Introduction to R, module06

Steve Simon

Created 2020-04-04

Introduction

```
suppressMessages(
  suppressWarnings(
    library(tidyverse)))
R.version.string
## [1] "R version 4.1.1 (2021-08-10)"
Sys.Date()
## [1] "2022-05-09"
```

What is longitudinal data

- Definition
  - Measurements taken at different times
- Closely related datasets
  - Crossover
  - Pre-test/post-test
  - Repeated measures
  - Split plot

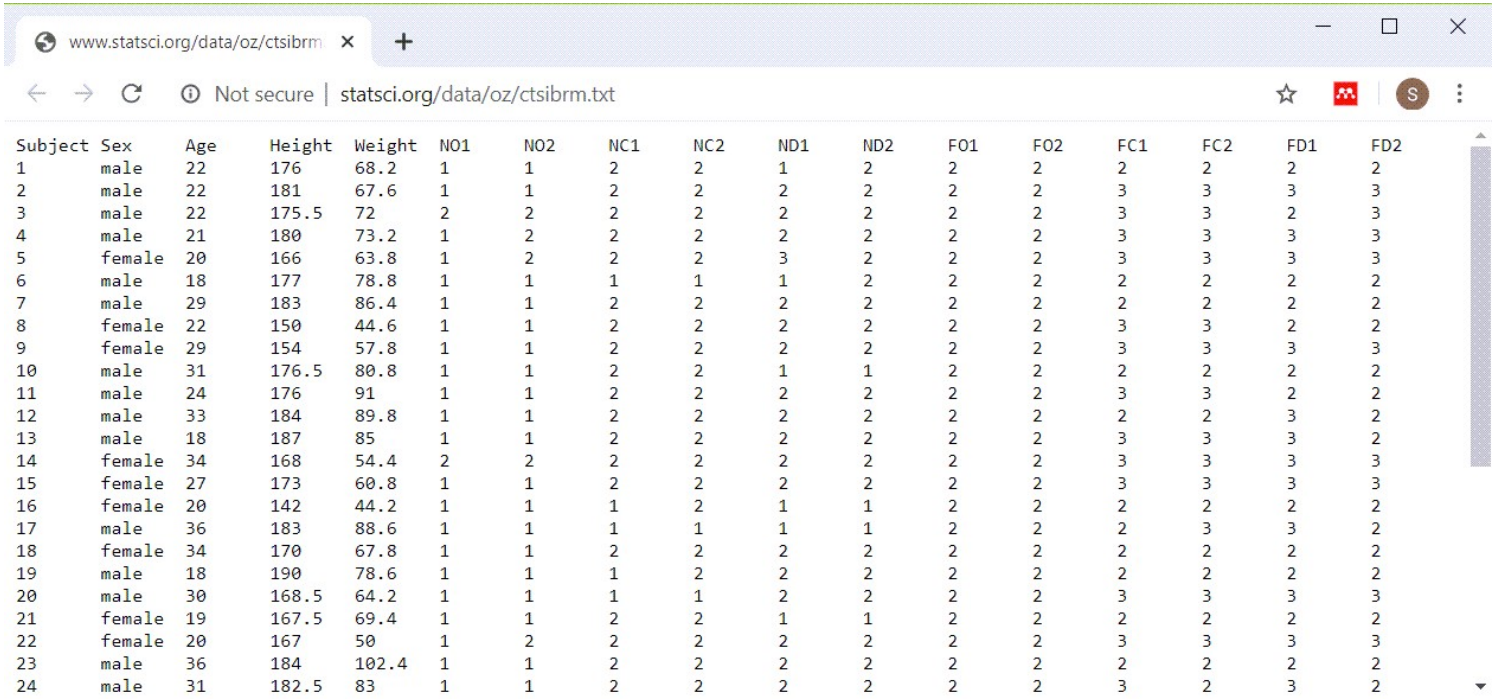
Two formats for longitudinal data

- Short and fat format
  - Many columns
  - Not so many rows
- Tall and thin format
  - Not so many columns
  - Many rows

Example: effect of surface and vision on balance

- Repeated measures experiment
  - Vision has 3 levels
    - Eyes open, eyes closed, dome
  - Surface has 2 levels
    - Normal or foam
  - Two replications of each format
  - 40 subjects, 3*2*2=12 measurements

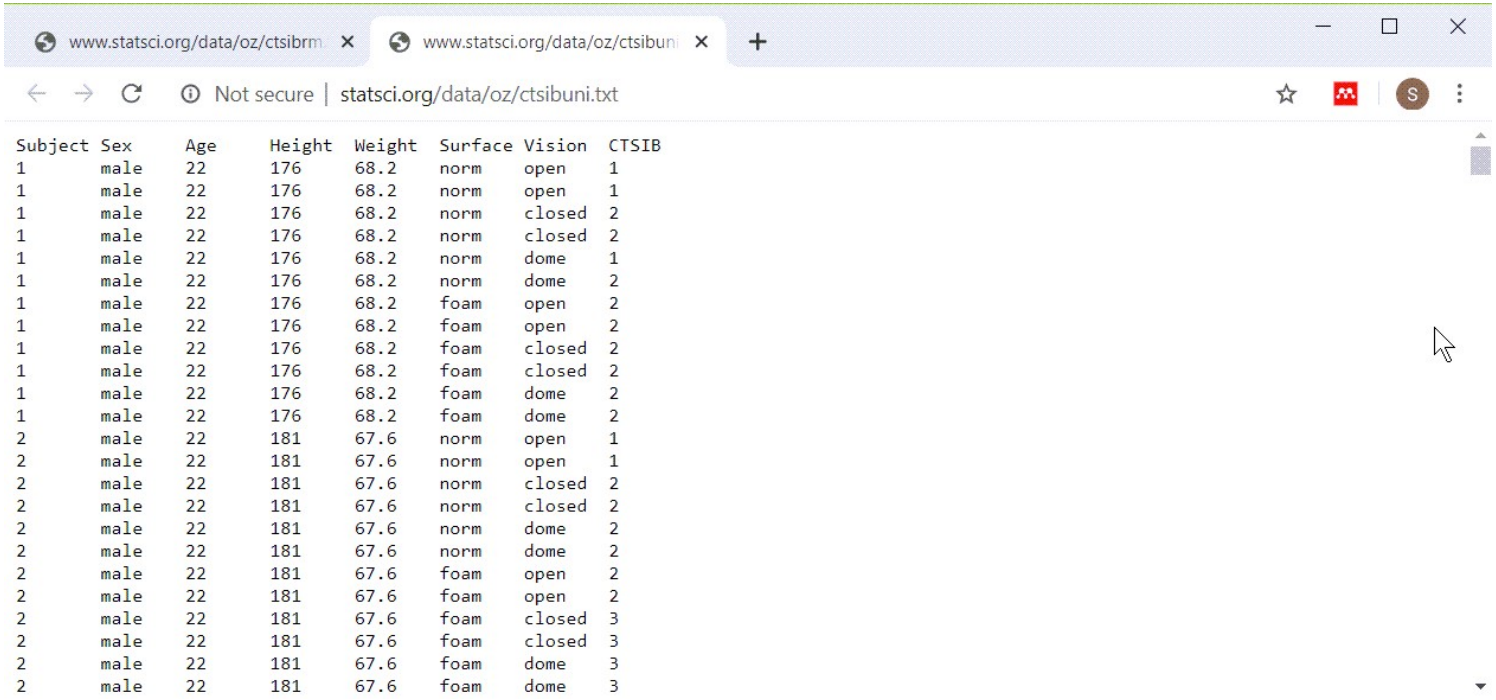
Short and fat example



Subject	Sex	Age	Height	Weight	NO1	NO2	NC1	NC2	ND1	ND2	FO1	FO2	FC1	FC2	FD1	FD2
1	male	22	176	68.2	1	1	2	2	1	2	2	2	2	2	2	2
2	male	22	181	67.6	1	1	2	2	2	2	2	2	3	3	3	3
3	male	22	175.5	72	2	2	2	2	2	2	2	2	3	3	2	3
4	male	21	180	73.2	1	2	2	2	2	2	2	2	3	3	3	3
5	female	20	166	63.8	1	2	2	2	3	2	2	2	3	3	3	3
6	male	18	177	78.8	1	1	1	1	1	2	2	2	2	2	2	2
7	male	29	183	86.4	1	1	2	2	2	2	2	2	2	2	2	2
8	female	22	150	44.6	1	1	2	2	2	2	2	2	3	3	2	2
9	female	29	154	57.8	1	1	2	2	2	2	2	2	3	3	3	3
10	male	31	176.5	80.8	1	1	2	2	1	1	2	2	2	2	2	2
11	male	24	176	91	1	1	2	2	2	2	2	2	3	3	2	2
12	male	33	184	89.8	1	1	2	2	2	2	2	2	2	2	3	2
13	male	18	187	85	1	1	2	2	2	2	2	2	3	3	3	2
14	female	34	168	54.4	2	2	2	2	2	2	2	2	3	3	3	3
15	female	27	173	60.8	1	1	2	2	2	2	2	2	3	3	3	3
16	female	20	142	44.2	1	1	1	2	1	1	2	2	2	2	2	2
17	male	36	183	88.6	1	1	1	1	1	1	2	2	2	3	3	2
18	female	34	170	67.8	1	1	2	2	2	2	2	2	2	2	2	2
19	male	18	190	78.6	1	1	1	2	2	2	2	2	2	2	2	2
20	male	30	168.5	64.2	1	1	1	1	2	2	2	2	3	3	3	3
21	female	19	167.5	69.4	1	1	2	2	1	1	2	2	2	2	2	2
22	female	20	167	50	1	2	2	2	2	2	2	2	3	3	3	3
23	male	36	184	102.4	1	1	2	2	2	2	2	2	2	2	2	2
24	male	31	182.5	83	1	1	2	2	2	2	2	2	3	2	3	2

Longitudinal data stored one row per subject

Tall and thin example



Subject	Sex	Age	Height	Weight	Surface	Vision	CTSIB
1	male	22	176	68.2	norm	open	1
1	male	22	176	68.2	norm	open	1
1	male	22	176	68.2	norm	closed	2
1	male	22	176	68.2	norm	closed	2
1	male	22	176	68.2	norm	dome	1
1	male	22	176	68.2	norm	dome	2
1	male	22	176	68.2	foam	open	2
1	male	22	176	68.2	foam	open	2
1	male	22	176	68.2	foam	closed	2
1	male	22	176	68.2	foam	closed	2
1	male	22	176	68.2	foam	dome	2
1	male	22	176	68.2	foam	dome	2
2	male	22	181	67.6	norm	open	1
2	male	22	181	67.6	norm	open	1
2	male	22	181	67.6	norm	closed	2
2	male	22	181	67.6	norm	closed	2
2	male	22	181	67.6	norm	dome	2
2	male	22	181	67.6	norm	dome	2
2	male	22	181	67.6	foam	open	2
2	male	22	181	67.6	foam	open	2
2	male	22	181	67.6	foam	closed	3
2	male	22	181	67.6	foam	closed	3
2	male	22	181	67.6	foam	dome	3
2	male	22	181	67.6	foam	dome	3

Longitudinal data stores with multiple rows per patient

Which format is better?

- Short and fat advantages:
  - easy to compute change scores
  - easy to examine correlations over time
  - easy to insure consistency of demographic data
- Short and fat disadvantages:
  - hard to read because of the excessive need to scroll left and right

Which format is better?

- Tall and thin advantages:
  - easy to plot longitudinal trends
  - less need for missing value codes
  - easy to read because most scrolling is up and down
- Tall and thin disadvantages
  - hard to maintain consistency of demographic variables

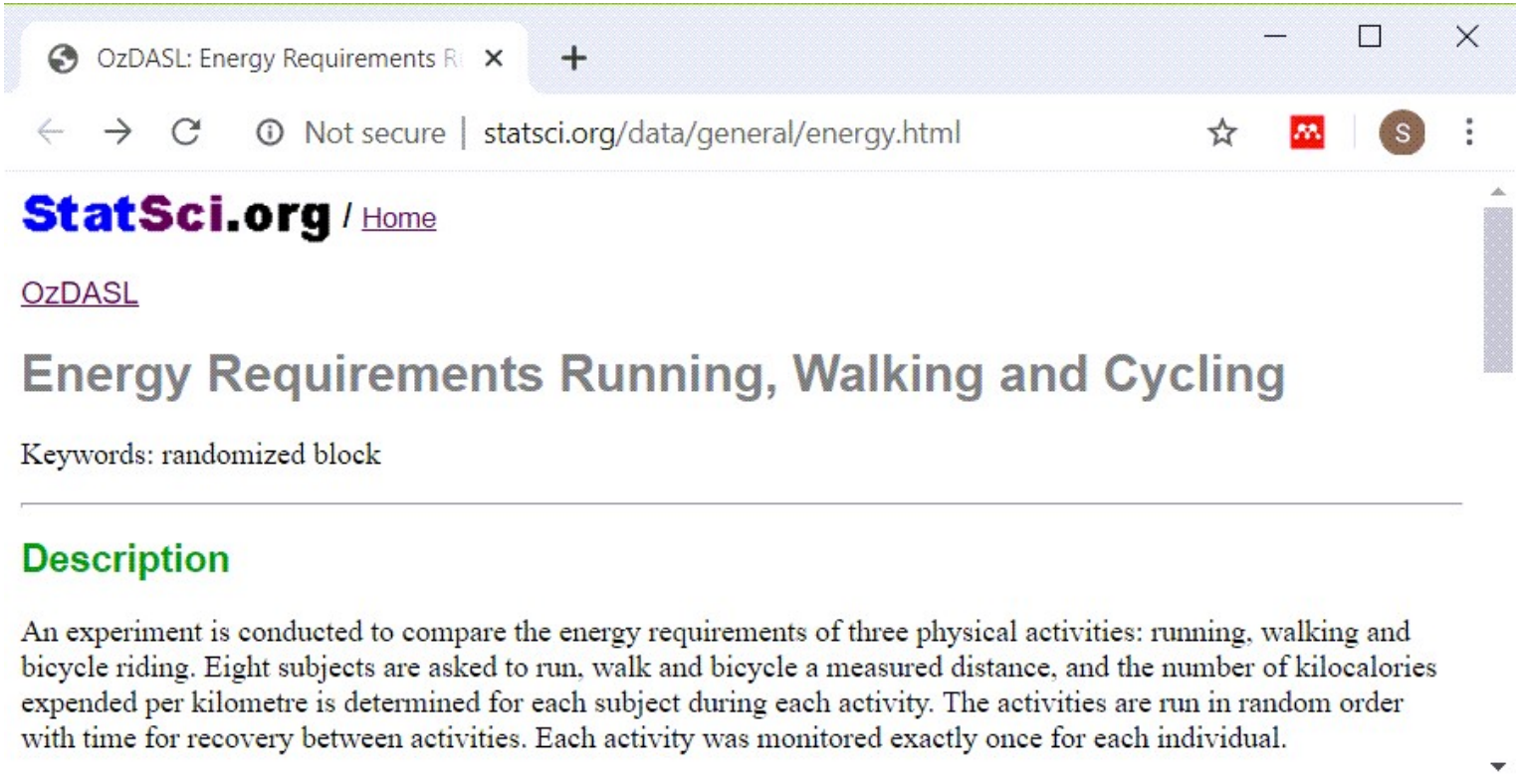
Break #1

- What have you learned
  - Two formats for longitudinal data
- What is coming next
  - Converting to tall and thin format

Energy dataset (short and fat format)

- Completely randomized block design
  - Blocks are subjects (8 total)
  - Treatment are exercise
    - Running, walking, cycling
    - There are 3 measurements per subject

Energy dataset (short and fat format)



Energy dataset (short and fat format)

Import energy dataset (short and fat format)

```
fi <- "../data/energy.txt"
en <- read_table(fi, col_types="nnnn")
```

Energy dataset, glimpse

```
glimpse(en)
## Rows: 8
## Columns: 4
## $ Subject <dbl> 1, 2, 3, 4, 5, 6, 7, 8
## $ Running <dbl> 1.4, 1.5, 1.8, 1.7, 1.~
## $ Walking <dbl> 1.1, 1.2, 1.3, 1.3, 0.~
## $ Cycling <dbl> 0.7, 0.8, 0.7, 0.8, 0.~
```

Converting to tall and thin, code

```
en_tall <-
  pivot_longer(en,
    c(Running,
      Walking,
      Cycling),
    names_to="activity",
    values_to="energy")
```

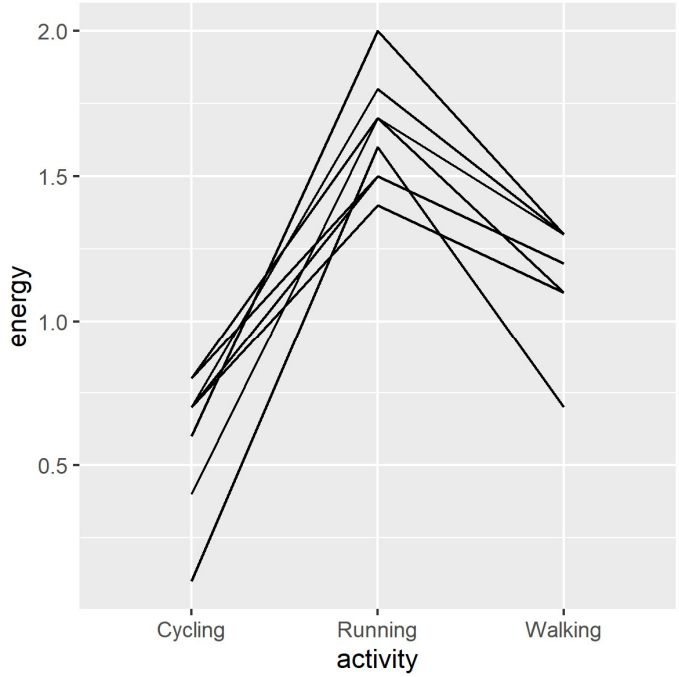
Converting to tall and thin, output

```
glimpse(en_tall)
## Rows: 24
## Columns: 3
## $ Subject  <dbl> 1, 1, 1, 2, 2, 2, 3, ~
## $ activity <chr> "Running", "Walking",~
## $ energy   <dbl> 1.4, 1.1, 0.7, 1.5, 1~
```

Lineplot

```
activity_lineplot <- ggplot(en_tall,
   aes(x=activity,
       y=energy,
       group=Subject)) +
     geom_line()
ggsave(
   "../images/activity-by-energy.png",
   activity_lineplot, width=4, height=4)
```

Lineplot

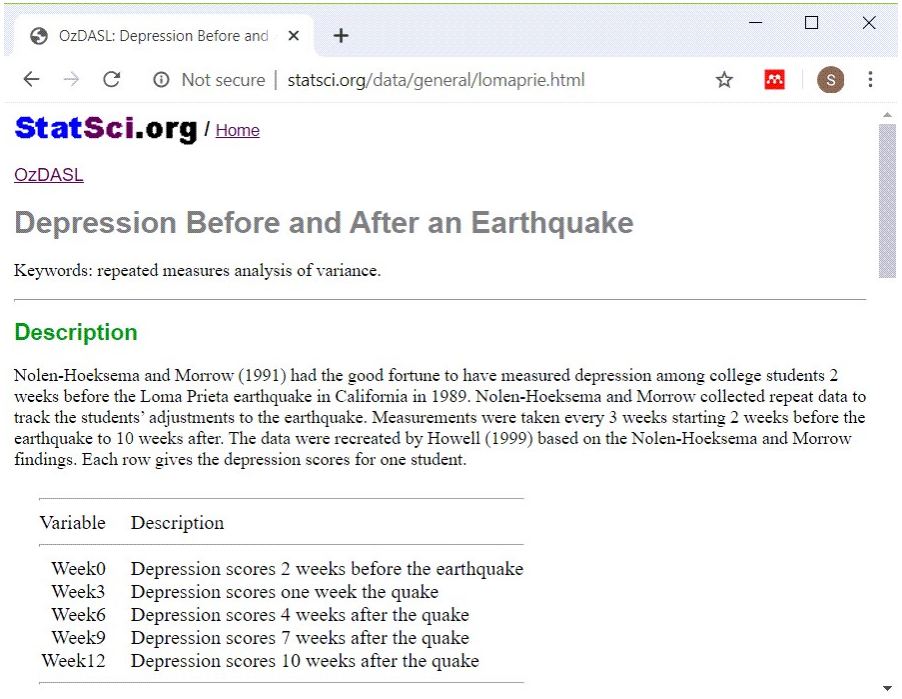


Plot showing activity levels by activity

Earthquake dataset

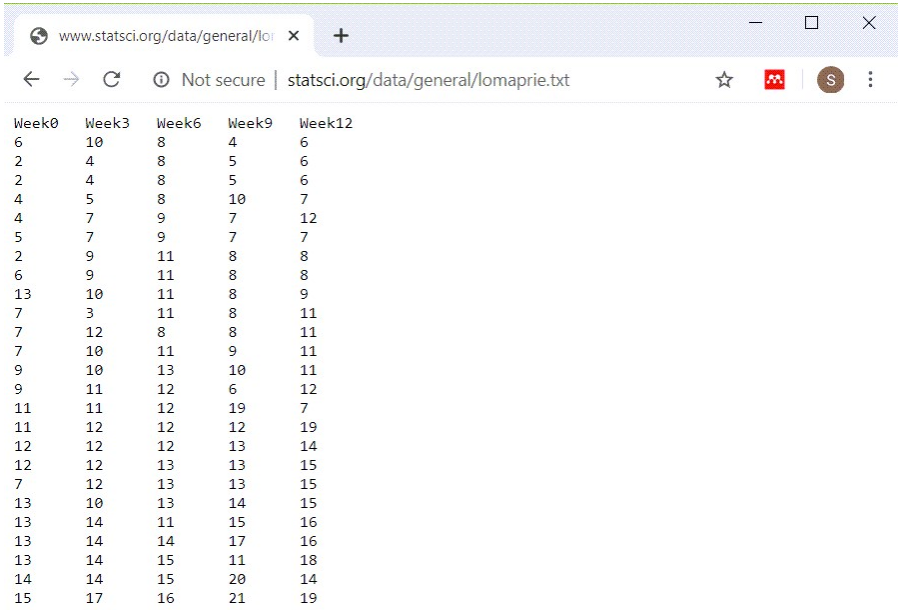
- Longitudinal study of stress
  - Study started two weeks prior to major earthquake (Week0)
  - Researchers added extra stress measurments
    - Week3, Week6, Week9, Week12
  - There are 25 subjects, 5 measurements

Earthquake dataset



Screenshot of data dictionary website

Earthquake dataset



Week0	Week3	Week6	Week9	Week12
6	10	8	4	6
2	4	8	5	6
2	4	8	5	6
4	5	8	10	7
4	7	9	7	12
5	7	9	7	7
2	9	11	8	8
6	9	11	8	8
13	10	11	8	9
7	3	11	8	11
7	12	8	8	11
7	10	11	9	11
9	10	13	10	11
9	11	12	6	12
11	11	12	19	7
11	12	12	12	19
12	12	12	13	14
12	12	13	13	15
7	12	13	13	15
13	10	13	14	15
13	14	11	15	16
13	14	14	17	16
13	14	15	11	18
14	14	15	20	14
15	17	16	21	19

View of the earthquake dataset

Read in the earthquake data

```
fn <- "../data/quake.txt"
qu <- read_table(fn, col_types="nnnnn")
```

Check structure of the earthquake data

```
glimpse(qu)
## Rows: 25
## Columns: 5
## $ Week0  <dbl> 6, 2, 2, 4, 4, 5, 2, 6,~
## $ Week3  <dbl> 10, 4, 4, 5, 7, 7, 9, 9~
## $ Week6  <dbl> 8, 8, 8, 8, 9, 9, 11, 1~
## $ Week9  <dbl> 4, 5, 5, 10, 7, 7, 8, 8~
## $ Week12 <dbl> 6, 6, 6, 7, 12, 7, 8, 8~
```

Convert to tall and thin format

```
qu$id <- 1:25
qu_tall <- pivot_longer(qu,
  contains("Week"),
  names_to="time",
  values_to="depression")
```

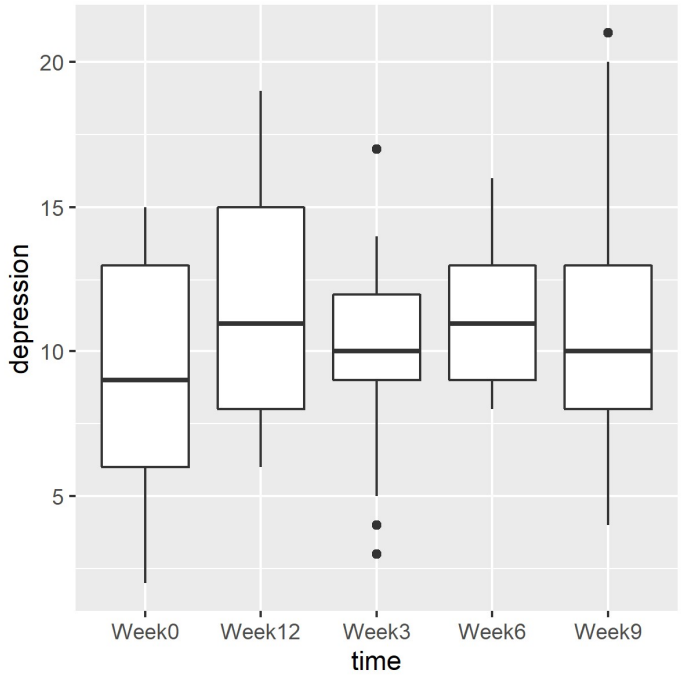
Display new structure

```
glimpse(qu_tall)
## Rows: 125
## Columns: 3
## $ id         <int> 1, 1, 1, 1, 1, 2, 2~
## $ time       <chr> "Week0", "Week3", "~
## $ depression <dbl> 6, 10, 8, 4, 6, 2, ~
```

Boxplot

```
depression_boxplot01 <-
  ggplot(
      qu_tall,
      aes(x=time, y=depression)) +
    geom_boxplot()
ggsave(
  "../images/time-by-depression01.png",
  depression_boxplot01, width=4, height=4)
```

Boxplot



Boxplots showing depression levels over time

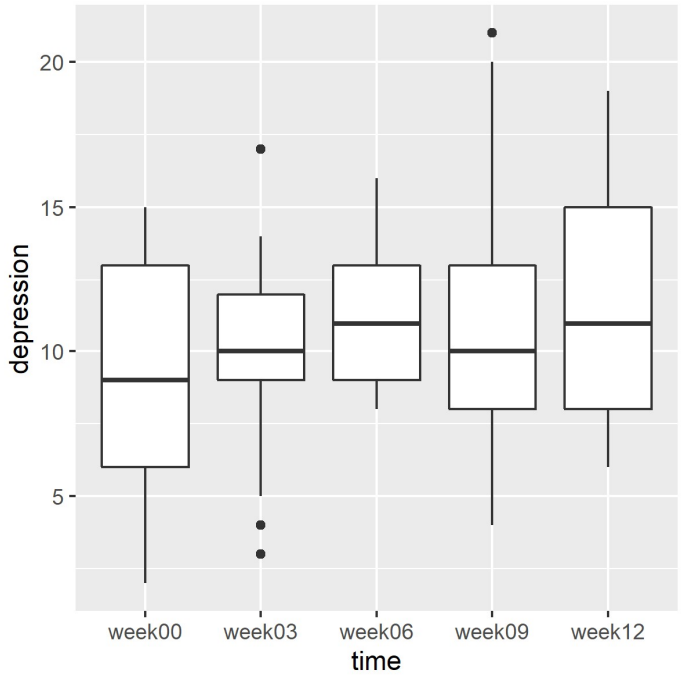
Boxplot

```
qu_tall$time <- case_when(
  qu_tall$time=="Week0"~"week00",
  qu_tall$time=="Week3"~"week03",
  qu_tall$time=="Week6"~"week06",
  qu_tall$time=="Week9"~"week09",
  qu_tall$time=="Week12"~"week12")
```

Re-drawn boxplot

```
depression_boxplot02 <-
  ggplot(qu_tall,
    aes(x=time, y=depression)) +
  geom_boxplot()
ggsave("../images/time-by-
depression02.png",
  depression_boxplot02, width=4, height=4)
```

Boxplot



Modified boxplots showing depression levels over time

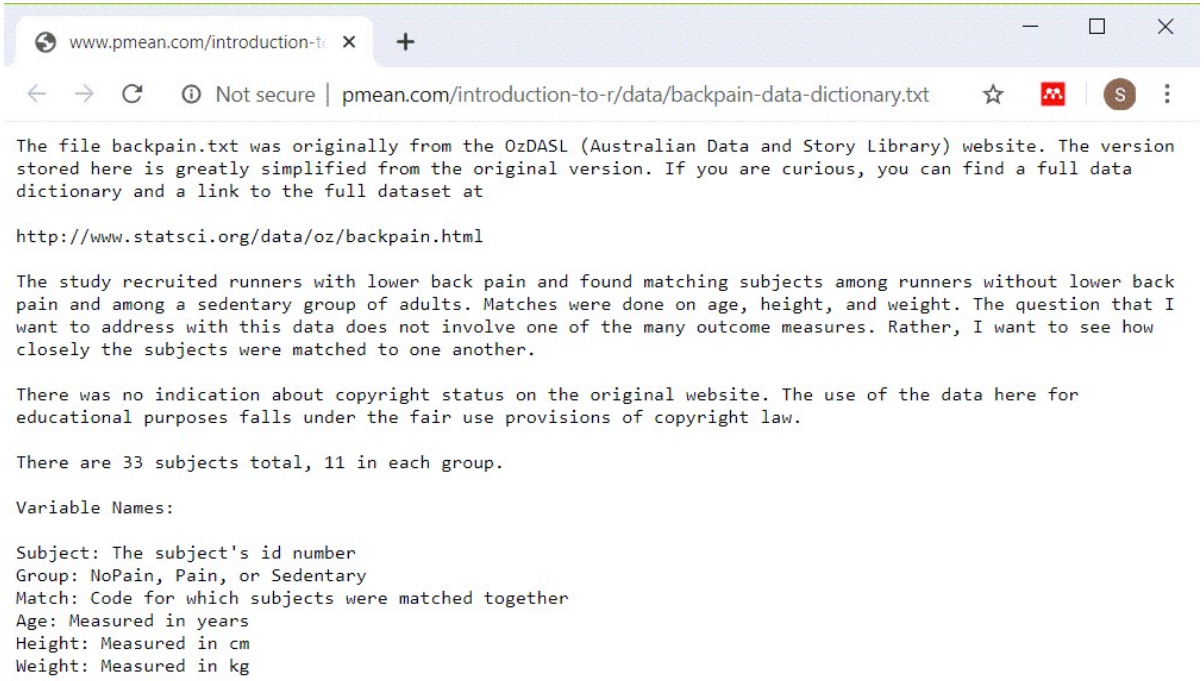
Break #2

- What have you learned
  - Converting to tall and thin format
- What is coming next
  - Converting to short and fat format

Backpain dataset

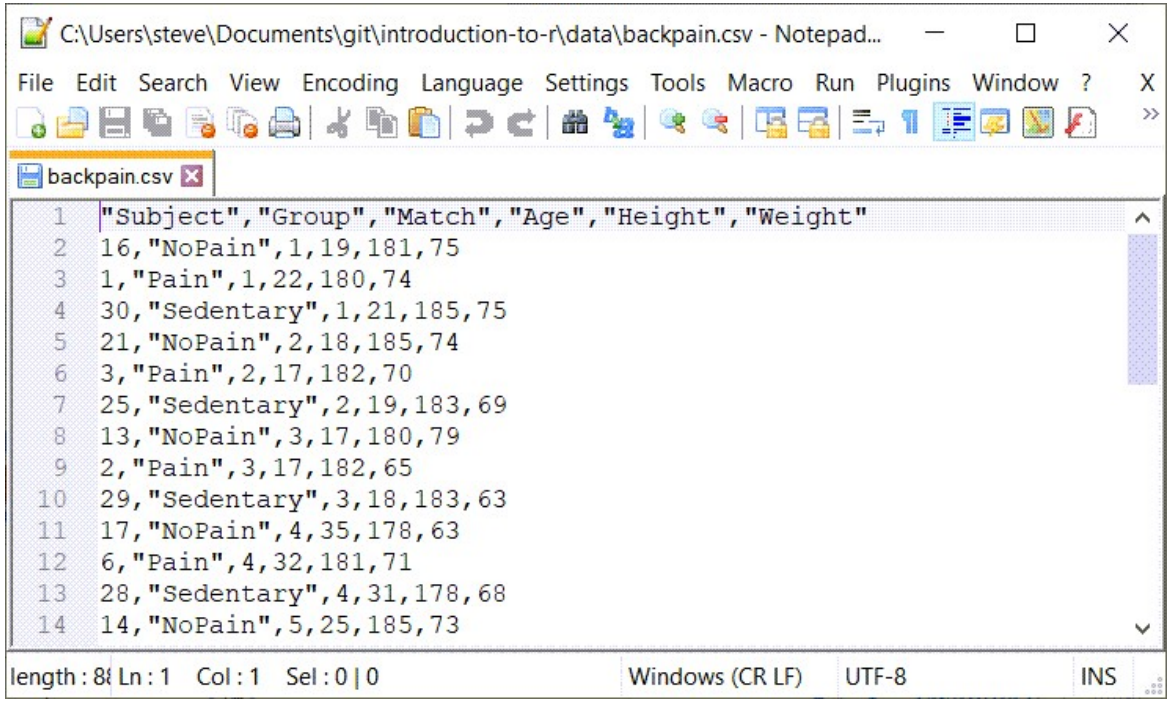
- Matched case-control study
  - Study of 11 runners with back pain
  - Two control groups
    - Runners without pain, Sedentary volunteers
    - Matched by age, height, weight
  - Outcome variables
    - Flexibility and length of various muscle groups
  - Also collected covariates
    - Type of running, number of years running
  - Our focus: quality of matching
  - Convert to one row per matched triple

Backpain overview



Data dictionary for backpain dataset

Backpain dataset



Partial view of backpain raw data

Reading in the backpain dataset

```
fn <- "../data/backpain.csv"
pain <- read_csv(fn, col_types="ncnnn")
```

Display

```
glimpse(pain)
## Rows: 33
## Columns: 6
## $ Subject <dbl> 16, 1, 30, 21, 3, 25, ~
## $ Group   <chr> "NoPain", "Pain", "Sed~
## $ Match   <dbl> 1, 1, 1, 2, 2, 2, 3, 3~
## $ Age     <dbl> 19, 22, 21, 18, 17, 19~
## $ Height  <dbl> 181, 180, 185, 185, 18~
## $ Weight  <dbl> 75, 74, 75, 74, 70, 69~
```

Converting to short and fat

```
pain_fat <- pivot_wider(pain,
   id_cols=Match,
   names_from=Group,
   values_from=c(Age, Height, Weight))
```

Display new structure

```
glimpse(pain_fat)
## Rows: 11
## Columns: 10
## $ Match                <dbl> 1, 2, 3, 4, 5~
## $ Age_NoPain           <dbl> 19, 18, 17, 3~
## $ Age_Pain             <dbl> 22, 17, 17, 3~
## $ Age_Sedentary        <dbl> 21, 19, 18, 3~
## $ Height_NoPain        <dbl> 181, 185, 180~
## $ Height_Pain          <dbl> 180, 182, 182~
```

Remaining variables

```
names(pain_fat)[7:10]
## [1] "Height_Sedentary"
## [2] "Weight_NoPain"
## [3] "Weight_Pain"
## [4] "Weight_Sedentary"
```

Backpain plot code (1 of 3)

```
age_range <- range(c(
  pain_fat$Age_Pain,
  pain_fat$Age_NoPain,
  pain_fat$Age_Sedentary))
```

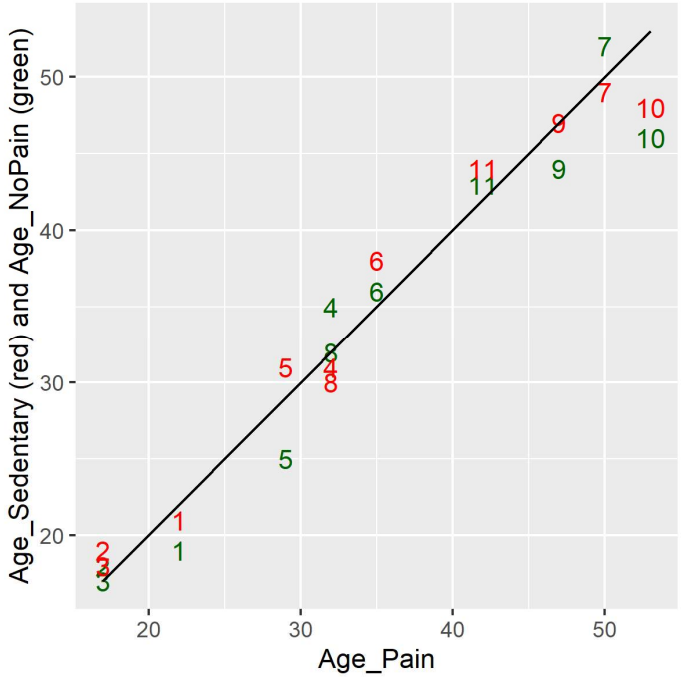
Backpain plot code (2 of 3)

```
agreement_age <- ggplot(pain_fat,
   aes(x=Age_Pain,
       y=Age_NoPain,
       label=Match)) +
     geom_text(color="darkgreen") +
     geom_text(aes(y=Age_Sedentary),
color="red") +
     ylab("Age_Sedentary (red) and
Age_NoPain (green)" )
```

Backpain plot code (3 of 3)

```
agreement_age <-
  agreement_age +
    expand_limits(
      x=age_range, y=age_range) +
    geom_segment(
      x=age_range[1], xend=age_range[2],
      y=age_range[1], yend=age_range[2])
ggsave(
  "../images/agreement_age.png",
  agreement_age, width=4, height=4)
```

Plots of agreement



Plot showing agreement in ages

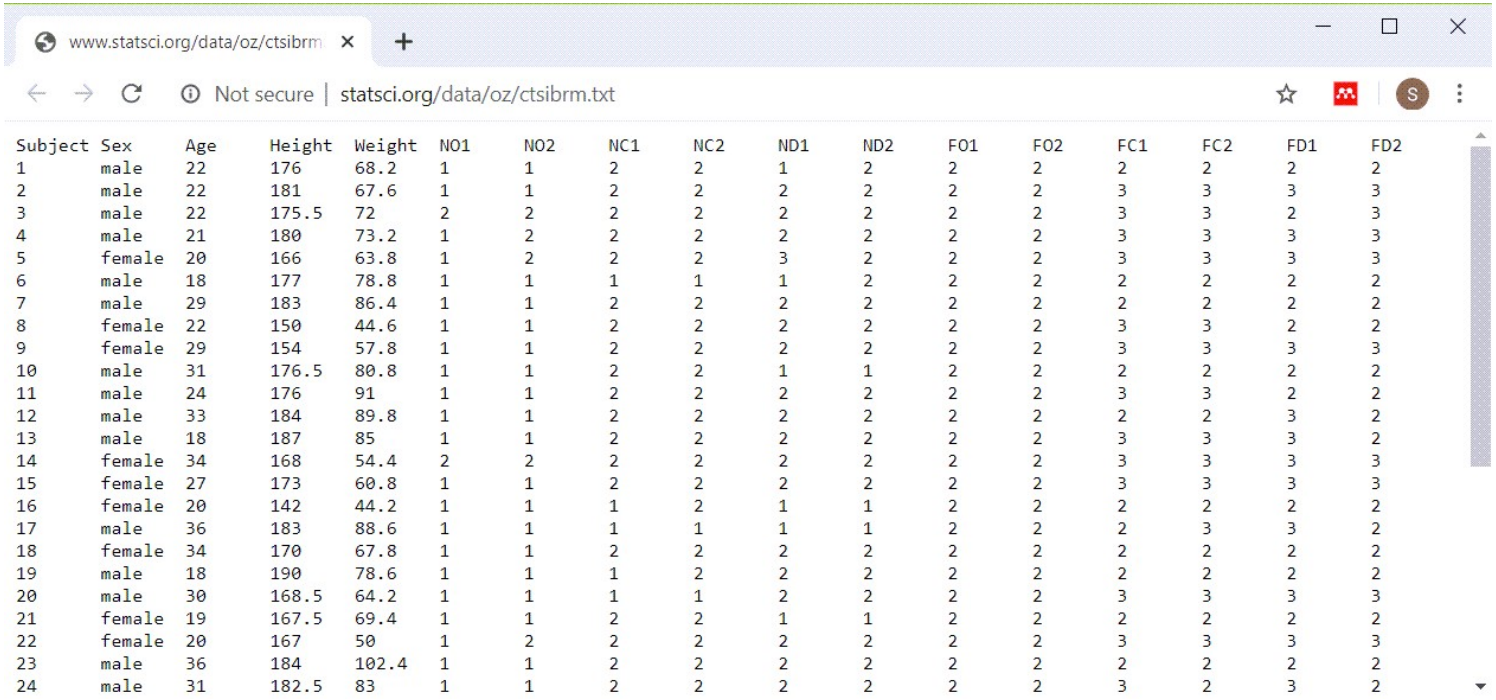
Break #3

- What have you learned
  - Converting to short and fat format
- What is coming next
  - Separating into time constant/time varying tables

One last recommendation

- Both formats have problems
  - Tall and thin: repetition of demographic information
  - Short and fat: poor handling of missing value
- Ideal solution: normalization
  - Put time constant data in first table
  - Put time varying data in second table

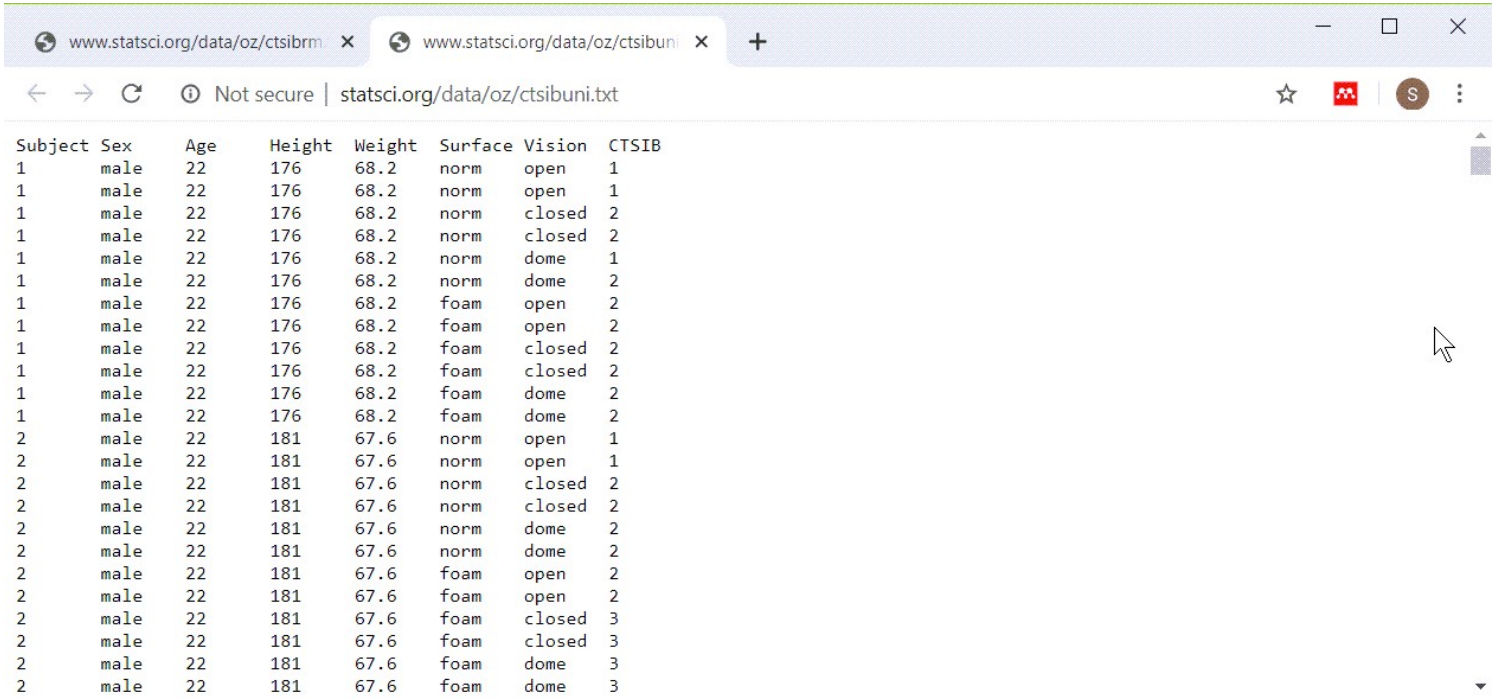
Balance data set: Short and fat format



Subject	Sex	Age	Height	Weight	NO1	NO2	NC1	NC2	ND1	ND2	FO1	FO2	FC1	FC2	FD1	FD2
1	male	22	176	68.2	1	1	2	2	1	2	2	2	2	2	2	2
2	male	22	181	67.6	1	1	2	2	2	2	2	2	3	3	3	3
3	male	22	175.5	72	2	2	2	2	2	2	2	2	3	3	2	3
4	male	21	180	73.2	1	2	2	2	2	2	2	2	3	3	3	3
5	female	20	166	63.8	1	2	2	2	3	2	2	2	3	3	3	3
6	male	18	177	78.8	1	1	1	1	1	2	2	2	2	2	2	2
7	male	29	183	86.4	1	1	2	2	2	2	2	2	2	2	2	2
8	female	22	150	44.6	1	1	2	2	2	2	2	2	3	3	2	2
9	female	29	154	57.8	1	1	2	2	2	2	2	2	3	3	3	3
10	male	31	176.5	80.8	1	1	2	2	1	1	2	2	2	2	2	2
11	male	24	176	91	1	1	2	2	2	2	2	2	3	3	2	2
12	male	33	184	89.8	1	1	2	2	2	2	2	2	2	2	3	2
13	male	18	187	85	1	1	2	2	2	2	2	2	3	3	3	2
14	female	34	168	54.4	2	2	2	2	2	2	2	2	3	3	3	3
15	female	27	173	60.8	1	1	2	2	2	2	2	2	3	3	3	3
16	female	20	142	44.2	1	1	1	2	1	1	2	2	2	2	2	2
17	male	36	183	88.6	1	1	1	1	1	1	2	2	2	3	3	2
18	female	34	170	67.8	1	1	2	2	2	2	2	2	2	2	2	2
19	male	18	190	78.6	1	1	1	2	2	2	2	2	2	2	2	2
20	male	30	168.5	64.2	1	1	1	1	2	2	2	2	3	3	3	3
21	female	19	167.5	69.4	1	1	2	2	1	1	2	2	2	2	2	2
22	female	20	167	50	1	2	2	2	2	2	2	2	3	3	3	3
23	male	36	184	102.4	1	1	2	2	2	2	2	2	2	2	2	2
24	male	31	182.5	83	1	1	2	2	2	2	2	2	3	2	3	2

Longitudinal data stored one row per subject

Tall and thin example



Subject	Sex	Age	Height	Weight	Surface	Vision	CTSIB
1	male	22	176	68.2	norm	open	1
1	male	22	176	68.2	norm	open	1
1	male	22	176	68.2	norm	closed	2
1	male	22	176	68.2	norm	closed	2
1	male	22	176	68.2	norm	dome	1
1	male	22	176	68.2	norm	dome	2
1	male	22	176	68.2	foam	open	2
1	male	22	176	68.2	foam	open	2
1	male	22	176	68.2	foam	closed	2
1	male	22	176	68.2	foam	closed	2
1	male	22	176	68.2	foam	dome	2
1	male	22	176	68.2	foam	dome	2
2	male	22	181	67.6	norm	open	1
2	male	22	181	67.6	norm	open	1
2	male	22	181	67.6	norm	closed	2
2	male	22	181	67.6	norm	closed	2
2	male	22	181	67.6	norm	dome	2
2	male	22	181	67.6	norm	dome	2
2	male	22	181	67.6	foam	open	2
2	male	22	181	67.6	foam	open	2
2	male	22	181	67.6	foam	closed	3
2	male	22	181	67.6	foam	closed	3
2	male	22	181	67.6	foam	dome	3
2	male	22	181	67.6	foam	dome	3

Longitudinal data stores with multiple rows per patient

Read balance in short and fat structure

```
fn <- "../data/balance1.txt"
short_and_fat_data <- read_tsv(fn,
col_types="ncnnnnnnnnnnnnnnnnnn")
```

Display balance short and fat structure

```
glimpse(short_and_fat_data)
## Rows: 40
## Columns: 17
## $ Subject <dbl> 1, 2, 3, 4, 5, 6, 7, 8~
## $ Sex     <chr> "male", "male", "male"~
## $ Age     <dbl> 22, 22, 22, 21, 20, 18~
## $ Height  <dbl> 176.0, 181.0, 175.5, 1~
## $ Weight  <dbl> 68.2, 67.6, 72.0, 73.2~
## $ NO1     <dbl> 1, 1, 2, 1, 1, 1, 1, 1~
```

Additional variables

```
names(short_and_fat_data)[7:17]
##  [1] "NO2" "NC1" "NC2" "ND1" "ND2"
"FO1"
##  [7] "FO2" "FC1" "FC2" "FD1" "FD2"
```

Create time constant data

```
time_constant <- c(
  "Subject",
  "Sex",
  "Age",
  "Height",
  "Weight")
time_constant_data <-
  short_and_fat_data[ , time_constant]
```

Structure of time constant data

```
glimpse(time_constant_data)
## Rows: 40
## Columns: 5
## $ Subject <dbl> 1, 2, 3, 4, 5, 6, 7, 8~
## $ Sex     <chr> "male", "male", "male"~
## $ Age     <dbl> 22, 22, 22, 21, 20, 18~
## $ Height  <dbl> 176.0, 181.0, 175.5, 1~
## $ Weight  <dbl> 68.2, 67.6, 72.0, 73.2~
```

Read balance in tall and thin format

```
fn <- "../data/balance2.txt"
tall_and_thin_data <-
  read_table(fn, col_types="ncnnnccn")
```

Balance data, tall and thing structure

```
glimpse(tall_and_thin_data)
## Rows: 480
## Columns: 8
## $ Subject <dbl> 1, 1, 1, 1, 1, 1, 1, 1~
## $ Sex     <chr> "male", "male", "male"~
## $ Age     <dbl> 22, 22, 22, 22, 22, 22~
## $ Height  <dbl> 176, 176, 176, 176, 17~
## $ Weight  <dbl> 68.2, 68.2, 68.2, 68.2~
## $ Surface <chr> "norm", "norm", "norm"~
```

Additional variables

```
names(tall_and_thin_data)[7:8]
## [1] "Vision" "CTSIB"
```

Create time varying table

```
time_variable <- c(
  "Subject",
  "Surface",
  "Vision",
  "CTSIB")
time_variable_data <-
  tall_and_thin_data[ , time_variable]
```

Display structure of time varying table

```
glimpse(time_variable_data)
## Rows: 480
## Columns: 4
## $ Subject <dbl> 1, 1, 1, 1, 1, 1, 1, 1~
## $ Surface <chr> "norm", "norm", "norm"~
## $ Vision  <chr> "open", "open", "close~
## $ CTSIB   <dbl> 1, 1, 2, 2, 1, 2, 2, 2~
```

Summary

- Two formats
  - Short and fat
  - Tall and thin
- pivot_longer
  - converts to tall and thin
- pivot_wider
  - converts to short and fat
- Alternative approach
  - Time constant table
  - Time variable table