Univariate statistics for Legionnaires disease

Data source

This program uses data from a fictional study of Legionnaires disease and produces some simple univariate statistics: means, standard deviations, and percentiles. There is a data dictionary that provides more details about the data.

Libraries

Here are the libraries you need for this program.

```
        library(tidyverse)
```

Reading the data

Here is the code to read the data and show a glimpse. There are 31 columns total, but I am showing just a few of the columns here.

```
        fn <- "../data/legionnaires-disease.txt"
        ld_raw_data <- read_tsv(fn, col_types="cnn")
        glimpse(ld_raw_data)
```

```
Rows: 8
Columns: 3
$ `Room Number` <chr> "121", "163", "125", "264", "233", "218", "324", "325"
$ Before        <dbl> 11.8, 8.2, 7.1, 14.0, 10.8, 10.1, 14.6, 14.0
$ After         <dbl> 10.1, 7.2, 3.8, 12.0, 8.3, 10.5, 12.1, 13.7
```

Rename, 1

Notice how R encloses the first variable name (Room Number) in back-quotes. This is needed when a variable includes an embedded blank. You should rename this variable at your first opportunity.

```
        names(ld_raw_data)[1] <- "Room_Number"
        glimpse(ld_raw_data)
```

```
Rows: 8
Columns: 3
$ Room_Number <chr> "121", "163", "125", "264", "233", "218", "324", "325"
$ Before      <dbl> 11.8, 8.2, 7.1, 14.0, 10.8, 10.1, 14.6, 14.0
$ After       <dbl> 10.1, 7.2, 3.8, 12.0, 8.3, 10.5, 12.1, 13.7
```

Rename, 2

I find that many of the mistakes that I make are due to inconsistencies in how I name variables. Capitalization is one of the biggest problems. So I have gotten into the habit of converting variable names to all lower case. That way I don't have to worry about whether it is "Before" or "before". Here is the code to convert every capital letter to a lowercase letter.

```
        names(ld_raw_data) <- tolower(names(ld_raw_data))
        glimpse(ld_raw_data)
```

```
Rows: 8
Columns: 3
$ room_number <chr> "121", "163", "125", "264", "233", "218", "324", "325"
$ before      <dbl> 11.8, 8.2, 7.1, 14.0, 10.8, 10.1, 14.6, 14.0
$ after       <dbl> 10.1, 7.2, 3.8, 12.0, 8.3, 10.5, 12.1, 13.7
```

Calculate means and standard deviations before remediation

```
        ld_raw_data |>
          summarize(
```

```
            before_mn=mean(before),
            before_sd=sd(before))
```

```
# A tibble: 1 × 2
  before_mn before_sd
      <dbl>     <dbl>
1      11.3      2.79
```

The average colony count per cubic foot before remediation, 11.3, is quite large. The standard deviation, 2.8, represents a moderate amount of variation in this variable.

Calculate means and standard deviations after remediation

```
        ld_raw_data |>
          summarize(
            after_mn=mean(after),
            after_sd=sd(after))
```

```
# A tibble: 1 × 2
  after_mn after_sd
     <dbl>    <dbl>
1     9.71     3.18
```

The average colony count per cubic foot after remediation, 9.7, is still quite large. The standard deviation, 3.2, represents a moderate amount of variation in this variable and is roughly comparable to the variation before remediation.

Calculate median and range before intervention

You could also use “median(before)” and “min(before)” and “max(before)” in the code below.

```
        ld_raw_data |>
          summarize(
            before_median=quantile(before, probs=0.5),
```

```
            before_min=quantile(before, probs=0),
            before_max=quantile(before, probs=1))
```

```
# A tibble: 1 × 3
  before_median before_min before_max
          <dbl>      <dbl>      <dbl>
1          11.3        7.1       14.6
```

The median colony count before remediation, 11.3, is roughly the same as the mean. The data ranges from 7.1 to 14.6 colonies per cubic centimeter, a fairly wide range.

Calculate median and range after intervention

```
        ld_raw_data |>
          summarize(
            after_median=quantile(after, probs=0.5),
            after_min=quantile(after, probs=0),
            after_max=quantile(after, probs=1))
```

```
# A tibble: 1 × 3
  after_median after_min after_max
         <dbl>     <dbl>     <dbl>
1         10.3       3.8      13.7
```

The median colony count, 10.3, is slightly lower after remediation. The data range from 3.8 to 13.7 colonies per cubic centimeter and is about as wide as the range before remediation.

Additional comments

The names that you choose for the left hand side of the equal sign are arbitrary. You should choose a descriptive name, but you have lots of options. A median of the before and after values could be called

- Before_median, After_median
- Median0, Median1
- Second_quartile_A, Second_quartile_B

- or many other reasonable choices.

Calculate a change score

For data like this with two measurements before and after an intervention, you should compute a change score. The way the computations are done below, a positive value means a reduction in colony counts. Note that any time you make a major change in a dataset, you should save it with a different name. That makes it easier for you to back up if you end up going down a blind alley.

```
ld_raw_data |>
  mutate(change=before-after) -> ld_change_scores
glimpse(ld_change_scores)
```

```
Rows: 8
Columns: 4
$ room_number <chr> "121", "163", "125", "264", "233", "218", "324", "325"
$ before      <dbl> 11.8, 8.2, 7.1, 14.0, 10.8, 10.1, 14.6, 14.0
$ after       <dbl> 10.1, 7.2, 3.8, 12.0, 8.3, 10.5, 12.1, 13.7
$ change      <dbl> 1.7, 1.0, 3.3, 2.0, 2.5, -0.4, 2.5, 0.3
```

Calculate mean and standard deviation for the change in bacterial counts. Interpret the results

```
ld_change_scores |>
  summarize(
    change_mn=mean(change),
    change_sd=sd(change))
```

```
# A tibble: 1 × 2
  change_mn change_sd
      <dbl>     <dbl>
1      1.61      1.24
```

The average colony count per cubic foot the change, 1.61, is still small. The standard deviation, 1.2, represents a small amount of variation between the rooms.

Calculate median and range for the change in bacterial counts.

```
          ld_change_scores |>
            summarize(
              change_median=quantile(change, probs=0.5),
              change_min=quantile(change, probs=0),
              change_max=quantile(change, probs=1))
```

```
# A tibble: 1 × 3
  change_median change_min change_max
          <dbl>      <dbl>      <dbl>
1          1.85     -0.400        3.3
```

Then median colony count for the change in bacterial counts, 1.9, is very similar to the mean. The range of counts is between -0.4 to 3.3, which demonstrates a small variation in the change in colony counts between all the rooms.