Homework 2

Güney Berkay Ateş
120200047

In my conducted multiple regression analysis, I aimed to explore the relationships between the dependent variable, `heart.disease`, and two independent variables, `biking` and `smoking`. The regression model was fitted using the lm() function in R, providing insights into the quantitative impact of biking and smoking on the occurrence of heart disease. The summary of the regression model allowed us to examine the coefficients, assess the statistical significance of predictors, and analyze the overall goodness-of-fit. Subsequent diagnostic plots, including scatterplot matrices and residuals vs. fitted values plots, helped us evaluate the assumptions of linearity, homoscedasticity, and normality of residuals. Additionally, we examined the potential multicollinearity among the predictor variables.

In the R code I ran on my 'heart' dataset, I performed a multiple regression analysis to understand the relationship between 'heart.disease' and the variables 'biking' and 'smoking'. The regression equation I obtained is heart.disease = 14.98 - 0.20 * biking + 0.18 * smoking. Both 'biking' and 'smoking' turned out to be highly significant, with negative and positive coefficients, respectively. The residuals exhibited a relatively small spread, indicating a good fit. The high multiple R-squared value of 0.9796 suggests that almost 98% of the variability in 'heart.disease' can be explained by the linear relationship with 'biking' and 'smoking'. The diagnostic plots, including scatterplot matrices and residuals vs. fitted values plots, helped assess the assumptions of the model. Moreover, a check for multicollinearity revealed a strong negative correlation between 'heart.disease' and 'biking'. The hypothesis test on the coefficient of 'biking' confirmed its significant association with heart disease. This analysis provides valuable insights into how both 'biking' and 'smoking' collectively influence the occurrence of heart disease in my dataset.

Code:

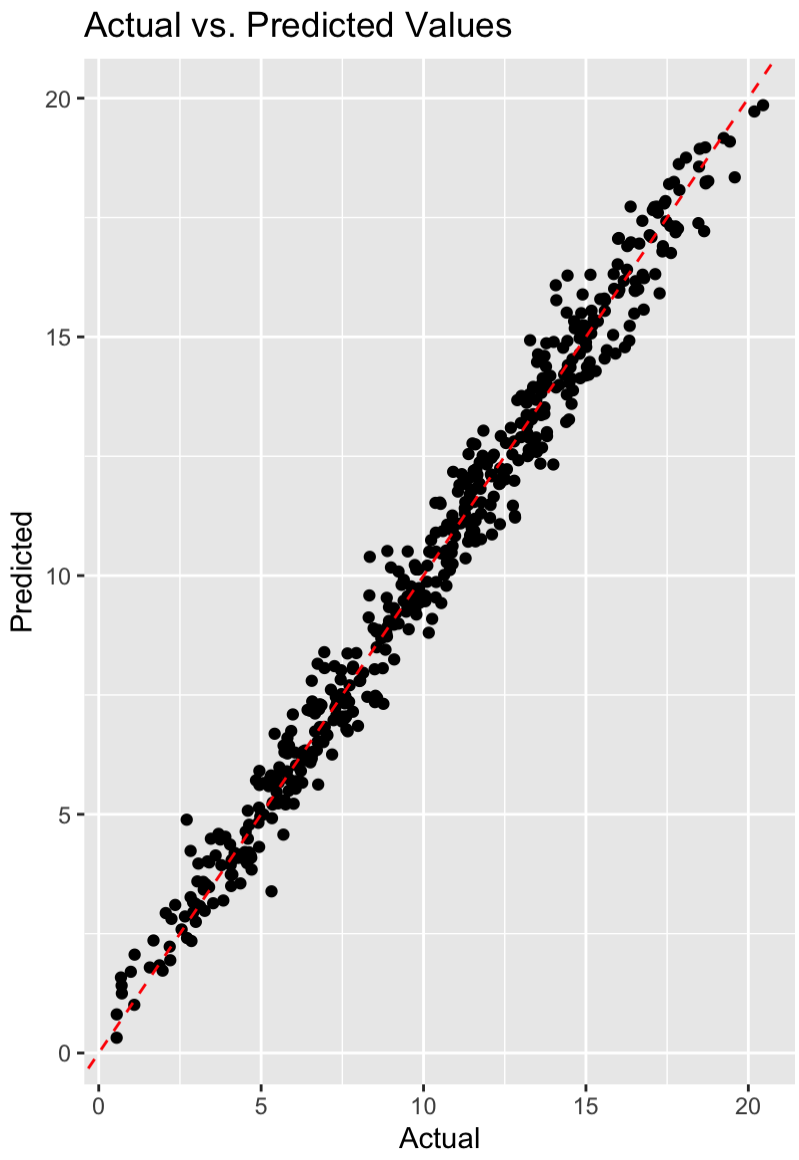
```
# Install and load necessary libraries
install_and_load <- function(package_name) {
  if (!requireNamespace(package_name, quietly = TRUE)) {
    install.packages(package_name)
  }
  library(package_name, character.only = TRUE)
}

# Install and load libraries
install_and_load("ggplot2")
```

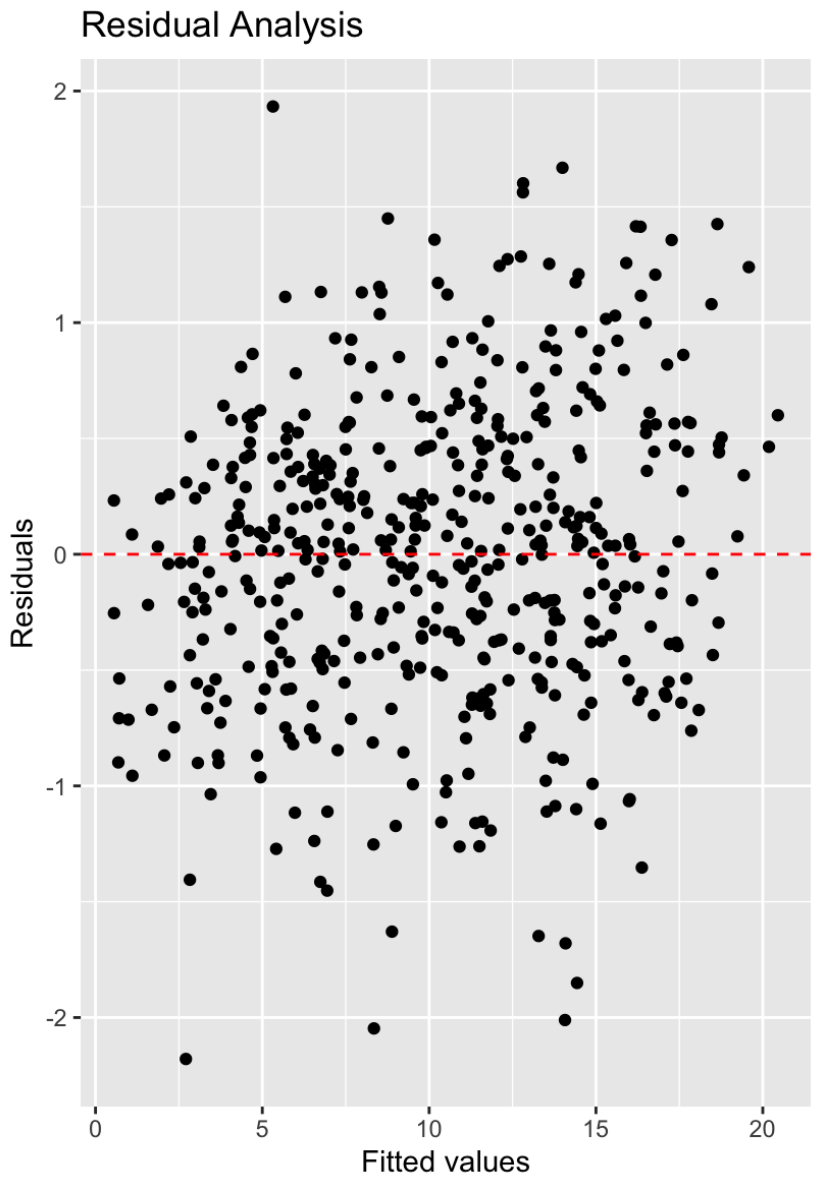
```
install_and_load("dplyr")

# Assuming your dataset is named 'heart'
# Replace 'heart' with the actual name of your dataset if it's different

# Fit the multiple regression model
model <- lm(heart.disease ~ biking + smoking, data = heart)

# Display the summary of the regression model
summary(model)

# Check assumptions
par(mfrow = c(2, 2))
plot(model)

# Make predictions
predictions <- predict(model, newdata = heart)

# Visualize actual vs. predicted values
ggplot(heart, aes(x = heart.disease, y = predictions)) +
  geom_point() +
  geom_abline(intercept = 0, slope = 1, color = "red", linetype = "dashed") +
  labs(title = "Actual vs. Predicted Values", x = "Actual", y = "Predicted")

# Residual analysis
residuals <- residuals(model)

# Visualize residuals
ggplot(heart, aes(x = heart.disease, y = residuals)) +
  geom_point() +
  geom_hline(yintercept = 0, color = "red", linetype = "dashed") +
  labs(title = "Residual Analysis", x = "Fitted values", y = "Residuals")

# Scatterplot Matrix
pairs(heart[, c("heart.disease", "biking", "smoking")], main = "Scatterplot Matrix")

# Residuals vs. Fitted Values Plot
plot(model, 1, main = "Residuals vs. Fitted Values", col = "blue")
abline(h = 0, col = "red", lty = 2)

# Check normality of residuals
qqnorm(residuals)
qqline(residuals)
```

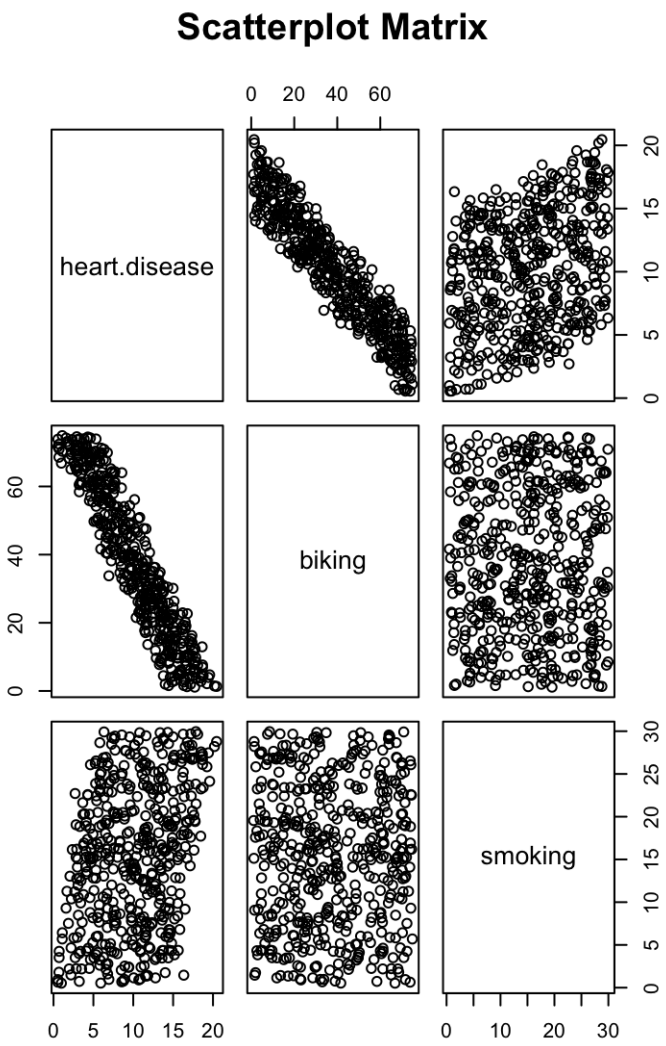
```
# Check homoscedasticity
plot(model, which = 3)

# Check for multicollinearity
cor(heart[, c("heart.disease", "biking", "smoking")])

# Perform hypothesis tests on coefficients
summary(htest <- coef_test(model, hypothesis = "biking = 0"))
```

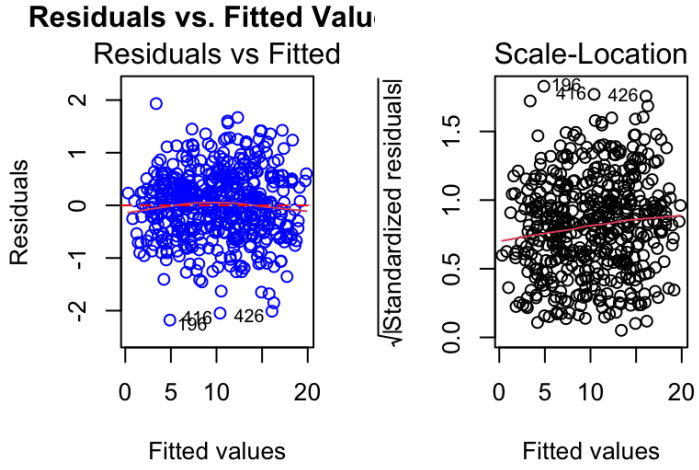
.

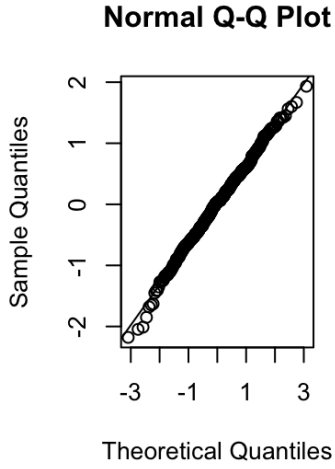
Plots:


Actual vs. Predicted Values
Predicted
Actual
0
5
10
15
20


Residual Analysis
Residuals
Fitted values
2
1
0
-1
-2
0
5
10
15
20


Scatterplot Matrix
heart.disease
biking
smoking
0 20 40 60
0 5 10 15 20
0 20 40 60
0 5 10 15 20
0 10 20 30
0 5 10 15 20 25 30


Residuals vs. Fitted Valu
Residuals vs Fitted
Residuals
Fitted values
196
416
426
Scale-Location
√|Standardized residuals|
Fitted values


Normal Q-Q Plot
Sample Quantiles
Theoretical Quantiles

Console:

```
> # Install and load libraries
> install_and_load("ggplot2")
> install_and_load("dplyr")
> # Fit the multiple regression model
> model <- lm(heart.disease ~ biking + smoking, data = heart)
> # Display the summary of the regression model
> summary(model)

Call:
lm(formula = heart.disease ~ biking + smoking, data = heart)

Residuals:
    Min      1Q  Median      3Q     Max
-2.1789 -0.4463  0.0362  0.4422  1.9331

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 14.984658   0.080137  186.99   <2e-16 ***
biking      -0.200133   0.001366 -146.53   <2e-16 ***
smoking      0.178334   0.003539   50.39   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.654 on 495 degrees of freedom
Multiple R-squared:  0.9796,  Adjusted R-squared:  0.9795
F-statistic: 1.19e+04 on 2 and 495 DF,  p-value: < 2.2e-16

> # Check assumptions
> par(mfrow = c(2, 2))
> plot(model)
> # Make predictions
> predictions <- predict(model, newdata = heart)
> # Visualize actual vs. predicted values
> ggplot(heart, aes(x = heart.disease, y = predictions)) +
+   geom_point() +
+   geom_abline(intercept = 0, slope = 1, color = "red", linetype = "dashed") +
+   labs(title = "Actual vs. Predicted Values", x = "Actual", y = "Predicted")
> # Residual analysis
> residuals <- residuals(model)
> # Visualize residuals
> ggplot(heart, aes(x = heart.disease, y = residuals)) +
+   geom_point() +
+   geom_hline(yintercept = 0, color = "red", linetype = "dashed") +
```

```
+   labs(title = "Residual Analysis", x = "Fitted values", y = "Residuals")
> # Scatterplot Matrix
> pairs(heart[, c("heart.disease", "biking", "smoking")], main = "Scatterplot Matrix")
> # Residuals vs. Fitted Values Plot
> plot(model, 1, main = "Residuals vs. Fitted Values", col = "blue")
> abline(h = 0, col = "red", lty = 2)
> # Check normality of residuals
> qqnorm(residuals)
> qqline(residuals)
> # Check homoscedasticity
> plot(model, which = 3)
> # Check for multicollinearity
> cor(heart[, c("heart.disease", "biking", "smoking")])
              heart.disease      biking    smoking
heart.disease     1.0000000 -0.93545547 0.30913098
biking           -0.9354555  1.00000000 0.01513618
smoking           0.3091310  0.01513618 1.00000000
> # Perform hypothesis tests on coefficients
> # For example, testing if the coefficient of biking is significantly different from zero
> summary(htest <- coef_test(model, hypothesis = "biking = 0"))
```