Exploratory Data Analysis

Pandas DataFrame

Jay Urbain, PhD

Topics

- Data science process
- Getting data
- Data - defined
- Descriptive statistics
- Visualization

Data Science Process

- Define problem/ask questions – hypothesize
- **Data collection**
- **Data exploration**
- Data modeling
- Data analysis
- Visualization and communication of results

Collecting data

- Internal sources
  - Already collected by an organization
  - E.g., transaction data, click-stream data, survey data, text, images
- External sources
  - Outside source through download or API
    - Web service, RDF (Resource Description Framework), RSS (Rich Site Summary) web feed, data download (UCI Machine Learning repository), web scraping.
    - E.g., Census, Medicaid, Google trends, Twitter API, Google Maps, stock market data, research datasets
  - Outside source where acquisition requires special processing
    - E.g., data on websites/web scraping, pictures, printed form, i.e., pdf.

APIs, wrappers

- HTML Parsing
  - curl requests and BeautifulSoup
- APIs
  - Makes the call for us (the author is “allowing us” to access the data)
  - http://www.pythonforbeginners.com/api/list-of-python-apis
  - Examples:
    - Google
    - Amazon (price data)
    - Twitter (tweets)
    - Facebook (social network)
    - Web services that we author

Data

- Datum
  - A qualitative or quantitative value
- Data
  - Set of such values

Data types

- Simple:
  - Numeric: integers, floats
  - Boolean: binary or true false values
  - Strings: sequence of symbols

- Complex:
  - Tuples, sets, lists, dictionaries, arrays, records
  - Date and time
  - Time series
  - Images
  - Temporal
  - Text
  - Geolocation

Types of data values

- **Quantitative variable**: is numerical and can be either:
  - discrete - a finite number of values are possible in any bounded interval.
    - For example: number of students is a discrete variable
  - continuous - an infinite number of values are possible in any bounded interval
    - For example: Height and weight are a continuous variables
- **Categorical variable**: no inherent ordering among the values
  - For example: {SE, CS BE, CE, EE}
- **Ordinal variable**: ordering among the values
  - For example: {assistant professor, associate professor, professor}

Data formats

- **Structured data**
  - Each data record represented in a potentially complex form: dictionary, graph, relational data, tabular data.
  - Data has type
- **Tabular data**
  - Structured data where each row represents a single record or sample
  - Each column represents a single feature or measurement
  - Example: database, csv, tsv, xlsx, etc.
- **Unstructured data**
  - Text
  - Images
- **Semi-structured data**
  - Mix of structured and unstructured data. E.g., XML, JSON, etc.

Tabular data

- Typically work with tabular data, often need to put data into tabular form.
- Each record is a single object or event, and has measurements.
- Each measurement is an attribute or value of the data
- The number of attributes is the dimension.
- Assume each record has the same kind of object

	ID	DUR	FREQ	NRS	RAD	PIDAY	PIWORK	PPISOC	PICHOR	LBS	...	SEX	EMP	EDU	HT	WT	RACE	PI	FUN	DEP	SLEEP
0	5	1	2	6	0	3	4	4	3	0	...	0	1	0	72	123	5	14	11	20	11
1	8	4	2	3	1	2	2	3	3	0	...	0	1	6	69	175	5	10	7	12	9
2	10	3	3	2	1	2	2	1	2	0	...	1	1	8	67	130	5	7	5	10	10
3	11	4	2	7	1	4	4	4	4	0	...	1	1	6	62	108	5	16	12	18	17

Scale

- **Ratio scale** (numerical):
  - Units are equally spaced
  - True zero
  - Mathematical operations of +, -, /, * are valid
  - E.g., height, weight
- **Interval scale** (numerical):
  - Units are equally spaced, but there is no true zero.
  - Mathematical operations of +, -, /, * are ***not*** valid
  - E.g., temperatures measured in Celsius or Fahrenheit, 0 degrees on a compass.
  - Never an absence of value. 0 degrees F is a meaningful value and is a temperature. 0 degrees on a compass is a direction.
- **Ordinal scale** (categorical)
  - The order of the units/categories is important.
  - E.g., grades: A, B, C, …
- **Nominal scale** (categorical, ordered=True):
  - Categories have no order with respect to another
  - E.g., major: SE, CS, ME, BE, EE, CE

Common data problems

- Missing values
  - How to fill in?, delete?
- Wrong values
  - How to identify, how to correct?
- Inconsistent values
  - Date formats, naming conventions
- Data integration
  - How to join data?
- Data not in a properly formatted table
  - Convert cross-tabulation to table
- Not applicable
  - Data can not answer our hypothesis

	ID	DUR	FREQ	NRS	RAD	PIDAY	PIWORK	PPISOC	PICHOR	LBS	LBST	FUS	OPI	INJ	EXE
0	5	1	2	6	0	3	4	4	3	0	nan	nan	1	1	1
1	8	4	2	3	1	2	2	3	3	0	nan	nan	0	0	1
2	10	3	3	2	1	2	2	1	2	0	nan	nan	0	1	1
3	11	4	2	7	1	4	4	4	4	0	nan	nan	0	0	1
4	12	3	3	6	1	3	4	2	3	0	nan	nan	0	0	0
5	13	3	3	7	1	4	4	4	4	0	nan	nan	0	0	1
6	14	3	1	2	0	1	2	1	2	0	nan	nan	nan	1	1
7	15	4	2	1	0	2	1	1	2	0	nan	nan	1	nan	1
8	16	3	2	5	0	3	3	3	3	0	nan	nan	0	0	nan
9	17	4	2	4	0	4	4	3	4	0	nan	nan	0	0	1
10	18	4	2	5	1	5	5	5	5	2	3	0	1	1	1

Messy data

- Measuring individual deliveries; the variables are Time, Day, and Number of Produce.

	Friday	Saturday	Sunday
Morning	15	158	10
Afternoon	2	90	20
Evening	55	12	45

- Problem: each column header represents a single value rather than a variable.
- Row headers are “hiding” the Day variable.
- The values of the variable, “Number of Produce”, is not recorded in a single column.

- *How to fix?*

Messy data

- We need to reorganize the information to make explicit the event were observing and the variables associated to this event.

ID	Time	Day	Number
1	Morning	Friday	15
2	Morning	Saturday	158
3	Morning	Sunday	10
4	Afternoon	Friday	2
5	Afternoon	Saturday	9
6	Afternoon	Sunday	20
7	Evening	Friday	55
8	Evening	Saturday	12
9	Evening	Sunday	45

Messiness

- What object or event are we measuring? What are the variables in this dataset?

Delivery	Amount
On Sunday	
10:30	43
12:30	12
12:35	30
On Monday	
11:30	29
11:57	87
11.59	63
On Tuesday	
11:33	19
11:15	27
12.59	54

- *How to fix?*

Messiness

- Measuring individual deliveries; the variables are Time, Day, Number of Produce:

Days	times	Amount
Sunday	10:30	43
Sunday	12:30	12
Sunday	12:35	30
Monday	11:30	29
Monday	11:57	87
Monday	11.59	63
Tuesday	11:33	19
Tuesday	11:15	27
Tuesday	12.59	54

Common causes of messiness

- Column headers are values, not variable name
- Variables are stored in rows or columns
- Multiple (different) variables are stored in one column
- Multiple types of experimental units stored in same table

Note:

- We want each file to correspond to a dataset, each column to represent a single variable, and each row to represent a single observation.
- We want to tabularize the data. This makes Python happy, makes us happy.
- Take the database course!

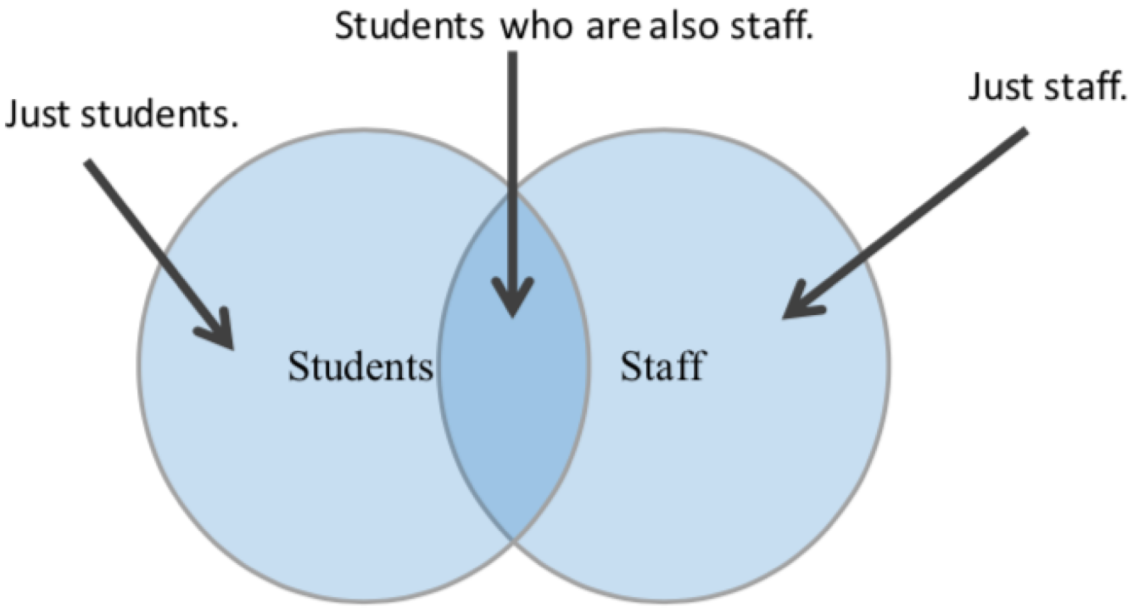
Pandas Data Structures

- Series (1 dimensional)
- DataFrame (2 dimensional, a table)
- Querying
  - *iloc[], for querying based on position (integer)*
  - *loc[], for querying rows based on label*
  - *Querying the DataFrame directly*
  - *Projecting a subset of columns*
    - *Use a Boolean mask to filter data*

Setting Data in Pandas

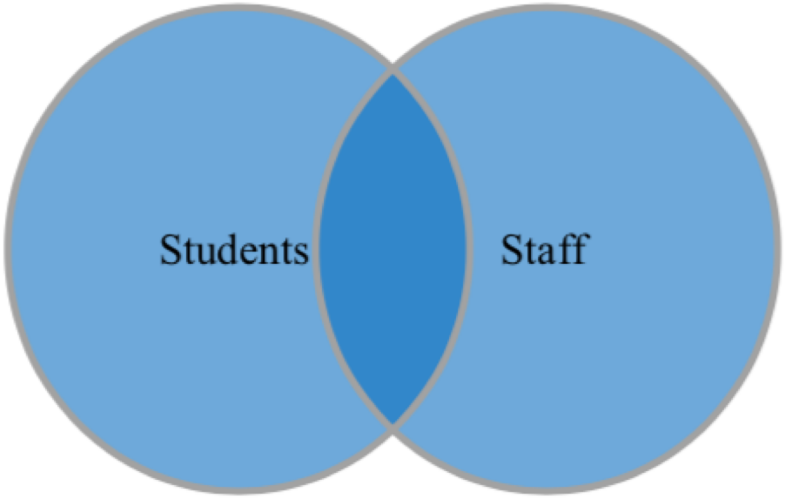
- To add new data
  - df[column] = [a, b, c] # assumes pandas col. is broadcast compatible with list
- To set default data (or overwrite all data):
  - *df[column]=2 # all values in column*

Venn Diagram for DataFrames



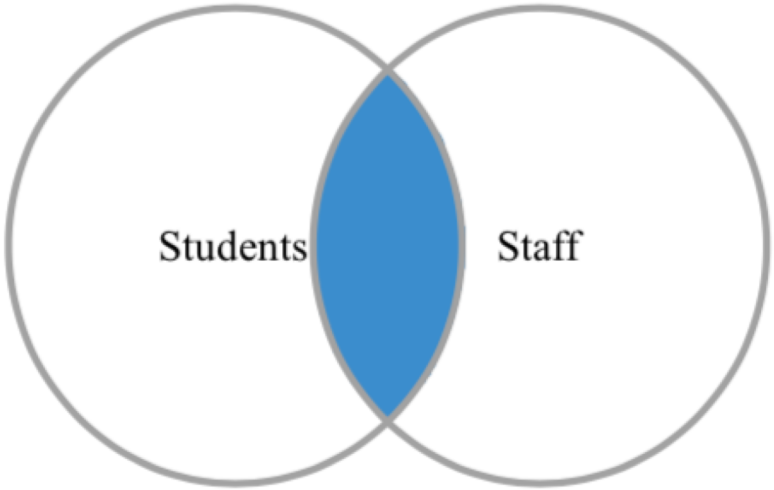
Details in hands-on tutorials

Full Outer join (union)



Details in hands-on tutorials

Inner join (intersection)



Details in hands-on tutorials

Aggregate statistics

- Need to summarize aggregate statistics

```
(df.set_index('STNAME').groupby(level=0)['POPESTIMATE2010', 'POPESTIMATE2011']
    .agg({'avg': np.average, 'sum': np.sum}))
```

	sum		avg	
	POPESTIMATE2010	POPESTIMATE2011	POPESTIMATE2010	POPESTIMATE2011
STNAME				
Alabama	4785161	4801108	71420.313433	71658.328358
Alaska	714021	722720	24621.413793	24921.379310
Arizona	6408208	6468732	427213.866667	431248.800000
Arkansas	2922394	2938538	38965.253333	39180.506667

Pivot tables

- Summarizing data in dataframe
- Rows represent one value, the columns are another value.
- Aggregates in cells. Allows you to see relationships between variables.

Pivot tables

```
In [317]: df = pd.read_csv('cars.csv')
```

```
In [318]: df.head()
```

Out[318]:

	YEAR	Make	Model	Size	(kW)	Unnamed: 5	TYPE	CITY (kWh/100 km)	HWY (kWh/100 km)	COMB (kWh/100 km)	CITY (Le/100 km)	HWY (Le/100 km)	COMB (Le/100 km)	(g/km)	RATIN
0	2012	MITSUBISHI	i-MIEV	SUBCOMPACT	49	A1	B	16.9	21.4	18.7	1.9	2.4	2.1	0	n/a
1	2012	NISSAN	LEAF	MID-SIZE	80	A1	B	19.3	23.0	21.1	2.2	2.6	2.4	0	n/a
2	2013	FORD	FOCUS ELECTRIC	COMPACT	107	A1	B	19.0	21.1	20.0	2.1	2.4	2.2	0	n/a
3	2013	MITSUBISHI	i-MIEV	SUBCOMPACT	49	A1	B	16.9	21.4	18.7	1.9	2.4	2.1	0	n/a
4	2013	NISSAN	LEAF	MID-SIZE	80	A1	B	19.3	23.0	21.1	2.2	2.6	2.4	0	n/a

```
In [319]: df.pivot_table(values='(kW)', index='YEAR', columns='Make', aggfunc=np.mean)
```

Out[319]:

Make	BMW	CHEVROLET	FORD	KIA	MITSUBISHI	NISSAN	SMART	TESLA
YEAR								
2012	NaN	NaN	NaN	NaN	49.0	80.0	NaN	NaN
2013	NaN	NaN	107.0	NaN	49.0	80.0	35.0	280.000000
2014	NaN	104.0	107.0	NaN	49.0	80.0	35.0	268.333333
2015	125.0	104.0	107.0	81.0	49.0	80.0	35.0	320.666667
2016	125.0	104.0	107.0	81.0	49.0	80.0	35.0	409.700000

Next

- Start Pandas notebook tutorials