# Data resources

## Open data

► Critical data:

* GPS data on traffic flow
* Maps
* incidents and events
* weather

traffic can change with weather or other element.

► Challenge:

* collect different sources of data

Open data is organizations provide machine readable to support data science

* Publicly available
* Machine readable
* But it is not always usable and people need the right skills

Common format for open data is “Linked Open Data (LOD)”

* Enables data from different sources to be connected and queried

## API (Application Programmer Interface)

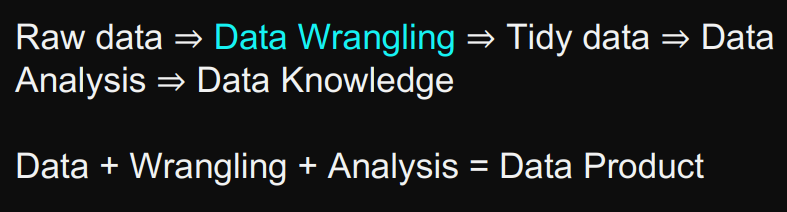
Routines providing programmatic access to an application

* like a user interface, designed for computers to access to the functionality of a software (e.g google map)
* Many companies are exposing their data and their website functionality as APIs

Data Wrangling

## Motivation

raw data comes in all shapes and sizes, make it clean and ready



* ► Data pre-processing
* ► Data preparation
* ► Data cleansing
* ► Data transformation

## Data quality problems

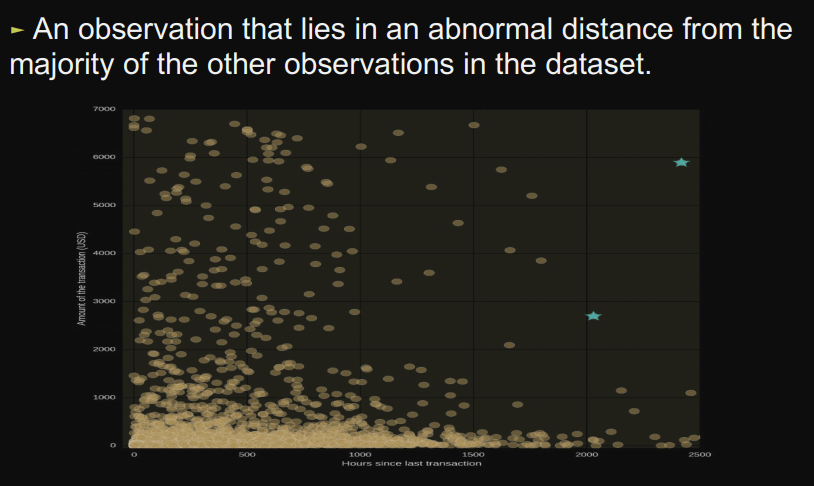
► Interpretability issue

► Data format issue

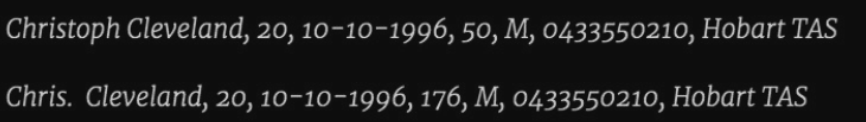
► Inconsistent and faulty data

► Missing and incomplete data

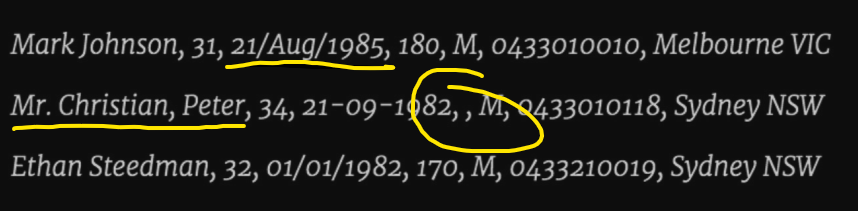
► Outliers



► Duplicates



E.g:



► mistyped data

► inconsistent entry

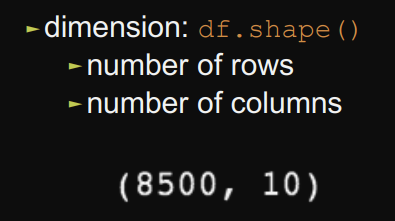
* common cases:
  + upper vs. lower case
  + inconsistency in domain value representation, e.g., 0 vs. No, 1 vs. Yes
* detecting and fixing
  + investigate unique domain values **(unique())**
  + make the representation consistent

►Misspelling

* investigate unique domain values **(unique())**
* string matching
  + calculate domain value frequencies **(value\_counts())**

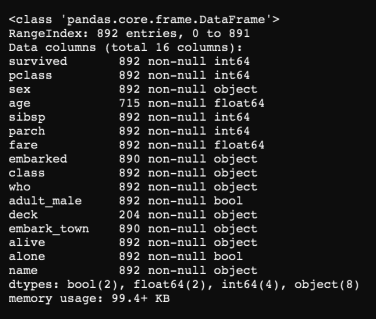
► extraneous (irrelevant or unrelated) data

## Data auditing in Python



►Head and tail rows: **df.head(); df.tail()**

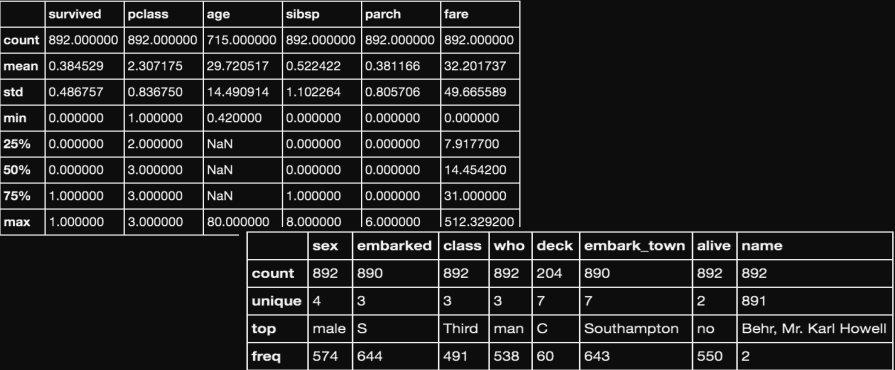
►Check basic information about the dataframe – number of records, whether there are null values, datatype: **df.info()**



►Check some basic statistics about columns:

► numerical columns: **df.describe()**

► object columns: **df.describe(include = [‘O’])**

****

►Check correlation amongst variables: **df.corr()**

