Applied Epidemiology I: Data Management

Enoch Yi-Tung Chen

Department of Medical Epidemiology and Biostatistics, Karolinska Insitutet

January 28, 2021

About me

- First came to Sweden as an exchange at Lund University, Spring 2018
- MSc in Public Health Sciences, Epi, KI (2018-2020)

About me

- First came to Sweden as an exchange at Lund University, Spring 2018
- MSc in Public Health Sciences, Epi, KI (2018-2020)
- Joined MEB as a thesis student in Jan 2020 and then Covid came
- Continued as a research assistant from June 2020 till now

About me

- First came to Sweden as an exchange at Lund University, Spring 2018
- MSc in Public Health Sciences, Epi, KI (2018-2020)
- Joined MEB as a thesis student in Jan 2020 and then Covid came
- Continued as a research assistant from June 2020 till now
- Fascinated by population-based epi studies
- Interested in stat methods in survival analysis and health economics

About me

- First came to Sweden as an exchange at Lund University, Spring 2018
- MSc in Public Health Sciences, Epi, KI (2018-2020)
- Joined MEB as a thesis student in Jan 2020 and then Covid came
- Continued as a research assistant from June 2020 till now
- Fascinated by population-based epi studies
- Interested in stat methods in survival analysis and health economics
- I love animals, so don't be surprised to see them in some example.

Something to know about Stata session

- It is my FIRST time to run a course.
- Will teach all the labs in Stata along with exercises and Q&A (see the schedule)
- Other softwares are welcome to use, but I may not be able to answer your questions on them. (I mainly use Stata or R.)

Something to know about Stata session

- It is my FIRST time to run a course.
- Will teach all the labs in Stata along with exercises and Q&A (see the schedule)
- Other softwares are welcome to use, but I may not be able to answer your questions on them. (I mainly use Stata or R.)
- Previous materials, including teaching videos, can be found at `https://enochytchen.com/courses/biostatbasics/`.
- But please refer to Canvas for the latest materials for this year's course.

Something to know about Stata session

- It is my FIRST time to run a course.
- Will teach all the labs in Stata along with exercises and Q&A (see the schedule)
- Other softwares are welcome to use, but I may not be able to answer your questions on them. (I mainly use Stata or R.)
- Previous materials, including teaching videos, can be found at `https://enochytchen.com/courses/biostatbasics/`.
- But please refer to Canvas for the latest materials for this year's course.
- Questions are welcome. But please give me codes (and log files) and 2-3 working days. enoch.yitung.chen@ki.se

Acknowledgements

This course material in data management is based on my learning from Anna Johansson's workshop at KI library[1], teachings in Good Data Management Practice in Epidemiological Research, and MEB Guidelines for Documentation and Archiving Version 6 [2]. I personally want to thank for their effort on education in data management.
I especially want to thank Marlene Stratmann for reviewing the slides and Prof. Paul Dickman for providing me with suggestions to improving the teaching.

[1] This workshop is currently available on KI Play as well.

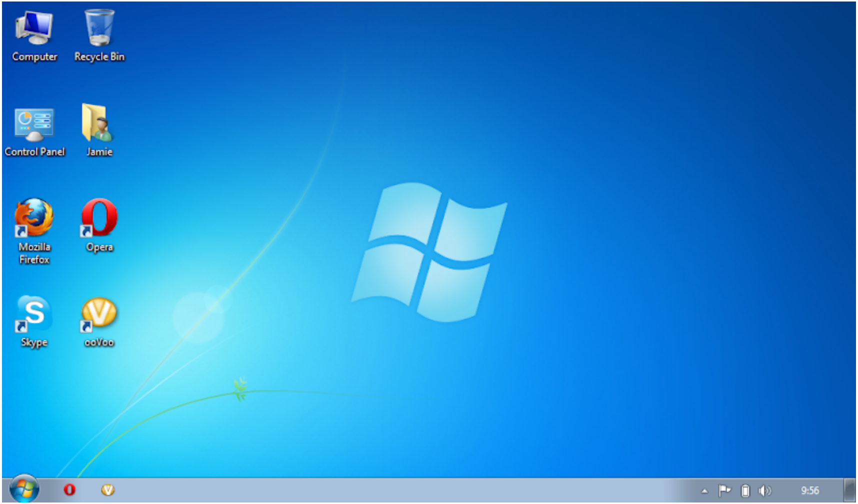
[2] The Department of Medical Epidemiology and Biostatistics, Karolinska Institutet. MEB Guidelines for Documentation and Archiving Version 6. 2018.

Outline

1. What if no data management?
2. Aims of data management (also learning outcomes)
3. Good folder structure
4. Good documents
5. Good Readme.txt
6. Good master.do
7. Good habits on coding
8. Other do's and don'ts
9. Wrap it up

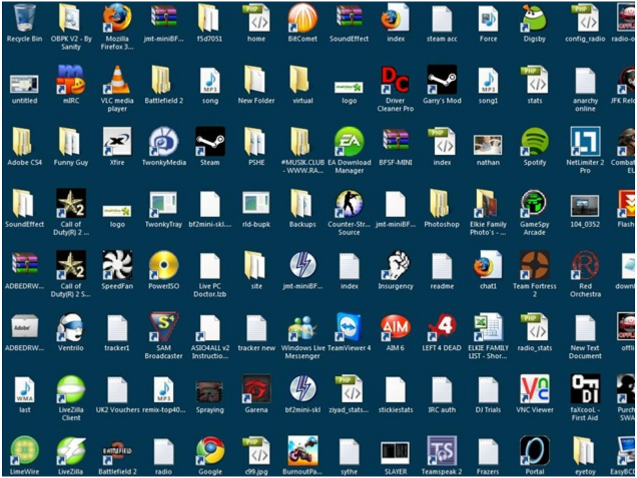
What if no data management?

In the beginning,



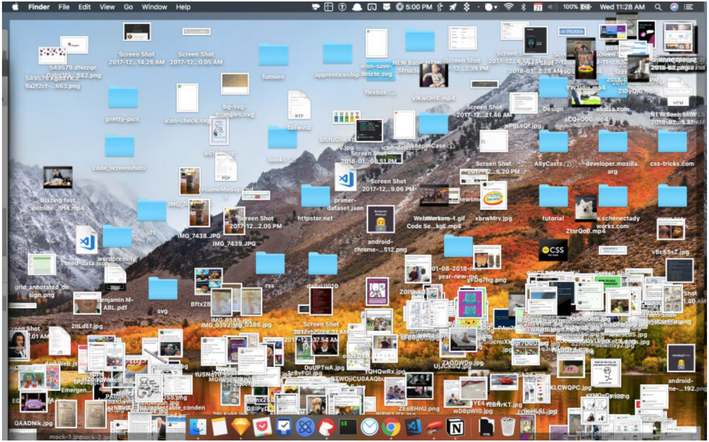
What if no data management?

On the half-way of the research,



What if no data management?

At the end, or saying you cannot even walk till the end?



What if no data management?

Imagine now

- if you want to correct Table I, where is the do file for descriptive analysis?

What if no data management?

Imagine now

- if you want to correct Table I, where is the do file for descriptive analysis?
- if your supervisor says, ”Please summarise how far you’ve gone in this project.” You probably cannot just drop him/her your syntax.

What if no data management?

Imagine now

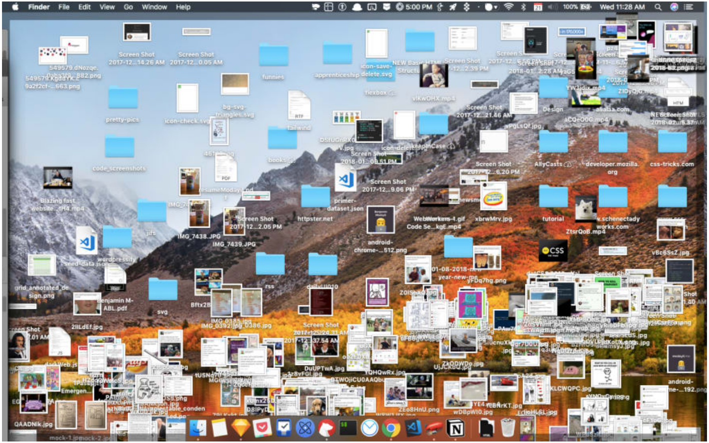
- if you want to correct Table I, where is the do file for descriptive analysis?
- if your supervisor says, "Please summarise how far you've gone in this project." You probably cannot just drop him/her your syntax.
- if your classmate asks you to teach her how to write a certain Stata code, you remember you've done it before, but where did you put it?

What if no data management?

Imagine now

- if you want to correct Table I, where is the do file for descriptive analysis?
- if your supervisor says, "Please summarise how far you've gone in this project." You probably cannot just drop him/her your syntax.
- if your classmate asks you to teach her how to write a certain Stata code, you remember you've done it before, but where did you put it?
- if your collaborator needs to take over your analysis, can he/she understand what you've completed?

What if no data management?



What if no data management?

So I would say you need to have a friend called

Data Management

Aims of data management (also learning outcomes)

- To ensure the analysis is reproducible

Aims of data management (also learning outcomes)

- To ensure the analysis is reproducible
- To work coherently and efficiently with yourself

Aims of data management (also learning outcomes)

- To ensure the analysis is reproducible
- To work coherently and efficiently with yourself
- To ensure the project can be understood by others (supervisors, collaborators, and future readers)

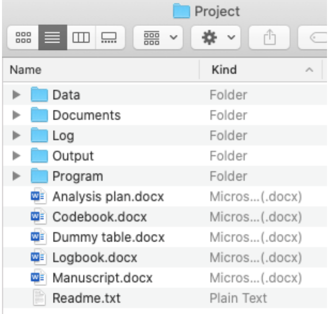
Aims of data management (also learning outcomes)

- To ensure the analysis is reproducible
- To work coherently and efficiently with yourself
- To ensure the project can be understood by others (supervisors, collaborators, and future readers)
- To create a good work flow and enhance accuracy of work

Good folder structure

The core elements of folders are listed below:

- Data
- Documents
- Log
- Output
- Program



Project

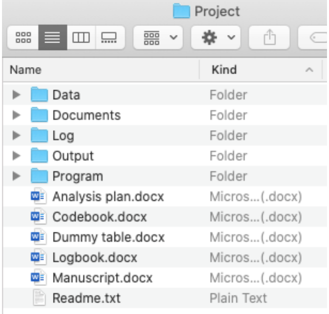
Name	Kind
Data	Folder
Documents	Folder
Log	Folder
Output	Folder
Program	Folder
Analysis plan.docx	Micros...(.docx)
Codebook.docx	Micros...(.docx)
Dummy table.docx	Micros...(.docx)
Logbook.docx	Micros...(.docx)
Manuscript.docx	Micros...(.docx)
Readme.txt	Plain Text

Figure: Good project folder structure. (Please bear with me that I am Mac user!)

Good documents

Besides good folder structure, you should also consider keeping good documents

- Analysis plan
- Codebook[3]
- Dummy table
- Logbook[3]
- Manuscript



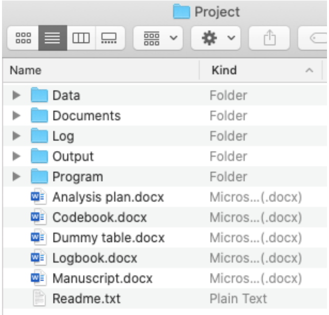
Name	Kind
Data	Folder
Documents	Folder
Log	Folder
Output	Folder
Program	Folder
Analysis plan.docx	Micros...(.docx)
Codebook.docx	Micros...(.docx)
Dummy table.docx	Micros...(.docx)
Logbook.docx	Micros...(.docx)
Manuscript.docx	Micros...(.docx)
Readme.txt	Plain Text

Figure: Good project folder structure.

[3] can be included in analysis plan as well

Good Readme.txt

- You should illustrate how to use these documents/folders in the Readme.txt.
- A good Readme.txt is a good tourist guide in this project folder.

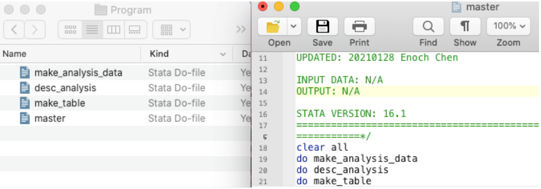


Name	Kind
Data	Folder
Documents	Folder
Log	Folder
Output	Folder
Program	Folder
Analysis plan.docx	Micros...(.docx)
Codebook.docx	Micros...(.docx)
Dummy table.docx	Micros...(.docx)
Logbook.docx	Micros...(.docx)
Manuscript.docx	Micros...(.docx)
Readme.txt	Plain Text

Figure: Good project folder structure.

Good master.do

- master.do file tells the order of executing the do files.
- Do not do all the analyses in the same do file.
- Separate them and use master.do to organise them.



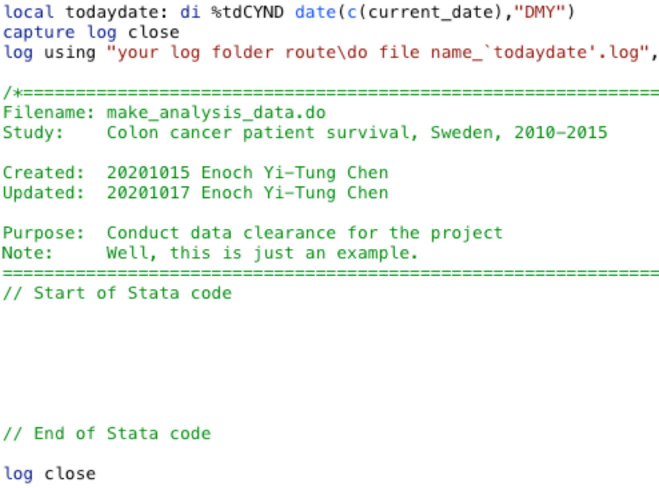
Name	Kind	Da
make_analysis_data	Stata Do-file	Ye
desc_analysis	Stata Do-file	Ye
make_table	Stata Do-file	Ye
master	Stata Do-file	Ye

```
UPDATED: 20210128 Enoch Chen

INPUT DATA: N/A
OUTPUT: N/A

STATA VERSION: 16.1
==========================================
===========*/
clear all
do make_analysis_data
do desc_analysis
do make_table
```

Figure: Once you execute master.do, it will run all the specified do-files.

Good habit on coding

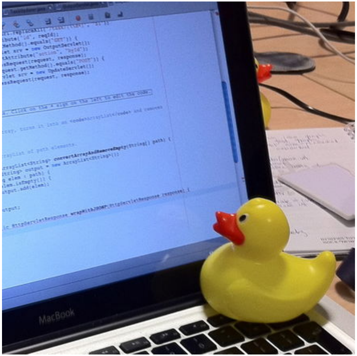
- **log on**
- Filename
- Study
- Created
- Updated
- Purpose
- Note
- **Program**
- log close



```
local todaydate: di %tdCYND date(c(current_date),"DMY")
capture log close
log using "your log folder route\do file name_`todaydate'.log",

/*=============================================================
Filename: make_analysis_data.do
Study:    Colon cancer patient survival, Sweden, 2010-2015

Created:  20201015 Enoch Yi-Tung Chen
Updated:  20201017 Enoch Yi-Tung Chen

Purpose:  Conduct data clearance for the project
Note:     Well, this is just an example.
===============================================================
// Start of Stata code






// End of Stata code

log close
```

Good habit on coding

- Talk to yourself what you are doing.
- You've got a friend in me! (Parallel analysis)
- Rubber duck debugging



Other do's and don'ts

1. Use a shared drive/project server.
   (Required to do that because of data privacy.)

Other do's and don'ts

1. Use a shared drive/project server.
   (Required to do that because of data privacy.)
2. Give appropriate names to your files and variables.

Other do's and don'ts

1. Use a shared drive/project server.
   (Required to do that because of data privacy.)
2. Give appropriate names to your files and variables.
   - No stupid names, such as new1, new2, new3, final1, final2, final3, latest1

Other do's and don'ts

1. Use a shared drive/project server.
   (Required to do that because of data privacy.)
2. Give appropriate names to your files and variables.
   - No stupid names, such as new1, new2, new3, final1, final2, final3, latest1
   - No space, special character, dots (in case, the software cannot read.)

Other do's and don'ts

1. Use a shared drive/project server.
   (Required to do that because of data privacy.)
2. Give appropriate names to your files and variables.
   - No stupid names, such as new1, new2, new3, final1, final2, final3, latest1
   - No space, special character, dots (in case, the software cannot read.)
   - For binomial variables, $= 1$ implies yes, and $= 0$ implies no.

Other do's and don'ts

1. Use a shared drive/project server.
   (Required to do that because of data privacy.)
2. Give appropriate names to your files and variables.
   - No stupid names, such as new1, new2, new3, final1, final2, final3, latest1
   - No space, special character, dots (in case, the software cannot read.)
   - For binomial variables, $= 1$ implies yes, and $= 0$ implies no.
   - Label your variables, please!

Other do's and don'ts

1. Use a shared drive/project server.
   (Required to do that because of data privacy.)
2. Give appropriate names to your files and variables.
   - No stupid names, such as new1, new2, new3, final1, final2, final3, latest1
   - No space, special character, dots (in case, the software cannot read.)
   - For binomial variables, $= 1$ implies yes, and $= 0$ implies no.
   - Label your variables, please!
3. Same names for linking files (.do .r .sas → .log → .doc)

Other do's and don'ts

1. Use a shared drive/project server.
   (Required to do that because of data privacy.)
2. Give appropriate names to your files and variables.
   - No stupid names, such as new1, new2, new3, final1, final2, final3, latest1
   - No space, special character, dots (in case, the software cannot read.)
   - For binomial variables, $= 1$ implies yes, and $= 0$ implies no.
   - Label your variables, please!
3. Same names for linking files (.do .r .sas → .log → .doc)
4. Don't replace the original files or variables.

Other do's and don'ts

1. Use a shared drive/project server.
   (Required to do that because of data privacy.)
2. Give appropriate names to your files and variables.
   - No stupid names, such as new1, new2, new3, final1, final2, final3, latest1
   - No space, special character, dots (in case, the software cannot read.)
   - For binomial variables, $= 1$ implies yes, and $= 0$ implies no.
   - Label your variables, please!
3. Same names for linking files (.do .r .sas → .log → .doc)
4. Don't replace the original files or variables.
5. Don't edit the data directly. Please write syntax.

Wrap it up

- In summary, a good data management contains GOOD
  1. folder structure
  2. documents
  3. readme
  4. habits

Wrap it up

- In summary, a good data management contains GOOD
  1. folder structure
  2. documents
  3. readme
  4. habits
- How can this lecture help you?
- The templates you can use for DM your current and future projects.