# ADS2 Coding Challenge 1

0049

2022-1-5

```
library(tidyverse)
```

## 1. Neuroanatomy

### Import the data set. Convert it to a long format.

```
Neuron_data <- read.csv(
  "D:/R_document/ADS_practical/ADS_final1/AIS_lengths.csv")
head(Neuron_data)
```

```
##   animal genotype pyramidal1 pyramidal2 pyramidal3 pyramidal4 pyramidal5
## 1   3121       WT       18.6       25.6       31.0         39         44
## 2   7684       wt       26.6       26.2       25.2         21         13
## 3   9770   mutant        6.2       25.8       -3.8          8          9
## 4   3437   mutant       17.8       39.8       24.1         29         36
## 5   4166   mutant       30.3       23.1       26.8         20         35
## 6   6701   mutant       26.8       22.9       30.5        430         15
```

The data is imported successfully. Next I will convert it to a long format.

```
Neuron_long <- gather(
  Neuron_data, "pyramidal1", "pyramidal2", "pyramidal3", "pyramidal4", "pyramidal5",
  key = "pyramidal",
  value = "AIS_length")
head(Neuron_long)
```

```
##   animal genotype  pyramidal AIS_length
## 1   3121       WT pyramidal1       18.6
## 2   7684       wt pyramidal1       26.6
## 3   9770   mutant pyramidal1        6.2
## 4   3437   mutant pyramidal1       17.8
## 5   4166   mutant pyramidal1       30.3
## 6   6701   mutant pyramidal1       26.8
```

```
tail(Neuron_long)
```

```
##     animal genotype  pyramidal AIS_length
## 145   4495   mutant pyramidal5         38
## 146   8764   mutant pyramidal5         14
## 147   3467   mutant pyramidal5         28
## 148   5209       wt pyramidal5         NA
## 149   4678   mutant pyramidal5         NA
## 150   5070       wt pyramidal5         15
```

The new data is now in a **long** format.

### Explain briefly what "long format" means.

The long format has one observation and one measurement per row. That is, many rows can constitute a whole observation. In this data set, usually 5 rows can constitute a whole observation, which is the AIS_length.

### Clean the data. Explain the steps you are taking for data cleaning.

First, look for any NA value.

```
Neuron_long <- na_if(Neuron_long, "")
anyNA(Neuron_long)
```

```
## [1] TRUE
```

So the data set has some missing values.

```
Neuron_noNA <- na.omit(Neuron_long)
anyNA(Neuron_noNA)
```

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

Now the data set has no missing values.
Next I want to find some duplication.

```
which(duplicated(Neuron_noNA))
```

```
## integer(0)
```

Very good, no duplicated values.
Last, I want to find some strange patterns.

```
table(Neuron_noNA$animal)
```

```
##
## 1085 1810 2749 3121 3342 3354 3437 3467 3503 4166 4495 4568 4678 4721 5070 5209
##    5    5    5    5    5    4    5    5    5    5    5    5    4    5    5    2
## 6158 6399 6477 6607 6701 6961 7042 7155 7684 8764 9050 9278 9770
##    5    5    5    5    5    5    5    5    5    5    5    3    5
```

```
table(Neuron_noNA$genotype)
```

```
##
##    mut mutant     wt     WT
##      3     63     67      5
```

The name of genotype is not very clean.

```
Neuron_noNA[Neuron_noNA$genotype == "mut",]$genotype <- "mutant"
Neuron_noNA[Neuron_noNA$genotype == "WT",]$genotype <- "wt"
table(Neuron_noNA$genotype)
```

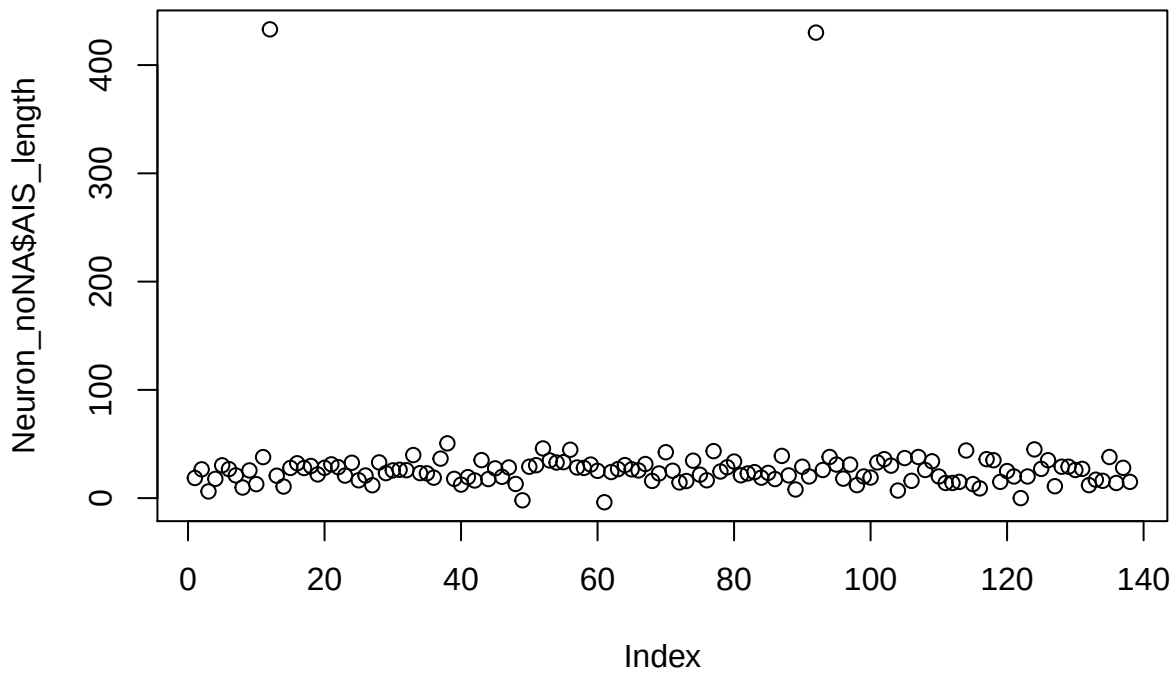
```
##
## mutant     wt
##     66     72
```

It is clean now.

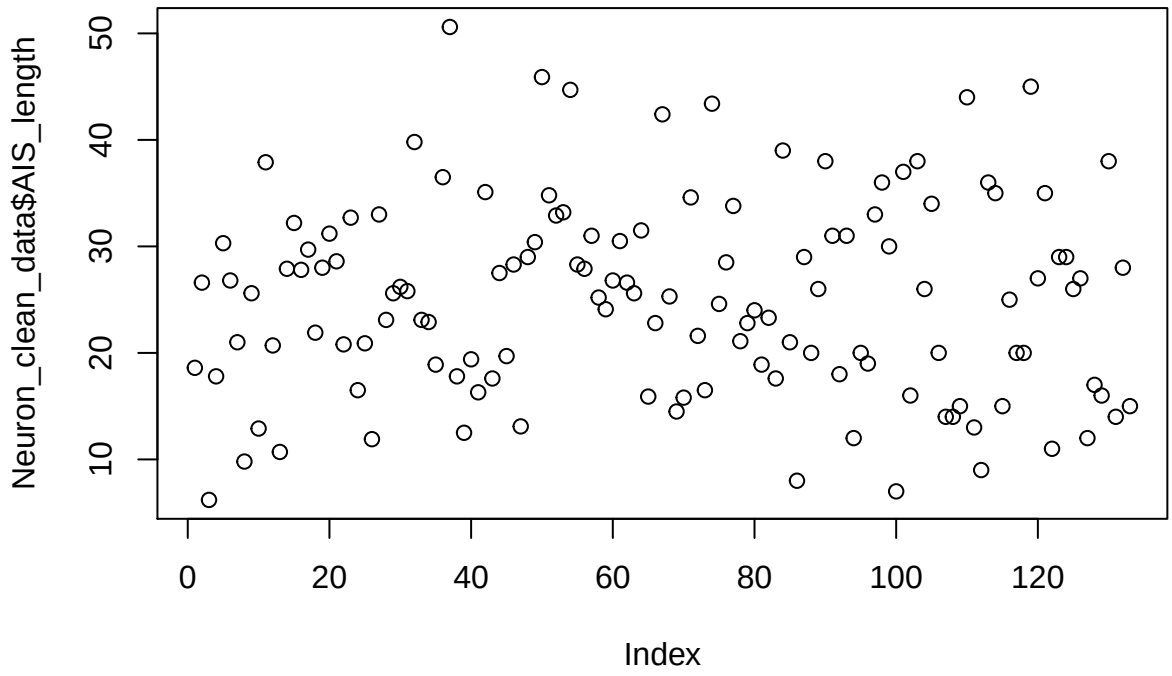
```
table(Neuron_noNA$pyramidal)
```

```
##
## pyramidal1 pyramidal2 pyramidal3 pyramidal4 pyramidal5
##         29         29         28         27         25
```

```
plot(Neuron_noNA$AIS_length)
```



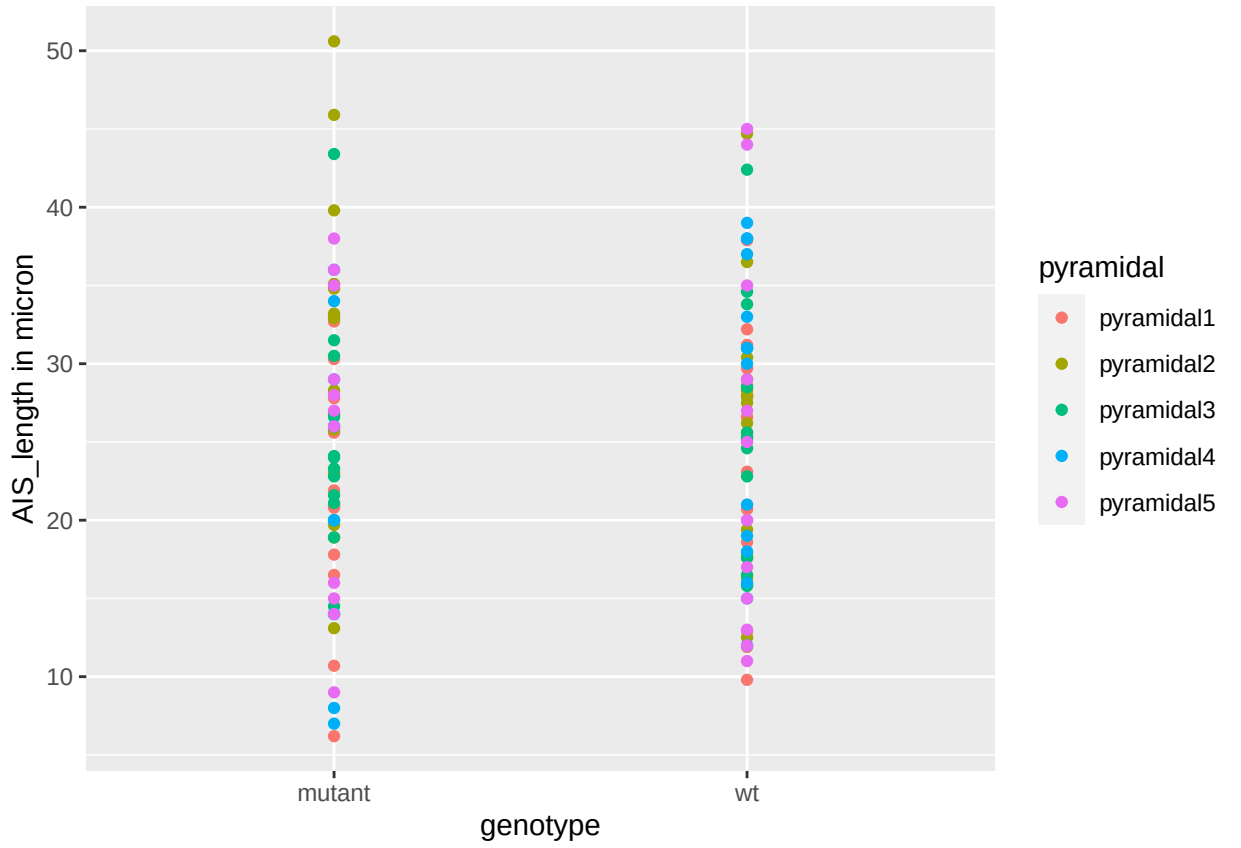
```
Neuron_clean_data <- Neuron_noNA[Neuron_noNA$AIS_length>0 & Neuron_noNA$AIS_length < 100, ]
plot(Neuron_clean_data$AIS_length)
```



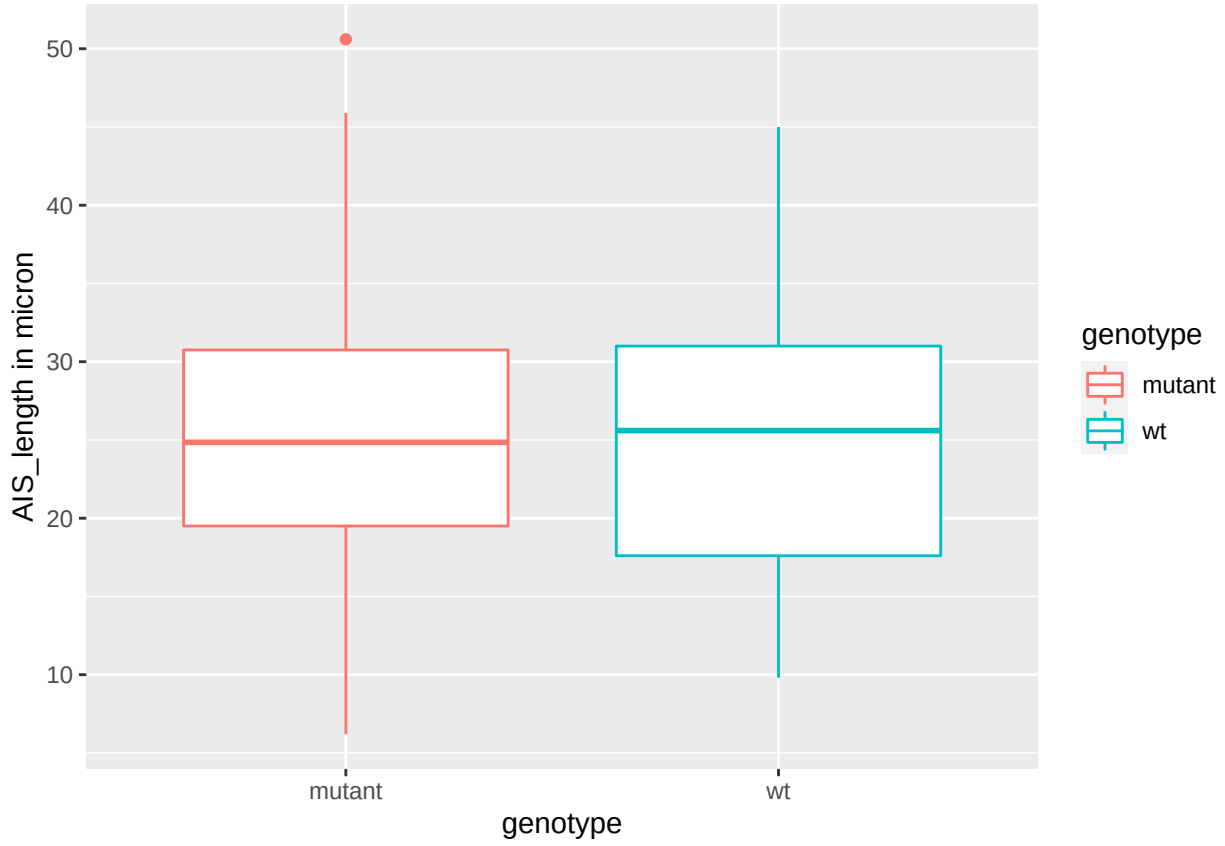

Some data for AIS_length are strange. So I delete them. Now the data is clean.

### Plot the data in a useful way.

```
ggplot(Neuron_clean_data, aes(x = genotype, y = AIS_length)) +
  geom_point(aes(color = pyramidal)) +
  labs(y = "AIS_length in micron")
```




```
ggplot(Neuron_clean_data, aes(x = genotype, y = AIS_length)) +
  geom_boxplot(aes(color = genotype)) +
  labs(y = "AIS_length in micron")
```




## 2. Hospital patients

### Import the data and plot them in a useful way.

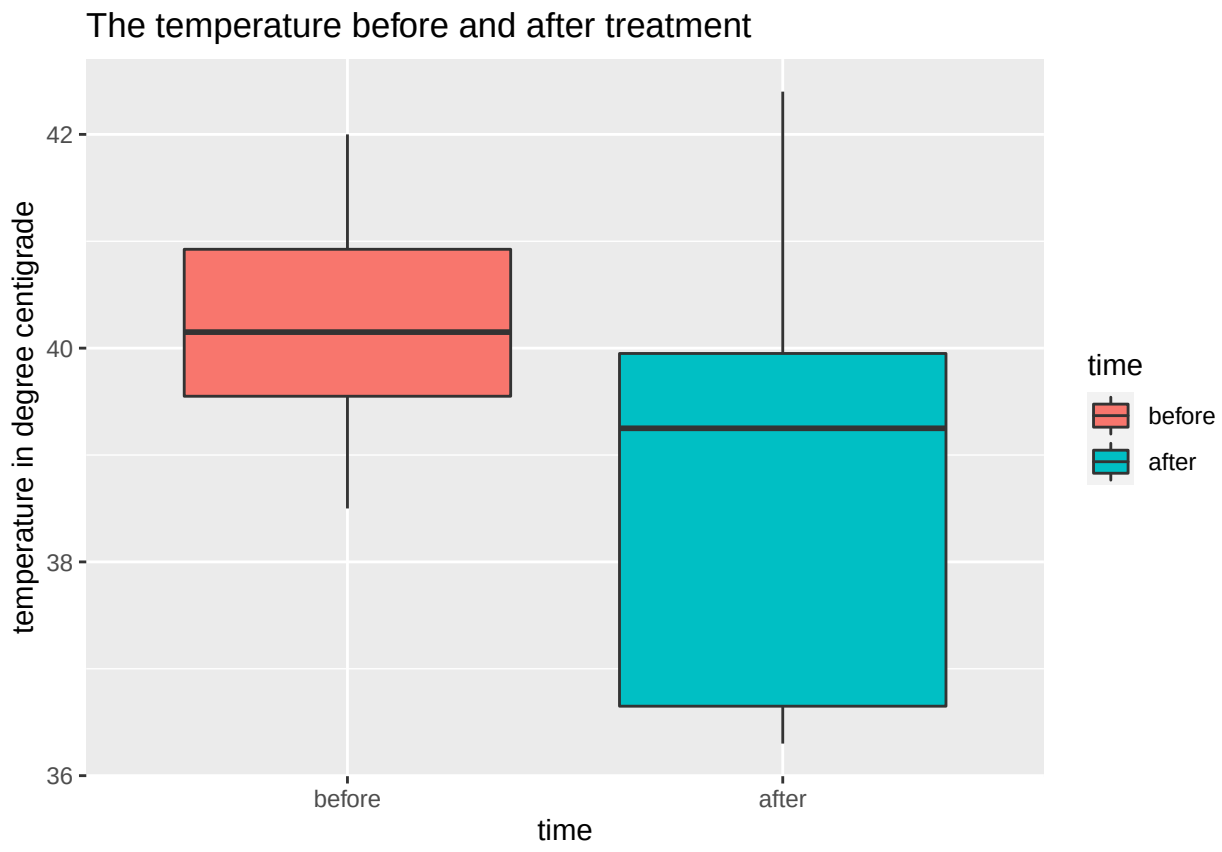
```
fever_data <- read.csv("D:/R_document/ADS_practical/ADS_final1/fever_data.csv")
head(fever_data)
```

```
##   patient_ID time_1 time_2
## 1        243   42.0   36.3
## 2        635   40.6   39.3
## 3        231   41.3   40.3
## 4        263   39.4   36.7
## 5        193   38.5   40.2
## 6        538   39.7   39.7
```

The data is successfully imported. First I convert the data set into a long format and make the description more clear.
Then, I try to draw a boxplot.

```
fever_wide <- gather(fever_data, "time_1", "time_2", key = "time", value = "temperature")
fever_wide[fever_wide$time == "time_1",]$time <- "before"
fever_wide[fever_wide$time == "time_2",]$time <- "after"
```

```
fever_wide$time <- factor(fever_wide$time, levels = c("before", "after"))
ggplot(data = fever_wide, aes(x = time, y = temperature)) +
  geom_boxplot(aes(fill = time)) +
  labs(title = "The temperature before and after treatment",
       y = "temperature in degree centigrade")
```




### Is the new treatment successful at reducing body temperature within six hours?

To test whether the treatment is successful or not, statistical tests are needed. First I extract subsets from the raw data.
The **Null hypothesis** is that the temperature after the treatment is not lower than before. So the new treatment is not successful.
The **Alternative hypothesis** is that the temperature after the treatment is lower than before. So the new treatment is successful.

```
before <- fever_wide[fever_wide$time == "before",]$temperature
after <- fever_wide[fever_wide$time == "after",]$temperature
```

Second, I want to know whether the samples are normally distributed.

The **Null hypothesis** is that the samples are normally distributed.
The **Alternative hypothesis** is that the samples are not normally distributed.

```
shapiro.test(before)$p
```

```
## [1] 0.4180528
```

```
shapiro.test(after)$p
```

```
## [1] 0.004580259
```

For the after treatment group, the p-value is much less than 0.05. So I reject the H0 hypothesis. The data of the after treatment group is not normally distributed. Therefore, I choose nonparametric test rather than parametric test.

```
wilcox.test(before, after, exact = F)$p.value
```

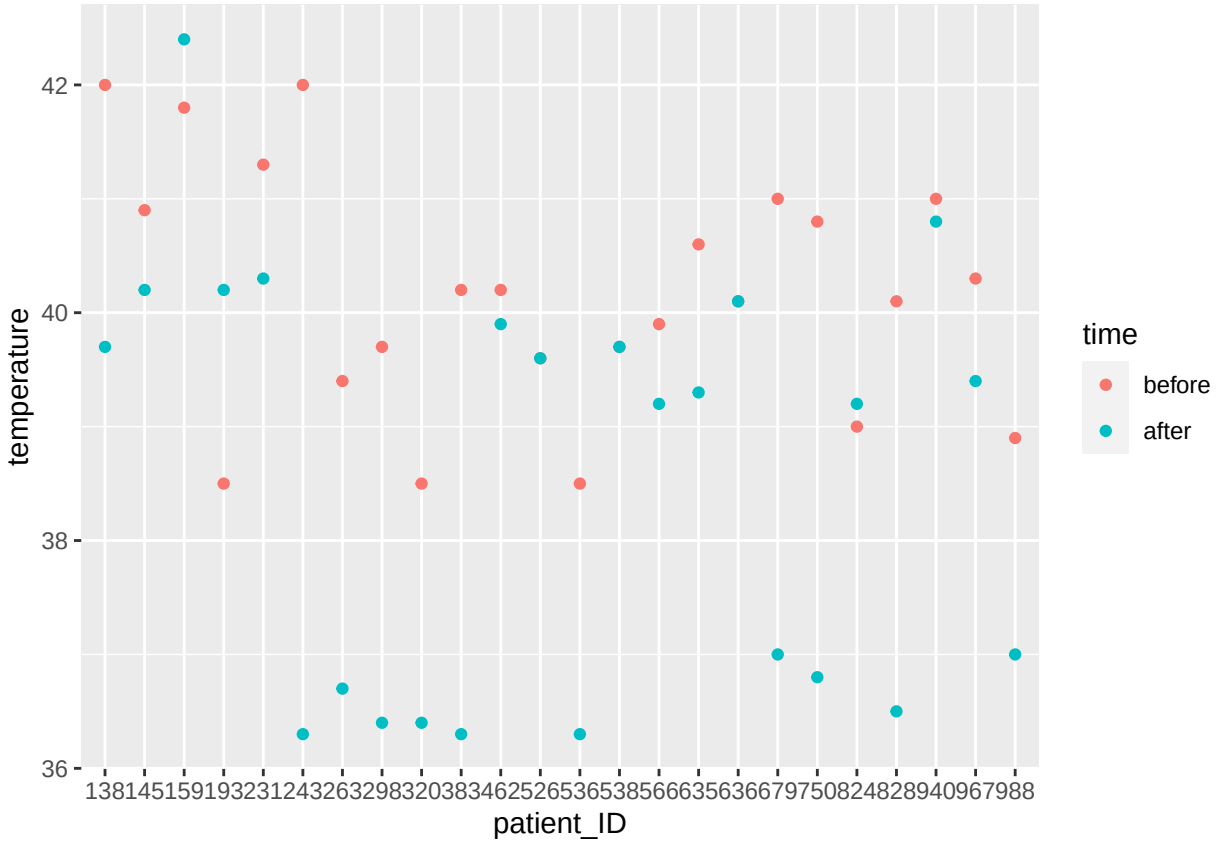
```
## [1] 0.003157465
```

The p-value of wilcox test is about 0.003, much less than 0.05. So I reject the H0 hypothesis. Also from the boxplot, I know that the overall temperature is lower in after treatment group. So, the new treatment **is successful** at reducing body temperature within six hours.

### One of the doctors suspects that the new fever treatment may work for some patients, but not for others. Would you agree with that suggestion?

I draw the points of temperatures before and after treatment for each patients.

```
fever_wide$patient_ID <- factor(fever_wide$patient_ID)
ggplot(data = fever_wide, aes(x = patient_ID, y = temperature)) +
  geom_point(aes(color = time))
```




I agree with that suggestion since some patients have higher temperature after treatment such as No.159 and No.824. So the treatment may work for some patients, not all.

### What could be done to follow up on this study?

First, I test whether the sample size is enough big to have sufficient power and decrease the type2 error.

```
mean <- abs(mean(after)-mean(before))
sd(after)
```

```
## [1] 1.846378
```

```
sd(before)
```

```
## [1] 1.061855
```

```
# The sd(after) is larger.
power.t.test(delta = mean, sd = sd(after),
             sig.level = 0.05, power = 0.8,
             type = "paired", alternative = "one.sided")$n
```

```
## [1] 9.779405
```

We should choose 10 patients. Therefore, the sample size is enough for the power. However the temperature is not normally distributed, so parametric tests can not be used. I will choose more patients to make the data more normally distributed and then use paired-t.test, which is better than wilcox test. Additionally, only the temperature within 6 hours after treatment is not enough, I will add a measurement to test all patients' temperatures 1 day after the treatment to make the conclusion more convinced.

## 3. Fly genetics

### If we irradiate 14 000 flies using the mutagenesis protocol, approximately what proportion of the genes will have been targeted?

```
sample_1 <- sample(1:14000, 14000, replace = T)
length(unique(sample_1))
```

```
## [1] 8770
```

About 8770 genes will have been targeted.

### If we irradiate 100 000 flies, what are the chances that we have targeted every gene at least once?

```
count1 = 0
for (i in 1:1000){
  sample_2 <- sample(1:14000, 100000, replace = T)
  if (length(unique(sample_2)) == 14000) {
    count1 = count1 + 1
  }
}
count1/1000
```

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

The chance is 0.

### We would like at least a 90 % chance of targeting every gene at least once. What should we do to determine the number of flies we need to irradiate to achieve this? (Note that you are not asked to implement the solution here, but you should provide text or pseudocode to explain how you would do it.)

Here are my solutions: For each possible number, I will simulate 10000 times. For each time, I choose x genes from 14000 genes with replace = T. Then I use length(unique(x)) to determine the number of unique genes. If the number is above 90% of the whole gene, I will count this as an effective procedure. In 10000 times, if the number of effective procedure is large enough (nearly 10000), I will ensure that x is the number of files I need to irradiate. If the number of effective procedure is not large enough, I will increase x value and do another simulation until find the appropriate x value.

**Looking at the assumptions above, do you think they are realistic? Why or why not? What does that mean for the numbers you computed in response to the questions?**

I do not think they are realistic. In this cases, we assumed that the procedure is 100 % effective. However, we may get many different mutations or no mutation in one procedure. We also assumed that each gene is of equal size, and equally likely to be targeted by the procedure. However, in reality, genes have different lengths so some genes are prone to be mutated and some are not.
The numbers I computed in response to the questions are only in idealization, so they may be very different from the real numbers.