國立臺灣師範大學
National Taiwan Normal University

Basics in R

21 Aug 2024

Long count: 13.0.11.15.1 (1 Imox)

Sia Ming Yean

siayean@ntnu.edu.tw
@MYSia05381231

Workshop overview

Part 1: Getting to know R

- Basic operations in R

Part 2: Hands-on practice

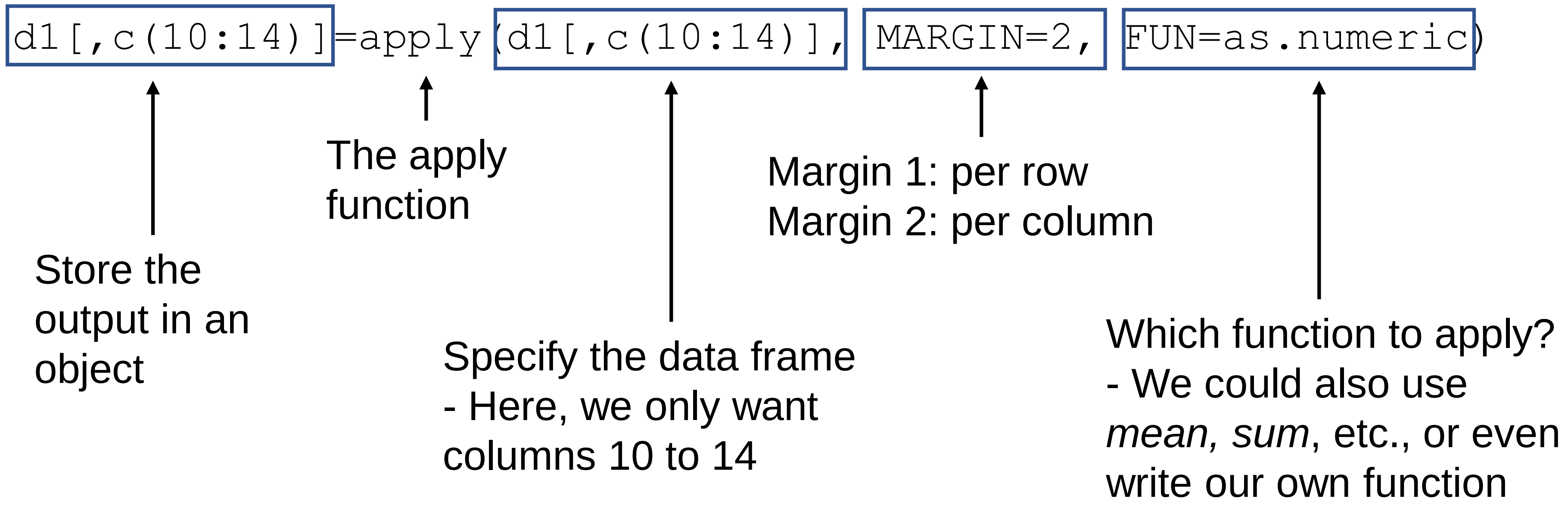
- Work on sample data
- Plot pretty graphs

Part 3: Learning to be independent

- Seek help from the web

Recap

`apply` function



```
d1[,c(10:14)]=apply(d1[,c(10:14)], MARGIN=2, FUN=as.numeric)
```

Store the output in an object

The apply function

Specify the data frame
- Here, we only want columns 10 to 14

Margin 1: per row
Margin 2: per column

Which function to apply?
- We could also use *mean, sum*, etc., or even write our own function

`merge` **function**

```
d.all=merge(d1, d2, by="id", all=T)
```

The merge function

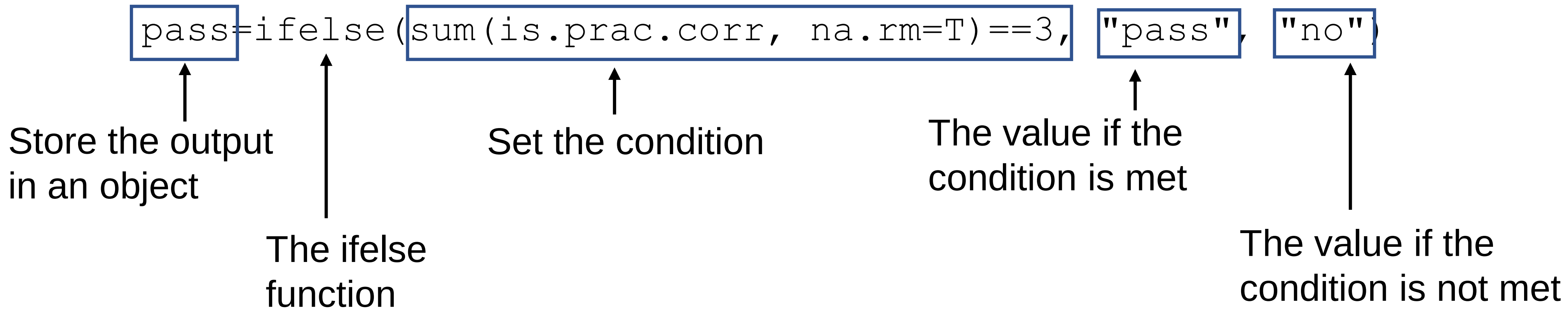
Store the output in an object

Specify the two data frames that we want to merge

Merge by which variable? If the variable names are different, use `by.x` and `by.y`

Do we want to keep all data points (even if it does not exist in one of the data frames)? We can also use `all.x` and `all.y`

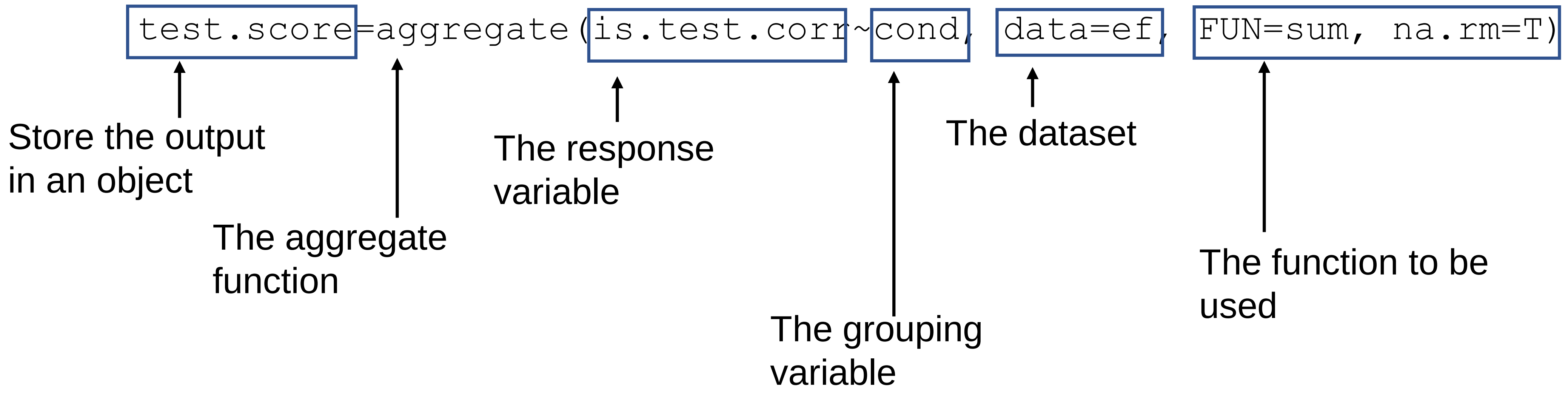
`ifelse` function



```
pass=ifelse(sum(is.prac.corr, na.rm=T)==3, "pass", "no")
```

Store the output in an object
The ifelse function
Set the condition
The value if the condition is met
The value if the condition is not met

`aggregate` **function**



```
test.score=aggregate(is.test.corr~cond, data=ef, FUN=sum, na.rm=T)
```

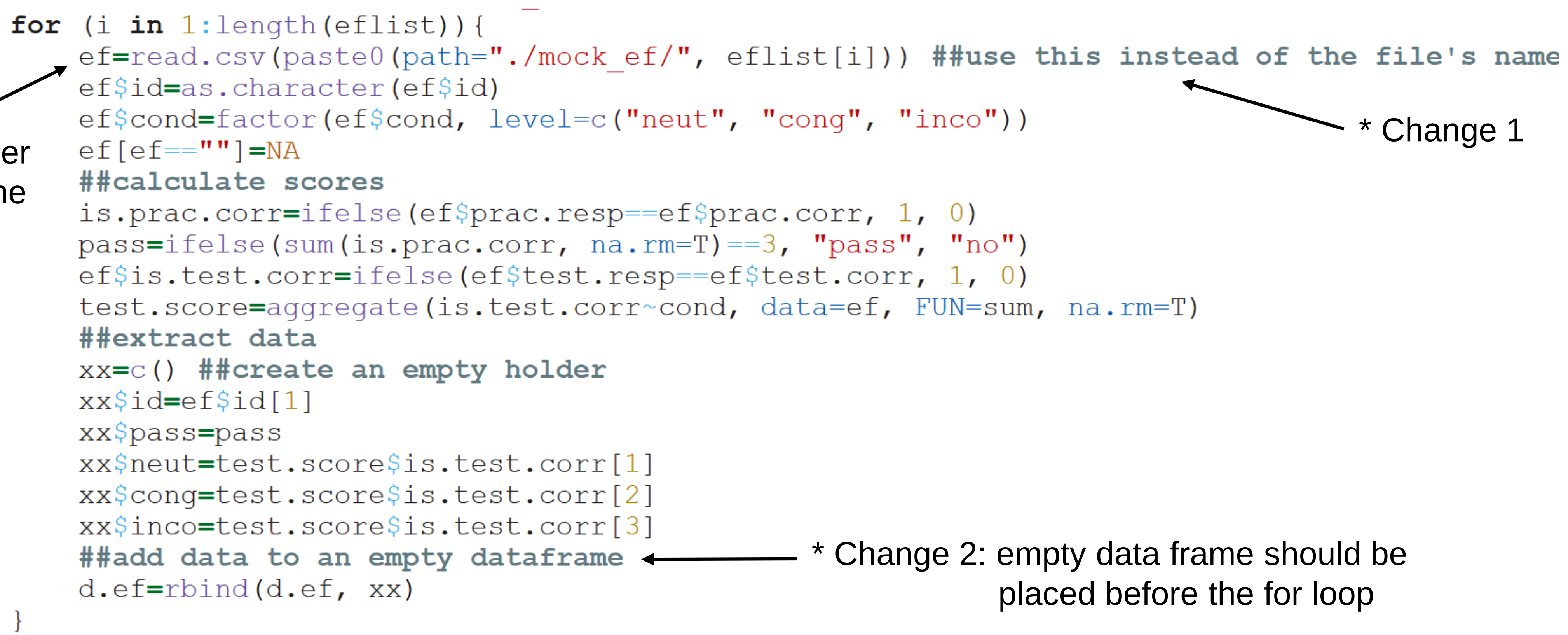
for loop

- This is how a `for loop` works in general:

```
for (i in 1:n){
    ...
}
```

You can read this as “for *i* in 1 to *n*”, where
- *n* is the total number of items
- *i* is the index of each of these items

What we want to do with the items



```
for (i in 1:length(eflist)){
    ef=read.csv(paste0(path="./mock_ef/", eflist[i])) ##use this instead of the file's name
    ef$id=as.character(ef$id)
    ef$cond=factor(ef$cond, level=c("neut", "cong", "inco"))
    ef[ef==""]=NA
    ##calculate scores
    is.prac.corr=ifelse(ef$prac.resp==ef$prac.corr, 1, 0)
    pass=ifelse(sum(is.prac.corr, na.rm=T)==3, "pass", "no")
    ef$is.test.corr=ifelse(ef$test.resp==ef$test.corr, 1, 0)
    test.score=aggregate(is.test.corr~cond, data=ef, FUN=sum, na.rm=T)
    ##extract data
    xx=c() ##create an empty holder
    xx$id=ef$id[1]
    xx$pass=pass
    xx$neut=test.score$is.test.corr[1]
    xx$cong=test.score$is.test.corr[2]
    xx$inco=test.score$is.test.corr[3]
    ##add data to an empty dataframe
    d.ef=rbind(d.ef, xx)
}
```

* Remember to indent the codes

* Change 1

* Change 2: empty data frame should be placed before the for loop

Plot

```
##plot 1: scatterplot
par(mar=c(3.4, 3.4, 0.5, 0.5), mgp=c(1.8, 0.5, 0), tcl=-0.3, las=1)
plot(x=x.pass$age, y=x.pass$read.dur, #ylim=c(0, 10),
     xlab="Age (year)", ylab="Reading duration (minutes)",
     pch=8, cex=1.3, lwd=2, col="blue",
     cex.axis=1.3, cex.lab=1.8)
##add line
age.read=lm(x.pass$read.dur~x.pass$age)
abline(age.read, lty=2, lwd=3)
```

To save/export the plot, you can either

- Click plots -> export
- Or use functions like `jpeg()` or `pdf()`

```
jpeg("plot1.jpg", width=500)
```
... (the plotting script)
```
dev.off()
```

Plotting parameters

- To place both plots side-by-side:

*Added more space

*Added title

```
##both plots together
par(mfrow=c(1, 2), mar=c(3.4, 3.6, 2, 2), mgp=c(2, 0.5, 0), tcl=-0.3, las=1)
##plot reading duration
plot(x.pass$grp, x.pass$read.dur, main="Reading duration",
     xlab="Group", ylab="Reading duration (minutes)", xaxt="n",
     cex.axis=1.3, cex.lab=1.6, lwd=2, col="deepskyblue")
mtext(text=c("interested", "not interested"), at=c(1, 2), side=1,
      line=0.4, cex=1.3)
##plot brain activation
plot(x.pass$grp, x.pass$avg.hbo, main="Brain activation",
     xlab="Group", ylab="Brain activation (HbO)", xaxt="n",
     cex.axis=1.3, cex.lab=1.6, lwd=2, col="deepskyblue")
mtext(text=c("interested", "not interested"), at=c(1, 2), side=1,
      line=0.4, cex=1.3)
```

Part 3
Learning to be independent

Some useful websites for further reading

- https://learningstatisticswithr.com/book/introR.html
- https://www.learnbyexample.org/r-introduction/

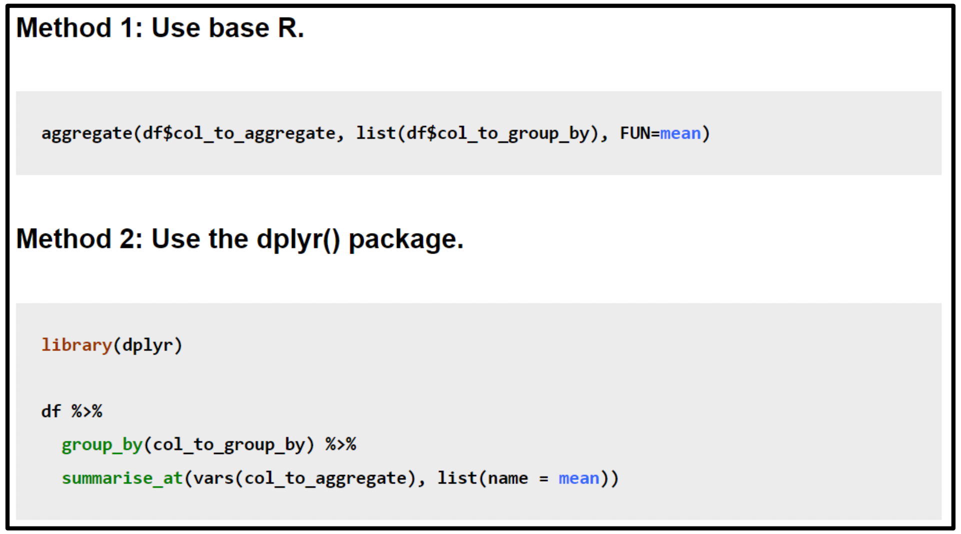
- Or, you could just google what you need, like what I always do.
- You don’t need to know the codes by heart. You just need to know what tools are available, and then google the codes whenever you need to use them.
- You can also have a snippets file to save codes that you reuse frequently.
- Practice makes perfect: it will become easier after several rounds of coding

Widely used packages

- Many R users (whom I know) use *tidyverse* to clean their data and *ggplot2* for plotting.
- Some R users find *tidyverse* to be more intuitive, but I think it is because they are more familiar with *tidyverse*.
- I prefer to use base R but this is just a personal preference.
- I would say to go for whatever that you are most comfortable with.
- Most websites provide solutions using base R and *tidyverse*:
  - Google: How to calculate means by group?
  - https://www.statology.org/r-mean-by-group/
- I like this website when it comes to plotting:

  www.r-graph-gallery.com

https://www.tidyverse.org/
https://dplyr.tidyverse.org/
https://ggplot2.tidyverse.org/



Method 1: Use base R.

```
aggregate(df$col_to_aggregate, list(df$col_to_group_by), FUN=mean)
```

Method 2: Use the dplyr() package.

```
library(dplyr)

df %>%
  group_by(col_to_group_by) %>%
  summarise_at(vars(col_to_aggregate), list(name = mean))
```

Data mining & PCA

- https://www.geeksforgeeks.org/data-mining-in-r/
- https://www.tutorialspoint.com/exploring-data-mining-with-r
- https://rpubs.com/uriel623/670548
- https://www.geeksforgeeks.org/confirmatory-factor-analysis-in-r/
- I won't go into details for two main reasons:
  - They're not something I usually do, so I'm not familiar with these topics (I'd be happy to talk about mixed effect models though)
  - My aim is to cover the basics in R, so they're beyond the workshop's scope
- The links above give good explanation and examples and they build upon what I presented today (except that they use *dplyr* while I use base R to achieve the same thing)

Thank you!
That’s the end of the workshop.

Please fill in the feedback form:

https://forms.gle/3eaJEAWYUD39auZ9A

✉ siayean@ntnu.edu.tw
@MYSia05381231