Week 5: Connecting with external data sourcess

IST 687

Corey Jackson

2021-02-10 21:19:05

Today's Agenda

- Announcements
- Breakout I (Complete Lab 5)
- Homework 5 Tips
- Next week's agenda
- Breakout II (Group Project Meeting)

Announcements

- Office Hours: Wed. 6-7pm EDT and by appointment
- Week 4 videos?
- R Cheetsheets
- Questions/concerns?

Week 4 Inferential statistics (Review)

Week 4 Inferential statistics (Review)

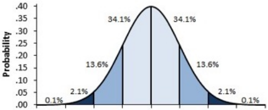
- Sampling from a population
- Replicating expressions
- Making inferences about a population statistic (is it extreme?)

Week 4: Sampling and Inference (Review)

- It may be necessary to draw samples from a population (or dataset).
- Samples can be drawn using `sample(x, n, replace = FALSE)` (values must be integers)
  - *x* is the population, *size* is the number of samples, and *replace* is whether to put a value back in the X once it't been drawn from the population.

Week 4: Sampling and Inference (Review)

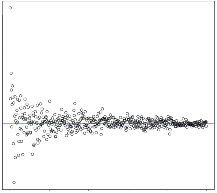
- We can make inferences about populations using single statistics (e.g., a sample mean) and comparing them to a distribution of values.



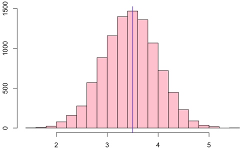
A possible exam question: Describe how you would compare two datasets, to determine if they were from the same population.

Week 4: Important terminology/code (Review)

Law of large numbers: As the size of a sample drawn from a random variable increases, the mean of more samples gets closer and closer to the true population mean.



Central limit theorem: Given a dataset with unknown distribution (it could be uniform, binomial or completely random), the sample means will approximate the normal distribution.



Week 4: Replication (Review)

- Replicating processes using `rep(expr, n)` and `replicate(n, expr)`

`rep(c("Corey","Home"),3)` or
`replicate(3,c("Corey","Home"))`

```
## [1] "Corey" "Home"  "Corey" "Home"  "Corey" "Home"
```

Lab 5: Storage Wars

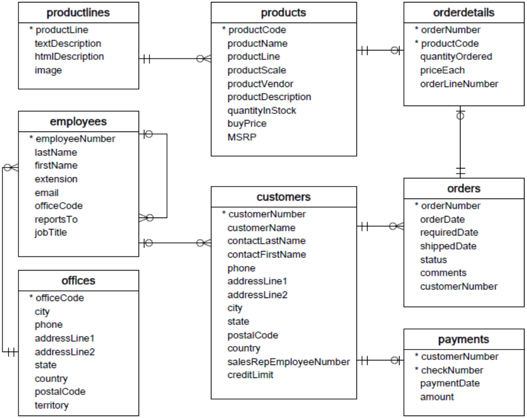
Lab 5: Storage Wars

Lab Goals:

- Using Structured Query Language (SQL)
- Subsetting data using SQL and R
- Subsetting data using SQL and R with conditionals

Groups for Pair Programming

Lab 5: Working with databases



productlines
* productLine
textDescription
htmlDescription
image

employees
* employeeNumber
lastName
firstName
extension
email
officeCode
reportsTo
jobTitle

offices
* officeCode
city
phone
addressLine1
addressLine2
state
country
postalCode
territory

products
* productCode
productName
productLine
productScale
productVendor
productDescription
quantityInStock
buyPrice
MSRP

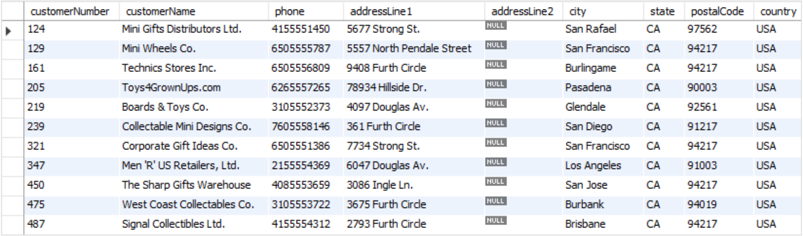
customers
* customerNumber
customerName
contactLastName
contactFirstName
phone
addressLine1
addressLine2
city
state
postalCode
country
salesRepEmployeeNumber
creditLimit

orderdetails
* orderNumber
* productCode
quantityOrdered
priceEach
orderLineNumber

orders
* orderNumber
orderDate
requiredDate
shippedDate
status
comments
customerNumber

payments
* customerNumber
* checkNumber
paymentDate
amount

Lab 5: Working with databases



customerNumber	customerName	phone	addressLine1	addressLine2	city	state	postalCode	country
124	Mini Gifts Distributors Ltd.	4155551450	5677 Strong St.	NULL	San Rafael	CA	97562	USA
129	Mini Wheels Co.	6505555787	5557 North Pendale Street	NULL	San Francisco	CA	94217	USA
161	Technics Stores Inc.	6505556809	9408 Furth Circle	NULL	Burlingame	CA	94217	USA
205	Toys4GrownUps.com	6265557265	78934 Hillside Dr.	NULL	Pasadena	CA	90003	USA
219	Boards & Toys Co.	3105552373	4097 Douglas Av.	NULL	Glendale	CA	92561	USA
239	Collectable Mini Designs Co.	7605558146	361 Furth Circle	NULL	San Diego	CA	91217	USA
321	Corporate Gift Ideas Co.	6505551386	7734 Strong St.	NULL	San Francisco	CA	94217	USA
347	Men 'R' US Retailers, Ltd.	2155554369	6047 Douglas Av.	NULL	Los Angeles	CA	91003	USA
450	The Sharp Gifts Warehouse	4085553659	3086 Ingle Ln.	NULL	San Jose	CA	94217	USA
475	West Coast Collectables Co.	3105553722	3675 Furth Circle	NULL	Burbank	CA	94019	USA
487	Signal Collectibles Ltd.	4155554312	2793 Furth Circle	NULL	Brisbane	CA	94217	USA

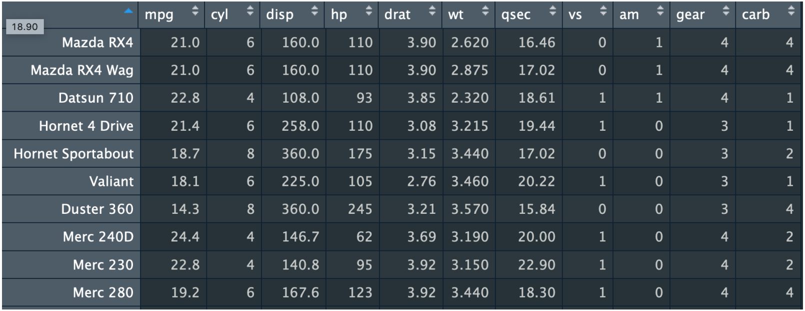
Lab 5: Working with SQL

- SQL is a standard language for storing, manipulating and retrieving data in databases.
- The basics of SQL:

SELECT *column1*, *column2*, *. . .*
FROM *table-name*

```
SELECT customerName
FROM customers
```

Lab 5: Working with SQL



	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
Mazda RX4	21.0	6	160.0	110	3.90	2.620	16.46	0	1	4	4
Mazda RX4 Wag	21.0	6	160.0	110	3.90	2.875	17.02	0	1	4	4
Datsun 710	22.8	4	108.0	93	3.85	2.320	18.61	1	1	4	1
Hornet 4 Drive	21.4	6	258.0	110	3.08	3.215	19.44	1	0	3	1
Hornet Sportabout	18.7	8	360.0	175	3.15	3.440	17.02	0	0	3	2
Valiant	18.1	6	225.0	105	2.76	3.460	20.22	1	0	3	1
Duster 360	14.3	8	360.0	245	3.21	3.570	15.84	0	0	3	4
Merc 240D	24.4	4	146.7	62	3.69	3.190	20.00	1	0	4	2
Merc 230	22.8	4	140.8	95	3.92	3.150	22.90	1	0	4	2
Merc 280	19.2	6	167.6	123	3.92	3.440	18.30	1	0	4	4

Lab 5: Working with SQL

- Return only mpg and cyl fields from `mtcars`
  - SELECT mpg,hp FROM mtcars

```
##   mpg  hp
## 1  21 110
## 2  21 110
```

- Return all data from `mtcars`
  - SELECT * FROM mtcars

```
##   mpg cyl disp  hp drat    wt  qsec vs am gear carb
## 1  21   6  160 110  3.9 2.620 16.46  0  1    4    4
## 2  21   6  160 110  3.9 2.875 17.02  0  1    4    4
```

Lab 5: Working with SQL

- SQL also takes conditionals using the WHERE clause
- Return mpg,disp,cyl from mtcars where the number of cylinders is 6

```
"SELECT mpg,disp,cyl FROM mtcars WHERE cyl = 6"
```

```
##    mpg  disp cyl
## 1 21.0 160.0   6
## 2 21.0 160.0   6
## 3 21.4 258.0   6
## 4 18.1 225.0   6
## 5 19.2 167.6   6
## 6 17.8 167.6   6
## 7 19.7 145.0   6
```

Lab 5: Working with SQL

- Applying functions over columns
- Get the minimum value in the mpg field in `mtcars`
    - SELECT min(mpg) FROM mtcars

```
##   min(mpg)
## 1     10.4
```

Note: You'll need to find out the appropirate functions for SQL queries for today's lab. Check w3schools.com .

Lab 5: Working with SQL (subqueries)

- You can supply the results of one query as the conditional of another.
- *Scenario*: Return the mpg, disp, and cyl for cars whose cylinder (cyl) match the minimum cylinder value in the data.

```
SELECT min(mpg) FROM mtcar
```

```
##   min(cyl)
## 1        4
```

```
SELECT mpg,disp,cyl FROM mtcars WHERE cyl = (select
min(cyl) from mtcars)
```

```
##    mpg  disp cyl
## 1 22.8 108.0   4
## 2 24.4 146.7   4
## 3 22.8 140.8   4
## 4 32.4  78.7   4
## 5 30.4  75.7   4
```

Lab 5: sqldf in R

- An R package for SQL queries `install.packages("sqldf")` and the function `sqldf()` to write SQL statements
- Using SQL statements in R requires packages to translate SQL to R language.
  - In a relational database:

    ```
    SELECT mpg,disp,cyl FROM mtcars WHERE cyl =
    (select min(cyl) from mtcars)
    ```
  - In R:

    ```
    sqldf(" SELECT mpg,disp,cyl FROM mtcars WHERE cyl
    = (select min(cyl) from mtcars)")
    ```

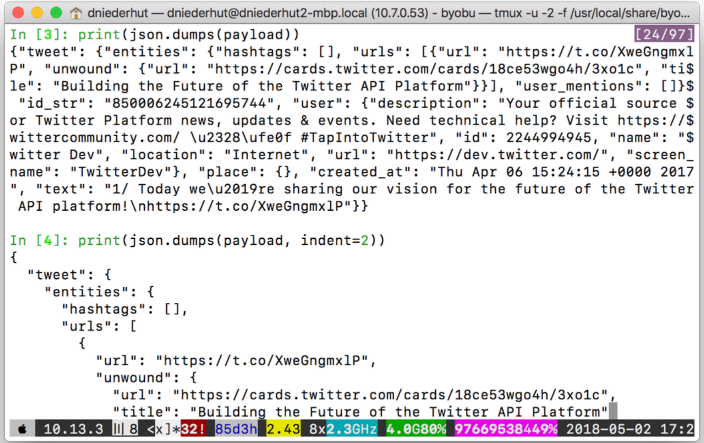
Homework 5

Homework 5 Tips

- Working with JSON data
- Aggregating data using `tapply()`
- Errors in the data

Homework 5 Tips: About JSON

- Many systems rely on non-SQL data stores e.g., MongoDB which outputs JSON document.

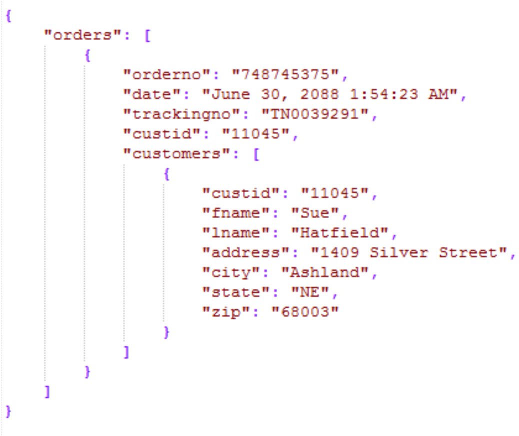

dniederhut — dniederhut@dniederhut2-mbp.local (10.7.0.53) - byobu — tmux -u -2 -f /usr/local/share/byo...
[24/97]

```
In [3]: print(json.dumps(payload))
{"tweet": {"entities": {"hashtags": [], "urls": [{"url": "https://t.co/XweGngmxl
P", "unwound": {"url": "https://cards.twitter.com/cards/18ce53wgo4h/3xo1c", "ti$
le": "Building the Future of the Twitter API Platform"}}], "user_mentions": []}$
 "id_str": "850006245121695744", "user": {"description": "Your official source $
or Twitter Platform news, updates & events. Need technical help? Visit https://$
wittercommunity.com/ \u2328\ufe0f #TapIntoTwitter", "id": 2244994945, "name": "$
witter Dev", "location": "Internet", "url": "https://dev.twitter.com/", "screen_
name": "TwitterDev"}, "place": {}, "created_at": "Thu Apr 06 15:24:15 +0000 2017
", "text": "1/ Today we\u2019re sharing our vision for the future of the Twitter
 API platform!\nhttps://t.co/XweGngmxlP"}}

In [4]: print(json.dumps(payload, indent=2))
{
  "tweet": {
    "entities": {
      "hashtags": [],
      "urls": [
        {
          "url": "https://t.co/XweGngmxlP",
          "unwound": {
            "url": "https://cards.twitter.com/cards/18ce53wgo4h/3xo1c",
            "title": "Building the Future of the Twitter API Platform"
```

 10.13.3 ||| 8 <x]*32! 85d3h 2.43 8x2.3GHz 4.0G80% 97669538449% 2018-05-02 17:2

Homework 5 Tips: About JSON

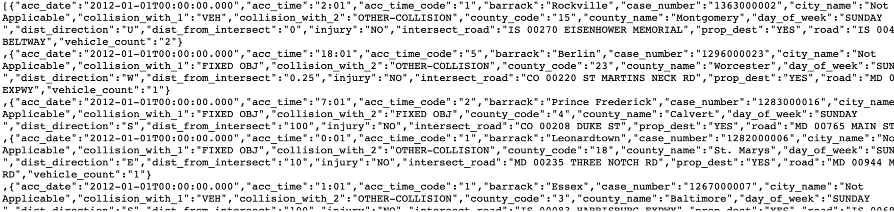


```
{
    "orders": [
        {
            "orderno": "748745375",
            "date": "June 30, 2088 1:54:23 AM",
            "trackingno": "TN0039291",
            "custid": "11045",
            "customers": [
                {
                    "custid": "11045",
                    "fname": "Sue",
                    "lname": "Hatfield",
                    "address": "1409 Silver Street",
                    "city": "Ashland",
                    "state": "NE",
                    "zip": "68003"
                }
            ]
        }
    ]
}
```

- Explore the data before importing, use a JavaScript Object Notation (JSON) converter

Homework 5 Tips: Working with JSON (Step 1)

- An example: Maryland DOT



```
[{"acc_date":"2012-01-01T00:00:00.000","acc_time":"2:01","acc_time_code":"1","barrack":"Rockville","case_number":"1363000002","city_name":"Not
Applicable","collision_with_1":"VEH","collision_with_2":"OTHER-COLLISION","county_code":"15","county_name":"Montgomery","day_of_week":"SUNDAY
","dist_direction":"U","dist_from_intersect":"0","injury":"NO","intersect_road":"IS 00270 EISENHOWER MEMORIAL","prop_dest":"YES","road":"IS 004
BELTWAY","vehicle_count":"2"}
,{"acc_date":"2012-01-01T00:00:00.000","acc_time":"18:01","acc_time_code":"5","barrack":"Berlin","case_number":"1296000023","city_name":"Not
Applicable","collision_with_1":"FIXED OBJ","collision_with_2":"OTHER-COLLISION","county_code":"23","county_name":"Worcester","day_of_week":"SUN
","dist_direction":"W","dist_from_intersect":"0.25","injury":"NO","intersect_road":"CO 00220 ST MARTINS NECK RD","prop_dest":"YES","road":"MD 0
EXPWY","vehicle_count":"1"}
,{"acc_date":"2012-01-01T00:00:00.000","acc_time":"7:01","acc_time_code":"2","barrack":"Prince Frederick","case_number":"1283000016","city_name
Applicable","collision_with_1":"FIXED OBJ","collision_with_2":"FIXED OBJ","county_code":"4","county_name":"Calvert","day_of_week":"SUNDAY
","dist_direction":"S","dist_from_intersect":"100","injury":"NO","intersect_road":"CO 00208 DUKE ST","prop_dest":"YES","road":"MD 00765 MAIN ST
,{"acc_date":"2012-01-01T00:00:00.000","acc_time":"0:01","acc_time_code":"1","barrack":"Leonardtown","case_number":"1282000006","city_name":"No
Applicable","collision_with_1":"FIXED OBJ","collision_with_2":"OTHER-COLLISION","county_code":"18","county_name":"St. Marys","day_of_week":"SUN
","dist_direction":"E","dist_from_intersect":"10","injury":"NO","intersect_road":"MD 00235 THREE NOTCH RD","prop_dest":"YES","road":"MD 00944 M
RD","vehicle_count":"1"}
,{"acc_date":"2012-01-01T00:00:00.000","acc_time":"1:01","acc_time_code":"1","barrack":"Essex","case_number":"1267000007","city_name":"Not
Applicable","collision_with_1":"VEH","collision_with_2":"OTHER-COLLISION","county_code":"3","county_name":"Baltimore","day_of_week":"SUNDAY
","dist_direction":"S","dist_from_intersect":"100","injury":"NO","intersect_road":"IS 00083 HARRISBURG EXPWY","prop_dest":"YES","road":"IS 0069
```

- Importing the data in R requires the use of a package called `RJSONIO`.

Homework 5 Tips: Working with JSON (Step 1)

- Data are imported in a lists that must be **unlisted**
- I imported the data from HW 5 and stored it in an object called `mydata`. Lets take a look...
  `summary(mydata)`

```
##      Length Class  Mode
## meta     1  -none- list
## data 18638  -none- list
```

- `unlist()` – takes a list and returns a simple vector
- You need to unlist the second element in `mydata` and place it in a `data.frame` (p. 118)

Homework 5 Tips: Aggregating using `tapply()`

- Data may need to be aggregated to get quick summaries e.g., mean score per student, time spent on a website per day. The `tapply()` function can be used.
- Three important arguments for `tapply(X, INDEX, FUNCTION)`
  - Think of X as the variable you want to compute, INDEX as the grouping variable, and function as the summary statistic you want to apply to X.
- An example using `mtcars()`. Get the mean mpg by cylinder. What would the `tapply` expression look like for our example?

```
##  [1] "mpg"  "cyl"  "disp" "hp"   "drat" "wt"   "qsec" "v
## [11] "carb"
```

Homework 5 Tips: Aggregating using `tapply()`

```
tapply(mtcars$mpg,mtcars$cyl,mean)
```

```
##        4        6        8
## 26.66364 19.74286 15.10000
```

Note: `na.rm = TRUE` can also be used to ignore columns containing `NAs`

e.g., `tapply(mtcars$mpg,mtcars$cyl,mean, na.rm=TRUE)`

Homework 5 Tips: Users errors (Step 3 and 4)

- Counting the number of characters in an object. What's the difference here?

"Corey"

```
nchar("Corey")
```

```
## [1] 5
```

"Corey "

nchar("Corey ")

```
## [1] 6
```

- Hint: Explore the data using `str()`. The `TRIM()` function for SQL and `gsub()` for R may be useful.

Group Project Meetings

Project update standard deliverables

Kanban Board: Your project to do-list

Project summary document: There are four questions: (1) What was accomplished since the last update (or since the project started) – these should be highlighted on the Kanban board, (2) What is working well for the team, (3) Plans for the next update, (4) Issues / what is not working well

Team process agreement: Contains information delineating responsibilities for the final project analysis, presentation, and summary document. Aware that change happens, this document can be updated at any time after the initial submission. *Due February 19th at 11:59pm AOE*

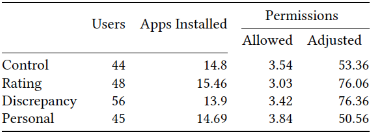
Project update specific deliverables

Exploratory Data Analysis: The goal here is to understand your dataset and explore important variables of interest. This will require a combination of coding and written descriptions. Produce analysis that:

- describes the dataset e.g., "There were 434 patient records in our dataset"
- describes key variables e.g., "male age = (mean = 35, sd = 2.3)"

Project update specific deliverables

EDA should make use of tables and written descriptions of data.



	Users	Apps Installed	Permissions	
			Allowed	Adjusted
Control	44	14.8	3.54	53.36
Rating	48	15.46	3.03	76.06
Discrepancy	56	13.9	3.42	76.36
Personal	45	14.69	3.84	50.56

A description: "A total of 241 participants completed our study: 151 participants (63%) were male, 89 participants (37%) were female, and one participant ($< 1\%$) chose not to answer. In terms of age, the majority of participants were between 21-30 (N=121, 50%), followed by 31-40 (N=83, 34%), and 41-50 (N=23, 10%)."

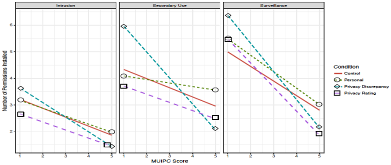
Source: Addressing The Privacy Paradox through Personalized Privacy Notifications (Jackson and Wang 2018)

Project update specific deliverables

Data Visualizations: The goal here is to understand your dataset through visualization. This will require a combination of coding and written descriptions of visualizations. Produce analysis that:

- shows data in graphical format and describes the components
- describes to the audience what phenemeonon you are observing in the data

Project update specific deliverables



Description: “A line chart showing the relationship between MUIPC concern scores and the number of related permissions granted by users in each experiment condition. The x-axis shows a user concern score while the y-axis shows the average number of permissions granted.”

Interpretation: “We observed a negative relationship across the board, where as concern increases, the number of permissions granted decreases. This suggests users who were not concerned granted more permissions.”

Next Week

- **Asynchronous**
  - Week 6 Introduction to visualization; Chapter 12
  - Submit HW 5 and Lab 5
  - Project update 2 (submit to LMS Tuesday by 11:59pm)
- **Live Session**
  - Lab 6: Data Viz
  - Project Updates: Come prepared to discuss your dataset, decision maker, research questions, and preliminary visualizations