Introduction to Data Science – Week 12

NETA LIVNEH

COURSE 55793

2019-2020

Last week recap

Jupyter notebook

Introduction to Numpy

Introduction to Pandas

This week

Introduction to Pandas – continued!

SQL kind of thinking

Selecting from a Dataframe

- Select a column by using a dot notation or the column label in bracket
- To select multiple columns insert a list of column labels
- Use df.loc[rows filter, column labels] to filter
- To select by index use df.iloc[]

```
>>> df.loc[df['Age'] > 30, ['Name', 'Age']]
        Name  Age
rank2   Jack   34
rank4  Ricky   42
```

```
>>> df.Name
rank1      Tom
rank2     Jack
rank3    Steve
rank4    Ricky
Name: Name, dtype: object
>>> df['Age']
rank1    28
rank2    34
rank3    29
rank4    42
Name: Age, dtype: int64
```


Filtering a db using Query

- To filter a database, you can also use df.query()
- The function receives a str that represents a boolean expression
- You can address the columns by their name (using quotes if the name has a space in it)
- Inplace parameter (default False), whould modify the data in place or not

- For example:

```
df.query('A > B')
```

equivalent to:

```
df[df[A] > df[B]]
```


Rules for specifying multiple filter criteria in Pandas

use **&** instead of **and**

use **|** instead of **or**

add **parentheses** around each condition to specify evaluation order

Aggregating and grouping in Pandas

Any **groupby** operation involves one of the following operations on the original object:

- **Splitting** the Object
- **Applying** a function
- **Combining** the results

df.groupby(['columns_to_group_by'])['columns_to_aggregate'].agg_function()

```
nba.groupby('Position').size()
```

```
Position
C       78
PF     100
PG      92
SF      85
SG     102
dtype: int64
```

```
nba.groupby(['Team','Position'])['Weight'].agg('mean')
```

```
Team                Position
Atlanta Hawks       C           250.000000
                    PF          239.500000
                    PG          179.000000
                    SF          210.500000
                    SG          208.000000
Boston Celtics      C           250.333333
                    PF          235.333333
                    PG          193.750000
                    SF          235.000000
                    SG          206.250000
```


Julia Evans
@b0rk

SQL query steps

When this SQL query runs, here's how I think of what happens:
every line in the query changes a table into another table

```
5  SELECT owner, count(*)
1  FROM cats
2  WHERE owner != 3
3  GROUP BY owner
4  HAVING count(*) = 2
6  ORDER BY owner DESC
```

① FROM cats

owner	name
1	libra
2	cinnamon
2	chanceuse
3	astra
4	lime
4	nikola

② WHERE owner != 3

owner	name
1	libra
2	cinnamon
2	chanceuse
~~3~~	~~astra~~ ← filter out this row
4	lime
4	nikola

③ GROUP BY owner

owner	name
1	libra
2	cinnamon
2	chanceuse
4	lime
4	nikola

④ HAVING count(*) = 2

owner	name
2	cinnamon
2	chanceuse
4	lime
4	nikola

⑤ SELECT owner, count(*)

owner	count(*)
→ 2	2
→ 4	2

⑥ ORDER BY owner DESC

	owner	count(*)
sort ↓	4	2
	2	2