

# **DATA SCIENCE**

## **DATA SOURCES**

## LAST TIME:

**I. CLEANING DATA**

**II. MISSING DATA**

## EXERCISES:

**III. NUMPY**

**IV. PANDAS**

**V. BOKEH & MATPLOTLIB**

# **QUESTIONS?**

**WHAT WAS THE MOST INTERESTING THING YOU LEARNT?**

**WHAT WAS THE HARDEST TO GRASP?**

**I. DATA SOURCES**

**II. DATA FORMATS**

**III. APIS**

**EXERCISES:**

**IV. RETRIEVE DATA FROM VARIOUS SOURCES**

**V. KIMONO LABS & OTHER APIS**

- **EXPLORE VARIOUS DATA SOURCES**
- **UNDERSTAND DIFFERENT DATA FORMATS**
- **BE ABLE TO RETRIEVE DATA FROM APIS**

---

**DATA FORMAT, ACCESS & TRANSFORMATION**

---

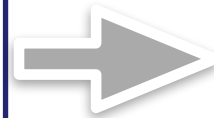
# **DATA SOURCES**

## DATA FLOW

### Data Retrieval



### Data ETL and Aggregation



### Data Visualization



### Machine Learning



**DATA FLOW**

## Data Retrieval



## Data ETL and Aggregation



## Data Visualization




## Machine Learning






## DATA SOURCES



**UCI**  
Machine Learning Repository  
Center for Machine Learning and Intelligent Systems











[About](#) [Citation Policy](#) [Donate a Data Set](#) [Contact](#)

☒ Repository ☐ Web



[View ALL Data Sets](#)

Browse Through: **298 Data Sets** Table View [List View](#)

|   | Name   | Data Types   | Default Task        | Attribute Types            | # Instances | # Attributes | Year |
|---|--|--------------|---------------------|----------------------------|-------------|--------------|------|
|    | <a href="#">Abalone</a>                      | Multivariate | Classification      | Categorical, Integer, Real | 4177        | 8            | 1995 |
|    | <a href="#">Adult</a>                        | Multivariate | Classification      | Categorical, Integer       | 48842       | 14           | 1996 |
|    | <a href="#">Annealing</a>                    | Multivariate | Classification      | Categorical, Integer, Real | 798         | 38           |      |
|    | <a href="#">Anonymous Microsoft Web Data</a> |              | Recommender-Systems | Categorical                | 37711       | 294          | 1998 |
|    | <a href="#">Arrhythmia</a>                   | Multivariate | Classification      | Categorical, Integer, Real | 452         | 279          | 1998 |
|  | <a href="#">Artificial Characters</a>        | Multivariate | Classification      | Categorical, Integer, Real | 6000        | 7            | 1992 |
|  | <a href="#">Audiology (Original)</a>         | Multivariate | Classification      | Categorical                | 226         |              | 1987 |
|  | <a href="#">Audiology (Standardized)</a>     | Multivariate | Classification      | Categorical                | 226         | 69           | 1992 |
|  | <a href="#">Auto MPG</a>                     | Multivariate | Regression          | Categorical, Real          | 398         | 8            | 1993 |
|  | <a href="#">Automobile</a>                   | Multivariate | Regression          | Categorical, Integer, Real | 205         | 26           | 1987 |

**Default Task**

Classification (213)  
Regression (41)  
Clustering (36)  
Other (50)

**Attribute Type**

Categorical (36)  
Numerical (161)  
Mixed (56)

**Data Type**

Multivariate (228)  
Univariate (15)  
Sequential (26)  
Time-Series (43)  
Text (27)  
Domain-Theory (20)  
Other (21)

**Area**

Life Sciences (75)  
Physical Sciences (41)  
CS / Engineering (78)  
Social Sciences (20)  
Business (14)  
Game (9)  
Other (59)

**# Attributes**

Less than 10 (74)  
10 to 100 (129)  
Greater than 100 (46)

**# Instances**


Less than 100 (15)  
100 to 1000 (113)  
Greater than 1000 (140)

**Format Type**





Matrix (213)  
Non-Matrix (85)

Source: <http://archive.ics.uci.edu/ml/datasets.html>

## DATA SOURCES




Espanol

Follow Us:
 




 1-800-FED-INFO (333-4636)

| Services and Information  | Government Agencies and Elected Officials  | Blog  |
|---|--|---|
| <ul style="list-style-type: none"> <li>• <a href="#">Benefits, Grants, and Loans</a></li> <li>• <a href="#">Businesses and Nonprofits</a></li> <li>• <a href="#">Consumer Complaints and Protection</a></li> <li>• <a href="#">Consumer Publications</a></li> <li>• <a href="#">Disasters, Public Safety, and Laws</a></li> <li>• <a href="#">Environment, Energy, and Agriculture</a></li> </ul> | <ul style="list-style-type: none"> <li>• <a href="#">Government Sales and Auctions</a></li> <li>• <a href="#">Health Insurance, Nutrition, and Food Safety</a></li> <li>• <a href="#">History, Genealogy, and Culture</a></li> <li>• <a href="#">Immigration, Citizenship, and International</a></li> <li>• <a href="#">Jobs, Training, and Education</a></li> <li>• <a href="#">Mortgages, Housing, and Family</a></li> </ul> | <ul style="list-style-type: none"> <li>• <a href="#">Passports and Travel</a></li> <li>• <a href="#">Public Service and Volunteerism</a></li> <li>• <a href="#">Reference and General Government</a></li> <li>• <a href="#">Register to Vote and Elections</a></li> <li>• <a href="#">Science and Technology</a></li> <li>• <a href="#">Unclaimed Money, Taxes, and Credit Reports</a></li> </ul> |


**More for Developers**

- [Other USA.gov Resources](#)
- [USA.gov GitHub Account](#)

**From Other Federal Agencies**

- [Other Federal Government Developer Resources](#)
- [Other Federal Government GitHub Accounts](#)

### About The Data

1.USA.gov URLs are created whenever anyone shortens a .gov or .mil URL using [bitly](#).

We provide a raw [pub/sub](#) feed of data created any time anyone clicks on a 1.USA.gov URL. The pub/sub endpoint responds to http requests for any 1.USA.gov URL and returns a stream of JSON entries, one per line, that represent real-time clicks.

If you are using the 1.USA.gov data and have questions, feedback, or want to tell us about your product, please [e-mail us](#).

### How to Access The Data

Source: <http://www.usa.gov/About/developer-resources/1usagov.shtml>

## DATA SOURCES



Source: <http://www.kaggle.com/>

- 1) PETE SKOMOROCH (LINKEDIN) [HTTPS://DELICIOUS.COM/PSKOMOROCH/DATASET](https://delicious.com/pskomoroch/dataset)
- 2) HILARY MASON (ACCEL PARTNERS, BITLY) [HTTPS://BITLY.COM/BUNDLES/HMASON/1](https://bitly.com/bundles/hmason/1)
- 3) KEVIN CHAI (U. OF NEW SOUTH WALES, SYDNEY) [HTTP://KEVINCHAI.NET/DATASETS](http://kevinchai.net/datasets)
- 4) JEFF HAMMERBACHER (CLOUDERA) [HTTP://WWW.QUORA.COM/JEFF-HAMMERBACHER/INTRODUCTION-TO-DATA-SCIENCE-DATA-SETS](http://www.quora.com/Jeff-Hammerbacher/introduction-to-data-science-data-sets)
- 5) JERRY SMITH (3I-MIND) [HTTP://DATASCIENTISTINSIGHTS.COM/2013/10/07/DATA-REPOSITORIES-MOTHERS-MILK-FOR-DATA-SCIENTISTS/](http://datascientistinsights.com/2013/10/07/data-repositories-mothers-milk-for-data-scientists/)
- 6) GREGORY PIATETSKY-SHAPIO (KDD) [HTTP://WWW.KDNUGGETS.COM/DATASETS/INDEX.HTML](http://www.kdnuggets.com/datasets/index.html)
- 7) [HTTP://WWW.QUORA.COM/DATA/WHERE-CAN-I-FIND-LARGE-DATASETS-OPEN-TO-THE-PUBLIC](http://www.quora.com/Data/Where-can-I-find-large-datasets-open-to-the-public)
- 8) [HTTPS://GITHUB.COM/CAESAR0301/AWESOME-PUBLIC-DATASETS](https://github.com/caesar0301/awesome-public-datasets)

**PAIR EXERCISE:**

**CHOOSE A DATA SOURCE AND LOOK AT WHAT DATA YOU CAN GET**

**DISCUSS HOW YOU WOULD USE THE DATA**

---

**DATA FORMAT, ACCESS & TRANSFORMATION**

---

**QUESTIONS?**

---

**DATA FORMAT, ACCESS & TRANSFORMATION**

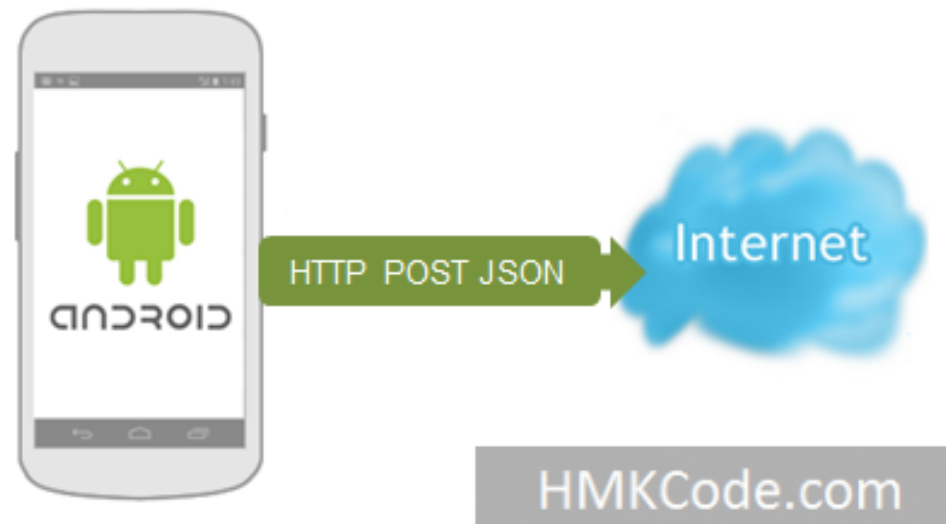
---

**JSON, CSV, ETC...**

**JSON** (JavaScript Object Notation) is:  
a lightweight **data-interchange** format  
a **string**



**JSON** can be passed  
between **applications**  
easy for **machines** to parse and generate



JSON are passed through applications  
as **strings**  
and converted into native objects per language.

JSON are passed through applications as **strings** and converted into native objects per language.

```
{ "empinfo" :  
  {  
    "employees" : [  
      {  
        "name" : "Scott Philip",  
        "salary" : f44k,  
        "age" : 27,  
      },  
      {  
        "name" : "Tim Henn",  
        "salary" : f40k,  
        "age" : 27,  
      },  
      {  
        "name" : "Long Yong",  
        "salary" : f40k,  
        "age" : 28,  
      }  
    ]  
  }  
}
```

```
import json  
  
py_object = [ { 'a':'A', 'b':(2, 4), 'c':3.0 } ]  
  
json_string = json.dumps(py_object)  
  
print 'JSON:', json_string
```

JSON: [{"a": "A", "c": 3.0, "b": [2, 4]}]

```
decoded = json.loads(json_string)
```

<https://docs.python.org/2/library/json.html>

## CSV (Comma Separated Values):

```
name , game , points  
John , basketball , 3  
Mary , volleyball , 5  
James , ping pong , 2  
...
```

## CSV (Comma Separated Values):

- easy to read and write
- structured like a table
- very common
- can export to/from MS Excel



<https://docs.python.org/2/library/csv.html>

---

## OTHER DATA FORMATS

---

txt

tsv

xml

dat

images

binary

etc...

---

**DATA FORMAT, ACCESS & TRANSFORMATION**

---

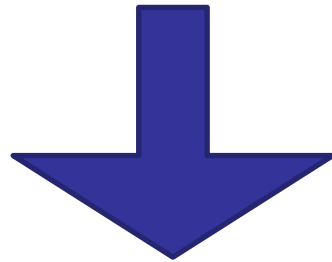
**APIs**

**APIs** (Application Programming Interface)  
allow people to **interact** with the structures of  
an application

- get
- put
- delete
- update
- ...

Best practices for APIs are to  
use **RESTful** principles.

Best practices for APIs are to  
use **RESTful** principles.



Representational State Transfer (REST)

## RESTFUL EXAMPLE

RESTful API HTTP methods

| Resource  | GET  | PUT   | POST  | DELETE  |
|---|--|---|---|---|
| <b>Collection URI, such as</b><br><code>http://example.com/resources/</code>    | <b>List</b> the URIs and perhaps other details of the collection's members.  | <b>Replace</b> the entire collection with another collection.                                     | <b>Create</b> a new entry in the collection. The new entry's URI is assigned automatically and is usually returned by the operation. <sup>[9]</sup> | <b>Delete</b> the entire collection.                  |
| <b>Element URI, such as</b><br><code>http://example.com/resources/item17</code> | <b>Retrieve</b> a representation of the addressed member of the collection, expressed in an appropriate Internet media type. | <b>Replace</b> the addressed member of the collection, or if it does not exist, <b>create</b> it. | Not generally used. Treat the addressed member as a collection in its own right and <b>create</b> a new entry in it. <sup>[9]</sup>                 | <b>Delete</b> the addressed member of the collection. |

- The Base URL
- An interactive media type (usually JSON)
- Operations (GET, PUT, POST, DELETE)
- Driven by http requests



---

## REST API EXAMPLE

---

Collection



**GET <https://api.instagram.com/v1/users/10>**



Operation

---

## REST API EXAMPLE

---

**GET https://api.instagram.com/v1/users/  
search/?q=andy**



Querystring

<https://dev.twitter.com/rest/public>

<https://developer.linkedin.com/docs/signin-with-linkedin>

<http://www.pythonapi.com/>

**PAIR EXERCISE:**

<http://www.pythonapi.com/>

**1) CHOOSE 1 API: WHAT DATA YOU CAN GET?**

**2) INSTALL PYTHON MODULE, TRY TO EXTRACT DATA**

**3) DISCUSS: HOW COULD YOU LEVERAGE THAT API? HOW COULD YOU USE THE DATA?**

## KIMONO LABS

[www.kimonolabs.com](http://www.kimonolabs.com)

kimono

Turn websites into structured APIs from your browser in seconds



Get started, click to install

---

**DATA FORMAT, ACCESS & TRANSFORMATION**

---

**QUESTIONS?**