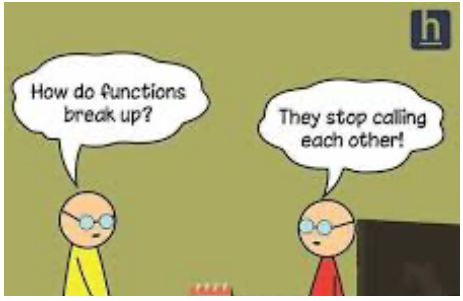
If you can and you want to, turn on your camera so I feel less lonely :)



DATA 8
Spring 2021

Discussion 5

Functions and Table Programming

Materials: tinyurl.com/d8-disc 5
or access through kevin-miao.com
under teaching

Today

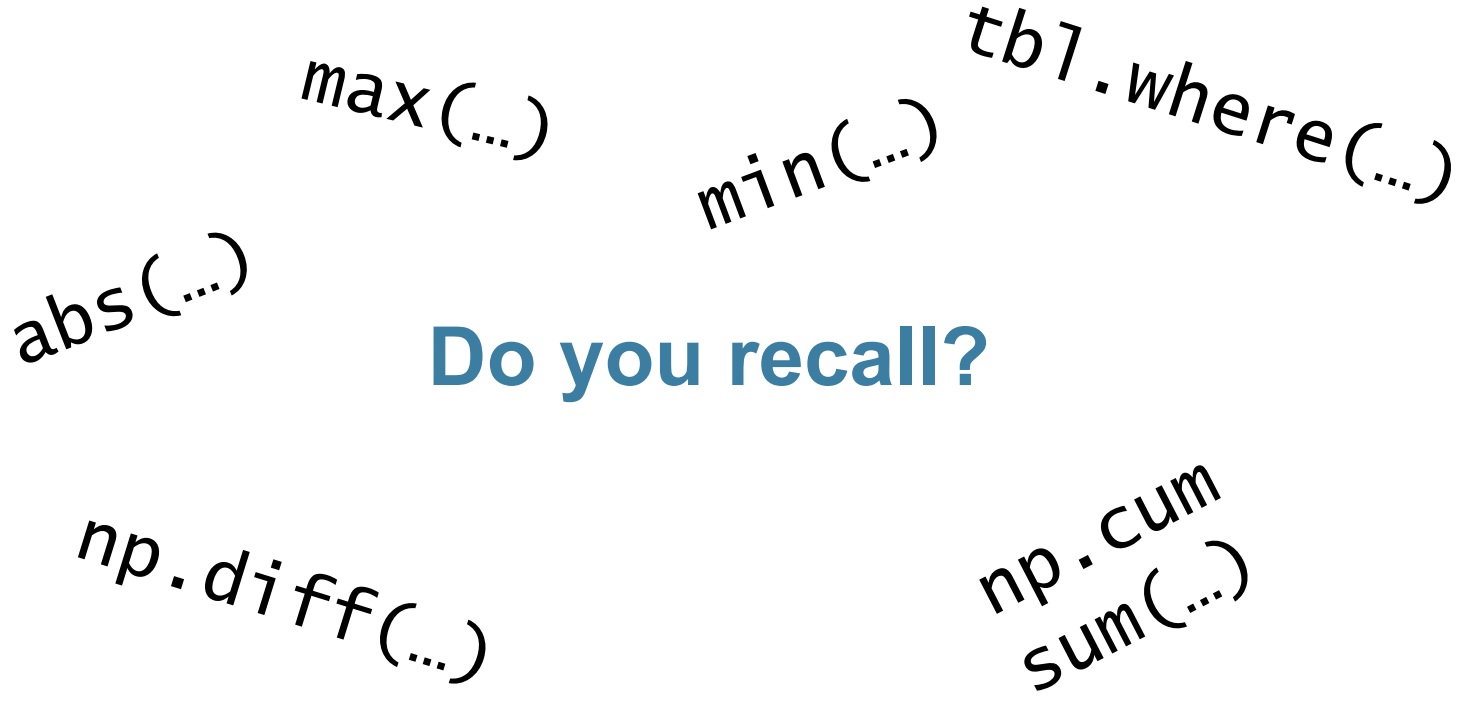
- Announcements
- Review: Functions, Groups, ~~Pivots~~
- Worksheet
  - This week's worksheet is very long and contains a lot of programming questions too.
  - I created an auxiliary notebook that you can use/play around with!
  - Link: www.tinyurl.com/d8-disc 5

Announcements

* Regrades for lab/HW, please email me!

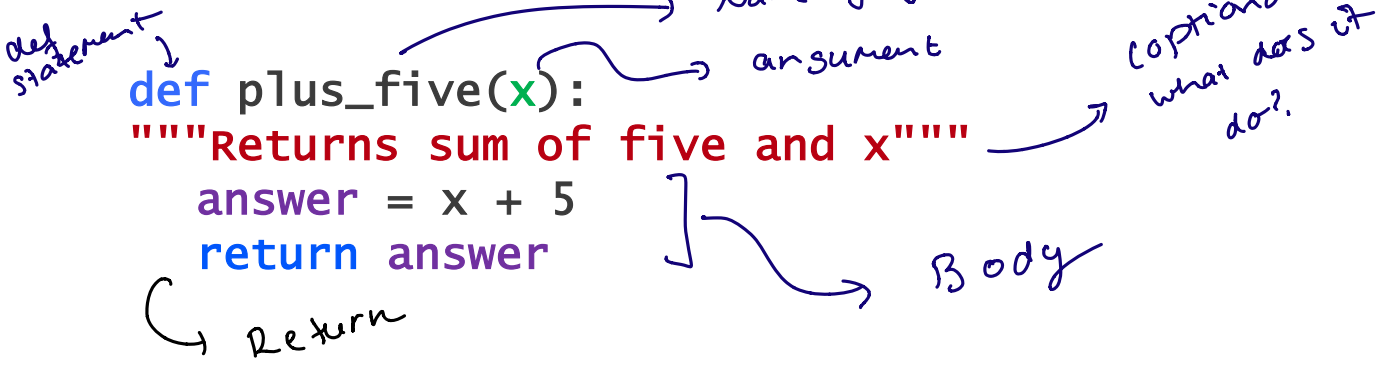
- **Homework 4** due date has been pushed back to Sunday
  - Early submission will be due Saturday
- **Project 1 checkpoint** will be due this Friday
  - Must finish all the question up to the checkpoint & pass public tests
  - Ensure that your partner is added on OkPy
- **No vitamin question** during discussion today
- *Informal OH:* Feel free to stay after discussion, if you have homework/project/course related questions. I booked off time from 9-9:30 AM.

* Students who are in TX, hang in there!


max(…)
min(…)
tbl.where(…)
abs(…)
Do you recall?
np.diff(…)
np.cum
sum(…)

Functions

- They all are **functions**! Someone else just wrote them for you.
- Now we will start writing our own functions
  - Now, we don't have to type the same code again and again
- **Anatomy of a function**



Group and Pivot

- **Group**

  `tbl.group(column(s), func)`

  - We take a column (or columns) and group together all values that are the same
  - Then we call **func** on it (i.e. average, median)
  - If you don't specify a function, it will default to count

- **Pivot** ← will learn in lecture today!

`tbl.pivot(col1, col2, values, collect)`

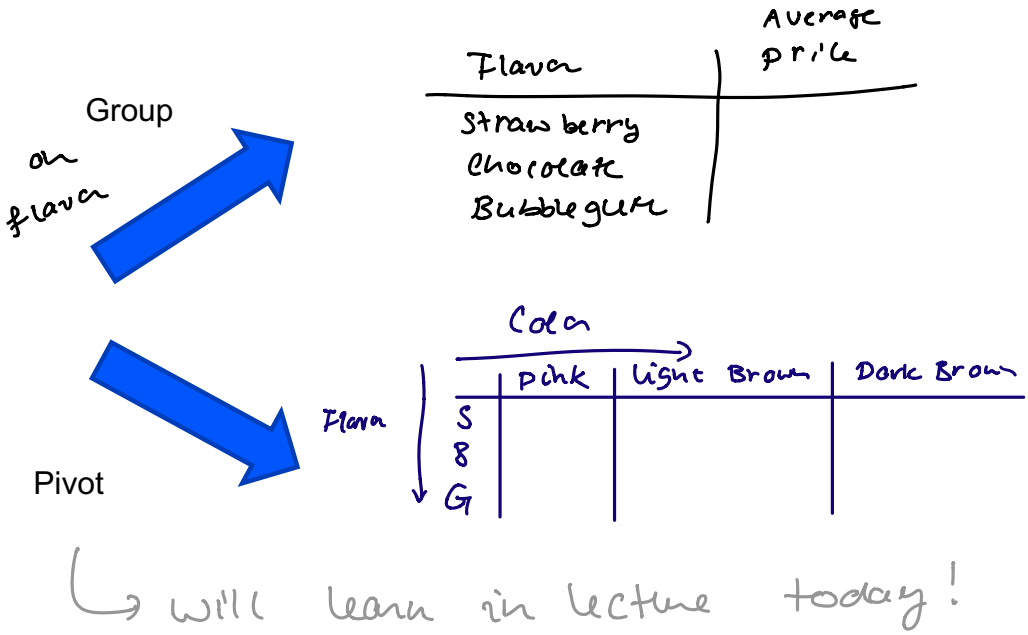
  - We take two columns and split them out over the x- and y-axis
  - Then we call `collect` on `values` (i.e. average, median)
  - Kind of like grouping, but then 2D

Flavor	Color	Price
strawberry	pink	3.55
chocolate	light brown	4.75
chocolate	dark brown	5.25
strawberry	pink	5.25
chocolate	dark brown	5.25
bubblegum	pink	4.75

Group and Pivot

Let's pretend we take the average

Flavor	Color	Price
strawberry	pink	3.55
chocolate	light brown	4.75
chocolate	dark brown	5.25
strawberry	pink	5.25
chocolate	dark brown	5.25
bubblegum	pink	4.75



To the worksheet! ✍️

tinyurl.com/d8-disc 5

Question 1a

Question 1. Fun with Functions

a. The following code has a number of errors in it. Which ones can you identify?

```
def hypotenuse(a, b):
    ```Returns the length of the hypotenuse of a right triangle,
    the squareroot of a squared + b squared```
    squares = make_array(a, b)*2
    sum = sum(squares)
    squareroot = np.sqrt(sum)
    print(squareroot)
A = 5
B = 5
C = squareroot
```

colon

* we are taking a square

Return

can't call on a variable inside function

sum is already a function → we need another name
```

Question 1bc

b. Write a function that takes in one argument, a table `tbl`, another argument, a name of a column in that table `col`, and a boolean `largest`, and returns a table that contains the rows that have the ten largest or ten smallest values for the specified column, largest if the boolean `largest` is `True`, smallest if the boolean argument is `False`.

```
def top_ten(tbl, col, largest):
    sorted_tbl = tbl.sort(col, descending = largest)
    ten_rows = sorted_tbl.take(np.arange(10))
    return ten_rows
```

c. Can a function take no arguments? When would you use a function with no arguments? How do you call a function without arguments? How does that compare to using a function as an argument?

Yes, it's possible. If you want to write a function that's not dependent on the input.

Question 2

Question 2. Ian has opened up a chocolate store where he sells small boxes of chocolates in groups of different sizes and colors. His table `chocolates` is as follows:

Color	Shape	Amount	Price ($)
Dark	Round	4	1.30
Milk	Rectangular	6	1.20
White	Rectangular	12	2.00
Dark	Round	7	1.75
Milk	Rectangular	9	1.40
Milk	Round	2	1.00

Notice that the table contains multiple rows containing information about chocolates of the same color. We would like to figure out how many chocolates of each color he has for sale in total, and what the cost would be to purchase all chocolates of each unique color.

a. Write a line of code that will return a new table which displays the total number of boxes for each color.

```
chocolates.groupC'column')
```

b. Write a line of code which will return a new table with the total number of chocolates and the total cost for each unique color. For example, the row for “Dark” should have a total of 4+7=11 chocolates, and a total cost of $1.30 + $1.75 = 3.05.

```
chocolates.drop('shape').groupC'color', sum)
```

Question 3abc

Question 3. Some rows from the table `ca` are shown below. The table contains information about the most common baby names in California and the number of those occurrences in a particular year, from the years 1910-2019. (This dataset was submitted by a fellow Data 8 student!)

State	Sex	Year	Name	Occurrence
CA	F	1910	Mary	295
CA	F	1910	Helen	239
CA	F	1910	Dorothy	220
CA	F	1910	Margaret	163
CA	F	1911	Mary	10

a. Write a line of code that will return the most popular name over all the years.
Hint: Think about how to use the second argument in `.group`

```
ca.group('Name', sum).sort('occurrence sum', descending = True).column('Name').item(0)
```

b. Instead of the most popular name over all the years, write a line of code that will return the top 10 most popular names over all the years.

c. The top 10 names all appeared to be male names. Write a line of code that would return the most popular female names instead.

```
ca.where('Sex', are.equal_to('F')).group('Name', sum).sort('occurrence sum',
descending = True).take(np.arange(10)).column('Name')
```

Question 3def

State	Sex	Year	Name	Occurrence
CA	F	1910	Mary	295
CA	F	1910	Helen	239
CA	F	1910	Dorothy	220
CA	F	1910	Margaret	163

Question 3. Some rows from the table `ca` are shown below. The table contains information about the most common baby names in California and the number of those occurrences in a particular year, from the years 1910-2019. (This dataset was submitted by a fellow Data 8 student!)

d. Write a line of code that will return the most popular female name in 1969

e. Write a function `most_popular_female_name` that takes in a year as an argument and returns the most popular female name in that year.

```
def most_popular_female_name(year):
    return ca.where('Sex','F').where('Year',year).sort('Occurrence', descending=True).
           column('Name').item(0)
```

f. The `ca` table is from 1920-2019. Define the years table with a column year and a row for each year from 1910-2019 (inclusive). Then create the table `popular_female_names` that has 2 columns, a year and a column for the female name that is most popular.

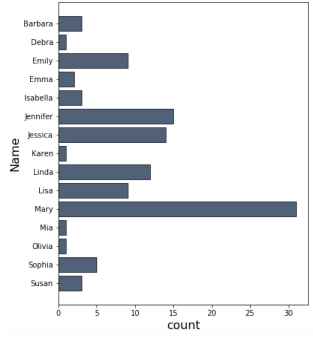
```
years = Table.with_column('Year', np.arange(1910, 2020))
most_popular_female_names_array = years.apply(most_popular_female_name, 'year')
popular_female_names = years.with_column('Popular', most_popular_female_name_array)
```

Question 3g

Question 3. Some rows from the table `ca` are shown below. The table contains information about the most common baby names in California and the number of those occurrences in a particular year, from the years 1910-2019. (This dataset was submitted by a fellow Data 8 student!)

State	Sex	Year	Name	Occurrence
CA	F	1910	Mary	295
CA	F	1910	Helen	239
CA	F	1910	Dorothy	220
CA	F	1910	Margaret	163

g. Write a line of code that will generate the following bar chart:



End of Section
How did I do?

https://tinyurl.com/kevind8feedback