Medical Cost Prediction

Data Analysis 2 - WS22/23

Philipp Habicht - 621854

2023-03-31

Contents

1. Introduction **3**

2. Data Set **3**

2.1 Origin of Data Set 3

2.2 Libraries 3

2.3 Data Summary 5

2.4 Data Cleaning and Preprocessing 6

2.5 Explanatory Data Analysis (EDA) 7

2.5.1 Target Variable 8

2.5.2 Sex 8

2.5.3 Smoker 11

2.5.4 Region 11

2.5.5 Children 13

2.5.6 BMI & Age 16

2.5.7 Scatterplots BMI, Age & Children 16

2.5.8 Correlation Matrix 20

3. Regression Analysis **21**

3.1 Regression Preprocessing 21

3.2 Linear Regression 21

3.3 Polynomial Regression 32

3.3.1 Polynomial Regression (2nd degree) 33

3.3.1 Polynomial Regression (3nd degree) 40

4. Conclusion **50**

5. Bibliography **51**

List of Figures

1 Fig: 2.1: Target Variable Charges 9
2 Fig: 2.2: Boxplot Charges-Sex 10
3 Fig: 2.3:Boxplot Charges-Smoker 12
4 Fig: 2.4: Boxplot Charges-Regions 14
5 Fig: 2.5: Boxplot Charges-Children 15
6 Fig: 2.6: Distribution Sex-BMI 17
7 Fig: 2.7: Scatterplots Charges- Children, BMI, Smoker 18
8 Fig: 2.7: Scatterplots Charges- Children, BMI, Smoker 19
9 Fig: 2.7: Scatterplots Charges- Children, BMI, Smoker 19
10 Fig: 2.8: Correlation Matrix 20
11 Fig: 3.1: Cross-Validation Base-Model 25
12 Fig: 3.2: Leverage Plot Base-Model 27
13 Fig: 3.3: Residual Plots Base-Model 27
14 Fig: 3.4: Cross-Validation Step-Model 30
15 Fig: 3.5: Residual Plots Step-Model 32
16 Fig: 3.6: Cross-Validation Poly2-Model 37
17 Fig: 3.7: Leverage Plot Ploy2-Model 38
18 Fig: 3.8: Residual Plots Poly2-Model 39
19 Fig: 3.9: Cross-Validation Poly3-Model 47
20 Fig: 3.10: Leverage Plot Poly3-Model 48
21 Fig: 3.11: Residual Plots Poly3-Model 49

1. Introduction

The cost of health insurance is a critical concern for many individuals and families, and accurately predicting insurance charges can provide valuable insights for policyholders and insurance providers alike. The primary objective of this research study is to examine the accuracy of medical expense prediction models based on several demographic and lifestyle factors. In more deatil, ultimate goal is to determine how accurately we can predict insurance costs based on variables available in the medical expense data set, including age, gender, BMI, number of children, smoking habits, and geographic region.

In this study, I employ both statistical techniques to construct and compare linear and polynomial regression predictive models for medical expenditures. The evaluation process relies on 10-fold cross-validation, utilizing relevant metrics such as R-squared and root mean squared error to assess model accuracy. To mitigate the risk of overfitting, we rigorously perform cross-validation on both the training and testing data sets. This approach ensures that the models demonstrate robust generalization capabilities when applied to previously unseen data. Furthermore, I investigate the relationships between predictor variables and medical costs through the application of descriptive statistics and data visualization methods.

2. Data Set

The medical cost data set originates from the field of medical research and health economics. The data set contains information on the medical costs of individuals based on their age, gender, BMI, number of children, smoking habits, and geographic region in the United States. However, the medical cost data set does not explicitly mention whether the costs are yearly or summed up over a lifetime. The data set only provides information on the medical costs of individuals based on various factors. This data set can be used to study the factors that influence medical costs and to build predictive models of medical expenditures.

2.1 Origin of Data Set

The medical cost data set was originally sourced from the United States government's National Health Interview Survey, which is conducted by the National Center for Health Statistics (NCHS), part of the Centers for Disease Control and Prevention (CDC) in the year 2013. The data set was made available on Kaggle, a platform for data science competitions, and has since been used in research studies and machine learning projects (Choi M., 2018).

2.2 Libraries

To begin with, all necessary libraries are imported for further analysis.

```
library(readxl)
library(caret)
```

```
## Lade nötiges Paket: ggplot2
```

```
## Lade nötiges Paket: lattice
```

```
library(magrittr)
library(ggcorrplot)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
```

```
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.3.0      v stringr 1.5.0
## v readr   2.1.2      v forcats 0.5.2
## v purrr   1.0.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x tidyr::extract()   masks magrittr::extract()
## x dplyr::filter()    masks stats::filter()
## x dplyr::lag()       masks stats::lag()
## x purrr::lift()      masks caret::lift()
## x purrr::set_names() masks magrittr::set_names()
```

```
library(dplyr)
library(ggplot2)
library(gridExtra)
```

```
##
## Attache Paket: 'gridExtra'
##
## Das folgende Objekt ist maskiert 'package:dplyr':
##
##     combine
```

```
library(plotly)
```

```
##
## Attache Paket: 'plotly'
##
## Das folgende Objekt ist maskiert 'package:ggplot2':
##
##     last_plot
##
## Das folgende Objekt ist maskiert 'package:stats':
##
##     filter
##
## Das folgende Objekt ist maskiert 'package:graphics':
##
##     layout
```

```
library(ggcorrplot)
library(car)
```

```
## Lade nötiges Paket: carData
##
## Attache Paket: 'car'
##
## Das folgende Objekt ist maskiert 'package:dplyr':
##
##     recode
##
## Das folgende Objekt ist maskiert 'package:purrr':
##
##     some
```

```
library(lmtest)
```

```
## Lade nötiges Paket: zoo
##
## Attache Paket: 'zoo'
##
## Die folgenden Objekte sind maskiert von 'package:base':
##
##     as.Date, as.Date.numeric
```

```
library(glmnet)
```

```
## Lade nötiges Paket: Matrix
##
## Attache Paket: 'Matrix'
##
## Die folgenden Objekte sind maskiert von 'package:tidyr':
##
##     expand, pack, unpack
##
## Loaded glmnet 4.1-7
```

```
library(GGally)
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

2.3 Data Summary

```
# Read in data set
data <- read.csv("/Users/hawk/Documents/HU_Three/DS2/insurance.csv", sep = ",")

# Check dimensionality of data
print(dim(data))
```

```
## [1] 1338    7
```

```
# Overview
head(data, 5)
```

```
##   age    sex    bmi children smoker    region   charges
## 1  19 female 27.900        0    yes southwest 16884.924
## 2  18   male 33.770        1     no southeast  1725.552
## 3  28   male 33.000        3     no southeast  4449.462
## 4  33   male 22.705        0     no northwest 21984.471
## 5  32   male 28.880        0     no northwest  3866.855
```

```
# Data types
str(data)
```

```
## 'data.frame':    1338 obs. of  7 variables:
##  $ age     : int  19 18 28 33 32 31 46 37 37 60 ...
##  $ sex     : chr  "female" "male" "male" "male" ...
##  $ bmi     : num  27.9 33.8 33 22.7 28.9 ...
##  $ children: int  0 1 3 0 0 0 1 3 2 0 ...
##  $ smoker  : chr  "yes" "no" "no" "no" ...
##  $ region  : chr  "southwest" "southeast" "southeast" "northwest" ...
##  $ charges : num  16885 1726 4449 21984 3867 ...
```

```
# Summary
summary(data)
```

```
##       age            sex                 bmi           children
##  Min.   :18.00   Length:1338        Min.   :15.96   Min.   :0.000
##  1st Qu.:27.00   Class :character   1st Qu.:26.30   1st Qu.:0.000
##  Median :39.00   Mode  :character   Median :30.40   Median :1.000
##  Mean   :39.21                      Mean   :30.66   Mean   :1.095  
##  3rd Qu.:51.00                      3rd Qu.:34.69   3rd Qu.:2.000  
##  Max.   :64.00                      Max.   :53.13   Max.   :5.000  
##     smoker             region             charges     
##  Length:1338        Length:1338        Min.   : 1122  
##  Class :character   Class :character   1st Qu.: 4740  
##  Mode  :character   Mode  :character   Median : 9382  
##                                        Mean   :13270  
##                                        3rd Qu.:16640  
##                                        Max.   :63770  
```

```
# Character variables
table(data$region)
```

```
## 
## northeast northwest southeast southwest 
##       324       325       364       325 
```

```
table(data$smoker)
```

```
## 
##   no  yes 
## 1064  274 
```

In this particular block of code, the CSV file containing the data is imported into R. Upon examination, it is evident that the data set comprises 1338 observations and seven distinct variables. To gain an initial understanding of the data, the first five observations are considered, along with the data types of the columns. The data set includes the following variables: age, sex, BMI, children, smoker, region, and charges. Age, BMI, children, and charges are classified as numerical variables, while sex, smoker, and region are categorized as character variables. In more detail, age and children are identified as integer variables. The mean age of the individuals is 39.21 years, and the mean BMI is 30.66. The majority of the individuals have at least one child (mean = 1.095), and there are more non-smokers than smokers. The data set includes individuals from four different regions, with an approximately equal number of individuals in each region. The charges column displays the cost of medical care for each individual, with a wide range from $1,122 to $63,770.

2.4 Data Cleaning and Preprocessing

```
# Check missing values in data
colSums(is.na(data))
```

```
##      age      sex      bmi children   smoker   region  charges 
##        0        0        0        0        0        0        0 
```

```
# Check duplicates
data[duplicated(data),]
```

```
##     age  sex   bmi children smoker    region  charges
## 582  19 male 30.59        0     no northwest 1639.563
```

```
# Found one duplicate and will drop it
data <- data %>% distinct()

# Transform categorical variable region to dummy variable
dummy_reg <- model.matrix(~ region - 1, data = data)

# Convert the matrix to a data frame
dummy_reg_df <- as.data.frame(dummy_reg)
```

```r
# Join the original data frame with the dummy variables data frame
data <- cbind(data, dummy_reg_df)

# Remove the original categorical column
data <- data %>% select(-region)

# Transform binary variables sex and smoker to 0 and 1
data <- data %>%
  mutate(sex = ifelse(sex == "female", 0, ifelse(sex == "male", 1, sex)))
data <- data %>%
  mutate(smoker = ifelse(smoker == "no", 0, ifelse(smoker == "yes", 1, smoker)))

# Change data types to numeric
data <- sapply(data, as.numeric)

# Data Frame
data <- data.frame(data)

# Overview
head(data, 5)
```

```
##   age sex    bmi children smoker   charges regionnortheast regionnorthwest
## 1  19   0 27.900        0      1 16884.924               0               0
## 2  18   1 33.770        1      0  1725.552               0               0
## 3  28   1 33.000        3      0  4449.462               0               0
## 4  33   1 22.705        0      0 21984.471               0               1
## 5  32   1 28.880        0      0  3866.855               0               1
##   regionsoutheast regionsouthwest
## 1               0               1
## 2               1               0
## 3               1               0
## 4               0               0
## 5               0               0
```

In this section, several data preprocessing steps are performed on the data set. Initially, an examination for missing values is conducted across each variable. Fortunately, the data set does not contain any missing values. Next, a duplicated row with identical values to another observation is identified. To ensure the uniqueness of each observation and avoid repetition, the duplicate row is removed. Additionally, the categorical variable "region" is transformed into dummy variables, while binary variables "sex" and "smoker" are converted to 0 and 1, fulfilling essential preprocessing requirements. This process ensures that machine learning algorithms and statistical models, which require numerical input data, can effectively process the information. Simultaneously, data types are modified to numeric to maintain consistency and enable accurate computation during analysis (Tsai et al., 2018). Consequently, the first five rows of the modified data frame are displayed, reflecting these transformations.

2.5 Explanatory Data Analysis (EDA)

Explanatory Data Analysis (EDA) is an important process in the field of data science that involves exploring and summarizing the main characteristics of a data set to better understand the underlying patterns and relationships in the data, especially to the target variable 'charges' (Tukey, 1977).

2.5.1 Target Variable

```r
# Calculate the mean
y_mean <- mean(data$charges)

# Create a histogram of the 'charge' variable
histogram_plot <- ggplot(data, aes(x = charges)) +
  geom_histogram(binwidth = 1000, fill = "blue", alpha = 0.6) +
  geom_vline(aes(xintercept = y_mean),
             color = "red", linetype = "dashed", linewidth = 0.5) +
  theme_minimal() +
  labs(title = "Histogram of Charges", x = "Charges", y = "Frequency") +
  annotate("text", x = y_mean,
           y = Inf,
           label = paste("Mean =", round(y_mean)),
           vjust = 2, hjust = -0.3, color = "red") +
  theme(plot.title = element_text(hjust = 0.5))

# Create a density plot of the 'charge' variable
density_plot <- ggplot(data, aes(x = charges)) +
  geom_density(fill = "blue", alpha = 0.6) +
  theme_minimal() +
  labs(title = "Density Plot of Charges", x = "Charges", y = "Density") +
  theme(plot.title = element_text(hjust = 0.5))

# Display the two plots side by side
grid.arrange(histogram_plot, density_plot, ncol = 2)
```

First, a histogram and density plot is created for the target variable “charges” in the data set, along with a vertical line indicating the mean value of the variable, which is $13.279. The histogram shows a right skewed distribution with a long tail on the right side and a kink around $38,000. Further analysis with categorical characteristics may be needed to investigate this.

2.5.2 Sex

```r
# Calculate the mean charges and sample sizes for each sex
mean_charges <- data %>%
  group_by(sex) %>%
  summarize(mean_charges = mean(charges), n = n())

# Create a boxplot of the 'charge' variable separated by sex
boxplot_sex <- ggplot(data, aes(x = factor(sex,
    labels = c(paste0("Women (n = ", mean_charges$n[1], ")"),
               paste0("Men (n = ", mean_charges$n[2], ")"))),
    y = charges, fill = factor(sex, labels = c("Women", "Men")))) +
  geom_boxplot(alpha = 0.6) +
  stat_summary(fun = mean, geom = "point", shape = 4, size = 2, color = "red") +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5), legend.position = "none") +
  labs(title = "Boxplot of Charges by Sex", x = "Sex", y = "Charges") +
  scale_fill_manual(values = c("Women" = "pink", "Men" = "skyblue"), name = "Sex") +
  guides(fill = guide_legend(override.aes = list(fill = c("pink", "skyblue")))) +
  annotate("text", x = c(1, 2), y = mean_charges$mean_charges,
           label = paste("Mean:", round(mean_charges$mean_charges)),
```

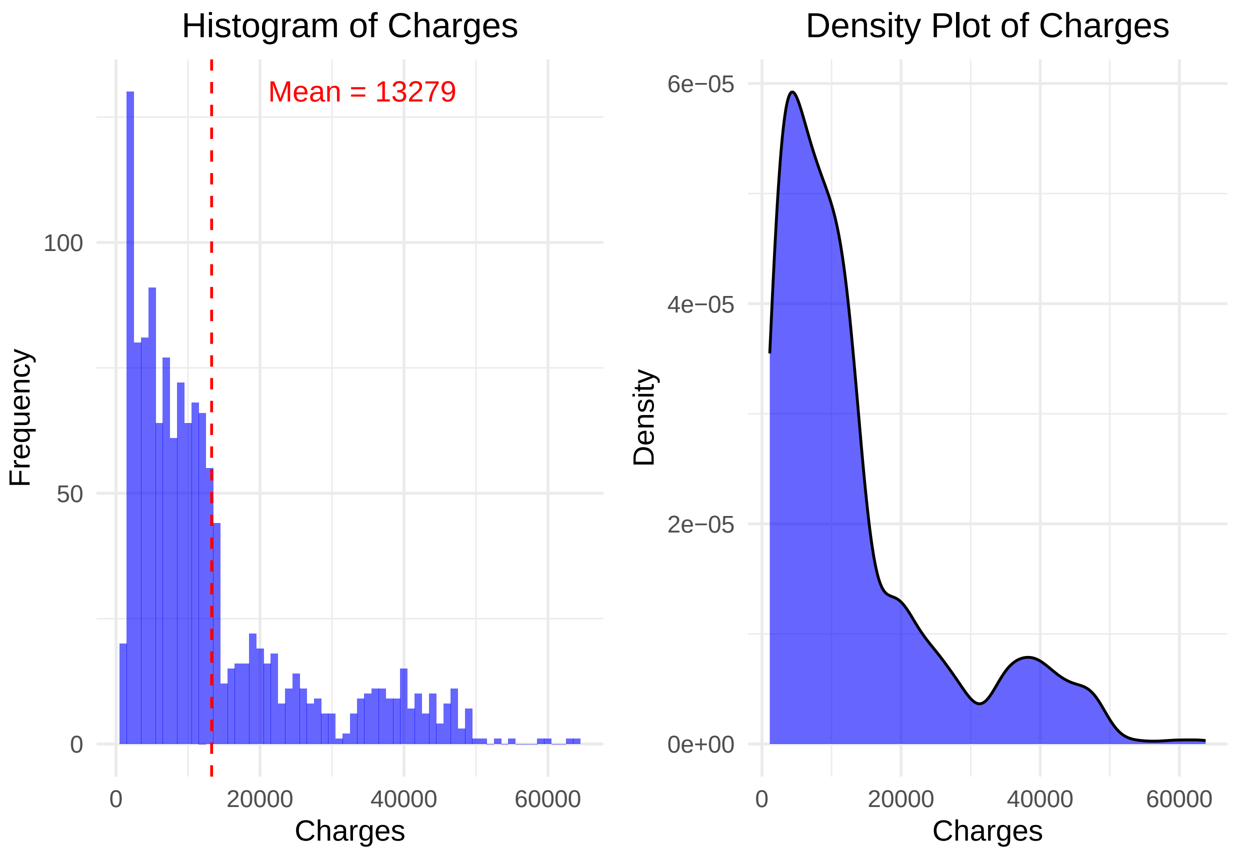


Figure 1: Fig: 2.1: Target Variable Charges

```
          color = "red", size = 3, vjust = 1.7)

# Display the boxplot
boxplot_sex
```

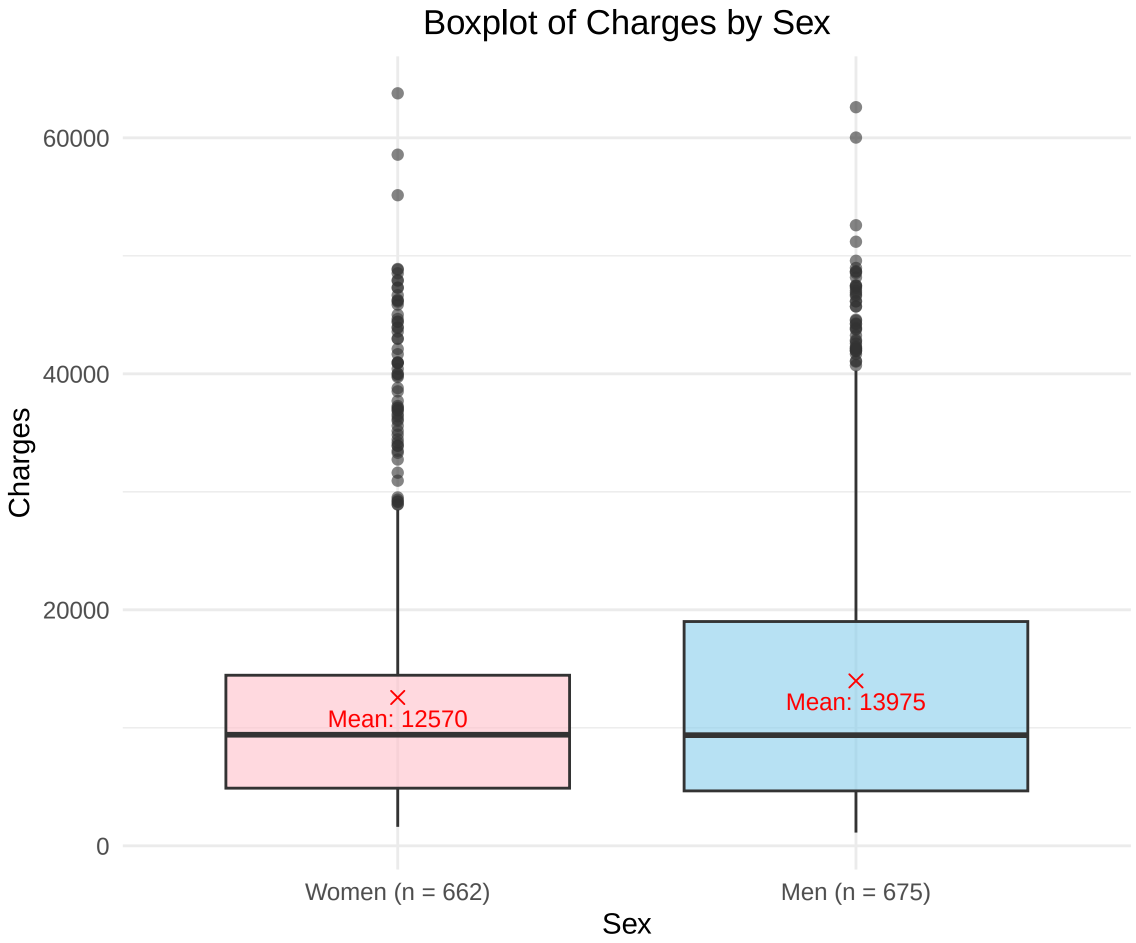


Figure 2: Fig: 2.2: Boxplot Charges-Sex

Second, all variables are considered in the connection with the target variable 'charges'. In particular, boxplots are created for the binary and dummy variables.

When considering the number of two genders of men (n = 662) and women (n = 675), there is virtually no difference in terms of frequency. However, it is evident from the boxplot that although the median of both groups is almost the same, men have more outliers at the top, which is also reflected in the mean value. The mean for women is $12,570 and for men is $13,975, which means that on average men have slightly higher insurance expenditures on average.

2.5.3 Smoker

```r
# Calculate the means for each group
mean_charges <- data %>%
  group_by(smoker) %>%
  summarize(mean_charges = mean(charges), n = n())

# Create a boxplot of the 'charge' variable separated by smoker
boxplot_smoker <- ggplot(data, aes(x = factor(smoker,
    labels = c(paste0("Non-smoker (n = ", mean_charges$n[1], ")"),
              paste0("Smoker (n = ", mean_charges$n[2], ")"))),
    y = charges, fill = factor(smoker, labels = c("Non-smoker", "Smoker")))) +
  geom_boxplot(alpha = 0.6) +
  stat_summary(fun = mean, geom = "point", shape = 4, size = 2, color = "red") +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5), legend.position = "none") +
  labs(title = "Boxplot of Charges by Smoker", x = "Smoker", y = "Charges") +
  scale_fill_manual(values = c("Non-smoker" = "lightblue", "Smoker" = "orange"),
                    name = "Smoker") +
  guides(fill = guide_legend(override.aes = list(fill = c("lightblue", "orange")))) +
  annotate("text", x = c(1,2), y = mean_charges$mean_charges,
           label = paste("Mean:", round(mean_charges$mean_charges)),
           color = "red",
           vjust = ifelse(seq_along(mean_charges$mean_charges) == 2, 1.8, -1),
           size = 3)

# Display the boxplot
boxplot_smoker
```

Looking at the no-smoker and smoker boxplot, there is a large difference in the frequency of the two groups. Non-smokers are greatly overrepresented (n = 1063), whereas smokers (n = 274) are greatly underrepresented. Furthermore, a very large difference is also observed with regard to insurance expenditures. Looking at the median, non-smokers are significantly underrepresented in the charges. On average, they spend $8,441, whereas smokers spend an average of $32050, which is almost four times as much. Due to the large differences between the groups, it is likely that this variable has a large impact on the predictive accuracy of the following models.

2.5.4 Region

```r
# Create a new data frame with region and charges columns
data_region <- data %>%
  select(regionnortheast, regionnorthwest, regionsoutheast, regionsouthwest, charges) %>%
  pivot_longer(cols = starts_with("region"), names_to = "region",
               values_to = "region_bool") %>%
  filter(region_bool == 1)

# Calculate the sample sizes for each region
sample_sizes <- data_region %>%
  group_by(region) %>%
  summarize(n = n())

# Calculate the means for each region
means_region <- data_region %>%
  group_by(region) %>%
```

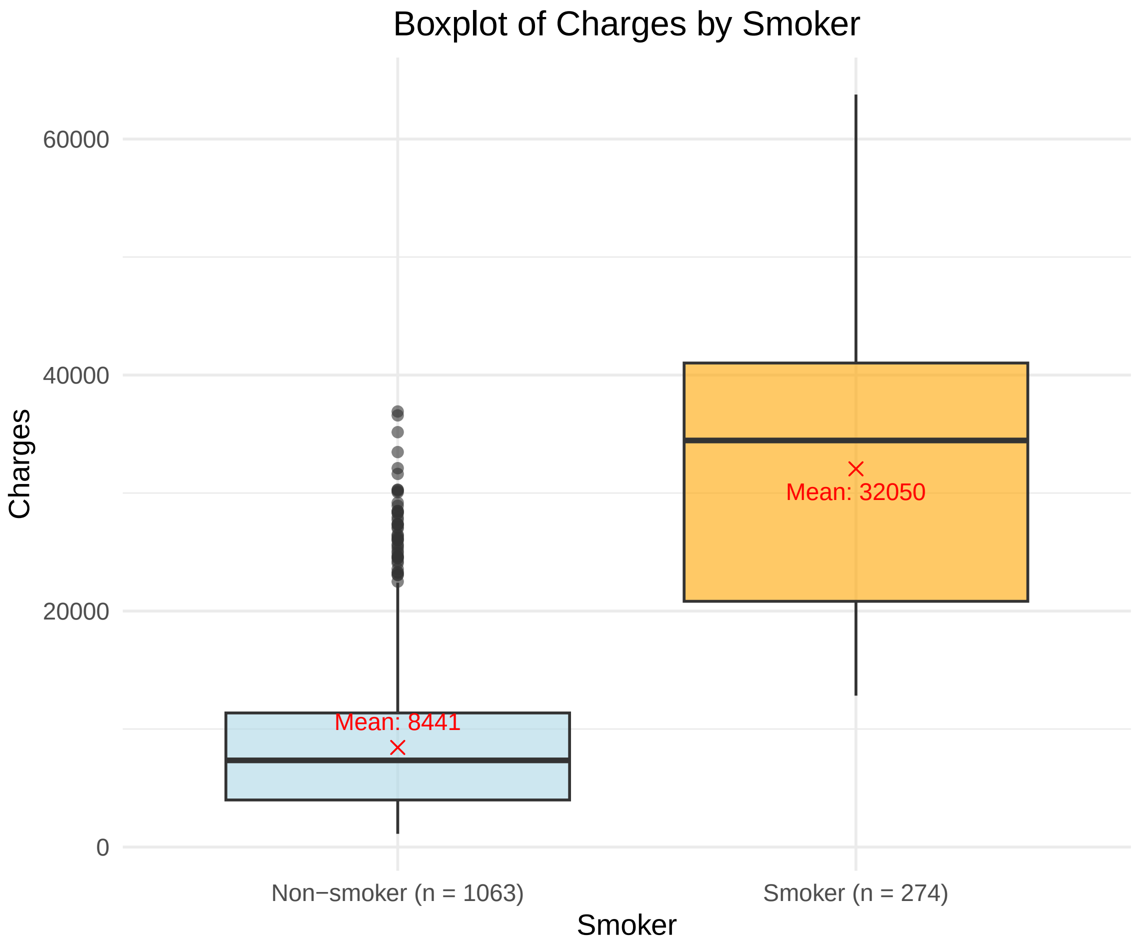


Figure 3: Fig: 2.3:Boxplot Charges-Smoker

```
  summarize(mean_charges = mean(charges))

# Create the boxplots for each region with sample sizes
boxplot_region <- ggplot(data_region, aes(x = region, y = charges, fill = region)) +
  geom_boxplot(alpha = 0.6) +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5), legend.position = "none",
        axis.text.x = element_text(angle = 0, vjust = 0.5, hjust=0.5),
        axis.title.y = element_text(vjust = 1),
        panel.grid.major.x = element_blank()) +
  labs(title = "Boxplot of Charges by Region", x = "Region", y = "Charges") +
  scale_fill_manual(values = c("lightblue", "orange", "lightgreen", "pink"),
                    name = "Region") +
  scale_x_discrete(labels = c(paste0("Northeast\n(n = ", sample_sizes$n[1], ")"),
                              paste0("Northwest\n(n = ", sample_sizes$n[2], ")"),
                              paste0("Southeast\n(n = ", sample_sizes$n[3], ")"),
                              paste0("Southwest\n(n = ", sample_sizes$n[4], ")"))) +
  geom_text(data = means_region,
            aes(x = region, y = mean_charges,
                label = paste0("Mean: ", round(mean_charges))),
            color = "red", vjust = 1.5, size = 3) +
  stat_summary(data = data_region,
               fun = mean, geom = "point", shape = 4, size = 2, color = "red")

# Display the boxplots
boxplot_region
```

Next, the boxplot of charges by region is examined. The four different regions, northeast (n = 324), northwest (n = 324), southeast (n = 364) and southwest (n = 325) do not differ much in their frequencies. Both in the medians and in means of the charges per region, there are no appreciable dissimilarities. Since the dummies do not show any significant differences with respect to the target variable, I expect that they do not have a major influence on the predictive accuracy of the model.

2.5.5 Children

```
# Children sequence
children <- seq(0,5,by = 1)

# Calculate the sample sizes for each group
n_children <- data %>%
  group_by(children) %>%
  summarize(n = n())

# Create a boxplot of the 'charge' variable separated by smoker
boxplot_children <- ggplot(data, aes(x = factor(children), y = charges)) +
  geom_boxplot(alpha = 0.6, fill = "skyblue") +
  stat_summary(fun = mean, geom = "point", shape = 4, size = 2, color = "red") +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5), legend.position = "none") +
  labs(title = "Boxplot of Charges by Number of Children",
       x = "Number of Children", y = "Charges") +
  scale_x_discrete(labels = paste(children, " Children", sep = "")) +
  scale_x_discrete(labels = c(paste0("0 Children\n(n = ", n_children$n[1], ")"),
```

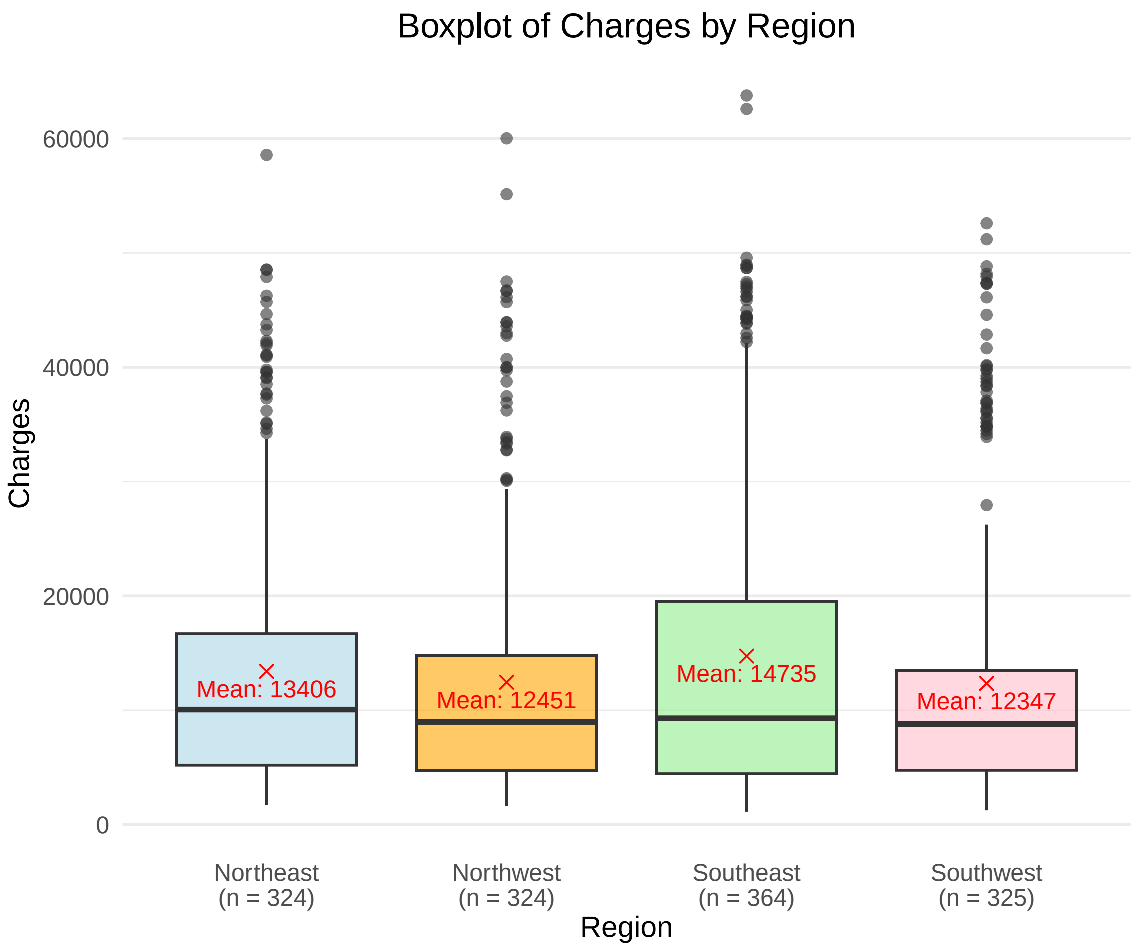


Figure 4: Fig: 2.4: Boxplot Charges-Regions

```
                               paste0("1 Child\n(n = ", n_children$n[2], ")"),
                               paste0("2 Children\n(n = ", n_children$n[3], ")"),
                               paste0("3 Children\n(n = ", n_children$n[4], ")"),
                               paste0("4 Children\n(n = ", n_children$n[5], ")"),
                               paste0("5 Children\n(n = ", n_children$n[6], ")")))
```

```
## Scale for x is already present.
## Adding another scale for x, which will replace the existing scale.
```

```
# Display the boxplot
boxplot_children
```

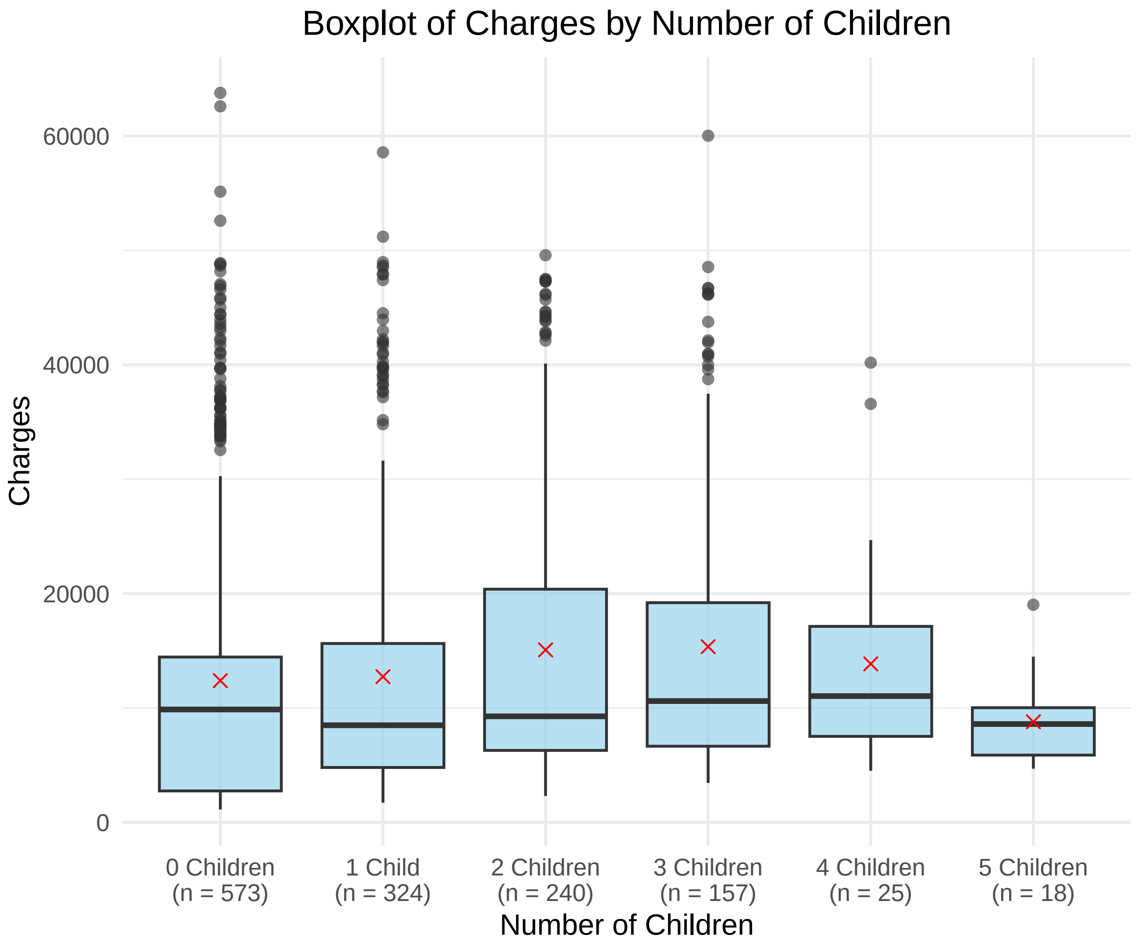


Figure 5: Fig: 2.5: Boxplot Charges-Children

Considering the boxplot between charges and the number of children, it is first apparent that the higher the number of children, the smaller the sample size of the group. The group with no children is the most represented (n = 573) and the group with the most children, 5, is the least represented (n = 18). Second, an increasing trend is shown in charges as the number of children increases. However, this trend flattens out for groups with four or more children. However, this may also be because the groups with 4 and 5 children have very few observations.

2.5.6 BMI & Age

```r
# Compute means of age and BMI variables
age_mean <- mean(data$age)
bmi_mean <- mean(data$bmi)

# Create histogram plot of age variable with mean annotation
age <- ggplot(data, aes(x = age)) +
  geom_histogram(binwidth = 3, fill = "lightblue", color = "black") +
  geom_vline(xintercept = age_mean, linetype = "dashed", color = "red") +
  labs(x = "Age", y = "Frequency") +
  ggtitle("Distribution of Age") +
  annotate("text", x = 45, y = 200, label = paste0("Mean: ", round(age_mean, 1)),
           color = "red") +
  theme(plot.title = element_text(hjust = 0.5))

# Create histogram plot of BMI variable with mean annotation
bmi <- ggplot(data, aes(x = bmi)) +
  geom_histogram(binwidth = 1, fill = "lightgreen", color = "black") +
  geom_vline(xintercept = bmi_mean, linetype = "dashed", color = "red") +
  labs(x = "BMI", y = "Frequency") +
  ggtitle("Distribution of BMI") +
  annotate("text", x = 40, y = 100, label = paste0("Mean: ", round(bmi_mean, 1)),
           color = "red") +
  theme(plot.title = element_text(hjust = 0.5))

# Combine plots
grid.arrange(age, bmi, ncol = 2)
```

Additionally, the two variables age and BMI are analyzed. A distribution map was created for each variable. In terms of age, there is a clear increase in the number of young respondents under the age of 20. These are the most common group. The rest of the respondents are approximately equally distributed in terms of age, with fewer respondents from the age of about 50 years. The average age, as already mentioned, is 39.2 years. The distribution of the BMI is normally distributed with a mean of 30.7. However, a BMI above 30 is considered obese and thus, the average respondent is overweight.

2.5.7 Scatterplots BMI, Age & Children

```r
# Change to factor
data$smoker <- factor(data$smoker, levels = c(0, 1),
                      labels = c("No Smoker", "Smoker"))

# Loop over features to create image
for (feat in c('age', 'bmi', 'children')) {
  plot <- ggplot(data = data, aes_string(x = feat,
                                         y = 'charges',
                                         group = 'smoker',
                                         fill = 'smoker',
                                         col = 'smoker')) +
    geom_jitter() +
    geom_smooth(method = 'lm') +
    ggtitle(glue::glue("Charges vs {feat}"))  +
    theme(plot.title = element_text(hjust = 0.5)) +
    labs(fill = "Smoker", color = "Smoker") +
```

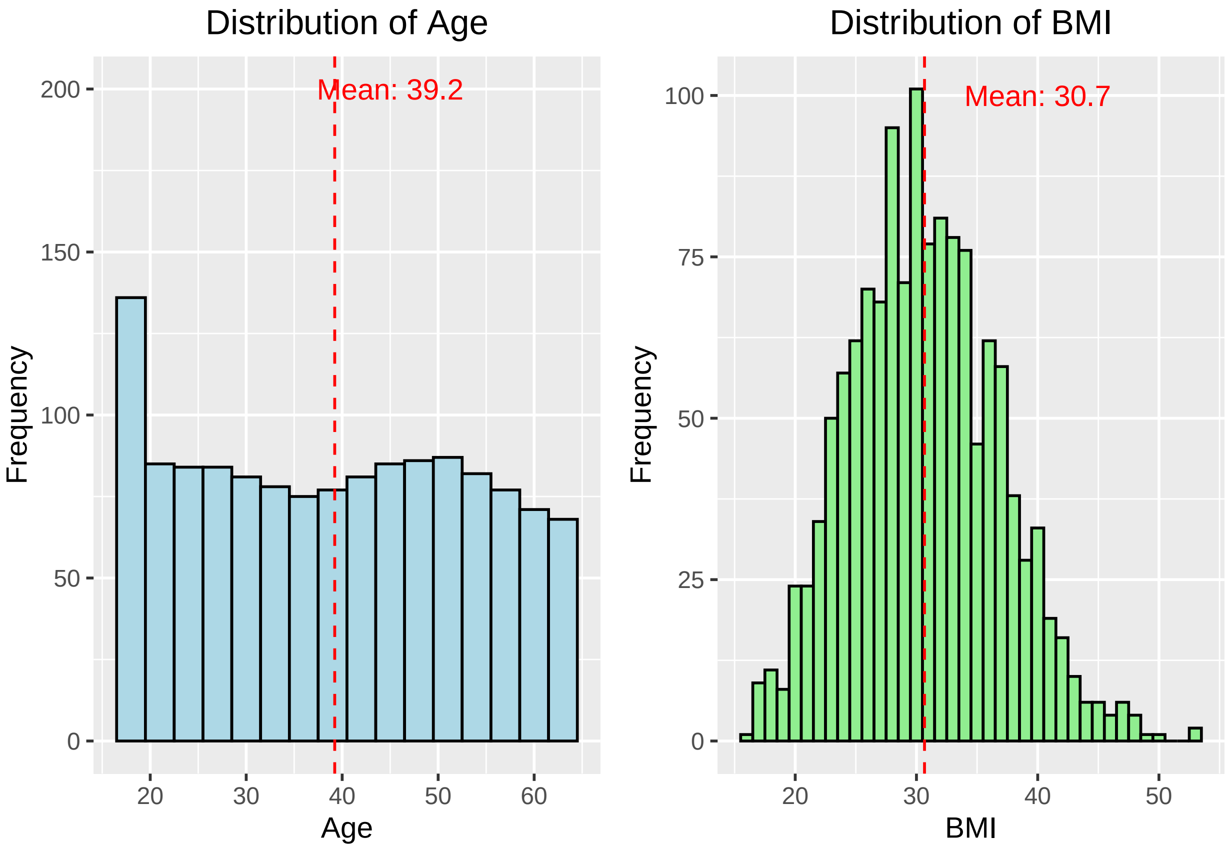


Figure 6: Fig: 2.6: Distribution Sex-BMI

```
    scale_fill_manual(values = c("#00BFC4", "#F8766D"),
                      labels = c("No Smoker", "Smoker")) +
    scale_color_manual(values = c("#00BFC4", "#F8766D"),
                       labels = c("No Smoker", "Smoker"))
  print(plot)
}
```

```
## Warning: `aes_string()` was deprecated in ggplot2 3.0.0.
## i Please use tidy evaluation ideoms with `aes()`
```

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

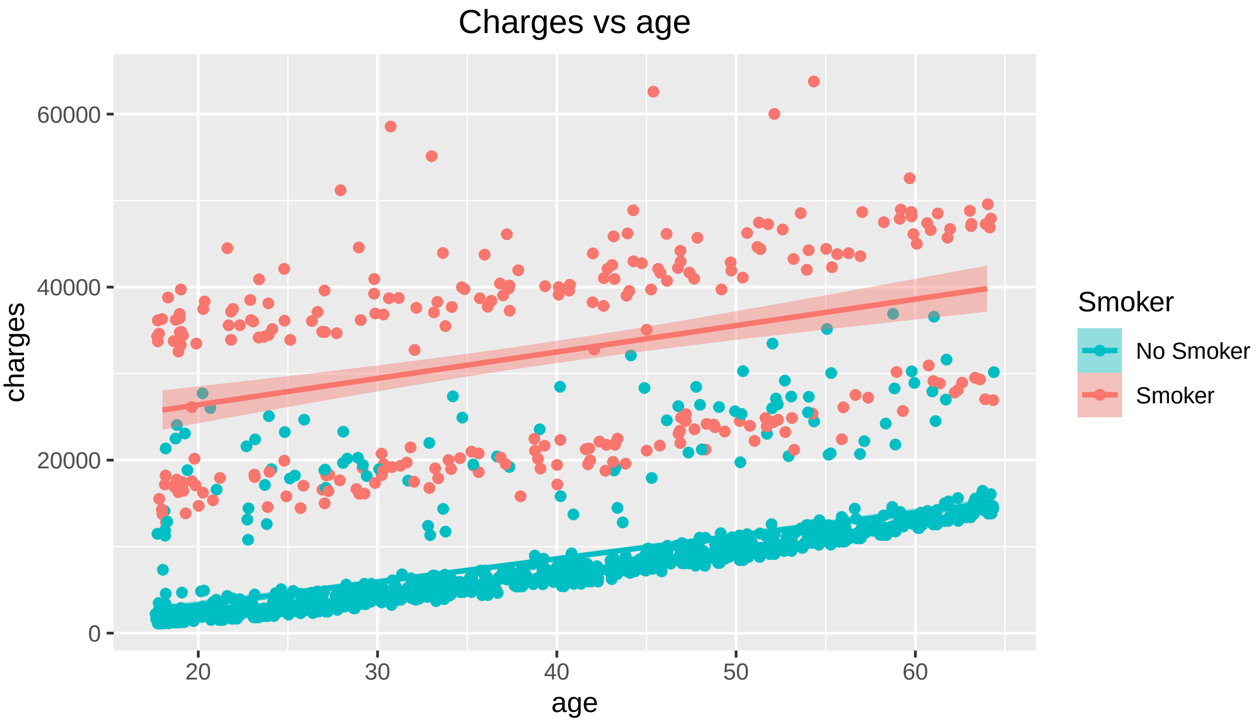


Figure 7: Fig: 2.7: Scatterplots Charges- Children, BMI, Smoker

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

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

```
# Rechange to numeric
data$smoker <- as.numeric(data$smoker)
```

To see the correlation between age, BMI and the number of children with the target variable charges, three scatter plots are created for both smokers and non-smokers, and a linear regression line is added to each plot to show the trend. This analysis can provide insight into the factors that influence medical costs for smokers and non-smokers based on their age, BMI and the number of children (Tukey, 1977).

The plots have a lot of valuable information for the analysis. Initially, it can be seen that in all plots smokers pay a significantly higher charge, since the smokers' line is above the non-smokers' line in all plots. In addition, the following key points can be concluded.

First, the charges increase slightly with age, regardless of whether smokers or non-smokers. Second, it can be seen that the charges also increase the higher the BMI is. Third, people who have more children are less likely to smoke cigarettes.

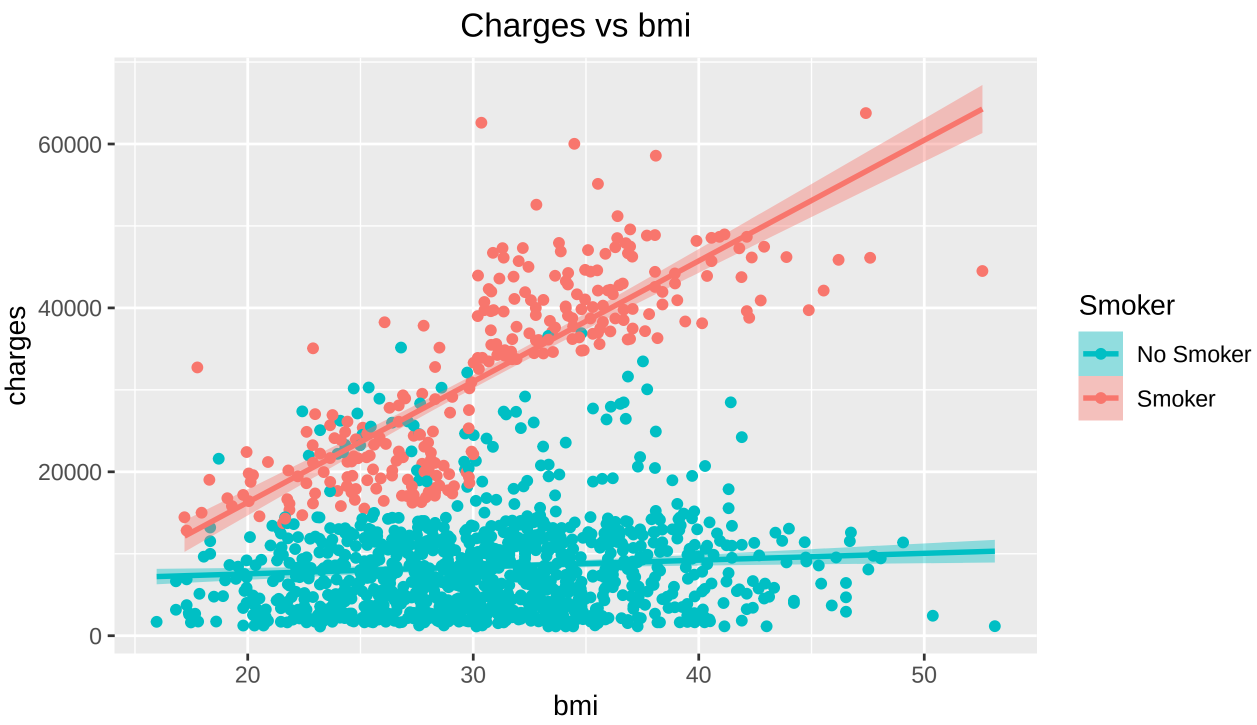


Figure 8: Fig: 2.7: Scatterplots Charges- Children, BMI, Smoker

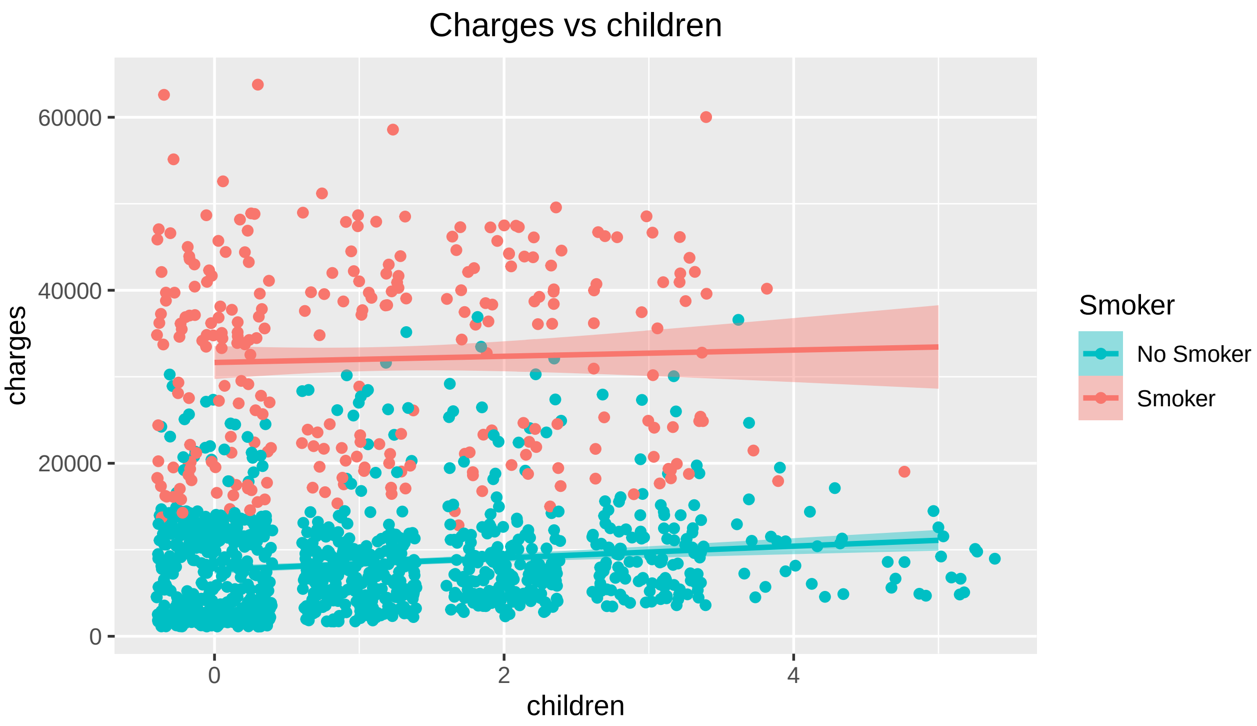


Figure 9: Fig: 2.7: Scatterplots Charges- Children, BMI, Smoker

2.5.8 Correlation Matrix

```
# Compute correlation matrix
cor_matrix <- cor(data)

# Plot heatmap correlation matrix
ggcorrplot(cor_matrix, type = "lower", colors = c("#6D9EC1", "white", "#E46726"),
           lab_size = 3, lab = TRUE,
           method = "circle",
           title = "Correlation Matrix")
```

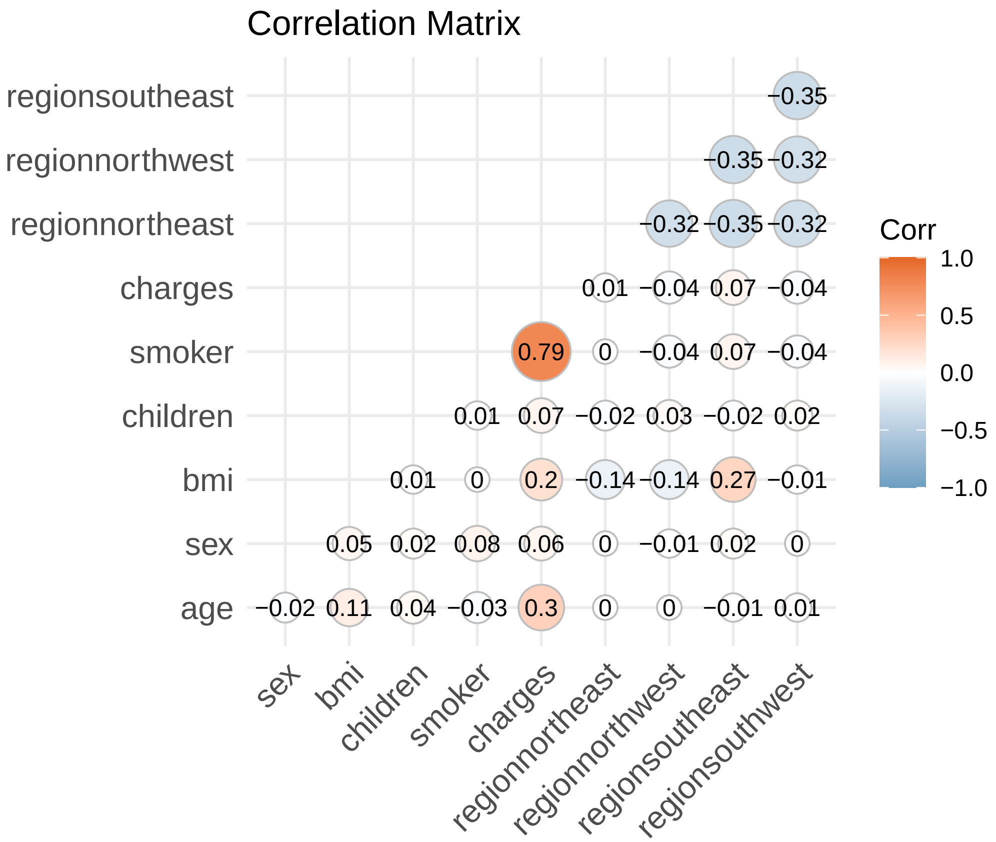


Figure 10: Fig: 2.8: Correlation Matrix

Lastly, the given correlation matrix shows the pairwise correlation coefficients between the variables in the medical cost data set. The correlation coefficients range from -1 to 1 and measure the linear relationship between two variables. A correlation coefficient of 1 indicates a perfect positive linear relationship, a coefficient of -1 indicates a perfect negative linear relationship, and a coefficient of 0 indicates no linear relationship (Tukey, 1977).

The matrix shows a positive correlation between age and charges (0.30), indicating that older individuals tend to have higher medical charges. Similarly, there is a positive correlation between BMI and charges (0.20), which means that people with a higher BMI also tend to have higher medical charges. Furthermore, there is a very small correlation between the number of children and medical charges, with a correlation coefficient of 0.07. We can see that the correlation between smoker and charges is 0.79, indicating a very strong positive correlation between smoking habits and medical charges. Also, the variable smoker has the highest impact on the target variable charges.

3. Regression Analysis

Following, regression analysis is performed to accurately predict the target variable charges. Since charges is a metric variable, a regression analysis is the right method to predict the medical costs of an individual based on the variables: age, sex, BMI, children, smoker and region. Regression analysis is a statistical method used to examine the relationship between a dependent variable and one or more independent variables. It is commonly used to predict the value of the dependent variable based on the values of the independent variables (Zhang, et al., 2015).

In this analysis, R-squared and RMSE are used to assess the accuracy of the model. RMSE (root mean square error) and R-squared are commonly used metrics for evaluating the accuracy of regression models. RMSE measures the average distance between the predicted values and the actual values, and it penalizes larger errors more heavily than smaller ones. An advantage of the RMSE is that it is in the same units as the target variable. Overall, it provides a good measure of the model's accuracy in predicting the target variable. On the other hand, R-squared measures the proportion of the variance in the target variable that is explained by the predictor variables in the model. It gives an indication of how well the model fits the data and how much of the variability in the target variable is accounted for by the model (James et al., 2013).

When the research question is to maximize accuracy, both RMSE and R-squared are important metrics to consider. RMSE measures the absolute accuracy of the model, while R-squared measures the relative accuracy in relation to the total variability in the data. A high R-squared value indicates that the model explains a large proportion of the variability in the target variable and is a good fit for the data. A low RMSE value indicates that the model has a small average distance between the predicted values and the actual values, which also implies good accuracy (James et al., 2013).

Overall, using both RMSE and R-squared as evaluation metrics can provide a comprehensive assessment of the model's accuracy and its ability to explain the variability in the target variable (James et al., 2013). Additionally, to detect overfitting, 10-fold cross validation is performed and the best model is selected based on the lowest RMSE. Also, a seed is used to ensure reproducibility of the results.

3.1 Regression Preprocessing

It is important to consider several steps before fitting a regression model. One of these steps is dummy encoding, which can result in perfect multicollinearity if all binary variables are included in the model (James et al., 2013).

To avoid this issue, the reference category "regionsouthwest" is excluded from the model. Additionally, scaling the data is performed to bring all variables to the same scale and avoid biased estimates due to differences in the variable magnitudes. Scaling the data also helps to improve model convergence and reduce the computational complexity of the regression algorithm (James et al., 2013).

```
# Remove regionsouthwest
data <- subset(data, select = -regionsouthwest)

# Convert data to a data frame
data <- as.data.frame(data)

# Scale all predictor variables
scaled_data <- data %>%
  select(-charges) %>%
  mutate_all(scale) %>%
  cbind(charges = data$charges)
```

3.2 Linear Regression

Linear regression is a statistical method used to model the linear relationship between a dependent variable and one or more independent variables. In this section, we will perform linear regression on the medical costs data set using the lm() function in R. The code below fits a linear regression model to our data, where the dependent variable is charges and the independent variables are age, sex, BMI, children, smoker and region (full model). We do this to obtain a benchmark for the model's predictive power before performing any feature selection or tuning (James et al., 2013).

The output of the summary function shows that the model has an R-squared value of 0.7507, indicating that 75.07% of the variability in the target variable is explained by the independent variables. Additionally, the F-statistic has a very low p-value (< 2.2e-16), suggesting that the model is significant.

The custom function "perform_cv" was created to evaluate the performance and generalization ability of a regression model using cross-validation with k folds, either with full model or stepwise regression. It computes and prints the RMSE and R-squared for each fold, and returns mean RMSE and R-squared values in a data frame, as well as RMSE values for each fold.

The perform_cv function applies k-fold cross-validation on a linear regression model and has different modifications for further analysis in this study, resulting in RMSE and R-squared values for each fold. The mean results show a mean train RMSE of 6041 mean test RMSE of 6059, and a mean R-squared of 0.7508, indicating that the model has a reasonable fit and is not overfitting, which can also be seen the in the plot.

```
# Create first linear regression model
set.seed(50)
base_model <- lm(charges ~ ., data = scaled_data)

# Summary
summary(base_model)
```

```
##
## Call:
## lm(formula = charges ~ ., data = scaled_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11305.1  -2850.3   -979.9   1395.0  29992.8
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)      13279.12     165.85  80.067  < 2e-16 ***
## age               3606.09     167.30  21.555  < 2e-16 ***
## sex                -64.76     166.65  -0.389 0.697630
## bmi               2069.59     174.54  11.857  < 2e-16 ***
## children           572.43     166.24   3.443 0.000593 ***
## smoker            9629.70     166.91  57.693  < 2e-16 ***
## regionnortheast    411.54     204.94   2.008 0.044836 *
## regionnorthwest    261.85     204.79   1.279 0.201266
## regionsoutheast    -33.48     209.64  -0.160 0.873149
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6064 on 1328 degrees of freedom
## Multiple R-squared:  0.7507, Adjusted R-squared:  0.7492
## F-statistic:   500 on 8 and 1328 DF,  p-value: < 2.2e-16
```

```
# 10 fold cross validation function
perform_cv <- function(data, stepwise = FALSE, k = 10) {

  # Create folds
  folds <- createFolds(data$charges, k = k)
```

```r
  # Initialize vectors to store results
  rmse_train_vec <- numeric(k)
  rmse_test_vec <- numeric(k)
  r_squared_vec <- numeric(k)

  # Loop over folds
  for (i in 1:k) {

    # Get training and testing data for this fold
    train_data <- data[-folds[[i]], ]
    test_data <- data[folds[[i]], ]

    # Fit stepwise regression model
    if (stepwise) {
      model_fit <- step(lm(charges ~ ., data = train_data),
                        direction = "backward", trace = 0)
    } else {
      # Fit normal regression model
      model_fit <- lm(charges ~ ., data = train_data)
    }

    # Make predictions on train and test data
    train_preds <- predict(model_fit, newdata = train_data)
    test_preds <- predict(model_fit, newdata = test_data)

    # Calculate RMSE for train and test data
    rmse_train_vec[i] <- sqrt(mean((train_data$charges - train_preds)^2))
    rmse_test_vec[i] <- sqrt(mean((test_data$charges - test_preds)^2))
    r_squared_vec[i] <- summary(model_fit)$r.squared

    # Print the results for this fold
    cat("Fold", i, "Train RMSE:", rmse_train_vec[i],
        "Test RMSE:", rmse_test_vec[i],
        "R-squared:", r_squared_vec[i], "\n")
  }

  # Compute mean train and test RMSE
  mean_rmse_train <- mean(rmse_train_vec)
  mean_rmse_test <- mean(rmse_test_vec)
  mean_r_squared <- mean(r_squared_vec)

  # Combine mean results into a data frame
  mean_results <- data.frame(mean_rmse_train = mean_rmse_train,
                             mean_rmse_test = mean_rmse_test,
                             mean_r_squared = mean_r_squared)

  # Combine RMSE for each fold into a data frame
  rmse_results <- data.frame(train_rmse = rmse_train_vec,
                             test_rmse = rmse_test_vec)

  # Return a list containing both mean and RMSE results
  return(list(mean_results = mean_results, rmse_results = rmse_results))
}
```

```
# Apply cv function for basic linear regression
set.seed(50)
cv_results_base <- perform_cv(scaled_data, k = 10)
```

```
## Fold 1 Train RMSE: 6127.961 Test RMSE: 5239.622 R-squared: 0.744402
## Fold 2 Train RMSE: 5996.019 Test RMSE: 6494.268 R-squared: 0.7607002
## Fold 3 Train RMSE: 6103.052 Test RMSE: 5515.859 R-squared: 0.7521324
## Fold 4 Train RMSE: 6023.444 Test RMSE: 6249.84 R-squared: 0.744503
## Fold 5 Train RMSE: 6009.766 Test RMSE: 6360.568 R-squared: 0.7514773
## Fold 6 Train RMSE: 6034.422 Test RMSE: 6137.851 R-squared: 0.7527761
## Fold 7 Train RMSE: 6110.261 Test RMSE: 5434.032 R-squared: 0.7456813
## Fold 8 Train RMSE: 5967.488 Test RMSE: 6710.775 R-squared: 0.7546683
## Fold 9 Train RMSE: 6031.718 Test RMSE: 6163.635 R-squared: 0.7528136
## Fold 10 Train RMSE: 6009.813 Test RMSE: 6373.934 R-squared: 0.7496982
```

```
cv_mean_results_base <- cv_results_base$mean_results
cv_mean_results_base
```

```
##   mean_rmse_train mean_rmse_test mean_r_squared
## 1        6041.394       6068.038      0.7508852
```

```
# Function plot the mean train and test RMSE for each fold
plot_cv_results <- function(cv_results) {
  # Extract the train and test RMSE results
  df <- cv_results$rmse_results
  df$fold <- 1:nrow(df)

  # Plot the results
  ggplot(data = df, aes(x = fold)) +
    geom_line(aes(y = train_rmse, color = "Train"), linewidth = 1) +
    geom_point(aes(y = train_rmse, color = "Train")) +
    geom_line(aes(y = test_rmse, color = "Test"), linewidth = 1) +
    geom_point(aes(y = test_rmse, color = "Test")) +
    labs(x = "Fold", y = "RMSE", title = "Train and Test RMSE") +
    scale_x_discrete(labels = 1:10) +
    scale_color_manual(name = "Data",
                       values = c("Train" = "skyblue", "Test" = "orange")) +
    theme_minimal() +
    theme(plot.title = element_text(hjust = 0.5))
}
```

```
# Apply plot function
plot_cv_results(cv_results_base)
```

In order to assess the validity of our linear regression model and identify potential influential observations, we perform a leverage and Cook's distance analysis. We calculate the leverage values of the observations in our model and plot these leverage values to identify any observations with high leverage. Here, no outliers can be identified in the high leverage region.

To assess heteroscedasticity of the coefficients, we calculated the variance inflation factor (VIF) for each predictor variable in the base model. The VIF measures the degree of multicollinearity between each predictor and the other predictor variables in the model. The VIF values for the predictor variables in our base model were all below 2.5, indicating that multicollinearity was not a major concern (O'Brien, 2007).

Next, residual analysis is an important step in regression analysis as it helps to assess the validity of the model and identify potential outliers or influential observations that may require further investigation or

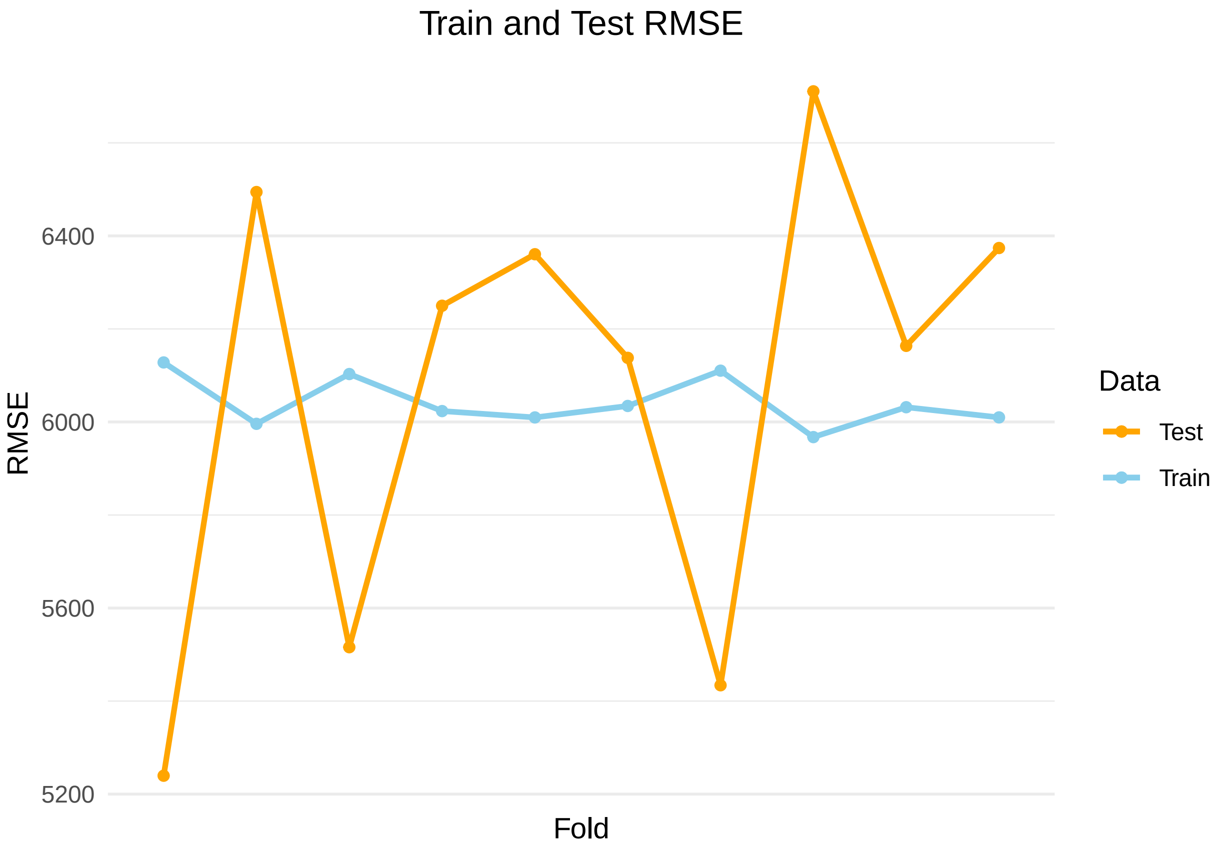


Figure 11: Fig: 3.1: Cross-Validation Base-Model

exclusion from the model. By plotting the residual diagram, four different plots evolve.

The top left panel shows a plot of the residuals versus the fitted values. If the residuals are randomly distributed around zero, it suggests that the linear regression model is appropriate and captures the underlying relationship between the dependent and independent variables. However, in this case we see that our residuals are not fully linear and that the linear regression misses a group of residuals in prediction (James et al., 2013).

The top right panel shows a QQ plot of the residuals. If the residuals are normally distributed, the points on the plot will follow a straight line. For the linear regression model we can detect that the most points are not laying on the line which indicates that our residuals are not normally distributed. Therefore, we check by performing a Shapiro-Wilk normality test which is a statistical test used to determine whether a set of data follows a normal distribution. In this case, the test was performed on the residuals of the linear regression model. The test result shows that the p-value is less than the significance level of 0.05, indicating that the null hypothesis of normality is rejected, and the residuals are not normally distributed (James et al., 2013).

The scale-location plot in the bottom left pannel shows the standardized residuals plotted against the square root of the standardized predicted values. This plot is used to assess if the variance of the errors is constant across the range of the predicted values. Our plot shows two scatter clouds, one larger on the left and a smaller one on the right, which indicates that our residuals are heteroscedastic. This idea holds to be true after performing a Breusch-Pagan test. The Breusch-Pagan test is a test for heteroscedasticity of the residuals, where a low p-value indicates evidence of heteroscedasticity in the model. In this case, the studentized Breusch-Pagan test of the base model resulted in a p-value of $<$ 2.2e-16, indicating strong evidence of heteroscedasticity in the model (James et al., 2013).

The bottom right panel shows a plot of the residuals versus the leverage values. Leverage values measure how far an observation is from the center of the independent variable distribution. Observations with high leverage values can have a disproportionate effect on the regression results. However, this plot supports our prior analysis about Leverage and Cook's distance since it does not show any significant outliers (James et al., 2013).

In conclusion, the assumptions for linear regression include linearity, independence, homoscedasticity, normality of errors, and absence of multicollinearity. In our analysis, the requirements for normality of residuals and absence of non-linearity and heteroscedasticity were not met, indicating that the assumptions of linear regression may not be appropriate for our data. Further analysis is needed to explore potential transformations or alternative models to improve predictive accuracy.

```r
# Leverage
plot(hatvalues(base_model), pch=19, main="Leverage", cex=0.5)
n <- nrow(scaled_data)
p <- ncol(scaled_data)
abline(h=(1:3)*(p+1)/n, col=c("black", "darkred", "red"))
```

```r
# Residual analysis
par(mfrow = c(2, 2))
plot(base_model, col = "darkblue")
```

```r
# Find rows with high leverage
high_leverage_rows <- which(hatvalues(base_model) > (p+1)*3/n)

# Print the row numbers
cat("Rows with high leverage:", paste(high_leverage_rows, collapse = ", "))
```

```
## Rows with high leverage:
```

```r
# Test on multicollinearity by variance inflation factor for each predictor
vif(base_model)
```

```
##          age        sex        bmi   children     smoker
##     1.016794   1.008944   1.106742   1.004017   1.012100
```

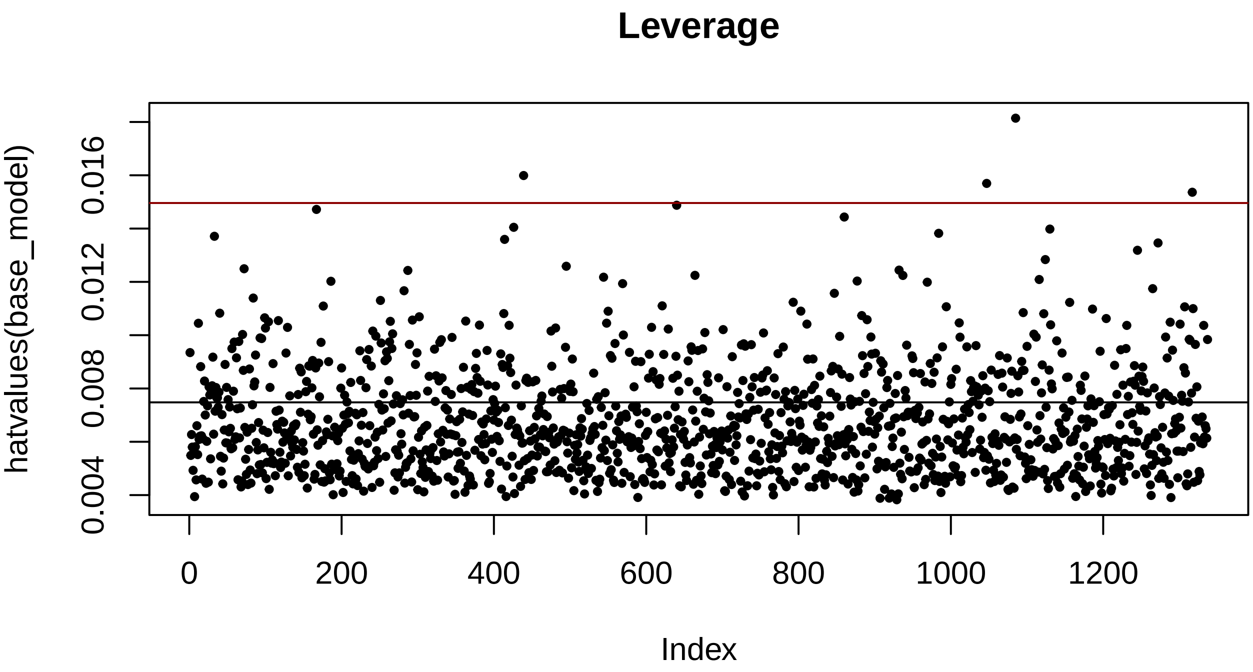


Figure 12: Fig: 3.2: Leverage Plot Base-Model

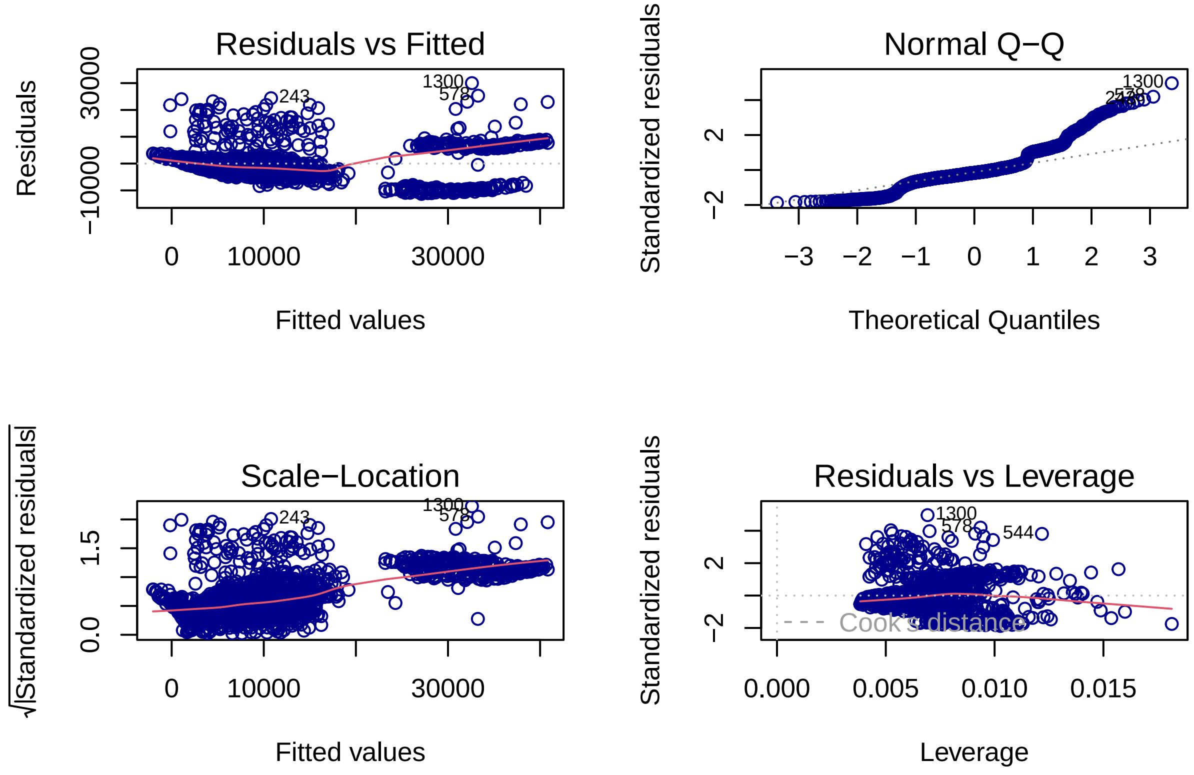


Figure 13: Fig: 3.3: Residual Plots Base-Model

```
## regionnortheast regionnorthwest regionsoutheast
##        1.525846        1.523624        1.596665
```

```
# Extract the residuals
resid <- residuals(base_model)

# Perform Shapiro-Wilk test for normality of residuals
shapiro.test(resid)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  resid
## W = 0.89909, p-value < 2.2e-16
```

```
# Breusch-Pagan Test to detect heteroscedasticity
bptest(base_model)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  base_model
## BP = 121.57, df = 8, p-value < 2.2e-16
```

To reduce the dimensionality in the full linear regression model and further reduce the RMSE and thus the accuracy, we use backward step regression to eliminate all non-significant predictors in the model.

The output shows the summary of the backward stepwise regression model, which is a modified version of the base model. The backward stepwise regression model excludes the variables sex and regionsoutheast, and has the same R-squared value of 0.7507 compared to the base model's R-squared. However, the p-value of the F-statistic for the backward stepwise regression model is much lower, indicating that the model is a better fit for the data. Additionally, the backward stepwise regression model has a slightly lower residual standard error, suggesting that it better predicts the target variable charges (James et al., 2013).

The results of the new cross-validation using the backward step-wise selected model show similar performance to the previous model, with mean train RMSE of 6044 and mean test RMSE of 6076, indicating that the new model did not significantly improve the performance compared to the previous one.

```
# Perform backward step-wise selection of the base model using AIC
model_step <- step(base_model, direction = "backward", trace = 0)

# Summary
summary(model_step)
```

```
##
## Call:
## lm(formula = charges ~ age + bmi + children + smoker + regionnortheast +
##     regionnorthwest, data = scaled_data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -11332  -2827  -1003   1382  29903
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      13279.1      165.7  80.122  < 2e-16 ***
## age               3608.4      167.1  21.599  < 2e-16 ***
## bmi               2061.7      171.7  12.008  < 2e-16 ***
## children           572.1      166.0   3.445 0.000588 ***
## smoker            9623.1      166.0  57.979  < 2e-16 ***
## regionnortheast    427.3      178.4   2.395 0.016769 *
## regionnorthwest    278.0      178.5   1.557 0.119698
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6060 on 1330 degrees of freedom
## Multiple R-squared:  0.7507, Adjusted R-squared:  0.7496
## F-statistic: 667.5 on 6 and 1330 DF,  p-value: < 2.2e-16
```

```
# Apply cv function for step-wise regression
set.seed(50)
cv_results_step <- perform_cv(scaled_data, stepwise = TRUE, 10)
```

```
## Fold 1 Train RMSE: 6128.477 Test RMSE: 5234.489 R-squared: 0.7443589
## Fold 2 Train RMSE: 5997.394 Test RMSE: 6480.925 R-squared: 0.7605904
## Fold 3 Train RMSE: 6104.007 Test RMSE: 5507.277 R-squared: 0.7520549
## Fold 4 Train RMSE: 6026.462 Test RMSE: 6214.247 R-squared: 0.7442469
## Fold 5 Train RMSE: 6013.962 Test RMSE: 6377.27 R-squared: 0.7511301
## Fold 6 Train RMSE: 6038.441 Test RMSE: 6156.543 R-squared: 0.7524466
## Fold 7 Train RMSE: 6115.626 Test RMSE: 5440.496 R-squared: 0.7452344
## Fold 8 Train RMSE: 5971.382 Test RMSE: 6729.788 R-squared: 0.7543481
## Fold 9 Train RMSE: 6031.905 Test RMSE: 6162.214 R-squared: 0.7527982
## Fold 10 Train RMSE: 6009.831 Test RMSE: 6371.96 R-squared: 0.7496967
```

```
cv_mean_results_step <- cv_results_step$mean_results
cv_mean_results_step
```

```
##   mean_rmse_train mean_rmse_test mean_r_squared
## 1        6043.749       6067.521      0.7506905
```

```
# Apply cv plot function to detect overfitting
plot_cv_results(cv_results_step)
```

```
# Combine
comb <- rbind(cv_mean_results_base, cv_mean_results_step)
rownames(comb) <- c("LR-full-model", "LR-step-model")
comb
```

```
##               mean_rmse_train mean_rmse_test mean_r_squared
## LR-full-model        6041.394       6068.038      0.7508852
## LR-step-model        6043.749       6067.521      0.7506905
```

The residuals analysis shows that there are no significant differences between the full linear regression model and the stepwise linear regression model. We were able to reduce the high leverage points to only two instead of four in the full model. The model has heterescedastic residuals, not normally distributed residuals, and there is no multicolinearity of coefficients.

```
# Leverage
plot(hatvalues(model_step), pch=19, main="Leverage", cex=0.5)
n <- nrow(scaled_data)
p <- ncol(scaled_data)
abline(h=(1:3)*(p+1)/n, col=c("black", "darkred", "red"))
```

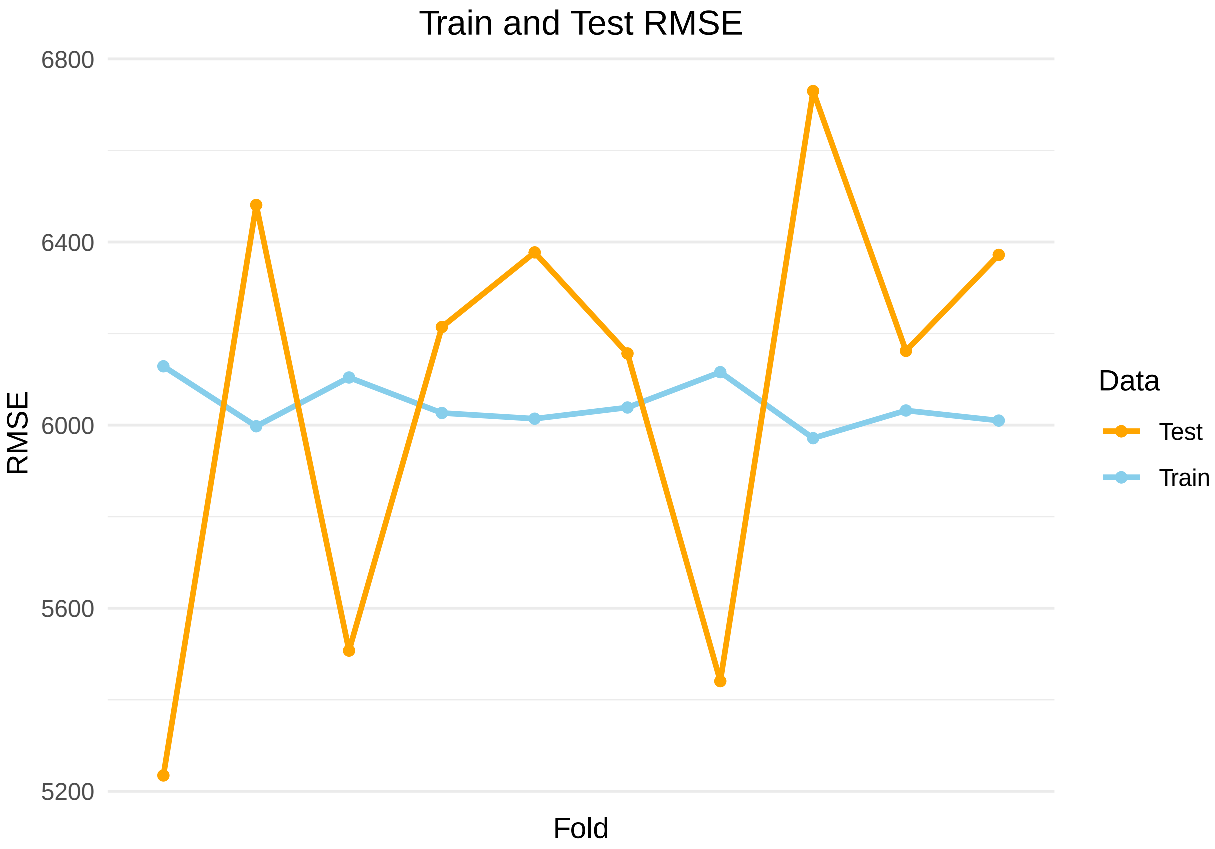
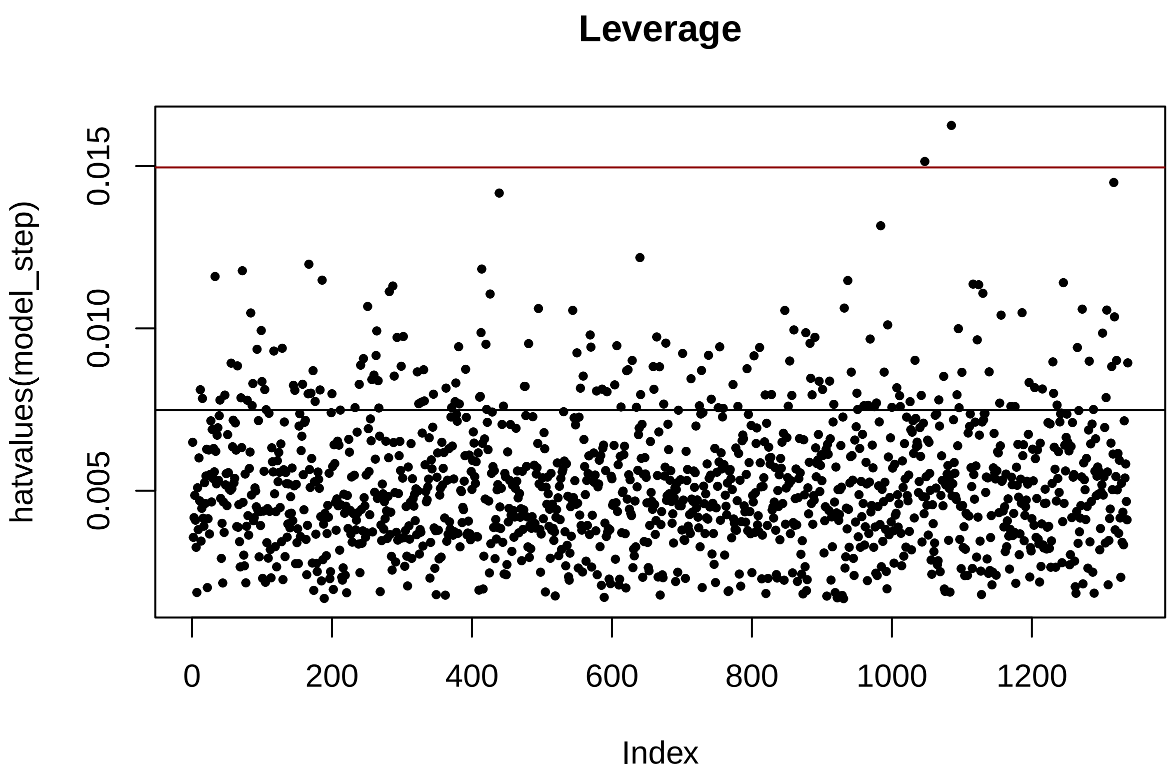


Figure 14: Fig: 3.4: Cross-Validation Step-Model



```
# Residual analysis
par(mfrow = c(2, 2))
plot(model_step, col = "darkblue")
```

```
# Find rows with high leverage
high_leverage_rows <- which(hatvalues(model_step) > (p+1)*3/n)

# Print the row numbers
cat("Number of high leverage points:", paste(length(high_leverage_rows), collapse = ", "))
```

```
## Number of high leverage points: 0
```

```
# Calculate Cook's distance for rows with high leverage
cooks_dist <- cooks.distance(model_step)[high_leverage_rows]

# Only showing Cook's distance values greater than 1
cooks_dist[cooks_dist > 1]
```

```
## named numeric(0)
```

```
# Test on multicollinearity by variance inflation factor for each predictor
vif(model_step)
```

```
##             age             bmi        children          smoker regionnortheast 
##        1.015304        1.072387        1.002824        1.002150        1.158346 
## regionnorthwest 
##        1.159435 
```

```
# Extract the residuals
resid <- residuals(model_step)

# Perform Shapiro-Wilk test for normality of residuals
```

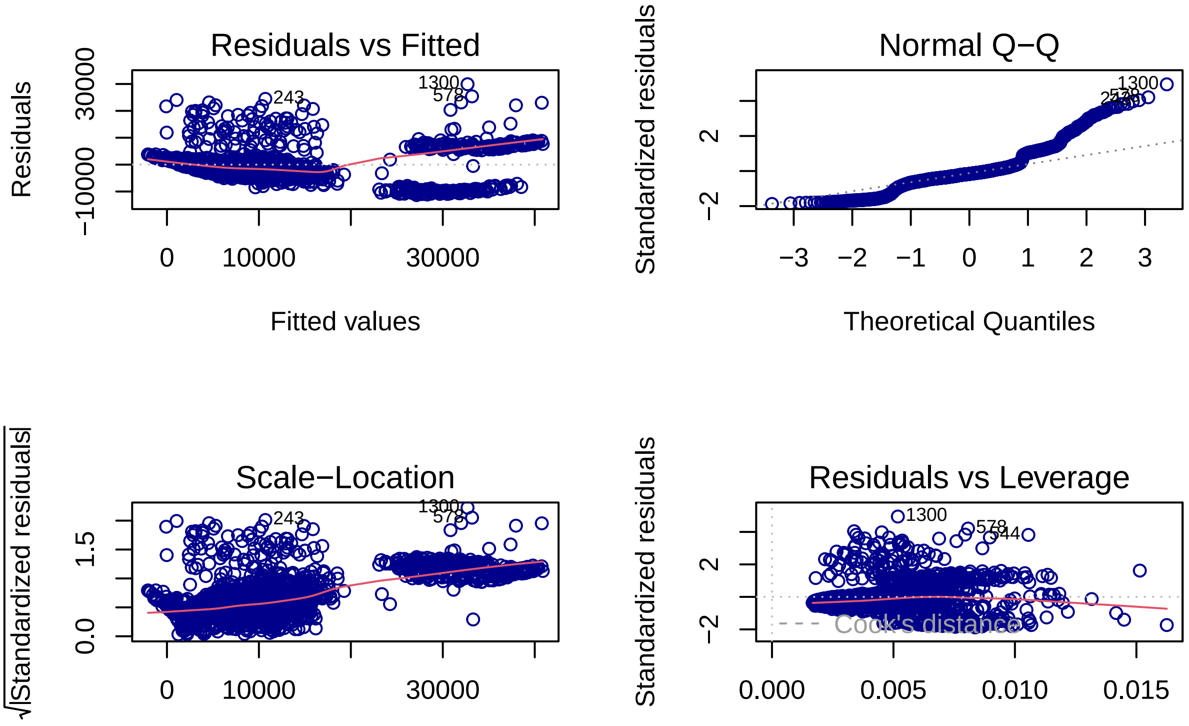


Figure 15: Fig: 3.5: Residual Plots Step-Model

```
shapiro.test(resid)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  resid
## W = 0.89926, p-value < 2.2e-16
```

```
# Breusch-Pagan Test to detect heteroscedasticity
bptest(model_step)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  model_step
## BP = 119.72, df = 6, p-value < 2.2e-16
```

3.3 Polynomial Regression

Based on the results of our linear regression analysis, it appears that we need a different approach to further fit the data and improve the accuracy of our model for predicting the cost of medical costs. One possibility is polynomial regression, in which a polynomial function is fitted to the data instead of a linear function.

In the context of linear regression, polynomial regression, though capable of modeling non-linear relationships, is still considered a linear approach due to the linearity of the coefficients. It serves as an extension of linear regression, accommodating non-linear patterns by raising predictor variables to higher-order powers. However, it is important to be cautious when using polynomial regression, as overfitting can easily occur if the degree of the polynomial is too high. Therefore, it is important to carefully select the degree of the polynomial and use cross-validation techniques to evaluate the model performance. In our case we test degree 2 and degree 3 to increase our predictive accuracy (James et al., 2013).

3.3.1 Polynomial Regression (2nd degree)

We start building the formula and data set for second degree polynomial regression which includes all predictor variables in the data set. The formula was constructed by pasting together the terms for the 2nd degree polynomial for each predictor variable using the poly function. The resulting formula was then used to create a new data frame, poly2_data, that contains the model matrix and the target variable.

Looking at the summary output of the polynomial regression (2nd degree) it has a significantly higher R-squared value of 0.8477 compared to the linear regression R-squared from the base model which had a value of 0.7507. Since polynomial regression is prone to overfit the data we need to see how the train and test RMSE behaves using cross validation. Additionally, there are plenty of ‘NA’ values. This is because these are binary variables and there is no quadratic relationship between those. To get ride of these many unnecessary predictors we immediately use the step function here to decrease the dimensionality and complexity of the model.

```r
# Split up data
X <- scaled_data[-9]
y <- scaled_data$charges

# Build formula for polynomial regression full model 2nd degree
formula <- as.formula(
  paste(
    ' ~ .^2 + ',
    paste('poly(', colnames(X), ', 2, raw=TRUE)[, 2]', collapse = ' + ')
  )
)

# Check formula
formula
```

```
## ~.^2 + poly(age, 2, raw = TRUE)[, 2] + poly(sex, 2, raw = TRUE)[,
##     2] + poly(bmi, 2, raw = TRUE)[, 2] + poly(children, 2, raw = TRUE)[,
##     2] + poly(smoker, 2, raw = TRUE)[, 2] + poly(regionnortheast,
##     2, raw = TRUE)[, 2] + poly(regionnorthwest, 2, raw = TRUE)[,
##     2] + poly(regionsoutheast, 2, raw = TRUE)[, 2]
```

```r
# Build data frame for polynomial regression
poly2_data <- as.data.frame(model.matrix(formula, data = X))
poly2_data$charges <- y

# Build model polynomial regression
set.seed(50)
model_poly2 <- lm(formula = charges ~., data = poly2_data)

# Summary
summary(model_poly2)
```

```
##
## Call:
## lm(formula = charges ~ ., data = poly2_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10634.8  -1874.8  -1226.8   -368.3  30757.1
##
## Coefficients: (9 not defined because of singularities)
##                                             Estimate Std. Error t value
## (Intercept)                                 12974.41     245.76  52.794
## `(Intercept)`                                     NA         NA      NA
## age                                          3605.30     139.46  25.852
## sex                                          -245.52     132.49  -1.853
## bmi                                          2015.33     142.99  14.094
## children                                      957.62     182.84   5.238
## smoker                                       9654.48     134.51  71.774
## regionnortheast                               575.51     164.80   3.492
## regionnorthwest                               272.46     166.53   1.636
## regionsoutheast                               130.86     171.47   0.763
## `poly(age, 2, raw = TRUE)[, 2]`               782.35     163.02   4.799
## `poly(sex, 2, raw = TRUE)[, 2]`                   NA         NA      NA
## `poly(bmi, 2, raw = TRUE)[, 2]`              -235.73     110.37  -2.136
## `poly(children, 2, raw = TRUE)[, 2]`         -141.64     120.23  -1.178
## `poly(smoker, 2, raw = TRUE)[, 2]`                NA         NA      NA
## `poly(regionnortheast, 2, raw = TRUE)[, 2]`       NA         NA      NA
## `poly(regionnorthwest, 2, raw = TRUE)[, 2]`       NA         NA      NA
## `poly(regionsoutheast, 2, raw = TRUE)[, 2]`       NA         NA      NA
## `age:sex`                                     128.15     133.52   0.960
## `age:bmi`                                      56.35     140.40   0.401
## `age:children`                                -76.74     144.76  -0.530
## `age:smoker`                                  -27.54     134.93  -0.204
## `age:regionnortheast`                        -275.92     166.33  -1.659
## `age:regionnorthwest`                        -171.89     166.67  -1.031
## `age:regionsoutheast`                          46.16     170.23   0.271
## `sex:bmi`                                      21.18     140.31   0.151
## `sex:children`                               -132.98     133.88  -0.993
## `sex:smoker`                                  -17.04     136.16  -0.125
## `sex:regionnortheast`                         -66.66     163.73  -0.407
## `sex:regionnorthwest`                          23.10     163.46   0.141
## `sex:regionsoutheast`                          91.84     167.86   0.547
## `bmi:children`                                 20.86     140.03   0.149
## `bmi:smoker`                                 3640.74     138.06  26.371
## `bmi:regionnortheast`                         200.37     175.48   1.142
## `bmi:regionnorthwest`                         166.98     188.80   0.884
## `bmi:regionsoutheast`                        -166.45     177.05  -0.940
## `children:smoker`                            -188.07     140.31  -1.340
## `children:regionnortheast`                    209.22     159.62   1.311
## `children:regionnorthwest`                    330.57     163.30   2.024
## `children:regionsoutheast`                     95.67     164.83   0.580
## `smoker:regionnortheast`                     -149.10     169.49  -0.880
## `smoker:regionnorthwest`                     -213.67     173.33  -1.233
## `smoker:regionsoutheast`                     -360.87     167.48  -2.155
## `regionnortheast:regionnorthwest`                 NA         NA      NA
## `regionnortheast:regionsoutheast`                 NA         NA      NA
## `regionnorthwest:regionsoutheast`                 NA         NA      NA
##                                             Pr(>|t|)
## (Intercept)                                  < 2e-16 ***
## `(Intercept)`                                     NA
## age                                          < 2e-16 ***
## sex                                         0.064082 .
## bmi                                          < 2e-16 ***
## children                                    1.90e-07 ***
## smoker                                       < 2e-16 ***
## regionnortheast                             0.000495 ***
## regionnorthwest                             0.102051
## regionsoutheast                             0.445523
## `poly(age, 2, raw = TRUE)[, 2]`             1.78e-06 ***
## `poly(sex, 2, raw = TRUE)[, 2]`                   NA
## `poly(bmi, 2, raw = TRUE)[, 2]`             0.032880 *
## `poly(children, 2, raw = TRUE)[, 2]`        0.238974
## `poly(smoker, 2, raw = TRUE)[, 2]`                NA
## `poly(regionnortheast, 2, raw = TRUE)[, 2]`       NA
## `poly(regionnorthwest, 2, raw = TRUE)[, 2]`       NA
## `poly(regionsoutheast, 2, raw = TRUE)[, 2]`       NA
## `age:sex`                                   0.337350
## `age:bmi`                                   0.688223
## `age:children`                              0.596129
## `age:smoker`                                0.838324
## `age:regionnortheast`                       0.097374 .
## `age:regionnorthwest`                       0.302569
## `age:regionsoutheast`                       0.786300
## `sex:bmi`                                   0.880022
## `sex:children`                              0.320775
## `sex:smoker`                                0.900418
## `sex:regionnortheast`                       0.683960
## `sex:regionnorthwest`                       0.887656
## `sex:regionsoutheast`                       0.584408
## `bmi:children`                              0.881601
## `bmi:smoker`                                 < 2e-16 ***
## `bmi:regionnortheast`                       0.253707
## `bmi:regionnorthwest`                       0.376620
## `bmi:regionsoutheast`                       0.347310
## `children:smoker`                           0.180356
## `children:regionnortheast`                  0.190173
## `children:regionnorthwest`                  0.043136 *
## `children:regionsoutheast`                  0.561730
## `smoker:regionnortheast`                    0.379205
## `smoker:regionnorthwest`                    0.217888
## `smoker:regionsoutheast`                    0.031363 *
## `regionnortheast:regionnorthwest`                 NA
## `regionnortheast:regionsoutheast`                 NA
## `regionnorthwest:regionsoutheast`                 NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4792 on 1300 degrees of freedom
## Multiple R-squared:  0.8477, Adjusted R-squared:  0.8435
## F-statistic: 200.9 on 36 and 1300 DF,  p-value: < 2.2e-16
```

As we can see we could drop a lot of unnecessary variables in our polynomial regression model without any big changes in the R-squared value. Looking at the cross validation results we see a big improvement in the RMSE which has a mean value of 5138 after 10-fold cross validation and a mean R-squared of 0.8420. Also, the model is not overfitting which is noticeable in the cross validation plot since model generalizes well on the testing data.

```r
# Step function
set.seed(50)
model_poly2_step <- step(model_poly2, direction = "backward", trace = 0)

# Summary
summary(model_poly2_step)
```

```
##
## Call:
## lm(formula = charges ~ age + sex + bmi + children + smoker +
##     regionnortheast + regionnorthwest + `poly(age, 2, raw = TRUE)[, 2]` +
##     `poly(bmi, 2, raw = TRUE)[, 2]` + `poly(children, 2, raw = TRUE)[, 2]` +
##     `age:regionnortheast` + `age:regionnorthwest` + `bmi:smoker` +
##     `bmi:regionnortheast` + `bmi:regionnorthwest` + `children:smoker` +
##     `children:regionnorthwest` + `smoker:regionsoutheast`, data = poly2_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11191.9  -1767.4  -1299.0   -436.1  31188.1
##
## Coefficients:
##                                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)                           12970.2      242.7  53.446  < 2e-16 ***
## age                                    3631.4      132.7  27.374  < 2e-16 ***
## sex                                    -232.5      131.7  -1.766 0.077656 .
## bmi                                    2014.3      139.2  14.466  < 2e-16 ***
## children                                969.1      181.0   5.354 1.02e-07 ***
## smoker                                 9653.3      132.0  73.151  < 2e-16 ***
## regionnortheast                         527.6      142.1   3.713 0.000213 ***
## regionnorthwest                         230.8      143.8   1.605 0.108799
## `poly(age, 2, raw = TRUE)[, 2]`         787.1      161.0   4.889 1.14e-06 ***
## `poly(bmi, 2, raw = TRUE)[, 2]`        -240.5      104.3  -2.307 0.021219 *
## `poly(children, 2, raw = TRUE)[, 2]`   -167.9      118.2  -1.421 0.155462
## `age:regionnortheast`                  -300.1      139.5  -2.152 0.031614 *
## `age:regionnorthwest`                  -212.3      139.2  -1.525 0.127401
## `bmi:smoker`                           3627.2      133.1  27.242  < 2e-16 ***
## `bmi:regionnortheast`                   280.5      148.2   1.892 0.058699 .
## `bmi:regionnorthwest`                   246.5      165.4   1.490 0.136395
## `children:smoker`                      -204.0      136.8  -1.491 0.136245
## `children:regionnorthwest`              233.0      134.8   1.728 0.084153 .
## `smoker:regionsoutheast`               -212.4      130.9  -1.623 0.104859
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4773 on 1318 degrees of freedom
## Multiple R-squared:  0.8468, Adjusted R-squared:  0.8447
## F-statistic: 404.6 on 18 and 1318 DF,  p-value: < 2.2e-16
```

```r
# Apply cv function
set.seed(50)
cv_results_poly2_step <- perform_cv(poly2_data, stepwise = TRUE, 10)
```

```
## Fold 1 Train RMSE: 4855.312 Test RMSE: 3703.722 R-squared: 0.8395426
## Fold 2 Train RMSE: 4647.135 Test RMSE: 5593.697 R-squared: 0.8562569
## Fold 3 Train RMSE: 4788.358 Test RMSE: 4387.409 R-squared: 0.8474196
## Fold 4 Train RMSE: 4762.378 Test RMSE: 4702.731 R-squared: 0.8402857
## Fold 5 Train RMSE: 4689.307 Test RMSE: 5361.939 R-squared: 0.8486897
## Fold 6 Train RMSE: 4696.726 Test RMSE: 5366.524 R-squared: 0.8502353
## Fold 7 Train RMSE: 4825.604 Test RMSE: 3976.39 R-squared: 0.8413785
## Fold 8 Train RMSE: 4682.693 Test RMSE: 5404.532 R-squared: 0.8489357
## Fold 9 Train RMSE: 4736.4 Test RMSE: 4960.876 R-squared: 0.8475809
## Fold 10 Train RMSE: 4719.959 Test RMSE: 5091.826 R-squared: 0.8456102
```

```r
cv_mean_results_poly2_step <- cv_results_poly2_step$mean_results
cv_mean_results_poly2_step
```

```
##   mean_rmse_train mean_rmse_test mean_r_squared
## 1        4740.387       4854.965      0.8465935
```

```r
# Apply cv plot function to detect overfitting
plot_cv_results(cv_results_poly2_step)
```

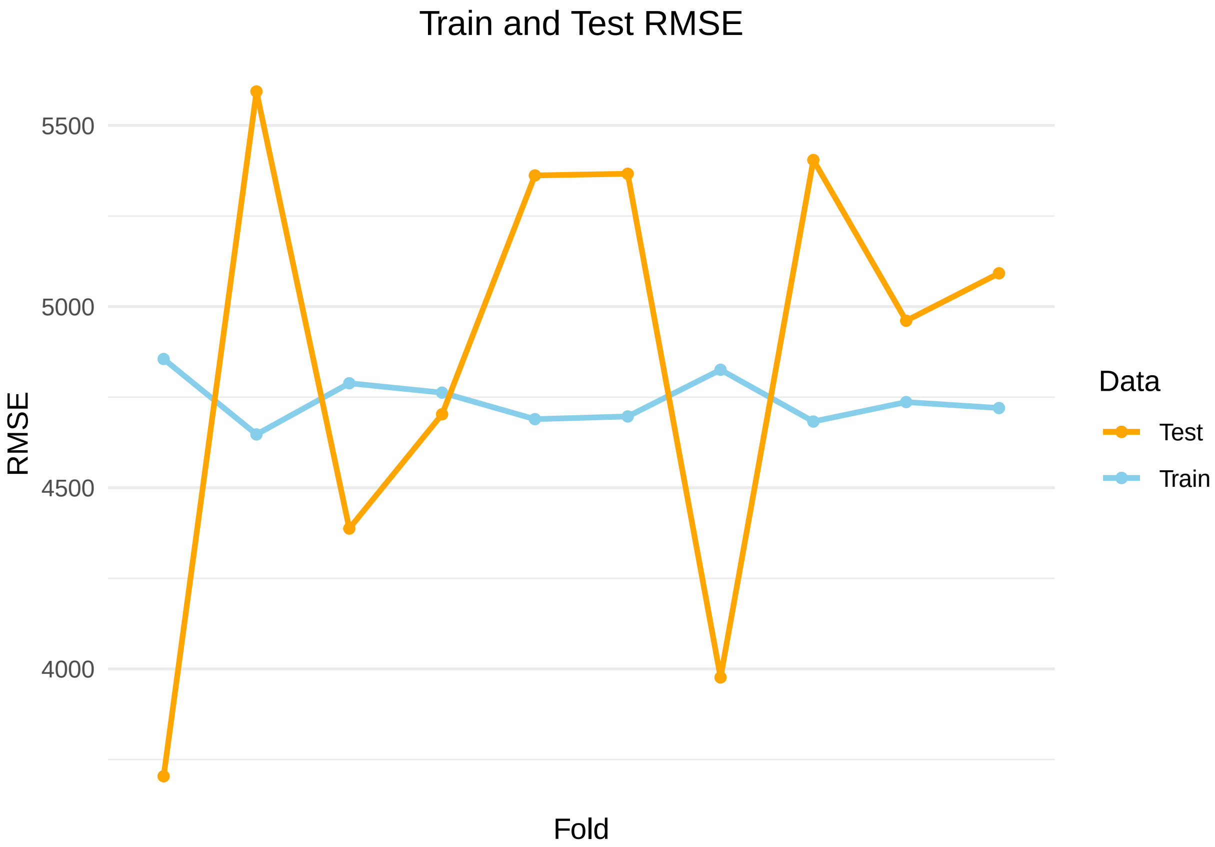


Figure 16: Fig: 3.6: Cross-Validation Poly2-Model

```r
# Combine
comb <- rbind(comb, cv_mean_results_poly2_step)
rownames(comb) <- c("LR-full-model", "LR-step-model", "Poly2-step-model")
comb
```

```
##                  mean_rmse_train mean_rmse_test mean_r_squared
## LR-full-model           6041.394       6068.038      0.7508852
## LR-step-model           6043.749       6067.521      0.7506905
## Poly2-step-model        4740.387       4854.965      0.8465935
```

We perform a residual analysis for the 2nd polynomial regression model. Compared to the linear regression, there are a lot more high leverage points. This is because the model includes higher-order terms of the predictor variables which can result in a more complex and flexible relationship between the predictor and response variable. This increased flexibility can allow the model to fit more closely to the data, but it can also make the model more sensitive to extreme values, leading to high leverage points. We further calculate the Cook's distance of these influential observations, which measures the influence of each observation on the regression coefficients. A high Cook's distance indicates that a particular observation has a large influence on the model and may be an outlier. In this case, the Cook's distance for the rows with high leverage are still very small and generally smaller than 1, indicating that they do not have a significant influence on the model. Based on this information we keep the data points in our analysis (James et al., 2013).

A look at the Residuals vs. Fitted plot shows that the residuals are better captured in the polynomial regression. Unfortunately, in the QQ plot it still looks like the residuals are not normally distributed. This is also shown by the Shapiro-Wilk test, which is still significant for the residuals. However, we were able to improve the heterescedasticity of our residuals when we look at the scale-location plot, as there are fewer points that are far from the red line. This also supports the Breusch-Pagan test, which is no longer significant. Finally, the Residuals vs. Leverage chart shows that there are no true outliers. The largest outlier has a Cook's distance of just over 0.1, which is still very small. Also, there is no sign of heteroskedasticity in the coefficients when looking at the VIF values, which are very small and around 1.

In conclusion, the polynomial regression model with 2nd degree captures our data way better than the linear regression model. Also our metrics, RMSE and R-squared could be improved significantly.

```r
# Leverage
plot(hatvalues(model_poly2_step), pch=19, main="Leverage", cex=0.5)
n <- nrow(scaled_data)
p <- ncol(scaled_data)
abline(h=(1:3)*(p+1)/n, col=c("black", "darkred", "red"))
```

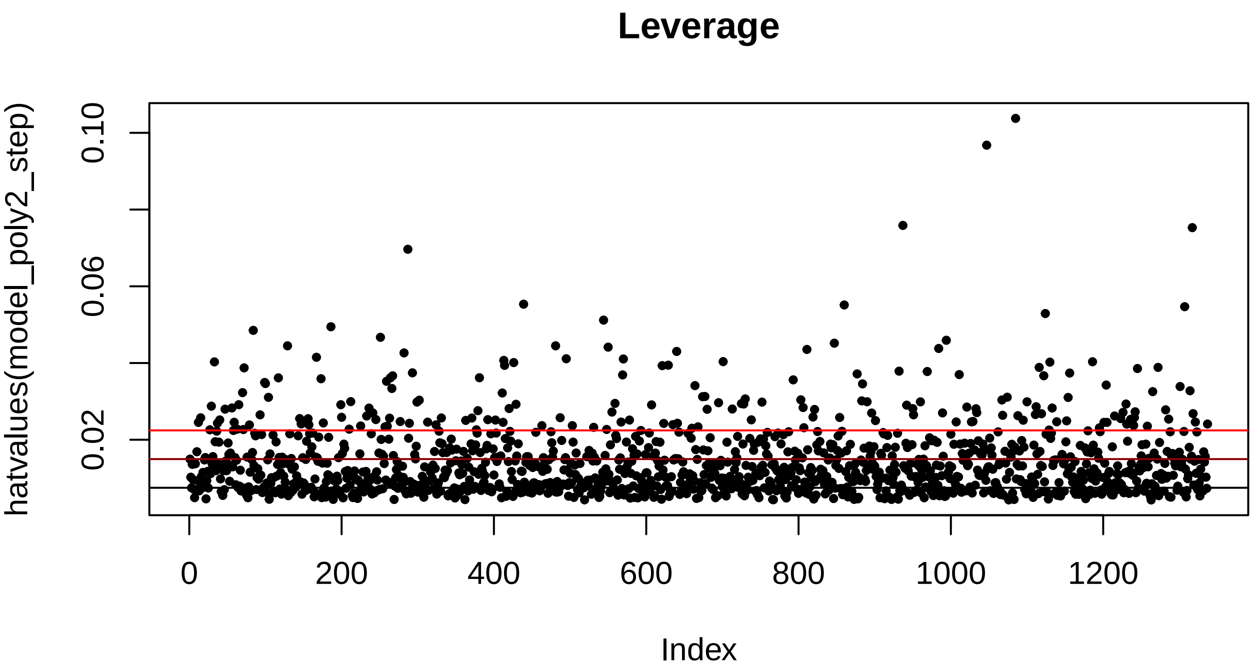


Figure 17: Fig: 3.7: Leverage Plot Ploy2-Model

```r
# Residual analysis
par(mfrow = c(2, 2))
plot(model_poly2_step, col = "darkblue")
```

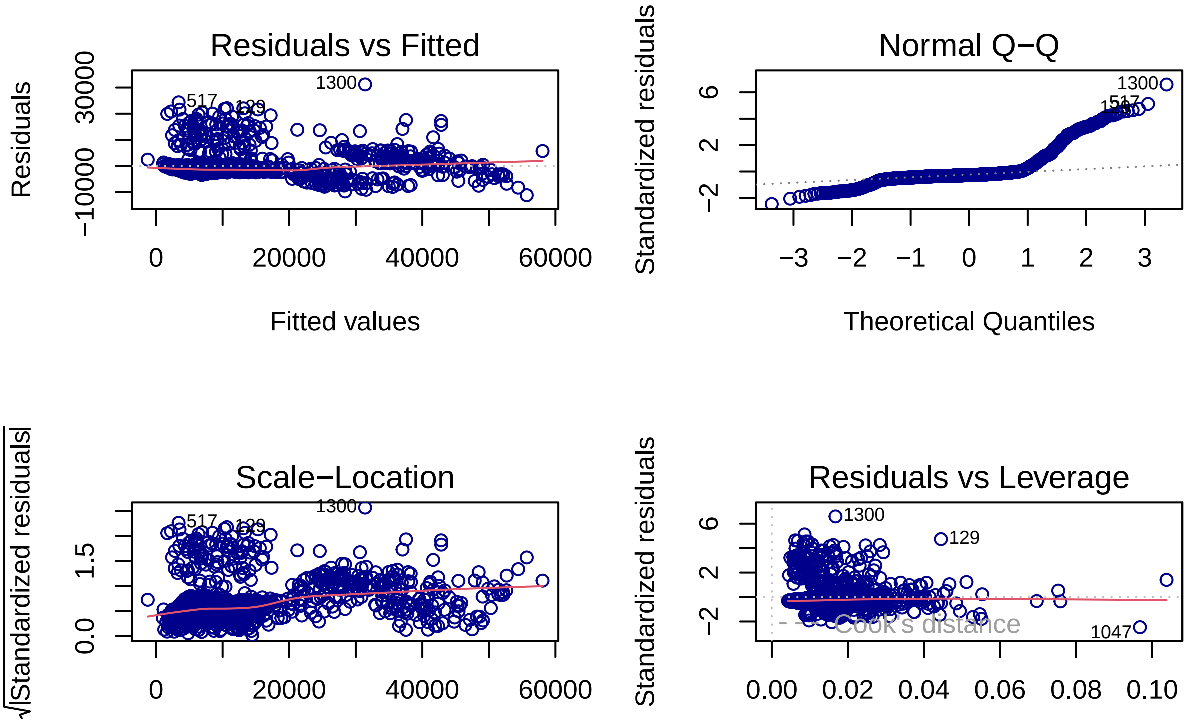


Figure 18: Fig: 3.8: Residual Plots Poly2-Model

```
# Find rows with high leverage
high_leverage_rows <- which(hatvalues(model_poly2_step) > (p+1)*3/n)

# Print the row numbers
cat("Number of very high leverage points:", paste(length(cooks_dist), collapse = ", "))
```

```
## Number of very high leverage points: 0
```

```
# Calculate Cook's distance for rows with high leverage
cooks_dist <- cooks.distance(model_poly2_step)[high_leverage_rows]

# Only showing Cook's distance values greater than 1
cooks_dist[cooks_dist > 1]
```

```
## named numeric(0)
```

```
# Test on multicollinearity by variance inflation factor for each predictor
vif(model_poly2_step)
```

```
##                                      age                                      sex
##                                 1.032043                                 1.016595
##                                      bmi                                 children
##                                 1.136990                                 1.921351
##                                   smoker                          regionnortheast
##                                 1.021211                                 1.183979
##                          regionnorthwest        `poly(age, 2, raw = TRUE)[, 2]`
##                                 1.212768                                 1.148023
##        `poly(bmi, 2, raw = TRUE)[, 2]` `poly(children, 2, raw = TRUE)[, 2]`
##                             1.237517                             1.796207
##                `age:regionnortheast`                `age:regionnorthwest`
##                             1.140595                             1.131745
##                         `bmi:smoker`                `bmi:regionnortheast`
##                             1.092625                             1.267200
##                `bmi:regionnorthwest`                    `children:smoker`
##                             1.319471                             1.030985
##           `children:regionnorthwest`             `smoker:regionsoutheast`
##                             1.023997                             1.101768
```

```
# Extract the residuals
resid <- residuals(model_poly2_step)

# Perform Shapiro-Wilk test for normality of residuals
shapiro.test(resid)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  resid
## W = 0.66128, p-value < 2.2e-16
```

```
# Breusch-Pagan Test to detect heteroscedasticity
bptest(model_poly2_step)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  model_poly2_step
## BP = 20.845, df = 18, p-value = 0.2873
```

To see if we can continue the improvements by making the model even more complex, we try polynomial regression 3. The process is actually the same as for polynomial regression model 2. We first create the formula and data set and fit the full model with all variables.

Here, the number of predictors increases even more. We see an R-squared value of 0.8543, which is slightly larger than that of the full model for regression 2.

3.3.1 Polynomial Regression (3nd degree)

```
# Build formula for polynomial regression full model 3rd degree
formula <- as.formula(
  paste(
    ' ~ .^3 + ',
    paste('poly(', colnames(X), ', 3, raw=TRUE)[, 2:3]', collapse = ' + ')
  )
)

# Check formula
formula
```

```
## ~.^3 + poly(age, 3, raw = TRUE)[, 2:3] + poly(sex, 3, raw = TRUE)[,
##     2:3] + poly(bmi, 3, raw = TRUE)[, 2:3] + poly(children, 3,
##     raw = TRUE)[, 2:3] + poly(smoker, 3, raw = TRUE)[, 2:3] +
##     poly(regionnortheast, 3, raw = TRUE)[, 2:3] + poly(regionnorthwest,
##     3, raw = TRUE)[, 2:3] + poly(regionsoutheast, 3, raw = TRUE)[,
##     2:3]
```

```
# Build data frame for polynomial regression
poly_data3 <- as.data.frame(model.matrix(formula, data = X))
poly_data3$charges <- y

# Build model polynomial regression
set.seed(50)
model_poly3 <- lm(formula = charges ~., data = poly_data3)

# Summary
summary(model_poly3)
```

```
##
## Call:
## lm(formula = charges ~ ., data = poly_data3)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9508.8 -1913.0 -1165.9  -270.3 30109.0
##
## Coefficients: (30 not defined because of singularities)
##                                                    Estimate Std. Error t value
## (Intercept)                                       12936.114    288.663  44.814
## `(Intercept)`                                            NA         NA      NA
## age                                                3774.401    349.451  10.801
## sex                                                -281.718    140.634  -2.003
## bmi                                                2672.274    222.979  11.984
## children                                            897.419    208.913   4.296
## smoker                                             9699.178    144.801  66.983
## regionnortheast                                     585.211    169.992   3.443
## regionnorthwest                                     284.244    171.516   1.657
## regionsoutheast                                      87.352    177.109   0.493
## `poly(age, 3, raw = TRUE)[, 2:3]2`                  835.965    172.589   4.844
## `poly(age, 3, raw = TRUE)[, 2:3]3`                 -107.676    187.819  -0.573
## `poly(sex, 3, raw = TRUE)[, 2:3]2`                       NA         NA      NA
## `poly(sex, 3, raw = TRUE)[, 2:3]3`                       NA         NA      NA
## `poly(bmi, 3, raw = TRUE)[, 2:3]2`                  -98.086    128.863  -0.761
## `poly(bmi, 3, raw = TRUE)[, 2:3]3`                 -207.671     65.872  -3.153
## `poly(children, 3, raw = TRUE)[, 2:3]2`            -334.070    284.183  -1.176
## `poly(children, 3, raw = TRUE)[, 2:3]3`              80.035    103.521   0.773
## `poly(smoker, 3, raw = TRUE)[, 2:3]2`                    NA         NA      NA
## `poly(smoker, 3, raw = TRUE)[, 2:3]3`                    NA         NA      NA
## `poly(regionnortheast, 3, raw = TRUE)[, 2:3]2`           NA         NA      NA
## `poly(regionnortheast, 3, raw = TRUE)[, 2:3]3`           NA         NA      NA
## `poly(regionnorthwest, 3, raw = TRUE)[, 2:3]2`           NA         NA      NA
## `poly(regionnorthwest, 3, raw = TRUE)[, 2:3]3`           NA         NA      NA
## `poly(regionsoutheast, 3, raw = TRUE)[, 2:3]2`           NA         NA      NA
## `poly(regionsoutheast, 3, raw = TRUE)[, 2:3]3`           NA         NA      NA
## `age:sex`                                           122.050    141.239   0.864
## `age:bmi`                                            51.335    157.925   0.325
## `age:children`                                      -99.448    152.984  -0.650
## `age:smoker`                                        -83.824    143.043  -0.586
## `age:regionnortheast`                              -373.151    177.042  -2.108
## `age:regionnorthwest`                              -206.085    175.374  -1.175
## `age:regionsoutheast`                                66.261    182.149   0.364
## `sex:bmi`                                            91.892    143.980   0.638
## `sex:children`                                     -104.868    136.581  -0.768
## `sex:smoker`                                        -80.347    139.429  -0.576
## `sex:regionnortheast`                              -101.646    167.812  -0.606
## `sex:regionnorthwest`                               -86.331    169.298  -0.510
## `sex:regionsoutheast`                                47.605    173.041   0.275
## `bmi:children`                                        6.149    146.811   0.042
## `bmi:smoker`                                       3854.340    153.461  25.116
## `bmi:regionnortheast`                               187.662    178.147   1.053
## `bmi:regionnorthwest`                               169.923    193.994   0.876
## `bmi:regionsoutheast`                              -169.912    180.830  -0.940
## `children:smoker`                                  -162.594    148.905  -1.092
## `children:regionnortheast`                          172.658    167.515   1.031
## `children:regionnorthwest`                          408.244    170.798   2.390
## `children:regionsoutheast`                           84.963    171.283   0.496
## `smoker:regionnortheast`                           -136.543    177.959  -0.767
## `smoker:regionnorthwest`                           -135.785    183.658  -0.739
## `smoker:regionsoutheast`                           -299.870    176.599  -1.698
## `regionnortheast:regionnorthwest`                        NA         NA      NA
## `regionnortheast:regionsoutheast`                        NA         NA      NA
## `regionnorthwest:regionsoutheast`                        NA         NA      NA
## `age:sex:bmi`                                      -195.407    143.364  -1.363
## `age:sex:children`                                   43.256    147.633   0.293
## `age:sex:smoker`                                    280.234    140.436   1.995
## `age:sex:regionnortheast`                            97.728    168.576   0.580
## `age:sex:regionnorthwest`                           185.345    168.277   1.101
## `age:sex:regionsoutheast`                           -35.023    171.223  -0.205
## `age:bmi:children`                                  175.173    166.785   1.050
## `age:bmi:smoker`                                    140.781    152.723   0.922
## `age:bmi:regionnortheast`                            49.815    173.373   0.287
## `age:bmi:regionnorthwest`                           112.447    189.219   0.594
## `age:bmi:regionsoutheast`                            43.259    171.260   0.253
## `age:children:smoker`                               -18.560    155.786  -0.119
## `age:children:regionnortheast`                     -301.708    177.124  -1.703
## `age:children:regionnorthwest`                     -243.519    179.821  -1.354
## `age:children:regionsoutheast`                      -40.099    183.873  -0.218
## `age:smoker:regionnortheast`                       -108.017    182.428  -0.592
## `age:smoker:regionnorthwest`                        -25.684    180.262  -0.142
## `age:smoker:regionsoutheast`                        -59.383    171.723  -0.346
## `age:regionnortheast:regionnorthwest`                    NA         NA      NA
## `age:regionnortheast:regionsoutheast`                    NA         NA      NA
## `age:regionnorthwest:regionsoutheast`                    NA         NA      NA
## `sex:bmi:children`                                  130.591    145.089   0.900
## `sex:bmi:smoker`                                   -116.056    142.480  -0.815
## `sex:bmi:regionnortheast`                          -218.305    176.696  -1.235
## `sex:bmi:regionnorthwest`                          -197.435    190.893  -1.034
## `sex:bmi:regionsoutheast`                          -120.597    175.262  -0.688
## `sex:children:smoker`                               -42.407    148.667  -0.285
## `sex:children:regionnortheast`                       75.896    162.592   0.467
## `sex:children:regionnorthwest`                       83.170    164.403   0.506
## `sex:children:regionsoutheast`                      130.472    166.829   0.782
## `sex:smoker:regionnortheast`                         14.637    175.068   0.084
## `sex:smoker:regionnorthwest`                       -336.710    183.659  -1.833
## `sex:smoker:regionsoutheast`                        132.803    172.929   0.768
## `sex:regionnortheast:regionnorthwest`                    NA         NA      NA
## `sex:regionnortheast:regionsoutheast`                    NA         NA      NA
## `sex:regionnorthwest:regionsoutheast`                    NA         NA      NA
## `bmi:children:smoker`                              -264.965    150.353  -1.762
## `bmi:children:regionnortheast`                       23.012    174.681   0.132
## `bmi:children:regionnorthwest`                      198.519    192.884   1.029
## `bmi:children:regionsoutheast`                       23.229    169.391   0.137
## `bmi:smoker:regionnortheast`                       -169.850    186.927  -0.909
## `bmi:smoker:regionnorthwest`                        101.734    216.432   0.470
## `bmi:smoker:regionsoutheast`                       -321.756    180.313  -1.784
## `bmi:regionnortheast:regionnorthwest`                    NA         NA      NA
## `bmi:regionnortheast:regionsoutheast`                    NA         NA      NA
## `bmi:regionnorthwest:regionsoutheast`                    NA         NA      NA
## `children:smoker:regionnortheast`                   -98.064    179.709  -0.546
## `children:smoker:regionnorthwest`                   251.586    177.219   1.420
## `children:smoker:regionsoutheast`                    15.314    179.756   0.085
## `children:regionnortheast:regionnorthwest`               NA         NA      NA
## `children:regionnortheast:regionsoutheast`               NA         NA      NA
## `children:regionnorthwest:regionsoutheast`               NA         NA      NA
## `smoker:regionnortheast:regionnorthwest`                 NA         NA      NA
## `smoker:regionnortheast:regionsoutheast`                 NA         NA      NA
## `smoker:regionnorthwest:regionsoutheast`                 NA         NA      NA
## `regionnortheast:regionnorthwest:regionsoutheast`        NA         NA      NA
##                                                   Pr(>|t|)    
## (Intercept)                                        < 2e-16 ***
## `(Intercept)`                                           NA    
## age                                                < 2e-16 ***
## sex                                               0.045370 *  
## bmi                                                < 2e-16 ***
## children                                          1.88e-05 ***
## smoker                                             < 2e-16 ***
## regionnortheast                                   0.000595 ***
## regionnorthwest                                   0.097721 .  
## regionsoutheast                                   0.621948    
## `poly(age, 3, raw = TRUE)[, 2:3]2`                1.43e-06 ***
## `poly(age, 3, raw = TRUE)[, 2:3]3`                0.566548    
## `poly(sex, 3, raw = TRUE)[, 2:3]2`                      NA    
## `poly(sex, 3, raw = TRUE)[, 2:3]3`                      NA    
## `poly(bmi, 3, raw = TRUE)[, 2:3]2`                0.446700    
## `poly(bmi, 3, raw = TRUE)[, 2:3]3`                0.001656 ** 
## `poly(children, 3, raw = TRUE)[, 2:3]2`           0.239999    
## `poly(children, 3, raw = TRUE)[, 2:3]3`           0.439593    
## `poly(smoker, 3, raw = TRUE)[, 2:3]2`                   NA    
## `poly(smoker, 3, raw = TRUE)[, 2:3]3`                   NA    
## `poly(regionnortheast, 3, raw = TRUE)[, 2:3]2`          NA    
## `poly(regionnortheast, 3, raw = TRUE)[, 2:3]3`          NA    
## `poly(regionnorthwest, 3, raw = TRUE)[, 2:3]2`          NA    
## `poly(regionnorthwest, 3, raw = TRUE)[, 2:3]3`          NA    
## `poly(regionsoutheast, 3, raw = TRUE)[, 2:3]2`          NA    
## `poly(regionsoutheast, 3, raw = TRUE)[, 2:3]3`          NA    
## `age:sex`                                         0.387674    
## `age:bmi`                                         0.745188    
## `age:children`                                    0.515775    
## `age:smoker`                                      0.557979    
## `age:regionnortheast`                              0.035255 *
## `age:regionnorthwest`                              0.240170
## `age:regionsoutheast`                              0.716088
## `sex:bmi`                                          0.523444
## `sex:children`                                     0.442747
## `sex:smoker`                                       0.564544
## `sex:regionnortheast`                              0.544814
## `sex:regionnorthwest`                              0.610187
## `sex:regionsoutheast`                              0.783278
## `bmi:children`                                     0.966600
## `bmi:smoker`                                        < 2e-16 ***
## `bmi:regionnortheast`                              0.292354
## `bmi:regionnorthwest`                              0.381241
## `bmi:regionsoutheast`                              0.347591
## `children:smoker`                                  0.275073
## `children:regionnortheast`                         0.302879
## `children:regionnorthwest`                         0.016985 *
## `children:regionsoutheast`                         0.619954
## `smoker:regionnortheast`                           0.443063
## `smoker:regionnorthwest`                           0.459841
## `smoker:regionsoutheast`                           0.089749 .
## `regionnortheast:regionnorthwest`                        NA
## `regionnortheast:regionsoutheast`                        NA
## `regionnorthwest:regionsoutheast`                        NA
## `age:sex:bmi`                                      0.173123
## `age:sex:children`                                 0.769570
## `age:sex:smoker`                                   0.046208 *
## `age:sex:regionnortheast`                          0.562204
## `age:sex:regionnorthwest`                          0.270921
## `age:sex:regionsoutheast`                          0.837960
## `age:bmi:children`                                 0.293787
## `age:bmi:smoker`                                   0.356807
## `age:bmi:regionnortheast`                          0.773906
## `age:bmi:regionnorthwest`                          0.552438
## `age:bmi:regionsoutheast`                          0.800624
## `age:children:smoker`                              0.905186
## `age:children:regionnortheast`                     0.088746 .
## `age:children:regionnorthwest`                     0.175907
## `age:children:regionsoutheast`                     0.827403
## `age:smoker:regionnortheast`                       0.553883
## `age:smoker:regionnorthwest`                       0.886724
## `age:smoker:regionsoutheast`                       0.729547
## `age:regionnortheast:regionnorthwest`                    NA
## `age:regionnortheast:regionsoutheast`                    NA
## `age:regionnorthwest:regionsoutheast`                    NA
## `sex:bmi:children`                                 0.368252
## `sex:bmi:smoker`                                   0.415488
## `sex:bmi:regionnortheast`                          0.216880
## `sex:bmi:regionnorthwest`                          0.301209
## `sex:bmi:regionsoutheast`                          0.491519
## `sex:children:smoker`                              0.775502
## `sex:children:regionnortheast`                     0.640731
## `sex:children:regionnorthwest`                     0.613021
## `sex:children:regionsoutheast`                     0.434320
## `sex:smoker:regionnortheast`                      0.933382
## `sex:smoker:regionnorthwest`                      0.066988 .
## `sex:smoker:regionsoutheast`                      0.442653
## `sex:regionnortheast:regionnorthwest`                   NA
## `sex:regionnortheast:regionsoutheast`                   NA
## `sex:regionnorthwest:regionsoutheast`                   NA
## `bmi:children:smoker`                             0.078264 .
## `bmi:children:regionnortheast`                    0.895213
## `bmi:children:regionnorthwest`                    0.303576
## `bmi:children:regionsoutheast`                    0.890950
## `bmi:smoker:regionnortheast`                      0.363712
## `bmi:smoker:regionnorthwest`                      0.638402
## `bmi:smoker:regionsoutheast`                      0.074595 .
## `bmi:regionnortheast:regionnorthwest`                   NA
## `bmi:regionnortheast:regionsoutheast`                   NA
## `bmi:regionnorthwest:regionsoutheast`                   NA
## `children:smoker:regionnortheast`                 0.585382
## `children:smoker:regionnorthwest`                 0.155963
## `children:smoker:regionsoutheast`                 0.932123
## `children:regionnortheast:regionnorthwest`              NA
## `children:regionnortheast:regionsoutheast`              NA
## `children:regionnorthwest:regionsoutheast`              NA
## `smoker:regionnortheast:regionnorthwest`                NA
## `smoker:regionnortheast:regionsoutheast`                NA
## `smoker:regionnorthwest:regionsoutheast`                NA
## `regionnortheast:regionnorthwest:regionsoutheast`       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4766 on 1257 degrees of freedom
## Multiple R-squared:  0.8543, Adjusted R-squared:  0.8451
## F-statistic: 93.28 on 79 and 1257 DF,  p-value: < 2.2e-16
```

In addition, the step function is used to reduce the number of unnecessary predictors and to reduce the dimensionality of the model. In this model, all predictor variables except gender have a significant effect on health care costs, and smoking status has the strongest effect on health care costs, followed by age and BMI. In addition, the model has an R-squared value of 0.8514, which is slightly worse than that of the full third-degree polynomial regression model.

Looking at the cross-validation results, we cannot see any clear evidence of overfitting. However, the mean RMSE performance for the test data (mean RMSE = 5181) is slightly worse than for the second-degree polynomial regression (mean RMSE = 5138). However, mean R-squared (0.8449) is slightly better than for the second degree polynomial regression (0.8429).

```
# Step function
set.seed(50)
model_poly_step3 <- step(model_poly3, direction = "backward", trace = 0)

# Summary
summary(model_poly_step3)
```

```
##
## Call:
## lm(formula = charges ~ age + sex + bmi + children + smoker +
##     regionnortheast + regionnorthwest + `poly(age, 3, raw = TRUE)[, 2:3]2` +
##     `poly(bmi, 3, raw = TRUE)[, 2:3]3` + `age:regionnortheast` +
##     `age:regionnorthwest` + `bmi:smoker` + `bmi:regionnortheast` +
##     `bmi:regionnorthwest` + `children:regionnorthwest` + `age:sex:bmi` +
##     `age:sex:smoker` + `age:children:regionnortheast` + `age:children:regionnorthwest` +
##     `sex:smoker:regionnorthwest` + `bmi:children:smoker` + `bmi:smoker:regionnortheast` +
##     `bmi:smoker:regionsoutheast` + `children:smoker:regionnorthwest`,
##     data = poly_data3)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8913.2 -1819.5 -1181.2  -491.2 30148.3
##
## Coefficients:
##                                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)                       12629.32     205.63  61.417  < 2e-16 ***
## age                                3610.61     130.86  27.592  < 2e-16 ***
## sex                                -234.20     131.65  -1.779  0.07546 .
## bmi                                2770.91     204.98  13.518  < 2e-16 ***
## children                            821.68     136.54   6.018 2.29e-09 ***
## smoker                             9669.77     134.95  71.653  < 2e-16 ***
## regionnortheast                     562.96     140.50   4.007 6.50e-05 ***
## regionnorthwest                     298.51     142.13   2.100  0.03590 *
## `poly(age, 3, raw = TRUE)[, 2:3]2`  743.99     156.67   4.749 2.27e-06 ***
## `poly(bmi, 3, raw = TRUE)[, 2:3]3` -246.13      53.62  -4.590 4.86e-06 ***
## `age:regionnortheast`              -408.51     145.66  -2.805  0.00511 **
## `age:regionnorthwest`              -256.74     141.65  -1.812  0.07014 .
## `bmi:smoker`                       3780.34     138.36  27.323  < 2e-16 ***
## `bmi:regionnortheast`               277.05     141.18   1.962  0.04993 *
## `bmi:regionnorthwest`               282.46     158.70   1.780  0.07533 .
## `children:regionnorthwest`          306.79     133.53   2.297  0.02175 *
## `age:sex:bmi`                      -221.07     132.51  -1.668  0.09549 .
## `age:sex:smoker`                    257.04     131.86   1.949  0.05146 .
## `age:children:regionnortheast`     -332.11     146.58  -2.266  0.02363 *
## `age:children:regionnorthwest`     -232.72     149.66  -1.555  0.12019
## `sex:smoker:regionnorthwest`       -390.80     139.42  -2.803  0.00514 **
## `bmi:children:smoker`              -242.76     133.84  -1.814  0.06994 .
## `bmi:smoker:regionnortheast`       -252.97     145.44  -1.739  0.08220 .
## `bmi:smoker:regionsoutheast`       -403.13     137.94  -2.923  0.00353 **
## `children:smoker:regionnorthwest`   256.81     136.78   1.878  0.06066 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4710 on 1312 degrees of freedom
## Multiple R-squared:  0.8514, Adjusted R-squared:  0.8487
## F-statistic: 313.3 on 24 and 1312 DF,  p-value: < 2.2e-16
```

```
# Apply cv function
set.seed(50)
cv_results_poly3_step <- perform_cv(poly_data3, stepwise = TRUE, 10)
```

```
## Fold 1 Train RMSE: 4753.143 Test RMSE: 3872.294 R-squared: 0.8462245
## Fold 2 Train RMSE: 4577.792 Test RMSE: 5446.8 R-squared: 0.8605146
## Fold 3 Train RMSE: 4713.305 Test RMSE: 4386.299 R-squared: 0.8521652
## Fold 4 Train RMSE: 4672.6 Test RMSE: 4668.375 R-squared: 0.8462507
## Fold 5 Train RMSE: 4616.445 Test RMSE: 5346.287 R-squared: 0.8533553
## Fold 6 Train RMSE: 4602.635 Test RMSE: 5405.591 R-squared: 0.8561757
## Fold 7 Train RMSE: 4737.522 Test RMSE: 4133.733 R-squared: 0.8471163
## Fold 8 Train RMSE: 4586.403 Test RMSE: 5502.343 R-squared: 0.8550845
## Fold 9 Train RMSE: 4681.124 Test RMSE: 4841.626 R-squared: 0.8511178
## Fold 10 Train RMSE: 4630.294 Test RMSE: 5135.642 R-squared: 0.8514203
```

```
cv_mean_results_poly3_step <- cv_results_poly3_step$mean_results
cv_mean_results_poly3_step
```

```
##   mean_rmse_train mean_rmse_test mean_r_squared
## 1        4657.126       4873.899      0.8519425
```

```
# Apply cv plot function to detect overfitting
plot_cv_results(cv_results_poly3_step)
```

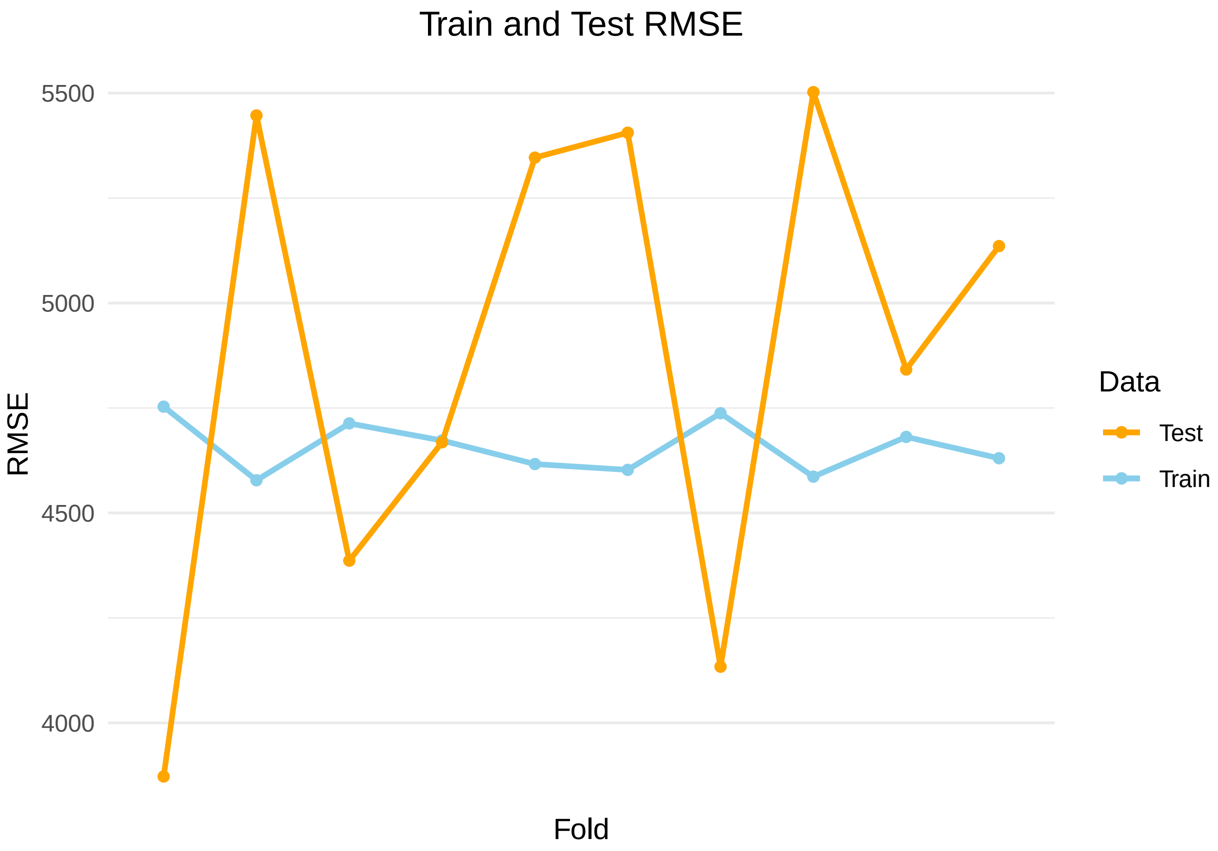


Figure 19: Fig: 3.9: Cross-Validation Poly3-Model

```
# Combine
comb <- rbind(comb, cv_mean_results_poly3_step)
rownames(comb) <- c("LR-full-model", "LR-step-model",
                    "Poly2-step-model", "Poly3-step-model")
comb
```

```
##                  mean_rmse_train mean_rmse_test mean_r_squared
## LR-full-model           6041.394       6068.038      0.7508852
## LR-step-model           6043.749       6067.521      0.7506905
## Poly2-step-model        4740.387       4854.965      0.8465935
## Poly3-step-model        4657.126       4873.899      0.8519425
```

There are actually no major differences in the residuals analysis compared to the 2nd degree polynomial regression. The plot of the residuals compared to the fitted values seems to show a slightly higher variance in the residuals. In addition, the residuals are still not normally distributed, as indicated by the QQ plot and the highly significant Shapiro-Wilk test. The Scale-Location plot and the result of the Breusch-Pagan test have also not changed significantly, indicating that the residuals are still not heteroskedastic. There is also no evidence of heteroskedasticity in the coefficients, looking at the VIF values, which are very small and around 1. Finally, we see an increased number of high leverage points as complexity has also increased. In the Residuals vs. Leverage plot, the largest leverage point has a Cook distance of about 0.25, which is still less than 1, suggesting that we keep all the high leverage points in the model.

```
# Leverage
plot(hatvalues(model_poly_step3), pch=19, main="Leverage", cex=0.5)
n <- nrow(scaled_data)
p <- ncol(scaled_data)
abline(h=(1:3)*(p+1)/n, col=c("black", "darkred", "red"))
```

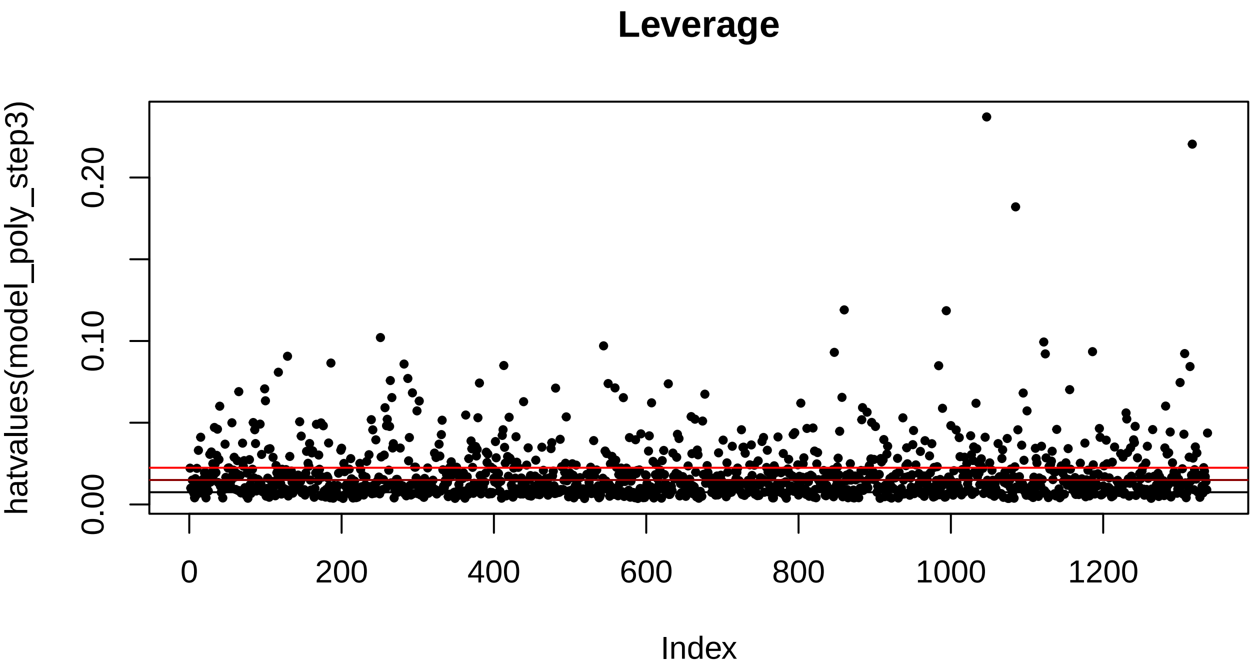


Figure 20: Fig: 3.10: Leverage Plot Poly3-Model

```
# Residual analysis
par(mfrow = c(2, 2))
plot(model_poly_step3, col = "darkblue")
```

```
# Find rows with high leverage
high_leverage_rows <- which(hatvalues(model_poly_step3) > (p+1)*3/n)

# Print the row numbers
cat("Number of high leverage points:", paste(length(cooks_dist), collapse = ", "))
```

```
## Number of high leverage points: 190
```

```
# Calculate Cook's distance for rows with high leverage
cooks_dist <- cooks.distance(model_poly_step3)[high_leverage_rows]

# Only showing Cook's distance values greater than 1
cooks_dist[cooks_dist > 1]
```

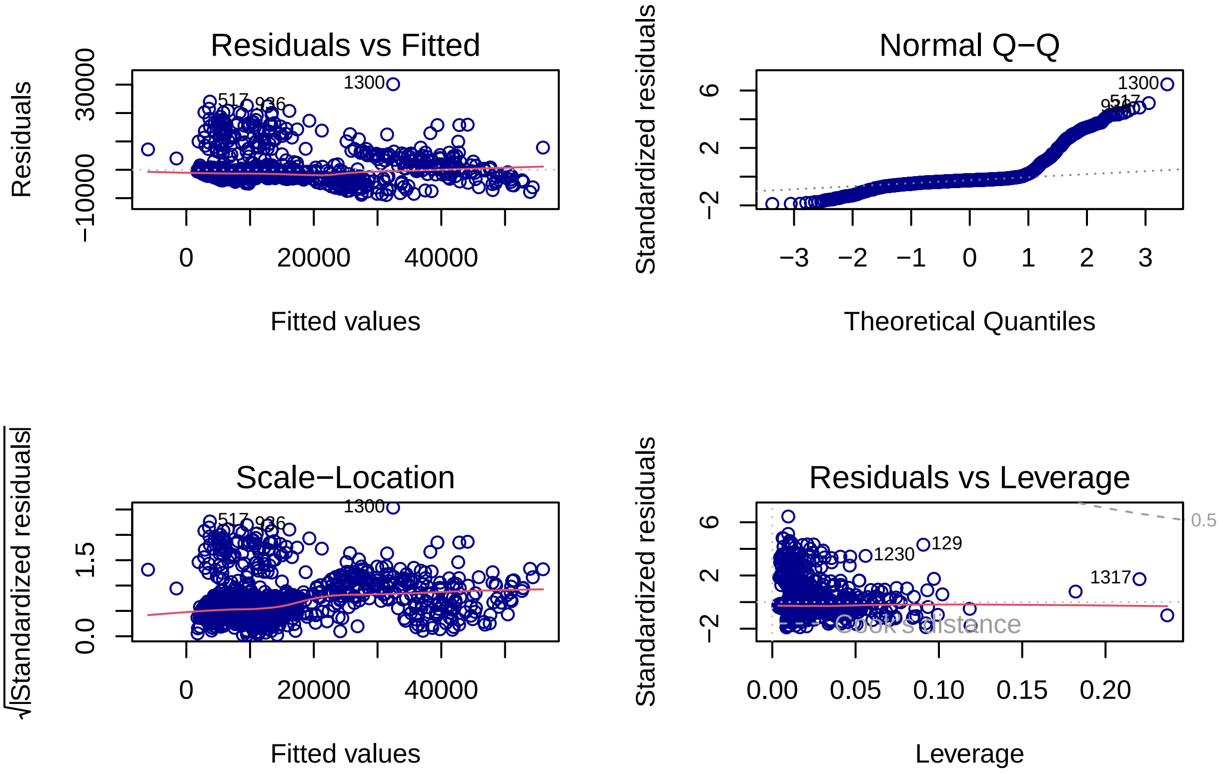


Figure 21: Fig: 3.11: Residual Plots Poly3-Model

```
## named numeric(0)
```

```
# Test on multicollinearity by variance inflation factor for each predictor
vif(model_poly_step3)
```

```
##                                age                                sex
##                           1.031113                           1.043560
##                                bmi                           children
##                           2.529930                           1.122555
##                             smoker                    regionnortheast
##                           1.096641                           1.188699
##                    regionnorthwest `poly(age, 3, raw = TRUE)[, 2:3]2`
##                           1.216428                           1.116215
## `poly(bmi, 3, raw = TRUE)[, 2:3]3`              `age:regionnortheast`
##                           2.579869                           1.277621
##              `age:regionnorthwest`                       `bmi:smoker`
##                           1.203545                           1.211404
##              `bmi:regionnortheast`              `bmi:regionnorthwest`
##                           1.180304                           1.247453
##         `children:regionnorthwest`                      `age:sex:bmi`
##                           1.031926                           1.070039
##                   `age:sex:smoker`     `age:children:regionnortheast`
##                           1.030028                           1.246235
##     `age:children:regionnorthwest`       `sex:smoker:regionnorthwest`
##                           1.216269                           1.091622
##              `bmi:children:smoker`       `bmi:smoker:regionnortheast`
##                           1.058912                           1.388802
##       `bmi:smoker:regionsoutheast`  `children:smoker:regionnorthwest`
##                           1.646044                           1.074360
```

```
# Extract the residuals
resid <- residuals(model_poly_step3)

# Perform Shapiro-Wilk test for normality of residuals
shapiro.test(resid)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  resid
## W = 0.65926, p-value < 2.2e-16
```

```
# Breusch-Pagan Test to detect heteroscedasticity
bptest(model_poly_step3)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  model_poly_step3
## BP = 27.801, df = 24, p-value = 0.2685
```

All in all, the 3rd degree polynomial is a lot better than linear regression models but compared to the 2nd degree polynomial it does not show improvements for the extra amount of complexity that we added to this model.

4. Conclusion

In conclusion, our analysis aimed to investigate the accuracy of predicting insurance costs based on variables available in the medical cost data set. We explored linear regression, backward stepwise regression, and polynomial regression models of degree 2 and 3, using cross-validation to evaluate the model performance.

Our findings showed that the polynomial regression model with degree 2 was the best model for predicting insurance costs, with a mean train RMSE of 5115 and a mean test RMSE of 5138, indicating that the model generalized well to unseen data. The model also had a high R-squared value of 0.8420, indicating that 84.20% of the variability in the target variable is explained by the independent variables.

Overall, our findings suggest that it is possible to predict insurance costs based on variables available in the medical cost data set with a reasonable level of accuracy, and that polynomial regression models can significantly improve predictive accuracy over linear regression models. However, it is important to carefully select the degree of the polynomial and use cross-validation techniques to evaluate model performance, as overfitting can easily occur if the degree of the polynomial is too high.

Two potential future research directions to enhance the results could be to explore the impact of including additional variables in the model, such as lifestyle factors or other medical conditions. Additionally, incorporating non-linear relationships between the independent and dependent variables, such as using a neural network model, could also improve the accuracy of insurance cost predictions.

5. Bibliography

Choi, M. (2018). Insurance. Retrieved from https://www.kaggle.com/mirichoi0218/insurance

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An Introduction to Statistical Learning: with Applications in R. Springer.

O'Brien, R. M. (2007). A Caution Regarding Rules of Thumb for Variance Inflation Factors. Quality & Quantity, 41(5), 673-690. doi: 10.1007/s11135-006-9018-6

Tsai, C. F., Lin, Y. H., Chang, H. Y., & Chen, W. L. (2018). A comparative study of categorical data encoding techniques for neural network classifiers. Applied Sciences, 8(7), 1134. https://doi.org/10.3390/app8071134

Tukey, J.W. (1977). Exploratory Data Analysis. Addison-Wesley.

Zhang, L., Wang, H., & Wang, L. (2015). Discrimination-Aware Regression Modeling for Insurance Premium Pricing. In Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 2135-2144). ACM.

Philipp Habicht