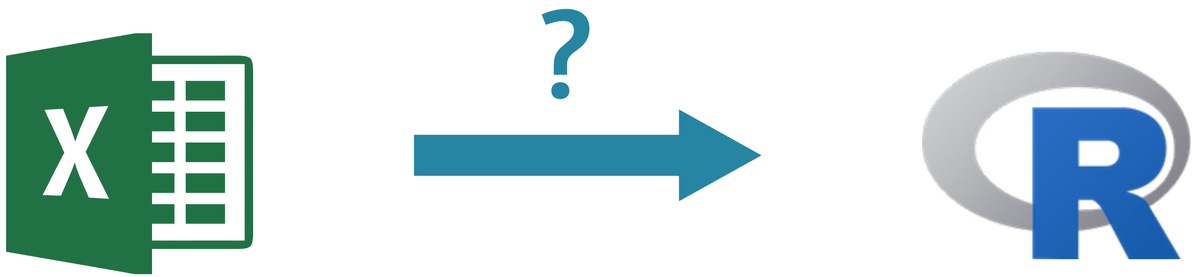
IMPORTING DATA IN R

Introduction read.csv

Importing data in R



5 types

- Flat files
- Data from Excel
- Databases
- Web
- Statistical software

Flat Files

states.csv

Comma Separated Values

```
state,capital,pop_mill,area_sqm
South Dakota,Pierre,0.853,77116
New York,Albany,19.746,54555
Oregon,Salem,3.970,98381
Vermont,Montpelier,0.627,9616
Hawaii,Honolulu,1.420,10931
```

Field names

?

```
> wanted_df
          state    capital pop_mill area_sqm
1 South Dakota     Pierre    0.853    77116
2      New York     Albany   19.746    54555
3        Oregon      Salem    3.970    98381
4       Vermont Montpelier    0.627     9616
5        Hawaii   Honolulu    1.420    10931
```

utils - read.csv

- Loaded by default when you start R

states.csv

```
state,capital,pop_mill,area_sqm
South Dakota,Pierre,0.853,77116
New York,Albany,19.746,54555
Oregon,Salem,3.970,98381
Vermont,Montpelier,0.627,9616
Hawaii,Honolulu,1.420,10931
```

```
> read.csv("states.csv", stringsAsFactors = FALSE)
```

Import strings as categorical variables?

What if file in datasets folder of home directory?

```
> path <- file.path("~", "datasets", "states.csv")

> path
[1] "~/datasets/states.csv"

> read.csv(path, stringsAsFactors = FALSE)
```

read.csv()

```
> read.csv("states.csv", stringsAsFactors = FALSE)

          state    capital pop_mill area_sqm
1 South Dakota     Pierre    0.853    77116
2     New York     Albany   19.746    54555
3       Oregon      Salem    3.970    98381
4      Vermont Montpelier    0.627     9616
5       Hawaii   Honolulu    1.420    10931

> df <- read.csv("states.csv", stringsAsFactors = FALSE)
> str(df)
'data.frame':5 obs. of  4 variables:
 $ state   : chr  "South Dakota" "New York" "Oregon" "Vermont" ...
 $ capital : chr  "Pierre" "Albany" "Salem" "Montpelier" ...
 $ pop_mill: num  0.853 19.746 3.97 0.627 1.42
 $ area_sqm: int  77116 54555 98381 9616 10931
```

states.csv

```
state,capital,pop_mill,area_sqm
South Dakota,Pierre,0.853,77116
New York,Albany,19.746,54555
Oregon,Salem,3.970,98381
Vermont,Montpelier,0.627,9616
Hawaii,Honolulu,1.420,10931
```

IMPORTING DATA IN R

Let’s practice!

IMPORTING DATA IN R

read.delim
read.table

Tab-delimited file

states.txt

```
state   capital pop_mill    area_sqm
South Dakota    Pierre  0.853   77116
New York    Albany  19.746  54555
Oregon  Salem  3.970   98381
Vermont Montpelier  0.627   9616
Hawaii  Honolulu    1.420   10931
```

```
> read.delim("states.txt", stringsAsFactors = FALSE)
         state    capital pop_mill area_sqm
1 South Dakota     Pierre    0.853    77116
2     New York     Albany   19.746    54555
3       Oregon      Salem    3.970    98381
4      Vermont Montpelier    0.627     9616
5       Hawaii   Honolulu    1.420    10931
```

Exotic file format

states2.txt

```
state/capital/pop_mill/area_sqm
South Dakota/Pierre/0.853/77116
New York/Albany/19.746/54555
Oregon/Salem/3.970/98381
Vermont/Montpelier/0.627/9616
Hawaii/Honolulu/1.420/10931
```

read.table()

- Read any tabular file as a data frame
- Number of arguments is huge

states2.txt

```
state/capital/pop_mill/area_sqm
South Dakota/Pierre/0.853/77116
New York/Albany/19.746/54555
Oregon/Salem/3.970/98381
Vermont/Montpelier/0.627/9616
Hawaii/Honolulu/1.420/10931
```

```
> read.table("states2.txt",
             header = TRUE,
             sep = "/",
             stringsAsFactors = FALSE)

          state    capital pop_mill area_sqm
1 South Dakota     Pierre    0.853    77116
2     New York     Albany   19.746    54555
3       Oregon      Salem    3.970    98381
4      Vermont Montpelier    0.627     9616
5       Hawaii   Honolulu    1.420    10931
```

first row lists variable names (default FALSE)

field separator is a forward slash

IMPORTING DATA IN R

Let’s practice!

IMPORTING DATA IN R

Final thoughts

Wrappers

- read.table() is the main function
- read.csv() = wrapper for CSV
- read.delim() = wrapper for tab-delimited files

read.csv

- Defaults
  - header = TRUE
  - sep = ","

states.csv

```
state,capital,pop_mill,area_sqm
South Dakota,Pierre,0.853,77116
New York,Albany,19.746,54555
Oregon,Salem,3.970,98381
Vermont,Montpelier,0.627,9616
Hawaii,Honolulu,1.420,10931
```

```
> read.table("states.csv", header = TRUE, sep = ",",
             stringsAsFactors = FALSE)

> read.csv("states.csv", stringsAsFactors = FALSE)
```

read.delim

- Defaults
  - header = TRUE
  - sep = "\t"

states.txt

```
state   capital pop_mill    area_sqm
South Dakota    Pierre  0.853   77116
New York    Albany  19.746  54555
Oregon  Salem  3.970   98381
Vermont Montpelier  0.627   9616
Hawaii  Honolulu    1.420   10931
```

```
> read.table("states.txt", header = TRUE, sep = "\t",
             stringsAsFactors = FALSE)

> read.delim("states.txt", stringsAsFactors = FALSE)
```

Documentation

```
> ?read.table
```

Description

Reads a file in table format and creates a data frame from it, with cases corresponding to lines and variables to fields in the file.

Usage

```
read.table(file, header = FALSE, sep = "", quote = "\"'",
           dec = ".", numerals = c("allow.loss", "warn.loss", "no.loss"),
           row.names, col.names, as.is = !stringsAsFactors,
           na.strings = "NA", colClasses = NA, nrows = -1,
           skip = 0, check.names = TRUE, fill = !blank.lines.skip,
           strip.white = FALSE, blank.lines.skip = TRUE,
           comment.char = "#",
           allowEscapes = FALSE, flush = FALSE,
           stringsAsFactors = default.stringsAsFactors(),
           fileEncoding = "", encoding = "unknown", text, skipNul = FALSE)

read.csv(file, header = TRUE, sep = ",", quote = "\"",
         dec = ".", fill = TRUE, comment.char = "", ...)

read.csv2(file, header = TRUE, sep = ";", quote = "\"",
          dec = ",", fill = TRUE, comment.char = "", ...)

read.delim(file, header = TRUE, sep = "\t", quote = "\"",
           dec = ".", fill = TRUE, comment.char = "", ...)

read.delim2(file, header = TRUE, sep = "\t", quote = "\"",
            dec = ",", fill = TRUE, comment.char = "", ...)
```

Locale differences

states_aye.csv

```
state,capital,pop_mill,area_sqm
South Dakota,Pierre,0.853,77116
New York,Albany,19.746,54555
Oregon,Salem,3.970,98381
Vermont,Montpelier,0.627,9616
Hawaii,Honolulu,1.420,10931
```

states_nay.csv

```
state;capital;pop_mill;area_sqm
South Dakota;Pierre;0,853;77116
New York;Albany;19,746;54555
Oregon;Salem;3,97;98381
Vermont;Montpelier;0,627;9616
Hawaii;Honolulu;1,42;10931
```

Locale differences

```
read.csv(file, header = TRUE, sep = ",", quote = "\"",
         dec = ".", fill = TRUE, comment.char = "", ...)

read.csv2(file, header = TRUE, sep = ";", quote = "\"",
          dec = ",", fill = TRUE, comment.char = "", ...)
```

```
read.delim(file, header = TRUE, sep = "\t", quote = "\"",
           dec = ".", fill = TRUE, comment.char = "", ...)

read.delim2(file, header = TRUE, sep = "\t", quote = "\"",
            dec = ",", fill = TRUE, comment.char = "", ...)
```

states_nay.csv

```
> read.csv("states_nay.csv", stringsAsFactors = FALSE)
                      state.capital.pop_mill.area_sqm
South Dakota;Pierre;0                        853;77116
New York;Albany;19                           746;54555
Oregon;Salem;3                                97;98381
Vermont;Montpelier;0                          627;9616
Hawaii;Honolulu;1                              42;10931

> read.csv2("states_nay.csv", stringsAsFactors = FALSE)
         state    capital pop_mill area_sqm
1 South Dakota     Pierre    0.853    77116
2     New York     Albany   19.746    54555
3       Oregon      Salem    3.970    98381
4      Vermont Montpelier    0.627     9616
5       Hawaii   Honolulu    1.420    10931
```

states_nay.csv

```
state;capital;pop_mill;area_sqm
South Dakota;Pierre;0,853;77116
New York;Albany;19,746;54555
Oregon;Salem;3,97;98381
Vermont;Montpelier;0,627;9616
Hawaii;Honolulu;1,42;10931
```

IMPORTING DATA IN R

Let’s practice!