✪ Psychological Statistics ✪

Data Arrangement

- Edited by Prof. **Changwei Wu**
- Graduate Institute of Mind, Brain and Consciousness (***GIMBC***), Taipei Medical University

In [ ]:
```
### [ Setup the working directory ]

setwd("/Users/wesley/[Course]/Python/R_Script")
# → Please edit the directory name in your computer.
getwd()
```

In [ ]:
```
### [ Loading the required libraries ]

#install.packages("dplyr")
#install.packages("readxl")
#install.packages("reshape2")
#install.packages("rstatix")

library("dplyr")
library("readxl")
library("reshape2")
library("rstatix")
```

---

(1) Loading Datasets

In [ ]:
```
### 1-1.[ Import data from CSV data files ]
# → Function: read.csv {utils}

Exp<-read.csv("Ex1.csv",header=T)
class(Exp)

# → "data frame" is a 2-D datasheet, like the data storage in an Excel file

Exp[1:5,]

# Exp %>% head(5) # same function
```

In [ ]:
```
### 1-2.[ Import data from .dat files ]
# → Function: read.delim {util}

clownData<-read.delim("coulrophobia.dat", header = TRUE)
```

In [ ]:
```
### 1-3.[ Import data from Excel files ]
# → Function: excel_sheets or read_excel {readxl}

excel_sheets('ExcelExample.xlsx')
tomatoXL <- read_excel('ExcelExample.xlsx')
wineXL1 <- read_excel('ExcelExample.xlsx', sheet=2)
```

In [ ]:
```r
### 1-4.[ Export data to files ]
# → Function: write.csv or write.table {utils}

write.table(select, "Table.txt", sep="\t", row.names = FALSE)
write.csv(select, "TEST.csv")
```

(2) Data arrangement / munging

In [ ]:
```r
### 2-1.[ Load diamonds ]

data(diamonds, package = 'ggplot2')

diamonds %>% head(5)
```

In [ ]:
```r
### 2-2.[ Apply functions ]

# → Function: lapply / sapply {base}

sapply(diamonds, mean)
```

In [ ]:
```r
### 2-3.[ Group by function ]

# → Function: aggregate {stats}

aggregate(price ~ cut, diamonds, mean)

# diamonds %>% group_by(cut) %>% summarize(AvgPrice=mean(price))
```

In [ ]:
```r
### 2-4.[ Relevel within variable ]

# → check the category names within a variable
levels(diamonds$color)

# → relevel Function: factor {base}

diamonds$color<-factor(diamonds$color, levels = levels(diamonds$color)[c(7, 1:6
levels(diamonds$color)
```

In [ ]:
```r
### 2-5.[ Rename categories within variable ]

# → check the category names within a variable
levels(diamonds$color)

# → Function: factor {base}

diamonds$color<-factor(diamonds$color, labels = c("Red", "White", "Blue", "Gree
levels(diamonds$color)
```

In [ ]:
```r
### 2-6.[ Keeping/removing variables ]

# → Function: select {dplyr}

Neo1 <- select(diamonds, carat, price)
Neo1 %>% head(5)

Neo2 <- diamonds %>% select(starts_with('c'))
# options: 'starts_with', 'ends_with', 'contains'
Neo2 %>% head(5)

Neo3 <- diamonds %>% select(-c('carat', 'price'))
Neo3 %>% head(5)
```

In [ ]:
```r
### 2-7.[ Selecting data/subjects ]

levels(diamonds$cut)

# → Function: filter {dplyr}

diamonds %>% filter(cut == 'Ideal') %>% head(5)

diamonds %>% filter(cut == 'Good' | price > 1000) %>% head(5) # '|'→ or

diamonds %>% filter(cut %in% c('Ideal', 'Good')) %>% head(5)
```

In [ ]:
```r
### 2-8.[ Selecting data/subjects ]

# → Function: slice {dplyr}

diamonds %>% slice(1:5)

diamonds %>% slice(-1) %>% head(5)
```

In [ ]:
```r
### 2-9.[ Creating new variables ]

# → Function: mutate {dplyr}

diamonds %>% mutate(price/carat) %>% head(5)

diamonds %>% select(carat, price) %>%
    mutate(ratio = price/carat) %>% arrange(ratio) %>% head(5)
```

---

(3) Restructure into Data Frame

In [ ]:
```
### 3-1.[ Change data from wide to long format ]

(spider <- read.delim("SpiderWide.dat"))

# → Now we change it into long format

spider.long <- spider %>%
    gather(key = "View", value = "Anxiety", picture, real)

spider.long %>% head(5)

# (melt00 <- reshape2::melt(spider, variable.name="View", value.name="Anxiety")
```

In [ ]:
```
### 3-2.[ Changing between the long form and the wide form ]
# → Function: melt/dcast {reshape2} & str_sub {stringr}

library("stringr")
library("reshape2")

Aid_00s <- read.csv("US_Foreign_Aid_00s.csv")
melt00 <- melt(Aid_00s, id.vars=c("Country.Name", "Program.Name"),
               variable.name="Year", value.name="Dollars")

melt00$Year <- as.numeric(str_sub(melt00$Year, start=3, 6))

cast00 <- dcast(melt00, Country.Name + Program.Name ~ Year, value.var = "Dollar

head(Aid_00s)
head(cast00)
```