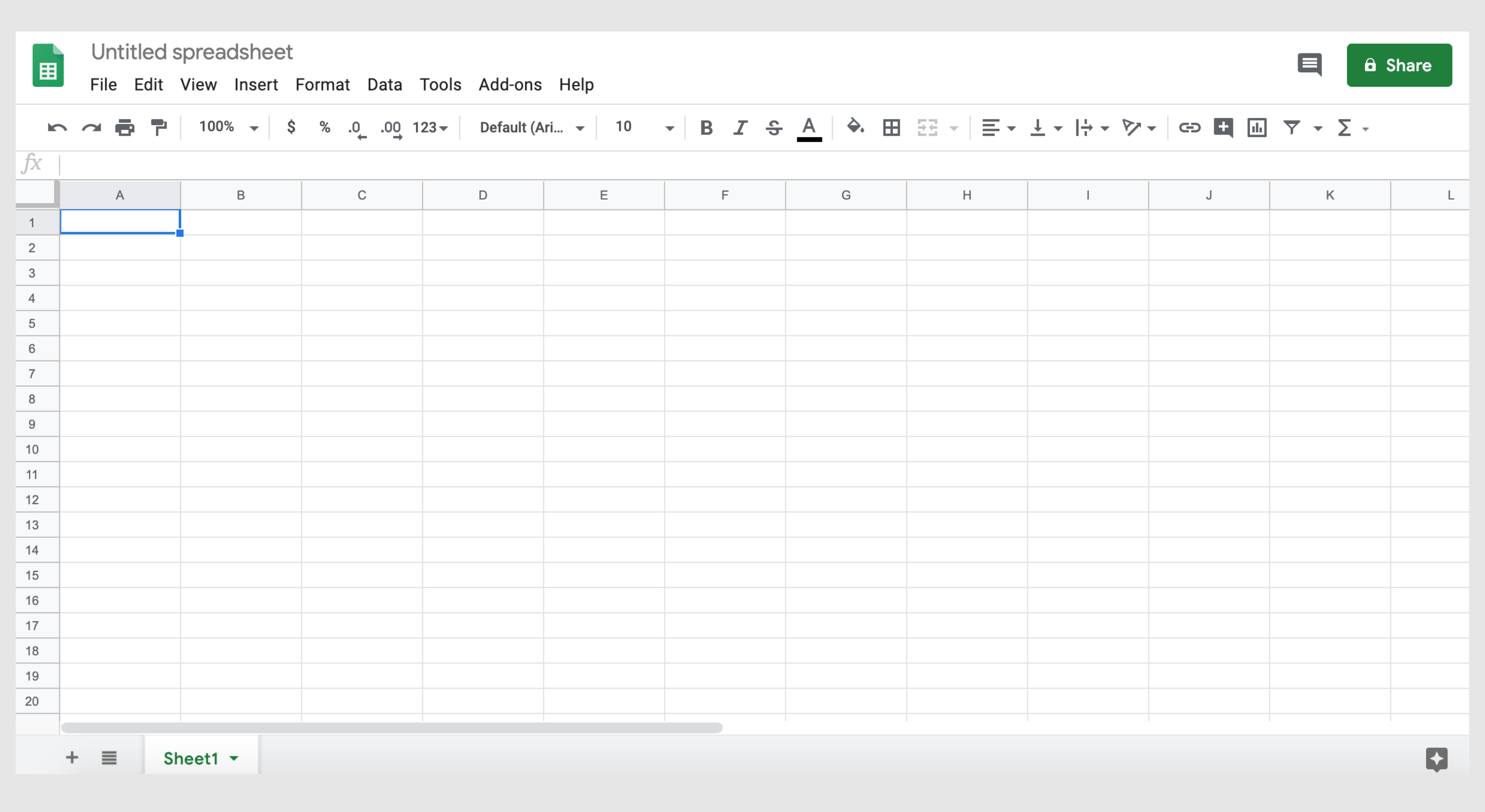
Advanced Data IO

Data Wrangling in R



Google Sheets

https://docs.google.com/spreadsheets

Reading data with the googlesheets package

```
install.packages("googlesheets")
library(googlesheets)
```

need figure

https://docs.google.com/spreadsheets/d/1WBrH655fxqKW1QqvD5hnqvvEWIvRzDJcKEg

need publish figure

```r
sheets_url = paste0("https://docs.google.com/spreadsheets/d/",
                    "1WBrH655fxqKW1QqvD5hnqvvEWIvRzDJcKEgjjFeYxeM")

gsurl1 = gs_url(sheets_url)

dat = gs_read(gsurl1)
date_read = lubridate::today()
head(dat)
```

```
# A tibble: 6 x 12
    Git Github     R Rstudio `Reproducible R… `R markdown` `Data import`
  <dbl>  <dbl> <dbl>   <dbl>            <dbl>        <dbl>         <dbl>
1     9      9     9       9                9            9            10
2     1      1     5       5                7            5             4
3     0      0     2       3                3            0             3
4     0      1     1       1                1            1             1
5     0      2     7       7                7            5             8
6     2      2     3       2                0            0             1
# … with 5 more variables: `Web scraping` <dbl>, `Data cleaning` <dbl>,
#   dplyr <dbl>, Bioconductor <dbl>, `Regular expressions` <dbl>
```

What if I don’t want it public?

```r
library(googlesheets4)
# May be necessary on rstudio.cloud
options(httr_oob_default=TRUE)
# Will ask you to log in
out = read_sheet(sheets_url)
```

Can also save and load a token

```
token = readr::read_rds("googledrive_token.rds")
library(googledrive)
drive_auth(token = token) # could also use googlesheets4::gs4_auth
library(googlesheets4)
out = read_sheet(sheets_url)
```

https://docs.google.com/spreadsheets/d/1j9vbv8MrVV7EK15vyz-rnhjiXhRkmIFEHgdv1_p1cCc/edit?usp=sharing

Google Sheets

https://SISBIB.github.io/Module1/lab
sheets-lab.Rmd

JSON: JavaScript Object Notation
Lists of stuff

** need figure ** https://en.wikipedia.org/wiki/JSON

Why JSON matters

** need figure **

https://developer.github.com/v3/search/

```
#install.packages("jsonlite")
library(jsonlite)
jsonData <- fromJSON("https://api.github.com/users/jtleek/repos")
head(jsonData)
```

```
         id                          node_id               name
1 155565363 MDEwOlJlcG9zaXRvcnkxNTU1NjUzNjM=               2018
2 264786491 MDEwOlJlcG9zaXRvcnkyNjQ3ODY0OTE=            ads2020
3 101394164 MDEwOlJlcG9zaXRvcnkxMDEzOTQxNjQ=         advdatasci
4 111447948 MDEwOlJlcG9zaXRvcnkxMTE0NDc5NDg= advdatasci-project
5  47568815 MDEwOlJlcG9zaXRvcnk0NzU2ODgxNQ==   advdatasci-swirl
6  41645119 MDEwOlJlcG9zaXRvcnk0MTY0NTExOQ==       advdatasci15
                  full_name private owner.login owner.id
1               jtleek/2018   FALSE      jtleek  1571674
2            jtleek/ads2020   FALSE      jtleek  1571674
3         jtleek/advdatasci   FALSE      jtleek  1571674
4 jtleek/advdatasci-project   FALSE      jtleek  1571674
5   jtleek/advdatasci-swirl   FALSE      jtleek  1571674
6       jtleek/advdatasci15   FALSE      jtleek  1571674
         owner.node_id                                     owner.avatar_url
1 MDQ6VXNlcjE1NzE2NzQ= https://avatars2.githubusercontent.com/u/1571674?v=4
2 MDQ6VXNlcjE1NzE2NzQ= https://avatars2.githubusercontent.com/u/1571674?v=4
3 MDQ6VXNlcjE1NzE2NzQ= https://avatars2.githubusercontent.com/u/1571674?v=4
```

Data frame structure from JSON

```r
dim(jsonData)
```

```
[1] 30 73
```

```r
head(jsonData$name)
```

```
[1] "2018"               "ads2020"            "advdatasci"
[4] "advdatasci-project" "advdatasci-swirl"   "advdatasci15"
```

```r
#Some of the columns is a data frame!
table(sapply(jsonData,class))
```

```
 character data.frame    integer    logical 
        52          2          9         10 
```

```r
dim(jsonData$owner)
```

```
[1] 30 18
```

```r
names(jsonData$owner)
```

```
 [1] "login"               "id"                  "node_id"
 [4] "avatar_url"          "gravatar_id"         "url"
 [7] "html_url"            "followers_url"       "following_url"
[10] "gists_url"           "starred_url"         "subscriptions_url"
[13] "organizations_url"   "repos_url"           "events_url"
[16] "received_events_url" "type"                "site_admin"
```

JSON Lab

https://SISBIB.github.io/Module1/lab
lab.Rmd

Web Scraping

need figure

http://bowtie-bio.sourceforge.net/recount/

This is data

View the source

What the computer sees

Ways to see the source

Chrome: 1. right click on page 2. select “view source”

Firefox: 1. right click on page 2. select “view source” Microsoft Edge: 1. right click on page 2. select “view source”

Safari 1. click on “Safari” 2. select “Preferences” 3. go to “Advanced” 4. check “Show Develop menu in menu bar” 5. click on “Develop” 6. select “show page source” 7. alternatively to 5./6., right click on page and select “view source”

https://github.com/simonmunzert/rscraping-jsm-2016/blob/c04fd91fec711df65c838e07723125155a7f2cda/02-scraping-with-rvest.r

Inspect element

Copy XPath

rvest package

```
recount_url = "http://bowtie-bio.sourceforge.net/recount/"
# install.packages("rvest")
library(rvest)
htmlfile = read_html(recount_url)

nds = html_nodes(htmlfile,
xpath='//*[@id="recounttab"]/table')
dat = html_table(nds)
dat = as.data.frame(dat)
head(dat)
```

```
       X1                                         X2      X3
1   Study                                       PMID Species
2 bodymap not published, but publicly available here   human
3  cheung                                   20856902   human
4    core                                   19056941   human
5   gilad                                   20009012   human
6    maqc                                   20167110   human
                                X4                               X5
1  Number of biological replicates Number of uniquely aligned reads
2                               19                    2,197,622,796
3                               41                      834,584,950
4                                2                        8,670,342
5                                6                       41,356,738
6 14 (technical)**  2 (biological)                       71,970,164
               X6              X7              X8
1   ExpressionSet     Count table Phenotype table
2            link            link            link
```

Little cleanup

```
colnames(dat) = as.character(dat[1,])
dat = dat[-1,]
head(dat)
```

```
       Study                                       PMID Species
2    bodymap not published, but publicly available here   human
3     cheung                                   20856902   human
4       core                                   19056941   human
5      gilad                                   20009012   human
6       maqc                                   20167110   human
7 montgomery                                   20220756   human
   Number of biological replicates Number of uniquely aligned reads
2                               19                    2,197,622,796
3                               41                      834,584,950
4                                2                        8,670,342
5                                6                       41,356,738
6 14 (technical)**  2 (biological)                       71,970,164
7                               60                     *886,468,054
    ExpressionSet     Count table Phenotype table
2            link            link            link
3            link            link            link
4            link            link            link
5            link            link            link
6 original pooled original pooled original pooled
7            link            link            link
                                            Notes
2 Illumina Human BodyMap 2.0 -- tissue comparison
3                                    HapMap - CEU
```

APIs

Application Programming Interfaces

figure https://developers.facebook.com/

In biology too!

http://www.ncbi.nlm.nih.gov/books/NBK25501/

figure

Step 0: Did someone do this already

https://ropensci.org/

Do it yourself

Read the docs

https://developer.github.com/v3/

Read the docs

Read the docs

A dissected example

https://api.github.com/search/repositories?q=created:2014-08-13+language:r+-user:cran

The base URL

https://api.github.com/search/repositories?q=created:2014-08-13+language:r+-user:cran

The Path: Search repositories

https://api.github.com/**search/repositories**?q=created:2014-08-13+language:r+-user:cran

Create a query - pass the `q` parameter

https://api.github.com/search/repositories?q=created:2014-08-13+language:r+-user:cran

Date repo was created

https://api.github.com/search/repositories?q=**created:2014-08-13**+language:r+-user:cran

Language repo is in

https://api.github.com/search/repositories?q=created:2014-08-13+**language:r**+-user:cran

Ignore repos from "cran"

https://api.github.com/search/repositories?q=created:2014-08-13+language:r+-user:cran

```r
#install.packages("httr")
library(httr)

query_url = paste0("https://api.github.com/", "search/repositories",
                   "?q=created:2014-08-13", "+language:r", "+-user:cran")

req = GET(query_url)
names(content(req))
```

```
[1] "total_count"        "incomplete_results" "items"
```

Not all APIs are “open”

https://apps.twitter.com/

```
myapp = oauth_app("twitter",
                   key="yourConsumerKeyHere",secret="yourConsumerSecretHere")
sig = sign_oauth1.0(myapp,
                     token = "yourTokenHere",
                      token_secret = "yourTokenSecretHere")
homeTL = GET("https://api.twitter.com/1.1/statuses/home_timeline.json", sig)

But you can get cool data
json1 = content(homeTL)
json2 = jsonlite::fromJSON(toJSON(json1))
json2[1,1:4]
```

```
                    created_at           id             id_str
1 Mon Jan 13 05:18:04 +0000 2014 4.225984e+17 422598398940684288

1 Now that P. Norvig's regex golf IPython notebook hit Slashdot, let's see if
```

Web + APIs lab

https://SISBIB.github.io/Module1/labs/web-api-lab.Rmd