5. Data Frames and Tidy Data

CT1100 - J. Duggan

Recap - Course Topics

Lecture(s)	Topic
1	Course Introduction
2	The Processing Cycle and Binary Data
3	Data in R with Atomic Vectors
4	The CRAN Library and Calling Functions in R
5	**Tidy Data and Data Frames**
6-7	**ggplot2** - A Grammar of Graphics
8-10	**dplyr** - A Grammar of Data Manipulation
11-12	Introduction to Hardware

R Data Types

	Homogenous	Heterogenous
1d	Atomic Vector	List
2d	Matrix	**Data Frame/Tibble**
nd	Array	

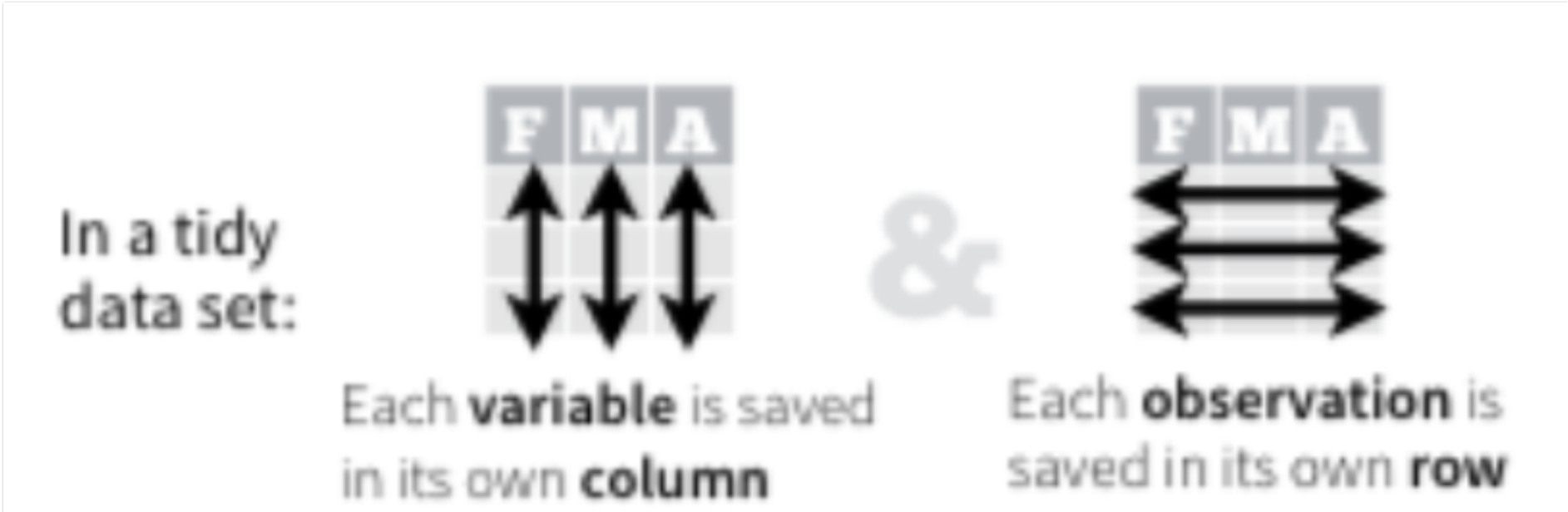
- The most common way of storing data in R
- Under the hood, a data frame is a list of equal-length vectors
- A two-dimensional structure (rectangular data), with rows and columns
- Similar to a worksheet in Excel

Tidy Data - Overview (Wickham 2017)

- What is data tidying?
  - Structuring datasets to facilitate analysis
- The tidy data standard is designed to:
  - Facilitate initial exploration and analysis of data
  - Simplify the development of data analysis tools that work well together
- Principles closely related to relational algebra (Codd 1990)
- Advantage to picking one consistent way of storing data. Easier to learn tools that work with tidy data because they have a underlying uniformity
- Specific advantage to placing variables in columns because it allows R's vectorised functions to shine.
- **dplyr**, **ggplot2** designed to work with tidy data

Rules for a Tidy Data Set

- Each variable must have its own column
- Each observation must have its own row
- Each value must have its own cell



A tidy data set

Time	Team	Scorer	From	Type	Points	Score
1	Dublin	Paul Mannion	Play	Point	1	1
2	Kerry	Sean O'Shea	Play	Point	1	1
3	Dublin	Dean Rock	Play	Point	1	2
4	Dublin	Dean Rock	Free	Point	1	3
10	Kerry	David Clifford	Play	Point	1	2
13	Kerry	Sean O'Shea	FortyFive	Point	1	3
14	Kerry	Stephen O'Brien	Play	Point	1	4
16	Dublin	Paul Mannion	Play	Point	1	4
18	Kerry	Sean O'Shea	Free	Point	1	5
19	Dublin	Jack McCaffrey	Play	Goal	3	7

mtcars data frame

A data frame with 32 observations on 11 variables.

- **mpg** Miles/(US) gallon
- **cyl** Number of cylinders
- **disp** Displacement (cu.in.)
- **hp** Gross horsepower
- **drat** Rear axle ratio
- **wt** Weight (1000 lbs)
- **qsec** 1/4 mile time
- **vs** V/S
- **am** Transmission (0 = automatic, 1 = manual)
- **gear** Number of forward gears
- **carb** Number of carburetors

mtcars sample data

```
knitr::kable(mtcars[1:10,1:6])
```

	mpg	cyl	disp	hp	drat	wt
Mazda RX4	21.0	6	160.0	110	3.90	2.620
Mazda RX4 Wag	21.0	6	160.0	110	3.90	2.875
Datsun 710	22.8	4	108.0	93	3.85	2.320
Hornet 4 Drive	21.4	6	258.0	110	3.08	3.215
Hornet Sportabout	18.7	8	360.0	175	3.15	3.440
Valiant	18.1	6	225.0	105	2.76	3.460
Duster 360	14.3	8	360.0	245	3.21	3.570
Merc 240D	24.4	4	146.7	62	3.69	3.190
Merc 230	22.8	4	140.8	95	3.92	3.150
Merc 280	19.2	6	167.6	123	3.92	3.440

mtcars using str()

```
str(mtcars)
```

```
## 'data.frame':    32 obs. of  11 variables:
##  $ mpg : num  21 21 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2
##  $ cyl : num  6 6 4 6 8 6 8 4 4 6 ...
##  $ disp: num  160 160 108 258 360 ...
##  $ hp  : num  110 110 93 110 175 105 245 62 95 123 ...
##  $ drat: num  3.9 3.9 3.85 3.08 3.15 2.76 3.21 3.69 3.92 3.
##  $ wt  : num  2.62 2.88 2.32 3.21 3.44 ...
##  $ qsec: num  16.5 17 18.6 19.4 17 ...
##  $ vs  : num  0 0 1 1 0 1 0 1 1 1 ...
##  $ am  : num  1 1 1 0 0 0 0 0 0 0 ...
##  $ gear: num  4 4 4 3 3 3 3 4 4 4 ...
##  $ carb: num  4 4 1 1 2 1 4 2 2 4 ...
```

head() and tail() functions

```
head(mtcars[,1:6])
```

```
##                   mpg cyl disp  hp drat    wt
## Mazda RX4         21.0   6  160 110 3.90 2.620
## Mazda RX4 Wag     21.0   6  160 110 3.90 2.875
## Datsun 710        22.8   4  108  93 3.85 2.320
## Hornet 4 Drive    21.4   6  258 110 3.08 3.215
## Hornet Sportabout 18.7   8  360 175 3.15 3.440
## Valiant           18.1   6  225 105 2.76 3.460
```

```
tail(mtcars[,1:6])
```

```
##                mpg cyl  disp  hp drat    wt
## Porsche 914-2  26.0   4 120.3  91 4.43 2.140
## Lotus Europa   30.4   4  95.1 113 3.77 1.513
## Ford Pantera L 15.8   8 351.0 264 4.22 3.170
## Ferrari Dino   19.7   6 145.0 175 3.62 2.770
```

subset()

- The subset( ) function can be used to select variables and observations.
- Takes the data frame, the conditions, and the columns to return

```
subset(mtcars,cyl==6,select=c("mpg","cyl"))
```

```
##                   mpg cyl
## Mazda RX4        21.0   6
## Mazda RX4 Wag    21.0   6
## Hornet 4 Drive   21.4   6
## Valiant          18.1   6
## Merc 280         19.2   6
## Merc 280C        17.8   6
## Ferrari Dino     19.7   6
```

Challenge 5.1: Using the subset function

- List all the cars that have an **mpg** greater than the average
- List the car(s) with the greatest displacement (**disp**)

Adding new columns to a data frame

- Often the initial data set may not contain sufficient information for analysis
- Adding new variables (columns) is an important feature to have
- Data frames support this: columns can be combined or new information used

```
mtcars$name <- rownames(mtcars)
mtcars[1:5,-(1:8)]
```

```
##                   am gear carb              name
## Mazda RX4          1    4    4         Mazda RX4
## Mazda RX4 Wag      1    4    4     Mazda RX4 Wag
## Datsun 710         1    4    1        Datsun 710
## Hornet 4 Drive     0    3    1    Hornet 4 Drive
## Hornet Sportabout  0    3    2 Hornet Sportabout
```

Challenge 1.6

Create a new column on mtcars that contains kilometers per gallon.

The tibble

- Tibbles are data frames, but they tweak some older behaviours to make life a little easier
- One of the unifying features of the tidyverse
- To coerce a data frame to a tibble, use as_tibble()
- A tibble can be created from individual vectors using tibble()
- The data set **ggplot2::mpg** is a tibble

Tibble abbreviations

```
as_tibble(mtcars)[1:2,1:6]
```

```
## # A tibble: 2 x 6
##     mpg   cyl  disp    hp  drat    wt
##   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1    21     6   160   110   3.9  2.62
## 2    21     6   160   110   3.9  2.88
```

Abbreviation	Data Type
int	integers
dbl	double (numeric)
chr	character vectors
dttm	date-times
fctr	categorical
date	dates

Summary

- Data frames/tibbles are the most common way of storing heterogeneous data in R
- Under the hood, a data frame is a list of equal-length vectors, and shares properties of both a list and a matrix
- Key for processing rectangular data, ideally in "tidy" format (every row is an observation, every column a variable)