Review of coding challenge for Semester 1

2118

2024-01-10

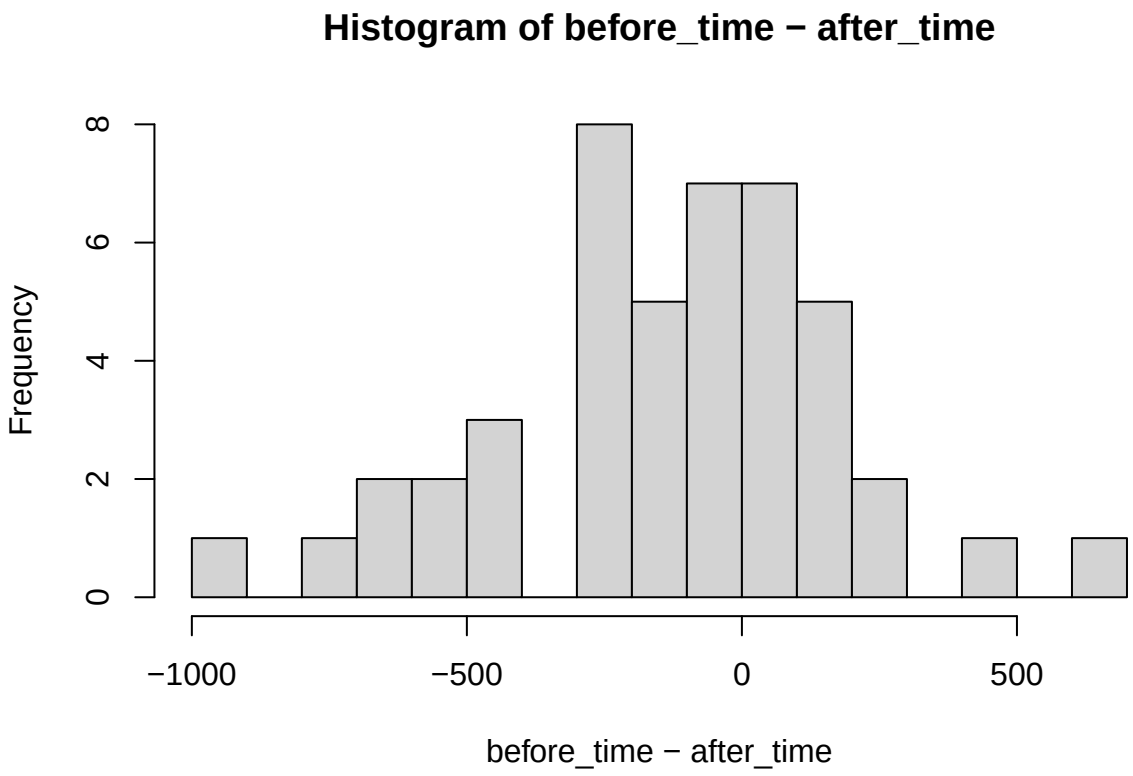
1. Benefits of swimming for long-distance runners

```
swim = read.table("swimming.txt", sep = '\t', header = T)
# head(swimming)
# summary(swim)
# str(swim)
```

1.1 Tidy the data and decide on suitable statistical test.

How are the before and after time distributed?

```
before_time = swim$before_minutes * 60 + swim$before_seconds
before_time = as.integer(before_time)
after_time = swim$after_minutes * 60 + swim$after_seconds
after_time = as.integer(after_time)
hist(before_time - after_time, breaks = 20)
```




```
shapiro.test(before_time - after_time)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  before_time - after_time
## W = 0.9736, p-value = 0.3883
```

```
swim2 = data.frame(swim$names, before_time, after_time)
# head(swim2)
```

- The improvement time (the difference between the after_time and the before_time) are all normally distributed.
- However, we cannot decide whether the improvement time is normally distributed through the histogram.
- It is better to use the shapiro test to test the normality.
- In the shapiro test, $p > 0.05$, so the improvement is normally distributed, so we can use the t test.
- Since every person in the data has a before_time and an after_time, the 2 values are paired.
- As we want to test the improvement time, We decided to use the paired 2-sample t-test. (NOTE: To get full mark, you should emphasize that we test the improvement time.)

1.2 The null and alternative hypotheses

- The null hypothesis (H0): The time used for the half-marathon after the swimming training is no shorter than that before the swimming training.

- The alternative hypothesis (HA) : The time used for the half-marathon after the swimming training is shorter than that before the swimming training.

1.3 Is there a statistically significant improvement on runners' times after swimming?

```
t.test(after_time, before_time, paired = T, alternative = "less")
```

```
##
##  Paired t-test
##
## data:  after_time and before_time
## t = 2.8221, df = 44, p-value = 0.9964
## alternative hypothesis: true mean difference is less than 0
## 95 percent confidence interval:
##      -Inf 206.0872
## sample estimates:
## mean difference
##        129.1778
```

```
t.test(after_time - before_time, mu = 0)
```

```
##
##  One Sample t-test
##
## data:  after_time - before_time
## t = 2.8221, df = 44, p-value = 0.007135
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##   36.9281 221.4275
## sample estimates:
## mean of x
##  129.1778
```

- NOTICE: t.test(x, y, alternative = 'greater') means the HA is y>x
- $p > 0.05$
- We cannot reject the null hypothesis.
- There is insufficient evidence to conclude that the time used for the half-marathon after the swimming training is shorter than that before the swimming training.
- Therefore, there is not a statistically significant improvement on runners' times after swimming.

2. Number of emergency room admissions

2.1 Import the dataset and plot the data in a useful way

```
hosp = read.csv("hospital_admissions.csv")
# head(hosp)
```

```r
# str(hosp)
hosp$week = as.factor(hosp$week)
hosp$weekday = as.factor(hosp$weekday)

subset(hosp, hosp$week == 1)
```

```
##    week weekday hour patients_per_hour
## 1     1  Monday    1                 2
## 2     1  Monday    2                 4
## 3     1  Monday    3                 7
## 4     1  Monday    4                 3
## 5     1  Monday    5                 3
## 6     1  Monday    6                 2
## 7     1  Monday    7                 3
## 8     1  Monday    8                 2
## 9     1  Sunday    1                 2
## 10    1  Sunday    2                 1
## 11    1  Sunday    3                 1
## 12    1  Sunday    4                 3
## 13    1  Sunday    5                 1
## 14    1  Sunday    6                 5
## 15    1  Sunday    7                 2
## 16    1  Sunday    8                 3
```

```r
hosp1 = aggregate(hosp$patients_per_hour,
                  by = list(hosp$week, hosp$weekday),
                  FUN = sum)
names(hosp1)[1] = "week"
names(hosp1)[2] = "weekday"
names(hosp1)[3] = "patients"
# str(hosp1)
# summary(hosp1)
g2.1 = ggplot(data = hosp1[hosp1$weekday == "Monday",],
            mapping = aes(x = week, y = patients)
            )
g2.1 = g2.1 + geom_bar(stat="identity", fill = "orange")
g2.1
```

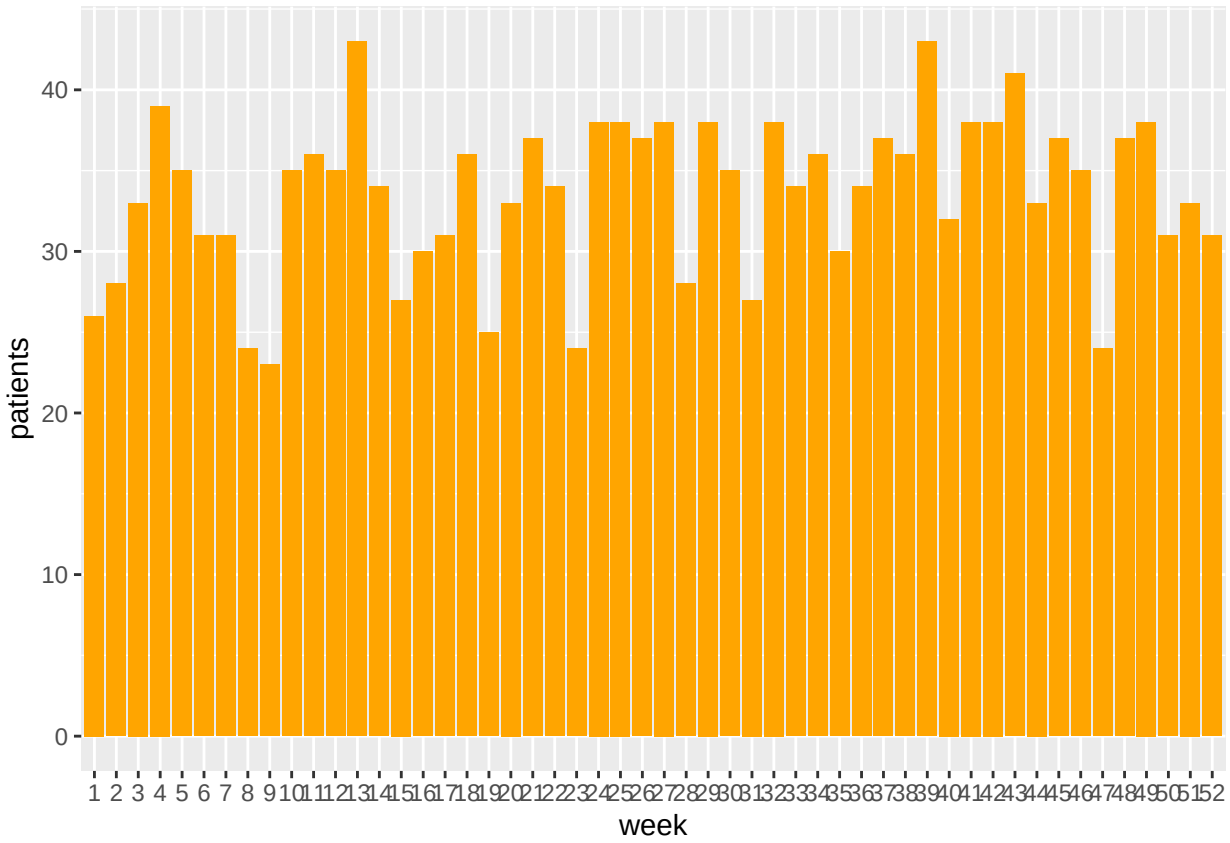


Figure 2.1: Patients on Monday during the year.

```
g2.2 = ggplot(data = hosp1[hosp1$weekday == "Sunday",],
              mapping = aes(x = week, y = patients)
              )
g2.2 = g2.2 + geom_bar(stat = "identity", fill = "purple")
g2.2
```

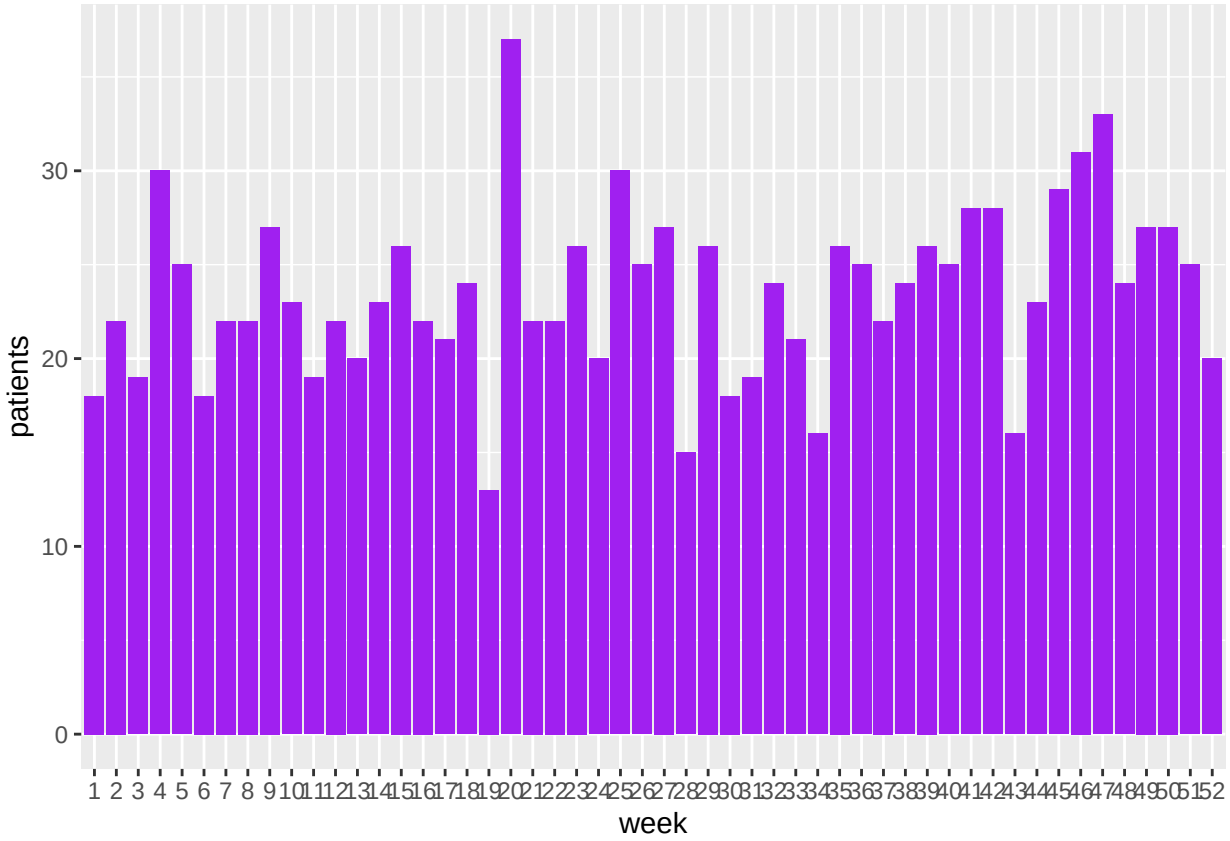
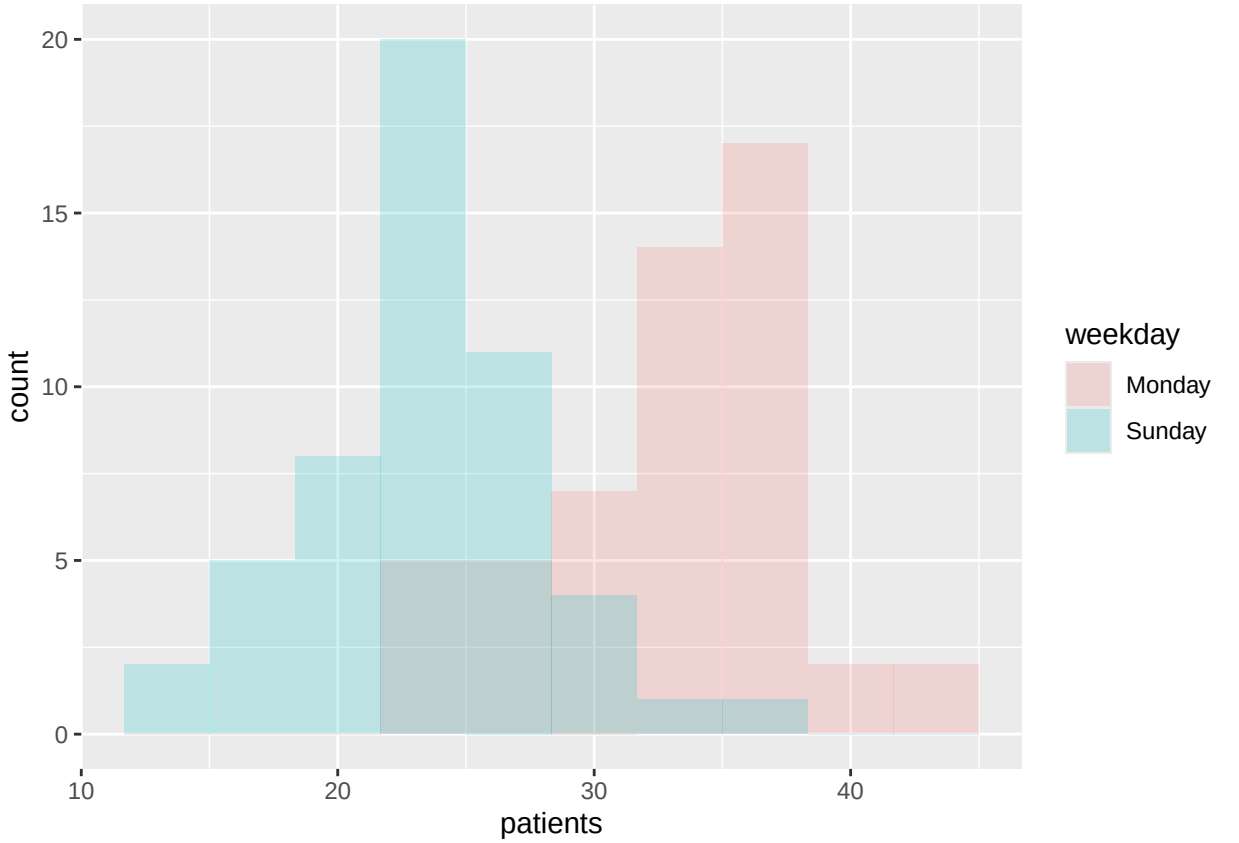


Figure 2.2: Patients on Sunday during the year.

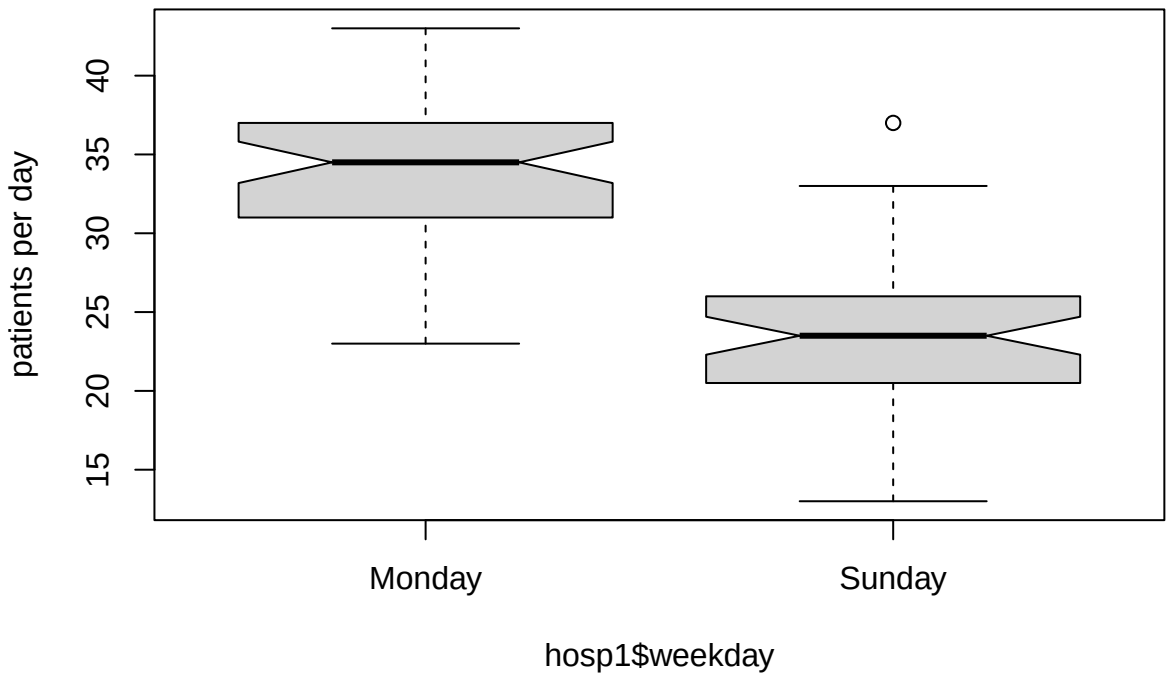
- We can change the x axis into patients_per_day and the y axis into the count.
- In this way, we remove the information of week, because we think the data are not paired, which means the patient number on Sunday and Monday are not correlated.
- This process is like to put the original patient number data into buckets to remove the week information.

```
#hosp1
g2.3 = ggplot(hosp1, mapping = aes(x = patients, fill = weekday))
g2.3 = g2.3 + geom_histogram(position = "identity", alpha = 0.2, bins = 10)
g2.3
```



- You can see that the patients on Sundays are normally distributed. Also, the patients on Mondays are normally distributed.

```
boxplot(hospl$patients ~ hospl$weekday, notch = T, ylab = "patients per day")
```



Is there a difference in patient admission rates between Mondays and Sundays?

- We first form the null (H0) and alternative (HA) hypothesis for this question.
- H0: The patient admission rate on Mondays is no different from that on Sundays.
- HA: The patient admission rate on Mondays is different from that on Sundays.

```
monsums = hosp1[hosp1$weekday == "Monday", ]$patients
sunsums = hosp1[hosp1$weekday == "Sunday", ]$patients
shapiro.test(monsums)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  monsums
## W = 0.95206, p-value = 0.03564
```

```
shapiro.test(sunsums)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  sunsums
## W = 0.987, p-value = 0.838
```

- The patients on both Monday and Sunday during the year are not normally distribution, so the data do not meet the condition to use the t-test.

```
wilcox.test(hosp1[hosp1$weekday == "Monday", ]$patients,
            hosp1[hosp1$weekday == "Sunday", ]$patients,
            paired = F)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  hosp1[hosp1$weekday == "Monday", ]$patients and hosp1[hosp1$weekday == "Sunday", ]$patients
## W = 2492.5, p-value = 1.159e-13
## alternative hypothesis: true location shift is not equal to 0
```

- p-value < 0.05
- We reject H0.
- There is sufficient evidence to conclude that the patient admission rate on Mondays is different from that on Sundays.
- Therefore, there is a significant difference in patient admission rates between Mondays and Sundays.
- Here provide another solution: simulation.

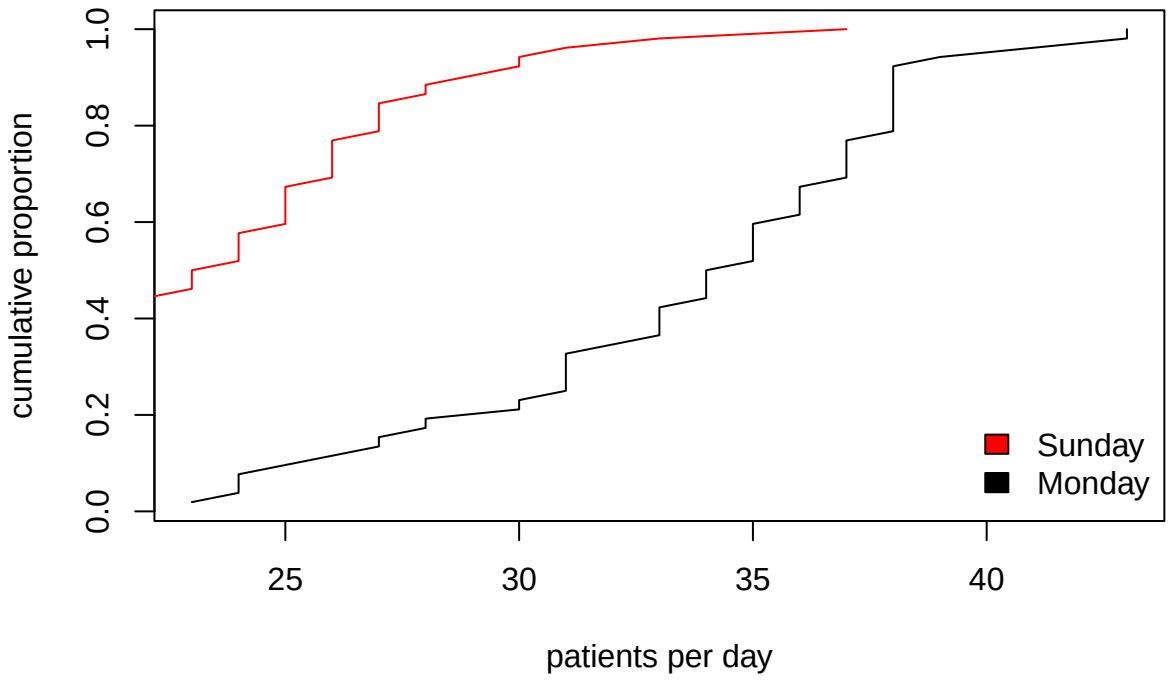
```
monsums = hosp1[hosp1$weekday == "Monday", ]$patients
sunsums = hosp1[hosp1$weekday == "Sunday", ]$patients

real_median = median(monsums) - median(sunsums)

random_sums = c(sunsums, monsums)

count = 0

reptimes = 10000

for(i in 1:reptimes){
  random_index = sample(seq(1,104), 52, replace = F)

  random_mon = random_sums[random_index]
  random_sun = random_sums[-random_index]

  random_median = median(random_mon) - median(random_sun)

  if(random_median >= real_median)
    count = count + 1
}

p_value = 1.0*count/reptimes
p_value
```

```
## [1] 0
```

- p-value < 0.05, so we reject H0.

- NOTE: p-value means the probability that the situation is the same or more extreme (refers to extreme like HA) if H0 is true.

```
#monsums
#sort(monsum)
plot(sort(monsums), 1:length(monsums)/length(monsums), type = 'l',
     ylab = "cumulative proportion", xlab = "patients per day")
lines(sort(sunsums), 1:length(sunsums)/length(sunsums), col = "red")
legend("bottomright", legend = c("Sunday", "Monday"), fill = c("red", "black"),
       bty = "n")
```




```
# bty = box type, default = "o", "n" means none.
```

2.3 Based on your findings, what advice would you give Dr. Horsey?

- We should arrange more staff on Mondays than on Sundays.

3. Spinal cord injury and novel biomaterials

3.1 Import, arrange the data (merge both pieces of data and make the data possible to analyse), and makeit suitable for analysis.

```
data1 = read.csv("SCI_before.csv")
data2 = read.csv("SCI_after.csv")
# head(data1)
# head(data2)
data1$patient_ID = as.factor(data1$patient_ID)
# levels(data1$patient_ID)
data2$patient_ID = as.factor(data2$patient_ID)
# summary(data1)
# summary(data2)
# library(dplyr)
# levels(data2$patient_ID)
data1 = arrange(data1, data1$patient_ID)
data2 = arrange(data2, data2$patient_ID)
# data1$patient_ID == data2$patient_ID
#data1
#data2
#match(data1$patient_ID, data2$patient_ID)
data = cbind(data1, data2$AIS_after)
names(data)[3] = "AIS_after"
# head(data)
```

Any NA?

```
anyNA(data)
```

```
## [1] FALSE
```

- No NA.

Any duplicated?

```
idx1 = which(duplicated(data))
idx2 = which(duplicated(data, fromLast = T))
idx1
```

```
## [1]  3  6  8 10 18 31 35 37
```

```
# data[c(idx1, idx2),  ]
data = data[-idx1, ]
```

Data type?

```
# summary(data)
data$AIS_before = as.factor(data$AIS_before)
data$AIS_after = as.factor(data$AIS_after)
```

Documentation: Remove 8 duplicated rows.

3.2 Check your data carefully. Identify features of the data and discuss your conclusions. Make illustrative plots.

```
g3.1 = ggplot(data = data,
              mapping = aes(x = AIS_before))
g3.1 = g3.1 + geom_bar(fill = "orange")
g3.1
```

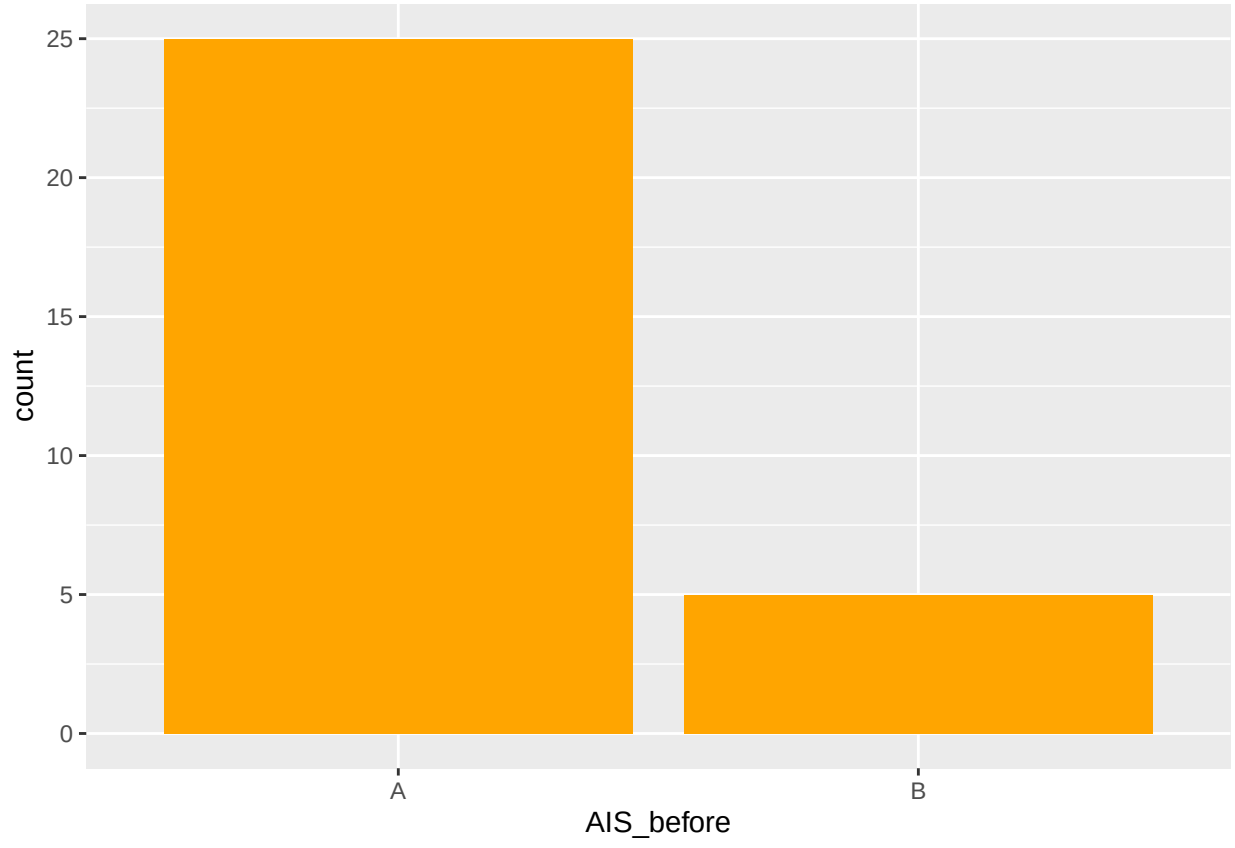


Figure 3.1: ALS level distribution before treatment.

```
g3.2 = ggplot(data = data,
              mapping = aes(x = AIS_after))
g3.2 = g3.2 + geom_bar(fill = "purple")
g3.2
```

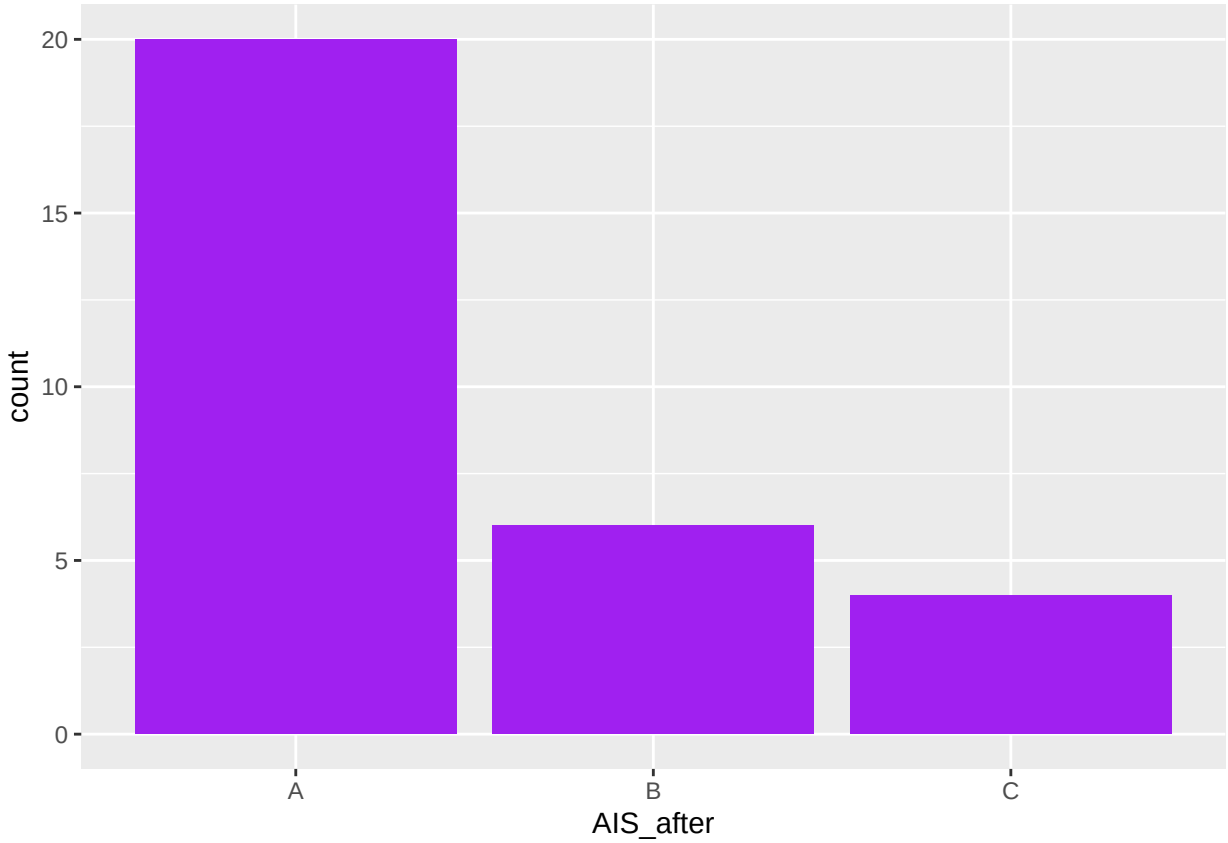


Figure 3.1: ALS level distribution after treatment.

3.3 Formulate the correct statistical hypothesis to compare the groups, choose the appropriate statistical test

- We first form the null (H0) and alternative (HA) hypothesis for this question.
- H0: The AIS score after treatment is no better than that before treatment.
- HA: The AIS score after treatment is better than that before treatment.
- Because the sample size is too small, we cannot decide whether it is normally distributed.
- Therefore, We use paired Wilcoxon test.
- We convert AIS score A,B,C,D,E into 5, 4, 3, 2 and 1.

```
a = c()
for (i in 1:nrow(data)) {
  x = data[i, "AIS_before"]
  if (x == "A")
    t = 5
  if (x == "B")
    t = 4
  if (x == "C")
    t = 3
  if (x == "D")
    t = 2
  if (x == "E")
```

```
    t = 1
  a = c(a, t)
}
#print(a)
b = c()
for (i in 1:nrow(data)) {
  x = data[i, "AIS_after"]
  if (x == "A")
    t = 5
  if (x == "B")
    t = 4
  if (x == "C")
    t = 3
  if (x == "D")
    t = 2
  if (x == "E")
    t = 1
  b = c(b, t)
}
wilcox.test(a, b, alternative = 'less', paired = T)
```

```
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  a and b
## V = 41, p-value = 0.9912
## alternative hypothesis: true location shift is less than 0
```

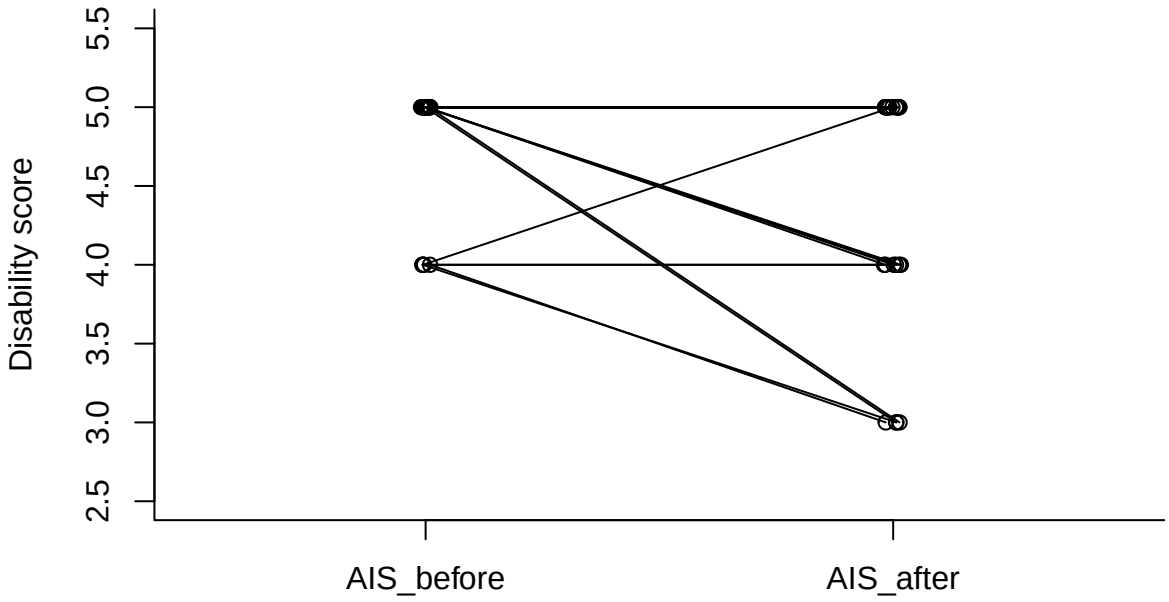
- p-value < 0.05
- We reject H0.
- There is sufficient evidence to conclude that the AIS score after treatment is better than that before treatment.

```
data_to_plot = cbind(rep(1, nrow(data)),a,rep(2, nrow(data)),b)
data_to_plot = as.data.frame(data_to_plot)
names(data_to_plot) = c("AIS_before", "x0", "AIS_after", "x1")
# str(data_to_plot)
data_to_plot[, c(1, 3)] = apply(data_to_plot[, c(1, 3)], MARGIN = c(1, 2), FUN = jitter, factor = 0.5)
# MARGIN = for a matrix 1 indicates rows, 2 indicates columns, c(1, 2) indicates rows and columns
data_to_plot
```

```
##    AIS_before x0 AIS_after x1
## 1   1.0001530  5  2.006037  4
## 2   0.9974535  5  2.001287  4
## 3   0.9939644  5  2.006197  5
## 4   1.0096883  5  1.983194  5
## 5   0.9967711  4  2.012714  4
## 6   0.9977320  5  2.009045  5
## 7   0.9902508  5  1.991249  5
## 8   1.0041466  5  1.987410  5
## 9   1.0011054  5  1.983496  4
## 10  1.0095075  4  1.984089  3
```

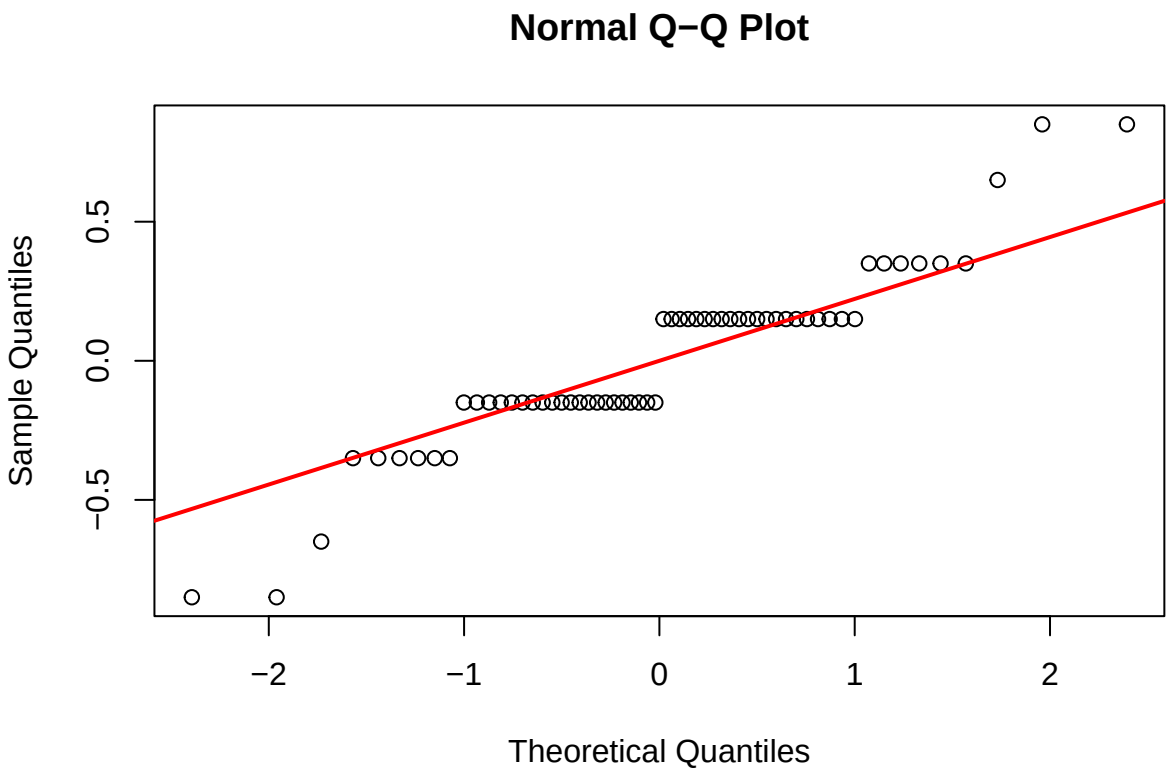
```
## 11  0.9927163  4  1.980216  4
## 12  1.0071134  5  2.006374  5
## 13  0.9950562  5  2.011349  5
## 14  1.0033615  5  1.990587  5
## 15  0.9938917  4  2.006894  3
## 16  1.0077219  5  1.986659  5
## 17  1.0050029  5  2.012896  5
## 18  1.0099905  5  2.010209  5
## 19  1.0066867  5  2.013654  3
## 20  0.9943013  4  1.998434  5
## 21  0.9976447  5  2.009213  5
## 22  1.0036036  5  2.013278  5
## 23  0.9995317  5  1.982961  5
## 24  0.9937778  5  1.991560  5
## 25  1.0056466  5  2.011706  5
## 26  0.9986607  5  2.016471  4
## 27  1.0001049  5  2.009151  5
## 28  0.9988427  5  2.006213  3
## 29  1.0092560  5  1.982557  5
## 30  0.9913011  5  1.986694  5
```

```r
plot(x = data_to_plot[ , 1], y = data_to_plot[ , 2],
     xlim = c(0.5, 2.5), ylim = c(2.5, 5.5),
     axes = F, xlab = "", ylab = "Disability score")
# add the point of AIS before AND hide the axis
points(x = data_to_plot[ , 3], y = data_to_plot[ , 4])
# add the point of AIS after
segments(x0 = data_to_plot[ , 1],
         y0 = data_to_plot[ , 2],
         x1 = data_to_plot[ , 3],
         y1 = data_to_plot[ , 4])
axis(2) # add y axis
axis(1, at = 1:2, labels = c("AIS_before", "AIS_after"))
# only draw the x axis between (1, 2)
box(bty = "l")
```




```
# bty = L means the frame looks like the letter 'L'
# add a box around the plot, the aim is to complete the x axis
```

```
row.names(data) = 1:nrow(data)
data$AIS_before = a
data$AIS_after = b
data = gather(data, key = "measurement", value = "AIS", AIS_before, AIS_after)
data$patient_ID = as.factor(data$patient_ID)
model = lm(AIS ~ measurement + patient_ID, data) #
# summary(model)
# shapiro.test(resid(model))
# use q-q plot (quantile-quantile plot) to validate normality.
qqdata = resid(model)
# summary(qqdata)
qqnorm(qqdata)
qqline(qqdata, col = "red", lwd = 2) # add a reference line
```




```
xlab("Theoretical Quantiles")
```

```
## $x
## [1] "Theoretical Quantiles"
##
## attr(,"class")
## [1] "labels"
```

```
ylab("Ordered Values")
```

```
## $y
## [1] "Ordered Values"
##
## attr(,"class")
## [1] "labels"
```

- Normality check shows that the residuals from a linear regression of the data are not normally distributed.
- Because most points are not tightly cluster around the reference line.

3.4 Discuss the results you got.

- The treatment has significant improvement effect on the spinal cord injury.

- The effect size.

```
effect_size = abs(mean(b)-mean(a))
effect_size
```

```
## [1] 0.3
```

- We should consider more factors that influence the AIS score.
- We should consider to use a quantitative score to evaluate the health condition instead of AIS score.