixed protein (x6), and a fixed iron (x10), as the dietary fiber increases by 1, the calories decreases between -1.145 and -0.140.

We are 95% confident that for a fixed total fat (x1), a fixed cholesterol (x2), a fixed dietary fiber (x5), a fixed protein (x6), and a fixed iron (x10), as the carbohydrates increases by 1, the calories increases between 3.975 and 4.056.

We are 95% confident that for a fixed total fat (x1), a fixed carbohydrates (x4), a fixed dietary fiber (x5), a fixed protein (x6), and a fixed iron (x10), as the cholesterol increases by 1, the calories decreases between -0.025 and -0.006.

We are 95% confident that for a fixed cholesterol (x2), a fixed carbohydrates (x4), a fixed dietary fiber (x5), a fixed protein (x6) and a fixed iron (x10), as the total fat increases by 1, the calories increases between 9.015 and 9.209.

Residual Analysis

This is checking the assumptions that need to be satisfied before it is appropriate to perform inference from a multiple regression model.

1. The random errors are independent of each other.
2. The random errors are normally distributed

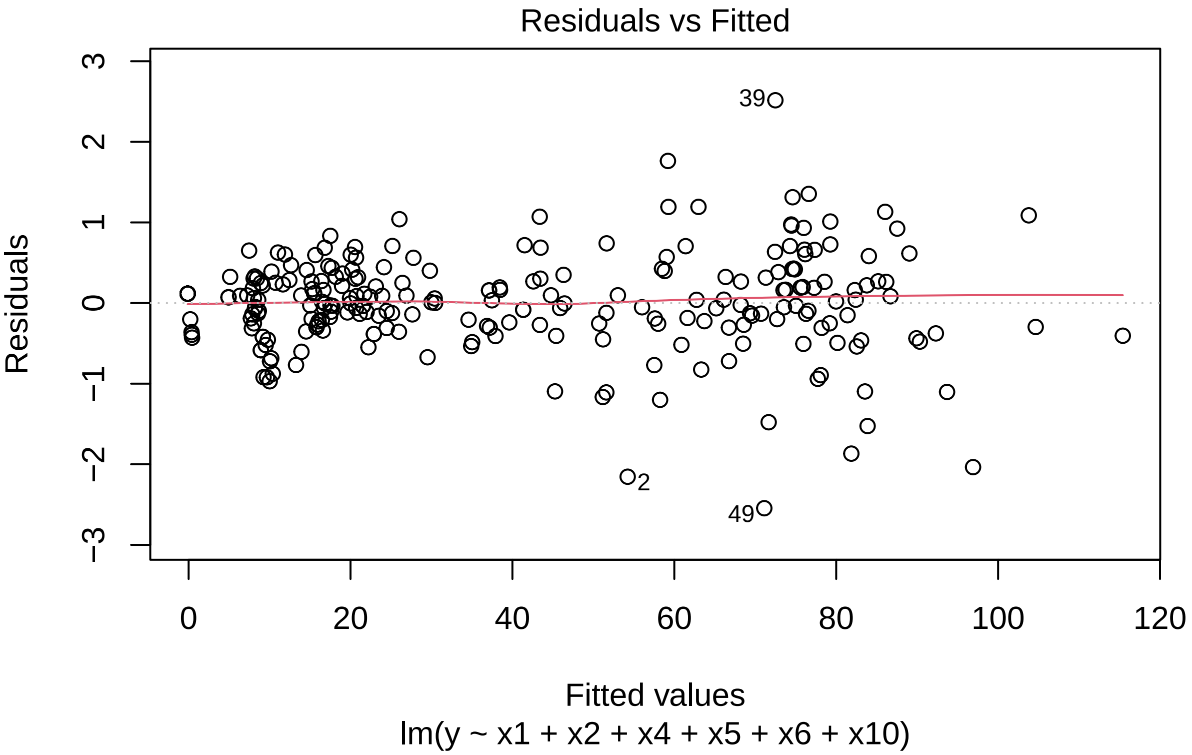
3. The random errors have constant variance (homoscedasticity)

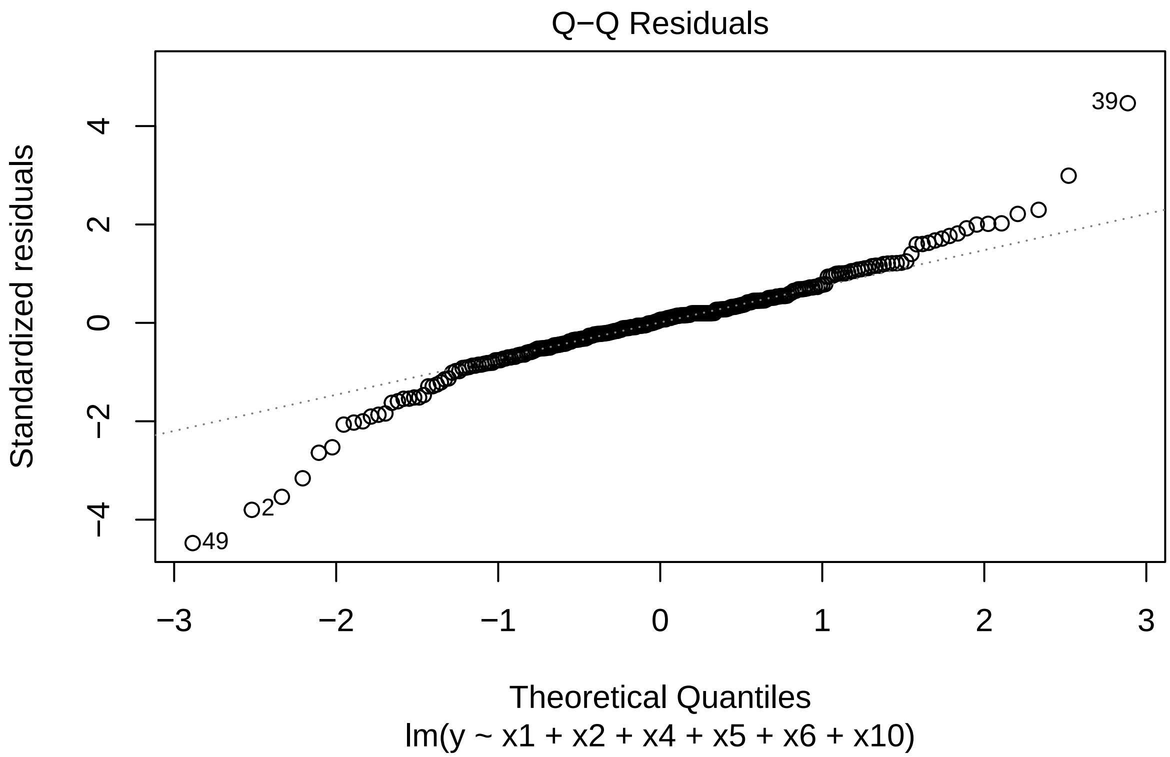
(Note: There is a lot more to residual analysis than just these things, but to keep things focused on the inferential aspects of linear regression we will just do this quick check.)

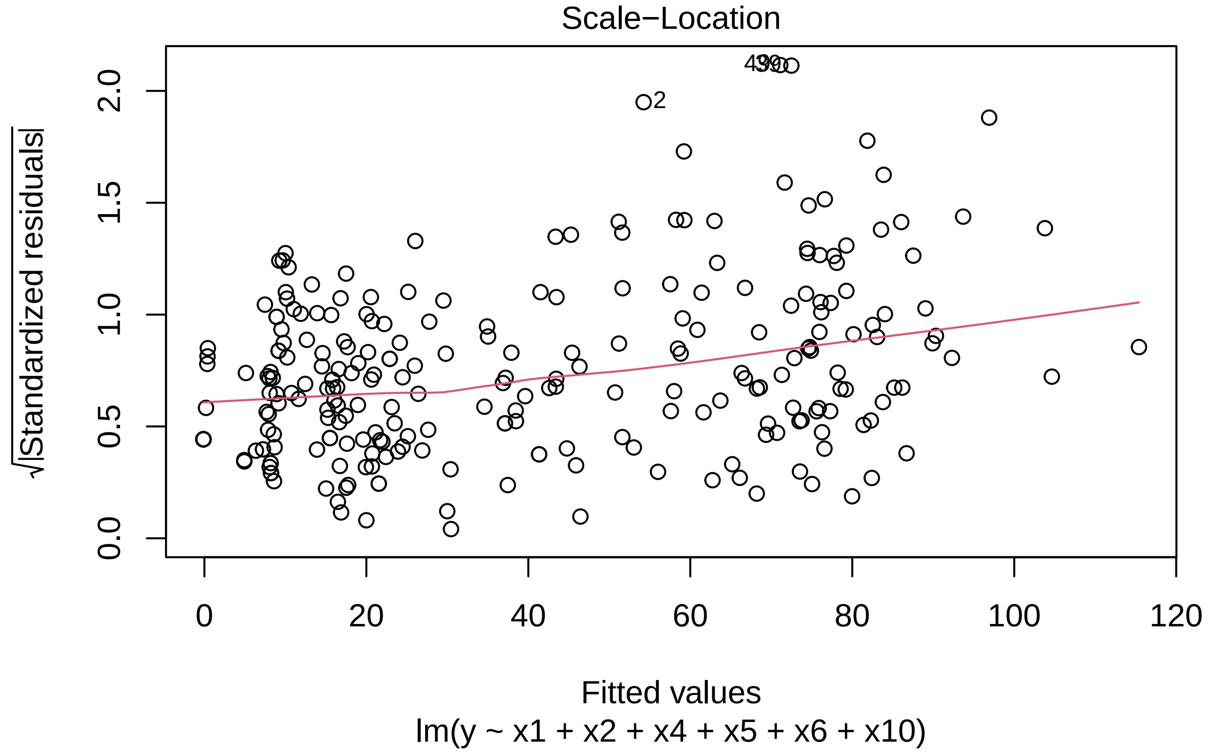
```
bptest(y~x1+x2+x4+x5+x6+x10, varformula = ~ fitted.values(model.5), studentize=TRUE, data=mcdonalds)
```

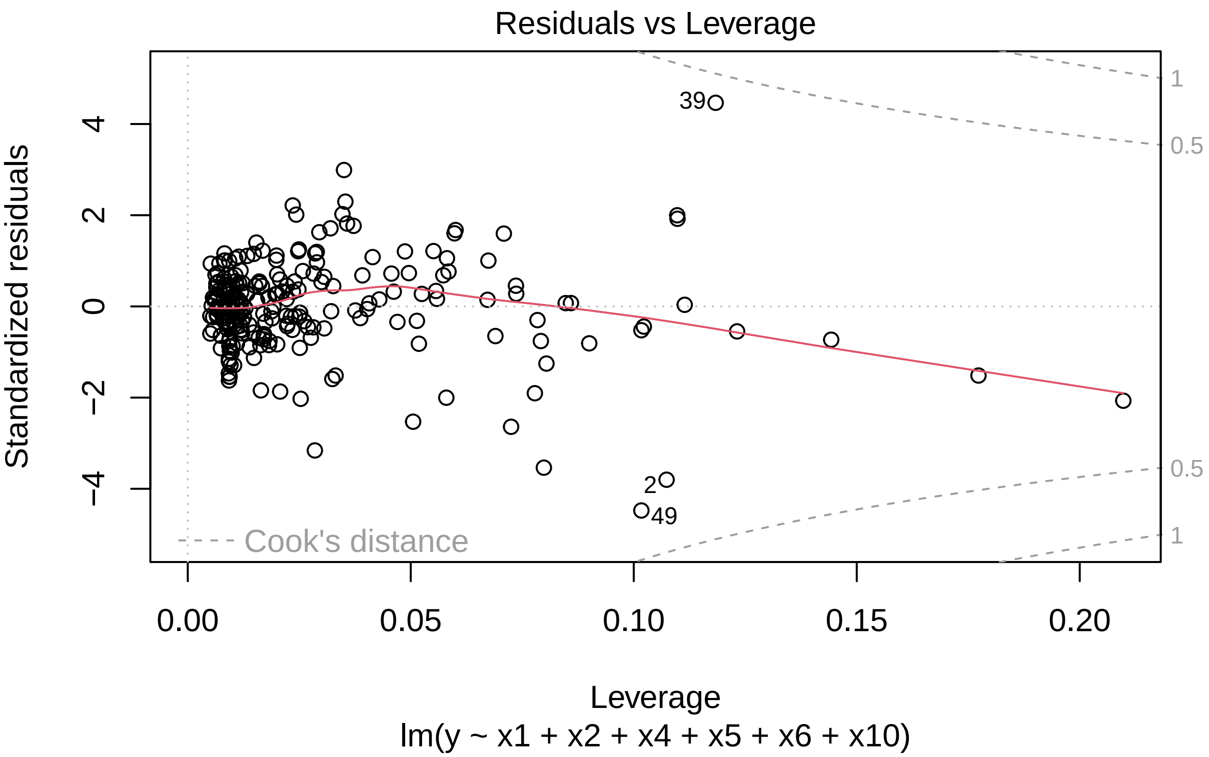
```
##
##  studentized Breusch-Pagan test
##
## data:  y ~ x1 + x2 + x4 + x5 + x6 + x10
## BP = 21.68, df = 1, p-value = 3.222e-06
```

```
#oldpar <- par(oma=c(0,0,3,0), mfrow=c(2,2))
plot(model.5)
```




Q−Q Residuals
Standardized residuals
4
2
0
−2
−4
39
2
49
−3
−2
−1
0
1
2
3
Theoretical Quantiles
lm(y ~ x1 + x2 + x4 + x5 + x6 + x10)


Scale−Location
√|Standardized residuals|
2.0
1.5
1.0
0.5
0.0
4390
2
0
20
40
60
80
100
120
Fitted values
lm(y ~ x1 + x2 + x4 + x5 + x6 + x10)



```
#par(oldpar)
```

Ho: The variability of the residuals is constant (homoscedasticity)

Ha: The variability of the residuals is not constant (heteroscedasticity)

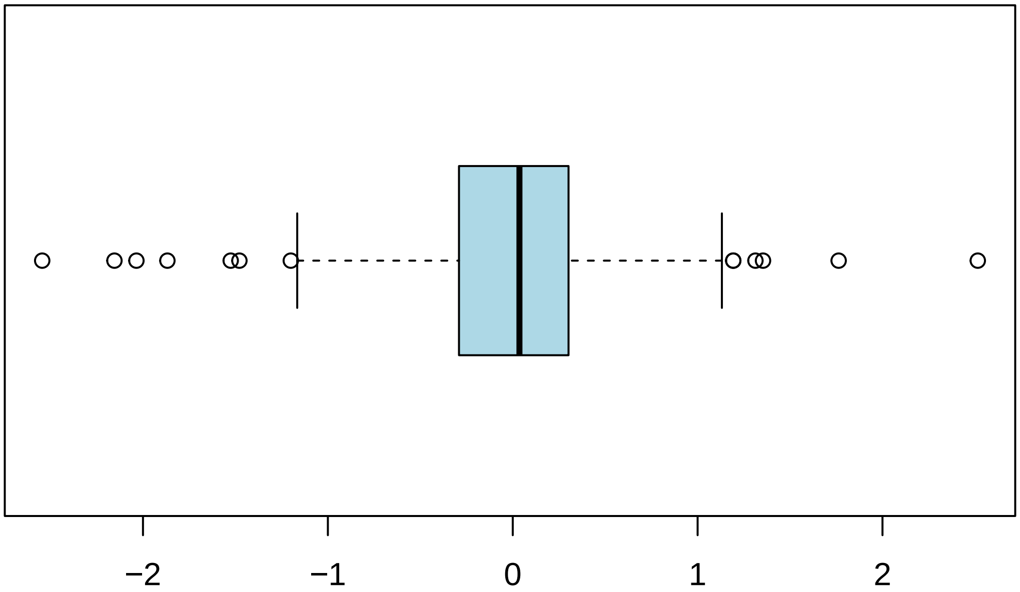
Test statistic: $\chi^2_{BP} = 21.68$

P-value: 3.222e-06

Conclusion (at the .05 level): Reject Ho in favor of Ha. There is sufficient evidence to conclude that the variability of the residuals is not constant (heteroscedasticity).

```
# Examination of the distribution of the residuals

boxplot(model.5$residuals, col="lightblue", horizontal = TRUE)
```



```
shapiro.test(model.5$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  model.5$residuals
## W = 0.94993, p-value = 1.066e-07
```

Ho: The residuals follow a normal distribution

Ha: The residuals do not follow a normal distribution

Test statistic: W* = 0.94993

P-value: 1.066e-07

Conclusion (at the .05 level): Reject Ho in favor of Ha. There is sufficient evidence to conclude that the residuals do not follow a normal distribution.

In multiple linear regression, there are several assumption: the residuals (errors) should be normally distributed and should have constant variance. However, in our case, residuals do not follow a normal distribution and have non-constant variance. This suggests that our model may not be a good fit for the Macdonald's menu data.We could explore alternative regression techniques or include interaction terms in the model to better capture the relationships in the data.