lab03-jp-garcia

AUTHOR
JP Garcia

Model Interpretation: Question 1

```
print("Interpretation B is correct, as the coefficient of the variable educ in the regres
```

```
[1] "Interpretation B is correct, as the coefficient of the variable educ in the
regression we see as 0.54136. This would, in turn, mean that for a one-unit increase in
the years of eduation (one year of additonal education), the average hourly wage will
increase by $0.54"
```

Model Interpretation Question 2

```
"print the correct Interpretation would be Interpretation B, for every additonal year of
```

```
[1] "print the correct Interpretation would be Interpretation B, for every additonal year
of eduation, the average hourly wage increased by 8.27. When the response variable in a
given regression model is logged with ln, the coefficient of the explanatory variable
would represent the percentage change in the response variable for a one unit change in
the explanatory variable. While this happens, the other variables would remain constant.
For case B, for every additonal year of education, the wage would increase by 8.27% "
```

Model Interpretation: Question 3

```
print("The correct interpretation would be interpretation A, for a 1 percent increase in
```

```
[1] "The correct interpretation would be interpretation A, for a 1 percent increase in
sales, the CEO's salary would increase by 0.257%. When both the response variable and
explanatory variable in a regression model get logged, the coefficient of the explanatory
variable would represent the percentage change in the response variable for a 1% change
in the explanatory side."
```

Fitbit: Question 1

```
library(dplyr)
```

```
Attaching package: 'dplyr'

The following objects are masked from 'package:stats':

    filter, lag

The following objects are masked from 'package:base':
```

```
    intersect, setdiff, setequal, union
```

```
fitbit <- read.csv("/Users/jpgarcia/Downloads/fitbit.csv")
head(fitbit)
```

```
       Date caloriesBurned  steps distance plans MinutesOfSedentaryActivities
1 01-01-2016          2.992 10.460     7,92     0                          685
2 01-03-2016          3.117 11.618     8,66    12                          776
3 01-04-2016          2.814 11.130     8,61     8                          900
4 01-05-2016          3.331 14.262     10,6     5                          666
5 01-06-2015          3.354 16.836    12,51     8                          586
6 01-06-2015          3.354 16.836    12,51     8                          586
  MinutesOfLightActivity MinutesOfModerateActivity MinutesOfIntenseActivity
1                    375                         0                        0
2                    217                        59                       29
3                    162                         3                       54
4                    361                        21                       41
5                    248                        35                       70
6                    248                        35                       70
  activityCalories MinutesOfSleep MinutesOfBeingAwake NumberOfAwakings
1            1.558            278                  15               14
2            1.584            420                  31               26
3            1.176            337                  51               23
4            1.937            347                  41               20
5            1.862            426                  54               33
6            1.862            426                  54               27
  MinutesOfRest
1           295
2           451
3           388
4           388
5           495
6           495
```

```
  fitbit <- fitbit %>%
    mutate(Date = as.Date(as.character(Date), format = "%d-%m-%Y"),
           day = factor(weekdays(Date), levels = weekdays(x = as.Date(seq(7),
      origin = "1950-01-01"))),
      month = factor(months(Date), levels = month.name))

  fitbit <- fitbit |>
    mutate(totalMinutes = MinutesOfSleep + MinutesOfLightActivity + MinutesOfModerateAct
           asleepPct = (MinutesOfSleep / totalMinutes)*100,
           sedentaryPct = (MinutesOfSedentaryActivities/ totalMinutes)*100)

  head(fitbit)
```

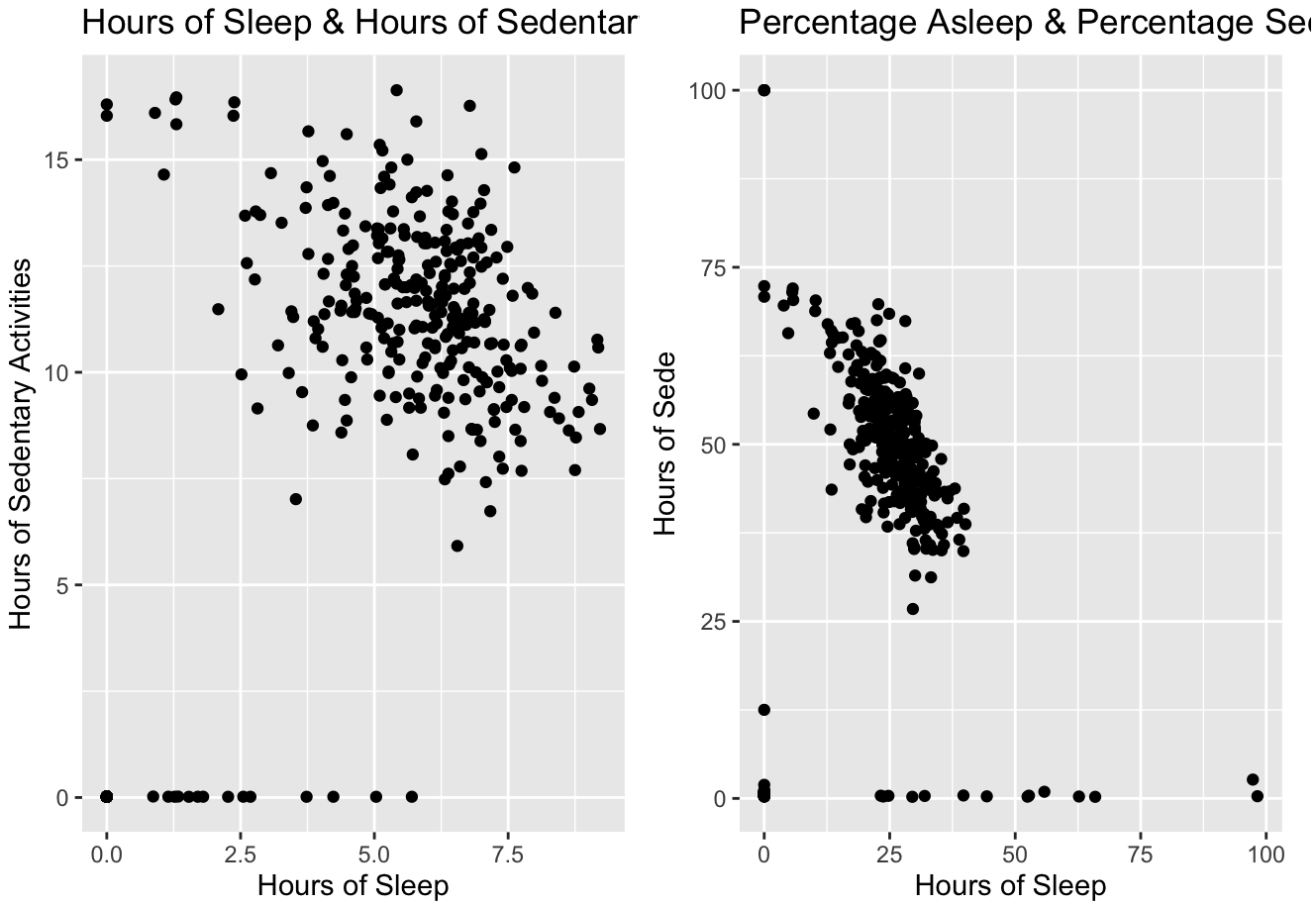
```
       Date caloriesBurned  steps distance plans MinutesOfSedentaryActivities
1 2016-01-01          2.992 10.460     7,92     0                          685
```

```
2 2016-03-01          3.117 11.618     8,66    12                          776
3 2016-04-01          2.814 11.130     8,61     8                          900
4 2016-05-01          3.331 14.262     10,6     5                          666
5 2015-06-01          3.354 16.836    12,51     8                          586
6 2015-06-01          3.354 16.836    12,51     8                          586
  MinutesOfLightActivity MinutesOfModerateActivity MinutesOfIntenseActivity
1                    375                         0                         0
2                    217                        59                        29
3                    162                         3                        54
4                    361                        21                        41
5                    248                        35                        70
6                    248                        35                        70
  activityCalories MinutesOfSleep MinutesOfBeingAwake NumberOfAwakings
1            1.558            278                  15               14
2            1.584            420                  31               26
3            1.176            337                  51               23
4            1.937            347                  41               20
5            1.862            426                  54               33
6            1.862            426                  54               27
  MinutesOfRest     day   month totalMinutes asleepPct sedentaryPct
1           295  Friday January         1338  20.77728     51.19581
2           451 Tuesday   March         1501  27.98135     51.69887
3           388  Friday   April         1456  23.14560     61.81319
4           388  Sunday     May         1436  24.16435     46.37883
5           495  Monday    June         1365  31.20879     42.93040
6           495  Monday    June         1365  31.20879     42.93040
```

Fitbit: Question 2

```
library(ggplot2)
library(patchwork)
library(dplyr)

fitbit <- fitbit |>
  mutate(
    totalMinutes = MinutesOfSleep + MinutesOfLightActivity + MinutesOfModerateActivity +
    pctAsleep = (MinutesOfSleep / totalMinutes) * 100,
    pctSedentary = (MinutesOfSedentaryActivities / totalMinutes) * 100
  )

fitbit$HoursOfSleep <- fitbit$MinutesOfSleep / 60
fitbit$HoursOfSedentary <- fitbit$MinutesOfSedentaryActivities / 60

p1 <- ggplot(fitbit, aes(x = HoursOfSleep, y = HoursOfSedentary)) +
  geom_point(fill = "lightblue") +
  labs(title = "Hours of Sleep & Hours of Sedentary Activities",
       x = "Hours of Sleep",
       y = "Hours of Sedentary Activities")
p2 <- ggplot(fitbit, aes(x = pctAsleep, y = pctSedentary)) +
  geom_point(fill = "lightblue") +
  labs(title = "Percentage Asleep & Percentage Sedentary",
```

```
       x = "Hours of Sleep",
       y = "Hours of Sede")
p3 <- p1 + p2
print(p3)
```



Fitbit: Question 3

```
library(dplyr)

fitbit |>
  summarise(correlation = cor(pctAsleep, pctSedentary))
```

```
  correlation
1   0.2459617
```

Fitbit: Question 4

```
library(dplyr)
fitbit <- fitbit |>
  mutate(
    totalMinutes = MinutesOfSleep + MinutesOfLightActivity + MinutesOfModerateActivity +
    pctAsleep = (MinutesOfSleep / totalMinutes) * 100,
    pctSedentary = (MinutesOfSedentaryActivities / totalMinutes) * 100
  )
```

```
lm_result <- lm(pctAsleep ~ pctSedentary, data = fitbit)
summary(lm_result)
```

```
Call:
lm(formula = pctAsleep ~ pctSedentary, data = fitbit)

Residuals:
    Min      1Q  Median      3Q     Max
-31.293  -5.567   1.208   5.803  81.844

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  16.36420    1.45441  11.251  < 2e-16 ***
pctSedentary  0.14929    0.03075   4.855 1.79e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 12.58 on 366 degrees of freedom
Multiple R-squared:  0.0605,    Adjusted R-squared:  0.05793
F-statistic: 23.57 on 1 and 366 DF,  p-value: 1.789e-06
```

Fitbit: Question 5

```
filtered_fitbit <- fitbit |>
  filter(pctSedentary > 0)

lm_filtered <- lm(pctAsleep ~ pctSedentary, data = filtered_fitbit)
summary(lm_filtered)
```

```
Call:
lm(formula = pctAsleep ~ pctSedentary, data = filtered_fitbit)

Residuals:
    Min      1Q  Median      3Q     Max
-31.293  -5.567   1.208   5.803  81.844

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  16.36420    1.45441  11.251  < 2e-16 ***
pctSedentary  0.14929    0.03075   4.855 1.79e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 12.58 on 366 degrees of freedom
Multiple R-squared:  0.0605,    Adjusted R-squared:  0.05793
F-statistic: 23.57 on 1 and 366 DF,  p-value: 1.789e-06
```

Inference for SLR: Question 1

```
library(dplyr)
bootstrap_coefficients <- replicate(10000, {
  residual <- residuals(lm_result)
  new_response <- fitted(lm_result) + sample(residual, replace = TRUE)
  coef(lm(new_response ~ pctSedentary, data = fitbit)) [2]
})

bootstrap_CI <- quantile(bootstrap_coefficients, c(0.025, 0.975))
bootstrap_CI
```

```
      2.5%      97.5% 
0.08914433 0.20797679 
```

Inference for SLR: Question 2

```
r_summary <-summary(lm_result)

test_statistic <- r_summary$coefficients["pctSedentary", "t value"]
p_val <- r_summary$coefficients["pctSedentary", "Pr(>|t|)"]

test_statistic
```

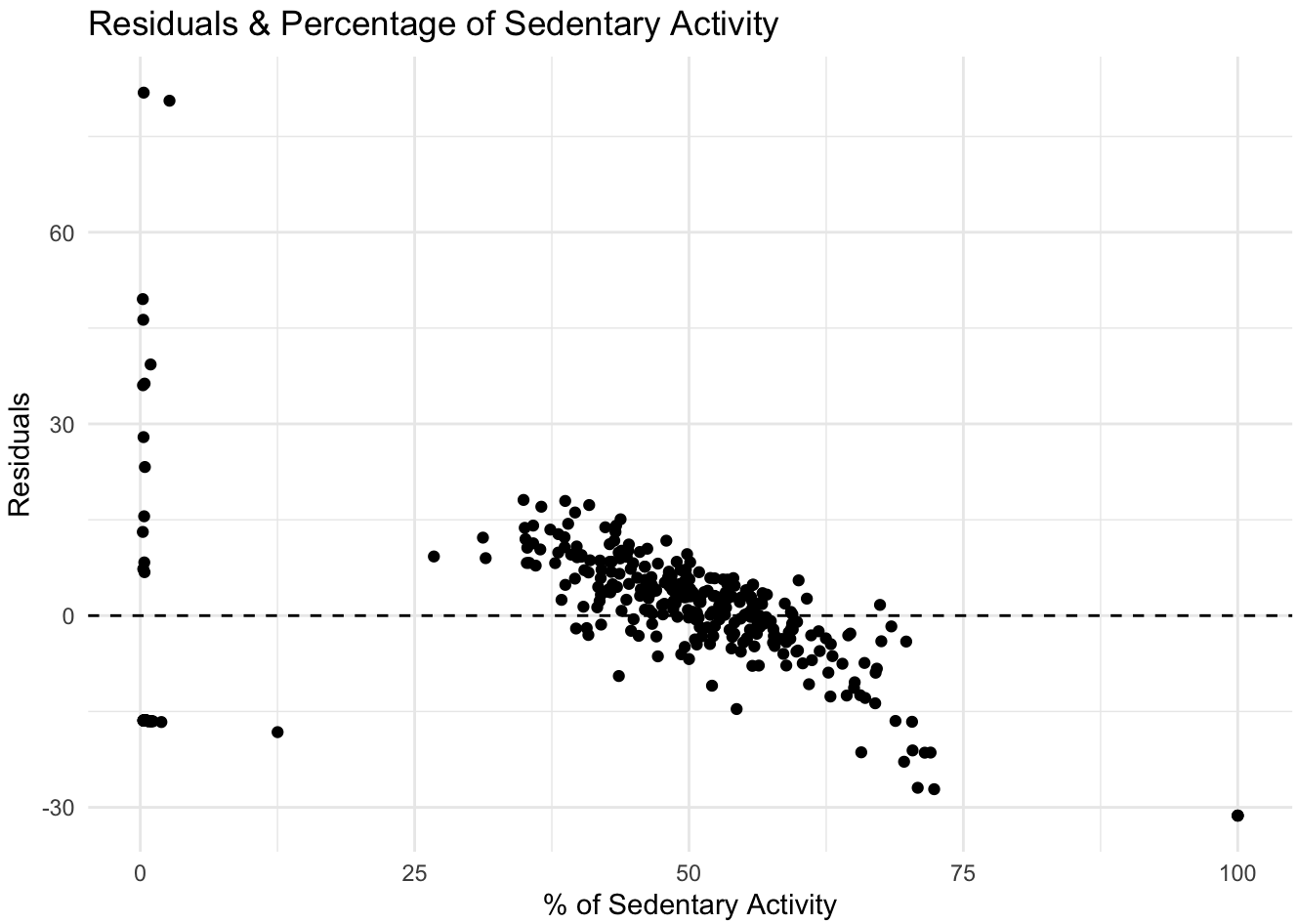
```
[1] 4.854661
```

```
p_val
```

```
[1] 1.789261e-06
```

Model Diagnostics: Question 1

```
library(ggplot2)

residual_vals <- residuals(lm_result)

ggplot(fitbit, aes(x = pctSedentary, y = residual_vals)) +
  geom_point() +
  geom_hline(yintercept = 0, linetype = "dashed") +
  theme_minimal() +
  labs( title = "Residuals & Percentage of Sedentary Activity",
        x = "% of Sedentary Activity",
        y = "Residuals")
```




```
print("Residuals do not seem to be scattered in a random manner around y = 0, and there i
```

```
[1] "Residuals do not seem to be scattered in a random manner around y = 0, and there is
a clear pattern or relationship. This would help us deduce that the linear model may not
adequately capture the relationship between the two variables"
```

```
print("Residuals seem to be centered around y  = 0, this would tell us that on averagem t
```

```
[1] "Residuals seem to be centered around y  = 0, this would tell us that on averagem the
error terms have an average of 0 which is desirable"
```

```
print(" The residuals are moreso spread out at the extremes of the percentage of sedentar
```

```
[1] " The residuals are moreso spread out at the extremes of the percentage of sedentary
activity, moreso at 0% and 100%, which indicates that the residuals do not have a
constant variance"
```

Local Fitting: Question 1

```
library(dplyr)
fitbit <- fitbit[fitbit$pctSedentary > 0, ]
```

```
polyfit <- lm(pctAsleep ~ poly(pctSedentary, 2), data = fitbit)
summary(polyfit)
```

```
Call:
lm(formula = pctAsleep ~ poly(pctSedentary, 2), data = fitbit)

Residuals:
    Min      1Q  Median      3Q     Max 
-20.233  -5.134  -0.117   3.201  87.232 

Coefficients:
                      Estimate Std. Error t value Pr(>|t|)    
(Intercept)              22.67       0.57  39.769  < 2e-16 ***
poly(pctSedentary, 2)1   61.05      10.93   5.583 4.61e-08 ***
poly(pctSedentary, 2)2 -119.34      10.93 -10.915  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 10.93 on 365 degrees of freedom
Multiple R-squared:  0.2917,    Adjusted R-squared:  0.2878 
F-statistic: 75.15 on 2 and 365 DF,  p-value: < 2.2e-16
```

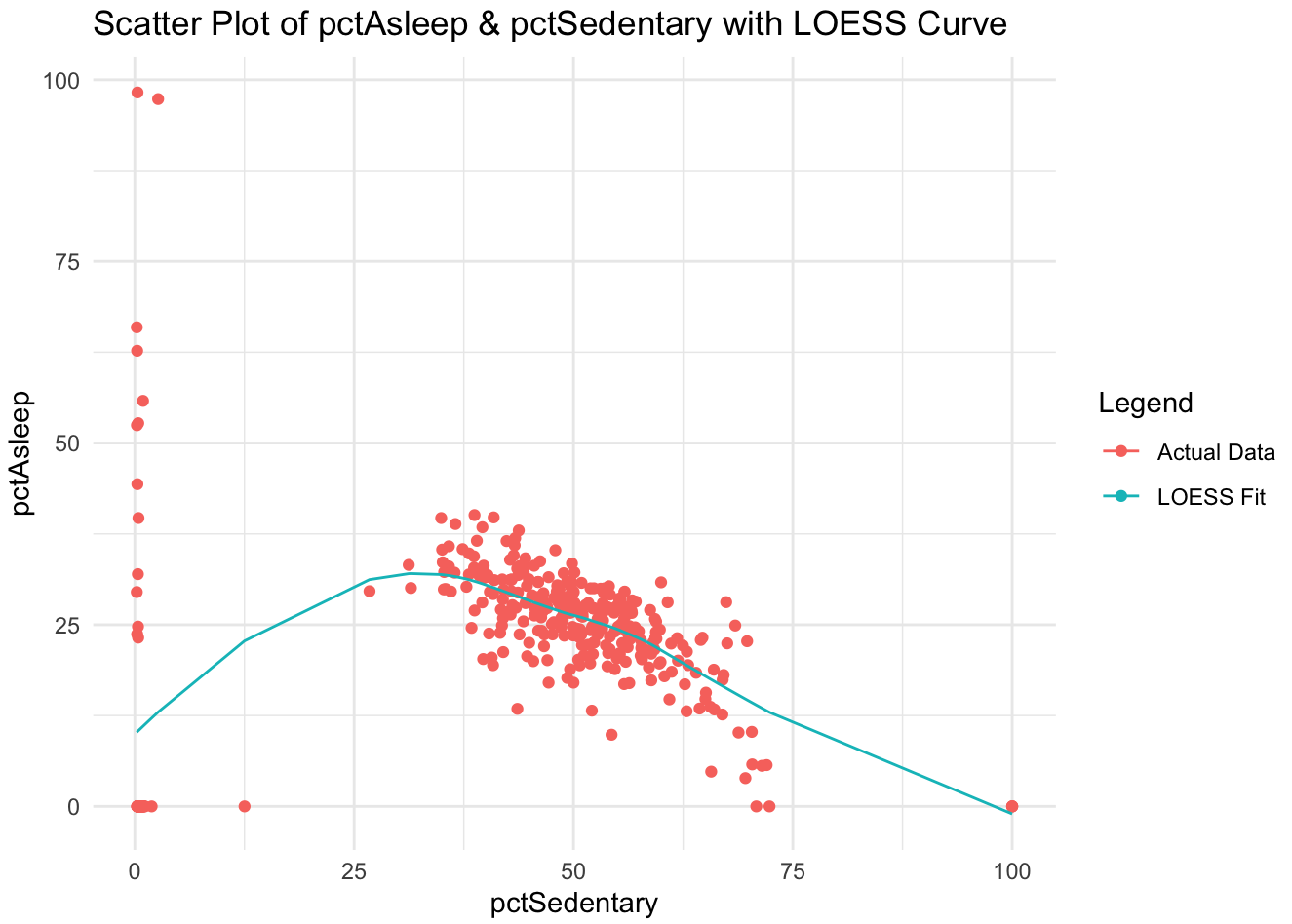
```
fitbit$predicted_pctAsleep_poly <- predict(polyfit)

loess_fit <- loess(pctAsleep ~ pctSedentary, data = fitbit)

fitbit$predicted_pctAsleep_loess <- predict(loess_fit)
head(fitbit)
```

```
        Date caloriesBurned  steps distance plans MinutesOfSedentaryActivities
1 2016-01-01          2.992 10.460     7,92     0                          685
2 2016-03-01          3.117 11.618     8,66    12                          776
3 2016-04-01          2.814 11.130     8,61     8                          900
4 2016-05-01          3.331 14.262     10,6     5                          666
5 2015-06-01          3.354 16.836    12,51     8                          586
6 2015-06-01          3.354 16.836    12,51     8                          586
  MinutesOfLightActivity MinutesOfModerateActivity MinutesOfIntenseActivity
1                    375                         0                        0
2                    217                        59                       29
3                    162                         3                       54
4                    361                        21                       41
5                    248                        35                       70
6                    248                        35                       70
  activityCalories MinutesOfSleep MinutesOfBeingAwake NumberOfAwakings
1            1.558            278                  15               14
2            1.584            420                  31               26
3            1.176            337                  51               23
4            1.937            347                  41               20
5            1.862            426                  54               33
```

```
6            1.862            426                 54                27
  MinutesOfRest     day   month totalMinutes asleepPct sedentaryPct pctAsleep
1           295  Friday January         1338  20.77728     51.19581  20.77728
2           451 Tuesday   March         1501  27.98135     51.69887  27.98135
3           388  Friday   April         1456  23.14560     61.81319  23.14560
4           388  Sunday     May         1436  24.16435     46.37883  24.16435
5           495  Monday    June         1365  31.20879     42.93040  31.20879
6           495  Monday    June         1365  31.20879     42.93040  31.20879
  pctSedentary HoursOfSleep HoursOfSedentary predicted_pctAsleep_poly
1     51.19581     4.633333        11.416667                 26.27793
2     51.69887     7.000000        12.933333                 26.12461
3     61.81319     5.616667        15.000000                 21.77871
4     46.37883     5.783333        11.100000                 27.44452
5     42.93040     7.100000         9.766667                 27.94436
6     42.93040     7.100000         9.766667                 27.94436
  predicted_pctAsleep_loess
1                  25.90273
2                  25.70908
3                  20.18757
4                  27.70460
5                  29.21188
6                  29.21188
```

Local Fitting: Question 2

```
library(ggplot2)

plot <- ggplot(fitbit, aes(x = pctSedentary, y = pctAsleep)) +
geom_point(aes(color = "Actual Data")) +
  geom_line(aes(y = predicted_pctAsleep_loess, color = "LOESS Fit")) +
  labs(title = "Scatter Plot of pctAsleep & pctSedentary with LOESS Curve", color = "Lege
  theme_minimal()
plot
```



Local Fitting: Question 3

```
print("The plot shows us a stable sedentary behavior over the year for all days of the we
```

```
[1] "The plot shows us a stable sedentary behavior over the year for all days of the
week"
```

```
print("Sundays typically have a higher sedentary proportion, suggesting more relaxtion or
```

```
[1] "Sundays typically have a higher sedentary proportion, suggesting more relaxtion or
inactivity in the results"
```

```
print("Friday and Monday are more active compared to Sunday")
```

```
[1] "Friday and Monday are more active compared to Sunday"
```

```
print("Differences between weekdays and weekends are considered minimal")
```

```
[1] "Differences between weekdays and weekends are considered minimal"
```

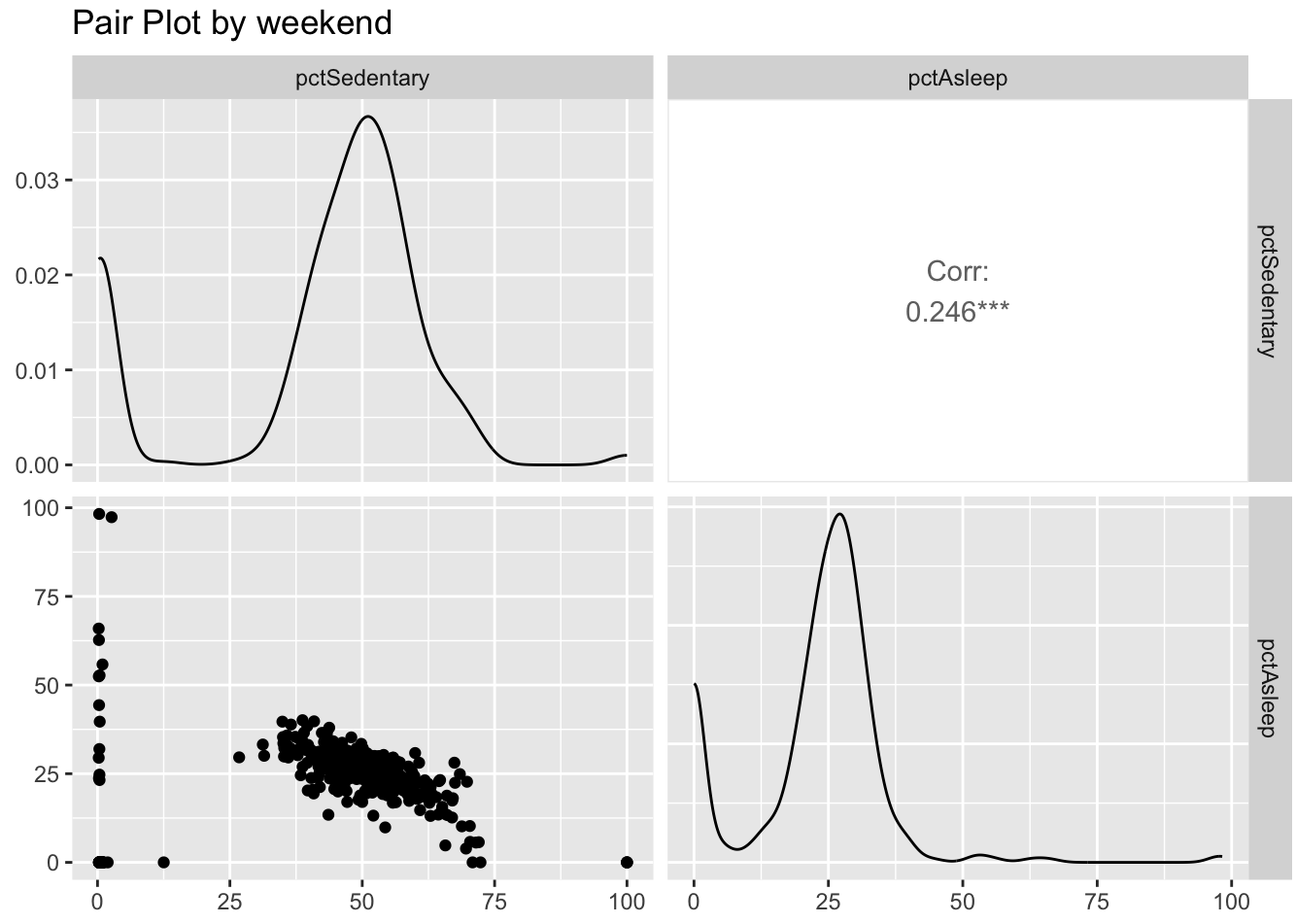
EDA: Question 1

```
library(ggplot2)
library(dplyr)
library(GGally)
```

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

```
fitbit$weekend <- fitbit$day %in% c("Saturday", "Sunday")
library(ggplot2)

ggpairs(fitbit, columns = c("pctSedentary", "pctAsleep"),
        color = "weekend",
        title = "Pair Plot by weekend")
```

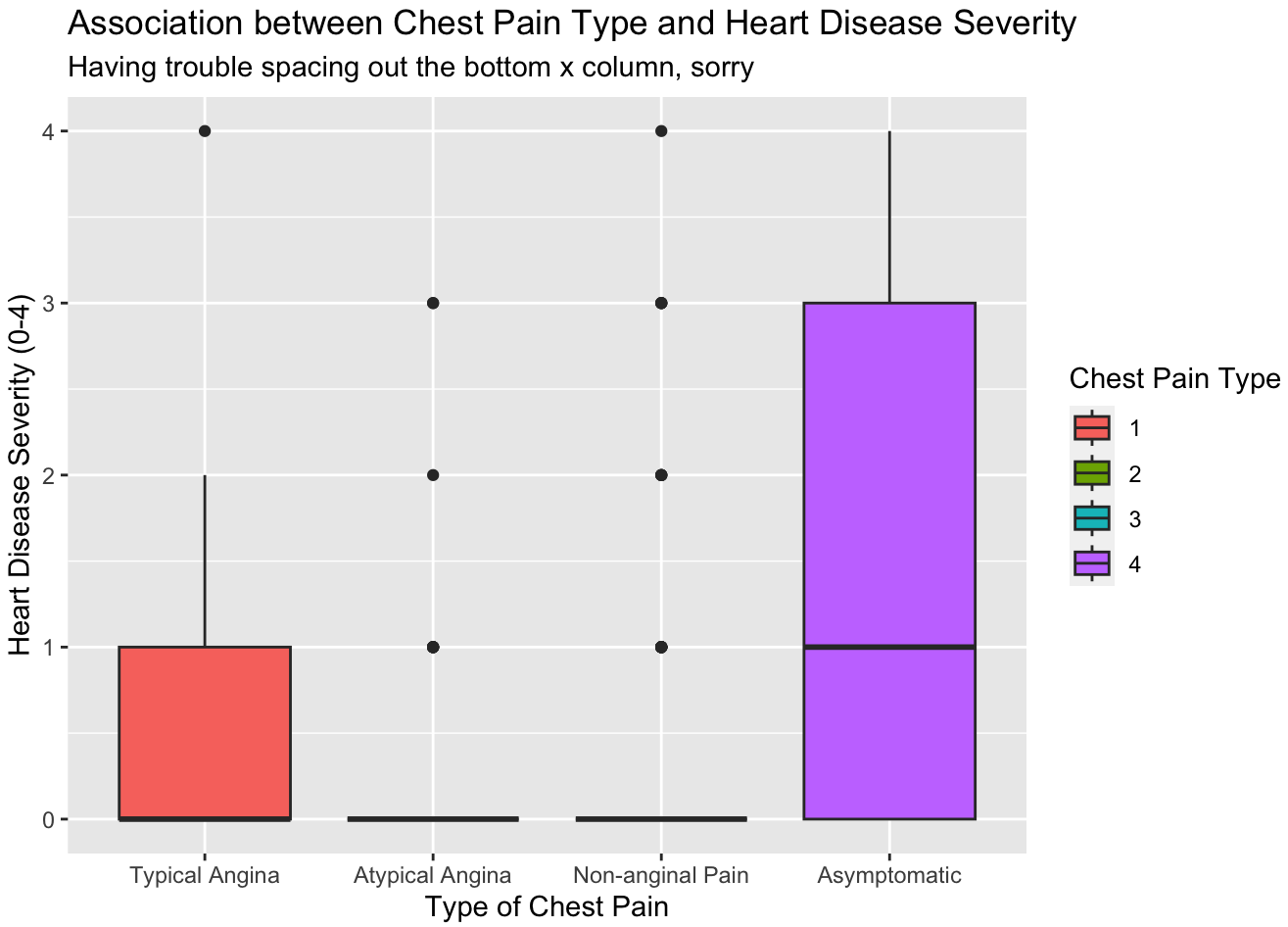
```
Warning in warn_if_args_exist(list(...)): Extra arguments: 'color' are being
ignored.  If these are meant to be aesthetics, submit them using the 'mapping'
variable within ggpairs with ggplot2::aes or ggplot2::aes_string.
```



EDA: Question 2

```
heart_disease <- read.csv("/Users/jpgarcia/jp-garcia-131a/heartDisease.csv")
library(ggplot2)
```

```
ggplot(heart_disease, aes(x = as.factor(cp), y = num)) +
  geom_boxplot(aes(fill = as.factor(cp))) +
  labs(title = "Association between Chest Pain Type and Heart Disease Severity",
       subtitle = "Having trouble spacing out the bottom x column, sorry",
       x = "Type of Chest Pain",
       y = "Heart Disease Severity (0-4)",
       fill = "Chest Pain Type") +
  scale_x_discrete(labels = c("1" = "Typical Angina",
                              "2" = "Atypical Angina",
                              "3" = "Non-anginal Pain",
                              "4" = "Asymptomatic"))
```



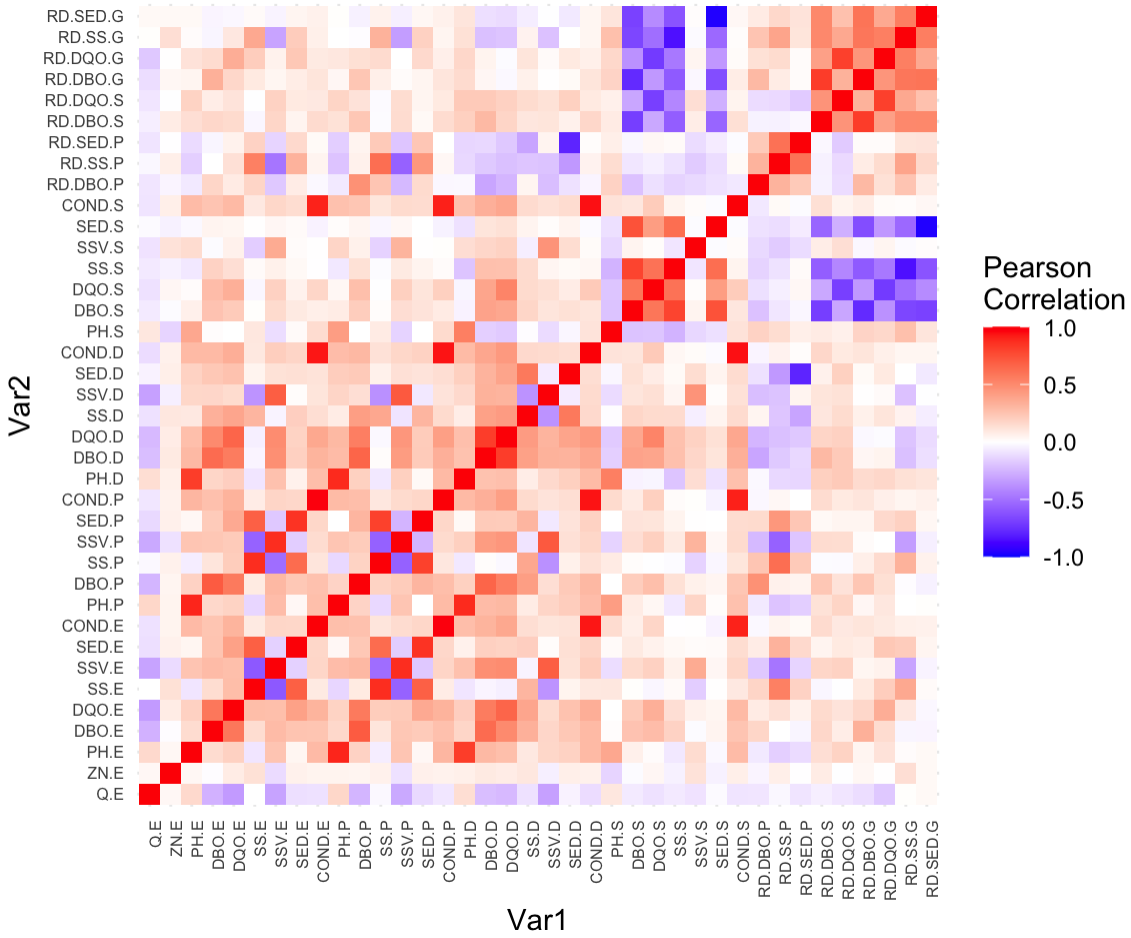
EDA: Question 3

```
library(dplyr)
library(reshape2)
variables <- read.csv("/Users/jpgarcia/Downloads/variabledescriptions.txt")
head(variables)
```

```
                        X1..Q.E.........input.flow.to.plant.
1                        2  ZN-E       (input Zinc to plant)
2                         3  PH-E       (input pH to plant)
3  4  DBO-E      (input Biological demand of oxygen to plant)
```

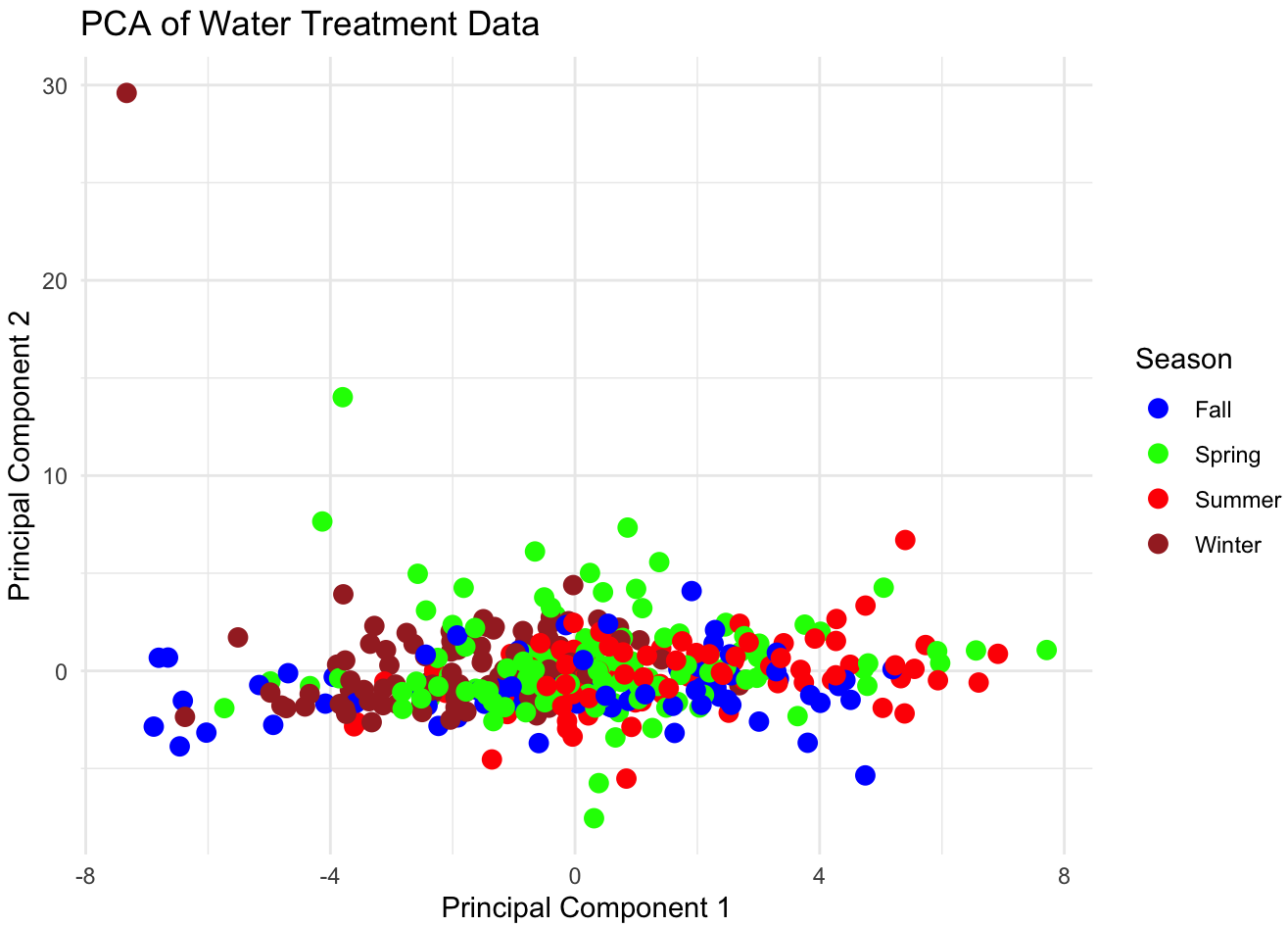
```
4     5  DQO-E      (input chemical demand of oxygen to plant)
5           6  SS-E       (input suspended solids to plant)
6      7  SSV-E      (input volatile supended solids to plant)
```

```
water_data <- read.csv("/Users/jpgarcia/Downloads/water-treatment-cleaned.csv")

numerical_water <- water_data[, !(names(water_data) %in% c("Date", "Year", "Month", "Day"

cor_matrix <- cor(numerical_water, method = "pearson", use = "complete.obs")

melted_cor_matrix <- melt(cor_matrix)

plot <- ggplot(data = melted_cor_matrix, aes(x = Var1, y = Var2, fill = value)) +
  geom_tile() +
  scale_fill_gradient2(low = "blue", high = "red", mid = "white",
                       midpoint = 0, limit = c(-1,1), name = "Pearson\nCorrelation")+
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, vjust = 1,
                                   size = 6, hjust = 1),
        axis.text.y = element_text(size = 6))+
  coord_fixed()
plot
```



PCA:

```
  library(RColorBrewer)
   water <- read.csv(file = "/Users/jpgarcia/Downloads/water-treatment-cleaned.csv",
       header = TRUE, stringsAsFactors = FALSE)
   water$Month <- factor(water$Month, levels = month.name)
   water$Day <- factor(water$Day, levels = weekdays(x = as.Date(seq(7),
       origin = "1950-01-01")))
   water$Year <- factor(water$Year, levels = c(90, 91),
       labels = c("1990", "1991"))
   colDays <- palette()
   names(colDays) <- levels(water$Day)
   colMonths <- c("coral4", brewer.pal(11, "Spectral"))
   names(colMonths) <- levels(water$Month)
   colYear <- c("blue", "green")
   names(colYear) <- levels(water$Year)
   colSeason <- c("Blue", "Green", "Red", "Brown")
   names(colSeason) <- c("Winter", "Spring", "Summer",
   "Fall")


numeric_data <- water[, sapply(water, is.numeric)]


pca_result <- prcomp(numeric_data, scale. = TRUE)

library(ggplot2)

scores <- as.data.frame(pca_result$x)

ggplot(scores, aes(x = PC1, y = PC2)) +
  geom_point(aes(color = water$Season), size = 3) +
  scale_color_manual(values = c("Blue", "Green", "Red", "Brown")) +
  labs(title = "PCA of Water Treatment Data",
       x = "Principal Component 1",
       y = "Principal Component 2",
       color = "Season") +
  theme_minimal()
```


PCA of Water Treatment Data
Principal Component 2
30
20
10
0
-8
-4
0
4
8
Principal Component 1
Season
Fall
Spring
Summer
Winter