# Creating Elegant Graphs with R Programming

Hazel Kavılı

WomenTechMakers'17 - Istanbul
March 19 - @BahcesehirUni

istanbul@rladies.org
hazel@rladies.org
@ZofiatheWitch

Today’s material’s

- universaltourist.github.io/wtm17istanbul/

# For Beginners

# Getting Started

R commands;

- are case sensitive
- can be separated either by a semi-colon (‘;’), or by a newline
- #comment

Objects;

- variables, arrays of numbers, character strings, functions

Need Help;

- ?summary
- help(summary)
- example(summary)

```
#this is WTM Istanbul
A <- 10
a <- 3
print(paste("A is", A))
print(paste("a is", a))
cat("A and a are equal? = ", A == a)

myNumbers <- c(1:10)
rep(myNumbers, times = 3)
twice <- rep(myNumbers, each = 2)


ls()
rm(a)
print(twice)
```

# Assignments, Basic Operators

Assignments

- use <- or -> symbol combination

Basic arithmetic operators

- +,-,*,/,^, %%

Logical operators

- <, >, <=, >=, ==, !=, !x, x & y, x | y

Others

- sum, sqrt, min, max, mean, var, sd, abs, summary

# Basic Operators

| | |
|---|---|
| `is.na(x)` | test if x is NA |
| `!is.na(x)` | test if x is not Na |
| `x %in% y` | test if x is in y |
| `!(x %in% y)` | test if x is not in y |
| `!x` | not x |

- `x <- c(1:15)`
- `movies <- read.csv("movies.csv", sep = ";", header = TRUE)`

- `seq(from = 2, to = 100, by = 2) -> y`

- `sum(x), min(x), max(x), mean(x), var(x), sqrt(x), sd(x), length(x)`

- `install.packages("tidyverse")`
- `library(tidyverse)`

# Workflow: Projects

1

New Project

**Create Project**

**New Directory**
Start a project in a brand new working directory

**Existing Directory**
Associate a project with an existing working directory

**Version Control**
Checkout a project from a version control repository

Cancel

2

New Project

Back **Project Type**

**Empty Project**
Create a new project in an empty directory

**R Package**
Create a new R package

**Shiny Web Application**
Create a new Shiny web application

Cancel

3

New Project

Back **Create New Project**

Directory name:
wtm17

Create project as subdirectory of:
~/Desktop
Browse...

Create a git repository

Open in new session

Create Project Cancel

Save your codes

&

Keep track of them

# R Script or R Markdown

- R Script: File -> New File -> R Script
- R Markdown: File -> New File -> R Markdown

**R Script**

- Type your code in the R Script
- Use curser + Run or highlight + Run
- Use # for comments
- Run the codes: Cmd + Enter or Highlight + Run

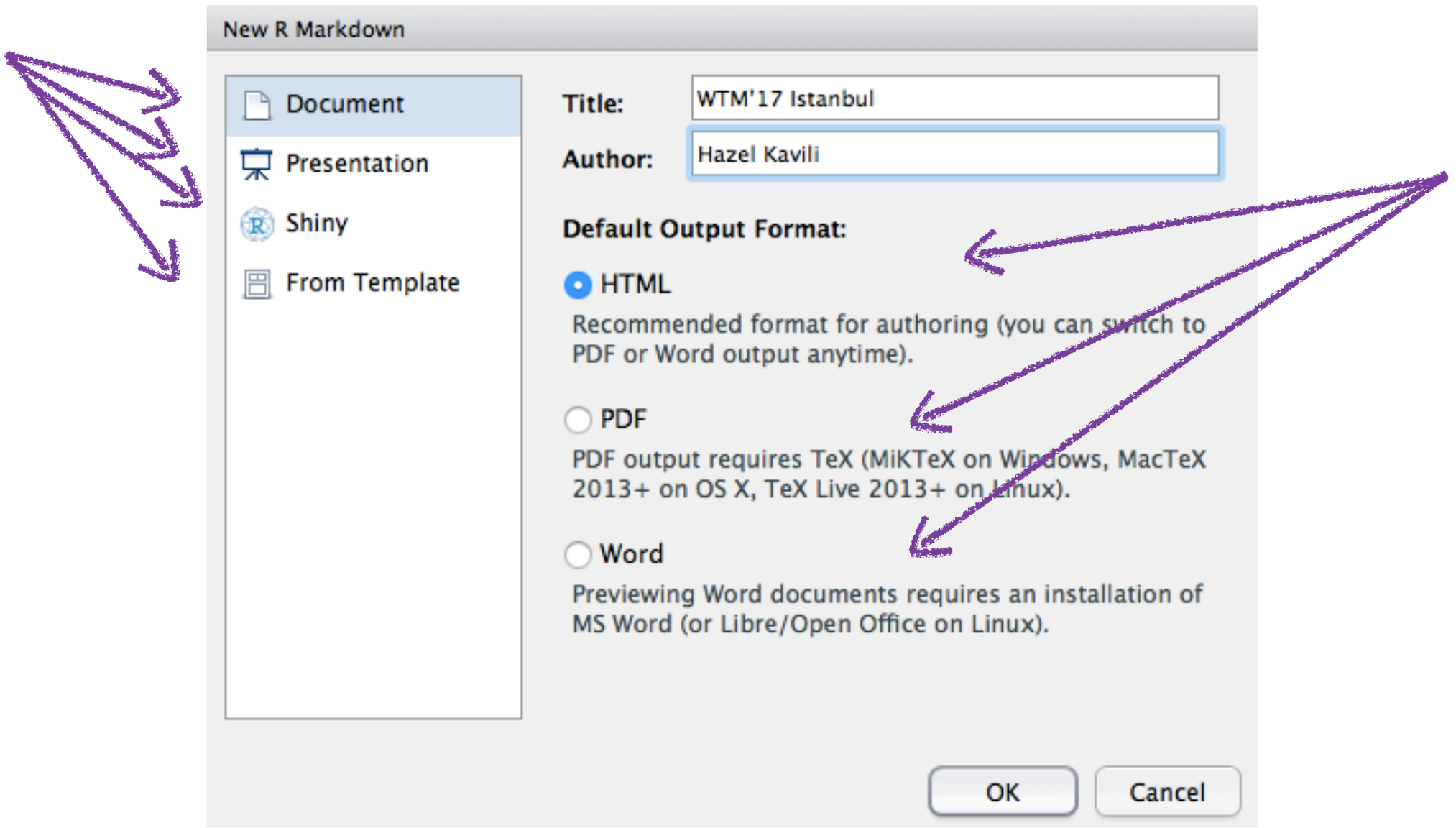
**R Markdown**

- Helps to create of dynamics documents, presentation and reports
- Fully reproducible
- Source: http://rmarkdown.rstudio.com/

# R Markdown

# Starting with R Markdown

- R Markdown: File -> New File -> R Markdown




# What is Markdown?

- Markdown is a particular type of markup language.
- Markup languages are designed produce documents from plain text
- PDF, Word, HTML
- Like LaTeX but more human friendly :)

# Why use Markdown?

- It is flexible
- Focus on content rather than coding debugging errors
- Markdown files can easily be converted to many different formats
- Fastest way to internet

# Important!

- You'll need to define any R objects that this document uses
- You'll need to load any packages that it uses
- The document won't have access to the objects that exist in your current r session

# R Markdown

- File -> New file -> R Markdown

YAML header

```
---
title: "WTM'17 Istanbul"
author: "Hazel Kavili"
date: "3/19/2017"
output: html_document
---
```

# R Markdown

```
```{r setup, include = FALSE}
knitr::opts_chunks$set(echo = TRUE)
```
```

If FALSE, knit will run the code chunk but not include the chunk in the final document

# R Markdown Basics

```
# Header1

## Header2

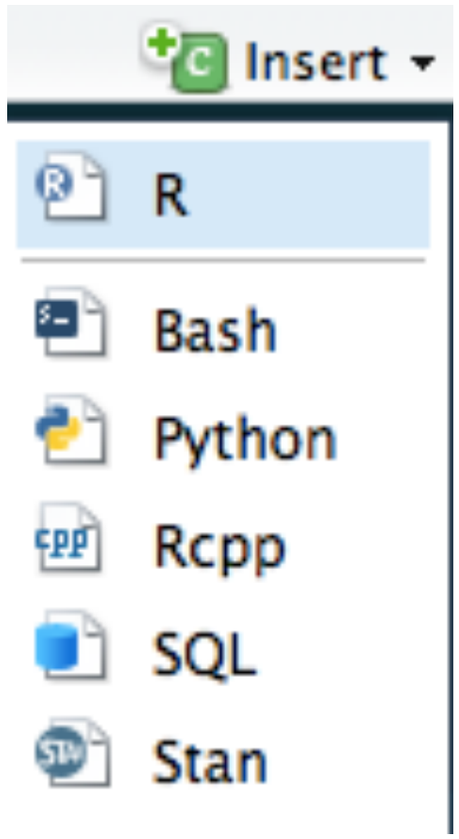
### Header3

#### Header4

##### Header5

###### Header6
```

# Header1

## Header2

### Header3

#### Header4

##### Header5

###### Header6

# R Markdown Basics

- Add picture

```
![caption](path)

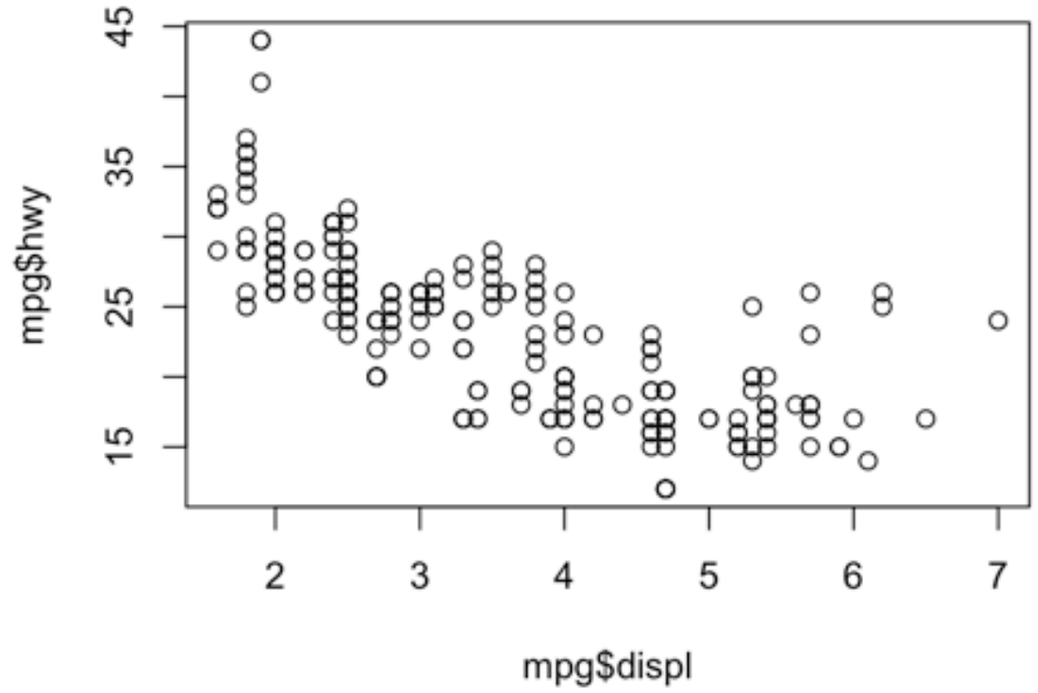
![earth](/Users/hazelkavili/Desktop/
earth.jpg)
```

- Add link

```
[caption](link)

[My_Github](https://github.com/
UniversalTourist)
```

# R Markdown Basics

- Add list

```
### My ordered list
1. apple
2. banana
3. milk
```

### My ordered list

1. apple
2. banana
3. milk

- *Italics* and **Bold**

```
### My style
Hello I am Hazel from
**Istanbul** and I am a
**huge** fan of
*Harry Potter*
```

### My style

Hello I am Hazel from **Istanbul** and I am a **huge** fan of *Harry Potter*

# R Markdown Basics

- Code Chunk




```
```{r}
some codes to run
```
```

- Code Chunk

```
```{r}
movies <- read.csv(“movies.csv”, header = TRUE,
sep = “,”)
```
```

# R Markdown Basics

- Code Chunk

````
```{r}
summary(movies)
ncol(movies)
```
````

- Code Chunk

````
```{r, warnings = FALSE, results = 'hide'}
library(tidyverse)
```
````

# R Markdown Basics

- Code Chunk

```
```{r engine = pyhton}
some python_code
```
```

# R Markdown Basics

- Error Messages

```
```{r, warning = FALSE, error = FALSE}
“four” + “five”
```
```

- echo = FALSE —> not display code - only results
- eval = FALSE —> not run or results - only display code
- results = ‘hide’ —> not display results - only run and display code

# R Markdown Basics

- Figures, plots

````
```{r, fig.width=5, fig.height=4, echo=TRUE}
plot(mpg$displ, mpg$hwy)

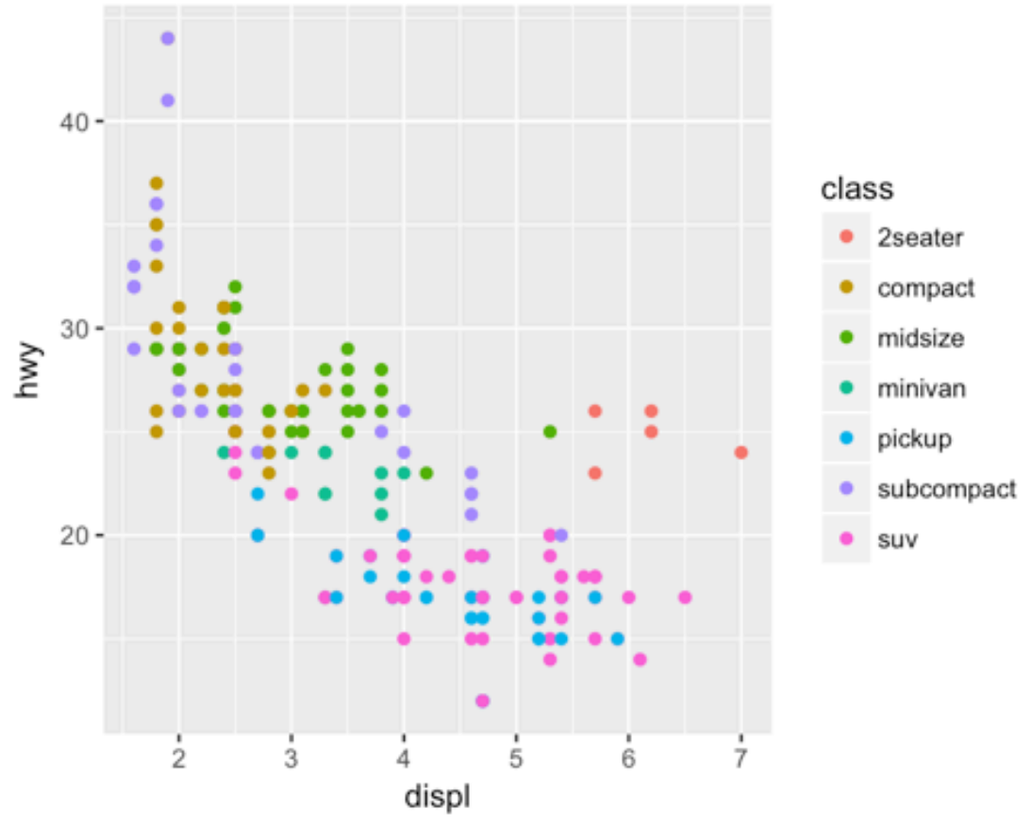
ggplot(data = mpg) +
geom_point(mapping = aes(x = displ, y = hwy, color
= class))
```
````

# R Markdown Basics

```
plot(mpg$displ, mpg$hwy)
```




```
ggplot(data = mpg) + geom_point(mapping = aes(x = displ, y = hwy, color = class))
```



# Packages

# &

# Loading Data

# Install.Packages & Library

Install the Packages by running the codes in the Console

- `install.packages("tidyverse")`

Then load the packages by running the following codes

- `library(tidyverse)`

Keep only *library(tidyverse)* command in your Markdown document!!

# tidyverse: Easily Install Tidyverse Packages

- broom, dplyr, tidyr, ggplot2, lubridate
- magrittr, purrr, modelr, readxl
- stringr, forcats, tibble

# Tidy Data

In **tidy data**

- Each variable forms a column
- Each observation forms a row
- Each type of observational unit forms a table

| | director_name | duration | actor_2_name |
|---|---|---|---|
| 1 | James Cameron | 178 | Joel David Moore |
| 2 | Gore Verbinski | 169 | Orlando Bloom |
| 4 | Christopher Nolan | 164 | Christian Bale |
| 5 | Andrew Stanton | 132 | Samantha Morton |
| 6 | Sam Raimi | 156 | James Franco |
| 7 | Nathan Greno | 100 | Donna Murphy |
| 8 | Joss Whedon | 141 | Robert Downey Jr. |
| 10 | Zack Snyder | 183 | Lauren Cohan |
| 11 | Bryan Singer | 169 | Marlon Brando |
| 13 | Gore Verbinski | 151 | Orlando Bloom |

variables

| | director_name | duration | actor_2_name |
|---|---|---|---|
| 1 | James Cameron | 178 | Joel David Moore |
| 2 | Gore Verbinski | 169 | Orlando Bloom |
| 4 | Christopher Nolan | 164 | Christian Bale |
| 5 | Andrew Stanton | 132 | Samantha Morton |
| 6 | Sam Raimi | 156 | James Franco |
| 7 | Nathan Greno | 100 | Donna Murphy |
| 8 | Joss Whedon | 141 | Robert Downey Jr. |
| 10 | Zack Snyder | 183 | Lauren Cohan |
| 11 | Bryan Singer | 169 | Marlon Brando |
| 13 | Gore Verbinski | 151 | Orlando Bloom |

observations

# Let’s Go!

- `install.packages(“tidyverse”)`
- `library(tidyverse)`
- `movies <- read.csv(“movies.csv", header=TRUE, sep=“,")`

# dplyr

# %>% (pipe) operator

- magrittr package
- basically tells R to take the value of that which is to the left and pass it to the right as an argument.
- cmd + shft + m
- kntr + shft + m

```
movies %>% select((country, title_year,
imdb_score) %>% filter(country == “UK”)
```

# select

Choosing is not losing!

- `select(dataframe, var1, var2,…)`
- `select(dataframe, 1:4, -2)`

Helper functions

- starts_with, ends_with, contains

```
movies %>% select(country,
title_year, imdb_score)
```

# mutate

Deals with info in your data which is not display

- `mutate(dataframe, new = var1 + var2)`
- `mutate(my_df, x = a + b, y = x + c)`

```
gainORlost <- movies %>%
              mutate(difference = gross - budget)
```

*mutate(dataframe, new_Var = expression)*

# filter

*Filter out rows, specific type of **observation***

```
filter(dataframe, logicaltest)
```

```
movies %>% select(country, title_year, imdb_score)
%>% filter (imdb_score >= 6 & country == "UK")
```

# arrange

Help order observation *(default ascending)*

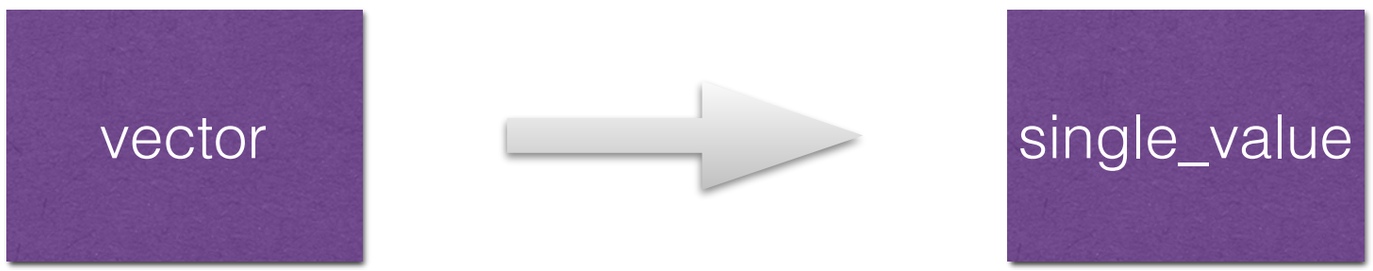
- `arrange(dataframe, var1)`
- `arrange(dataframe, var1, desc(var2))`

```
byBudget <- movies %>% arrange(budget)
```

# summarise

Builds a new dataset that contains only the summarising statistics

- `summarise(dataframe, newColname = expression,. . .)`
- `summarise(dataframe, sum = sum(A), avg = mean(B)..)`



# And there is more…

You can search for:

- `group_by`
- `rename`
- `sample_n & sample_frac`
- `transmute`
- `slice`

# dplyr examples

```
myDataImdb <- movies %>%
  select(country, title_year, imdb_score) %>%
  filter(country == "UK" | country == "USA"
| country == "Canada" |
  country == "Germany")
```

# dplyr examples

```
myDataMean <- movies %>%
 select(country, imdb_score, title_year) %>%
 filter(country == "UK" | country == “USA") %>%
 group_by(country) %>%
 summarise(scoreMean = mean(imdb_score))
```

# ggplot2

# &

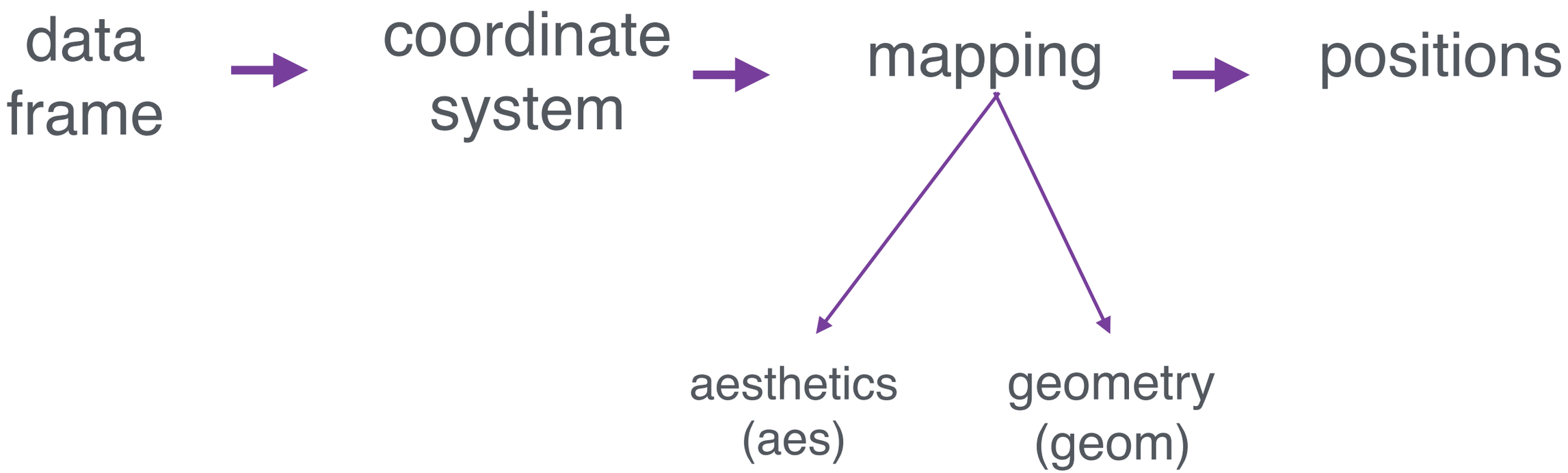
# Tufte style

# Why Good Visualisation?

- “The fundamental principles or rules of an art or science”
- “First step in creating a good sentence”
- Gain insight of data structure
- Help people understand

# Grammer of Graphics

- “It is a tool that enables us to concisely describe the components of a graphic.”



# ggplot basics

- `ggplot(data = <DATA>) +`
  `<GEOM_FUNCTION>(mapping = aes(<MAPPINGS>))`

# ggplot basics

- `ggplot(data = <DATA>) +`

`DATA` is our data frame! !

# ggplot basics

- `ggplot(data = <DATA>) +`
  `<GEOM_FUNCTION>(mapping = aes(<MAPPINGS>))`

- `<GEOM_FUNCTION>` : GEOMETRY OF YOUR GRAPH
  - geom_line()
  - geom_point()
  - geom_bar()
  - geom_boxplot()

# ggplot basics

- `ggplot(data = <DATA>) +`
  `<GEOM_FUNCTION>(mapping = aes(<MAPPINGS>))`

AESTHETICS of GRAPH

- x
- y
- color
- size
- transparency
- linetype, shape

# ggplot examples

- geom_point: IMDB scores vs years by countries

```
ggplot(data = myDataImdb) +
geom_point(mappings = aes(x = title_year,
y = imdb_score,
color = country))
```

# ggplot examples

- geom_bar: Top 10 genres

```
top10genres <- movies %>% count(genres) %>%
                top_n(n = 10, wt = n)
```

```
genresDisplay <- ggplot(data = top10genres) +
  geom_bar(aes(x = reorder(genres, n), y = n,
  fill = genres), stat = “identity”) +
  labs(y = “Frequencies”, x = “Genres”) +
  theme_bw()
```

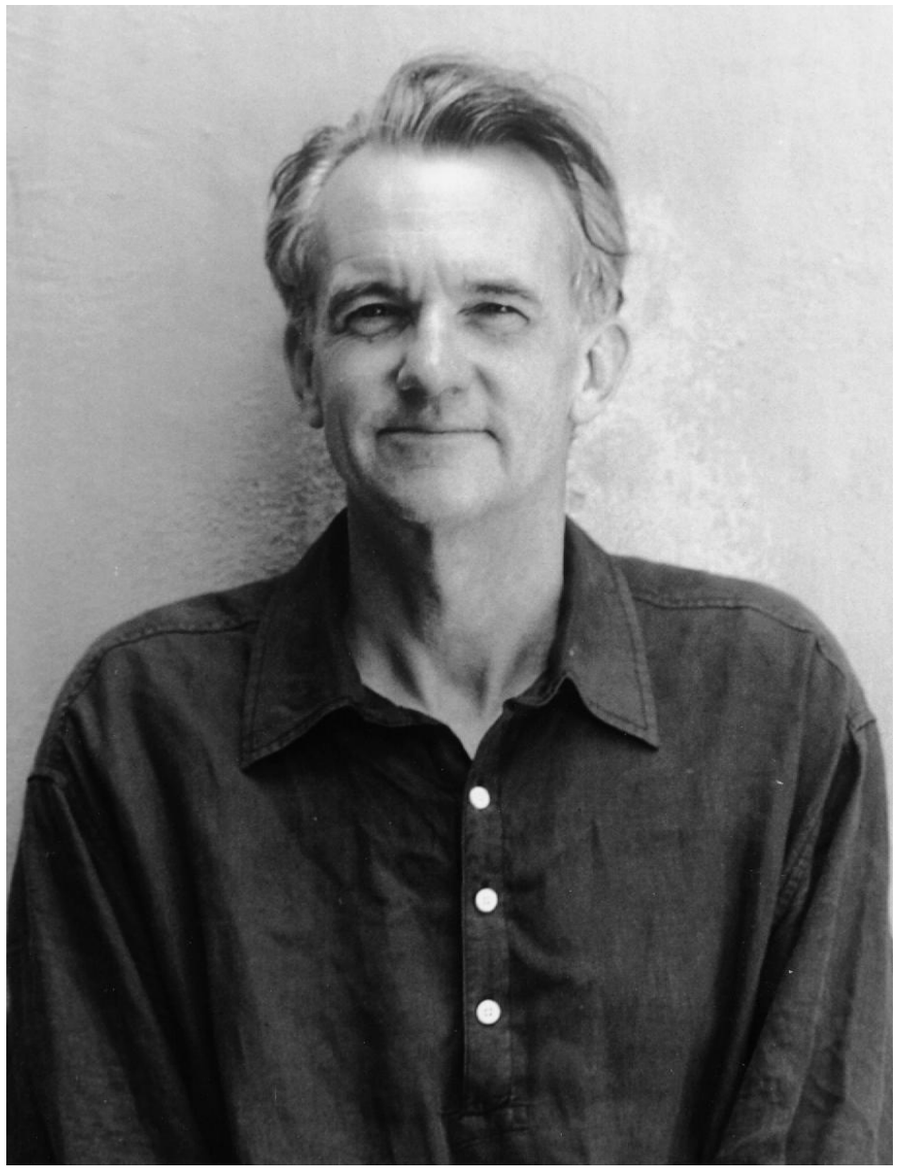
# ggplot examples

- geom_smooth and geom_point: IMDB scores vs. Budget

```
ggplot(data = movies, aes(x = budget, y =
imdb_score)) +
 geom_point(shape = 1) +
geom_smooth(method = lm, se = FALSE) +
labs(x = “Budget”, y = “IMDB scores”,
title = “The effect of Budget on IMDB
scores”) +
theme(legend.position = “top”,
plot.title = element_text(hjust = 0.5))
```

# Edward Tufte

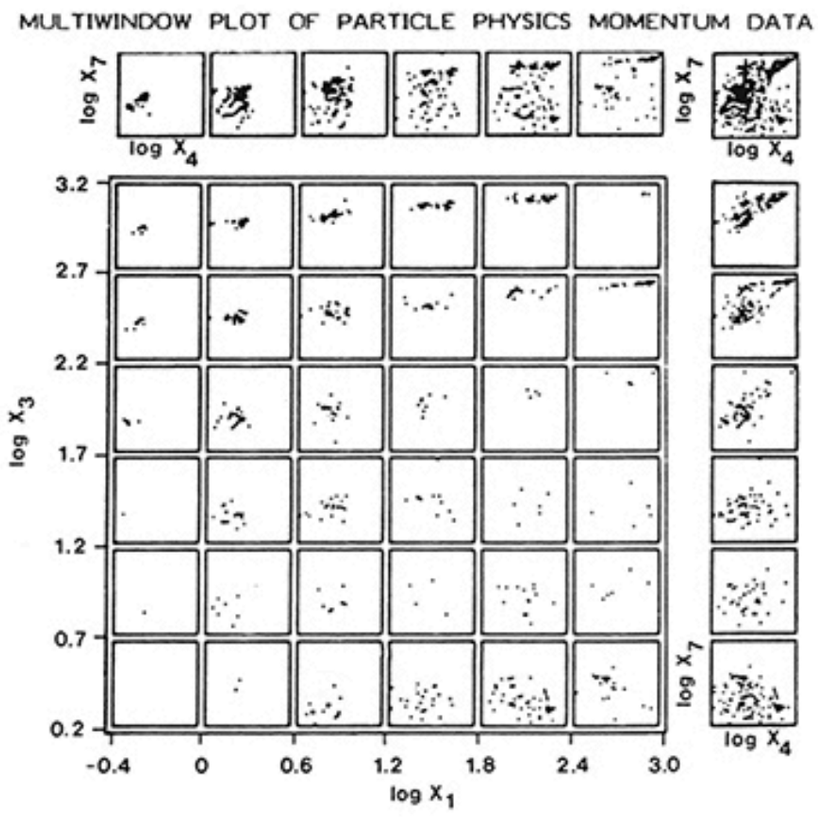
- Statistician and artist
- “Leonardo da Vinci of Data”
- The Visual Display of Quantitative Information

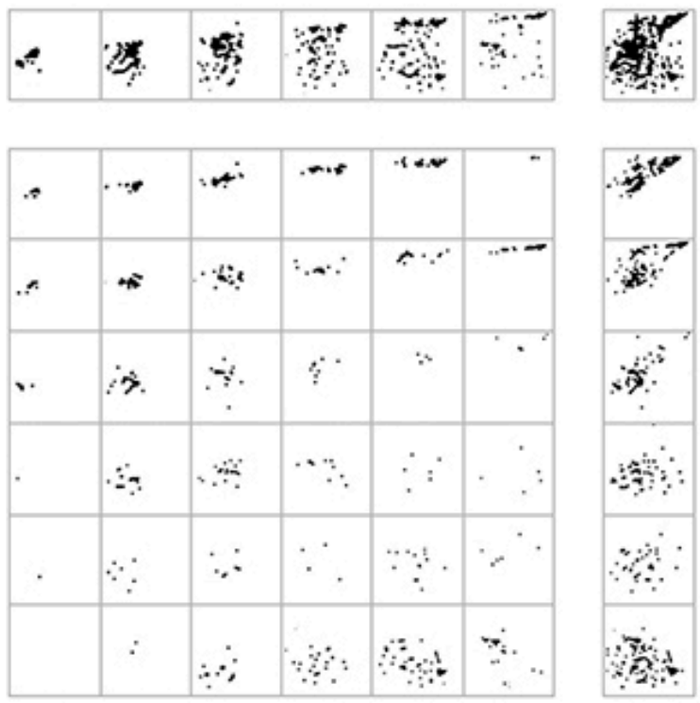


*MUST READ: http://motioninsocial.com/tufte/*

# Tufte style

- Chartjunk: Vibrations, grids and ducks
- Data-ink ratio
- Minimum ink usage
- Not much decoration for graph
- Grid should usually be muted or completely suppressed



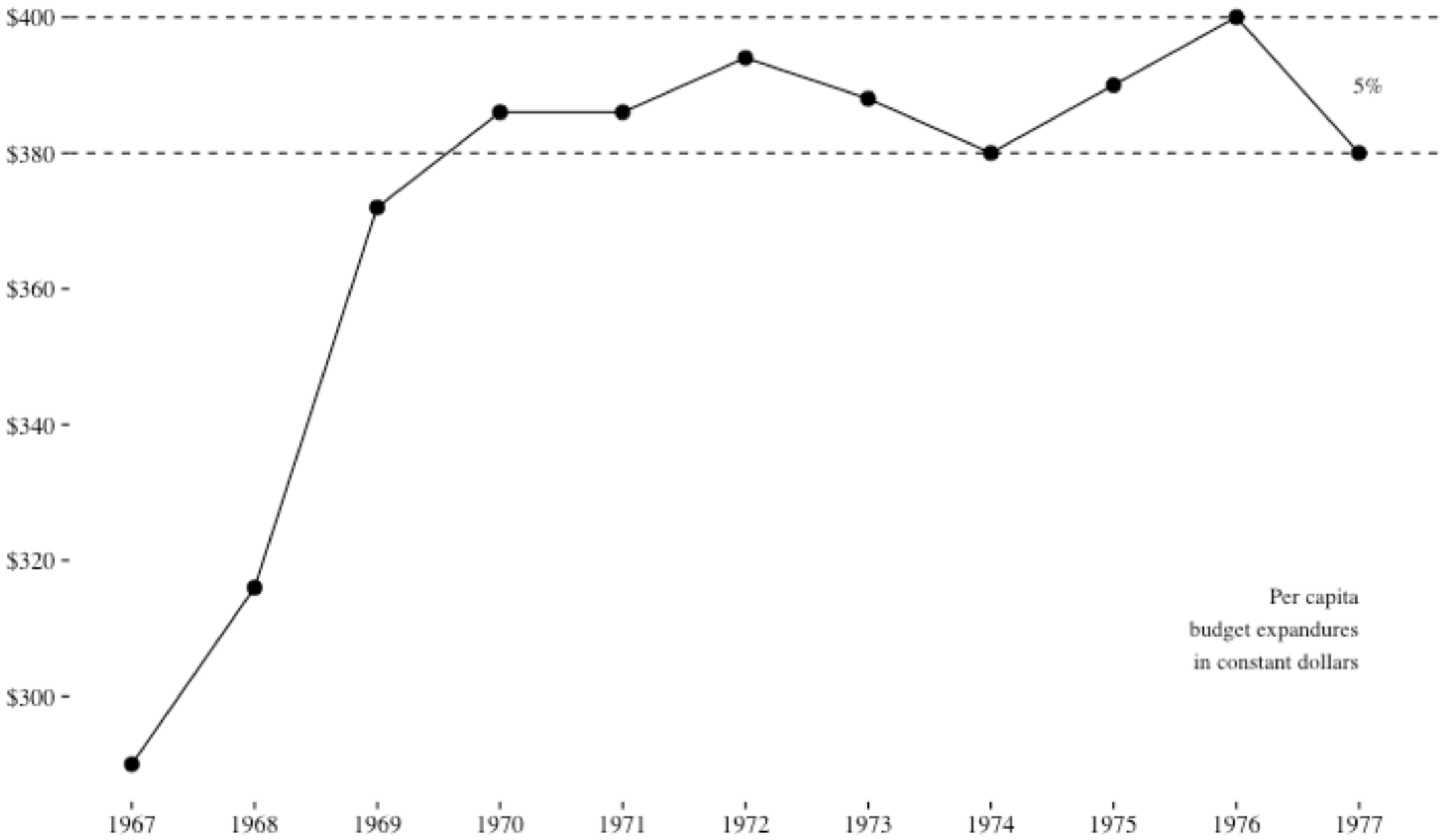
# Tufte style

Tufte's style is known for

- its extensive use of sidenotes
- tight integration of graphics with text
- and well-set typography

```
install.packages(“tufte”)
library(tufte)
File – New file – R Markdown (from template – tufte handout)
```

# Tufte style examples



# Tufte style codes

```
library(ggthemes)
x <- 1967:1977
y <- c(0.5,1.8,4.6,5.3,5.3,5.7,5.4,5,5.5,6,5)
d <- data.frame(x, y)
   ggplot(d, aes(x,y)) + geom_line() + geom_point(size=3) +
   theme_tufte(base_size = 15) +
    theme(axis.title=element_blank()) + geom_hline(yintercept = c(5,6),
    lty = 2) +
     scale_y_continuous(breaks=seq(1, 6, 1),
  label = sprintf("%s",seq(300,400,20))) +
   scale_x_continuous(breaks=x,label=x) +
    annotate("text", x = c(1977,1977.2), y = c(1.5,5.5), adj=1,
  family="serif",
            label = c("Per capita\nbudget expandures\nin constant
dollars", "5%"))
```

# End notes…

# Sources

- Mine Çetinkaya-Rundell's rpubs presentation
- R-Ladies Github
- http://docs.ggplot2.org/0.9.3.1/index.html
- R for Data Science book
- Berk Orbay's Github (github.com/berkorbay) (Sena Önen, Deniz Esin Emer project)
- İsmail Sezen's Github (github.com/isezen)

# Why/Why not use ggplot2?

- http://flowingdata.com/2016/03/22/comparing-ggplot2-and-r-base-graphics/
- http://varianceexplained.org/r/why-I-use-ggplot2/
- http://simplystatistics.org/2016/02/11/why-i-dont-use-ggplot2/
- https://github.com/tidyverse/ggplot2/wiki/Why-use-ggplot2

Today’s material’s

- universaltourist.github.io/wtm17istanbul/

"Data analysis starts with questions, not techniques"

Thank you coming!

R-Ladies Istanbul

istanbul@rladies.org

hazel@rladies.org

@ZofiatheWitch