

IST 5520: Data Science and Machine Learning with Python

3. Manage Data

Langtao Chen, Fall 2022
chenla@mst.edu

Reading

- ▶ Online Article: Statistics – Understanding the Levels of Measurement
 - <http://www.kdnuggets.com/2015/08/statistics-understanding-levels-measurement.html>
- ▶ 10 Minutes to Pandas
 - https://pandas.pydata.org/pandas-docs/stable/user_guide/10min.html
- ▶ Pandas Cheat Sheet
 - https://pandas.pydata.org/Pandas_Cheat_Sheet.pdf
- ▶ Regular Expression HOWTO
 - <https://docs.python.org/3/howto/regex.html>

Learning Objectives

- ▶ Understand dataset
- ▶ Be able to distinguish among different scales of measurement
- ▶ Be able to use pandas to collect, cleanse, and transform raw data
- ▶ Be able to use regular expression to process strings in Python
- ▶ Be able to use Pandas to manipulate dataset such as sub-setting, grouping, reshaping, merging data, making new variables, dealing with missing values, and plotting data
- ▶ Understand rescaling data and be able to use scikit-learn package to resale variables
- ▶ Be able to use Pandas to import flat files, connect to RDBMS
- ▶ Be able to use [requests](#) and [BeautifulSoup](#) modules to scrape data from the Internet

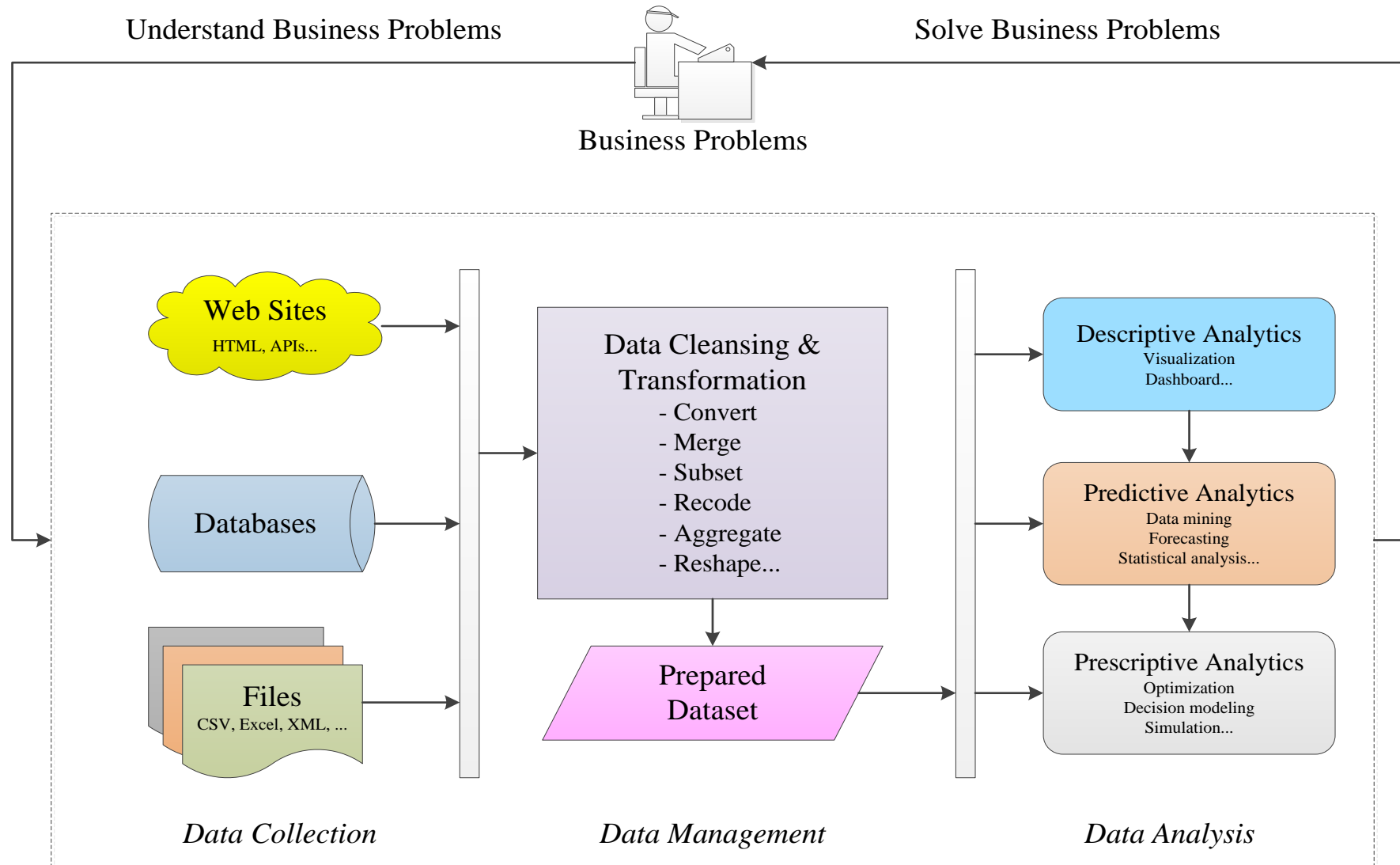
Prerequisite Knowledge of Data Formats

- ▶ This section assumes that you have the following working knowledge. If you are not familiar with some topics, please read the online materials specified in the links.
 - CSV file structure
 - ▶ https://en.wikipedia.org/wiki/Comma-separated_values
 - Relational database and SQL
 - ▶ SQL Tutorial at <http://www.w3schools.com/sql/default.asp>
 - HTML
 - ▶ HTML Tutorial at <http://www.w3schools.com/html/default.asp>
 - BeautifulSoup
 - ▶ BeautifulSoup Documentation at <https://www.crummy.com/software/BeautifulSoup/bs4/doc/>
 - XML
 - ▶ XML Tutorial at <http://www.w3schools.com/xml/default.asp>
 - XPath
 - ▶ XPath Tutorial at http://www.w3schools.com/xml/xpath_intro.asp
 - JSON
 - ▶ JSON Introduction at http://www.w3schools.com/js/js_json_intro.asp
 - Application programming interface (API)

AGENDA

- ▶ Data Management in Data Science
- ▶ Dataset and Scales of Measurement
- ▶ Pandas Data Structure
- ▶ Manipulate Data
 - Manipulate Strings
 - Manipulate Datasets
 - Rescale Data
- ▶ Import and Collect Data
 - Work with Flat Files and Relational Database
 - Collect Data from the Internet

Recap: Data Science Procedure



Why Data Management Matters?

- ▶ The raw data are usually messy:
 - Raw data may be unstructured, incomplete, inconsistent, and erroneous;
 - Thus we cannot directly apply raw data to statistical models.
- ▶ **Need to do “dirty” jobs:**
 - Collecting data from various sources
 - Cleansing data
 - Transforming data
 - Storing data in an appropriate repository
 - Aggregate data to a higher level
 -
- ▶ Pandas contains many useful string dataset manipulation operators

Unit of Analysis

- ▶ Unit of analysis is the major entity that is being analyzed in a BA project. For example:
 - In a customer churn project, the unit of analysis is at customer level
 - In a product price prediction project, the unit of analysis is at product level
- ▶ Unit of analysis guides the whole data management process:
 - All variables should be created and maintained at the specific unit of analysis.
 - These variables describe properties of a case of the unit of analysis.
- ▶ In case you have multiple units of analysis and they may be nested in some kind of hierarchy, you may need to split your original data into multiple datasets with each dataset having a clear structure.

Population vs. Sample

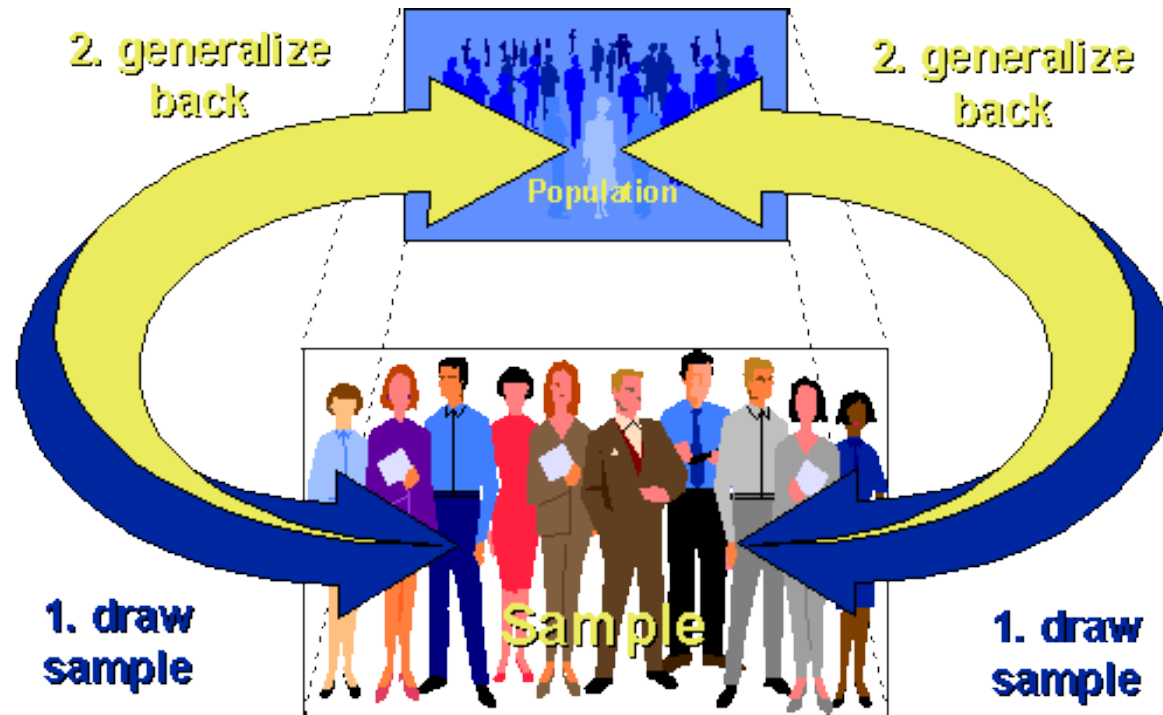
- ▶ Population: the whole body of subjects that are of interest to us in a particular study.
- ▶ Sample: a portion of the population from which we collect data.

For example, if we want to study college students in U.S.A:
population = all college students in U.S.A.
a sample = 1000 students randomly selected in MST

- ▶ In most cases, the dataset set we have in hand does not cover all subjects. However, we still want to know the patterns or relationships for the whole population.
- ▶ In order to make statements on population based on data collected from sample, the sample must be representative of the population.

Sampling Model

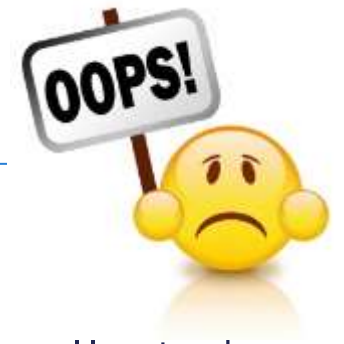
- ▶ Identify the population and then draw a fair sample from that population and do statistical analysis;
- ▶ Because the sample is representative of the population, you can automatically generalize your results back to the population.



How to Make Your Findings Generalizable?

- ▶ The purpose of business analytics/machine learning is to find patterns from the known data (training data) that can generalize to unknown data.
- ▶ If the training data are not representative of the whole population, the patterns you find could be seriously biased.
- ▶ A key principle of data management is to organize a dataset that is representative of the population.

Some Common Mistakes



- ▶ Raw data collected are not representative
 - Analytics project tries to find patterns at company level, but the data are collected from a single department.
- ▶ Improperly transform representative raw data into a nonrepresentative dataset
 - Drop a significant amount of observations with missing values.
 - Drop a significant amount of observations of a particular class, in order to make a balanced dataset.

AGENDA

- ▶ Data Management in Data Science
- ▶ Dataset and Scales of Measurement
- ▶ Pandas Data Structure
- ▶ Manipulate Data
 - Manipulate Strings
 - Manipulate Datasets
 - Rescale Data
- ▶ Import and Collect Data
 - Work with Flat Files and Relational Database
 - Collect Data from the Internet

Data and Data Set

- ▶ Data are the facts collected, analyzed, and interpreted.
- ▶ The data collected in a particular data science project are commonly referred to as a data set.

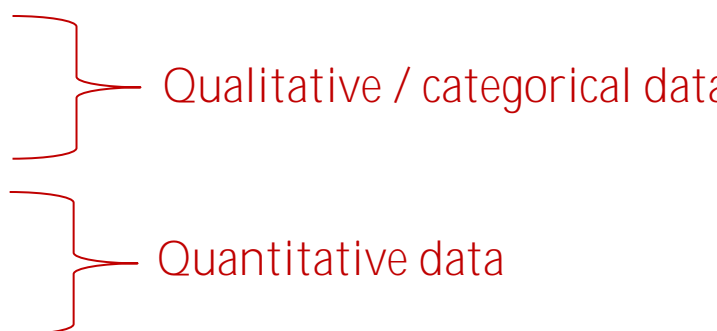
Data Set: Elements, Variables, and Observations

- ▶ Elements/subjects: entities of interest
- ▶ Variables: characteristic of elements
- ▶ Observation: the set of measurements obtained for an element

The diagram shows a data table with annotations. The word 'Variables' is positioned above the table with four arrows pointing to the columns 'mpg', 'cyl', 'hp', and 'wt'. The word 'Observations' is to the right of the table with an arrow pointing to the row for 'Mazda RX4 Wag'. The words 'Element' and 'Names' are to the left of the table, with arrows pointing to the rows for 'Mazda RX4' and 'Mazda RX4 Wag' respectively. The row for 'Mazda RX4 Wag' is highlighted with a blue border, and the cell containing '2.875' is also highlighted with a blue border.

car	mpg	cyl	hp	wt
Mazda RX4	21	6	110	2.62
Mazda RX4 Wag	21	6	110	2.875
Datsun 710	22.8	4	93	2.32
Hornet 4 Drive	21.4	6	110	3.215
Hornet Sportabout	18.7	8	175	3.44
Valiant	18.1	6	105	3.46

Scales of Measurement

- ▶ Scale/level of measurement determines:
 - the amount of information contained in data
 - data summarization and analysis methods that are appropriate
- ▶ Four types of scales
 - Nominal
 - Ordinal
 - Interval
 - Ratio

The diagram uses red curly braces to group the four scales. The first two, 'Nominal' and 'Ordinal', are grouped by a brace pointing to the text 'Qualitative / categorical data'. The next two, 'Interval' and 'Ratio', are grouped by a brace pointing to the text 'Quantitative data'.

 - Qualitative / categorical data
 - Quantitative data

Nominal Scale

- ▶ Numerical values are just names or labels of the attribute
 - Ordering of these values is meaningless
 - No mathematical calculation (+, -, *, /) applicable
- ▶ For example:
 - Gender (1 = “Male”, 0 = “Female”)
 - Student ID (1,2,3...)
 - Department (1 = “BIT”, 2 = “CS”...)
 - Zip code (65401, 65402...)

Ordinal Scale

- ▶ Attributes can be ranked/ordered.
- ▶ For example:
 - Football team rank (1st, 2nd, 3rd...)
 - Customer rating (1 = “Bad”, 2 = “OK”, 3 = “Excellent”)

Interval Scale

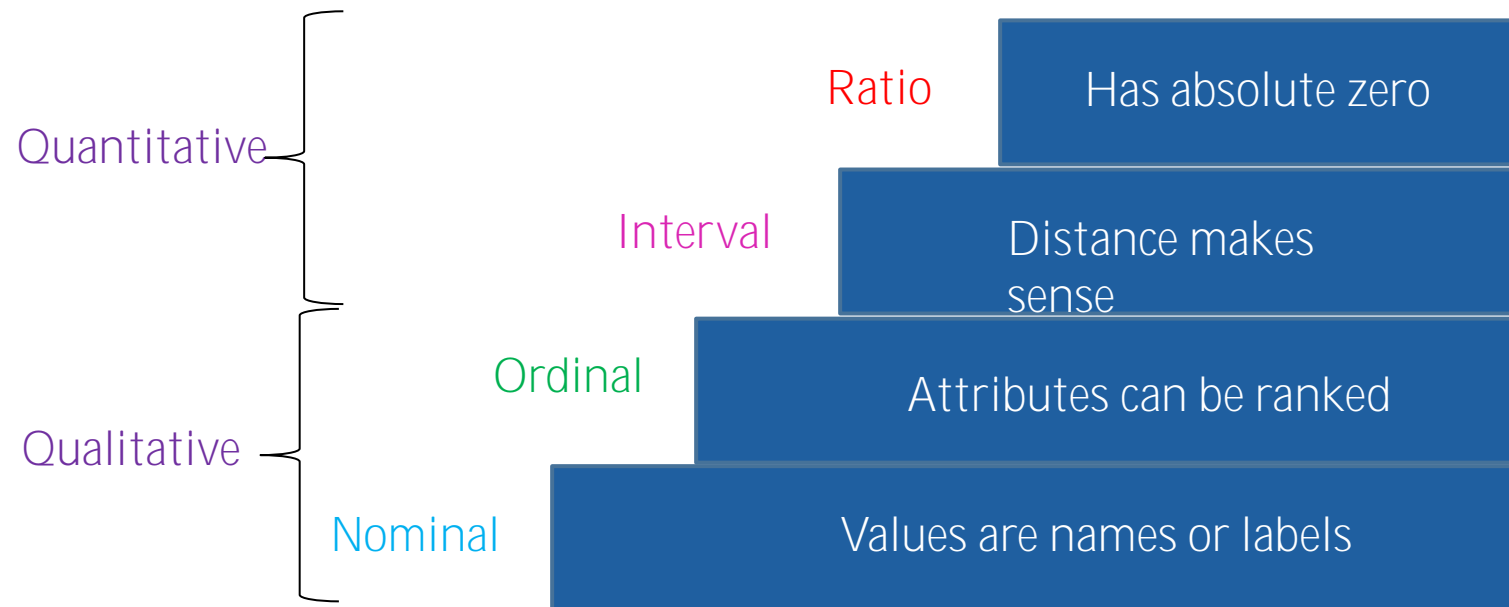
- ▶ Have all characteristics of ordinal scale
- ▶ Distance between attributes does have meaning.
- ▶ Ratios are not meaningful.
- ▶ For example:
 - Celsius or Fahrenheit temperature
 - ▶ The distance from 40 – 60 is same as the distance from 60 – 80
 - ▶ 80 cannot be said as twice hot as 40
 - SAT score
 - GMAT score

Ratio Scale

- ▶ Have all characteristics of interval scale
- ▶ A ratio of two values is meaningful.
- ▶ An absolute zero is meaningful.
- ▶ For example:
 - Weight
 - Height
 - Distance
 - Number of visits
 - Credit hours earned

Hierarchy of Measurement Scales

- ▶ A higher level scale contains all properties of its lower scale.
- ▶ From lower to higher levels, analysis tends to be more comprehensive. Improper use of lower level scales suffers information loss in the data
- ▶ In general, we prefer a higher scale of measurement than a lower one.



Scales of Measurement in Python

Scale of Measurement	Data Type in Python
Nominal	Boolean, Dummy Variables, Strings, Pandas Category
Ordinal	Base on models (usually numbers)
Interval	Numbers (integers, floats etc.)
Ratio	

Exercise

- ▶ Decide the scales of measurement for the following columns:

A sales summary of two stores by operating hour

Store#	City	Hour	Sale
101	Rolla, MO	9	\$1,000
101	Rolla, MO	10	\$1,100
101	Rolla, MO	11	\$1,200
102	St. Louis, MO	9	\$3,000
102	St. Louis, MO	10	\$3,300
102	St. Louis, MO	11	\$4,000

Note: In the hour column, 9 means time between 9AM and 10AM.

Are All Temperatures Interval Scale?

- ▶ Celsius and Fahrenheit scales are interval scales, since the location of zero is arbitrary
- ▶ The location of zero on the Kelvin scale is not arbitrary. Thus, the Kelvin scale is ratio scale.
 - Absolute zero is defined as the state at which particles stop moving.
 - However, researchers are trying to create a temperature below absolute zero.

Kelvin (K) \longleftrightarrow Celsius (°C)	Kelvin (K) \longleftrightarrow Fahrenheit (°F)
T_K : Temperature in Kelvin (K) T_C : Temperature in Celsius (°C) $T_K = T_C + 273.15$ $T_C = T_K - 273.15$	T_K : Temperature in Kelvin (K) Kelvin T_F : Temperature in Fahrenheit (°F) $T_K = \frac{5}{9} (T_F + 459.67)$ $T_F = \frac{9}{5} T_K - 459.67$

Image source:

<https://www.learnalberta.ca/content/memg/Division04/Interval%20Scale/index.html>

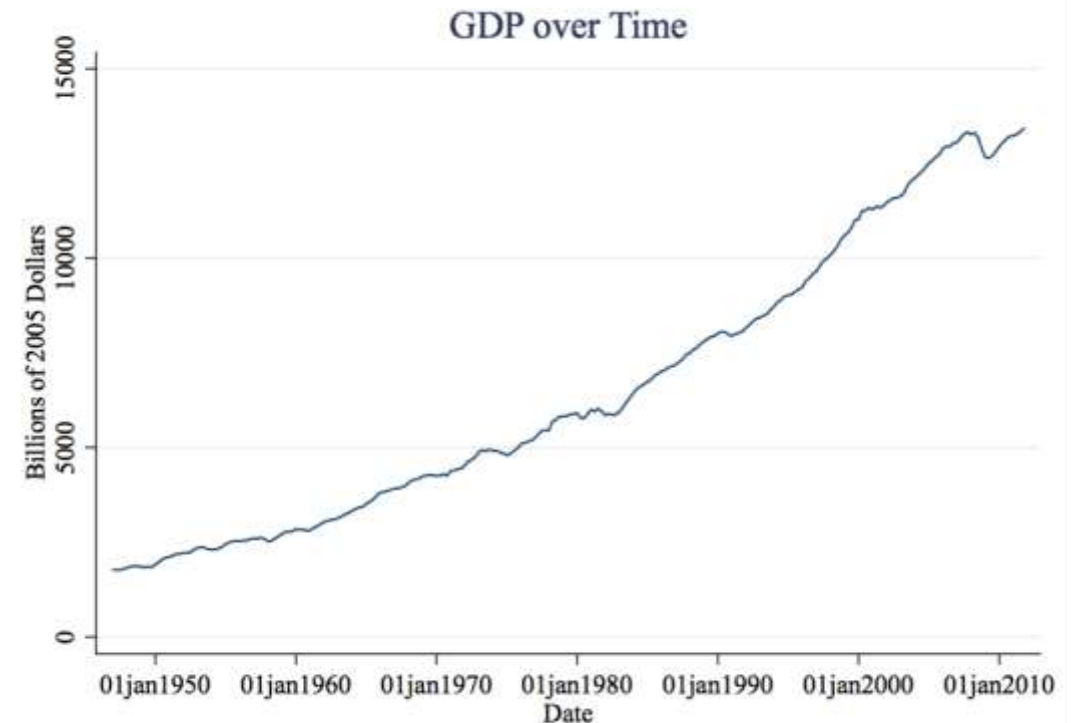
Things Could be Tricky: Measurement Scale for Year

- ▶ GDP has an increasing trend over time
- ▶ We want to predict world GDP in 2020 using the following regression model:

$$GDP_{year} = \beta_0 + \beta_1 * year$$

What is the scale of measurement for *year*?

Ratio



Things Could be Tricky: Measurement Scale for Year

- ▶ It seems a consumer's spending is partly determined by his/her income.
- ▶ We collected a dataset of annual spending and revenue from 100 consumers across a 5-year period from 2011 to 2015.
- ▶ We want to estimate the effect of annual income on annual spending by controlling for possible time effect.

$$Spend_{i,t} = \beta_0 + \beta_1 * Income_{i,t} + \theta * year_dummies \quad \Rightarrow \text{A panel regression}$$

where $t=1,2,\dots,5$, $i = 1, 2, \dots, 100$

What is the scale of measurement for *year*?

Nominal

AGENDA

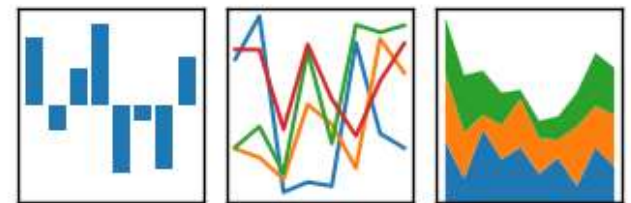
- ▶ Data Management in Data Science
- ▶ Dataset and Scales of Measurement
- ▶ Pandas Data Structure
- ▶ Manipulate Data
 - Manipulate Strings
 - Manipulate Datasets
 - Rescale Data
- ▶ Import and Collect Data
 - Work with Flat Files and Relational Database
 - Collect Data from the Internet

Pandas in Python

- ▶ Pandas provides high level data manipulation features on top of NumPy.
- ▶ Unlike NumPy that only supports homogeneous arrays and matrices, Pandas allows multiple data types in a single dataset.
- ▶ Since Pandas is an extension of NumPy, NumPy methods can be called by a Pandas object. Beyond that, Pandas provides a rich set of high level features.

pandas

$$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$$



Demo: Pandas Data Structure

AGENDA

- ▶ Data Management in Data Science
- ▶ Dataset and Scales of Measurement
- ▶ Pandas Data Structure
- ▶ **Manipulate Data**
 - Manipulate Strings
 - Manipulate Datasets
 - Rescale Data
- ▶ Import and Collect Data
 - Work with Flat Files and Relational Database
 - Collect Data from the Internet

String Manipulation with Pandas

- ▶ Pandas Series contains many useful vectorized string operators (under the “**str**” attribute):
 - Splitting and Replacing Strings
 - Extracting Substrings
 - Testing for Strings that Match or Contain a Pattern
 - Create Dummy Variables
 -

Regular Expressions in Python

- ▶ A regular expression (or RE) specifies a set of strings that matches it.
- ▶ To learn more, visit: <https://docs.python.org/3/howto/regex.html>

Some Special Symbols in RE

- ▶ Use backslash character ('\') to indicate special forms
- ▶ Repetition qualifiers: ***** (0 or more), **+** (1 or more), **?** (0 or 1), **{m,n}** (m to n repetition)
- ▶ **'.'**: any character except a newline
- ▶ **'^'**: start of a string
- ▶ **'\$'**: end of a string
- ▶ **'\d'**: any unicode decimal digit
- ▶ **'\D'**: not a unicode decimal digit
- ▶ **'\w'**: Unicode word character
- ▶ **'\W'**: not a unicode word character
- ▶ **'[]'**: a set of characters

What's the meaning of "**\d{3,5}**"?

For a complete list, refer to <https://docs.python.org/3/library/re.html#re-syntax>

Demo

- ▶ Manipulation Strings

Data Wrangling with Pandas

- ▶ Make New Variable
- ▶ Subset observations (rows)
- ▶ Subset variables (columns)
- ▶ Reshape data (conversion between long and wide format)
- ▶ Group data
- ▶ Handling missing data
- ▶ Merge datasets

For the detail, refer to [Pandas Cheat Sheet.pdf](#)

Comparing Some Common Methods with R

- ▶ Pandas syntax is more object-oriented, while R is more function-oriented

Function	Pandas	R
summary of the dataset	<code>df.info()</code>	<code>str(df)</code>
dimension of the dataset	<code>df.shape</code>	<code>dim(df)</code>
head	<code>df.head()</code>	<code>head(df)</code>
summary statistics	<code>df.describe()</code>	<code>summary(df)</code>
select columns	<code>df[['col1', 'col2']]</code>	<code>select(df, col1, col2)</code>
sorting	<code>df.sort_values(['col1', 'col2'])</code>	<code>arrange(df, col1, col2)</code>
create new columns from existing ones	<code>df.assign(c = df.a - df.b)</code>	<code>mutate(df, c=a-b)</code>
group by	<code>gdf = df.groupby('col1')</code>	<code>gdf <- group_by(df, col1)</code>
aggregate	<code>df.groupby('col1').sum()</code>	<code>summarise(gdf, total=sum(col1))</code>
distinct values	<code>df[['col1', 'col2']].drop_duplicates()</code>	<code>distinct(select(df, col1, col2))</code>
sampling	<code>df.sample(n=10)</code>	<code>sample_n(df, 10)</code>

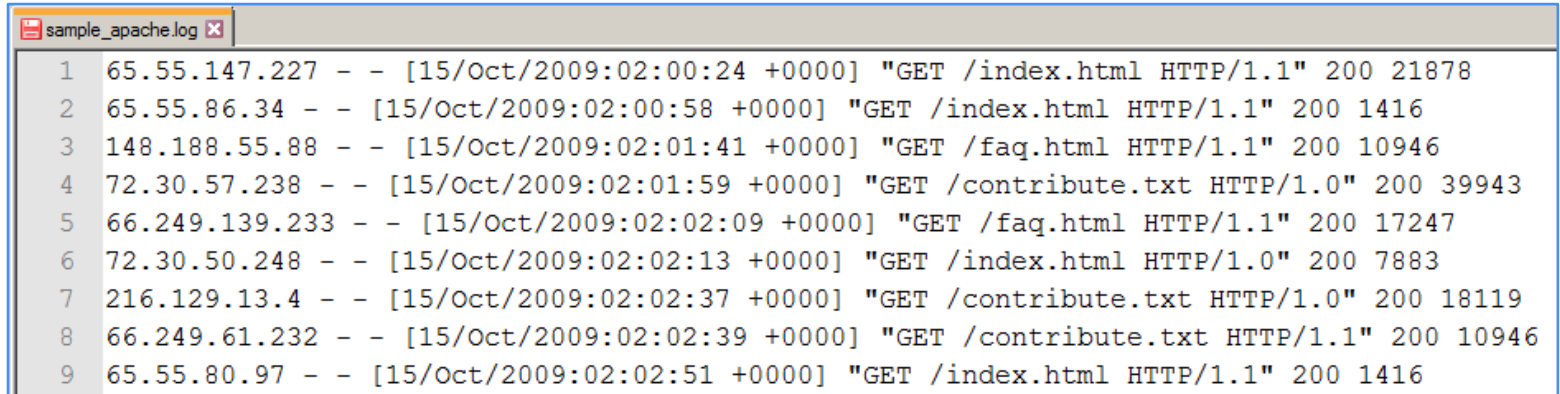
An Example: Apache Web Log Analysis

- ▶ Company XYZ runs a website. The company is confronting the challenge of extracting information and knowledge from this website to better understand how visitors access the online service.
- ▶ The objective of this data analytics project is to answer the following questions:
 - What are the top 10 countries from which the visitors come?
 - How many visits do we have for the FAQ page?
 - More.....

Apache Web Server Log

- ▶ The log file records all HTTP requests from browser clients to web server.

- Remote host
- Request time
- Request method
- Request URI
- Request protocol
- Status
- Size of request



1	65.55.147.227	- -	[15/Oct/2009:02:00:24 +0000]	"GET /index.html HTTP/1.1"	200	21878
2	65.55.86.34	- -	[15/Oct/2009:02:00:58 +0000]	"GET /index.html HTTP/1.1"	200	1416
3	148.188.55.88	- -	[15/Oct/2009:02:01:41 +0000]	"GET /faq.html HTTP/1.1"	200	10946
4	72.30.57.238	- -	[15/Oct/2009:02:01:59 +0000]	"GET /contribute.txt HTTP/1.0"	200	39943
5	66.249.139.233	- -	[15/Oct/2009:02:02:09 +0000]	"GET /faq.html HTTP/1.1"	200	17247
6	72.30.50.248	- -	[15/Oct/2009:02:02:13 +0000]	"GET /index.html HTTP/1.0"	200	7883
7	216.129.13.4	- -	[15/Oct/2009:02:02:37 +0000]	"GET /contribute.txt HTTP/1.0"	200	18119
8	66.249.61.232	- -	[15/Oct/2009:02:02:39 +0000]	"GET /contribute.txt HTTP/1.1"	200	10946
9	65.55.80.97	- -	[15/Oct/2009:02:02:51 +0000]	"GET /index.html HTTP/1.1"	200	1416

Demo

- ▶ Manipulation Datasets

Why Rescaling Data?

- ▶ Algorithms that rely on the similarity/distance between instances are sensitive to the scale of data. For example, kNN, SVM, and clustering methods (hierarchical clustering, k-means).
- ▶ Algorithms such as decision tree, naive Bayes, random forests are invariant to scaling.

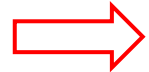
An Example

- ▶ Measuring similarity between cars by mpg and price.

Car	MPG	Price(\$)	Price(\$/1000)
A	32	24000	24
B	28	26000	26
C	22	27000	27

- ▶ Use original price:

- $\text{Euclidean_distance}(A,B) = 2000.001$
- $\text{Euclidean_distance}(B,C) = 1000.032$



B is more similar to C than to A

- ▶ Use price in thousand:

- $\text{Euclidean_distance}(A,B) = 2.828$
- $\text{Euclidean_distance}(B,C) = 8.062$



B is more similar to A than to C

Data Preprocessing and Normalization

- ▶ For the detailed methods supported by sklearn.preprocessing module, refer to
 - <http://scikit-learn.org/stable/modules/classes.html#module-sklearn.preprocessing>

<code>preprocessing.scale(X, axis=0, with_mean=True, with_std=True, copy=True)</code>	Center to the mean and component wise scale to unit variance.
<code>preprocessing.Binarizer([threshold, copy])</code>	Binarize/dichotomize data (set feature values to 0 or 1) according to a threshold
<code>preprocessing.MinMaxScaler([feature_range, copy])</code>	Transforms features by scaling each feature to a given range.
<code>preprocessing.Normalizer([norm, copy])</code>	Normalize samples individually to unit norm.
<code>preprocessing.maxabs_scale(X[, axis, copy])</code>	Scale each feature to the $[-1, 1]$ range without breaking the sparsity.

When you have both training and test dataset

- ▶ A wrong method:
 - Scale training and test dataset independently
- ▶ Both training and test datasets should be scaled by the same method
 - Use the training dataset to fit the scaling method
 - Use the fitted scaling method to transform both the training and test dataset
 - When the trained model is applied to do real prediction/classification, the same scaling method needs to be used to process the new data.

```
scaler = preprocessing.StandardScaler().fit(X_train)
X_train_scaled = scaler.transform(X_train)
X_test_scaled = scaler.transform(X_test)
```

Demo

- ▶ Rescaling Data

AGENDA

- ▶ Data Management in Data Science
- ▶ Dataset and Scales of Measurement
- ▶ Pandas Data Structure
- ▶ Manipulate Data
 - Manipulate Strings
 - Manipulate Datasets
 - Rescale Data
- ▶ Import and Collect Data
 - Work with Flat Files and Relational Database
 - Collect Data from the Internet

Common Data Import/Collection Methods

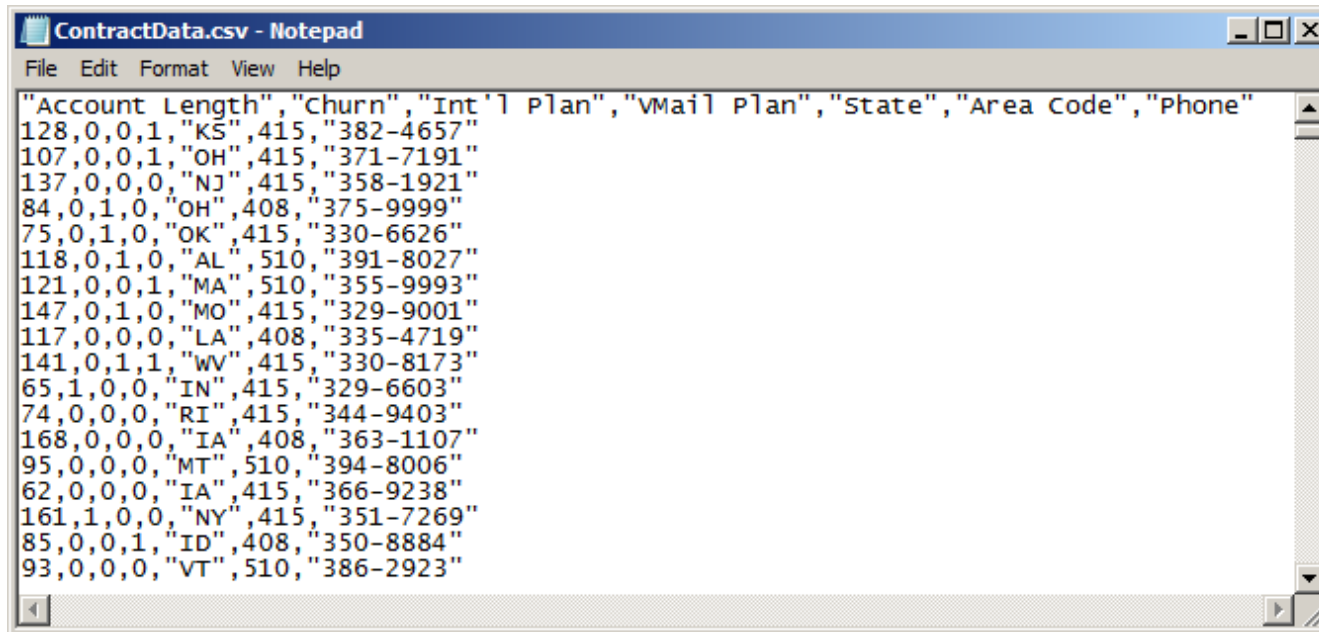
- ▶ Flat file such as comma-separated values (CSV), tab-separated values (TSV)
- ▶ Relational Database
- ▶ Web Scraping (copy and paste is inefficient)
 - HTML
 - API (XML and JSON)

Web scraping (web harvesting or web data extraction) is a computer software technique of extracting information from websites.

---- Wikipedia (https://en.wikipedia.org/wiki/Web_scraping)

Comma Separated Values (CSV) Files

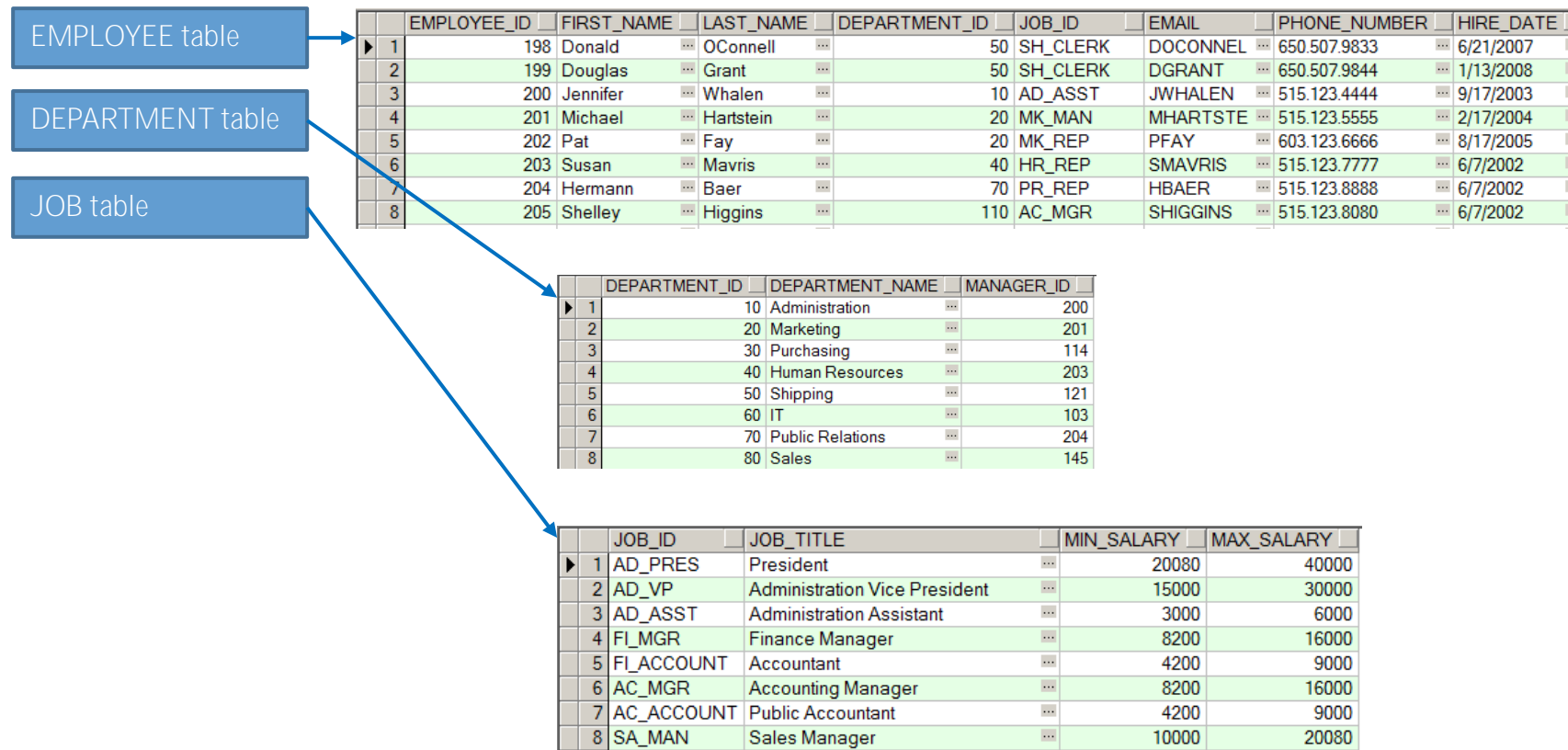
- ▶ CSV is a widely used data exchange format. Nowadays many companies are still using CSV to share information with their customers and suppliers.
- ▶ CSV files save tabular data in plain text.
 - Use comma (",") to separate data fields
 - May use the first line as a header containing field names
 - May use single or double quotation marks around some or all fields



```
ContractData.csv - Notepad
File Edit Format View Help
"Account Length","Churn","Int'l Plan","VMail Plan","State","Area Code","Phone"
128,0,0,1,"KS",415,"382-4657"
107,0,0,1,"OH",415,"371-7191"
137,0,0,0,"NJ",415,"358-1921"
84,0,1,0,"OH",408,"375-9999"
75,0,1,0,"OK",415,"330-6626"
118,0,1,0,"AL",510,"391-8027"
121,0,0,1,"MA",510,"355-9993"
147,0,1,0,"MO",415,"329-9001"
117,0,0,0,"LA",408,"335-4719"
141,0,1,1,"WV",415,"330-8173"
65,1,0,0,"IN",415,"329-6603"
74,0,0,0,"RI",415,"344-9403"
168,0,0,0,"IA",408,"363-1107"
95,0,0,0,"MT",510,"394-8006"
62,0,0,0,"IA",415,"366-9238"
161,1,0,0,"NY",415,"351-7269"
85,0,0,1,"ID",408,"350-8884"
93,0,0,0,"VT",510,"386-2923"
```

Relational Database

- ▶ A relational database consists of a collection of related data tables



Structured Query Language (SQL)

- ▶ A query language designed to manipulate data held in a relational DBMS
- ▶ Initially developed at IBM in the early 1970s
- ▶ Became a standard of the American National Standards Institute (ANSI) in 1986
- ▶ Beyond the ANSI-standard SQL, variants are supported by different DBMS platforms
 - Oracle: PL/SQL
 - Microsoft SQL Server: Transact-SQL
 - IBM DB2: SQL PL
 - MySQL: SQL/PSM

SQL Select Statement

- ▶ SELECT is the most common operation in SQL.
- ▶ SELECT retrieves data from one or more tables, or expressions.
- ▶ Standard SELECT statements have no persistent effects on the database.

- ▶ Syntax:

```
SELECT {ColumnName(s)}  
FROM {TableName(s)}  
WHERE {Condition(s)}
```

Where to Store Your Data?

Flat Files

- ▶ Flat file is a very simple design: it uses plain text to store tabular data.
- ▶ Flat files can be easily read or written by using Notepad, MS Excel, and other software tools.
- ▶ However, a plain text format is very difficult to store complicated data such as long textual data. Important format information may be lost.

Database

- ▶ Database is very good for storing structured data.
- ▶ The database architecture ensures efficient storage and access to the data.
- ▶ However, the rigid structure of databases does not allow for flexibility of dealing with unstructured data.
- ▶ In some cases, we cannot determine the format of the data since the full set of data has not been collected yet. In this case, a pre-determined fixed format would be problematic.

If the data structure is complicated, save the dataset into a pickle file. Pickle is a method for serializing Python structure objects. For more detail, refer to

<https://docs.python.org/3/library/pickle.html>



Demo

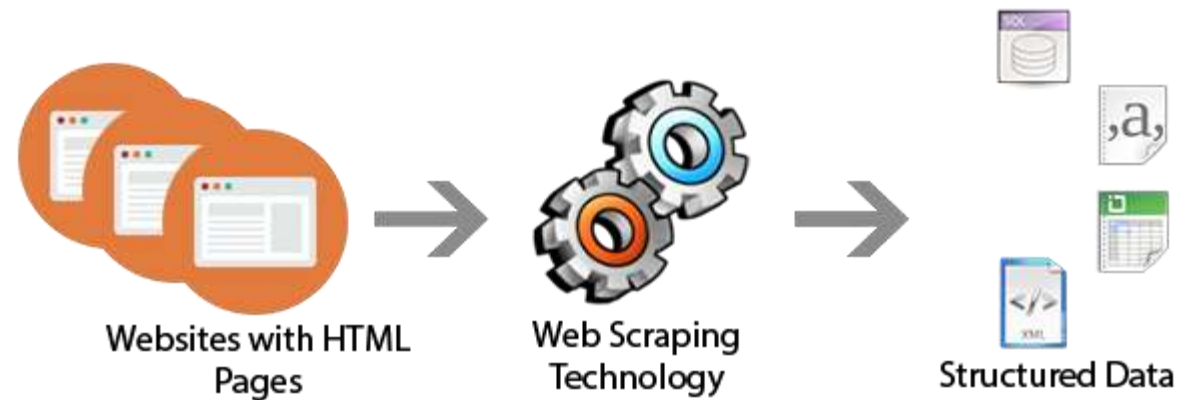
- ▶ Working with Flat Files and Relational Database

General Ways of Collecting Data From the Internet

- ▶ Web Scraping
 - Directly Parsing HTML
- ▶ Calling APIs Provided by the Website
 - Parsing XML Files Returned from the API
 - Parsing JSON Files Returned from the API

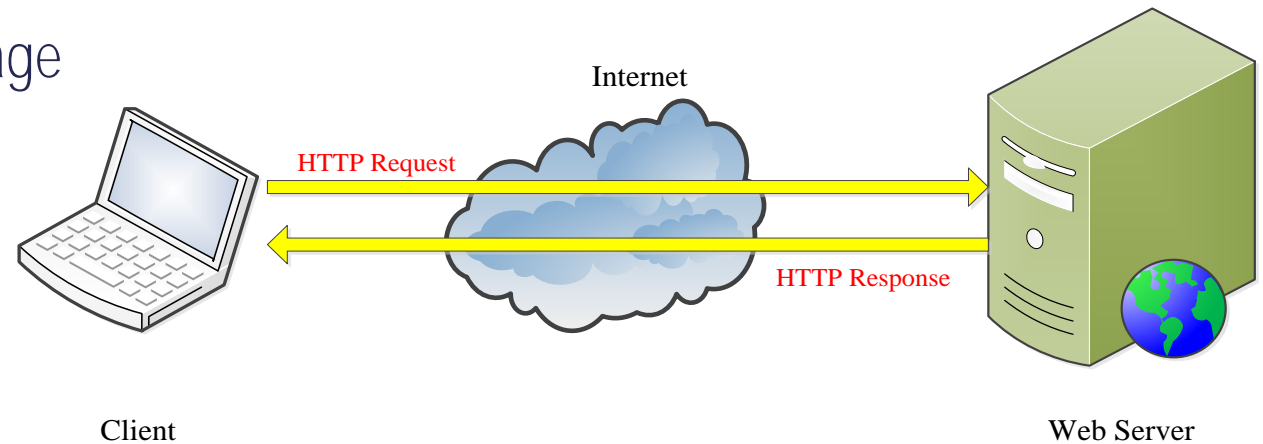
Web Scraping

- ▶ Web scraping is the process of extract data from websites by parsing the HTML tags.



Web Scraping Process

- ▶ Inspect the website to fully understand the web page structure and content
- ▶ Steps in scraping web pages
 - Specify URL
 - Get the web page content via HTTP request
 - Parse the web page content
 - Read specific elements in the web page
- ▶ Useful Packages in Python
 - requests: for HTTP communication
 - BeautifulSoup: for HTML parsing



An Example: Scraping Data from Online Forums

- ▶ Forum: <https://www.datasciencecentral.com/forum>

The screenshot shows the homepage of Data Science Central, an online resource for big data practitioners. The header includes the site's logo and a navigation menu with links to Home, AI, ML, DL, Analytics, Statistics, Big Data, DataViz, Hadoop, Podcasts, Webinars, Forums, Jobs, Membership, Groups, Search, and Contact. Below the header is a search bar and a link to subscribe to the DSC Newsletter. The main content area is titled 'Forum Discussions (673)' and features a list of discussions. The first three discussions are highlighted as 'Featured Discussions':

- What's the most advanced math you ever used in data science projects?**
I have used quite a bit of advanced math, especially to solve problems in experimental mathematics using data science methods. For instance...
Started by Vincent Granville
Latest Reply
- Need career guidance to become a Data Analyst**
Hi, My qualification is B Tech(ECE) I want to build my career as a Data Analyst. Please let me know what are the best online resources for...
Started by Rebbha Bhargav
Latest Reply
- One day, will humans be to AI what dogs are to humans now?**
Do you think that one day, humans will find a way to not work and enjoy the life, relying on robots to help them with their needs. Just lik...
Started by Vincent Granville
Latest Reply

Below the featured discussions is a table of discussions:

Discussions	Replies	Latest Activity
What's the most advanced math you ever used in data science projects? I have used quite a bit of advanced math, especially to solve problems in experimental mathematics using data science methods. For instance... Started by Vincent Granville	0	yesterday

The right sidebar contains a welcome message, a sign-up/sign-in button, and social media links. Below this is a 'RESOURCES' section with links to Join DSC, Free Books, Forum Discussions, Cheat Sheets, Jobs, Search DSC, DSC on Twitter, and DSC on Facebook. At the bottom is a 'VIDEOS' section featuring a video titled 'DSC Webinar Series: Analyze Data Faster with an Open Source Columnar Database'.

Inspect HTML Tags

- ▶ In Chrome browser, right-click and select “Inspect” menu.

The image shows a Chrome browser window with the 'Inspect' menu open. The menu options are: Back (Alt+Left Arrow), Forward (Alt+Right Arrow), Reload (Ctrl+R), Save as... (Ctrl+S), Print... (Ctrl+P), Cast..., Translate to English, AdBlock, View page source (Ctrl+U), and Inspect (Ctrl+Shift+I). A yellow arrow points from the 'Inspect' option to the developer tools interface.

The developer tools interface is open, showing the 'Elements' panel on the left and the 'Styles' panel on the right. The 'Elements' panel displays the HTML structure of the page, with the 'body' element selected. The 'Styles' panel shows the default styles for the selected element.

The browser window displays a forum page titled 'Forum Discussions (673)'. The page content includes a search bar, a 'Sort by' dropdown menu, and a list of featured discussions. The first discussion is titled 'What's the most advanced math you ever used in data science projects?' and is started by Vincent Granville. The second discussion is titled 'Need career guidance to become a Data Analyst' and is started by Rebbe Bhargav. The third discussion is titled 'One day, will humans be to AI what dogs are to humans now?' and is started by Vincent Granville.

Demo

- ▶ Web Scraping Example

Q & A

