Join our iClicker class:

https://join.iclicker.com/XSFZ

LECTURE 3

EDA & Wrangling Using Pandas

Using Pandas for Exploratory Data Analysis

CSCI 3022 @ CU Boulder

Maribeth Oscamou

Content credit: Acknowledgments

Meet The Course Team

Isabella Longo

Course Manager

Vincent Bowen

Course Manager

Kevin Buhler

Course Assistant

Grace Mudd

Course Assistant

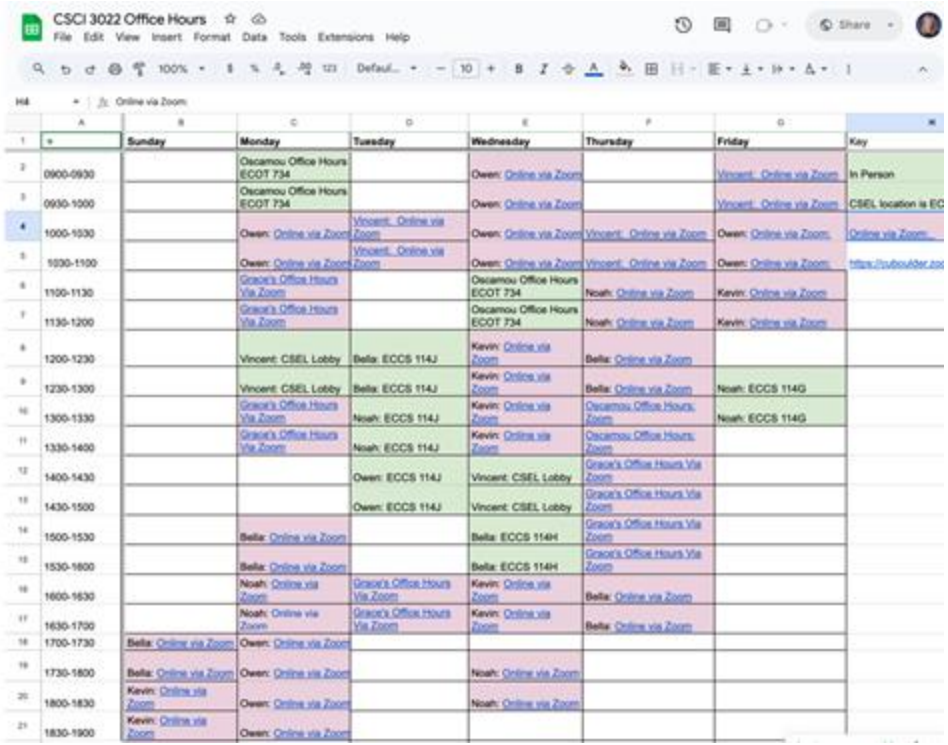
Noah Turner

Course Assistant

Owen Vangermeersch

Course Assistant

Office Hours:



CSCI 3022 Office Hours

	Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Key
0900-0930		Oscarnou Office Hours ECOT 734		Owen: Online via Zoom		Vincent: Online via Zoom	In Person
0930-1000		Oscarnou Office Hours ECOT 734		Owen: Online via Zoom		Vincent: Online via Zoom	CSEL location is EC
1000-1030		Owen: Online via Zoom	Vincent: Online via Zoom	Owen: Online via Zoom	Vincent: Online via Zoom	Owen: Online via Zoom;	Online via Zoom:
1030-1100		Owen: Online via Zoom	Vincent: Online via Zoom	Owen: Online via Zoom	Vincent: Online via Zoom	Owen: Online via Zoom;	https://ouboulder.zoo
1100-1130		Gracie's Office Hours Via Zoom		Oscarnou Office Hours ECOT 734	Noah: Online via Zoom	Kevin: Online via Zoom	
1130-1200		Gracie's Office Hours Via Zoom		Oscarnou Office Hours ECOT 734	Noah: Online via Zoom	Kevin: Online via Zoom	
1200-1230		Vincent: CSEL Lobby	Bella: ECCS 114J	Kevin: Online via Zoom	Bella: Online via Zoom		
1230-1300		Vincent: CSEL Lobby	Bella: ECCS 114J	Kevin: Online via Zoom	Bella: Online via Zoom	Noah: ECCS 114G	
1300-1330		Gracie's Office Hours Via Zoom	Noah: ECCS 114J	Kevin: Online via Zoom	Oscarnou Office Hours: Zoom	Noah: ECCS 114G	
1330-1400		Gracie's Office Hours Via Zoom	Noah: ECCS 114J	Kevin: Online via Zoom	Oscarnou Office Hours: Zoom		
1400-1430			Owen: ECCS 114J	Vincent: CSEL Lobby	Gracie's Office Hours Via Zoom		
1430-1500			Owen: ECCS 114J	Vincent: CSEL Lobby	Gracie's Office Hours Via Zoom		
1500-1530		Bella: Online via Zoom		Bella: ECCS 114H	Gracie's Office Hours Via Zoom		
1530-1600		Bella: Online via Zoom		Bella: ECCS 114H	Gracie's Office Hours Via Zoom		
1600-1630		Noah: Online via Zoom	Gracie's Office Hours Via Zoom	Kevin: Online via Zoom	Bella: Online via Zoom		
1630-1700		Noah: Online via Zoom	Gracie's Office Hours Via Zoom	Kevin: Online via Zoom	Bella: Online via Zoom		
1700-1730	Bella: Online via Zoom	Owen: Online via Zoom					
1730-1800	Bella: Online via Zoom	Owen: Online via Zoom		Noah: Online via Zoom			
1800-1830	Kevin: Online via Zoom	Owen: Online via Zoom		Noah: Online via Zoom			
1830-1900	Kevin: Online via Zoom	Owen: Online via Zoom					

https://canvas.colorado.edu/courses/117881/pages/hw-slash-office-hours

Jupyter Notebook and LaTeX Troubleshooting and Tips

Make sure before submitting to double check that your PDF includes all of the manually graded questions and plots, and that all code is fully visible in your PDF.

General best practices

- Make sure you have not renamed the .ipynb file. For example, HW 2 must be named hw02.ipynb
- Make sure you haven't inserted any new cells into the notebook.
- Make sure that you're in the 3022 instance of CSEL DataHub. You can do this by signing out of JupyterHub and then re-clicking the link. It should lead you to the page where you have to select the course "3022". The 3022 course has otter-grader installed in it. Other courses in the DataHub may not.
- If you make changes in your HW and run your export cell in your notebook more than once you should first delete the PDF (in the folder where the notebook is) and then re-run. It's possible that the version you submit is an earlier version of your HW.

First fixes to try

- **Save everything, delete the zip and pdf files and shut your browser window. Then open a new browser window and then restart your kernel and run through all of the cells and SAVE the nb before running the final export cell.**
- **As an extension, log out of coding.csel completely (after saving any work), close your browser, then launch a new one. Make sure you have selected CSCI 3022 as your coding environment.**

Latex Issues

- Check that there aren't any spaces after your dollar signs in LaTex:

https://docs.google.com/document/d/1ndr3Wj1PSF5qzlLMaBJznwh6QGeEXjd5TAJ6nf9EJvo/edit?usp=sharing

Course Logistics: Your First Week At A Glance

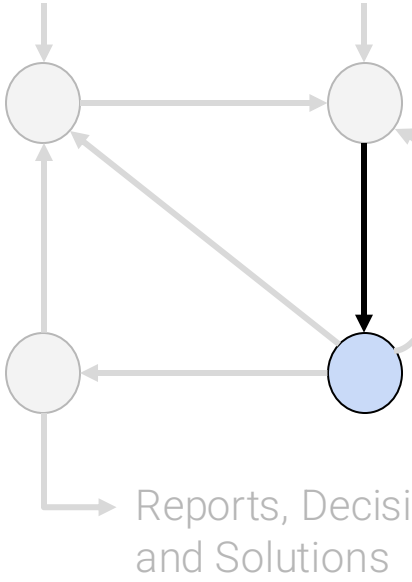
Mon 1/13	Tues 1/14	Wed 1/15	Thurs 1/16	Fri 1/17	
Attend & Participate in Class Office Hours Begin (See Schedule on Canvas)		Attend & Participate in Class	HW 1 Due 11:59pm via Gradescope (Includes Intro to CSCI 3022 Video assignment)	Attend & Participate in Class ~~In Class Quiz (beginning of class)~~	
				HW 2 released	

Getting To Know You:

I'd like to get a chance to be introduced to each of you!

1. **Please sign-up for a 15 min. timeslot (link on first announcement on Canvas and Piazza)** to meet with me during the first couple weeks to briefly introduce yourself and meet a few other classmates.

Plan for first 2 weeks



(Weeks 1 and 2)

EDA, Wrangling, and Data Visualization

Lesson 3 Learning Objectives:

- Identify 5 key data properties to consider when doing Exploratory Data Analysis
- Define what is meant by structure and granularity in terms of a set of data, and identify the structure and granularity of sample datasets
- Practice EDA with sample data

EDA & Wrangling

Lesson 3:

EDA - 5 key properties to consider

EDA Jupyter Demo

File Format
Variable Type
Multiple files
(Primary and Foreign Keys)

Structure -- the “shape” of a data file

Granularity -- how fine/coarse is each datum

Scope -- how (in)complete is the data

Temporality -- how is the data situated in time

Faithfulness -- how well does the data capture “reality”

Variables Are Columns

What does each **column** represent?

A **variable** is a **measurement** of a particular concept.

	Year	Candidate	Party	Popular vote	Result	%
0	1824	Andrew Jackson	Democratic-Republican	151271	loss	57.210122
1	1824	John Quincy Adams	Democratic-Republican	113142	win	42.789878
2	1828	Andrew Jackson	Democratic	642806	win	56.203927
3	1828	John Quincy Adams	National Republican	500897	loss	43.796073
4	1832	Andrew Jackson	Democratic	702735	win	54.574789
...	...	...	...	...	...	...
177	2016	Jill Stein	Green	1457226	loss	1.073699
178	2020	Joseph Biden	Democratic	81268924	win	51.311515
179	2020	Donald Trump	Republican	74216154	loss	46.858542
180	2020	Jo Jorgensen	Libertarian	1865724	loss	1.177979
181	2020	Howard Hawkins	Green	405035	loss	0.255731

A **row** represents one record (i.e. an observation)

A **column** represents some characteristic, or feature, of that observation (here, the political party of that person).

It has two common properties:

- **Datatype/Storage type**:
  How each variable value is stored in memory. `df[colname].dtype`
  - integer, floating point, boolean, object (string-like), etc.

  Affects which pandas functions you use.

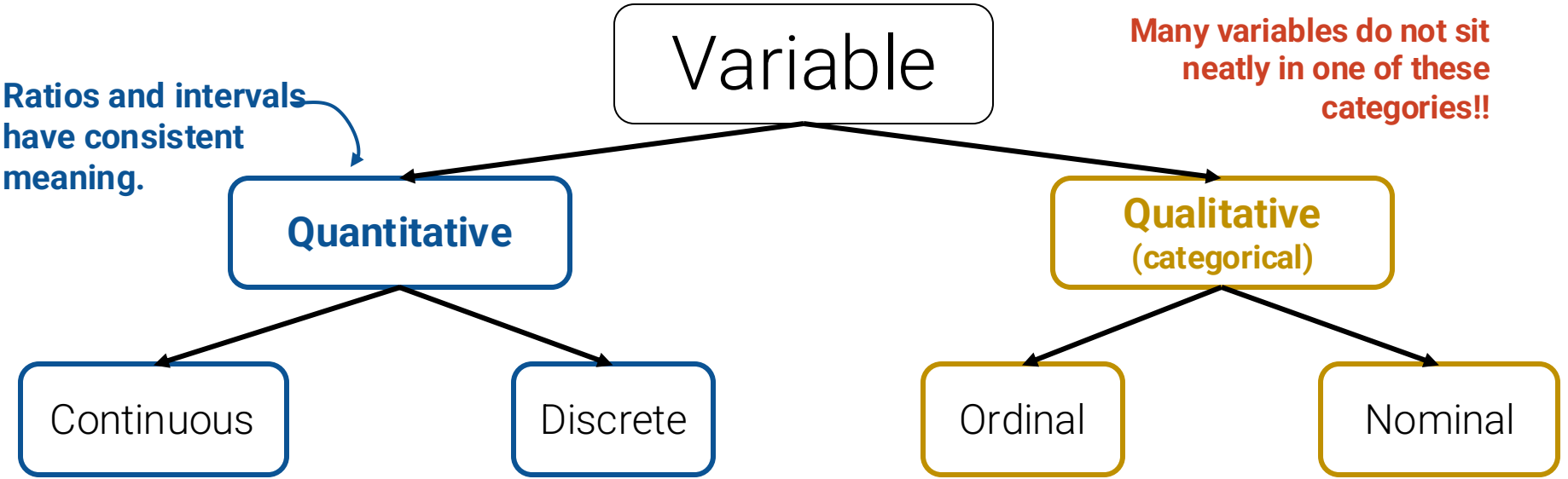
- **Variable type/Feature type**:
  Conceptualized measurement of information (and therefore what values it can take on).
  - Use expert knowledge
  - Explore data itself
  - Consult data codebook (if it exists).

  Affects how you ***visualize and interpret*** the data.

Storage types say what operations we can write **code to *compute***, while **feature types** say **what operations *make sense*** *for the data*.

⚠ In this class, "variable types" are conceptual!!

Variable Feature Types



Could be measured to arbitrary precision.

Examples:
- Price
- Temperature

Stored as integers where intervals have consistent meaning

Examples:
- Number of siblings
- Yrs of education

Categories w/ordered levels; no consistent meaning to difference

Examples:
- Preferences
- Level of education

Categories w/ no specific ordering.

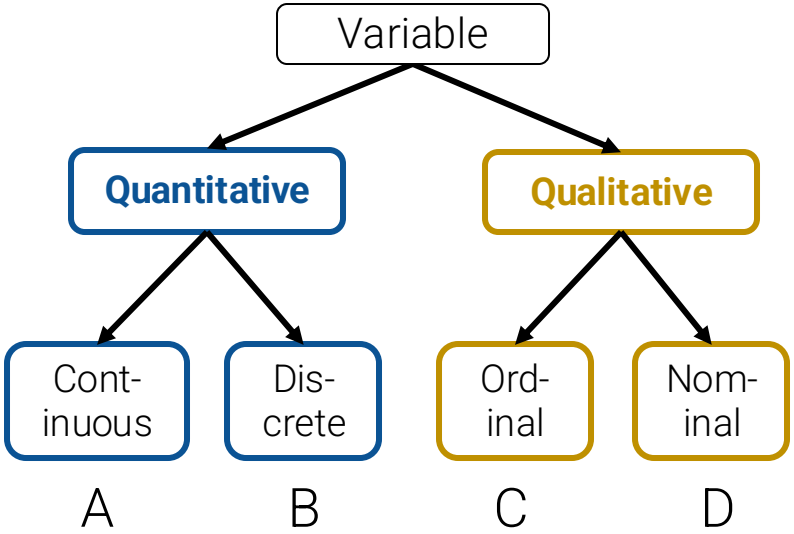
Examples:
- Political Affiliation
- CU ID number

Note that **qualitative variables** could have numeric levels; conversely, **quantitative variables** could be stored as strings!

Class Exercise

What is the feature type of each variable?

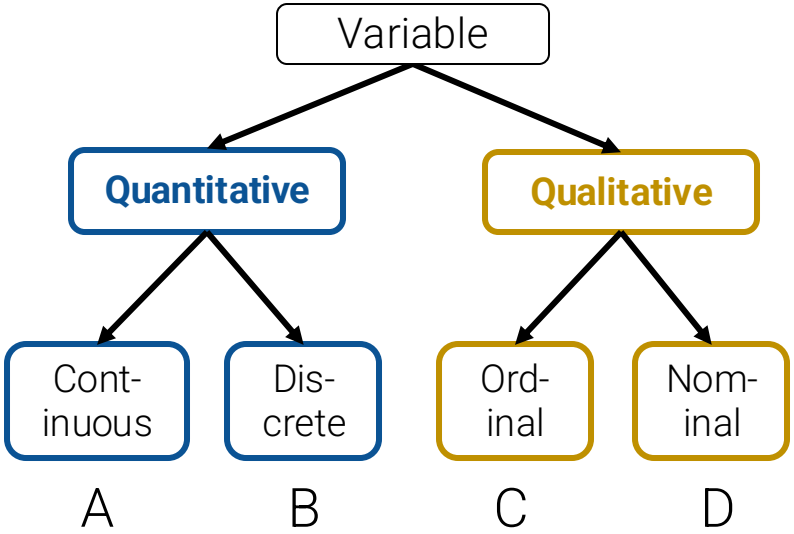
Q	Variable	Feature Type
1	CO_2 level (PPM)	
2	Number of siblings	
3	GPA	
4	Income bracket (low, med, high)	
5	Race	
6	Number of years of education	
7	Yelp Rating	



Class Exercise: Solutions

What is the feature type of each variable?

Q	Variable	Feature Type
1	CO_2 level (PPM)	**A. Quantitative Cont.**
2	Number of siblings	**B. Quantitative Discrete**
3	GPA	**A. Quantitative Cont.**
4	Income bracket (low, med, high)	**C. Qualitative Ordinal**
5	Race	**D. Qualitative Nominal**
6	Number of years of education	**B. Quantitative Discrete***
7	Yelp Rating	**C. Qualitative Ordinal***



*see speaker notes

Meta: For this exercise, The Feature Type variable is Qualitative Nominal.

Key Data Properties to Consider in EDA

Structure -- the “shape” of a data file

Granularity -- how fine/coarse is each datum

Scope -- how (in)complete is the data

Temporality -- how is the data situated in time

Faithfulness -- how well does the data capture “reality”

Primary Keys

Primary key: the column or set of columns in a table that *uniquely* determine the values in the remaining columns

- Primary keys are unique, but could be tuples.
- Examples: SSN, ProductIDs, …

Primary Key

Customers.csv

CustID	Addr
171345	Harmon..
281139	Main ..

Primary Key

Orders.csv

OrderNum	CustID	Date
1	171345	8/21/2017
2	281139	8/30/2017

Products.csv

ProdID	Cost
42	3.14
999	2.72

Primary Key

Purchases.csv

OrderNum	ProdID	Quantity
1	42	3
1	999	2
2	42	1

Primary Keys & Granularity

Primary key: the column or set of columns in a table that *uniquely* determine the values in the remaining columns

- Primary keys are unique, but could be tuples.
- Examples: SSN, ProductIDs, …

Granularity is the **concept** the primary key represents.

Example:

Granularity of Customer's table: Each row represents data for one unique customer.

Primary Key

Customers.csv

CustID	Addr
171345	Harmon..
281139	Main ..

Primary Key

Orders.csv

OrderNum	CustID	Date
1	171345	8/21/2017
2	281139	8/30/2017

Products.csv

ProdID	Cost
42	3.14
999	2.72

Primary Key

Purchases.csv

OrderNum	ProdID	Quantity
1	42	3
1	999	2
2	42	1

Granularity

What does each **record (row)** represent?

- Examples: a purchase, a person, a group of users, a house, a team

Do all records capture granularity at the same level?

- Some data will include summaries (aka **rollups**) as records

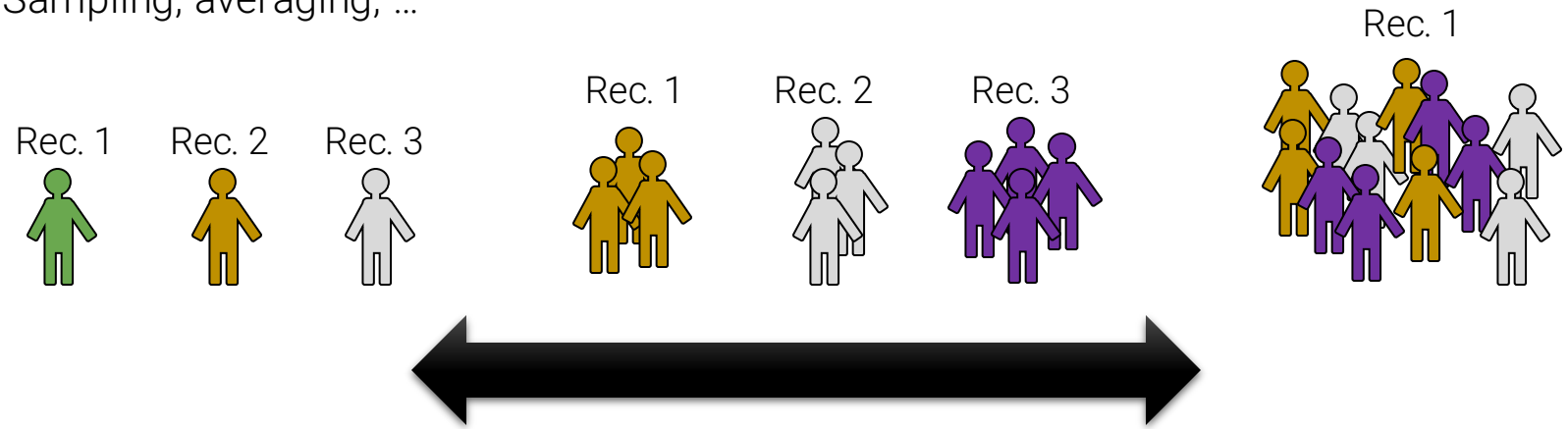
How were the records aggregated?

- Sampling, averaging, …

	Year	Candidate	Party	Popular vote	Result	%
0	1824	Andrew Jackson	Democratic-Republican	151271	loss	57.210122
1	1824	John Quincy Adams	Democratic-Republican	113142	win	42.789878
2	1828	Andrew Jackson	Democratic	642806	win	56.203927
3	1828	John Quincy Adams	National Republican	500897	loss	43.796073
4	1832	Andrew Jackson	Democratic	702735	win	54.574789
...	...	...	...	...	...	...
177	2016	Jill Stein	Green	1457226	loss	1.073699
178	2020	Joseph Biden	Democratic	81268924	win	51.311515
179	2020	Donald Trump	Republican	74216154	loss	46.858542
180	2020	Jo Jorgensen	Libertarian	1865724	loss	1.177979
181	2020	Howard Hawkins	Green	405035	loss	0.255731

A **row** represents one record (i.e. an observation)

A **column** represents some characteristic, or feature, of that observation (here, the political party of that person).



iClicker Poll:

```
elections.head()
```

	Year	Candidate	Party	Popular vote	Result	%
0	2024	Kamala Harris	Democratic	75019230	loss	48.34
1	2024	Donald Trump	Republican	77303568	win	49.81
2	2024	Jill Stein	Green	861155	loss	0.60
3	2024	Robert F. Kennedy Jr.	Independent	756383	loss	0.60
4	2024	Chase Oliver	Libertarian	650130	loss	0.40

How could we determine the granularity of the whole dataset?

Based on the first 5 rows of the DataFrame, what appears to be the granularity of the election dataset?

(i.e. each record represents data about a ….)

A). Presidential Candidate
B). Political Party
C). Political Party in a Specific Year
D). Presidential Candidate in a Specific Year
E). Candidate in a Political Party

iClicker Poll:

```
elections.head()
```

	Year	Candidate	Party	Popular vote	Result	%
0	2024	Kamala Harris	Democratic	75019230	loss	48.34
1	2024	Donald Trump	Republican	77303568	win	49.81
2	2024	Jill Stein	Green	861155	loss	0.60
3	2024	Robert F. Kennedy Jr.	Independent	756383	loss	0.60
4	2024	Chase Oliver	Libertarian	650130	loss	0.40

What Pandas functions do we need to use to determine the granularity of the whole dataset?

- Need to select a subset of column(s) that we think represent granularity
- Need to a way to determine uniqueness of entries in multiple columns

Based on the first 5 rows of the DataFrame, what appears to be the granularity of the election dataset?

(i.e. each record represents data about a ….)

A). Presidential Candidate
B). Political Party
C). Political Party in a Specific Year
D). Presidential Candidate in a specific year
E). Candidate in a Political Party

Key Data Properties to Consider in EDA

Structure -- the “shape” of a data file

Granularity -- how fine/coarse is each datum

Scope -- how (in)complete is the data

Temporality -- how is the data situated in time

Faithfulness -- how well does the data capture “reality”

Scope

Will my data be enough to answer my question?

- **Example**: I am interested in studying crime in California but I only have Berkeley crime data.
- **Solution**: collect more data/change research question

Is my data too expansive?

- **Example**: I am interested in student grades for Data100 but have student grades for all Data Science classes.
- **Solution: Filtering** ⇒ Implications on sample?
  - If the data is a sample I may have poor coverage after filtering (More on this next week)

Does my data cover the right time frame?

- Which brings us to **Temporality**

"Scope" questions are defined by your question/problem and inform if you need better-scoped data.

Temporality

Data changes – when was the data collected/last updated?

Periodicity — Is there periodicity? Diurnal (24-hr) patterns?

What is the meaning of the time and date fields? A few options:

- When the “event” happened?
- When the data was collected or was entered into the system?
- Date the data was copied into a database? (look for many matching timestamps)

Time depends on where! (**time zones** & daylight savings)

- Learn to use `datetime` Python library and Pandas `dt` accessors
- Regions have different datestring representations: 07/08/09?

Are there strange null values?

- E.g., **January 1st 1970**, January 1st 1900…?

Temporality: Unix / POSIX Time

Time measured in seconds since **January 1st 1970 UTC**

- Minus leap seconds …

UTC is Coordinated Universal Time

- International time standard
- Measured at 0 degrees latitude
  - Similar to Greenwich Mean Time (GMT)
- No daylight savings

Time Zones:

- San Francisco (**UTC-7**) with daylight savings

Jun 27, 2023 5:00pm PDT
1687910400

https://en.wikipedia.org/wiki/Coordinated_Universal_Time

What else?

Structure -- the “shape” of a data file

Granularity -- how fine/coarse is each datum

Scope -- how (in)complete is the data

Temporality -- how is the data situated in time

Faithfulness -- how well does the data capture “reality”

Faithfulness: Do I trust this data?

Does my data contain **unrealistic or "incorrect" values**?

- Dates in the future for events in the past
- Locations that don't exist
- Negative counts
- Misspellings of names
- Large outliers

Does my data violate **obvious dependencies**?

- E.g., age and birthday don't match

Was the data **entered by hand**?

- Spelling errors, fields shifted …
- Did the form require all fields or provide default values?

Are there obvious signs of **data falsification**?

- Repeated names, fake looking email addresses, repeated use of uncommon names or fields.

Signs that your data may not be faithful (and proposed solutions)

Truncated data
Early Microsoft Excel limits: 65536 Rows, 255 Columns

Duplicated Records or Fields
Identify and eliminate (use primary key).

Spelling Errors
Apply corrections or drop records not in a dictionary

Units not specified or consistent
Infer units, check values are in reasonable ranges for data

Time Zone Inconsistencies
Convert to a common timezone (e.g., UTC)

- Be aware of consequences in analysis when using data with inconsistencies.
- Understand the potential implications for how data were collected.

Missing Data???

Examples

" "	1970, 1900
0, -1	NaN
999, 12345	Null

NaN: "Not a Number"

Missing Data/Default Values: Solutions

A. Drop records with missing values

- **Caution**: check for biases induced by dropped values
  - When modeling data you can't drop missing values in your test/validation sets
  - Missing or corrupt records might be related to something of interest

B. Keep as NaN

C. Imputation/Interpolation: Inferring missing values

- **Average Imputation**: replace with the mean
  - When the variable's distribution is roughly bell shaped, or mostly crowded at the mean
- **Median Imputation**: replace with the median
  - When the variable's distribution is skewed (not crowded at the mean, rather at the left or right of it)
- **Mode Imputation:** replace with the mode
  - When the variable is frequently going to be a specific value, there is a high chance that the mode is the right inherent value for such data
- **Zero:** Replace with zero if it makes sense in the context of the data

Choice affects bias and uncertainty quantification (large statistics literature)

Essential question: why are the records missing?

Other Imputation Strategies (out of scope for this class)

- **Hot deck imputation**: replace with a random value
- **Regression imputation**: replace with a predicted value, using some model