Downloading files

Jeffrey Leek
Johns Hopkins Bloomberg School of Public Health

- Most likely going to download data from the internet
- Use R to get files to have more complete documentation

Get/set your working directory

- A basic component of working with data is knowing your working directory

- The two main commands are `getwd()` and `setwd()`.

- Be aware of relative versus absolute paths

  - **Relative** - `setwd("./data")`, `setwd("../")` _moves you up one directory level_

  - **Absolute** - `setwd("/Users/jtleek/data/")`

- Important difference in Windows `setwd("C:\\Users\\Andrew\\Downloads")`
  ↳ w/ absolute paths

Checking for and creating directories

- `file.exists("directoryName")` will check to see if the directory exists *in the working directory*

- `dir.create("directoryName")` will create a directory if it doesn't exist

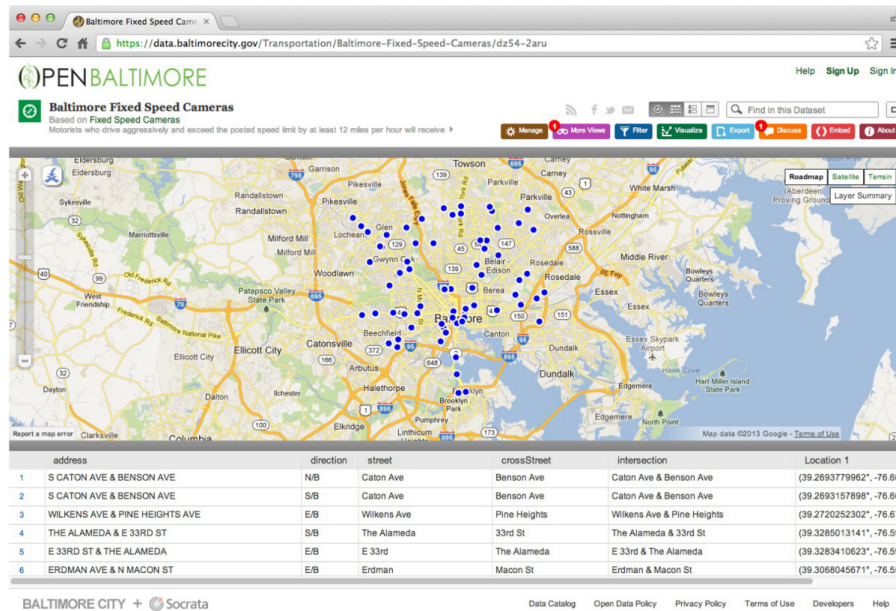- Here is an example checking for a "data" directory and creating it if it doesn't exist

```
if (!file.exists("data")) {
    dir.create("data")
}
```

Getting data from the internet - download.file()

- Downloads a file from the internet

- Even if you could do this by hand, helps with reproducibility

- Important parameters are *url*, *destfile*, *method*

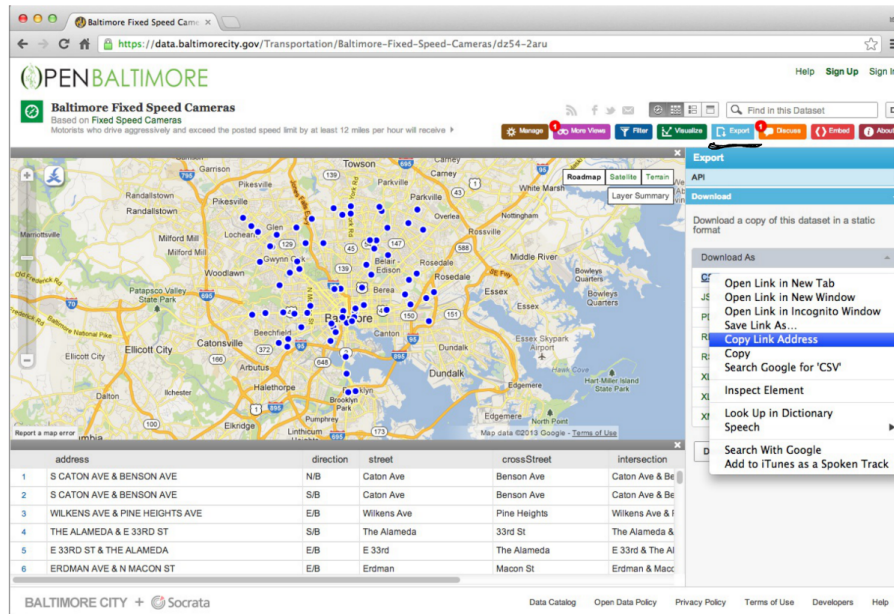- Useful for downloading tab-delimited, csv, and other files

⌐ Main way of getting data from the internet

⌐ Important Parameters

⌐ url — source

⌐ destfile — where data is going

⌐ method — discuss later, but important when working with 'https' addresses

Example - Baltimore camera data *- speed cameras*

https://data.baltimorecity.gov/Transportation/Baltimore-Fixed-Speed-Cameras/dz54-2aru

Example - Baltimore camera data

Handwritten annotations: "← Right clicked on 'CSV'", "← Copy Link Address for use in R script"

https://data.baltimorecity.gov/Transportation/Baltimore-Fixed-Speed-Cameras/dz54-2aru

Download a file from the web

```
fileUrl <- "https://data.baltimorecity.gov/api/views/dz54-2aru/rows.csv?accessType=DOWNLOAD"
download.file(fileUrl, destfile = "./data/cameras.csv", method = "curl")
list.files("./data")
```

```
## [1] "cameras.csv"
```

```
dateDownloaded <- date()
dateDownloaded
```

```
## [1] "Sun Jan 12 21:37:44 2014"
```

└ Need to use "curl" method when working with 'https'

└ Need to record the date that data was downloaded on, because it can change

Some notes about download.file()

- If the url starts with *http* you can use download.file()
- If the url starts with *https* on Windows you may be ok
- If the url starts with *https* on Mac you may need to set *method="curl"*
- If the file is big, this might take a while
- Be sure to record when you downloaded.

Can be useful to skip the download when rerunning a script to save time.

↳ Maybe do:
```
> if (!file.exists("file")) {
> [Download/doc code here]
> }
```