MH3510 Regression Analysis

Pu Fanyi* Zhang Kaichen† Shan Yi‡ Fu Yilin§ Mu Yichen¶

2024-11-11

Report / Code (html) / Code (pdf) / Github Repo

1 Setup and Load Data

```
library(ggplot2)
library(GGally)
library(tidyr)
library(reshape2)
library(latex2exp)
library(gridExtra)
library(ggside)
library(RColorBrewer)


draw <- function(drawing_function, file_path, width = 8, height = 8) {
  # Check if the directory exists
  dir_path <- dirname(file_path)
  if (!dir.exists(dir_path)) {
    stop("Directory does not exist: ", dir_path)
  }

  # Try to create the PDF
  tryCatch(
    {
      pdf(file_path, width = width, height = height)
      drawing_function()
      dev.off()
    },
    error = function(e) {
      message("Error creating PDF: ", e$message)
    }
  )

  # Display the plot
```

*FPU001@e.ntu.edu.sg
†ZHAN0564@e.ntu.edu.sg
‡SH0005YI@e.ntu.edu.sg
§FUYI0005@e.ntu.edu.sg
¶M220100@e.ntu.edu.sg

```
  drawing_function()
}
```

```
columns <- c("Y", "X1", "X2", "X3", "X4", "X5", "X6", "X7")
```

```
file <- "../assets/aadt.txt"
data_raw <- read.table(file, col.names = columns)
data_ori <- data_raw[, c("Y", "X1", "X2", "X3", "X4")]
data <- data_ori
head(data)
```

```
##      Y    X1 X2 X3 X4
## 1 1616 13404  2 52  2
## 2 1329 52314  2 60  2
## 3 3933 30982  2 57  2
## 4 3786 25207  2 64  2
## 5  465 20594  2 40  2
## 6  794 11507  2 44  2
```

```
image_folder <- "../assets/images/"
```

2 Overview of the data

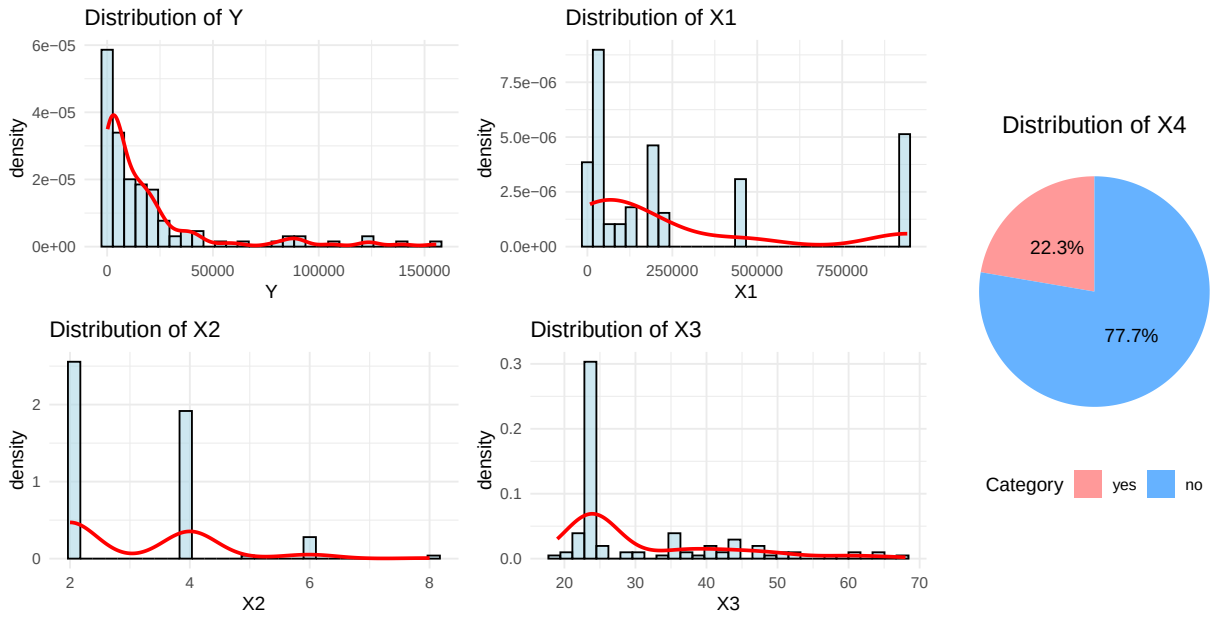
```
overview_image_path <- file.path(image_folder, "overview")
dir.create(overview_image_path, showWarnings = FALSE)
```

```
# Create separate plots for each variable
plot_Y <- ggplot(data, aes(x = Y)) +
  geom_histogram(aes(y = after_stat(density)),
    fill = "lightblue",
    color = "black", bins = 30, alpha = 0.6
  ) +
  geom_density(color = "red", linewidth = 1) +
  labs(title = "Distribution of Y") +
  theme_minimal()

plot_X1 <- ggplot(data, aes(x = X1)) +
  geom_histogram(aes(y = after_stat(density)),
    fill = "lightblue",
    color = "black", bins = 30, alpha = 0.6
  ) +
  geom_density(color = "red", linewidth = 1) +
  labs(title = "Distribution of X1") +
  theme_minimal()

plot_X2 <- ggplot(data, aes(x = X2)) +
  geom_histogram(aes(y = after_stat(density)),
    fill = "lightblue",
    color = "black", bins = 30, alpha = 0.6
```

```
  ) +
  geom_density(color = "red", linewidth = 1) +
  labs(title = "Distribution of X2") +
  theme_minimal()

plot_X3 <- ggplot(data, aes(x = X3)) +
  geom_histogram(aes(y = after_stat(density)),
    fill = "lightblue",
    color = "black", bins = 30, alpha = 0.6
  ) +
  geom_density(color = "red", linewidth = 1) +
  labs(title = "Distribution of X3") +
  theme_minimal()

# Arrange histograms in 2x2 grid
histogram_grid <- arrangeGrob(
  plot_Y, plot_X1, plot_X2, plot_X3,
  ncol = 2,
  nrow = 2
)
```

```
# Create pie chart for X4
x4_counts <- table(data$X4)
plot_X4 <- ggplot(data.frame(x4_counts), aes(x = "", y = Freq, fill = Var1)) +
  geom_bar(stat = "identity", width = 1) +
  coord_polar("y", start = 0) +
  labs(
    title = "Distribution of X4",
    fill = "Category"
  ) +
  scale_fill_manual(
    values = c("#FF9999", "#66B2FF"),
    labels = c("yes", "no")
  ) +
  theme_minimal() +
  theme(
    plot.title = element_text(hjust = 0.5, size = 14),
    legend.position = "bottom",
    legend.title = element_text(),
    panel.grid = element_blank(),
    axis.text = element_blank(),
    axis.title = element_blank(),
  ) +
  geom_text(
    aes(
      label = paste0(round(100 * Freq / sum(Freq), 1), "%")
    ),
    position = position_stack(vjust = 0.5)
  )

# Combine histogram grid and pie chart vertically
combined_plot <- grid.arrange(
  histogram_grid,
```

```
  plot_X4,
  ncol = 2,
  widths = c(3, 1)
)
```




```
combined_plot_image_path <- file.path(
  overview_image_path,
  "combined_histograms.pdf"
)
ggsave(combined_plot_image_path, combined_plot, width = 10, height = 5)
```

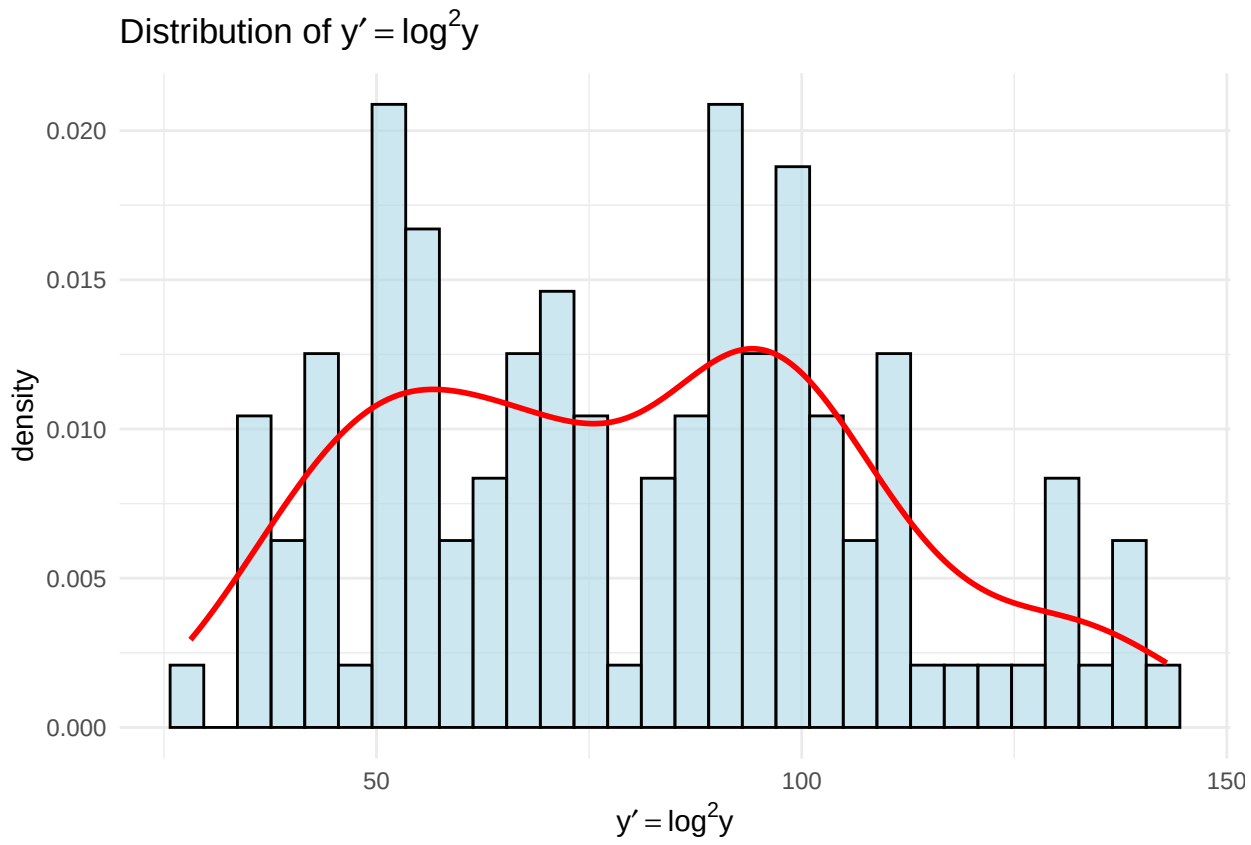
3 Graphic display of the observed data

```
pdf(file.path(image_folder, "overview.pdf"), width = 10, height = 10)
ggpairs(data,
  upper = list(continuous = wrap("points",
    color = "blue",
    alpha = 0.5, size = 2
  )),
  lower = list(continuous = wrap("points",
    color = "red",
    alpha = 0.5, size = 2
  )),
  diag = list(continuous = wrap("densityDiag", fill = "lightblue"))
)
dev.off()
```

```
## pdf
##   2
```

We let $y' = \log y$ to fix the skewness of the data.

```
y_prime <- log(data$Y)^2
y_prime_fig <- ggplot(data.frame(y_prime), aes(x = y_prime)) +
  geom_histogram(aes(y = after_stat(density)),
    fill = "lightblue",
    color = "black", bins = 30, alpha = 0.6
  ) +
  geom_density(color = "red", linewidth = 1) +
  labs(
    title = TeX("Distribution of $y' = \\log^2y$"),
    x = TeX("$y' = \\log^2y$")
  ) +
  theme_minimal()
y_prime_fig
```

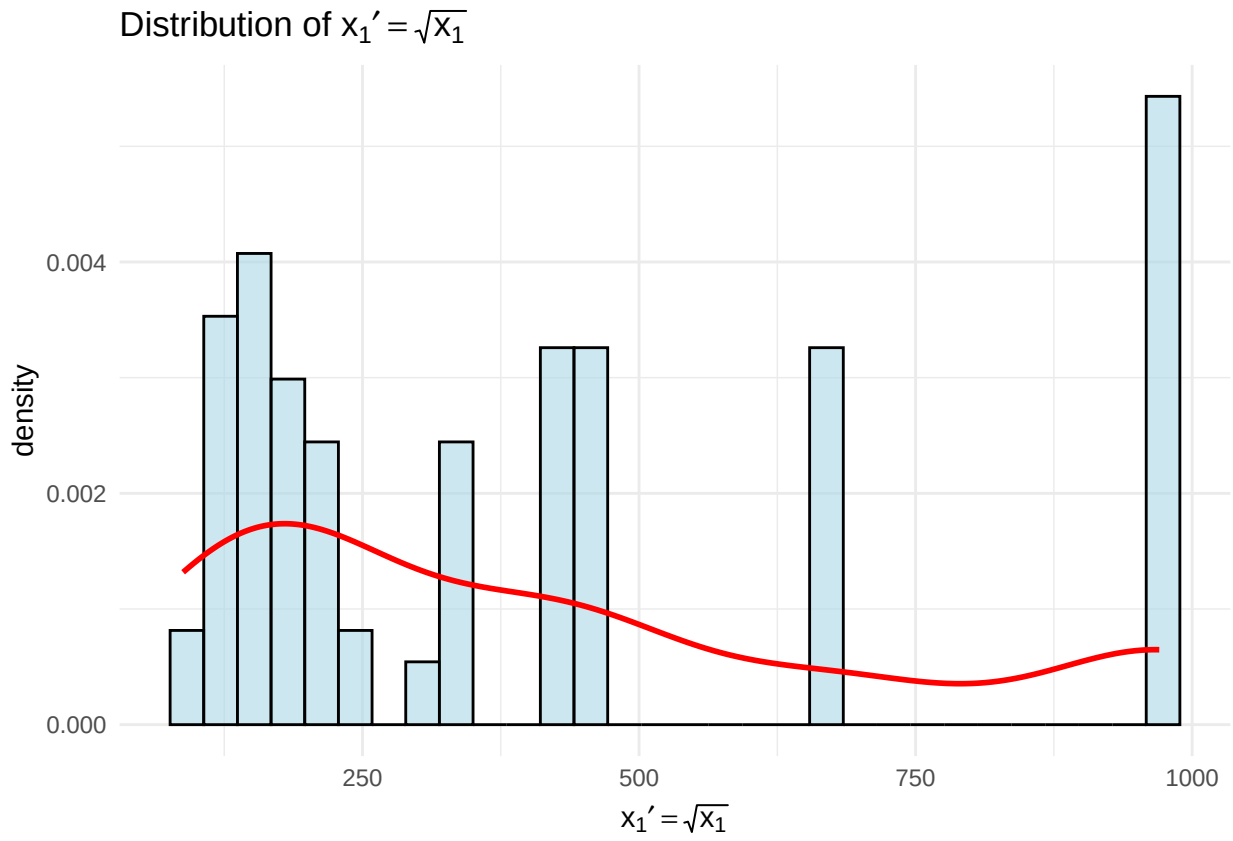


Finally, we replace the original y with y'.

```
data$Y <- y_prime
```

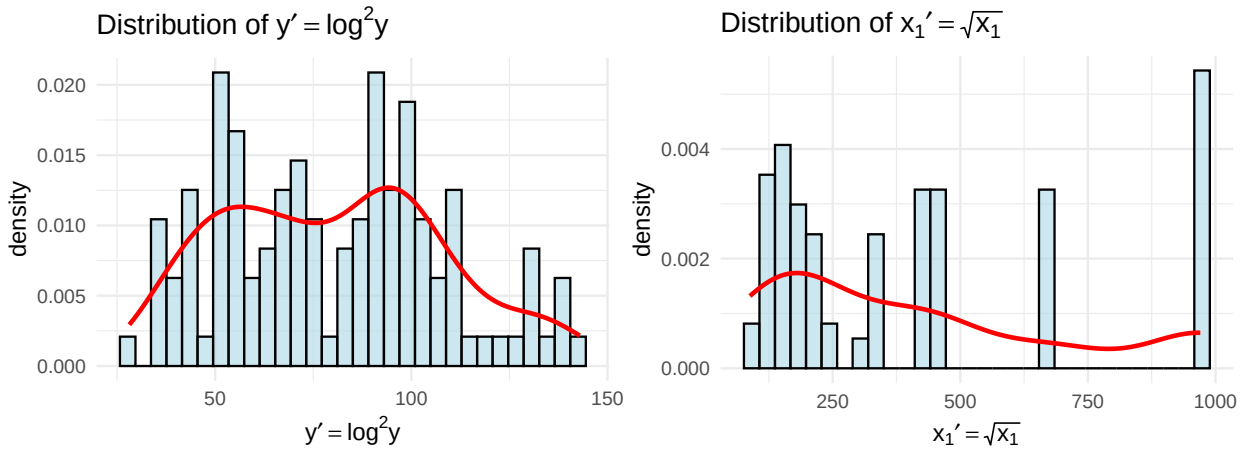
```
x1_prime <- sqrt(data$X1)
x1_prime_fig <- ggplot(data.frame(x1_prime), aes(x = x1_prime)) +
  geom_histogram(aes(y = after_stat(density)),
    fill = "lightblue",
    color = "black", bins = 30, alpha = 0.6
```

```
  ) +
  geom_density(color = "red", linewidth = 1) +
  labs(
    title = TeX("Distribution of $x_1' = \\sqrt{x_1}$"),
    x = TeX("$x_1'=\\sqrt{x_1}$")
  ) +
  theme_minimal()
x1_prime_fig
```




```
data$X1 <- x1_prime
```

```
# merge graphs
merged_graphs <- grid.arrange(
  y_prime_fig,
  x1_prime_fig,
  ncol = 2,
  widths = c(1, 1)
)
```




```
ggsave(file.path(overview_image_path, "transformed_variables.pdf"),
  merged_graphs,
  width = 8, height = 3
)
```

The distribution of y' and x_1', it can be seen that the distributions of y' and x_1' are relatively uniform compared with the original y and x_1.

Although x_2 and x_3 also have some skewness, considering that they are integers with a small range in the dataset, applying a transformation may not be useful. We have decided not to transform them for now.

4 Single Variable Analysis

```
slr_output_folder <- file.path(image_folder, "slr")
dir.create(slr_output_folder, showWarnings = FALSE)
```

4.1 X1

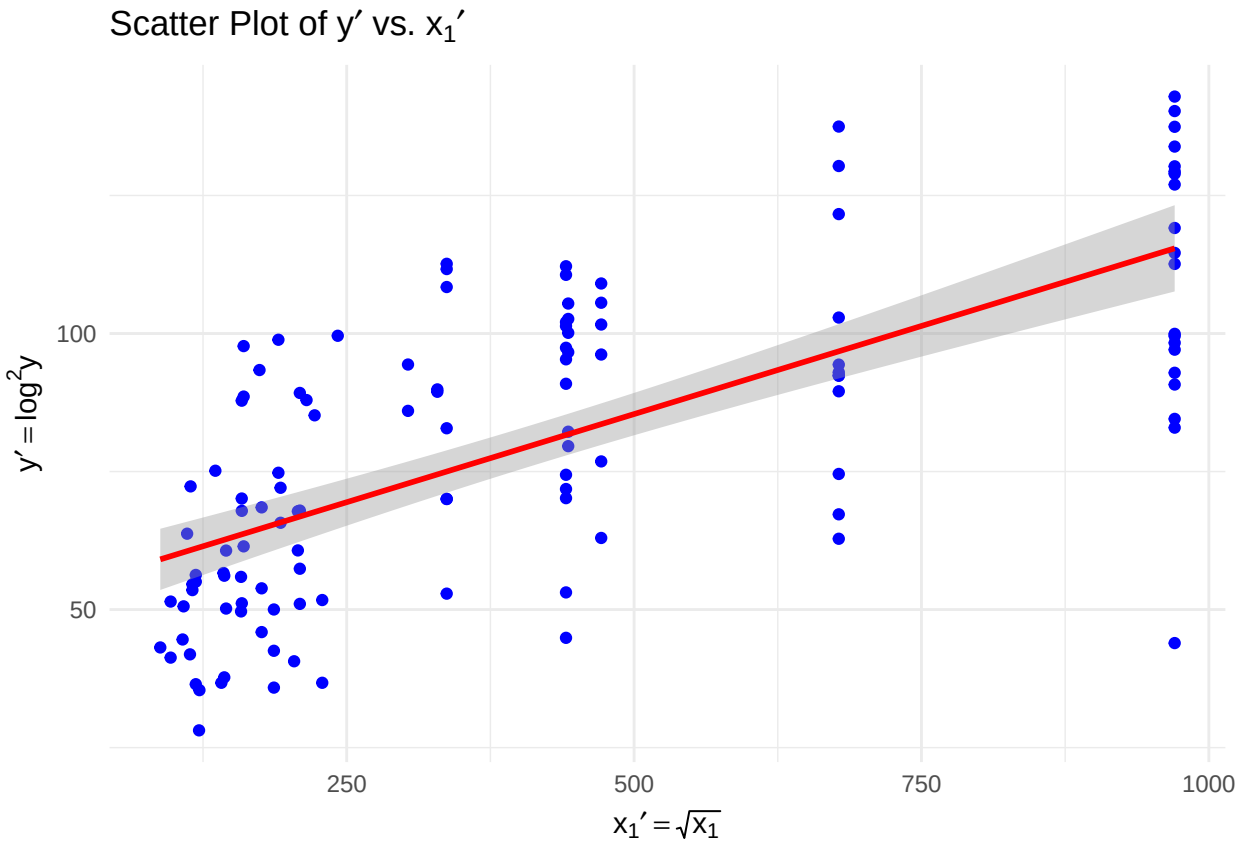
```
X1_output_folder <- file.path(slr_output_folder, "X1")
dir.create(X1_output_folder, showWarnings = FALSE)
```

```
# scatter plot
scatter_plot_path <- file.path(X1_output_folder, "scatter_plot.pdf")
scatter_plot_X1 <- ggplot(data, aes(x = x1_prime, y = y_prime)) +
  geom_point(color = "blue") +
  geom_smooth(method = "lm", color = "red") +
  labs(
    title = TeX("Scatter Plot of $y'$ vs. $x_1'$"),
    x = TeX("$x_1'=\\sqrt{x_1}$"), y = TeX("$y'=\\log^2 y$")
  ) +
  theme_minimal()
ggsave(scatter_plot_path, scatter_plot_X1, width = 8, height = 6)
```

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

```
scatter_plot_X1
```

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



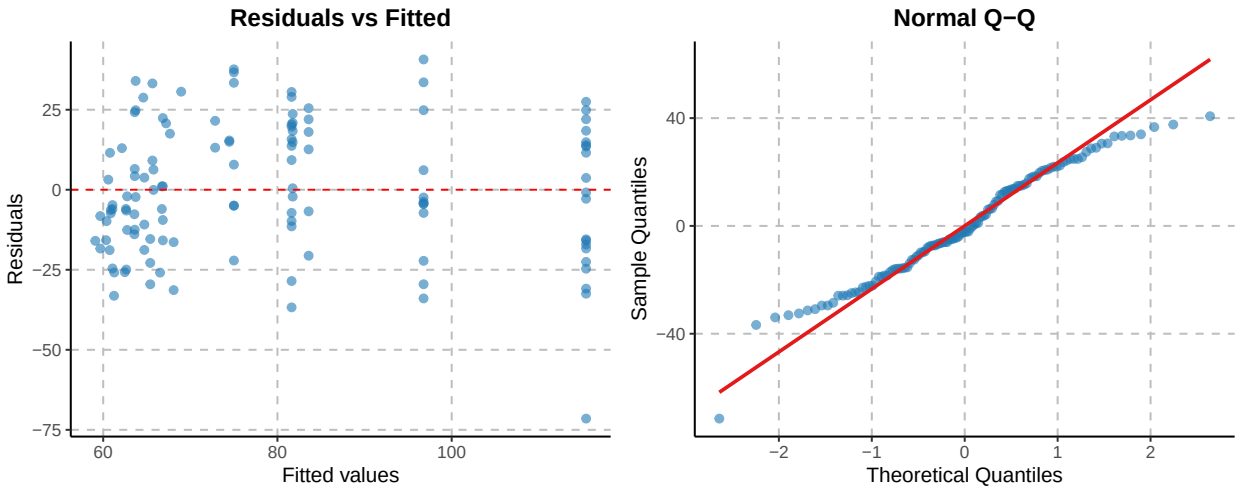

```
slr_X1 <- lm(y_prime ~ x1_prime, data = data)
summary(slr_X1)
```

```
##
## Call:
## lm(formula = y_prime ~ x1_prime, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -71.497 -15.752  -2.273  15.791  40.708
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 53.487764   3.232405   16.55   <2e-16 ***
## x1_prime     0.063839   0.006298   10.14   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 20.57 on 119 degrees of freedom
## Multiple R-squared:  0.4634, Adjusted R-squared:  0.4588
## F-statistic: 102.7 on 1 and 119 DF,  p-value: < 2.2e-16
```

We draw the residuals plot and QQ-plot to provide further insights into the model's adequacy.

```
# Create the main plot with the residuals
plot_resid_X1 <- ggplot(
  data = data.frame(fitted = fitted(slr_X1), residuals = resid(slr_X1)),
  aes(x = fitted, y = residuals)
) +
  # Add points on top
  geom_point(color = "#1f78b4", alpha = 0.6, size = 2) +
  # Original elements
  labs(
    x = "Fitted values",
    y = "Residuals",
    title = "Residuals vs Fitted"
  ) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  scale_fill_viridis_d(option = "C") + # Changed to discrete scale
  theme_classic(base_size = 12) +
  theme(
    plot.title = element_text(hjust = 0.5, face = "bold", size = 14),
    axis.title = element_text(size = 12),
    panel.grid.major = element_line(color = "gray", linetype = "dashed"),
  )

qqplot_X1 <- ggplot(
  data = data.frame(sample = resid(slr_X1)),
  aes(sample = sample)
) +
  stat_qq(color = "#1f78b4", alpha = 0.6, size = 2) +
  stat_qq_line(color = "#e31a1c", linewidth = 1) +
  labs(
    x = "Theoretical Quantiles",
    y = "Sample Quantiles",
    title = "Normal Q-Q"
  ) +
  theme_classic(base_size = 12) +
  theme(
    plot.title = element_text(hjust = 0.5, face = "bold", size = 14),
    axis.title = element_text(size = 12),
    panel.grid.major = element_line(color = "gray", linetype = "dashed")
  )

x1_fig <- grid.arrange(plot_resid_X1, qqplot_X1, ncol = 2)
```




```
ggsave(file.path(X1_output_folder, "residuals_vs_fitted_qqplot.pdf"),
  x1_fig,
  width = 10, height = 4
)
```

The residual plot in Figure 5 shows a generally even distribution of residuals around zero, though there are some signs of non-constant variance. This pattern might indicate that variability in traffic increases with larger predicted values, potentially due to factors such as urban infrastructure or public transport usage that could influence traffic in larger counties.

The QQ-plot of residuals shows that the residuals mostly follow a normal distribution, with minor deviations at the tails, suggesting that while the model performs well overall, there might be outliers or specific population segments where the prediction is less accurate.

```
aov <- anova(slr_X1)
aov
```

```
## Analysis of Variance Table
##
## Response: y_prime
##            Df Sum Sq Mean Sq F value    Pr(>F)
## x1_prime    1  43471   43471  102.75 < 2.2e-16 ***
## Residuals 119  50347     423
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Since the F value is in an F distribution, we conduct an F-test and obtain a p-value of less than 2×10^{-16}. The p-value is small enough for us to reject H_0 and conclude that the population actually has an influence on AADT.

The R^2 statistic is calculated as approximately 0.4634. This means that while there is some relationship between population and AADT, there are other factors that determine the AADT.

4.2 X2

The x-axis representing the number of lanes. The y-axis representing the AADT data after transformation.

```r
X2_output_folder <- file.path(slr_output_folder, "X2")
dir.create(X2_output_folder, showWarnings = FALSE)
```

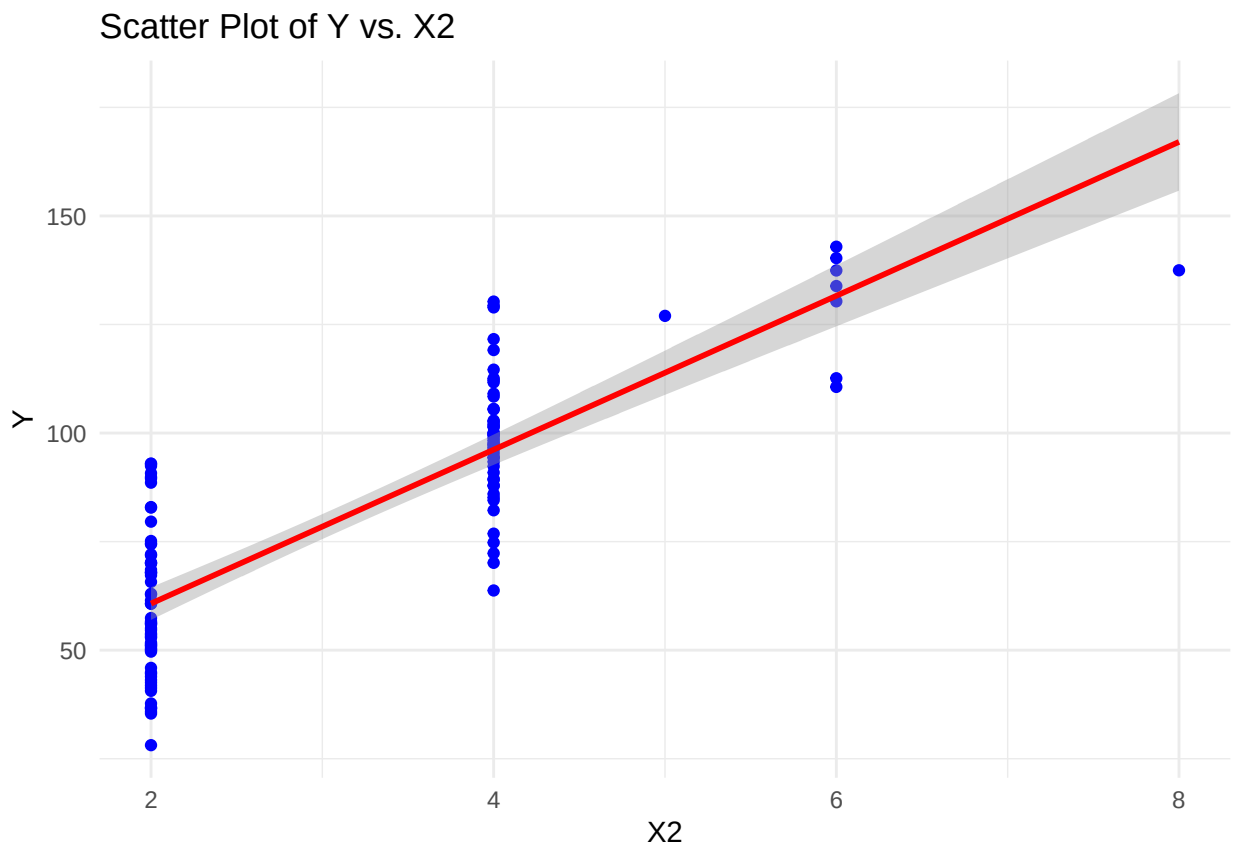
In this plot, we also include a trendline to indicate the general pattern in the data. A positive slope in the trendline has been ovserved in the plot, which means that as the number of lanes increases, AADT tends to increase as well. This aligns with common sense: adding lanes generally allows a road to support more vehicles, which increases its traffic capacity.

```r
# scatter plot
scatter_plot_path <- file.path(X2_output_folder, "scatter_plot.pdf")
scatter_plot_X2 <- ggplot(data, aes(x = X2, y = Y)) +
  geom_point(color = "blue") +
  geom_smooth(method = "lm", color = "red") +
  labs(title = "Scatter Plot of Y vs. X2", x = "X2", y = "Y") +
  theme_minimal()
ggsave(scatter_plot_path, scatter_plot_X2, width = 8, height = 6)
```

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

```r
scatter_plot_X2
```

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



```r
slr_X2 <- lm(Y ~ X2, data = data)
summary(slr_X2)
```

```
##
## Call:
## lm(formula = Y ~ X2, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -32.614 -10.185   0.024   9.303  34.108
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   25.304      3.754    6.74    6e-10 ***
## X2            17.718      1.118   15.85   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 15.92 on 119 degrees of freedom
## Multiple R-squared:  0.6786, Adjusted R-squared:  0.6759
## F-statistic: 251.3 on 1 and 119 DF,  p-value: < 2.2e-16
```

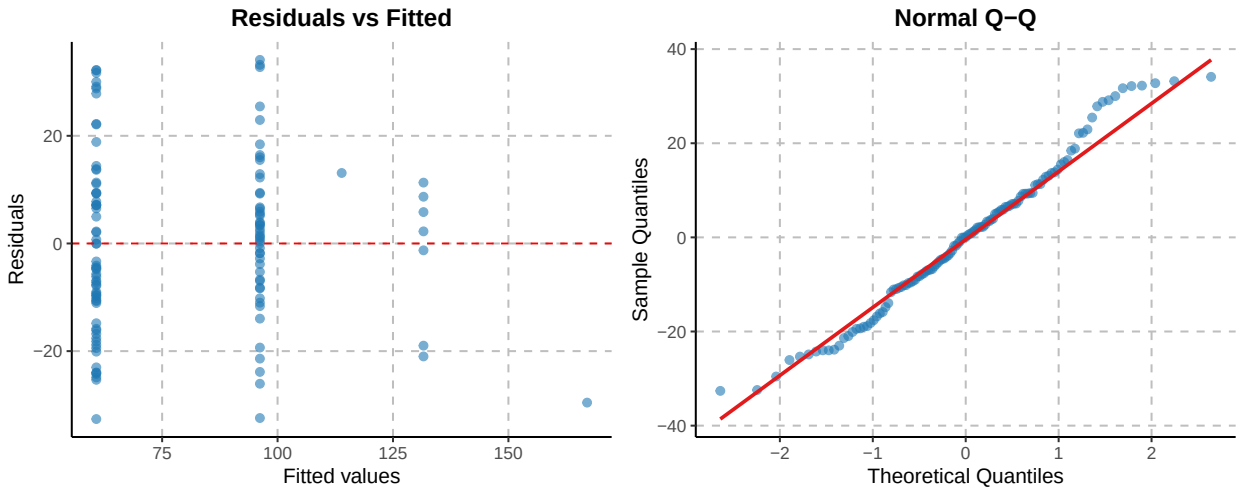
Intercept
When the number of lanes is zero, the model predicts y' of approximately 25.3. While this might not be meaningful practically (since we rarely see roads with zero lanes), it serves as a baseline.

Slope for x_2
For each additional lane, the model predicts an increase of about 17.7 in y'. This means adding lanes has a significant positive effect on traffic capacity.

Significance
The very low p-value ($< 2 \times 10^{-16}$) for the number of lanes indicates that this relationship is statistically significant, implying that the number of lanes is an important factor in determining AADT.

```r
# Create the main plot with the residuals
plot_resid_X2 <- ggplot(
  data = data.frame(fitted = fitted(slr_X2), residuals = resid(slr_X2)),
  aes(x = fitted, y = residuals)
) +
  # Add points on top
  geom_point(color = "#1f78b4", alpha = 0.6, size = 2) +
  # Original elements
  labs(
    x = "Fitted values",
    y = "Residuals",
    title = "Residuals vs Fitted"
  ) +
  # geom_smooth(method = "loess", color = "red", size = 1) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  scale_fill_viridis_d(option = "C") + # Changed to discrete scale
  theme_classic(base_size = 12) +
  theme(
    plot.title = element_text(hjust = 0.5, face = "bold", size = 14),
    axis.title = element_text(size = 12),
```

```
    panel.grid.major = element_line(color = "gray", linetype = "dashed"),
  )

qqplot_X2 <- ggplot(
  data = data.frame(sample = resid(slr_X2)),
  aes(sample = sample)
) +
  stat_qq(color = "#1f78b4", alpha = 0.6, size = 2) +
  stat_qq_line(color = "#e31a1c", size = 1) +
  labs(
    x = "Theoretical Quantiles",
    y = "Sample Quantiles",
    title = "Normal Q-Q"
  ) +
  theme_classic(base_size = 12) +
  theme(
    plot.title = element_text(hjust = 0.5, face = "bold", size = 14),
    axis.title = element_text(size = 12),
    panel.grid.major = element_line(color = "gray", linetype = "dashed")
  )

x2_fig <- grid.arrange(plot_resid_X2, qqplot_X2, ncol = 2)
```




```
ggsave(file.path(X2_output_folder, "residuals_vs_fitted_qqplot.pdf"),
  x2_fig,
  width = 10, height = 4
)
```

Residuals vs Fitted Plot The graph above examines how well the linear model predicts AADT based on the number of lanes. If the residuals (the differences between actual and predicted AADT) are scattered randomly around the zero line, it indicates that the number of lanes effectively explains variations in AADT. In practical terms, this would mean that road sections with different lane counts generally show predictable changes in traffic volume, and our model captures this relationship well. However, if a pattern appears in the residuals, such as a consistent underestimation or overestimation of AADT for certain lane counts, it might suggest that factors beyond lane count (e.g., location, road type) are also influencing traffic volume, indicating that our model may need additional variables to improve accuracy.

Normal QQ-Ploy it assesses whether the residuals are normally distributed, which is an assumption for linear regression. A normal distribution of residuals suggests that the relationship between lane count and AADT is generally well-captured by the linear model. If the points fall along a straight line in the Q-Q plot, it indicates that the model's errors are random and unbiased, meaning our predictions are reasonably reliable across different lane counts. However, significant deviations from this line might indicate that the relationship between lane count and AADT isn't fully linear or that other factors are affecting traffic volume in ways the model doesn't capture, possibly warranting a more complex or adjusted model.

```
aov <- anova(slr_X2)
aov
```

```
## Analysis of Variance Table
##
## Response: Y
##            Df Sum Sq Mean Sq F value    Pr(>F)
## X2          1  63665   63665  251.25 < 2.2e-16 ***
## Residuals 119  30153     253
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

According to Equation 5, we can build $H_0 : \beta_1 = 0$ and $H_1 : \beta_1 \neq 0$. We conduct an F-test by calculating the p value.

The p value is small enough for us to reject H_0 and conclude that the population actually has an influence on AADT.

The R^2 statistic is calculated as approximately 0.4634.

This means that while there is some relationship between population and AADT, there are other factors that determine AADT.

Summary

This analysis shows that the number of lanes has a strong, positive effect on AADT. This makes intuitive sense, as wider roads with more lanes are better suited to handle larger volumes of traffic. The linear model and diagnostic plots suggest that the relationship is well captured by our model, with residuals generally behaving as expected. In summary, as we add more lanes to a road, we can expect an increase in daily traffic capacity, which matches our common-sense understanding of road infrastructure and traffic flow.

4.3 X3

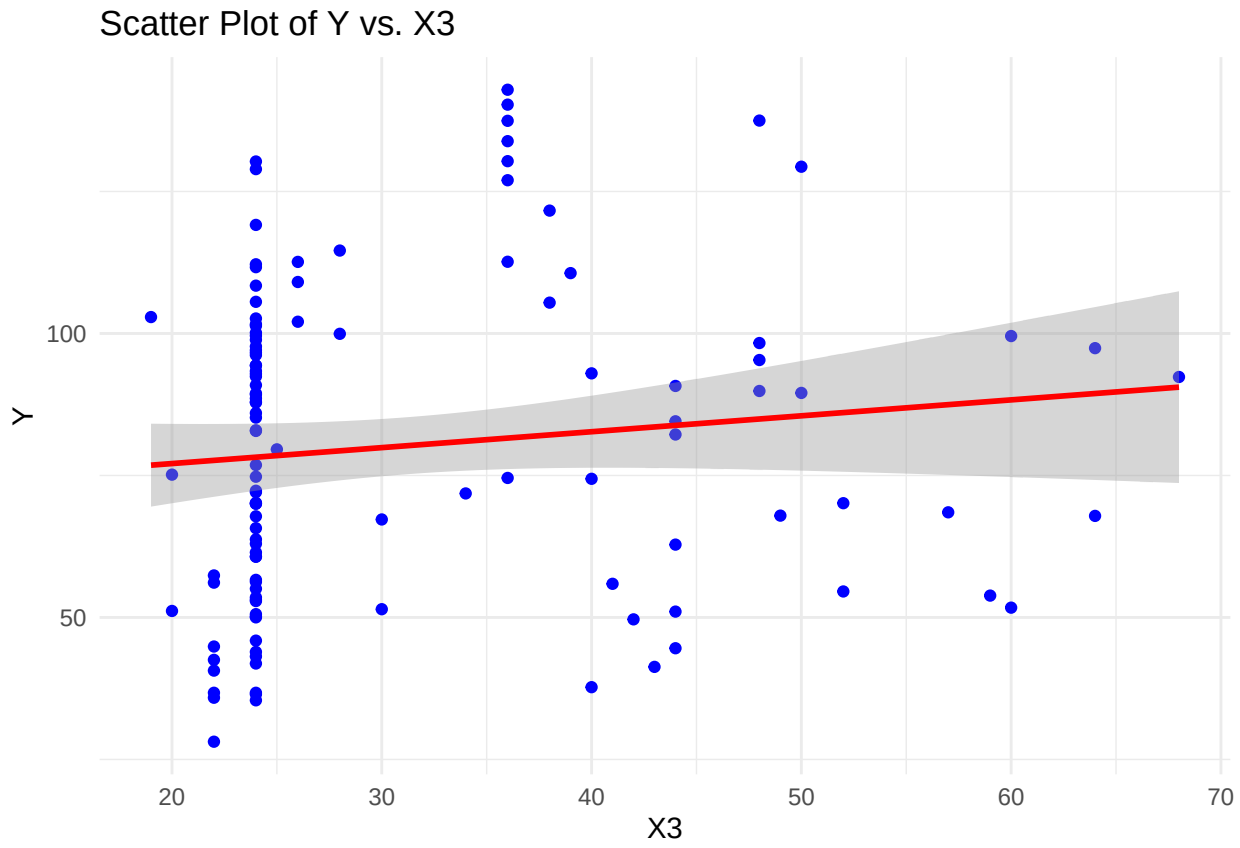
```
X3_output_folder <- file.path(slr_output_folder, "X3")
dir.create(X3_output_folder, showWarnings = FALSE)
```

```
# scatter plot
scatter_plot_path <- file.path(X3_output_folder, "scatter_plot.pdf")
scatter_plot_X3 <- ggplot(data, aes(x = X3, y = Y)) +
  geom_point(color = "blue") +
  geom_smooth(method = "lm", color = "red") +
  labs(title = "Scatter Plot of Y vs. X3", x = "X3", y = "Y") +
  theme_minimal()
ggsave(scatter_plot_path, scatter_plot_X3, width = 8, height = 6)
```

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

```
scatter_plot_X3
```

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



The scatter plot shows the relationship between Annual Average Daily Traffic (AADT) and the width of road sections (in feet). Each point represents a specific road section, with: The x-axis represents the width of the road section in feet x3. The y-axis represents the AADT.

In this plot, a trendline has been added to indicate the general pattern in the data. Although the trendline suggests a positive slope, indicating that as the road width increases, AADT tends to increase, this relationship is not statistically significant, as shown by the linear model.

```
slr_X3 <- lm(Y ~ X3, data = data)
summary(slr_X3)
```

```
##
## Call:
## lm(formula = Y ~ X3, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -49.532 -23.161   0.716  20.602  61.334
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)  71.4902     7.3225   9.763   <2e-16 ***
## X3            0.2803     0.2207   1.270    0.207
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 27.89 on 119 degrees of freedom
## Multiple R-squared:  0.01337,    Adjusted R-squared:  0.00508
## F-statistic: 1.613 on 1 and 119 DF,  p-value: 0.2066
```

Intercept: 71.49, which suggests that at a hypothetical width of zero, the model would predict an AADT of approximately 71.5, though this does not have practical meaning.

Slope for x3: 0.28, indicating a predicted increase in AADT of approximately 0.28 for each additional foot in road width. However, this effect is not statistically significant (p-value = 0.207), suggesting that road width does not significantly impact AADT in this model.

```
aov <- anova(slr_X3)
aov
```

```
## Analysis of Variance Table
##
## Response: Y
##            Df Sum Sq Mean Sq F value Pr(>F)
## X3          1   1254 1254.46  1.6127 0.2066
## Residuals 119  92564  777.85
```

R^2 Statistic
The R^2 statistic is approximately 0.013, implying that road width explains only about 1.3% of the variation in AADT, suggesting that other factors are more influential.

Summary

This analysis suggests that the width of a road section has a weak and statistically insignificant relationship with AADT. While wider roads generally allow for higher traffic volumes, this model does not capture that effect effectively.

4.4 X4

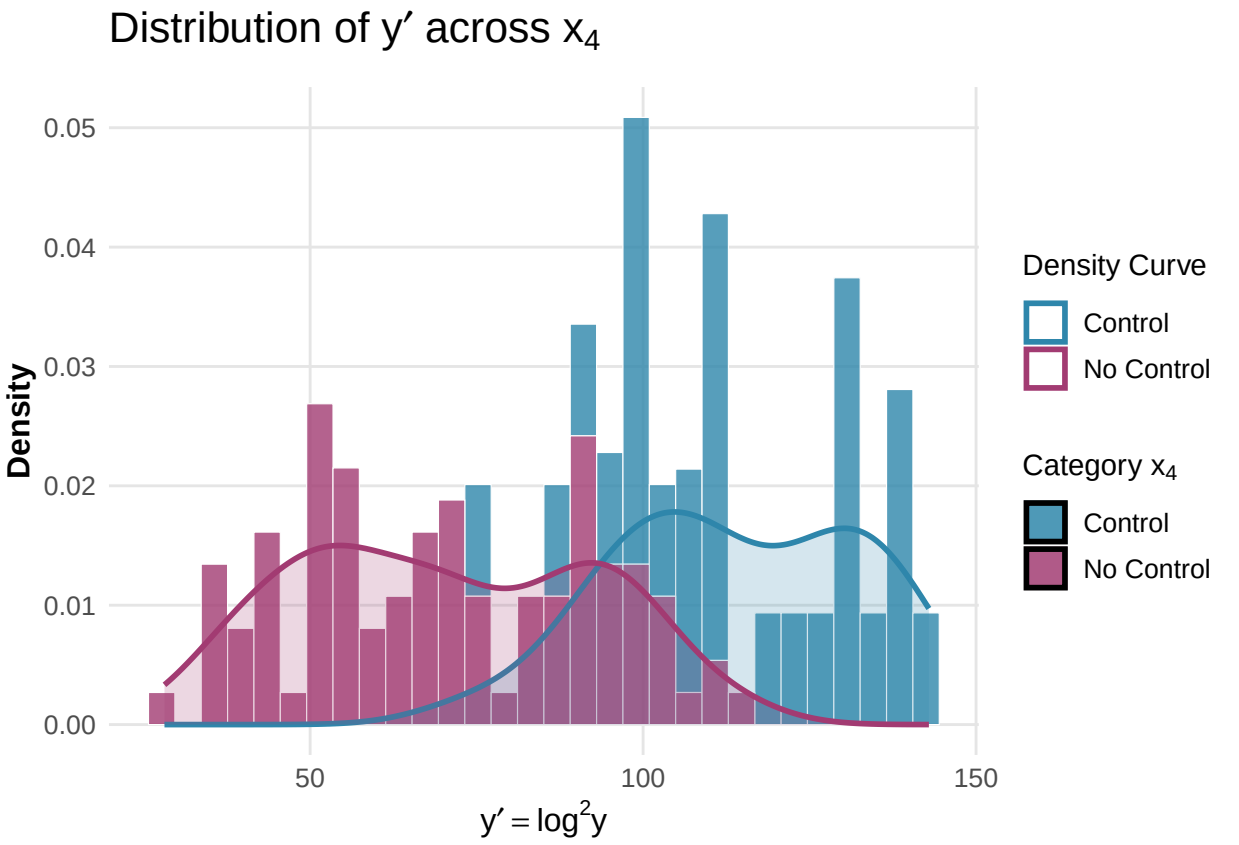
Controlling access to the road section is one of the important methods for improving traffic flow. In this chapter, we explore whether this approach can effectively increase the annual average daily traffic. We divide the data into two categories: the first category consists of data where control measures are implemented, and the second category consists of data without control measures. The distribution of these 2 categories

```
X4_output_folder <- file.path(slr_output_folder, "X4")
dir.create(X4_output_folder, showWarnings = FALSE)
```

```
# split data for X4 = 0 and X4 = 1
y_control <- data[data$X4 == 1, "Y"]
y_no_control <- data[data$X4 == 2, "Y"]
```

```
custom_colors <- c("#2E86AB", "#A23B72")

histogram_path <- file.path(X4_output_folder, "histogram.pdf")
```

```r
histogram_X4 <- ggplot(data, aes(x = Y, fill = factor(X4))) +
  geom_histogram(
    aes(y = after_stat(density)),
    bins = 30,
    alpha = 0.8,
    color = "white",
    size = 0.2
  ) +
  geom_density(
    aes(color = factor(X4)),
    alpha = 0.2,
    size = 1
  ) +
  labs(
    title = TeX("Distribution of $y'$ across $x_4$"),
    x = TeX("$y' = \\log^2 y$"),
    y = "Density",
    fill = TeX("Category $x_4$"),
    color = TeX("Density Curve")
  ) +
  theme_minimal() +
  theme(
    plot.title = element_text(
      size = 16, face = "bold",
      margin = margin(b = 10)
    ),
    plot.subtitle = element_text(
      size = 12, color = "gray50",
      margin = margin(b = 20)
    ),
    axis.title = element_text(size = 12, face = "bold"),
    axis.text = element_text(size = 10),
    legend.title = element_text(size = 11, face = "bold"),
    legend.text = element_text(size = 10),
    legend.position = "right",
    panel.grid.minor = element_blank(),
    panel.grid.major = element_line(color = "gray90"),
    plot.background = element_rect(fill = "white", color = NA),
    panel.background = element_rect(fill = "white", color = NA)
  ) +
  scale_fill_manual(values = custom_colors, labels = c("Control", "No Control")) +
  scale_color_manual(values = custom_colors, labels = c("Control", "No Control"))


ggsave(
  histogram_path,
  histogram_X4,
  width = 8,
  height = 4,
)

histogram_X4
```



We can build the model

$$y'_{ij} = \theta_i + \epsilon_{ij}, \quad i \in \{1, 2\}, j \in \{1, \cdots, n_i\}, \quad \epsilon_{ij} \sim \mathcal{N}(0, \sigma^2)$$

```
theta_1 <- mean(y_control)
theta_2 <- mean(y_no_control)
print(paste("theta_1 = ", theta_1))
```

```
## [1] "theta_1 =  114.097822707391"
```

```
print(paste("theta_2 = ", theta_2))
```

```
## [1] "theta_2 =  70.4815506087082"
```

So

$$\begin{cases} \hat{\theta}_1 = \overline{y_{1\cdot}} \approx 114.10 \\ \hat{\theta}_2 = \overline{y_{2\cdot}} \approx 70.48 \end{cases}$$

We build ANOVA table

```r
group_y <- c(y_control, y_no_control)
group <- factor(rep(
  c("control", "no control"),
  c(length(y_control), length(y_no_control))
))
anova_table <- aov(group_y ~ group)
summary(anova_table)
```

```
##              Df Sum Sq Mean Sq F value   Pr(>F)
## group         1  39903   39903   88.07 5.37e-16 ***
## Residuals   119  53916     453
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We calculate the p-value to be approximately 5.37×10^{1}. Since this value is very small, we conclude that there is a significant difference by controlling access to the road section.

5 Multiple Linear Regression

As x_4 is a classification variable, we create a dummy variable for it. So that $x_4 = 1$ means we select the first class (control) and $x_4 = 0$ means we select the second class.

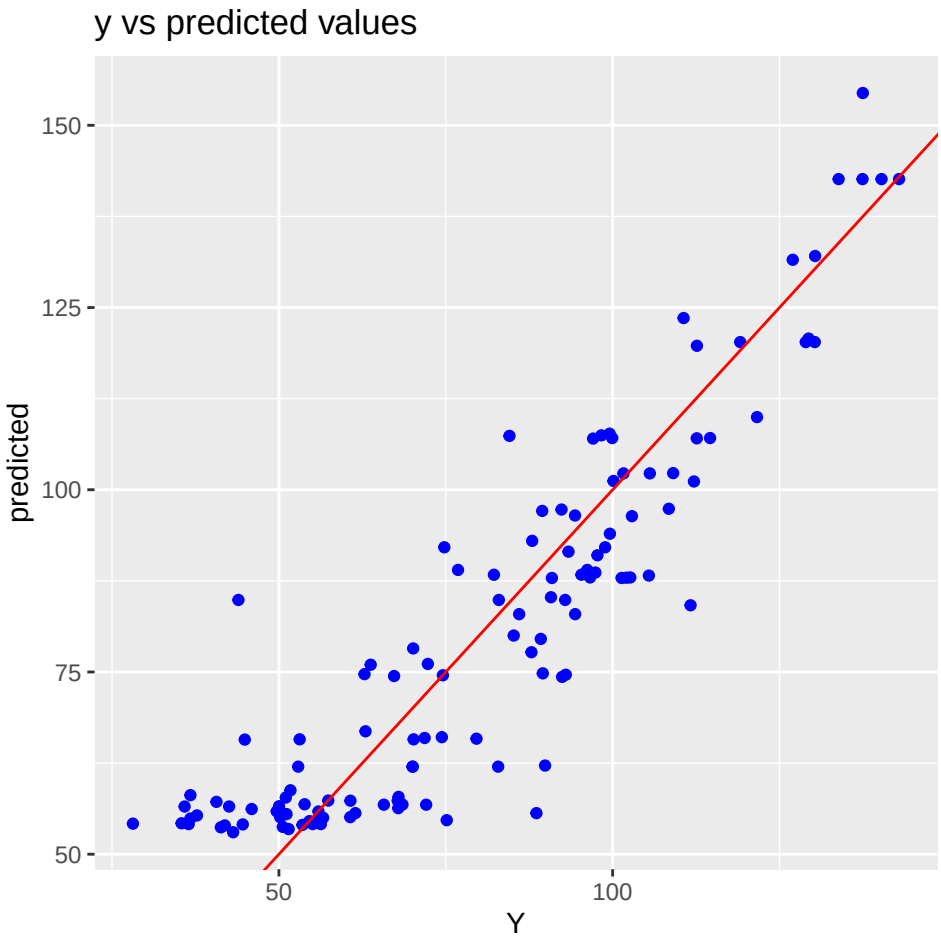
```r
### Full Model

mlr_output_folder <- file.path(image_folder, "mlr")
dir.create(mlr_output_folder, showWarnings = FALSE)
```

```r
mlr <- lm(Y ~ X1 + X2 + X3 + X4, data = data)
summary(mlr)
```

```
##
## Call:
## lm(formula = Y ~ X1 + X2 + X3 + X4, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -40.951  -7.647   0.037   8.356  32.947
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  53.739163   9.265768   5.800 5.85e-08 ***
## X1            0.036116   0.004247   8.503 7.38e-14 ***
## X2           11.069106   1.241316   8.917 8.11e-15 ***
## X3            0.018562   0.099140   0.187 0.851806
## X4          -13.241270   3.604210  -3.674 0.000363 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.2 on 116 degrees of freedom
## Multiple R-squared:  0.8161, Adjusted R-squared:  0.8098
## F-statistic: 128.7 on 4 and 116 DF,  p-value: < 2.2e-16
```

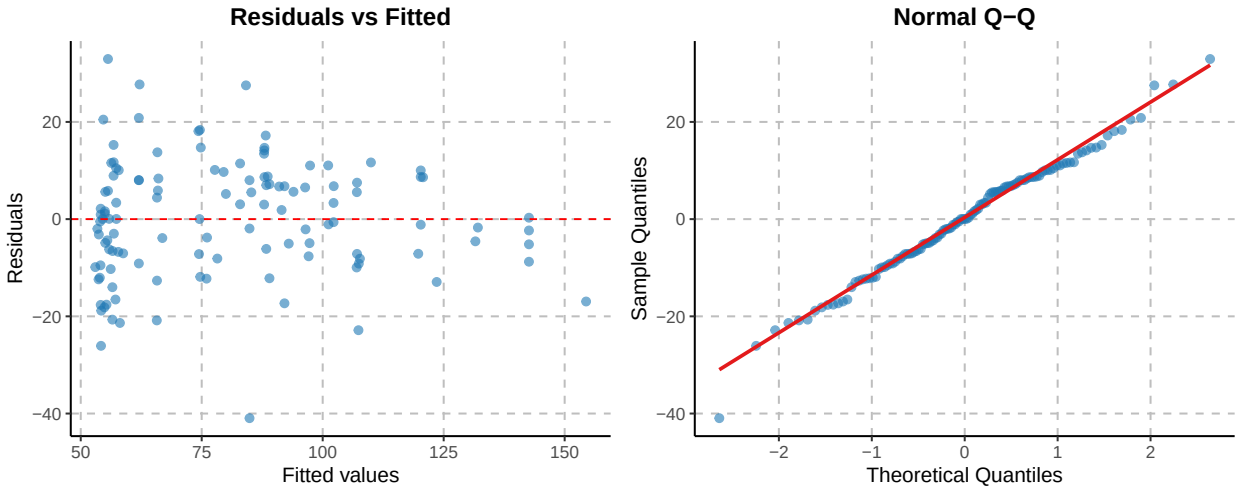
```r
predicted <- predict(mlr, data[, c("X1", "X2", "X3", "X4")])
```

```r
plot <- ggplot(data, aes(x = Y, y = predicted)) +
  geom_point(color = "blue") +
  labs(title = "y vs predicted values") +
  geom_abline(intercept = 0, slope = 1, color = "red")

file_path <- file.path(
  mlr_output_folder,
  "y_vs_predicted_values_full_model.pdf"
)
ggsave(file_path, plot, width = 5, height = 5)
plot
```




```r
plot_resid_mlr <- ggplot(
  data = data.frame(fitted = fitted(mlr), residuals = resid(mlr)),
  aes(x = fitted, y = residuals)
) +
  geom_point(color = "#1f78b4", alpha = 0.6, size = 2) +
  labs(
    x = "Fitted values",
    y = "Residuals",
```

```
    title = "Residuals vs Fitted"
  ) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  scale_fill_viridis_d(option = "C") +
  theme_classic(base_size = 12) +
  theme(
    plot.title = element_text(hjust = 0.5, face = "bold", size = 14),
    axis.title = element_text(size = 12),
    panel.grid.major = element_line(color = "gray", linetype = "dashed"),
  )

qqplot_mlr <- ggplot(
  data = data.frame(sample = resid(mlr)),
  aes(sample = sample)
) +
  stat_qq(color = "#1f78b4", alpha = 0.6, size = 2) +
  stat_qq_line(color = "#e31a1c", size = 1) +
  labs(
    x = "Theoretical Quantiles",
    y = "Sample Quantiles",
    title = "Normal Q-Q"
  ) +
  theme_classic(base_size = 12) +
  theme(
    plot.title = element_text(hjust = 0.5, face = "bold", size = 14),
    axis.title = element_text(size = 12),
    panel.grid.major = element_line(color = "gray", linetype = "dashed")
  )

mlr_res_fig <- grid.arrange(plot_resid_mlr, qqplot_mlr, ncol = 2)
```




```
ggsave(file.path(mlr_output_folder, "residuals_vs_fitted_qqplot.pdf"),
  mlr_res_fig,
  width = 10, height = 4
)
```

5.1 Adjusted Full Model

Figure shows the model predictions and residuals. We observe that the σ of the residuals depends on y', indicating heteroscedasticity. Therefore, we need to adjust the transformation to make σ constant.

```
data$Y <- data$Y^2
```

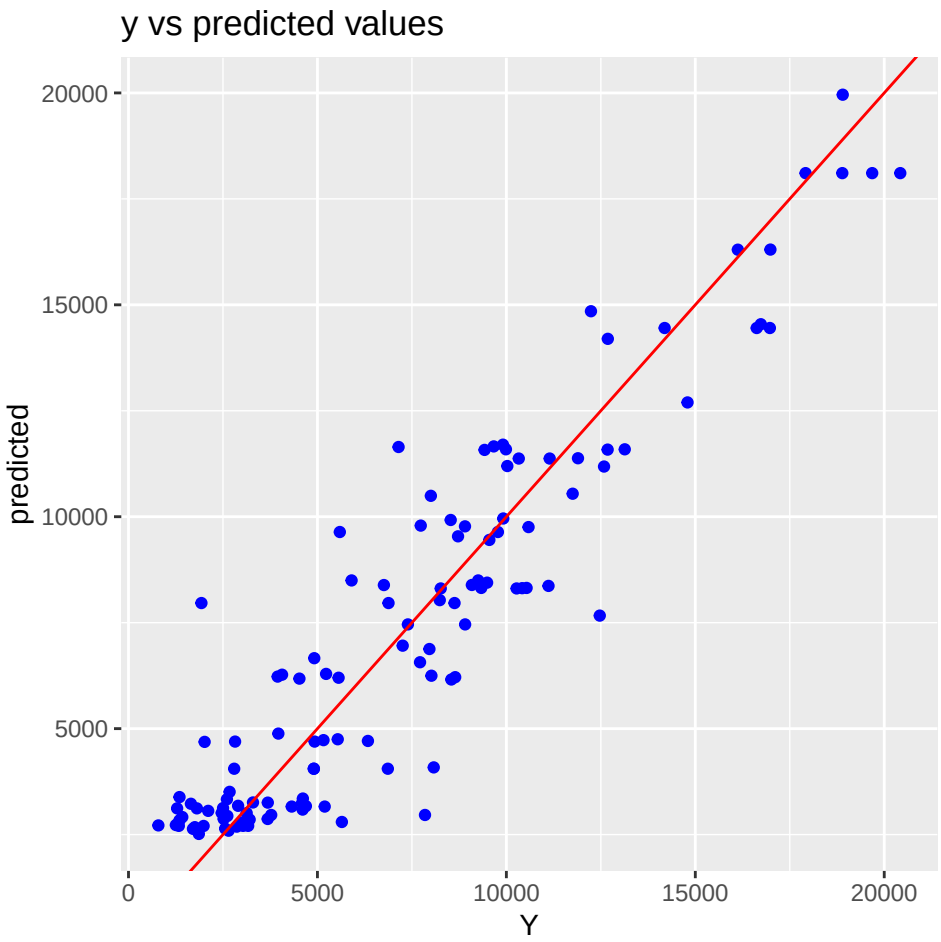
```
mlr_output_folder <- file.path(image_folder, "mlr")
dir.create(mlr_output_folder, showWarnings = FALSE)
```

```
mlr <- lm(Y ~ X1 + X2 + X3 + X4, data = data)
summary(mlr)
```

```
##
## Call:
## lm(formula = Y ~ X1 + X2 + X3 + X4, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6033.4 -1170.8    56.4  1096.4  4883.4
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4029.735   1348.156   2.989  0.00342 **
## X1              6.174      0.618   9.991  < 2e-16 ***
## X2           1806.718    180.610  10.003  < 2e-16 ***
## X3              3.421     14.425   0.237  0.81294
## X4          -2876.276    524.407  -5.485 2.46e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1774 on 116 degrees of freedom
## Multiple R-squared:  0.8644, Adjusted R-squared:  0.8598
## F-statistic: 184.9 on 4 and 116 DF,  p-value: < 2.2e-16
```

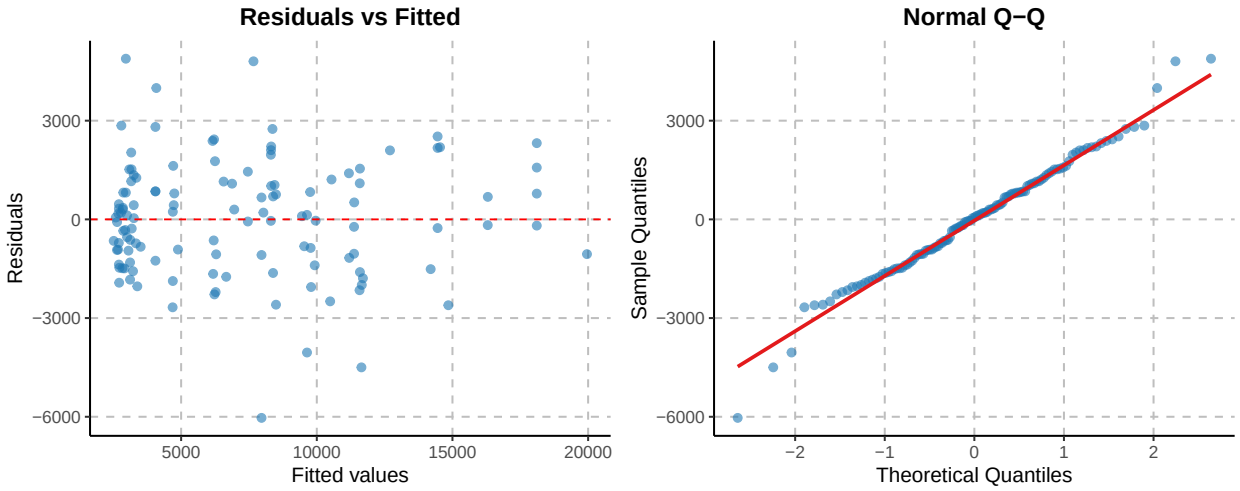
```
predicted <- predict(mlr, data[, c("X1", "X2", "X3", "X4")])
```

```
ggplot(data, aes(x = Y, y = predicted)) +
  geom_point(color = "blue") +
  labs(title = "y vs predicted values") +
  geom_abline(intercept = 0, slope = 1, color = "red")
```




```
plot_resid_mlr <- ggplot(
  data = data.frame(fitted = fitted(mlr), residuals = resid(mlr)),
  aes(x = fitted, y = residuals)
) +
  geom_point(color = "#1f78b4", alpha = 0.6, size = 2) +
  labs(
    x = "Fitted values",
    y = "Residuals",
    title = "Residuals vs Fitted"
  ) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  scale_fill_viridis_d(option = "C") +
  theme_classic(base_size = 12) +
  theme(
    plot.title = element_text(hjust = 0.5, face = "bold", size = 14),
    axis.title = element_text(size = 12),
    panel.grid.major = element_line(color = "gray", linetype = "dashed"),
  )

qqplot_mlr <- ggplot(
  data = data.frame(sample = resid(mlr)),
  aes(sample = sample)
) +
```

```
  stat_qq(color = "#1f78b4", alpha = 0.6, size = 2) +
  stat_qq_line(color = "#e31a1c", size = 1) +
  labs(
    x = "Theoretical Quantiles",
    y = "Sample Quantiles",
    title = "Normal Q-Q"
  ) +
  theme_classic(base_size = 12) +
  theme(
    plot.title = element_text(hjust = 0.5, face = "bold", size = 14),
    axis.title = element_text(size = 12),
    panel.grid.major = element_line(color = "gray", linetype = "dashed")
  )

mlr_res_fig <- grid.arrange(plot_resid_mlr, qqplot_mlr, ncol = 2)
```




```
ggsave(
  file.path(
    mlr_output_folder,
    "residuals_vs_fitted_qqplot_transformed.pdf"
  ),
  mlr_res_fig,
  width = 10, height = 4
)
```

We succeed in the goal by doing the transformation $y' = \log^4 y$

So The model becomes:

$$\log^4 y = \beta_0 + \beta_1\sqrt{x_1} + \beta_2 x_2 + \beta_3 x_3 + \beta_4(2 - x_4) + \epsilon$$

5.2 Adequacy of the model

To check the adequacy of the model, we build an ANOVA / ANCOVA table for each variable. We can find that x_1', x_2, x_4' have information with y, while x_3 can be removed as it have a great p-value.

```r
eliminate_x1_mlr <- lm(Y ~ X2 + X3 + X4, data = data)
eliminate_x2_mlr <- lm(Y ~ X1 + X3 + X4, data = data)
eliminate_x3_mlr <- lm(Y ~ X1 + X2 + X4, data = data)
eliminate_x4_mlr <- lm(Y ~ X1 + X2 + X3, data = data)
```

```r
anova(eliminate_x1_mlr, mlr)
```

```
## Analysis of Variance Table
##
## Model 1: Y ~ X2 + X3 + X4
## Model 2: Y ~ X1 + X2 + X3 + X4
##   Res.Df       RSS Df Sum of Sq      F    Pr(>F)
## 1    117 679542372
## 2    116 365252858  1 314289514 99.815 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
anova(eliminate_x2_mlr, mlr)
```

```
## Analysis of Variance Table
##
## Model 1: Y ~ X1 + X3 + X4
## Model 2: Y ~ X1 + X2 + X3 + X4
##   Res.Df       RSS Df Sum of Sq      F    Pr(>F)
## 1    117 680342805
## 2    116 365252858  1 315089947 100.07 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
anova(eliminate_x3_mlr, mlr)
```

```
## Analysis of Variance Table
##
## Model 1: Y ~ X1 + X2 + X4
## Model 2: Y ~ X1 + X2 + X3 + X4
##   Res.Df       RSS Df Sum of Sq      F Pr(>F)
## 1    117 365429973
## 2    116 365252858  1    177115 0.0562 0.8129
```

```r
anova(eliminate_x4_mlr, mlr)
```

```
## Analysis of Variance Table
##
## Model 1: Y ~ X1 + X2 + X3
## Model 2: Y ~ X1 + X2 + X3 + X4
##   Res.Df       RSS Df Sum of Sq      F    Pr(>F)
## 1    117 459976649
## 2    116 365252858  1  94723791 30.083 2.457e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

So we try to eliminate X_3.

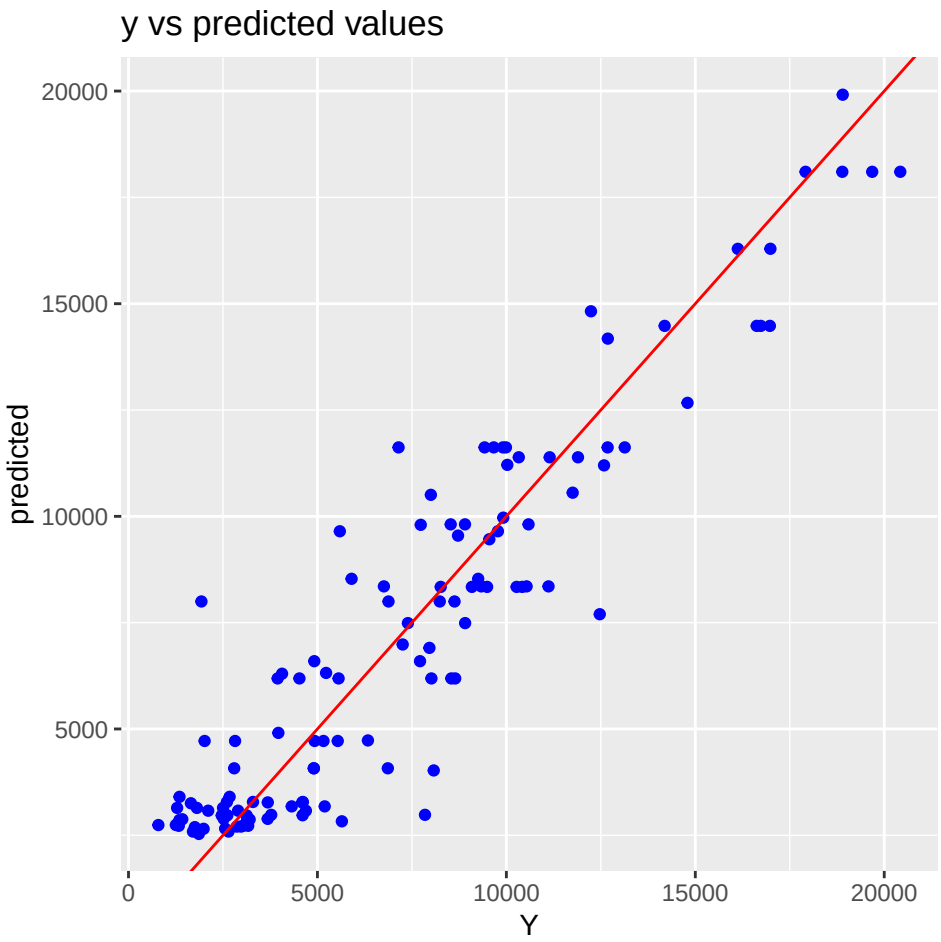
```
mlr <- eliminate_x3_mlr
```

```
summary(mlr)
```

```
##
## Call:
## lm(formula = Y ~ X1 + X2 + X4, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6068.1 -1185.4    60.4  1121.5  4865.5
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4079.1529  1326.5738   3.075  0.00262 **
## X1              6.1950     0.6093  10.168  < 2e-16 ***
## X2           1811.5492   178.7319  10.136  < 2e-16 ***
## X4          -2857.4933   516.2981  -5.535 1.94e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1767 on 117 degrees of freedom
## Multiple R-squared:  0.8644, Adjusted R-squared:  0.8609
## F-statistic: 248.6 on 3 and 117 DF,  p-value: < 2.2e-16
```

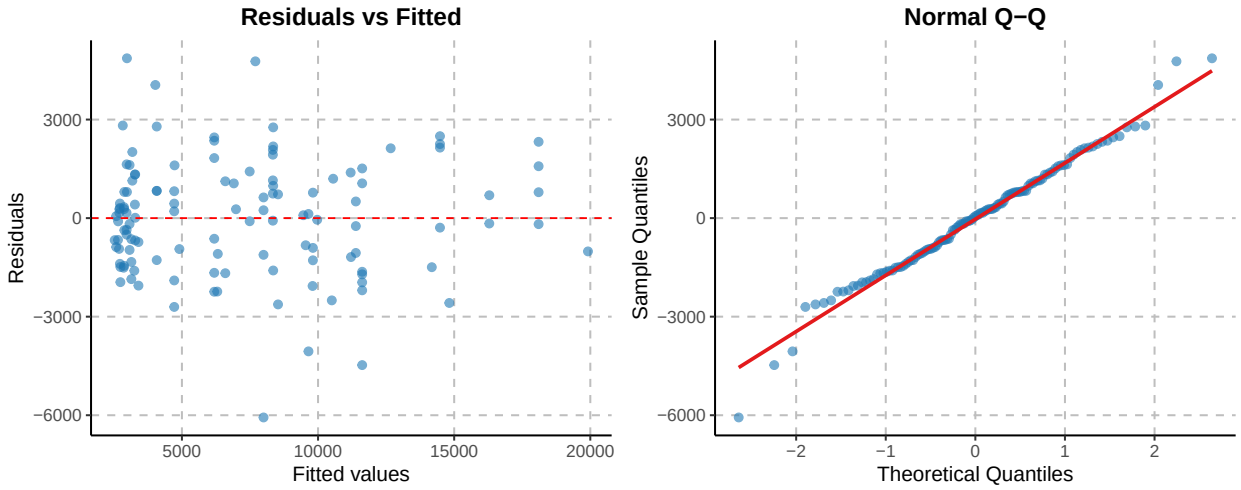
```
predicted <- predict(mlr, data)
```

```
plot <- ggplot(data, aes(x = Y, y = predicted)) +
  geom_point(color = "blue") +
  labs(title = "y vs predicted values") +
  geom_abline(intercept = 0, slope = 1, color = "red")
file_path <- file.path(
  mlr_output_folder,
  "y_vs_predicted_values_reduced_model.pdf"
)
ggsave(file_path, plot, width = 5, height = 5)
plot
```




```
plot_resid_mlr <- ggplot(
  data = data.frame(fitted = fitted(mlr), residuals = resid(mlr)),
  aes(x = fitted, y = residuals)
) +
  geom_point(color = "#1f78b4", alpha = 0.6, size = 2) +
  labs(
    x = "Fitted values",
    y = "Residuals",
    title = "Residuals vs Fitted"
  ) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  scale_fill_viridis_d(option = "C") +
  theme_classic(base_size = 12) +
  theme(
    plot.title = element_text(hjust = 0.5, face = "bold", size = 14),
    axis.title = element_text(size = 12),
    panel.grid.major = element_line(color = "gray", linetype = "dashed"),
  )

qqplot_mlr <- ggplot(
  data = data.frame(sample = resid(mlr)),
  aes(sample = sample)
) +
```

```
  stat_qq(color = "#1f78b4", alpha = 0.6, size = 2) +
  stat_qq_line(color = "#e31a1c", size = 1) +
  labs(
    x = "Theoretical Quantiles",
    y = "Sample Quantiles",
    title = "Normal Q-Q"
  ) +
  theme_classic(base_size = 12) +
  theme(
    plot.title = element_text(hjust = 0.5, face = "bold", size = 14),
    axis.title = element_text(size = 12),
    panel.grid.major = element_line(color = "gray", linetype = "dashed")
  )

mlr_res_fig <- grid.arrange(plot_resid_mlr, qqplot_mlr, ncol = 2)
```




```
ggsave(file.path(mlr_output_folder, "reduced_residuals_vs_fitted_qqplot.pdf"),
  mlr_res_fig,
  width = 10, height = 4
)
```

6 Prediction

6.1 Data

```
x1 <- 50000
x2 <- 3
x3 <- 60
x4 <- 2
```

Then we do transformation for every values:

$$\begin{cases} x_1' = \sqrt{x_1} \\ x_4' = 2 - x_4 \end{cases}$$

```r
x1 <- sqrt(x1)
x4 <- 2 - x4
```

6.2 Point Estimation

```r
y_hat <- predict(mlr, data.frame(X1 = x1, X2 = x2, X3 = x3, X4 = x4))
```

```r
y_hat
```

```
##        1
## 10899.05
```

```r
exp(y_hat^(1 / 4))
```

```
##        1
## 27379.73
```

We transform back

$$\hat{y} = \exp \sqrt[4]{\hat{y}'}$$

```r
y_hat <- exp(y_hat^(1 / 4))
```

```r
y_hat
```

```
##        1
## 27379.73
```

6.3 Interval Estimation

Let's try to get the 90% interval

```r
alpha <- 0.1
interval <- predict(mlr, data.frame(X1 = x1, X2 = x2, X3 = x3, X4 = x4),
  interval = "confidence", level = 1 - alpha
)
interval
```

```
##        fit      lwr      upr
## 1 10899.05 9329.203 12468.89
```

```r
l <- exp(interval[2]^(1 / 4))
r <- exp(interval[3]^(1 / 4))
print(paste("(", l, ", ", r, ")"))
```

```
## [1] "( 18544.1422848705 ,  38836.9550734659 )"
```