Data Management and Machine Learning in Automotive

Industrial Engineering for Advanced Automotive

Mauro Mangia – mauro.mangia@unibo.it

Who I am

- Assistant Professor at **University of Bologna** working in the **Statistical Signal Processing Research Group**
  - Research activities on algorithms and systems design for data/signal processing in different fields
    - automotive
    - railway
    - satellite
    - monitoring of civil structures
    - industrial plant
    - bio-signals
    - etc.

Why electronics and data processing?

- The idea to include data processing manipulation in a car is not new
- This is an analog computer that approximates fuel flow measurements based on airflow, which is proxy for understanding engine RPM, and throttle position. Unbelievably complex this can be adjusted via a number of screws, and changing interior parts known as "jets" that regulate the flow of fuel or air.

- While the engine shown is fairly modern, this dates back to the origin of the internal combustion engine in the 1870s.

Why electronics and data processing?

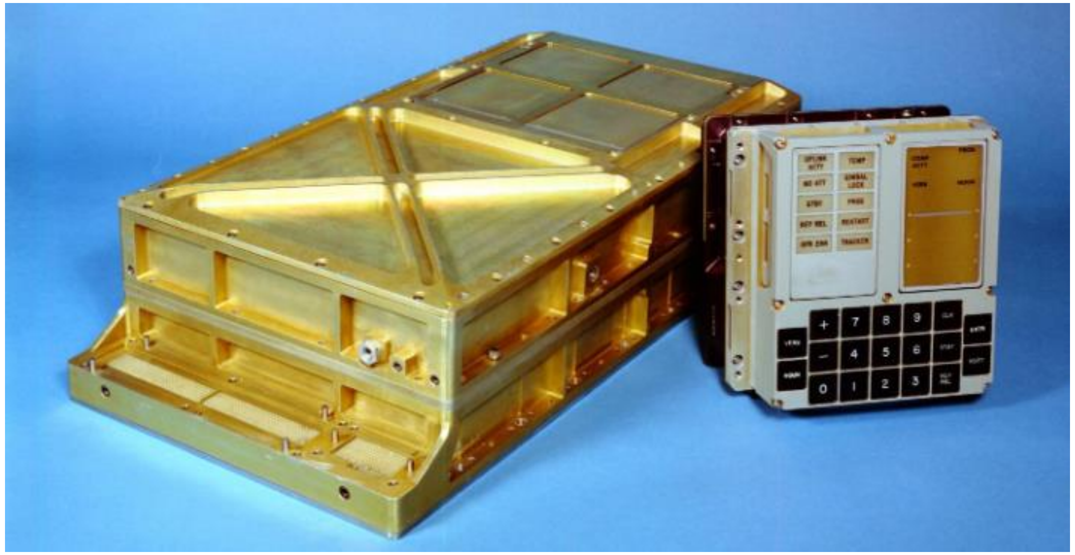
- That’s a purely electronic analog computer that takes a bunch of analog inputs for heat, speed, throttle position, fuel pressure, etc, compute when and how long to open fuel injectors. That’s a *Bosch D-Jetronic*, circa 1969, used through the 70s. Can be adjusted via diddling the variable resistors on the underside of the board.

Why electronics and data processing?

- This is a *Bosch DME ECU*(1984). This is a full digital computer, where the programming is carried in a re-programmable memory.
- This was watershed because it also integrated control of ignition and was the start of “*engine management*” - not just fuel control but ignition, making it possible to “manage” the entire combustion process.
- At the same time… in 1981, Toyota introduced the first production automatic transmission which was controlled by a microprocessor instead of purely by hydrostatic pressure in a recirculating valve body.

In the spacecraft domain …50 years ago

- The Apollo Guidance Computer (AGC)
  - ~2 MHz CPU clock frequency
  - 16 bit architecture
  - 3,840 bytes of main memory (RAM)
  - 69,120 bytes of non-volatile read-only memory (ROM)
  - All the running software was written in AGC assembly language [here available]



Today? ...Big Data Everywhere

VOLUME

Very (very) large amount of data
(orders of TB or PB)

VARIETY

different formats of data:

- Structured
- Semistructured
- Unstructured

VELOCITY

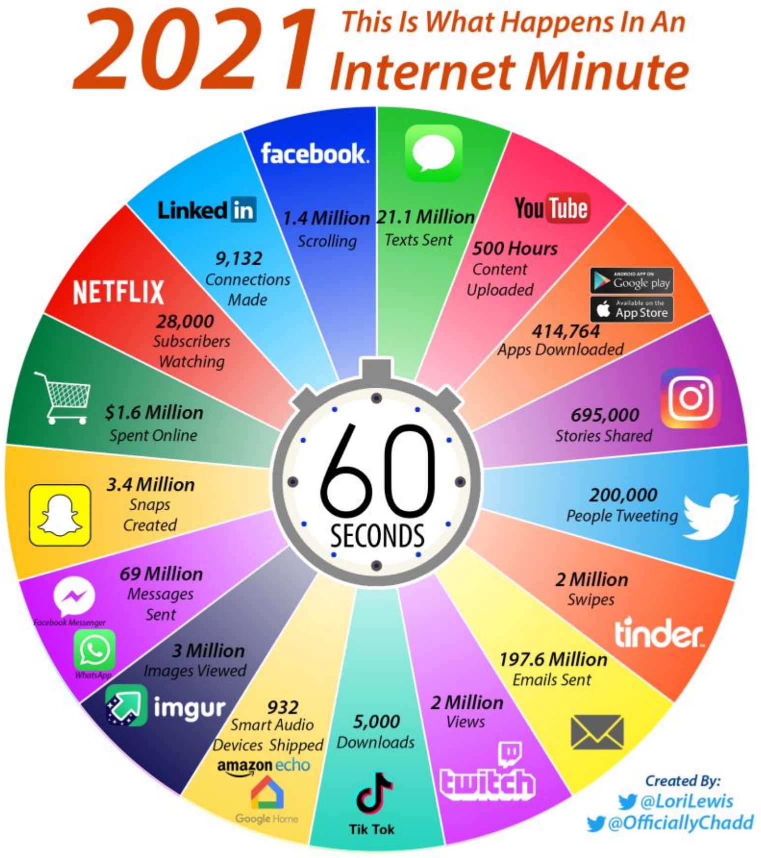
insane speed at which data is generated
(e.g., Twitter stream or Self-driving cars)

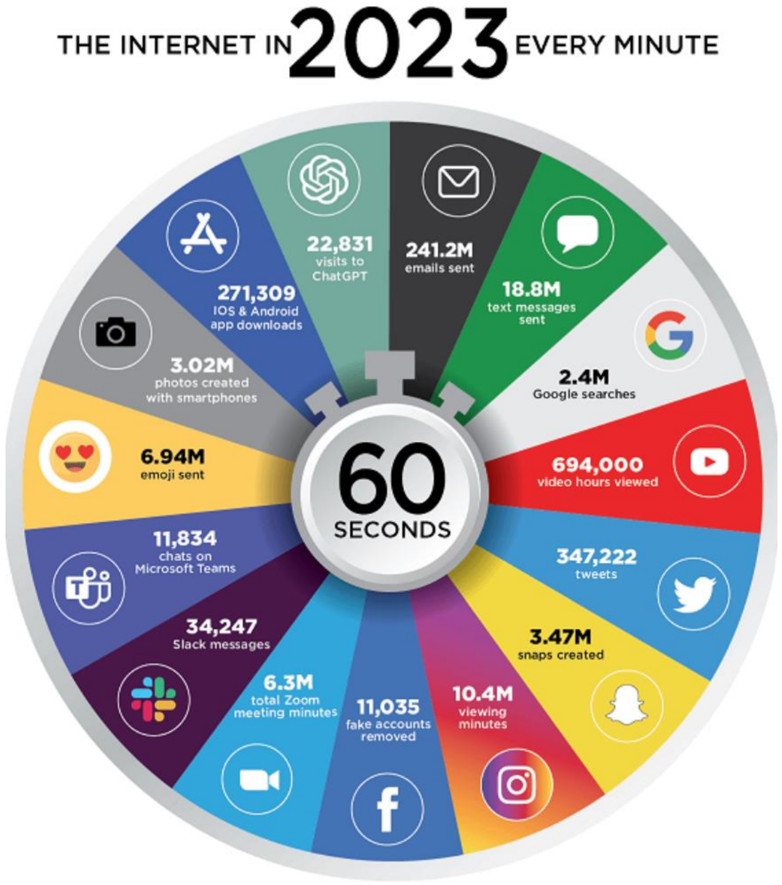
VERACITY

reliability of the data used to drive decision processes

How many data?

- Bid data due to many many many data sources



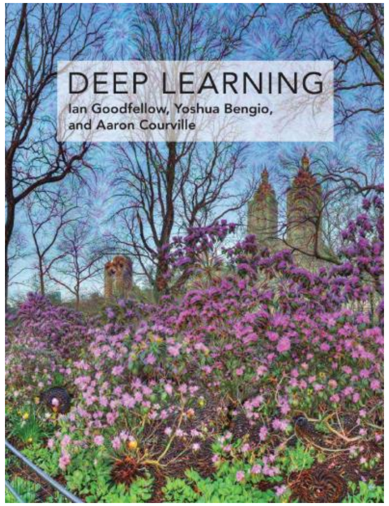


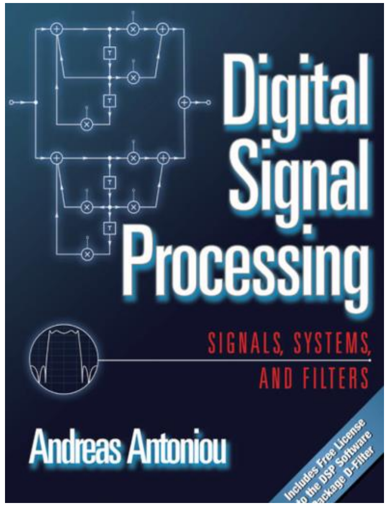
- 90 triaxial accelerometers (x, y, z) on selected external tendons, each generating 100 sample/sec/axis.

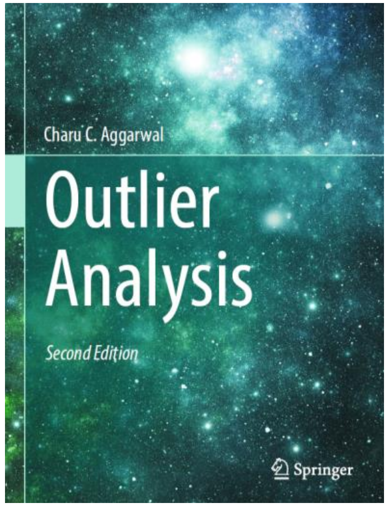
1568GB/year

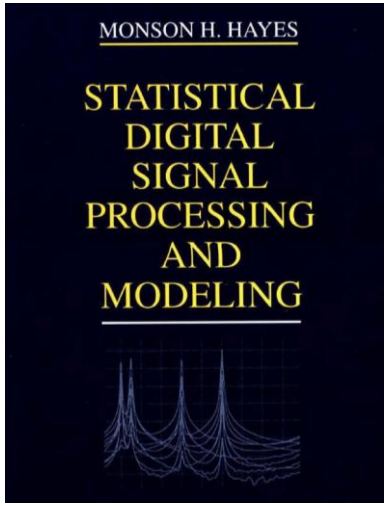
Suggested readings

- Digital Signal Processing: signals, systems and filters
  Andreas Antoniou, McGraw-Hill (2006)
- Outlier Analysis
  C. Aggarwal, Springer (2^{nd} ed. 2017)
- Deep Learning
  I. Goodfellow, Y. Bengio, A. Courville,
  MIT Press (2016)
- Statistical Digital Sygnal Processing and Modelling
  M. H. Hayes, John Wiley & Sons. (1996)




Digital Signal Processing
SIGNALS, SYSTEMS AND FILTERS
Andreas Antoniou
Includes Free License to the DSP Software Package D-Filter





Outline

- Machine Learning
- Python
- Hands on
    - Intro to Python
    - Data management
    - Machine learning example
    - Data visualization

Machine Learning

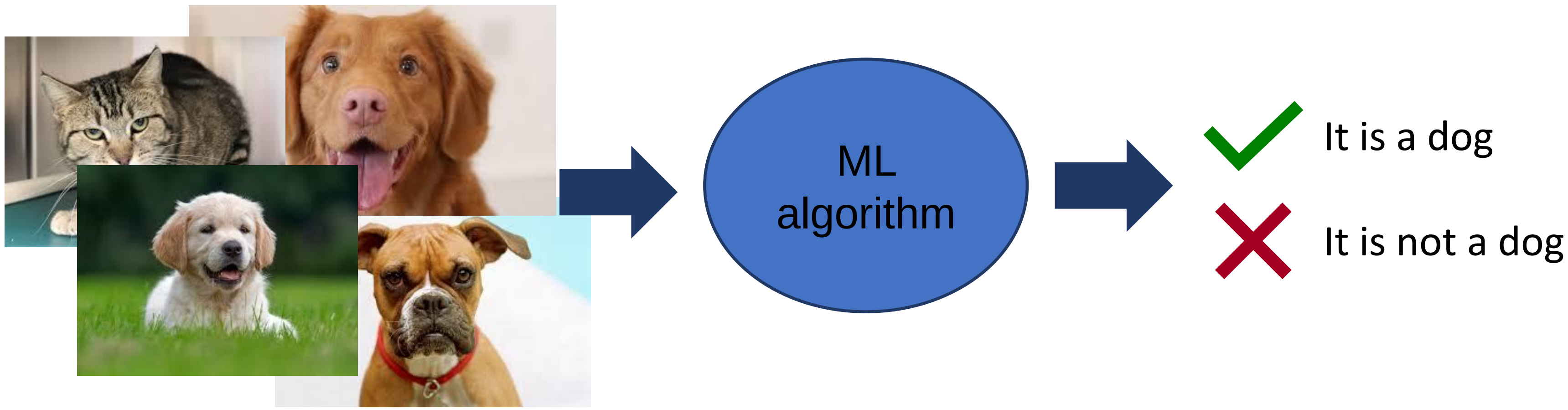
Machine Learning

A machine learning algorithm is an algorithm that is able to learn from data.

What do we mean by learning?

*“A computer program is said to learn from **experience** E with respect to some class of **tasks** T and **performance measure** P, if its performance at tasks in T , as measured by P , improves with experience E.”*

T. Mitchell, “Machine Learning”, McGraw-Hill, 1997



ML algorithm

✓ It is a dog

✗ It is not a dog

The Task

Nearly all task that a machine learning algorithm can perform fall into one of the two category:

- **Classification**: specifying which of k categories some input belongs to.
  - Object Recognition in images is one of the main example that find application in almost any field: medicine, industry, forensic, etc.
  - Anomaly Detection is about classifying an input normal or abnormal. A typical example is the credit card fraud detection.
- **Regression**: infer the relationship between input and output in order to predict a numerical value given some input.
  - It finds application in Finance forecasting, Trend Analysis, Marketing, Drug Response modeling, etc.
- **Synthesis and Sampling**: generate new examples that are similar to the ones that compose the machine’s experience:
  - Making a face younger or older, replicating the style of a painter, generating landscapes in video games, etc.

The Experience

For a Machine the Experience is the **dataset** it accesses to learn how to perform the task.

- In **Unsupervised Learning** the dataset is a collection of examples with no additional information.
    - The algorithm is often asked to learn the properties or the structure of the process that generated the dataset to perform tasks as density estimation, denoising, clustering or synthesis.
- In **Supervised Learning**, a label is associated to each example composing the dataset. The label represents the correct answer so that the machine can learn.
    - Classification based on Neural Networks is the typical example of supervised learning algorithm.

This distinction that seems quite straightforward is often very blurred as many methods can fall in both categories or none of them.

For example, **Reinforcement Learning** refers to algorithms that continuously learn by interacting with the environment.

The Performance Measure

In order to improve, the machine needs a **quantitative measure** to assess how well it is performing the task.

- It is often specific for the task. For instance, Accuracy is a common choice in classification, but it is not suited for regression.

In general, the dataset is split in two part:

- **Training set**: used to teach the model so it is the one that composes the experience of the machine.
- **Test set**: it is used to assess the model. For a fair assessment the performance must be measured on examples that are not part of the experience so that are not used during the training.

Sometimes, there is the necessity for a **validation set** used to tune some hyperparameter (parameter that cannot be trained with the model)

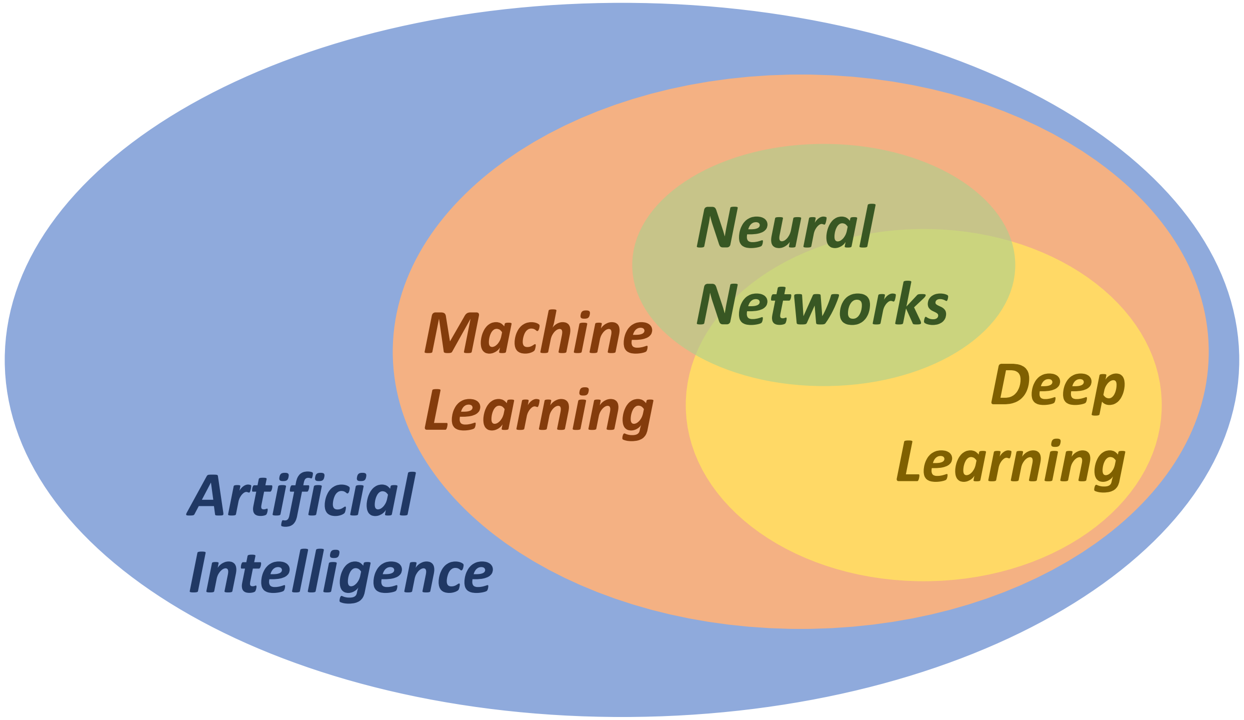
Terminology

- **Artificial Intelligence (AI)**
  - refer to systems that can perform tasks that typically require human intelligence. Often, AI requires to reverse-engineer human abilities and try to emulate it on machines that, thanks to the computational power, may outperform humans.
  - Examples: Image classification, Natural Language Processing, Self-driving cars, Web search engines, automatic decision-making, and so on.

Terminology

- **Neural Networks (NN)**
  - Specific class of ML algorithm often employed for tasks related to AI. The idea consists in miming the functioning of biological neural networks of animals brains.
- **Deep Learning (DL)**
  - Traditionally, data needed to be pre-processed (feature extraction, Fourier Transformation, etc.) before feeding a ML algorithm. Deep Learning refer ML models that are able to directly process data as it is generated.
  - Extremely more complex, typical example are the Convolutional Neural Network (CNN).

Terminology



ML in Automotive

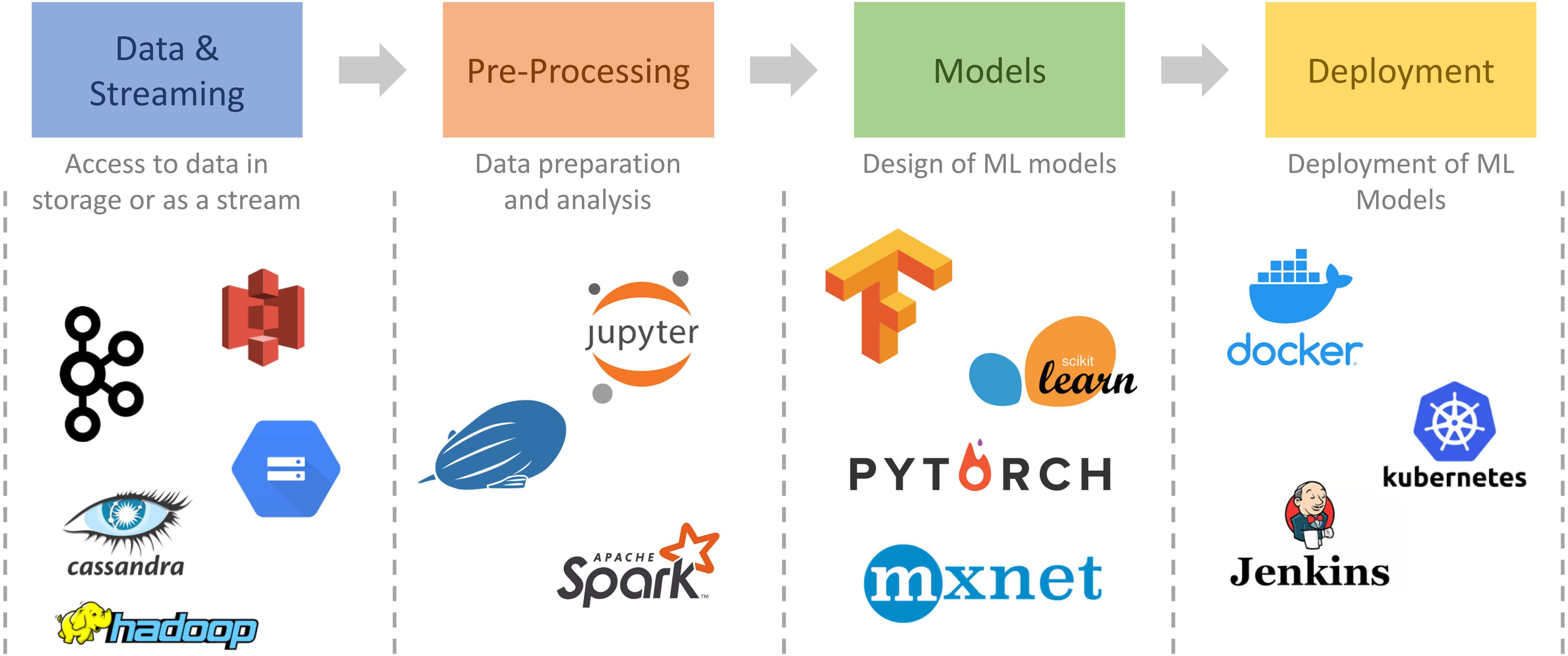
- Design
  - Acceleration of Rendering
  - Acceleration of FEM simulations
  - Generation of alternative solutions

- Manufacturing
  - Schedule/process optimization
  - Identification of defects in components
  - Plant Predictive maintenance

ML in Automotive

- Supply Chain
  - forecasting and replenishment optimization
  - Supplier ranking
- Quality Control
  - Automatic assessment of the quality of produced parts and ensembles
  - Automatic defect detection, classification and prediction
- Driver Behavior
  - Advanced Driver Assistance Systems (ADAS)
  - Driver Attention Alert

Machine Learning Pipeline



Data & Streaming

Access to data in storage or as a stream

Pre-Processing

Data preparation and analysis

Models

Design of ML models

Deployment

Deployment of ML Models

Python

Python

Python is an Interpreted and Object-Oriented Programming Language.

WHY Python?

- Simple syntax
- Very flexible
- Highly extensible
- Cross-platform
- Open-source with a huge community

Google says: *Python where we can, C++ where we must*

Applications

- Web and Internet Development
- Scientific and Numeric
- Education
- Desktop GUIs
- Software Development
- Business Applications

Basically anything, like English for spoken languages

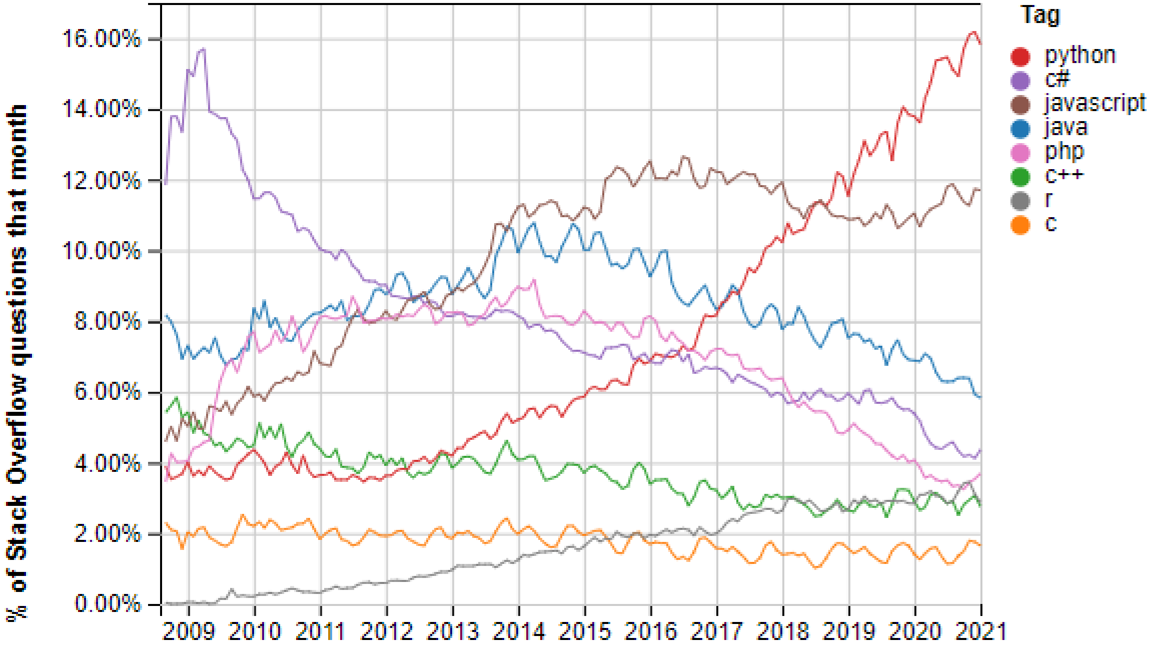
Popularity

- According to StackOverflow's survey and trends Python among all programming languages is the:
  - **1st most questioned**
  - **4th most used**

    behind JavaScript, HTML/CSS, and SQL
  - **2nd most loved**

    behind Rust
  - **1st most wanted**

    developers who do not yet use it say they want to learn it



Framework set-up

Installing Conda

- Download the latest version of **miniconda3** from here
  https://docs.conda.io/en/latest/miniconda.html

- Install it by following the instructions in here
  https://conda.io/projects/conda/en/latest/user-guide/install/index.html
  where you can find a step by step guide for any platform: Windows, Linux or macOS.

Opening conda

- On either Windows or macOS, find the **Anaconda Prompt** in the list of your programs and open it. On Windows you may run it as administrator.
- On Linux, open a **Terminal** and run the following command:

  ```
  $ conda activate base
  ```

After that, we are in conda and (hopefully) there will no difference between any platform.

Installing Python libraries

- Now we are in the conda base environment we can install some packages with the following commands:

```
$ conda install scipy
$ conda install matplotlib
$ conda install pandas
$ conda install seaborn
$ conda install –c conda-forge folium
$ conda install jupyterlab
$ conda install ipympl
```

Open Jupyter Lab

- Once everything is installed we can run Jupyter Lab:

  ```
  $ jupyter lab
  ```

Let’s start!