Working with **Data**

Vinayaka Gude, Ph.D.
Elon University

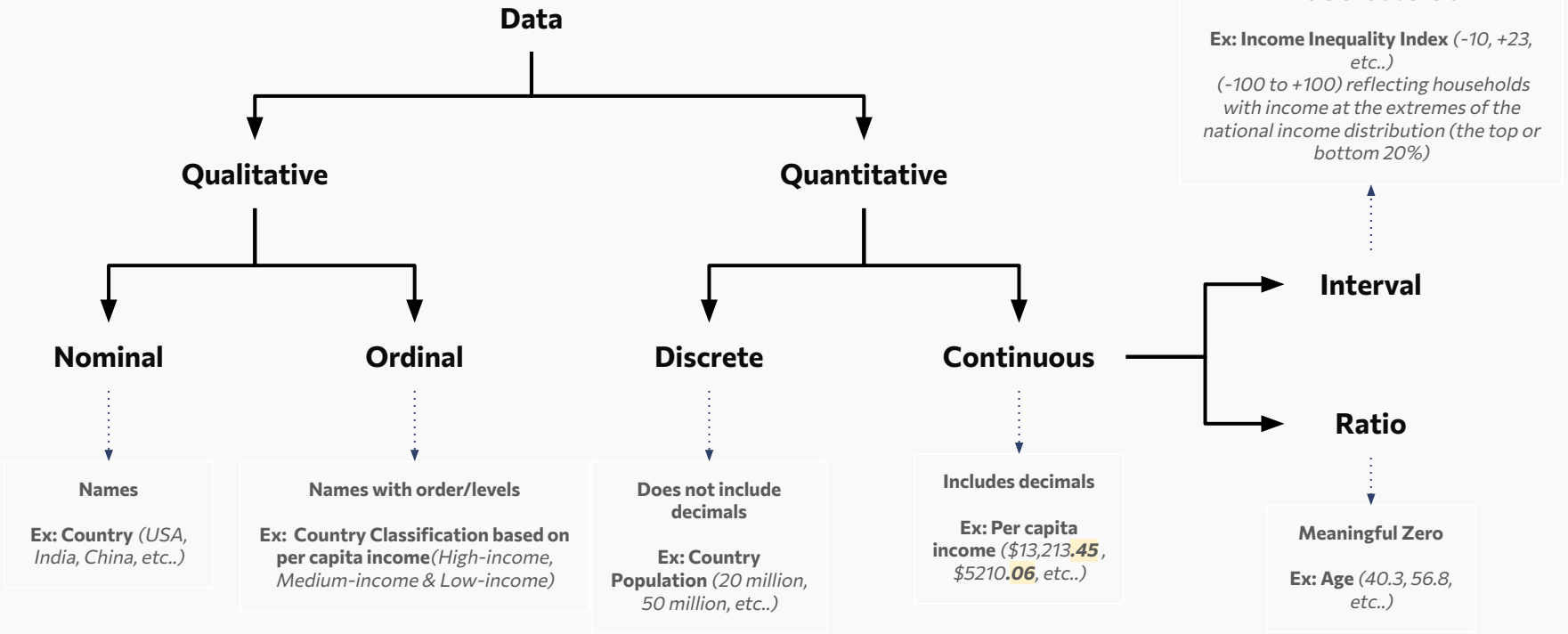
Agenda

01 Coding demo: Data cleaning using Pandas

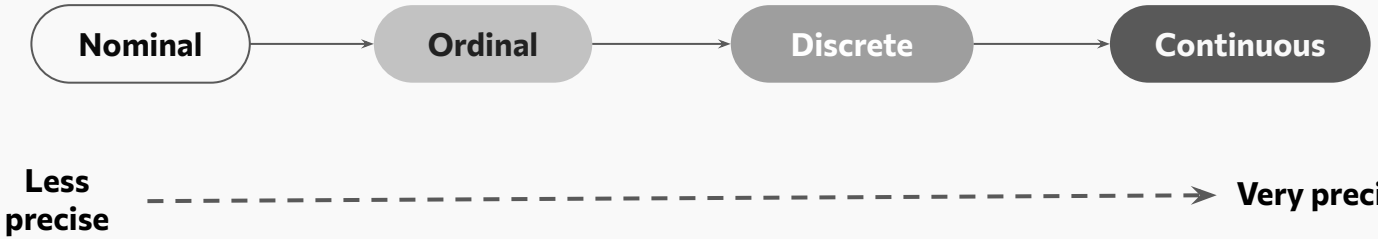
02 Data Collection

03 Project Introduction

Data Types


Data
Qualitative
Quantitative
Nominal
Ordinal
Discrete
Continuous
Interval
Ratio
Names
Ex: Country (USA, India, China, etc..)
Names with order/levels
Ex: Country Classification based on per capita income(High-income, Medium-income & Low-income)
Does not include decimals
Ex: Country Population (20 million, 50 million, etc..)
Includes decimals
Ex: Per capita income ($13,213.45 , $5210.06, etc..)
There is no true zero
Ex: Income Inequality Index (-10, +23, etc..)
(-100 to +100) reflecting households with income at the extremes of the national income distribution (the top or bottom 20%)
Meaningful Zero
Ex: Age (40.3, 56.8, etc..)

Data Types & Measurements



Personal Well being?

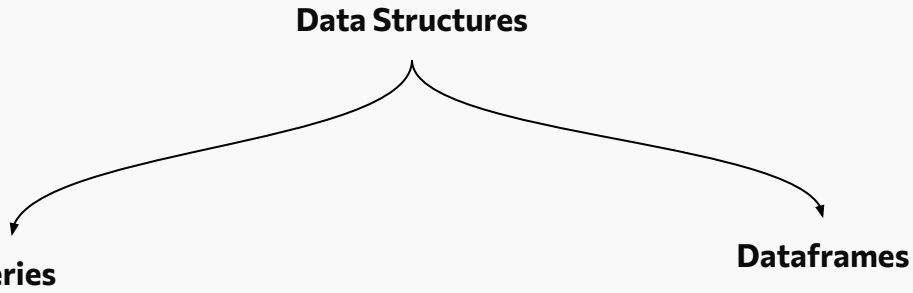
Let’s Code!

Introduction to **Data Structures**

Pandas Introduction

If not installed: `pip install pandas`

Once installed, call the library using: `import pandas as pd`



One-dimensional array-like object containing a sequence of values of the **same type** and an associated array of data labels, called its index.

Rectangular table of data and contains an ordered, named collection of columns, each of which can be a different value type (numeric, string, Boolean, etc.). The DataFrame has both a row and column index.

Pandas: Series

Create a series: `obj = pd.Series([4, 7, -5, 3])`

Create a series from dictionary:

```
sdata = {"Ohio": 35000, "Texas": 71000, 'Oregon': 16000, 'Utah': 5000}
obj2 = pd.Series(sdata)
states = ["California", "Ohio", "Oregon", "Texas"]
obj3 = pd.Series(sdata, index=states)
```

Use a specific index: `obj4 = pd.Series([4, 7, -5, 3], index=["d", "b", "a", "c"])`

Retrieve values from a series: `obj4["a"]` or `obj4[["c", "a", "d"]]`

Filter a series: `obj4[obj4 > 0]`

Array Operations: `obj2 * 2` or `obj4[["c", "a", "d"]]`

Check indexes: `"b" in obj4`

Check missing data: `obj4.isna()`

Pandas: DataFrame

Create a Dataframe:

```
data = {"state": ["Ohio", "Ohio", "Ohio", "Nevada", "Nevada", "Nevada"],
"year": [2000, 2001, 2002, 2001, 2002, 2003],
"pop": [1.5, 1.7, 3.6, 2.4, 2.9, 3.2]}
frame = pd.DataFrame(data)
```

View the Dataframe: `frame.head()` or `frame.tail()`

Retrieve columns: `frame["state"]` or `frame.year`

Setting index: `frame.set_index('state')`

Add index: `frame2 = frame.reindex(index=["a", "b", "c", "d])`

Dropping rows: `new_obj = frame2.drop("c")`

Python Libraries for Data Science

Data Analysis libraries

NumPy

SciPy

Pandas

SciKit-Learn

Visualization libraries

matplotlib

Seaborn

and many more ...

How to deal with missing values?

Delete rows

When a few rows have a lot of missing values

Delete columns

When a few columns have a lot of missing values

Replace with “0”

To avoid losing data, with less number of obs.

Replace with mean

To avoid losing data, ideal with less number of obs.
(Prefer this over replacing with 0)

Other approaches: **interpolation, clustering, etc..**

Missing Values with Pandas

Example DataFrame:

```
data = {"state": ["Ohio", "Ohio", "Ohio", "Nevada", "Nevada", "Nevada"],"year": [np.nan, 2001, np.nan, 2001, 2002, 2003],"pop": [1.5, 1.7, np.nan, 2.4, np.nan, 3.2]}
df = pd.DataFrame(data)
```

Delete columns: `df=df.dropna(axis=1, how='all')` or `df=df.dropna(axis=1, how='any')`

Delete rows: `df=df.dropna(axis=0, how='all')` or `df=df.dropna(axis=0, how='any')`

Impute a specific value: `df['pop']=df['pop'].fillna(0)`

Impute mean: `df['pop']=df['pop'].fillna(np.mean(df['pop']))`

Interpolate: `df["pop"] = df["pop"].interpolate(method="quadratic")`

Remove Duplicate rows: `df = df.drop_duplicates()`

Data Collection

Primary Sources
(First hand gathered data - involved)

- Indirect/Direct observations
- Interviews
- Surveys
- Experiments

Secondary Sources
(Collected by others - quick & easy)

- Government data sources
- Journals
- Public websites (need verification)

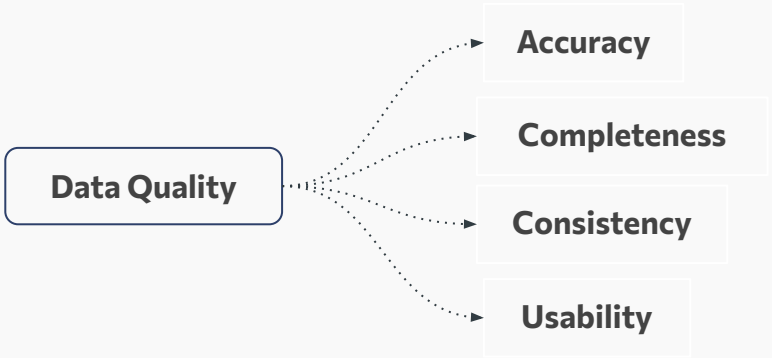
Public Data Sources

- UCI datasets
- Dataset source collection by topic
- Government datasets
- Google dataset Search
- IIIT datasets
- Data in Brief

Data Quality

Source Assessment

- Who collected or produced it?
- When was the data collected?
- What was their intent?
- Is it a reputable source?



Garbage IN - Garbage OUT

Group Project Task

Collect a dataset

- Verify Data Quality
- What decisions can you enable with this selected data?
- What are the limitations?

Thank you!

Any questions?

gude.vinayaka@outlook.com