Modern Data Science Methods for Educational Research

R for Data Analysis in Educational Research

การเตรียมข้อมูล 2 : Manipulating Data

ผศ.ดร.สิวะโชติ ศรีสุทธิยากร

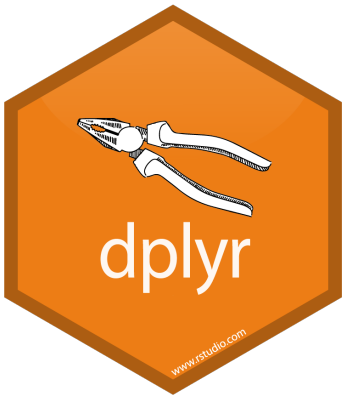
ภาควิชาวิจัยและจิตวิทยาการศึกษา
คณะครุศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

February 11, 2023


KRUROO
EDU @ CHULALONGKORN

1. dplyr package

dplyr package




```
1 install.packages("dplyr")
2 library(dplyr)
```

piping operator (%>%)

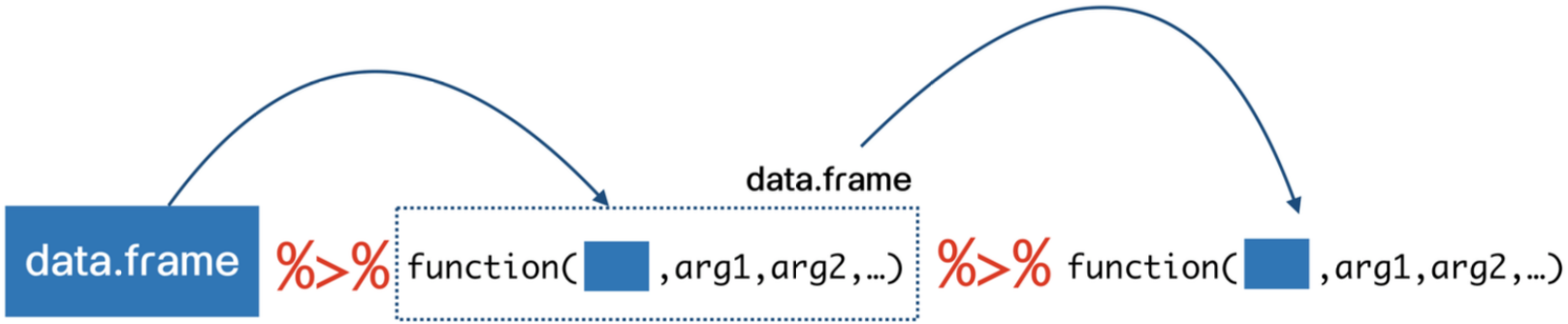


Figure 1: ที่มา : สิวะโชติ ศรีสุทธิยากร (2564)

piping operator (%>%)

จากตัวอย่าง messydata.xlsx เราสามารถใช้ piping operator
เพื่อเชื่อมต่อการดำเนินการในขั้นตอนต่าง ๆ เข้าด้วยกัน

```
# A tibble: 6 x 5
  ...1            Lecture.pre PBL.pre Lecture.post PBL.post
  <chr>                 <dbl>   <dbl>        <dbl>    <dbl>
1 Ancient One              20      NA           45       NA
2 Adam Warlock             16      NA           34       NA
3 Captain America          NA      18           NA       67
4 Colossus                 NA      25           NA       93
5 Captain Marvel           13      NA           50       NA
6 Diablo                   NA      17           NA       71
```

piping operator (%>%)

```
mytidy <- dat %>%
    gather( Lecture.pre, PBL.pre,
            Lecture.post, PBL.post,
          key = "method_time",
          value = "score") %>%
    separate(col = "method_time",
             into = c("method","time"),
             sep="[.]") %>%
    spread(key = "time",
       value = "score") %>%
  rename(student_name = ...1)
```

piping operator (%>%)

```
1 head(mytidy)
```

```
# A tibble: 6 x 4
  student_name    method   post   pre
  <chr>           <chr>   <dbl> <dbl>
1 Adam Warlock    Lecture    34    16
2 Adam Warlock    PBL        NA    NA
3 Ancient One     Lecture    45    20
4 Ancient One     PBL        NA    NA
5 Captain America Lecture    NA    NA
6 Captain America PBL        67    18
```

ขอบเขตของ dplyr

- Selecting variables
- Filtering cases
- Transforming variables
- Summarise variables

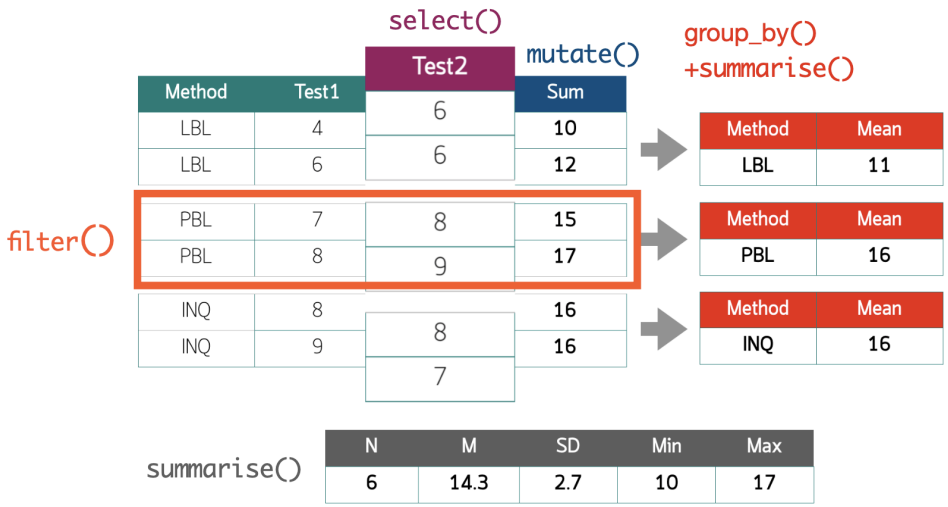


Figure 2: ที่มา : สิวะโชติ ศรีสุทธิยากร (2564)

ชุดข้อมูลที่ใช้เป็นตัวอย่าง

บทเรียนนี้จะใช้ชุดข้อมูล 2 ชุดได้แก่

- ชุดข้อมูลจากกิจกรรม My First Tidy Data
- ชุดข้อมูลจากไฟล์ข้อมูล mytidy2.csv

```
1 mytidy2 <- read.csv("mytidy2.csv", header=T)
2 head(mytidy2)
```

```
    X              name infect stress.1 stress.2 stress.3 str
1  20     Adam Warlock      2        3        2        3
2 973      Ancient One      2        3        2        3
3 520 Captain America      2        3        2        3
4  47  Captain Marvel      2        2        1        1
5 867         Colossus      1        3        2        3
6 335         Deadpool      2        1        1        1
  stu_engage_1 stu_engage_2 stu_engage_3 stu_engage_4 stu_
```

mytidy2

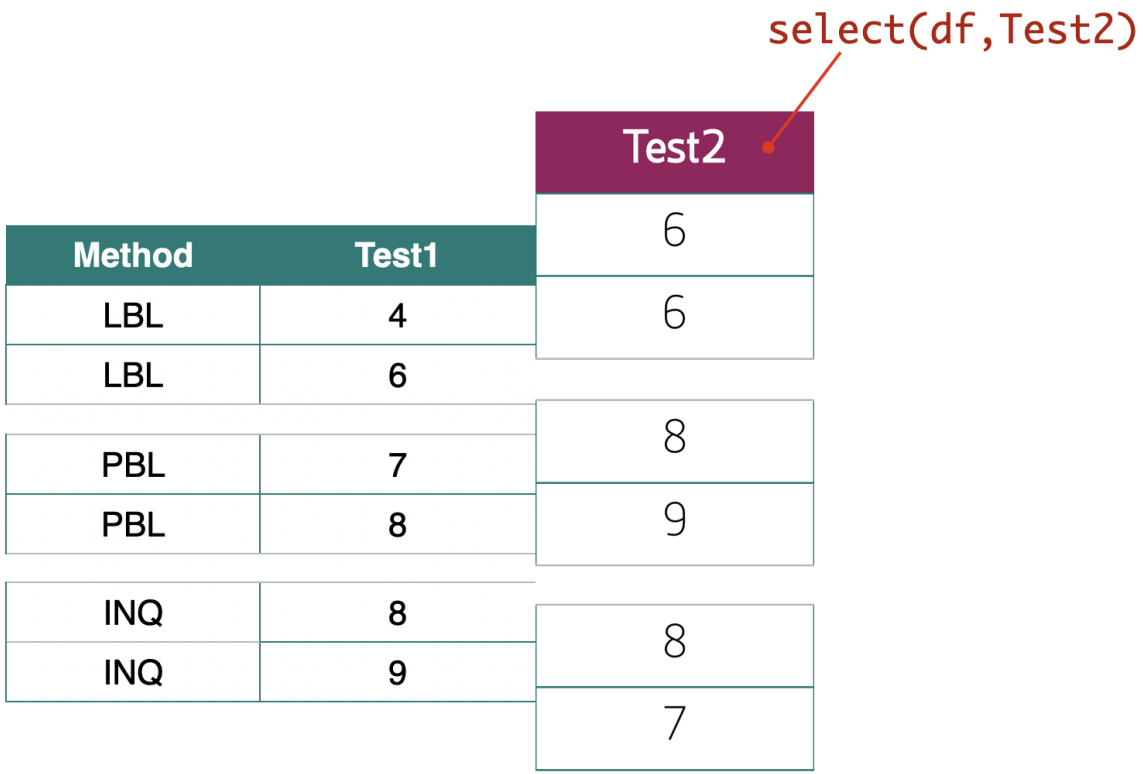
```
1 str(mytidy2)
```

```
'data.frame':   20 obs. of  16 variables:
 $ X            : int  20 973 520 47 867 335 308 218 439 869
 $ name         : chr  "Adam Warlock" "Ancient One" "Captain
 $ infect       : int  2 2 2 2 1 2 1 1 2 1 ...
 $ stress.1     : int  3 3 3 2 3 1 2 3 1 2 ...
 $ stress.2     : int  2 2 2 1 2 1 1 2 1 2 ...
 $ stress.3     : int  3 3 3 1 3 1 2 3 1 3 ...
 $ stress.4     : int  2 3 2 1 4 1 1 2 1 2 ...
 $ stress.5     : int  2 1 1 1 2 1 1 1 1 1 ...
 $ stu.engage.1: int  5 5 5 5 3 5 5 5 5 5 ...
 $ stu.engage.2: int  4 4 4 5 2 5 5 4 5 4 ...
 $ stu.engage.3: int  4 4 4 5 1 5 5 4 4 3 ...
 $ stu.engage.4: int  5 5 5 5 4 5 5 5 5 3 ...
```

2. Selecting variables

Selecting variables

การคัดเลือกตัวแปรจากชุดข้อมูลใน R สามารถทำได้หลายวิธีการ
บทเรียนนี้จะแนะนำวิธีการที่ใช้ฟังก์ชัน `select()` ของ package dplyr



Selecting variables

```
subset_dat <- mytidy2 %>%
              select(stress.1, stress.2, stress.3,
                     stress.4, stress.5, gpax.y2)
head(subset_dat)
```

	stress.1	stress.2	stress.3	stress.4	stress.5	gpax.y2
1	3	2	3	2	2	2.76
2	3	2	3	3	1	2.84
3	3	2	3	2	1	2.48
4	2	1	1	1	1	3.46
5	3	2	3	4	2	1.81
6	1	1	1	1	1	2.15

Selecting variables : selection helpers

- `everything()`: Matches all variables.
- `last_col()`: Select last variable, possibly with an offset.
- `starts_with()`: Starts with a prefix.
- `ends_with()`: Ends with a suffix.
- `contains()`: Contains a literal string.
- `matches()`: Matches a regular expression.
- `num_range()`: Matches a numerical range like x01, x02, x03.

Selecting variables : selection helpers

```
1 mytidy2 %>%
2       select(starts_with("stress"),
3              contains("engage"),
4              gpax.y2)
```

	stress.1	stress.2	stress.3	stress.4	stress.5	stu.engage
1	3	2	3	2	2	
2	3	2	3	3	1	
3	3	2	3	2	1	
4	2	1	1	1	1	
5	3	2	3	4	2	
6	1	1	1	1	1	
7	2	1	2	1	1	
8	3	2	3	2	1	
9	1	1	1	1	1	

Selecting variables : selection helpers

```
1 mytidy2 %>%
2       select(stress.1:stress.5)
```

	stress.1	stress.2	stress.3	stress.4	stress.5
1	3	2	3	2	2
2	3	2	3	3	1
3	3	2	3	2	1
4	2	1	1	1	1
5	3	2	3	4	2
6	1	1	1	1	1
7	2	1	2	1	1
8	3	2	3	2	1
9	1	1	1	1	1
10	2	2	3	2	1
11	3	2	3	2	3

3. Transforming variable

Transforming variable using dplyr

Method	Test1	Test2	Sum	Gain
LBL	4	6	**10**	**2**
LBL	6	6	**12**	**0**
PBL	7	8	**15**	**1**
PBL	8	9	**17**	**1**
INQ	8	8	**16**	**0**
INQ	9	7	**16**	**-2**

```
mutate(df,Sum=Test1+Test2,
Gain=Test2-Test1)
```

Transforming variable using dplyr

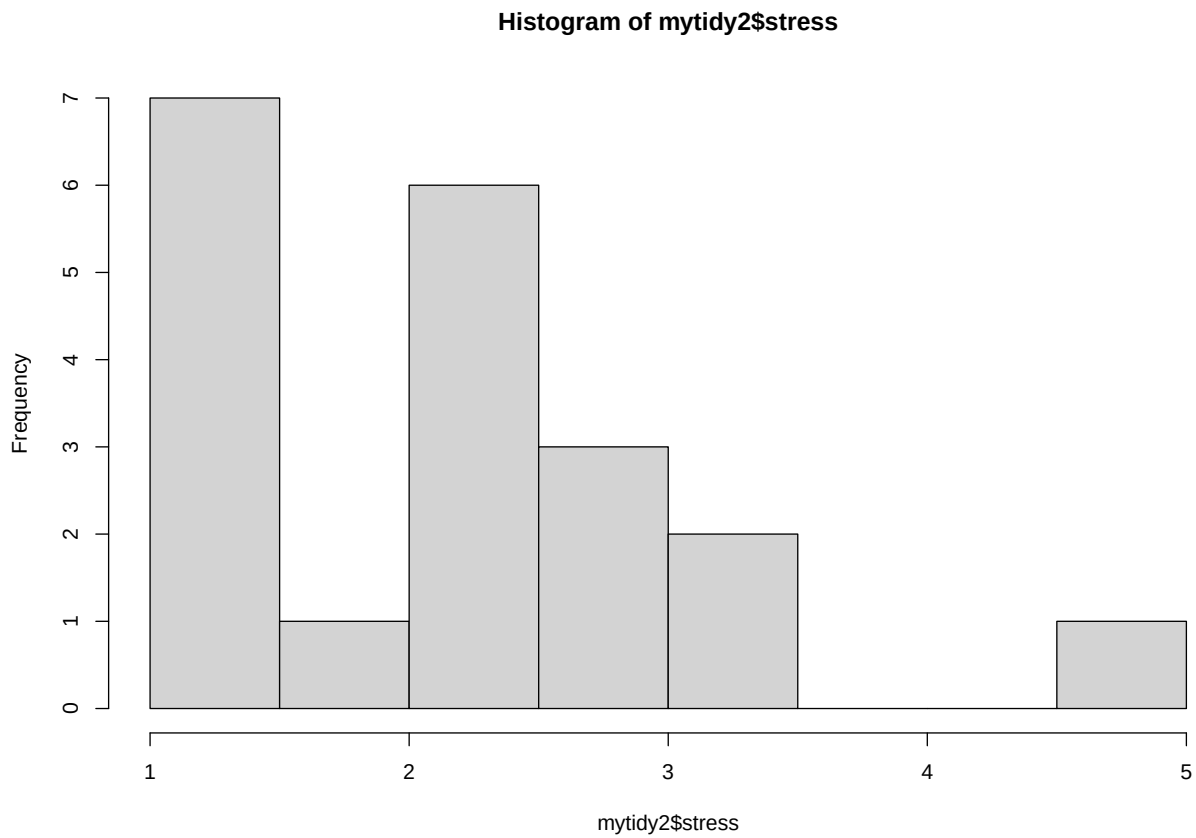
จากชุดข้อมูล mytidy2.csv

ลองคำนวณคะแนนความเครียดในการเรียนด้วยการเฉลี่ยคะแนนตัวบ่งชี้ความเครียด (stress.1, stress.2, …, stress.5)

```
mytidy2 <- mytidy2 %>%
  mutate(stress = (stress.1 + stress.2 +
           stress.3 + stress.4 + stress.5)/5)
# summary stat of stress
mytidy2 %>% select(stress) %>% summary()
# histogram of stress
hist(mytidy2$stress)
```

Transforming variable using dplyr

```
     stress
Min.   :1.00
1st Qu.:1.20
Median :2.20
Mean   :2.18
3rd Qu.:2.65
Max.   :5.00
```



Transforming variable using dplyr

สมมุติว่าผู้วิเคราะห์ต้องการสร้างตัวแปรใหม่ชื่อ `result` จากตัวแปรเดิมคือ `gpax.y2` โดยมีเกณฑ์ดังนี้

- ถ้า `gpax.y2` ≥ 1.5 หมายถึงสอบผ่าน (pass)
- ถ้า `gpax.y2` < 1.5 หมายถึงสอบตก (fail)

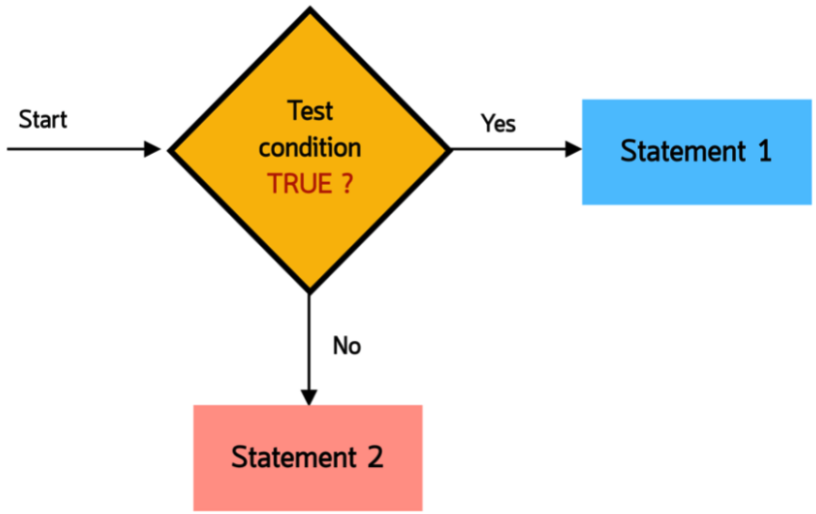


Figure 3: การทำงานของฟังก์ชัน ifelse() ที่มา : สิวะโชติ ศรีสุทธิยากร (2564)

Transforming variable using dplyr

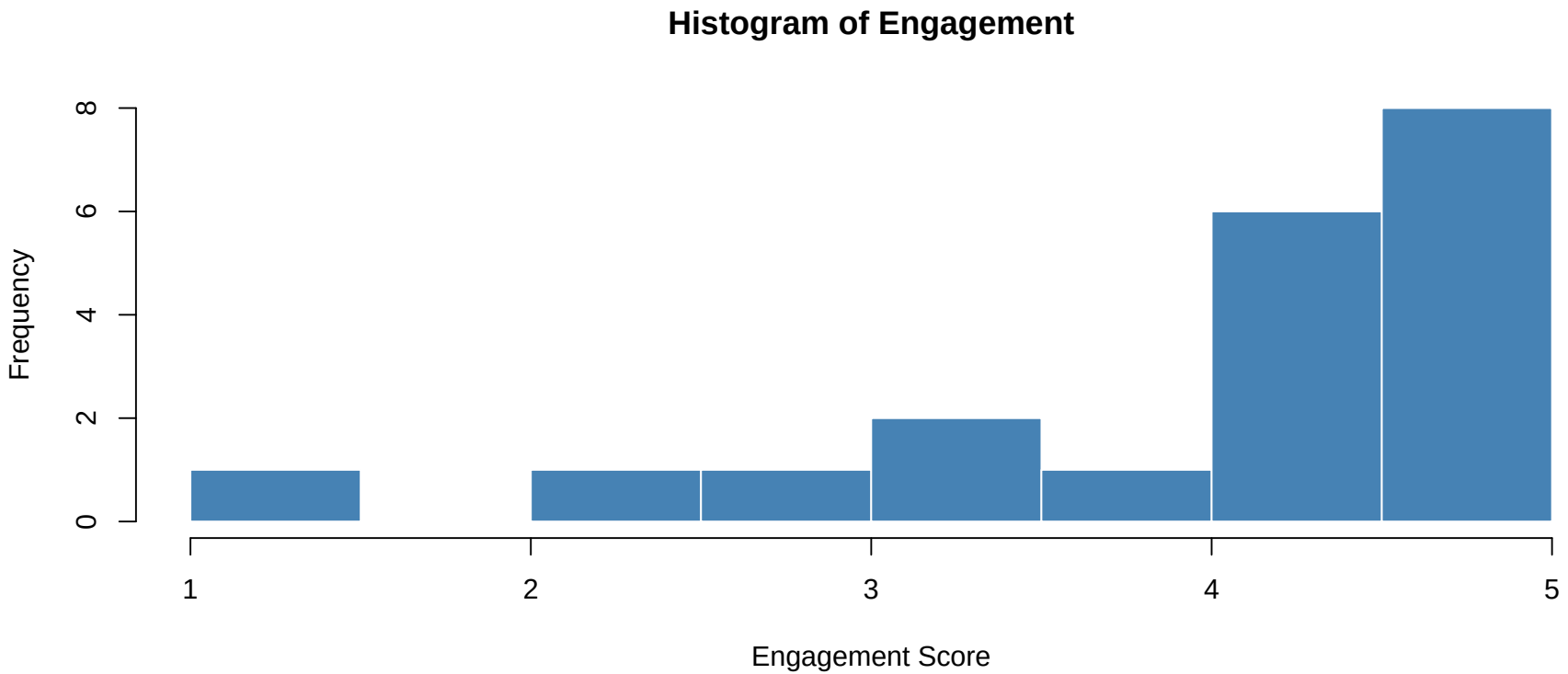
```
1 mytidy2 %>%
2   mutate(result = ifelse(gpax.y2 >= 1.5 ,1,0),
3          result = factor(result,
4                          labels=c("fail","pass")))
```

```
  result
fail: 2
pass:18
```

กิจกรรม : Calculate Student Engagement score

คำนวณคะแนน engagement ของนักเรียนรายบุคคลจากคะแนนเฉลี่ยตัวบ่งชี้ `stu.engage.1` - `stu.engage.4`

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  1.250   3.688   4.500   4.188   5.000   5.000
```



กิจกรรม : Transforming variable

ชุดข้อมูล mytidy2 มีตัวแปรจัดประเภท 3 ตัวได้แก่

- `infect` – เคยติด Covid-10 หรือไม่ (1 = yes และ 2 = no)
- `stu.itcap` – ความสามารถในการใช้ technology (1 = low, 2 = mid และ 3 = high)
- `internet` - สัญญาณ internet (1 =no problem และ 2 = have problem)

ขอให้ผู้เรียนเปลี่ยนสถานะของตัวแปรจัดประเภททั้ง 3 ในชุดข้อมูลข้างต้นให้เป็น Factor

4. Filtering cases

Filtering cases

การคัดกรองหน่วยข้อมูล
เป็นการเลือกหน่วยข้อมูลตามเงื่อนไขที่กำหนดจากชุดข้อมูล
การคัดกรองหน่วยข้อมูลใน R สามารถทำได้หลายวิธีการ
วิธีการพื้นฐานคือใช้การอ้างอิงสมาชิกแบบเมทริกซ์ เช่น

```
filter(data,Method=="PBL")
```

Method	Test1	Test2
LBL	4	6
LBL	6	6
PBL	7	8
PBL	8	9
INQ	8	8
INQ	9	7

Filtering cases using dplyr

<=	Less than or equal
==	Equal
!=	Not equal
>=	Greater than or equal
>	Greater than
<	Less than

Filtering cases using dplyr

```
1 mytidy2 %>%
2   filter(infect == 2)
```

	X	name	infect	stress.1	stress.2	stress.3	st
1	20	Adam Warlock	2	3	2	3	
2	973	Ancient One	2	3	2	3	
3	520	Captain America	2	3	2	3	
4	47	Captain Marvel	2	2	1	1	
5	335	Deadpool	2	1	1	1	
6	439	Dr. Strange	2	1	1	1	
7	462	Falcon	2	3	2	3	
8	921	Groot	2	1	1	1	
9	221	Hitman	2	4	2	4	
10	855	Hulk	2	5	5	5	
11	29	Iceman	2	3	2	4	

Filtering cases using dplyr

เราสามารถใช้ตัวดำเนินการเชิงตรรกะ ได้แก่ และ (,) หรือ (|)
เพื่อกำหนดเงื่อนไขที่ซับซ้อนขึ้นได้

```
1 mytidy2 %>%
2   filter(infect == 2 , gpax.y2 >3)
```

```
    X           name infect stress.1 stress.2 stress.3 stre
1  47 Captain Marvel      2        2        1        1
2 439    Dr. Strange      2        1        1        1
3  29         Iceman      2        3        2        4
4 562          Logan      2        4        2        3
  stu.engage.1 stu.engage.2 stu.engage.3 stu.engage.4 stu.
1            5            5            5            5
2            5            5            4            5
3            5            4            5            5
4            5            4            4            5
```

กิจกรรม : filtering cases

- นักเรียนที่มีเกรดเฉลี่ยสะสมน้อยกว่า 1.5 หรือมากกว่า 3.0 มีจำนวนกี่คน และมีใครบ้าง
- นักเรียนที่เก่งการใช้เทคโนโลยี (กลุ่มคล่องแคล่วมาก) และมีความเครียดในการเรียน (stress) น้อยกว่า 3 คะแนน มีกี่คน และในจำนวนนี้เกรดเฉลี่ยสะสมเป็นอย่างไร

arrange() function

ฟังก์ชัน `arrange()` ใช้สำหรับเรียงลำดับข้อมูลตามตัวแปรที่กำหนด โดยสามารถเรียงจากน้อยไปมาก หรือมากไปน้อยก็ได้ ตามกำหนด

ลองพิจารณาผลลัพธ์ต่อไปนี้

```
1 mytidy2 %>%
2 select(stress, gpax.y2) %>%
3 arrange(gpax.y2)
```

	stress	gpax.y2
1	5.0	1.06
2	3.4	1.33
3	1.4	1.72
4	2.8	1.81
5	2.2	2.05
6	1.0	2.15

```
1 mytidy2 %>%
2 select(stress, gpax.y2) %>%
3 arrange(desc(gpax.y2))
```

	stress	gpax.y2
1	1.0	3.48
2	1.2	3.46
3	2.4	3.30
4	2.0	3.22
5	2.8	3.20
6	1.0	2.98

กิจกรรม

นักเรียนที่มีเกรดเฉลี่ยสะสมน้อยกว่า 1.5 หรือมากกว่าเท่ากับ 3.0 และเคยติดเชื้อ Covid-19 คนที่มีคะแนนความเครียด (stress) สูงและต่ำที่สุดคือใคร

5. Summarise

Summarise stat using dplyr

Method	Test1	Test2	Sum
LBL	4	6	10
LBL	6	6	12
PBL	7	8	15
PBL	8	9	17
INQ	8	8	16
INQ	9	7	16

N	M	SD	Min	Max
6	14.3	2.7	10	17

```
summarise(data,n=n(),M=mean(Sum),SD=sd(Sum),min=min(Sum),max=max(Sum)
```

Summarise functions

- Center: `mean()`, `median()`
- Spread: `sd()`, `IQR()`, `mad()`
- Range: `min()`, `max()`, `quantile()`
- Count: `n()`, `n_distinct()`

Summarise stat using dplyr

จงหาค่าสถิติพื้นฐานของความเครียด และ gpax ในกลุ่มนักเรียนที่ไม่เคยติดเชื้อ Covid-19

```
mytidy2 %>%
  summarise(mean.stress = mean(stress),
            sd.stress = sd(stress),
            min.stress = min(stress),
            max.stress = max(stress),
            mean.gpax = mean(gpax.y2),
            sd.gpax = sd(gpax.y2),
            q1.gpax = quantile(gpax.y2, 0.25),
            q3.gpax = quantile(gpax.y2, 0.75))
```

Summarise stat using dplyr

```
                [,1]
mean.stress 2.1800000
sd.stress   1.0298697
min.stress  1.0000000
max.stress  5.0000000
mean.gpax   2.5125000
sd.gpax     0.6866385
q1.gpax     2.1250000
q3.gpax     3.0350000
```

Summarise stat using dplyr

ผู้วิเคราะห์ต้องการหาค่าเฉลี่ย และส่วนเบี่ยงเบนมาตรฐานของจำนวนอุปกรณ์ it ที่นักเรียนมี (`it.equip`) เมื่อดำเนินการคำนวณพบว่าได้ผลลัพธ์ดังนี้

```
mytidy2 %>%
  summarise(mean.it = mean(it.equip),
            sd.it = sd(it.equip))
```

```
  mean.it sd.it
1      NA    NA
```

ท่านคิดว่าปัญหาข้างต้นเกิดจากอะไร?

group_by() function

สมมติว่า
ต้องการหาค่าเฉลี่ยและส่วนเบี่ยงเบนมาตรฐานของคะแนนความเครียดระหว่างกลุ่มที่ไม่มีปัญหา และ มีปัญหาเกี่ยวกับสัญญาณ internet

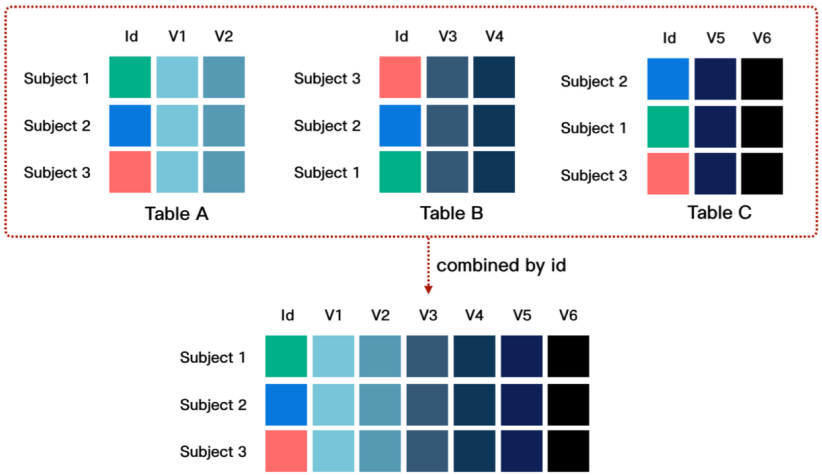
```
1 mytidy2 %>%
2   group_by(internet) %>%
3   summarise(mean = mean(stress),
4             sd = sd(stress))
```

```
# A tibble: 2 x 3
  internet       mean    sd
  <fct>         <dbl> <dbl>
1 no problem     1.88 0.832
2 have problem   2.28 1.09
```

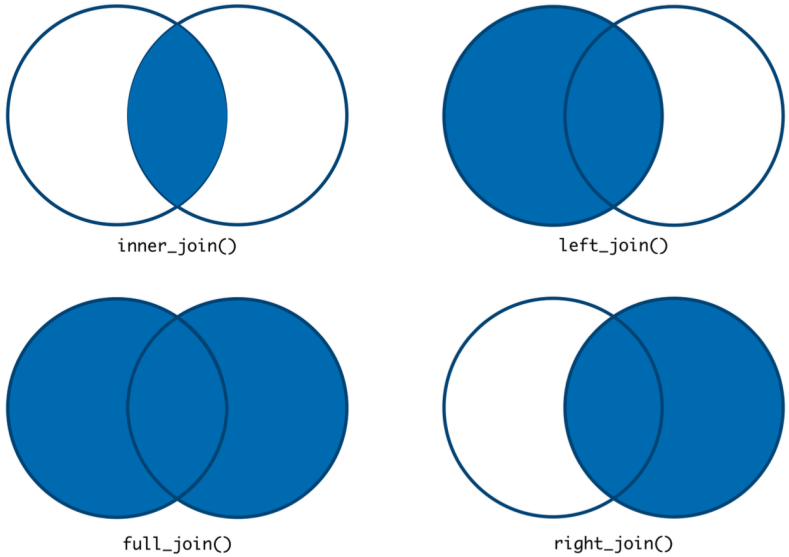
6. Merging datasets

Merging datasets

การรวมตารางข้อมูลตามคอลัมน์



ประเภทการรวมข้อมูล



ที่มา : สิวะโชติ ศรีสุทธิยากร (2564)

ตัวอย่าง

```r
X<-data.frame(id=c(1,2,3),
              x1=c(10,20,30),
              x2=c(5,7,9))
Y<-data.frame(id=c(1,2,5),
              y1=c(20,40,60),
              y2=c("F","M","F"))
```

```r
X
```

```
  id x1 x2
1  1 10  5
2  2 20  7
3  3 30  9
```

```r
Y
```

```
  id y1 y2
1  1 20  F
2  2 40  M
3  5 60  F
```

ตัวอย่าง : full_join() vs inner_join()

```
1 full_join(X,Y, by="id")
```

```
  id x1 x2 y1   y2
1  1 10  5 20    F
2  2 20  7 40    M
3  3 30  9 NA <NA>
4  5 NA NA 60    F
```

```
1 inner_join(X,Y, by="id")
```

```
  id x1 x2 y1 y2
1  1 10  5 20  F
2  2 20  7 40  M
```

Merging mytidy and mytidy2

```
1 head(mytidy,3)
```

```
# A tibble: 3 x 4
  student_name method   post   pre
  <chr>        <chr>   <dbl> <dbl>
1 Adam Warlock Lecture    34    16
2 Adam Warlock PBL        NA    NA
3 Ancient One  Lecture    45    20
```

```
1 head(mytidy2[,1:4],3)
```

```
    X            name infect stress.1
1  20    Adam Warlock      2        3
2 973     Ancient One      2        3
3 520 Captain America      2        3
```

Merging mytidy and mytidy2

```
full_mytidy <- full_join(mytidy,
                         mytidy2,
                         by=c("student_name" = "name"))
```

Merging mytidy and mytidy2

```
1 glimpse(full_mytidy)
```

```
Rows: 40
Columns: 21
$ student_name <chr> "Adam Warlock", "Adam Warlock", "Ancie
$ method       <chr> "Lecture", "PBL", "Lecture", "PBL", "I
$ post         <dbl> 34, NA, 45, NA, NA, 67, 50, NA, NA, 93
$ pre          <dbl> 16, NA, 20, NA, NA, 18, 13, NA, NA, 25
$ X            <int> 20, 20, 973, 973, 520, 520, 47, 47, 86
$ infect       <int> 2, 2, 2, 2, 2, 2, 2, 2, 1, 1, 2, 2, 1,
$ stress.1     <int> 3, 3, 3, 3, 3, 3, 2, 2, 3, 3, 1, 1, 2,
$ stress.2     <int> 2, 2, 2, 2, 2, 2, 1, 1, 2, 2, 1, 1, 1,
$ stress.3     <int> 3, 3, 3, 3, 3, 3, 1, 1, 3, 3, 1, 1, 2,
$ stress.4     <int> 2, 2, 3, 3, 2, 2, 1, 1, 4, 4, 1, 1, 1,
$ stress.5     <int> 2, 2, 1, 1, 1, 1, 1, 1, 2, 2, 1, 1, 1,
```

Q & A