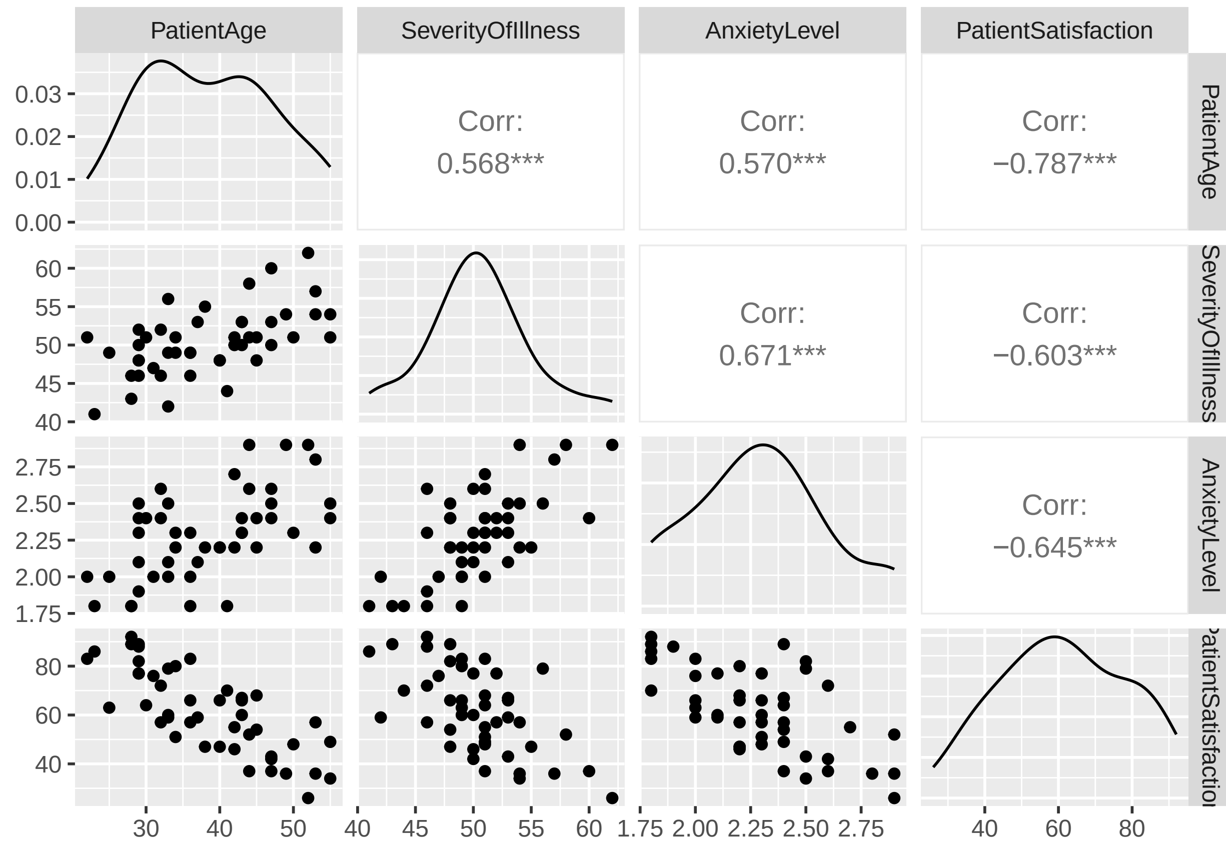
6.15.b

The scatter plot matrix shows correlations between any two variables, and visually represents it. It also represents the distributions of each variable. There seems to be a decent correlation with all variables.



	PatientAge	SeverityOfIllness	AnxietyLevel	PatientSatisfaction
PatientAge	1.0000	0.5680	0.5697	-0.7868
SeverityOfIllness	0.5680	1.0000	0.6705	-0.6029
AnxietyLevel	0.5697	0.6705	1.0000	-0.6446
PatientSatisfaction	-0.7868	-0.6029	-0.6446	1.0000

6.15.c

$\hat{Y} = 158.4913 + -1.1416x_1 + -0.442x_2 + -13.4702x_3$

b_2 is the coefficient for SeverityOfIllness. It shows that patient satisfication decreases by 0.442 every time you increase SeverityOfIllness by one.

6.16.a

$H_0 : \beta_1 = \beta_2 = \beta_3 = 0$

H_a : At least one $\beta_i \neq 0$ for $i \in \{1, 2, 3\}$

$F_s = 30.0520779$

$p = 1.5419726 \times 10^{-10}$

Decision rule: Reject H_0 if $p \leq \alpha$, otherwise accept H_0

Since $p \leq \alpha$, we reject H_0. Therefore, at least one $\beta_i \neq 0$, and this means that there is a regression relation.

6.17.a

The 90% confidence interval is $[64.5285363, 73.4920362]$

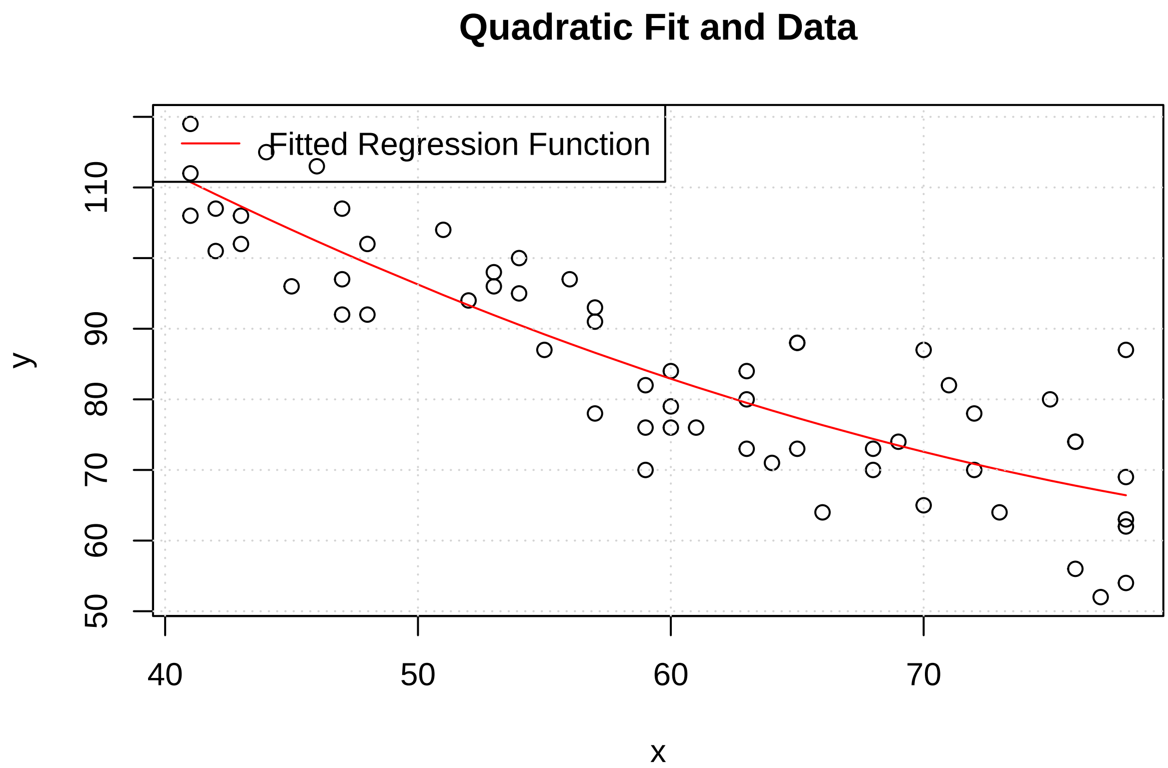
6.17.b

The 90% confidence interval is $[51.5096525, 86.51092]$

7.22

They observed the statistical significance of each regression coefficient rather than the significance of the overall model. There is a different F-statistic to compute when testing for whether there is significance in overall regression model. Even though the expanded model had less statistically significant coefficients, the model itself can still be statistically significant.

8.4.a



The quadratic function seems to be a good fit of the data. However, it looks like a linear function can also fit the data, which would be better as it is simpler.

8.4.b

$H_0 : \beta_1 = \beta_2 = 0$

H_a : At least one $\beta_i \neq 0$ for $i \in \{1, 2\}$

$F_s = 91.839821$

$p = 1.4839242 \times 10^{-18}$

Decision rule: Reject H_0 if $p \leq \alpha$, otherwise accept H_0

Since $p \leq \alpha$, we reject H_0. Therefore, at least one $\beta_i \neq 0$, and this means that there is a regression relation.

8.4.e

$H_0 : \beta_2 = 0$

$H_a : \beta_2 \neq 0$

$p = 0.0810869$

Decision rule: Reject H_0 if $p \leq \alpha$, otherwise accept H_0

Since $p > \alpha$, we accept H_0. Therefore, $\beta_2 = 0$ and we do not need the x^2 term in our model.

8.4.f

$\hat{Y} = 207.3496 + -2.9643x + 0.0148x^2$

Appendix

```
# Libraries
# Load required libraries
library(ggplot2)
library(GGally)
```

```
# 6.15.b
data <- read.table("CH06PR15.txt")
colnames(data) <- c("PatientSatisfaction", "PatientAge",
                    "SeverityOfIllness", "AnxietyLevel")

# Scatter Plot Matrix
ggpairs(data,
        columns = c("PatientAge", "SeverityOfIllness",
                    "AnxietyLevel", "PatientSatisfaction"),
        progress = FALSE)

# Correlation Matrix
correlation_matrix <- cor(data[, c("PatientAge", "SeverityOfIllness",
                                   "AnxietyLevel", "PatientSatisfaction")])
knitr::kable(round(data.frame(correlation_matrix), 4))
```

```
# 6.15.c
model <- lm(PatientSatisfaction ~ PatientAge +
              SeverityOfIllness + AnxietyLevel, data = data)
co <- round(coef(model), 4)
```

```
# 6.16.a
model_summary <- summary(model)
f_statistic <- model_summary$fstatistic[1]
df1 <- model_summary$fstatistic[2]
df2 <- model_summary$fstatistic[3]
p_value <- pf(f_statistic, df1 = df1, df2 = df2, lower.tail = FALSE)
```

```
# 6.17.a
new_data <- data.frame(PatientAge = 35, SeverityOfIllness = 45,
                       AnxietyLevel = 2.2)
confidence_interval <- predict(model, newdata = new_data,
                               interval = "confidence", level = 0.90)
```

```r
# 6.17.b
new_data <- data.frame(PatientAge = 35, SeverityOfIllness = 45,
                       AnxietyLevel = 2.2)
confidence_interval <- predict(model, newdata = new_data,
                               interval = "prediction", level = 0.90)
```

```r
# 8.4.a
data <- read.table("CH01PR27.txt")
colnames(data) <- c("y", "x")
model <- lm(y ~ x + I(x^2), data)

# Plot the data points
predicted <- predict(model)
plot(data$x, data$y, xlab = "x", ylab = "y", main = "Quadratic Fit and Data")

# Sort the data by x values for better visualization
sorted_data <- data[order(data$x), ]

# Plot the fitted regression function
lines(sorted_data$x, predicted[order(data$x)], col = "red")

# Add legend
legend("topleft", legend = "Fitted Regression Function", col = "red", lty = 1)

# Add grid for better visualization (optional)
grid()
```

```r
# 8.4.b
model_summary <- summary(model)
f_statistic <- model_summary$fstatistic[1]
df1 <- model_summary$fstatistic[2]
df2 <- model_summary$fstatistic[3]
p_value <- pf(f_statistic, df1 = df1, df2 = df2, lower.tail = FALSE)
```

```r
# 8.4.e
f_statistic <- model_summary$coefficients[]
p_value <- model_summary$coefficients["I(x^2)", "Pr(>|t|)"]
```

```r
# 8.4.f
co <- round(coef(model), 4)
```