



Ministry of Higher Education and Scientific Research
Djilali BOUNAAMA University - Khemis Miliana (UDBKM)
Faculty of Science and Technology
Department of Mathematics and Computer Science



Chapter 4

Data Science Sources

AIBD-M1-UEM112 : Introduction to Data Science

Noureddine AZZOUZA

n.azzouza@univ-dbkm.dz

Course Topics

1. Introduction

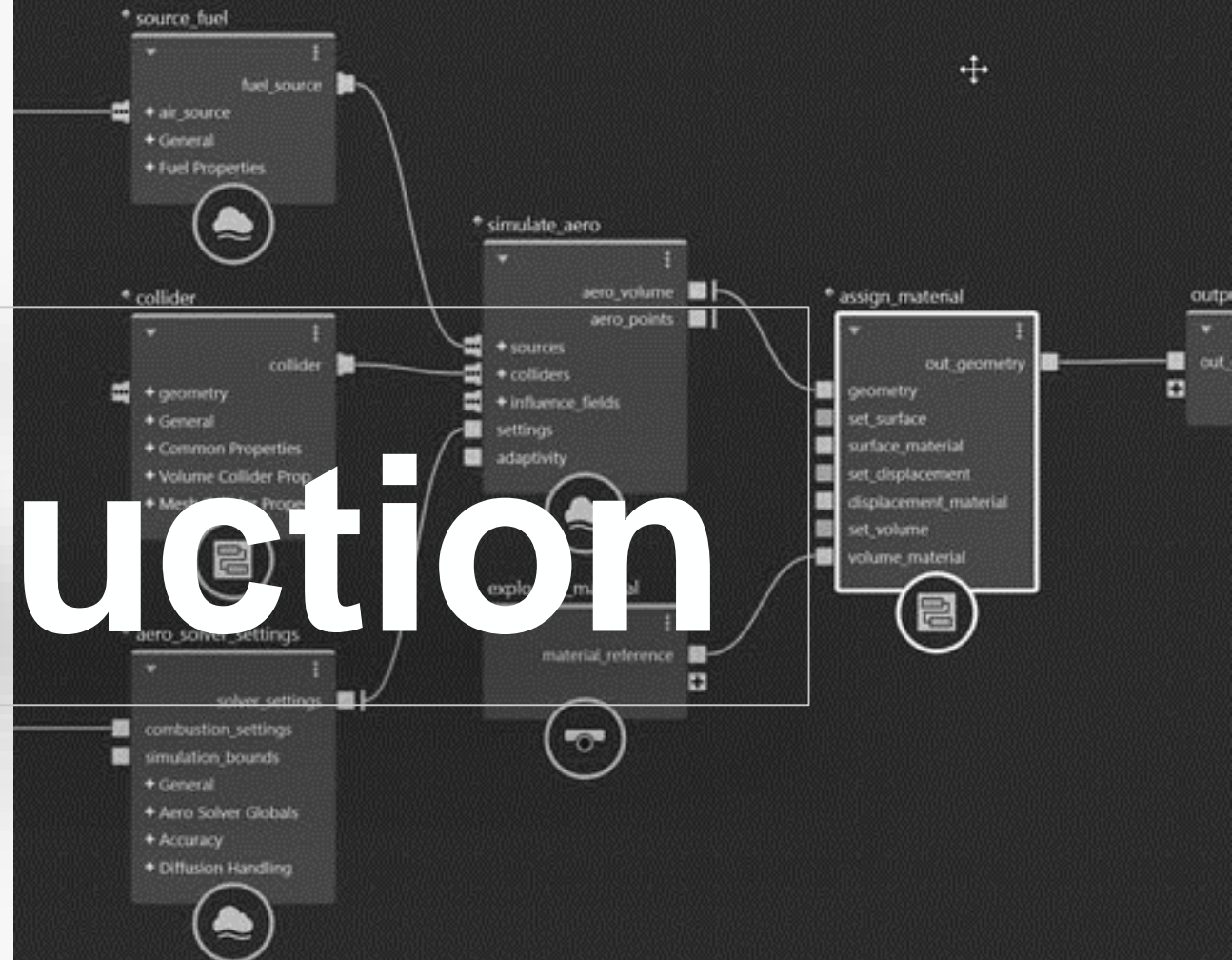
2. Data Sources and Data Collection

3. Primary Data and Secondary Data

4. Types of Data Sources

3. Data Sources in Data Science

Introduction



Introduction

- ✓ Data science is quickly evolving to be one of the hottest fields in the technology industry.
- ✓ With rapid advancements in computational performance, we can uncover patterns and insights about user behavior and world trends to an unprecedented extent.
- ✓ With the influx of buzzwords in the field of data science and relevant fields, a common question is “Data science sounds pretty cool - how to get started?”
- ✓ Here is a brief overview of steps that make up a data science lifecycle / Pipeline. For each step, we provide some useful resources.



Data Sources &

Data Collection



Data Sources

- ✓ Data sources are essentially where our data originates from.
- ✓ A data source is the physical or digital location where data is stored in different forms. In short, this is where the data comes from.
- ✓ The data source can be both where the data was originally created, but also where it was added. For example, as part of a digital transformation, many companies are digitizing their data.
- ✓ The place where they are stored electronically then becomes the source of this data.

Data Sources

- ✓ data sources can be digital or in paper format. the idea is to allow users to access and use data from this source.
- ✓ This is done in different ways, since the data source can take different forms, such as a database, a flat file, an inventory table, web scraping, streaming data, physical archives, etc.
- ✓ With the development of Big Data and new technologies, these different formats continue to evolve, making data sources ever more complex.
- ✓ The challenge for organizations is therefore to simplify them as much as possible.

Context

- ✓ Data sources can take different forms. But it mostly depends on the context.
- ✓ Very often, data sources and databases are confused. Both refer to where the information is stored. But the database is only one form of data source. It is also possible to consider the data source as a data provider, the use of self-service data, a type of computer storage, accounting, etc.
- ✓ Regardless of the format and context, the idea of the data source is to define where the data comes from and to describe the connections between the information.

Data Collection

- ✓ The purpose of data sources is to allow users to access the information they need, and possibly move or modify it.
- ✓ data experts must group all the information in one place in order to simplify its use and understanding.
- ✓ it is essential to design data sources from a user perspective in order to facilitate data processing.
- ✓ The information must then be stored consistently, both in terms of location and format.

Data Collection : Definition

- ✓ Data collection is the process of collecting and evaluating information or data from multiple sources to find answers to research problems, answer questions, evaluate outcomes, and forecast trends and probabilities. It is an essential phase in all types of research, analysis, and decision-making.
- ✓ Accurate data collection is necessary to make informed business decisions, ensure quality assurance, and keep research integrity.
- ✓ During data collection, the researchers must identify the data types, the sources of data, and what methods are being used.

Data Collection : Hypothesis

- ✓ Before an analyst begins collecting data, they must answer three questions first:
 1. What's the goal or purpose of this research?
 2. What kinds of data are they planning on gathering?
 3. What methods and procedures will be used to collect, store, and process the information?
- ✓ Additionally, we can break up data into qualitative and quantitative types. Qualitative data covers descriptions such as color, size, quality, and appearance. Quantitative data, unsurprisingly, deals with numbers, such as statistics, poll numbers, percentages, etc.

Need for Data Collection

- ✓ The concept of data collection isn't a new one, as we'll see later, but the world has changed. There is far more data available today, and it exists in forms that were unheard of a century ago. The data collection process has had to change and grow with the times, keeping pace with technology.
- ✓ Whether you're in the world of academia, trying to conduct research, or part of the commercial sector, thinking of how to promote a new product, you need data collection to help you make better choices.

Primary Data Source

- ✓ It is a collection of data from the source of origin. It provides the researcher with first-hand quantitative and raw information related to the statistical study. In short, the primary sources of data give the researcher direct access to the subject of research.

Primary Data Source

- ✓ There are various techniques for primary data collection, including:
- ✓ **a. Surveys and Questionnaires:** Researchers design structured questionnaires or surveys to collect data from individuals or groups. These can be conducted through face-to-face interviews, telephone calls, mail, or online platforms.
- ✓ **b. Interviews:** Interviews involve direct interaction between the researcher and the respondent. They can be conducted in person, over the phone, or through video conferencing. Interviews can be structured (with predefined questions), semi-structured (allowing flexibility), or unstructured (more conversational).

Primary Data Source

- ✓ **c. Observations:** Researchers observe and record behaviors, actions, or events in their natural setting. This method is useful for gathering data on human behavior, interactions, or phenomena without direct intervention.
- ✓ **d. Experiments:** Experimental studies involve the manipulation of variables to observe their impact on the outcome. Researchers control the conditions and collect data to draw conclusions about cause-and-effect relationships.
- ✓ **e. Focus Groups:** Focus groups bring together a small group of individuals who discuss specific topics in a moderated setting. This method helps in understanding opinions, perceptions, and experiences shared by the participants.

Secondary Data Source

- ✓ It is a collection of data from some institutions or agencies that have already collected the data through primary sources. It does not provide the researcher with first-hand quantitative and raw information related to the study. Hence, the secondary source of data collection interprets, describes, or synthesizes the primary sources. For example, reviews, government websites containing surveys or data, academic books, published journals, articles, etc.
- ✓ Even though primary sources provide more credibility to the collected data because of the presence of evidence, but good research will require both primary and secondary sources of data collection.

Secondary Data Source

- ✓ Secondary data can be obtained from various sources, including:
 - ✓ **a. Published Sources:** Researchers refer to books, academic journals, magazines, newspapers, government reports, and other published materials that contain relevant data.
 - ✓ **b. Online Databases:** Numerous online databases provide access to a wide range of secondary data, such as research articles, statistical information, economic data, and social surveys.

Secondary Data Source

- ✓ **c. Government and Institutional Records:** Government agencies, research institutions, and organizations often maintain databases or records that can be used for research purposes.
- ✓ **d. Publicly Available Data:** Data shared by individuals, organizations, or communities on public platforms, websites, or social media can be accessed and utilized for research.
- ✓ **e. Past Research Studies:** Previous research studies and their findings can serve as valuable secondary data sources. Researchers can review and analyze the data to gain insights or build upon existing knowledge.

Accurate Data Collection

- ✓ Accurate data collecting is crucial to preserving the integrity of research, regardless of the subject of study or preferred method for defining data (quantitative, qualitative).
- ✓ Errors are less likely to occur when the right data gathering tools are used (whether they are brand-new ones, updated versions of them, or already available).

Accurate Data Collection

- ✓ Among the effects of data collection done incorrectly, include the following -
 - Erroneous conclusions that squander resources
 - Decisions that compromise public policy
 - Incapacity to correctly respond to research inquiries
 - Bringing harm to participants who are humans or animals
 - Deceiving other researchers into pursuing futile research avenues
 - The study's inability to be replicated and validated

Accurate Data Collection

- ✓ Among the effects of data collection done incorrectly, include the following -
 - Erroneous conclusions that squander resources
 - Decisions that compromise public policy
 - Incapacity to correctly respond to research inquiries
 - Bringing harm to participants who are humans or animals
 - Deceiving other researchers into pursuing futile research avenues
 - The study's inability to be replicated and validated

Challenges in Data Collection

- ✓ **Data Quality Issues** : The main threat to the broad and successful application of ML is poor data quality. Data quality must be your top priority.
- ✓ **Inconsistent Data** : When working with various data sources, it's conceivable that the same information will have discrepancies between sources. The differences could be in formats, units, or occasionally spellings. Inconsistencies in data have a tendency to accumulate and reduce the value of data if they are not continually resolved.
- ✓ **Ambiguous Data** : errors can still occur in massive databases or data lakes. For data streaming at a fast speed, the issue becomes more overwhelming. Spelling mistakes can go unnoticed, formatting difficulties can occur, and column heads might be deceptive.

Challenges in Data Collection

- ✓ **Duplicate Data** : Streaming data, local databases, and cloud data lakes are just a few of the sources of data that modern enterprises must contend with. These sources are likely to duplicate and overlap each other quite a bit. For instance, duplicate contact information has a substantial impact on customer experience. The likelihood of biased outcomes increases when duplicate data are present. It can result in ML models with biased training data.
- ✓ **Too Much Data** : a data quality problem with excessive data exists. There is a risk of getting lost in an abundance of data when searching for information pertinent to your analytical efforts. Data scientists, data analysts, and business users devote 80% of their work to finding and organizing the appropriate data.

Challenges in Data Collection

- ✓ **Inaccurate Data** : data accuracy is crucial. Inaccurate information does not provide you with a true picture of the situation and cannot be used to plan the best course of action. Personalized customer experiences and marketing strategies underperform if your customer data is inaccurate. Data inaccuracies can be attributed to a number of things, including data degradation, human mistake, and data drift.
- ✓ **Hidden Data** : The majority of businesses only utilize a portion of their data. For instance, the customer service team might not receive client data from sales, missing an opportunity to build more precise and comprehensive customer profiles. Missing out on possibilities to develop novel products, enhance services, and streamline procedures is caused by hidden data.

Challenges in Data Collection

- ✓ **Finding Relevant Data** : Finding relevant data is not so easy. There are several factors that we need to consider while trying to find relevant data, which include : Relevant Domain, Relevant demographics, Relevant Time period ...
- ✓ **Dealing With Big Data** : These traits typically result in increased challenges while storing, analyzing, and using additional methods of extracting results. Big data refers especially to data sets that are quite enormous or intricate that conventional data processing tools are insufficient. The overwhelming amount of data, both unstructured and structured, that a business faces on a daily basis.

Steps of Data Collection Process

In the Data Collection Process, there are 5 key steps. They are explained briefly below :

- ✓ 1. Decide What Data You Want to Gather
- ✓ 2. Establish a Deadline for Data Collection
- ✓ 3. Select a Data Collection Approach
- ✓ 4. Gather Information
- ✓ 5. Examine the Information and Apply Your Findings

Types of

Data Sources



Data Sources

- ✓ Data sources can take different forms. But it mostly depends on the context.
- ✓ Very often, data sources and databases are confused. Both refer to where the information is stored. But the database is only one form of data source. It is also possible to consider the data source as a data provider, the use of self-service data, a type of computer storage, accounting, etc.
- ✓ Regardless of the format and context, the idea of the data source is to define where the data comes from and to describe the connections between the information.

Objectifs

- ✓ The purpose of data sources is to allow users to access the information they need, and possibly move or modify it.
- ✓ data experts must group all the information in one place in order to simplify its use and understanding.
- ✓ it is essential to design data sources from a user perspective in order to facilitate data processing.
- ✓ The information must then be stored consistently, both in terms of location and format.

Text Files

- ✓ The most basic method for data storage is using a text file. The content in the text file is structured and will follow a specific format.
- ✓ The most common usage of text file appears in logging. The log entries are stored in a specific format which can be read and extracted using a programming language.
- ✓ String parsing or Regex can be used to split each parameter in the entry.

Text Files

- #regex for parsing log file
- regex = "<your_regex_here>"
- #read the text file and use re.findall()
- File = open("text.log",'r')
- for i in File.readlines():
- print(re.findall(regex,i))parameter in t

```
log15 - Blocco note
File Modifica Cerca ?
2002-04-26 10:20:20 Accessing JGachesim: Graziano Aretusi

DETAILS OF OPERATIONS

2002-04-26 10:20:31 Configuration selected
2002-04-26 10:20:39 config_java module: Type of processor: E0086
2002-04-26 10:20:55 config_java module: Memory size: 16 KB
2002-04-26 10:21:10 config_java module: Memory size: 32 KB
2002-04-26 10:21:25 config_java module: Placement algorithm: set associative
2002-04-26 10:21:43 config_java module: Update algorithm: write through
2002-04-26 10:22:01 config_java module: Update algorithm: copy back
2002-04-26 10:22:20 config_java module: Cache size: 8 KB
2002-04-26 10:22:47 config_java module: Block size: 128 b
2002-04-26 10:22:53 config_java module: Block size: 64 b
2002-04-26 10:23:04 config_java module: Number of blocks per set: 8
2002-04-26 10:23:12 config_java module: Number of sets: 16
2002-04-26 10:23:18 Saving default parameters
2002-04-26 10:23:25 default.cfr successfully updated
2002-04-26 10:23:31 Execution selected
2002-04-26 10:23:39 exe_java.module: trace
2002-04-26 10:27:01 exe_java.module: execution stopped
2002-04-26 10:27:04 exe_java.module: statistics
2002-04-26 10:27:13 Statistics selected
2002-04-26 10:27:19 statistics_java.module: complete
2002-04-26 10:27:27 statistics_java.module: screen1
2002-04-26 10:29:18 statistics_java.module: screen2
2002-04-26 10:32:05 statistics_java.module: screen3
2002-04-26 10:35:20 Exit statistics selected
2002-04-26 10:35:28 Execution selected
2002-04-26 10:35:59 exe_java.module: exe
2002-04-26 10:37:00 exe_java.module: execution stopped
2002-04-26 10:37:05 exe_java.module: statistics
2002-04-26 10:37:16 Statistics selected
2002-04-26 10:37:21 statistics_java.module: summary
2002-04-26 10:41:09 Exit statistics selected
2002-04-26 10:43:45 Logoff JGachesim: Graziano Aretusi

SUMMARY OF MAIN OPERATIONS PERFORMED BY Graziano Aretusi

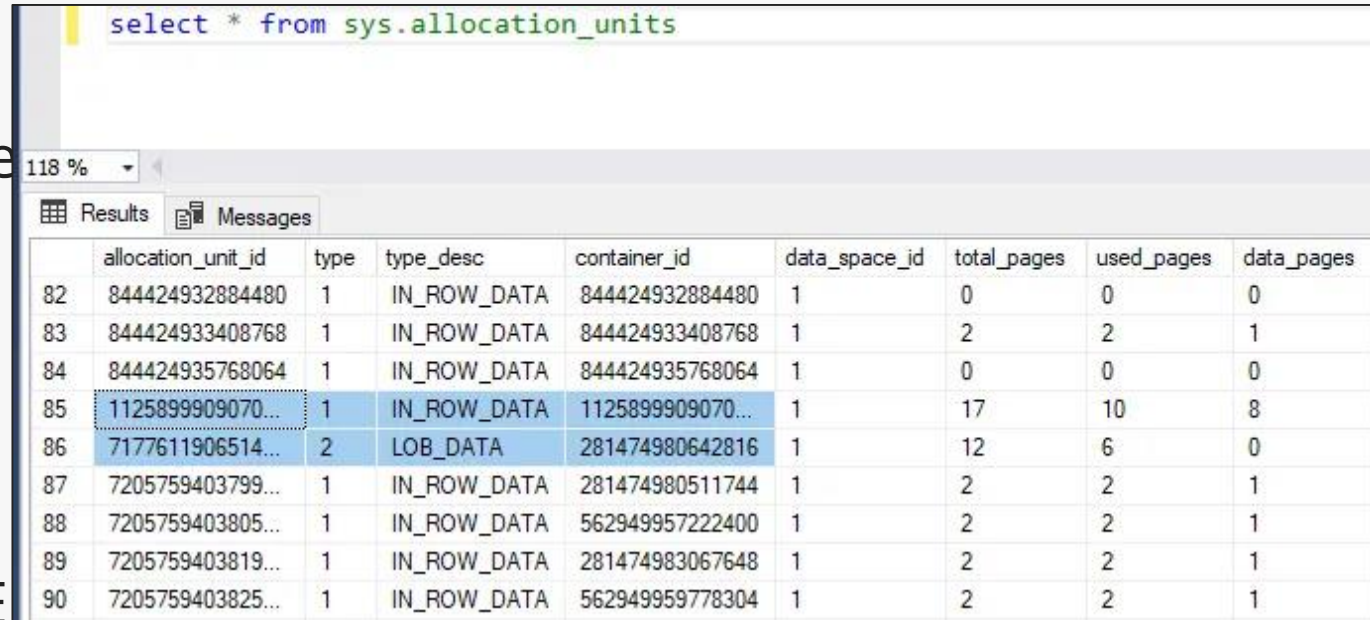
total time spent in trace tests: 00:03:22
total time spent for configuration: 00:03:01
total time spent for watching animated simulation:00:03:10
total idle time: 00:05:18
total time spent for reasoning between operations: 00:02:43
```

Relational and NoSQL Databases

- ✓ These include database management systems (DBMS) like MySQL, Oracle, PostgreSQL, Microsoft SQL Server, and SQLite.
- ✓ Relational databases store data in structured tables with predefined schemas and support SQL for data querying and manipulation.
- ✓ NoSQL Databases: Non-relational or NoSQL databases like MongoDB, Cassandra, or Redis store data in flexible, non-tabular formats like key-value pairs, documents, or graph structures.
- ✓ They are suitable for handling unstructured or semi-structured data.

Relational and NoSQL Databases

- ✓ `import mysql.connector`
- ✓ `mydb=mysql.connector.connect(host='localhost', user='root', password='password_here')`
- ✓ `cursor=mydb.cursor()`
- ✓ `#execute query`
- ✓ `mycursor.execute("QUERY_HERE")`
- ✓ `#commiting confirms the changes to the database`
- ✓ `mydb.commit()`



The screenshot shows a SQL query result in a database management tool. The query is `select * from sys.allocation_units`. The results are displayed in a table with the following columns: `allocation_unit_id`, `type`, `type_desc`, `container_id`, `data_space_id`, `total_pages`, `used_pages`, and `data_pages`. The table contains 9 rows of data, with rows 85 and 86 highlighted in blue.

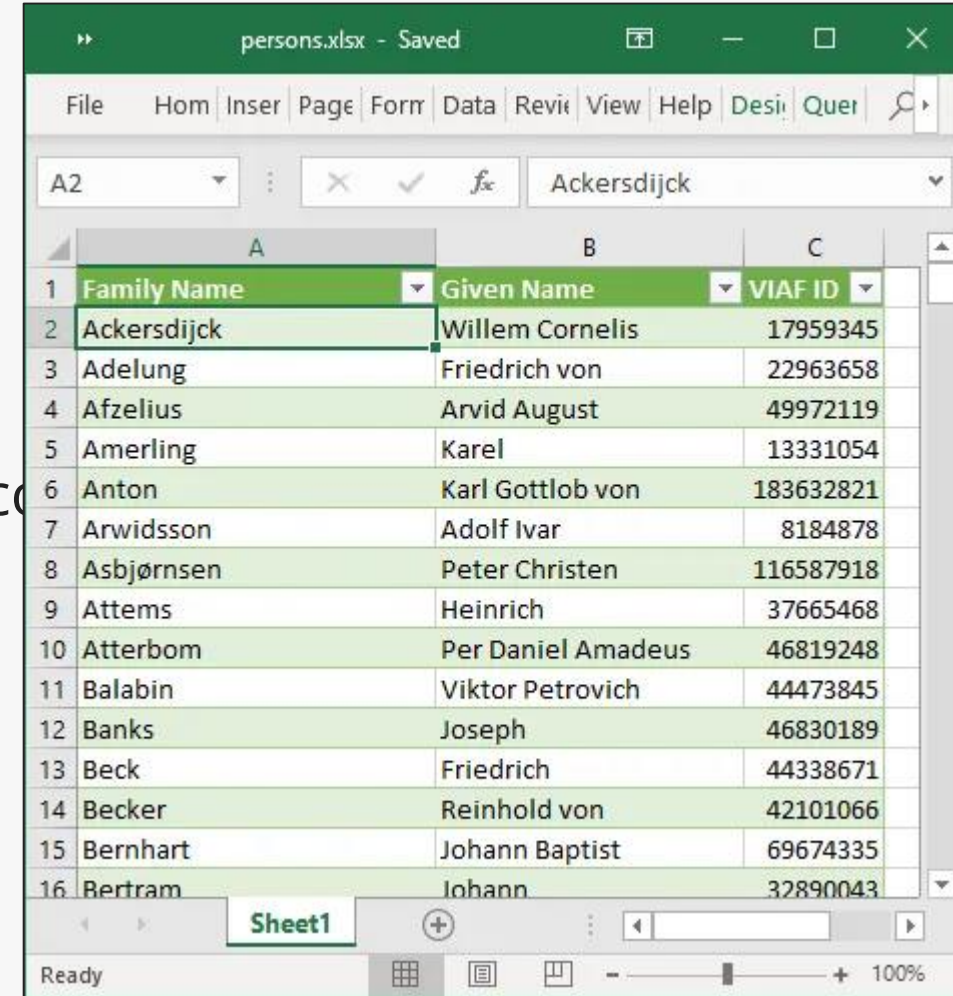
	allocation_unit_id	type	type_desc	container_id	data_space_id	total_pages	used_pages	data_pages
82	844424932884480	1	IN_ROW_DATA	844424932884480	1	0	0	0
83	844424933408768	1	IN_ROW_DATA	844424933408768	1	2	2	1
84	844424935768064	1	IN_ROW_DATA	844424935768064	1	0	0	0
85	1125899909070...	1	IN_ROW_DATA	1125899909070...	1	17	10	8
86	7177611906514...	2	LOB_DATA	281474980642816	1	12	6	0
87	7205759403799...	1	IN_ROW_DATA	281474980511744	1	2	2	1
88	7205759403805...	1	IN_ROW_DATA	562949957222400	1	2	2	1
89	7205759403819...	1	IN_ROW_DATA	281474983067648	1	2	2	1
90	7205759403825...	1	IN_ROW_DATA	562949959778304	1	2	2	1

CSV / Spreadsheets files

- ✓ One of the most common ways in which data is stored is in a CSV file.
- ✓ CSV file consists of data that are comma-separated. When opened in software like Excel, CSV displays like an excel sheet, where data is stored column-wise and row-wise.
- ✓ CSV files can be easily accessed and processed using programming language like Python.

CSV / Spreadsheets files

- ✓ import pandas as pd
- ✓ `data = pd.read_csv("file.csv")`
- ✓ `print(data)` #prints full dataframe
- ✓ `print(data['column_name'])` #prints a single column



The screenshot shows an Excel spreadsheet titled "persons.xlsx - Saved". The spreadsheet contains a table with three columns: "Family Name", "Given Name", and "VIAF ID". The data is as follows:

Family Name	Given Name	VIAF ID
Ackersdijck	Willem Cornelis	17959345
Adelung	Friedrich von	22963658
Afelius	Arvid August	49972119
Amerling	Karel	13331054
Anton	Karl Gottlob von	183632821
Arwidsson	Adolf Ivar	8184878
Asbjørnsen	Peter Christen	116587918
Attems	Heinrich	37665468
Atterbom	Per Daniel Amadeus	46819248
Balabin	Viktor Petrovich	44473845
Banks	Joseph	46830189
Beck	Friedrich	44338671
Becker	Reinhold von	42101066
Bernhart	Johann Baptist	69674335
Bertram	Johann	32890043

Cloud Data warehouses / Cloud Databases

- ✓ Data science often correlates with cloud platforms. With the ability to set up huge machines and elastic property, cloud computing is emerging globally and has great future potential. The major storage solutions offered by the cloud are data warehouses and cloud database.
- ✓ Although the functionality remains the same, warehouses are used to store large amounts of incoming data for analytics purposes, while cloud database stores the usual customer data in the cloud. Both of these can be accessed from an application using its respective APIs.
- ✓ GCP provides the google cloud API which requires you to send the credentials to connect to any service, while for AWS you can use pyodbc library to connect to any service using the connection string that the

Cloud Data warehouses / Cloud Databases

- ✓ Data science often correlates with cloud platforms. With the ability to set up huge machines and elastic property, cloud computing is emerging globally and has great future potential. The major storage solutions offered by the cloud are data warehouses and cloud database.
- ✓ Although the functionality remains the same, warehouses are used to store large amounts of incoming data for analytics purposes, while cloud database stores the usual customer data in the cloud. Both of these can be accessed from an application using its respective APIs.
- ✓ GCP provides the google cloud API which requires you to send the credentials to connect to any service, while for AWS you can use pyodbc library to connect to any service using the connection string that the

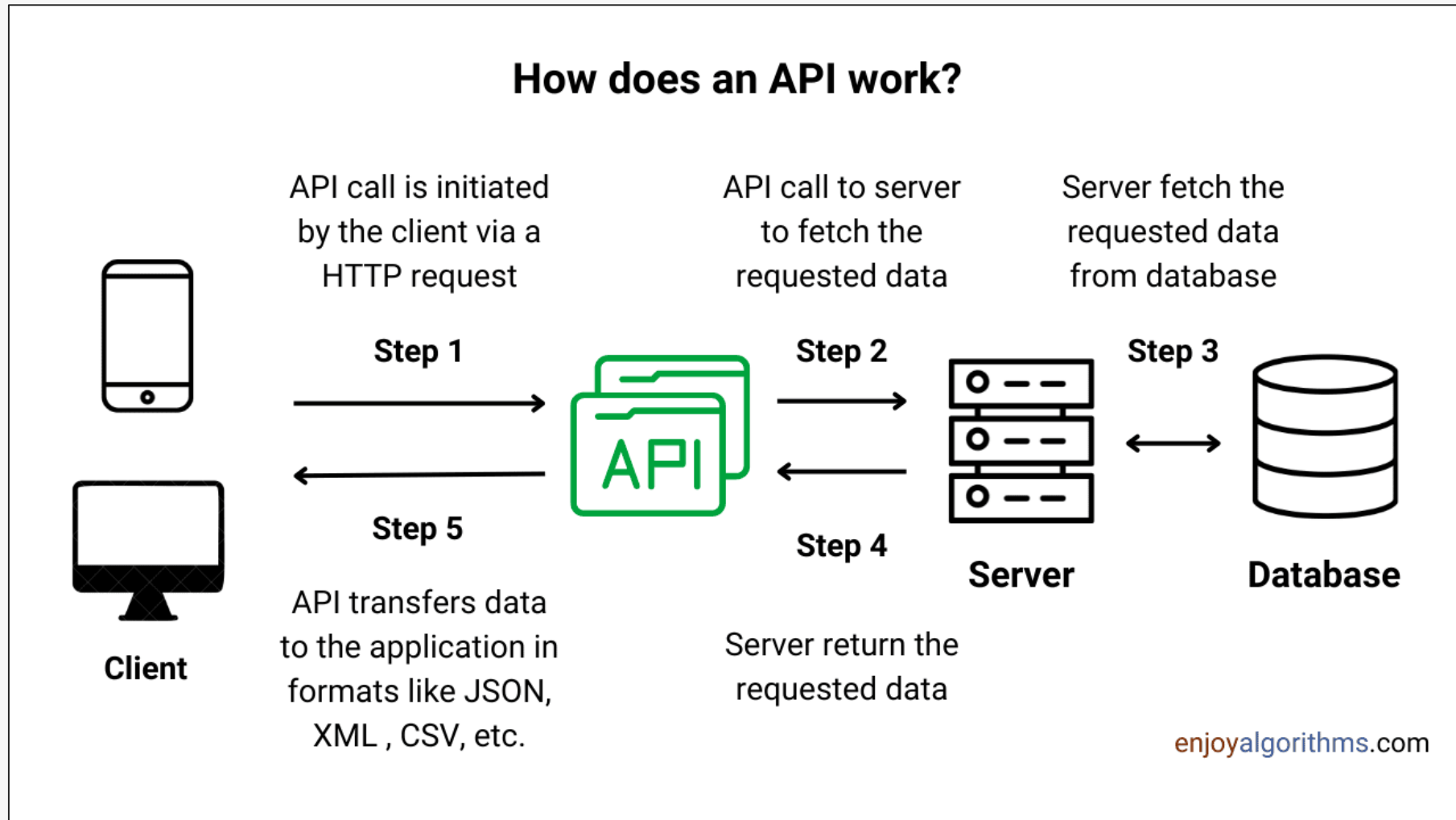
APIs (Application Programming Interfaces)

- ✓ API is a set of protocols, routines, tools, and standards that enable software applications to communicate with each other.
- ✓ API defines how software components should interact with each other and provides a way for developers to access the functionality of a particular application or service, without having to understand the underlying code.
- ✓ APIs are mostly used in web and mobile applications to integrate different software systems and enable the exchange of data between them.

APIs (Application Programming Interfaces)

- ✓ **Application** : It refers to the software, service, or code that a programmer wants to interact with or use in their own application.
- ✓ **Programming**: It is the protocol established between the application and the interface. For example, APIs following the SOAP protocol return data in XML format, whereas RESTful APIs can return data in many formats, most prominently in JSON format.
- ✓ **Interfaces**: The interface is the abstraction of implementation. While the User Interface (UI) is made for the users to interact with the application, APIs are made for application programmers to use in their applications. It provides a set of methods or functions that the application can call to perform specific actions or access specific data.

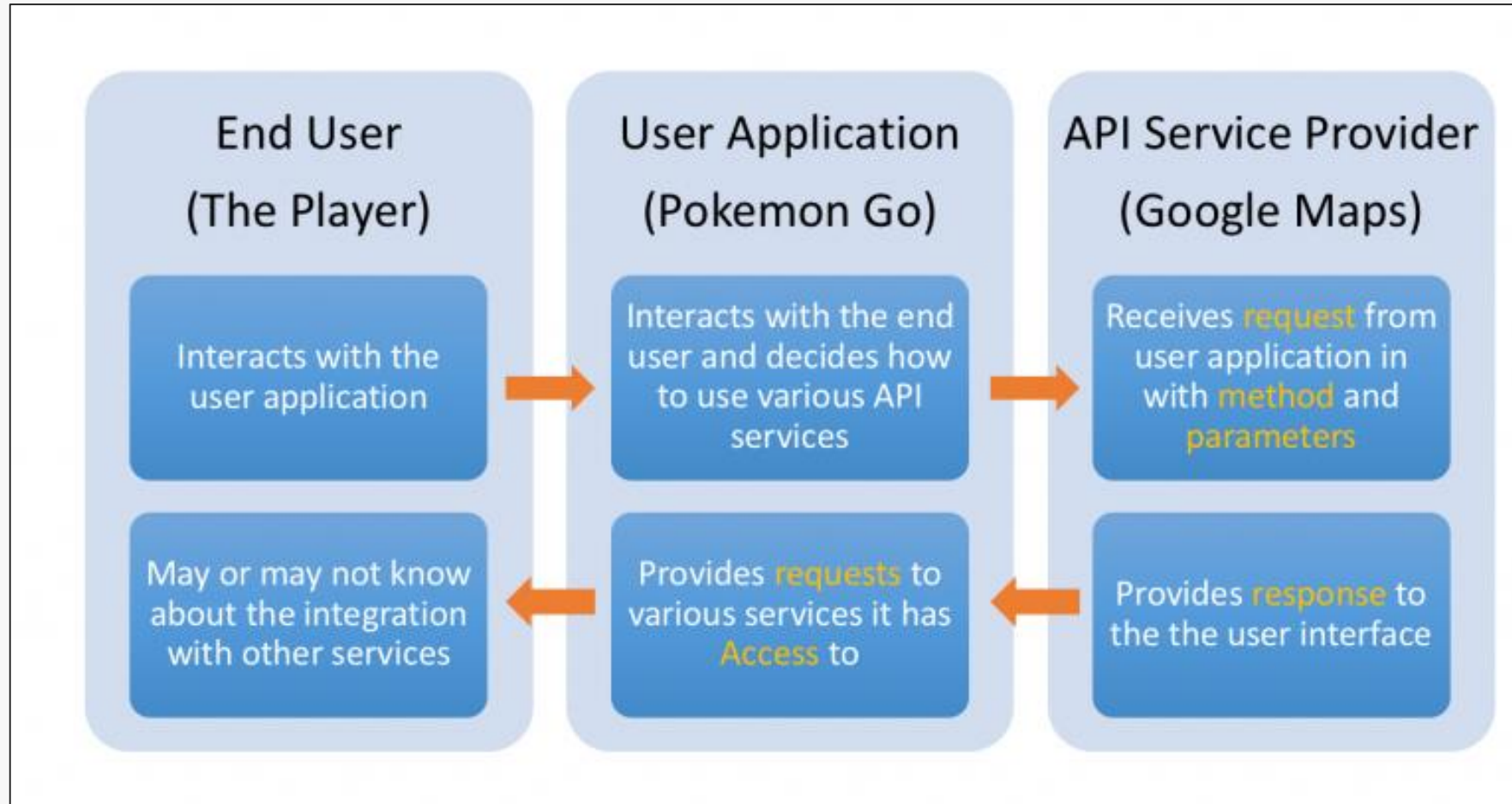
APIs : Mechanism



Elements of an API

- ✓ **Access:** is the user or who is allowed to ask for data or services?
- ✓ **Request:** is the actual data or service being asked for (e.g., if I give you current location from my game (Pokemon Go), tell me the map around that place). A Request has two main parts:
 - **Methods:** i.e. the questions you can ask, assuming you have access (it also defines the type of responses available).
 - **Parameters:** additional details you can include in the question or response.
- ✓ **Response:** the data or service as a result of your request.

Elements of an API



API Categories

- ✓ **Web-based system** : A web API is an interface to either a web server or a web browser. These APIs are used extensively for the development of web applications. These APIs work at either the server end or the client end. examples :Twitter REST API, Facebook Graph API, Amazon S3 REST API, etc.
- ✓ **Operating system** :There are multiple OS based API that offers the functionality of various OS features that can be incorporated in creating windows or mac applications.

examples of OS based API are Cocoa, Carbon, WinAPI, etc.

API Categories

- ✓ **Database system** : Interaction with most of the database is done using the API calls to the database. These APIs are defined in a manner to pass out the requested data in a predefined format that is understandable by the requesting client. This makes the process of interaction with databases generalised and thereby enhancing the compatibility of applications with the various database. examples :Drupal 8 Database API, Django API.
- ✓ **Hardware system** :These APIs allows access to the various hardware components of a system. They can establish communication to the hardware. Due to which it makes possible for a range of functions from the collection of sensor data to even display on your screens.

examples :QUANT Electronic, WareNet CheckWare, OpenVX .

API protocols

- ✓ **SOAP (Simple Object Access Protocol)**: Built with XML, SOAP enables endpoints to send and receive data through SMTP and HTTP. SOAP APIs make it easier to share information between apps running in different environments or languages.
- ✓ **XML-RPC (XML-Remote Procedure Call)**: The XML-RPC protocol relies on a specific XML format to transfer data. XML-RPC is older than SOAP, but much simpler, and relatively lightweight in that it uses minimum bandwidth.
- ✓ **JSON-RPC**: Like XML-RPC, JSON-RPC is a remote procedure call, but JSON (JavaScript Object Notation) is used instead of XML to transfer the data.
- ✓ **REST (Representational State Transfer)**: REST is a set of web API architecture principles. REST APIs—also known as a RESTful API—are APIs that adhere to certain REST architectural constraints. It's possible to build RESTful APIs with SOAP protocols, but the two standards are usually viewed as competing specifications.

API data formats

✓ JSON objects

- JSON is the most popular format for API calls. It consists of key-value pairs and arrays that look like the example below.
- One core advantage of using JSON is speed. Since it is compact and more parsable, it runs faster than XML, another popular data interchange standard prior to JSON. It's also very scalable and supports a larger number of data objects sent from the server. Though many would argue that it's not as secure as XML or CSV format.

API data formats

```
1  {
2    "string": "Hi",
3    "number": 2.5,
4    "boolean": true,
5    "null": null,
6    "object": { "name": "Kyle", "age": 24 },
7    "array": ["Hello", 5, false, null, { "key": "value", "number": 6 }],
8    "arrayOfObjects": [
9      { "name": "Jerry", "age": 28 },
10     { "name": "Sally", "age": 26 }
11  ]
12 }
13
```

API data formats

✓ CSV

- CSV is a data storage format that stores data values (plain text) in a list format separated by commas. It's actually more compact and even faster than JSON, but it's rarely used in a web development environment since it lacks a hierarchical structure.
- For what it lacks in scalability, it is more accessible due to its tabular format. A spreadsheet or relational database can easily access a CSV file while providing a variety of functional features in analytics and data manipulations

API data formats

✓ CSV

```
ticker,date,open,high,low,close,adj close,volume  
AAPL,2021-02-02,135.73,136.3,134.61,134.99,134.359,79426446  
AAPL,2021-02-01,133.75,135.38,130.932,134.14,133.513,104212352  
AAPL,2021-01-29,135.83,136.74,130.21,131.96,131.343,172209910  
AAPL,2021-01-28,139.52,141.99,136.7,137.09,136.449,137245542  
AAPL,2021-01-27,143.43,144.3,140.41,142.06,141.396,121162513
```

API data formats

✓ XML

- XML, as previously mentioned, was the data exchange format for API prior to JSON. It's a markup language that's both human and machine readable. Though it's not compact and optimized for read speed. It's considered verbose and redundant when compared to JSON.
- Though XML is more suited for combining information sets from different systems such as metadata.

API data formats

✓ XML

```
<note>  
  <to>Tove</to>  
  <from>Jani</from>  
  <heading>Reminder</heading>  
  <body>Don't forget me this weekend!</body>  
</note>
```

Popular API

- ✓ Facebook API
- ✓ Google Map API
- ✓ Twitter API
- ✓ IBM Watson API
- ✓ Quandl API

Web Scraping

- ✓ Web scraping is a technique to fetch data from websites. While surfing on the web, many websites don't allow the user to save data for personal use. One way is to manually copy-paste the data, which is both tedious and time-consuming. Web Scraping is the automation of the data extraction process from websites.
- ✓ This event is done with the help of web scraping software known as web scrapers. They automatically load and extract data from the websites based on user requirements. These can be custom built to work for one site or can be configured to work with any website.

Web Scraping Techniques

- ✓ **Manual Extraction Techniques:** Manually copy-pasting the site content comes under this technique. Though tedious, time taking and repetitive it is an effective way to scrap data from the sites having good anti-scraping measures like bot detection.
- ✓ **Automated Extraction Techniques:** Web scraping software is used to automatically extract data from sites based on user requirement.
 - HTML and DOM Parsing
 - Web Scraping Software

Web Scraping Steps

- ✓ Web scraping is the process of collecting data from websites using automatized scripts. It's used, of course, to gather large amounts of data that would be impossible to gather manually.
- ✓ It consists of three main steps:
 - 1—Fetch / Crawl the page ;
 - 2— Parse / Transform the HTML;
 - 3—Extract information and Store Data.

Web Scraping Tools : BeautifulSoup

- ✓ BeautifulSoup is a Python web scraping library that extracts data from HTML and XML files. It parses HTML and XML documents and generates a parse tree for web pages, making data extraction easy.
- ✓ BeautifulSoup's excellent support for encoding detection is a valuable feature that can yield better outputs for authentic HTML sites that do not fully disclose their encoding.
- ✓ BeautifulSoup is built on well-known Python parsers like lxml and html5lib, enabling us to experiment with various parsing techniques or trade off speed for flexibility.

Web Scraping Tools : BeautifulSoup

```
from bs4 import BeautifulSoup
import requests
import os, os.path, csv

listingurl = "http://www.espn.com/college-sports/football/recruiting/databaseresults/_/sportid/24/class/2006/sc

response = requests.get(listingurl)
soup = BeautifulSoup(response.text, "html.parser")

listings = []
for rows in soup.find_all("tr"):
    if ("oddrow" in rows["class"]) or ("evenrow" in rows["class"]):

        name = rows.find("div", class_="name").a.get_text()
        hometown = rows.find_all("td")[1].get_text()
        school = hometown[hometown.find(",")+4:]
        city = hometown[:hometown.find(",")+4]
        position = rows.find_all("td")[2].get_text()
        grade = rows.find_all("td")[4].get_text()

        listings.append([name, school, city, position, grade])

with open("footballers.csv", 'a', encoding='utf-8') as toWrite:
    writer = csv.writer(toWrite)
    writer.writerows(listings)

print("ESPN College Football listings fetched.")
```

Web Scraping Tools : BeautifulSoup

Pros	Cons
<ul style="list-style-type: none">• The library helps in maintaining the code's simplicity and adaptability.• offers a strong community to address all web scraping challenges for both new and experienced developers.• The primary benefit of using BeautifulSoup for developers is that it offers excellent and thorough documentation.	<ul style="list-style-type: none">• The use of proxies is not simple with BeautifulSoup. As a result, using BeautifulSoup to download vast volumes of data from the same site without having your IP blacklisted or banned is difficult.• BeautifulSoup can't function independently as a parser. It requires you to install dependencies before using it.

Web Scraping Tools : Scrapy

- ✓ Scrapy is one of the most popular Python web scraping libraries.
- ✓ Scrapy is a web crawling and screen scraping library to quickly and efficiently crawl websites and extract structured data from their pages.
- ✓ Scrapy can be used as a library, i.e., you can use it for various tasks, including monitoring, automated testing, and data mining.

Web Scraping Tools : Scrapy

- ✓ Scrapy is one of the most popular Python web scraping libraries.
- ✓ Scrapy is a web crawling and screen scraping library to quickly and efficiently crawl websites and extract structured data from their pages.
- ✓ Scrapy can be used as a library, i.e., you can use it for various tasks, including monitoring, automated testing, and data mining.

Web Scraping Tools : Scrapy

Pros	Cons
<ul style="list-style-type: none">• Scrapy's robust support for extensibility lets you add your features using signals and a simple API (middlewares, extensions, and pipelines).• Scrapy provides an interactive shell terminal that is IPython-aware and allows you to test out CSS and XPath expressions to scrape data when creating or debugging your spiders.• Scrapy provides strong encoding support and auto-detection feature for dealing with foreign, non-standard, and broken encoding declarations.	<ul style="list-style-type: none">• Scrapy does not work well with javaScript-based websites.• Various operating systems have different installation techniques for Scrapy.

Data Sources in

Data Science



General Resources

- ✓ Google Dataset Search. <https://datasetsearch.research.google.com/>
- ✓ Google Scholar. <https://scholar.google.com/>
- ✓ Papers With Code. <https://paperswithcode.com/>
- ✓ Kaggle. <https://www.kaggle.com/>
- ✓ Wikipedia's List of datasets for machine-learning research : https://en.wikipedia.org/wiki/List_of_datasets_for_machine-learning_research
- ✓ Rapid API Hub. <https://rapidapi.com/hub>

Datasets

- ✓ U.S. Government's open data (DATA.GOV). <https://data.gov/>
- ✓ United Kingdom Find Open Data. <https://www.data.gov.uk/>
- ✓ Open Government Data (OGD). <https://data.gov.in/>
- ✓ Microsoft Research Open Data. <https://www.microsoft.com/en-us/research/project/microsoft-research-open-data/>
- ✓ Kaggle (most popular): <https://www.kaggle.com/discussions/general/260690>
- ✓ Reddit. <https://www.reddit.com/r/datasets/>
- ✓ Awesome Public Datasets. <https://github.com/awesomedata/awesome-public-datasets>
- ✓ Data World. <https://data.world/>



Ministry of Higher Education and Scientific Research
Djalali BOUNAAMA University - Khemis Miliana (UDBKM)
Faculty of Science and Technology
Department of Mathematics and Computer Science



Chapter 4

Data Science Sources

AIBD-M1-UEM112 : Introduction to Data Science

Noureddine AZZOUZA

n.azzouza@univ-dbkm.dz