CH4

2023-07-27

1. Merge

(1) rbind

```
library(tidyverse)
dat <- read.csv("C:\\Users\\phl02\\Desktop\\P\\bio\\ch4\\Ch4_chb.csv")
```

```
a <- c(1,2,3)
b <- c(4,5,6)
c <- rbind(a,b)
c
```

```
##   [,1] [,2] [,3]
## a    1    2    3
## b    4    5    6
```

```
a <- c(1,2,3,4)
b <- c(5,6,7)
c <- rbind(a,b)
c
```

```
##   [,1] [,2] [,3] [,4]
## a    1    2    3    4
## b    5    6    7    5
```

실제 데이터프레임 형태의 데이터를 이용해서 실습

```
temp1 <- dat %>%
  select(id, age, lc, hcc) %>%
  filter(id<5) %>%
  print()
```

```
##   id age lc hcc
## 1  1  54  1   1
## 2  2  45  0   0
## 3  3  49  1   0
## 4  4  26  0   0
```

```
temp2 <- dat %>%
  filter(id>6) %>%
  select(id, age, lc, hcc) %>%
  print()
```

```
##   id age lc hcc
## 1  7  49  1   1
## 2  8  50  0   0
## 3  9  49  1   0
## 4 10  50  0   0
```

```
temp3 <- rbind(temp1, temp2)
temp3
```

```
##   id age lc hcc
## 1  1  54  1   1
## 2  2  45  0   0
## 3  3  49  1   0
## 4  4  26  0   0
## 5  7  49  1   1
## 6  8  50  0   0
## 7  9  49  1   0
## 8 10  50  0   0
```

(2) cbind

```
a <- c(1,2,3)
b <- c(4,5,6)
c <- cbind(a,b)
c
```

```
##      a b
## [1,] 1 4
## [2,] 2 5
## [3,] 3 6
```

실제 데이터프레임 형태의 데이터를 이용해서 실습

```
temp4 <- dat %>%
  select(id, age, lc, hcc) %>%
  filter(id<5) %>%
  print()
```

```
##   id age lc hcc
## 1  1  54  1   1
## 2  2  45  0   0
## 3  3  49  1   0
## 4  4  26  0   0
```

```
temp5 <- dat %>%
  filter(id<5) %>%
  select(b_alt, b_bil, b_inr) %>%
  print()
```

```
##   b_alt b_bil b_inr
## 1    67   1.2  1.17
## 2    32   1.1  0.90
## 3   106   2.3  1.03
## 4   159   1.4  1.04
```

```
temp6 <- cbind(temp4, temp5)
temp6
```

```
##   id age lc hcc b_alt b_bil b_inr
## 1  1  54  1   1    67   1.2  1.17
## 2  2  45  0   0    32   1.1  0.90
## 3  3  49  1   0   106   2.3  1.03
## 4  4  26  0   0   159   1.4  1.04
```

(3) merge

```
temp1 <- dat %>%
  filter(id <4) %>%
  select(id, age, gender) %>%
  print()
```

```
##   id age gender
## 1  1  54      M
## 2  2  45      F
## 3  3  49      M
```

```
temp2 <- dat %>%
  filter(id %in% c(1,2,4,5,6)) %>%
  select(id, lc, hcc) %>%
  print()
```

```
##   id lc hcc
## 1  1  1   1
## 2  2  0   0
## 3  4  0   0
## 4  5  1   0
## 5  6  1   0
```

inner join

```
merge(temp1, temp2, by='id')
```

```
##   id age gender lc hcc
## 1  1  54      M  1   1
## 2  2  45      F  0   0
```

outer join

```
merge(temp1, temp2, by='id', all=TRUE)
```

```
##   id age gender lc hcc
## 1  1  54      M  1   1
## 2  2  45      F  0   0
## 3  3  49      M NA  NA
## 4  4  NA   <NA>  0   0
## 5  5  NA   <NA>  1   0
## 6  6  NA   <NA>  1   0
```

left join

```
merge(temp1, temp2, by='id', all.x=TRUE)
```

```
##   id age gender lc hcc
## 1  1  54      M  1   1
## 2  2  45      F  0   0
## 3  3  49      M NA  NA
```

right join

```
merge(temp1, temp2, by='id', all.y=TRUE)
```

```
##   id age gender lc hcc
## 1  1  54      M  1   1
## 2  2  45      F  0   0
## 3  4  NA   <NA>  0   0
## 4  5  NA   <NA>  1   0
## 5  6  NA   <NA>  1   0
```

2. Tidyverse를 이용한 merge

2.1 inner join

```
inner_join(temp1, temp2, by='id')
```

```
##   id age gender lc hcc
## 1  1  54      M  1   1
## 2  2  45      F  0   0
```

2.2 full join

```
full_join(temp1, temp2, by='id')
```

```
##   id age gender lc hcc
## 1  1  54      M  1   1
## 2  2  45      F  0   0
## 3  3  49      M NA  NA
## 4  4  NA   <NA>  0   0
## 5  5  NA   <NA>  1   0
## 6  6  NA   <NA>  1   0
```

2.3 left join

```
left_join(temp1, temp2, by='id')
```

```
##   id age gender lc hcc
## 1  1  54      M  1   1
## 2  2  45      F  0   0
## 3  3  49      M NA  NA
```

2.4 right join

```
right_join(temp1, temp2, by='id')
```

```
##   id age gender lc hcc
## 1  1  54      M  1   1
## 2  2  45      F  0   0
## 3  4  NA   <NA>  0   0
## 4  5  NA   <NA>  1   0
## 5  6  NA   <NA>  1   0
```

2.5 semi join, anti join

```
semi_join(temp1, temp2, by='id')
```

```
##   id age gender
## 1  1  54      M
## 2  2  45      F
```

```
anti_join(temp1, temp2, by='id')
```

```
##   id age gender
## 1  3  49      M
```

2.6 intersect, union, setdiff

```
temp.x <- dat %>%
  select(id, age, gender) %>%
  filter(id<4) %>%
  print( )
```

```
##   id age gender
## 1  1  54      M
## 2  2  45      F
## 3  3  49      M
```

```
temp.y <- dat %>%
  select(id, age, gender) %>%
  filter(between (id, 2, 4)) %>%
  print( )
```

```
##   id age gender
## 1  2  45      F
## 2  3  49      M
## 3  4  26      M
```

```
intersect(temp.x, temp.y)
```

```
##   id age gender
## 1  2  45      F
## 2  3  49      M
```

```
union(temp.x, temp.y)
```

```
##   id age gender
## 1  1  54      M
## 2  2  45      F
## 3  3  49      M
## 4  4  26      M
```

```
setdiff(temp.x, temp.y)
```

```
##   id age gender
## 1  1  54      M
```

2.7 bind_rows, bind_cols

```
bind_rows(temp.x, temp.y)
```

```
##   id age gender
## 1  1  54      M
## 2  2  45      F
## 3  3  49      M
## 4  2  45      F
## 5  3  49      M
## 6  4  26      M
```

```
bind_cols(temp.x, temp.y)
```

```
## New names:
## * `id` -> `id...1`
## * `age` -> `age...2`
## * `gender` -> `gender...3`
## * `id` -> `id...4`
## * `age` -> `age...5`
## * `gender` -> `gender...6`
```

```
##   id...1 age...2 gender...3 id...4 age...5 gender...6
## 1      1      54          M      2      45          F
## 2      2      45          F      3      49          M
## 3      3      49          M      4      26          M
```

rbind와는 다른 기능
서로 다른 그룹의 데이터를 합치면서 그룹의 이름을 1개의 새로운 변수로 만들 수 있음

```
bind_rows(entecavir=temp.x, tenofovir=temp.y, .id = 'treatment')
```

```
##   treatment id age gender
## 1 entecavir  1  54      M
## 2 entecavir  2  45      F
## 3 entecavir  3  49      M
## 4 tenofovir  2  45      F
## 5 tenofovir  3  49      M
## 6 tenofovir  4  26      M
```

3. Tidy 데이터

```
library(tidyverse)
dat <- read.csv("C:\\Users\\phl02\\Desktop\\P\\bio\\ch4\\Ch4_chb2.csv")
```

3.1 Tidy데이터의 특징

임상연구 데이터에서 행은 환자 1명을 의미하며 가로로 수집한 변수들이 계속 나열되게 됨
==> 시각화와 분석을 할 때는 tidy한 데이터 형태가 더 다루기 쉬움

tidy 형태 특징

- 각 변수는 개별 열로 되어 있어야 함
- 개별 관찰치는 행으로 되어 있어야 함
- 개별 테이블은 개별 관찰치에 의해 만들어진 데이터를 나타내야 함
- 만약 여러 개의 테이블이 존재한다면 최소 1개 이상의 열이 공유

3.2 Tidy 데이터 만들기 연습

long 형태로 변형

```
alt.long <- dat %>%
  gather(2:6, key='observation', value='alt_result') %>%
  arrange(id)
head(alt.long,10)
```

```
##    id observation alt_result
## 1   1       b_alt         67
## 2   1      m6_alt         35
## 3   1     m12_alt         37
## 4   1     m18_alt         28
## 5   1     m24_alt         31
## 6   2       b_alt         32
## 7   2      m6_alt         16
## 8   2     m12_alt         18
## 9   2     m18_alt         13
## 10  2     m24_alt         14
```

```
dim(dat)
```

```
## [1] 30  6
```

```
dim(alt.long)
```

```
## [1] 150   3
```

wide 형태로 변형

```
alt.wide<-alt.long %>%
  spread(observation, alt_result)
head(alt.wide,10)
```

```
##    id b_alt m12_alt m18_alt m24_alt m6_alt
## 1   1    67      37      28      31     35
## 2   2    32      18      13      14     16
## 3   3   106      27      25      28    108
## 4   4   159      29      28      23     24
## 5   5    94      35      36      19     57
## 6   6    32      17      21      11     26
## 7   7   104      42      37      32     51
## 8   8   143      16      50      19     24
## 9   9    31      24      59      54     25
## 10 10   239      34      37      28     30
```

```
dim(alt.long)
```

```
## [1] 150   3
```

```
dim(alt.wide)
```

```
## [1] 30  6
```