Data Wrangling and EDA, Part I

Exploratory Data Analysis and its role in the data science lifecycle

- Introduced to the DataFrame concept

  Series: The column of data with “Label”

  DataFrame: The collection of series with the same indices

- DataFrame access methods

  Filtering: slicing with boolean conditions

  df. loc: location by index or labels

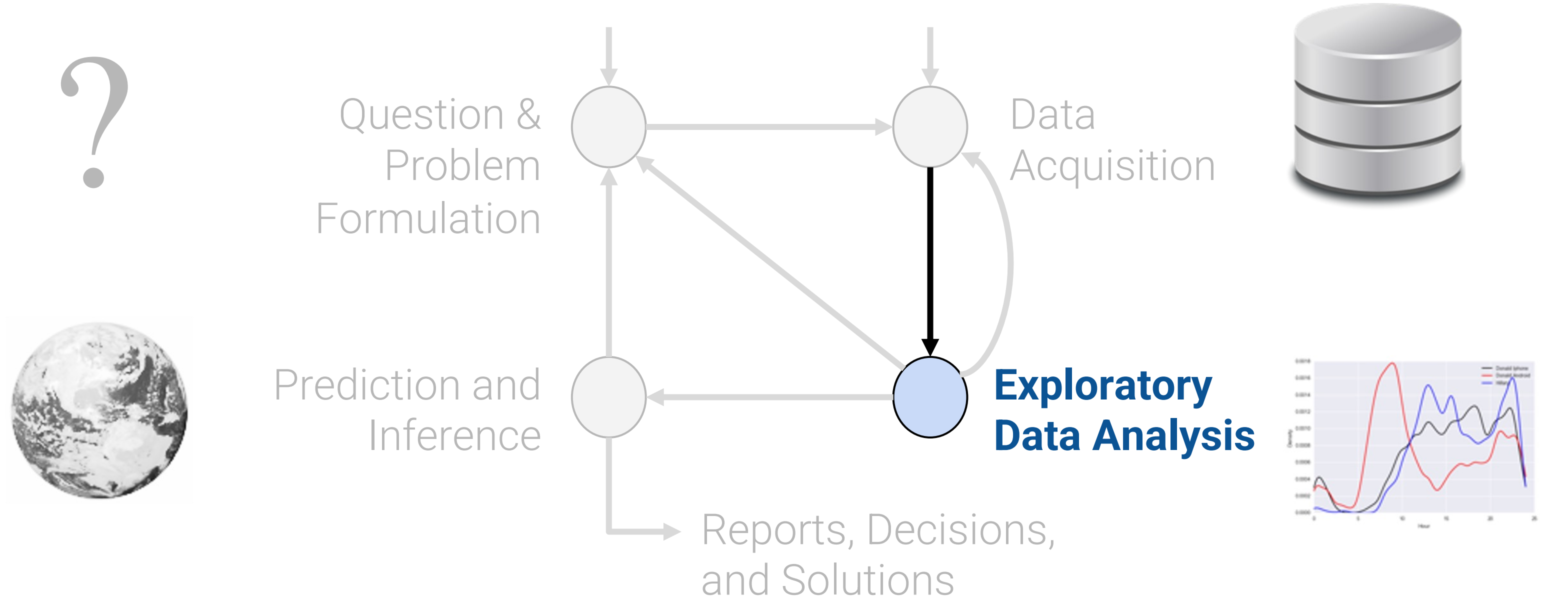
  df. iloc: location by integer index

  Groupby and pivot: for data aggregation

The Next Step

EDA Guiding Principles

Plan for First Few Weeks



(Weeks 1-2)

Exploring and Cleaning Tabular Data
From `datascience` to `pandas`

(Week 3)

Data Science in Practice
EDA, Data Cleaning, Text processing (regular expressions)

Structure: Tabular Data

Lecture 06

- Pandas, Part III
  - Groupby Review
  - More on Groupby
  - Pivot Tables
  - Joining Tables
- **EDA, Part I**
  - **Structure: Tabular Data**
  - Granularity
  - Structure: Variable Types

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

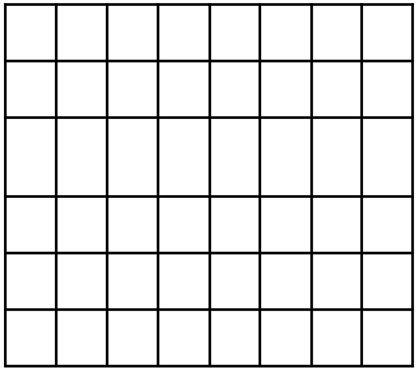
Faithfulness -- how well does the data capture "reality"

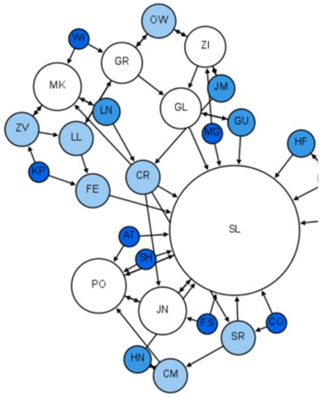
Rectangular and Non-rectangular Data

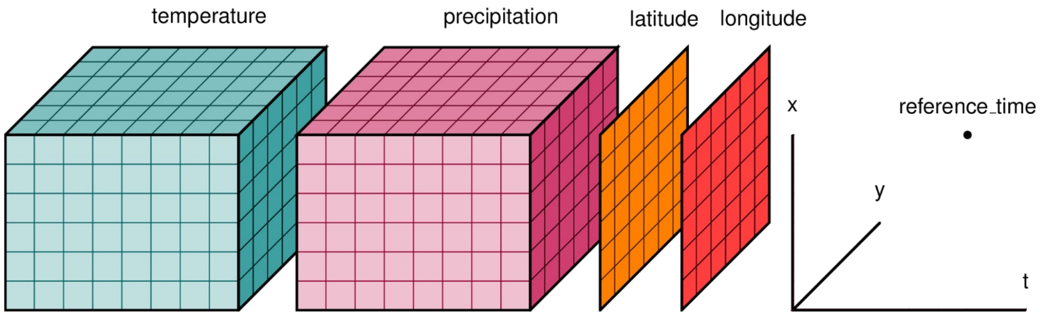
Data come in many different shapes.

Non-rectangular data

Rectangular data





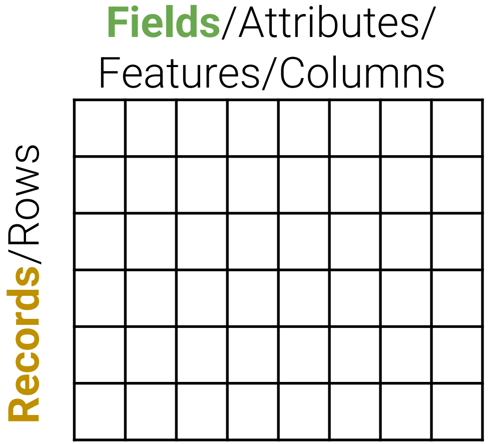


Rectangular Data

We often prefer rectangular data for data analysis (why?)

- Regular structures are easy manipulate and analyze
- A big part of data cleaning is about transforming data to be more rectangular

Two kinds of rectangular data: **Tables** and **Matrices**.



Tables (a.k.a. `DataFrames` in R/Python and relations in SQL)

- Named columns with different types
- Manipulated using data transformation languages (map, filter, group by, join, …)

Matrices

- Numeric data of the same type (float, int, etc.)
- Manipulated using linear algebra

Tuberculosis – United States, 2021

CDC Morbidity and Mortality Weekly Report (MMWR) 03/25/2022.

Summary

What is already known about this topic?

The number of reported U.S. tuberculosis (TB) cases decreased sharply in 2020, possibly related to multiple factors associated with the COVID-19 pandemic.

What is added by this report?

Reported TB incidence (cases per 100,000 persons) increased 9.4%, from 2.2 during 2020 to 2.4 during 2021 but was lower than incidence during 2019 (2.7). Increases occurred among both U.S.-born and non–U.S.-born persons.

What are the implications for public health practice?

Factors contributing to changes in reported TB during 2020–2021 likely include an actual reduction in TB incidence as well as delayed or missed TB diagnoses. Timely evaluation and treatment of TB and latent tuberculosis infection remain critical to achieving U.S. TB elimination.

What is **incidence**?
Why use it here?

How was "9.4% increase" computed?

Question: Can we **reproduce** these rates using government data?

Demo Slides

CSV: Comma-Separated Values

Tuberculosis in the US [CDC source].

CSV is a very common **tabular file format**.

- **Records** (rows) are delimited by a newline: `'\n'`, `"\r\n"`
- **Fields** (columns) are delimited by commas: `','`

Pandas: `pd.read_csv(header=...)`

Fields/Attributes/Features/Columns

Records/Rows

	U.S. jurisdiction	TB cases 2019	...
0	Total	8,900	...
1	Alabama	87	...

Granularity

Lecture 06

- Pandas, Part III
  - Groupby Review
  - More on Groupby
  - Pivot Tables
  - Joining Tables
- **EDA, Part I**
  - Structure: Tabular Data
  - **Granularity**
  - Structure: Variable Types

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”

Granularity: How Fine/Coarse Is Each Datum?

What does each **record** represent?

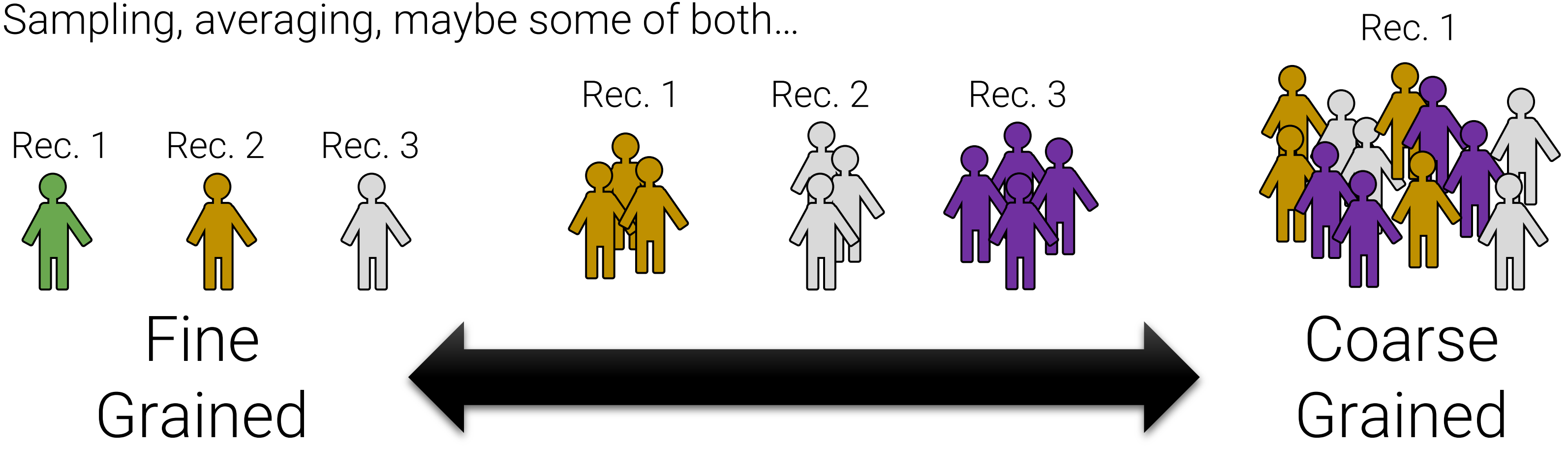
- Examples: a purchase, a person, a group of users

Do all records capture granularity at the same level?

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

If the data are **coarse**, how were the records aggregated?

- Sampling, averaging, maybe some of both…



Structure: Variable Types

Lecture 06

(we're back to this)

Variable Type

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

Let's look at records with the same granularity.

What does each **column** represent?

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

The U.S. Jurisdiction **variable**

	U.S. jurisdiction	TB cases 2019	...
1	Alabama	87	...
2	Alaska	58	...
...	...	...	...

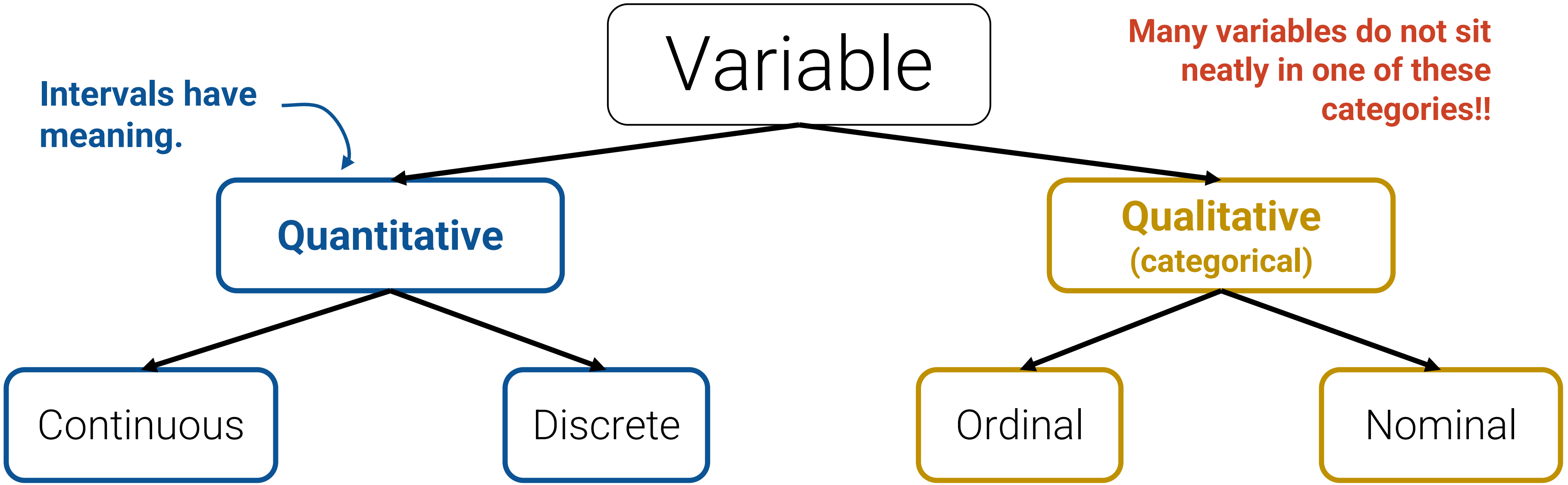
It has two common properties:

- **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.

Variable Feature Types



Continuous: Could be measured to arbitrary precision.

Examples:
- Price
- Temperature

Discrete: Finite possible values

Examples:
- Number of siblings
- Yrs of education

Ordinal: Categories w/ordered levels; no consistent meaning to difference

Examples:
- Preferences
- Level of education

Nominal: Categories w/ no specific ordering.

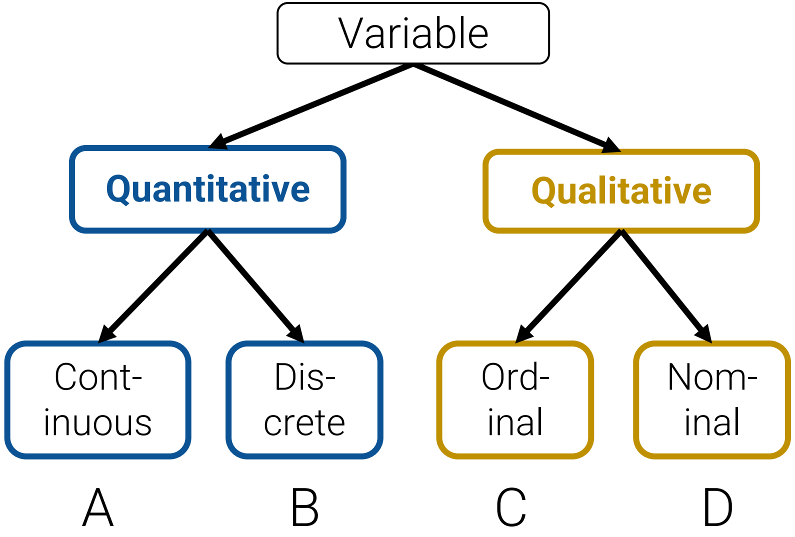
Examples:
- Political Affiliation
- Cal ID number

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

Variable Types

What is the feature type (i.e., variable type) of each variable?

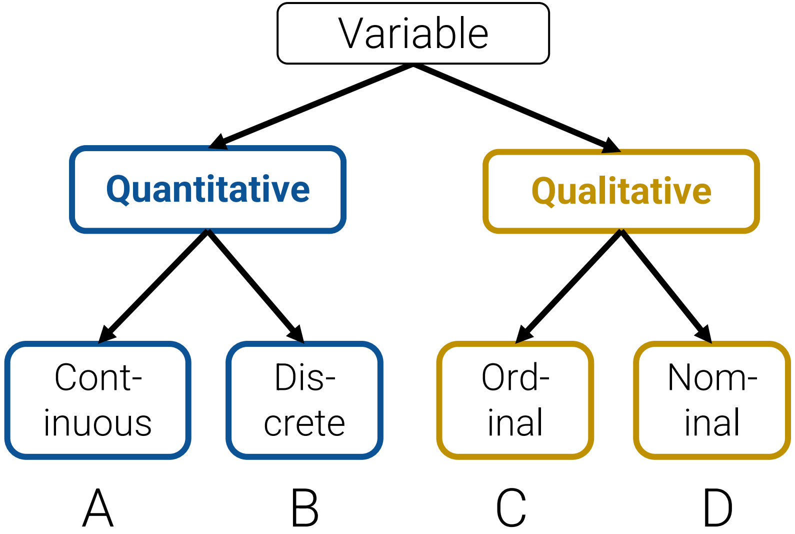
Q	Variable	Feature Type
1	CO_2 level (ppm)	
2	Number of siblings	
3	GPA	
4	Income bracket (low, med, high)	
5	Race/Ethnicity	
6	Number of years of education	
7	Yelp Rating	



Variable Types

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/Ethnicity	**D. Qualitative Nominal**
6	Number of years of education	**B. Quantitative Discrete**
7	Yelp Rating	**C. Qualitative Ordinal**



Many of these examples show how "shaggy" these categories are!!
We will revisit variable types when we learn how to visualize variables.

LECTURE 6

Data Wrangling and EDA, Part II

Exploratory Data Analysis and its role in the data science lifecycle.

Data Science Fall 2023 @ Knowledge Stream

Structure

- Multiple Files
- More File Formats

Scope and Temporality

Faithfulness (and Missing Values)

- Demo: Mauna Loa CO2

Today's Roadmap

Lecture 6

Multiple Files

Lecture 6

Structure

- **Multiple Files**
- More File Formats

Scope and Temporality

Faithfulness (and Missing Values)

- Demo: Mauna Loa CO2

File Format

Variable Type

Multiple files
(Primary and Foreign Keys)

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”

What is incidence?

Summary

What is already known about this topic?

The number of reported U.S. tuberculosis (TB) cases decreased sharply in 2020, possibly related to multiple factors associated with the COVID-19 pandemic.

What is added by this report?

Reported TB incidence (cases per 100,000 persons) increased 9.4%, from 2.2 during 2020 to 2.4 during 2021 but was lower than incidence during 2019 (2.7). Increases occurred among both U.S.-born and non–U.S.-born persons.

What are the implications for public health practice?

Factors contributing to changes in reported TB during 2020–2021 likely include an actual reduction in TB incidence as well as delayed or missed TB diagnoses. Timely evaluation and treatment of TB and latent tuberculosis infection remain critical to achieving U.S. TB elimination.

CDC Morbidity and Mortality Weekly Report (MMWR) 03/25/2022.

What is **incidence**?
Why use it here?

How was "9.4% increase" computed?

Question: Can we **reproduce** these rates using government data?

Defining incidence

From the CDC report: **TB incidence** is computed as the number of “cases per 100,000 persons using mid-year population estimates from the U.S. Census Bureau.”

- Incidence is useful when comparing case rates across differently sized populations.

$$\textbf{TB incidence} = \frac{\text{\# TB cases in population}}{\text{\# groups in population}}$$

(group: 100,000 people)

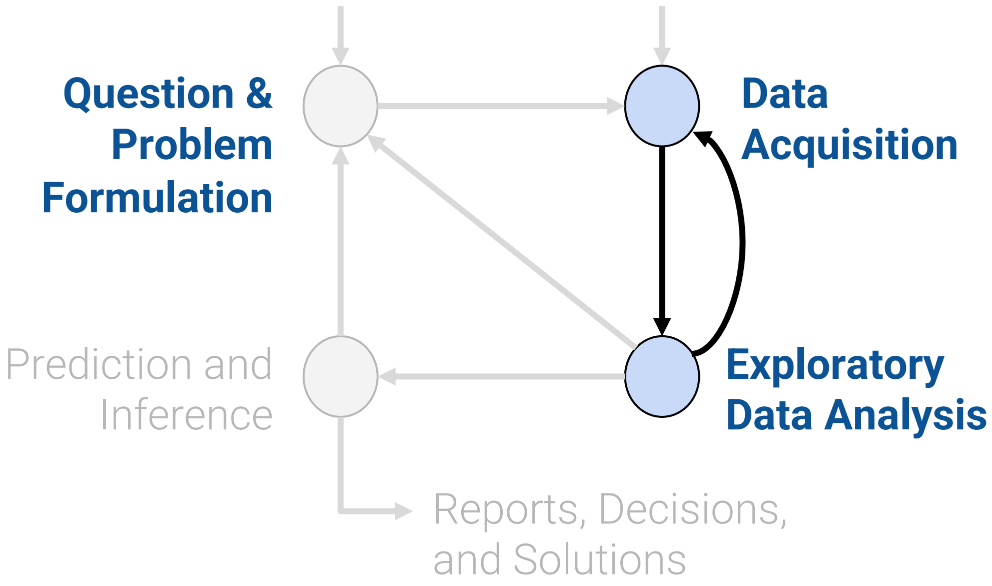
$$= \frac{\text{\# TB cases}}{\text{(population/100,000)}}$$

$$= \frac{\text{\# TB cases}}{\text{population}} \times 100{,}000$$

We don’t have U.S. Census population data in our DataFrame.
We need to acquire it to verify incidence!

The Data Science Lifecycle is a Cycle

In practice, EDA informs whether you need more data to address your research question.



Structure: Primary Keys and Foreign Keys

Sometimes your data comes in multiple files:

- Often data will reference other pieces of data.
- Alternatively, you will collect multiple pieces of related data.

Use `pd.merge` to **join** data on **keys**.

Customers.csv

CustID	Addr
171345	Harmon..
281139	Main ..

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

Purchases.csv

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

Structure: Primary Keys and Foreign Keys

Sometimes your data comes in multiple files:

- Often data will reference other pieces of data.
- Alternatively, you will collect multiple pieces of related data.

Use `pd.merge` to join data on **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: OrderNum, 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

Structure: Primary Keys and Foreign Keys

Sometimes your data comes in multiple files:

- Often data will reference other pieces of data.
- Alternatively, you will collect multiple pieces of related data.

Use `pd.merge` to join data on **keys**.

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

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

Foreign keys: the column or sets of columns that reference primary keys in other tables.

Primary Key

Customers.csv

CustID	Addr
171345	Harmon..
281139	Main ..

Foreign 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

Purchases.csv

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

More Files Formats

Lecture 6

Structure

- Multiple Files
- **More File Formats**

Scope and Temporality

Faithfulness (and Missing Values)

- Demo: Mauna Loa CO2

Demo Slides

TSV: Tab Separated Values

Another common table file format.

- Records are delimited by a newline: `'\n'`, `"\r\n"`
- **Fields** are delimited by `'\t'` (tab)

`pd.read_csv`: Need to specify
`delimiter='\t'`

Issues with CSVs and TSVs:

- Commas, tabs in records (use `quotechar` parameter)
- Quoting
- …

Demo Slides

JSON: JavaScript Object Notation

A less common file format.

- Very similar to Python dictionaries
- Strict formatting "quoting" addresses some issues in CSV/TSV
- **Self-documenting**: Can save metadata (data about the data) along with records in the same file

`To reads JSON file:`
`pd.read_json()` function, which works for most simple JSON files.

We will dive deeper into exactly how a JSON can structured.

Demo Slides

JSON: JavaScript Object Notation

Berkeley covid cases by day (City of Berkeley)

A less common file format.

- Very similar to Python dictionaries
- Strict formatting "quoting" addresses some issues in CSV/TSV
- **Self-documenting**: Can save metadata (data about the data) along with records in the same file

Issues

- Not rectangular
- Each record can have different fields
- Nesting means records can contain tables – complicated

Reading a JSON into pandas often requires some EDA.

Are the data in a standard format or encoding?

- Tabular data: CSV, TSV, Excel, SQL
- Nested data: JSON or XML

Are the data organized in **records** or nested?

- Can we define records by parsing the data?
- Can we reasonably un-nest the data?

Does the data reference other data?

- Can we join/merge the data?
- Do we need to?

What are the **fields** in each record?

- How are they encoded? (e.g., strings, numbers, binary, dates …)
- What is the type of the data?

Structure -- the "shape" of a data file

Granularity -- how fine/coarse is each datum

Scope -- how (in)complete is the data

Summary

You will do the most data wrangling when analyzing the structure of your data.

Scope and Temporality

Lecture 6

Structure

- Multiple Files
- More File Formats

Scope and Temporality

Faithfulness (and Missing Values)

- Demo: Mauna Loa CO2

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 San Francisco crime data.
- **Solution**: collect more data/change research question

Is my data too expansive?

- **Example**: I am interested in student grades for DataScience but have student grades for all classes.
- **Solution: Filtering** ⇒ Implications on sample?
  - If the data is a sample I may have poor coverage after filtering

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…?

Faithfulness (and Missing Values)

Lecture 6

Structure

- Multiple Files
- More File Formats
- Scope and Temporality

Faithfulness (and Missing Values)

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).

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

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

Missing Data???

Examples

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

NaN: “Not a Number”

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

Missing Data/Default Values: Solutions

A. Drop records with missing values

- Probably most common
- **Caution**: check for biases induced by dropped values
    - 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 an average value
    - Which average? Often use closest related subgroup mean.
- **Hot deck imputation**: replace with a random value
- **Regression imputation**: replace with a predicted value, using some model
- **Multiple imputation**: replace with multiple random values.

Missing Data/Default Values: Solutions

A. Drop records with missing values

- Probably most common
- **Caution**: check for biases induced by dropped values
  - 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 an average value
  - Which average? Often use closest related subgroup mean.
- **Hot deck imputation**: replace with a random value
- **Regression imputation**: replace with a predicted value, using some model
- **Multiple imputation**: replace with multiple random values.

(beyond this course)

Choice affects bias and uncertainty quantification (large statistics literature)

Essential question: why are the records missing?

Summary: How do you do EDA/Data Wrangling?

Examine **data and metadata**:

- What is the date, size, organization, and structure of the data?

Examine each **field/attribute/dimension** individually

Examine **pairs of related dimensions**

- Stratifying earlier analysis: break down grades by major …

Along the way:

- **Visualize**/summarize the data
- **Validate assumptions** about data and collection process. Pay particular attention to when data were collected.
- Identify and **address anomalies**
- Apply data transformations and corrections (next lecture)
- **Record everything you do**! (why?)
  - Developing in Jupyter Notebooks promotes reproducibility of your own work.