ETF3231/5231 Week 2 - Time series data, `tsibble` objects and time series graphics

Sao Yang Hew

Friday - 10 March 2023 (12:30pm - 2:00pm)

2023

Table of Contents

1. Assignment and pipe operators
2. Tidy data
3. `tsibble` objects
4. Plotting
5. Reading and writing files
6. Changing plot axes

Assignment operator

- `x <- 1 + 3` (i.e., x is 1 + 3, or assign 1 + 3 to x)
- Can support more complex commands (e.g., x is a table created from these following sets of commands).
- Shortcuts:
  - Windows: Alt + -
  - Mac: Option + -

Pipe operator

- In base R: `f(x, ...)` means **applying x and some other arguments (...) on x to a function f.**
- If we use a pipe (%>%), `x %>% f(...)`
- Shortcuts:
  - Windows: Ctrl + Shift + M
  - Mac: Cmd + Shift + M
- If we stack some functions on top of another
  - `f4(f3(f2(f1(x, ...),...),...),...)`
  - `x %>% f1(...)  %>% f2(...)  %>% f3(...)  %>% f4(...)`
  - Whenever possible, use the 2nd option for readability.

Asking for help in R

- `help(x)` or `?x`: gives description of functions or datasets within R documentations provided by authors.

What is tidy data?

- `tibbles` are a table format in R (other formats include `data.frame` or `data.table`).
- It is great to use as they are quick, reproducible, and have a structured back-end so users are not worried about making mistakes they cannot detect.
- `tibbles` have the unique feature of having two attributes within the table:
  - keys: identifiers of an observation's characteristics: e.g., nationality, customer ID, university major
  - values: often numeric, measured indicator of said observation: e.g., height, weight, income
- For a dataset to be "tidy" (to be easily processed in any programming language), it is recommended that:
  - keys are presented in rows
  - values are presented in columns

tsibble objects

- tsibble objects are similar to tibbles. They have keys and values, it will be ideal that you keep them in a tidy format, but there is an extra component in the table called the **index** that represents time.
- Index can be of different frequencies (e.g., quarterly, daily, monthly), regular or irregular.

pivot_longer

```
table2 <- table1 %>% pivot_longer(
  !variable1, !variable2,
  names_to = "key1",
  values_to = "value1"
)
```

pivot_wider

```
table1 <- table2 %>% pivot_wider(
  names_from = key1,
  values_from = value1
)
```

A few useful functions to work with tibbles/tsibbles

- `filter()`: selects specific rows
- `select()`: selects specific columns
- `mutate()`: creates new variables
- `group_by() %>% summarise()`: gives you summary statistics of some variables

A few plot functions for time series

- `autoplot(Variable)`
- `gg_season(Variable)`
- `gg_subseries(Variable)`

Faceting

```
table2 %>% autoplot(Variable) +
  facet_grid(Variable ~ ., scales = 'free')
```

How would this work then?

```
table2 %>% autoplot(Variable) +
  facet_grid(. ~ Variable, scales = 'free')
```

Few ways to read a few files

- `read_csv()`
- `readxl::read_excel()`
- Download files online (e.g., from the fpp3 textbook):

```
download.file("http://OTexts.com/fpp3/extrafiles/tute1.csv",
  tute1 <- tempfile())

tute1 <- read_csv(tute1)
```

Need to make them tibbles (or tsibbles), depending on purpose or function used using `as_tibble()` or `as_tsibble()`

The labs function

```
table2 %>% autoplot(Variable) +
  facet_grid(Variable ~ ., scales = 'free') +
  labs(
    x = "Quarter",
    y = "Variable",
    title = "Plot A",
    subtitle = "Time series plot"
  )
```