



INSTITUT NATIONAL
DES SCIENCES
APPLIQUÉES
TOULOUSE

IA FRAMEWORKS

INTRODUCTION & SPARK

Philippe Besse & Brendan Guillouet

November 4th, 2019

Institut National des Sciences Appliquées de Toulouse

INTRODUCTION

TABLE OF CONTENTS

Introduction

Definition & Context

Organization & Evaluation

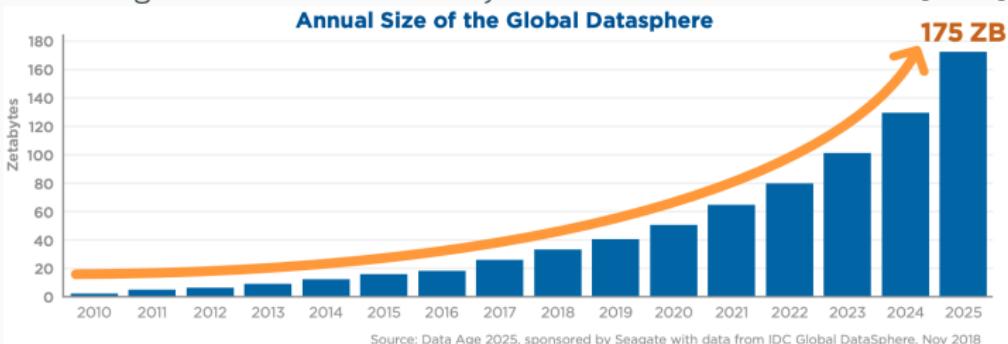
Hadoop, MapReduce, Spark

INTRODUCTION

DEFINITION & CONTEXT

5 YEARS AGO : **BIG DATA** and its 3 Vs :

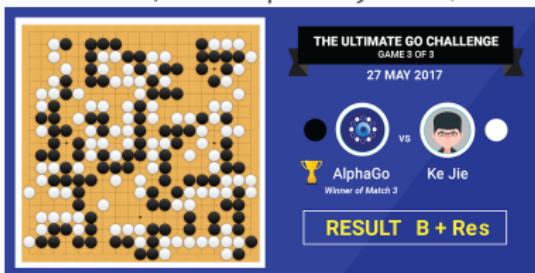
- **VOLUME** : Exponential growth.
 - According to IDC : from 33 zeta byte in 2018 to 175 in 2025. IDC [2018]



- **VARIETY** : numeric, signal, images, time series, graph, text etc...
- **VELOCITY** : real time decision.
- **(VALORISATION)**.

Now ARTIFICIAL INTELLIGENCE (again)

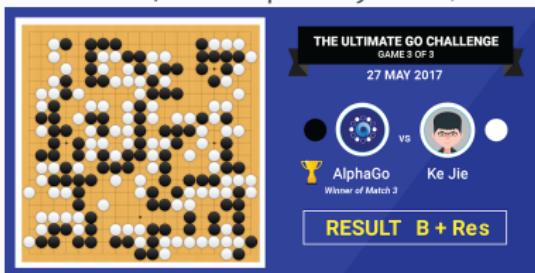
- BIG DATA,
- COMPUTATION POWER AVAILABILITY :GPU, cloud computing,
- 'NEW' ALGORITHMS :Deep Learning, Reinforcement learning,
- that produce impressive results, (Alpha Go, Atari Game defeated, write poetry etc..).



- .. with the help of some marketing.

Now **ARTIFICIAL INTELLIGENCE** (again)

- **BIG DATA,**
- **COMPUTATION POWER AVAILABILITY** :GPU, cloud computing,
- **'NEW' ALGORITHMS** :Deep Learning, Reinforcement learning,
- that produce impressive results, (Alpha Go, Atari Game defeated, write poetry etc..).



- .. with the help of some marketing.

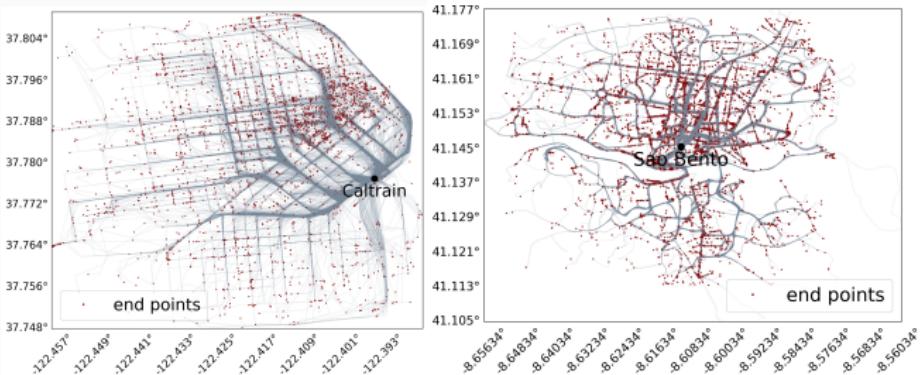
The definition is still very unclear, but IA is still NOT here.

A WIDE RANGE OF APPLICATION FIELDS...

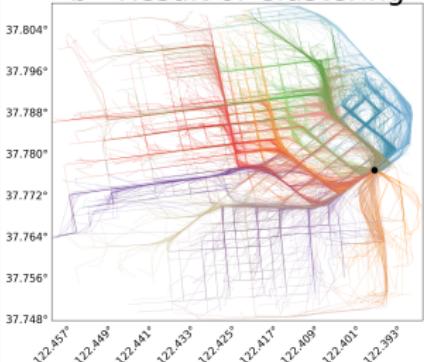
..at different level of advancement.

- **E-COMMERCE** : Recommendation system, advertising, targeting.
- **TEXT TRANSLATION** : DeepL , Google Translate.
- **IMAGE & VIDEO** : Classification, videosurveillance etc..
- **GAME** : Chess, Go, Atari Game, StarCraft.
- **PUBLIC** : administrations, e-health (*open data*)
- **INDUSTRY** : 4.0 industry, Autonomous vehicle.
- **METEOROLOGY** Hybridation of learning and physical model
- **CERTIFICATION**
- **INTERNET OF THINGS**

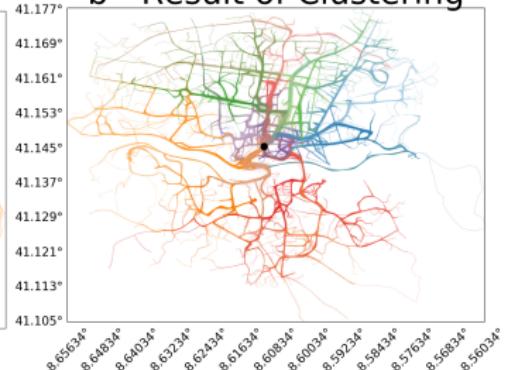
VARIETY : TRAJECTORY CLUSTERING FROM GPS DATA.



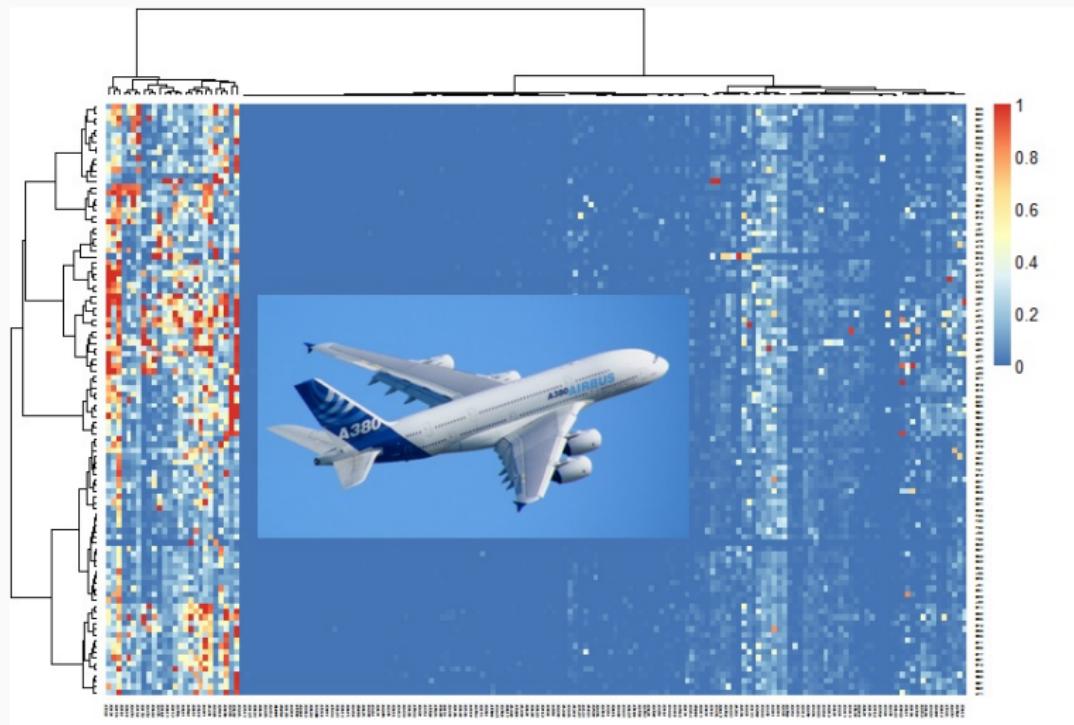
b - Result of Clustering



b - Result of Clustering



ANALYSIS OF FLIGHT INCIDENT MESSAGES



- Detection of field (agriculture) and batiment (military) from satellite images. *SSII & Startup*
- Anonymization of name on text documents. *SSII*
- Fall detection of elder people. *PME*
- Autonomous drone flight. *SSII*
- Rain prediction in Ivory Coast. *Startup*
- Detection of diabetic retinopathy from x-ray photo. *Startup-Norway*
- Estimation of fish results. *Research Center*

WHAT DO WE NEED TO START?

- Local or cloud computing? (*GCP, AWS, AZURE*)
- Size of disk? Size of RAM? CPU or GPU? How many of its?
- From where and how did you get the data : local/bucket?
- How are they stored ? Csv, Data base ? Which one ?
- Distributed architecture ? (*Hadoop, Spark*)
- In which format do you expect the results ?
- How it is automate ?



DataFarming

A NEW ECONOMIC MODEL

SITUATION :

- Algorithm are available and free (python library, github etc..)
- Most used Software (Tensorflow, Spark,...) are under free licences (GNU, MIT, Apache)
- Margin on hardware are not really valuable

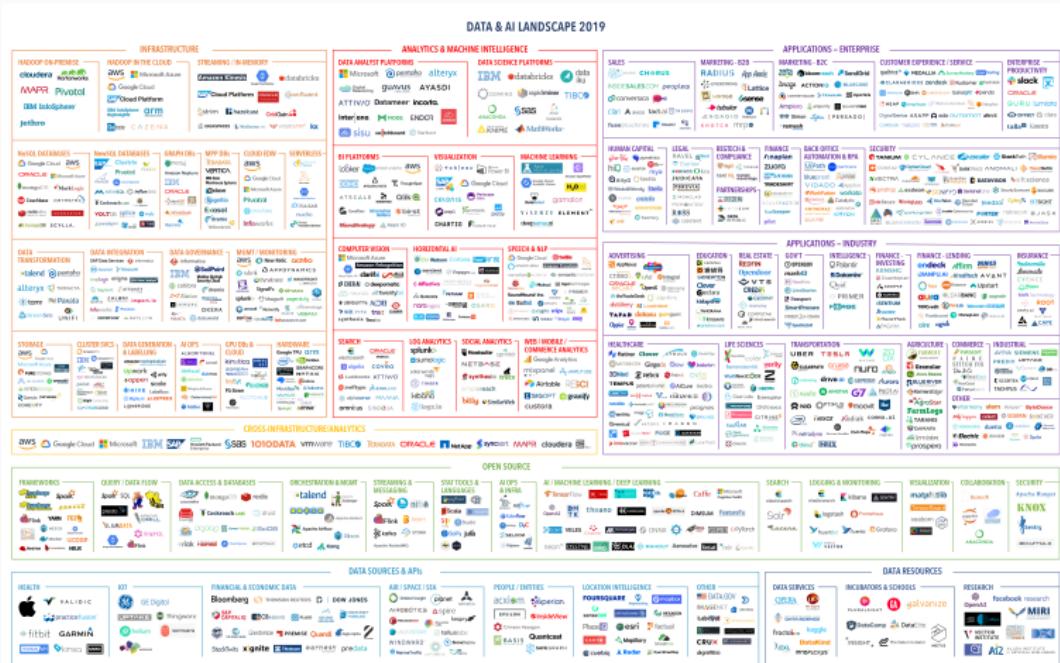
NEW STRATEGY : SELL SERVICES

- Enthought (Canopy), Continuum analytics (Anaconda), Horton Works, Cloudera (Hadoop, Spark...), Databricks (Spark), Oxdata (H2O), Revolution Analytics (RHadoop) – Microsoft

NEW MARKET : CLOUD COMPUTING (PICK-AND-SHOVEL PLAY)

- Actors : Amazon Web Service Platform, Microsoft Azure, Google Cloud Computing **aaaS**, Software **aaaS**, Service **aaaS**...
- A huge variety of services, Storage, Instances, ML Pipeline, DataFlow, etc....

BIG DATA & AI LANDSCAPE (TURCK, 2019)



July 16, 2019 - FINRA 2019 VERSION

© Matt Turck (@mattturck), Lisa Xu (@lisaxu92), & FirstMark (@firstmarkcap). mattturck.com/data2019

JOB SPECIFICATION

TASK

- Managing data Warehousing
- Ensure Data accessibility & consumption via API.
- Continuously monitoring and testing the system to ensure optimized performance.
- Extraction Transformation & Load
- Cleaning and organizing raw data.
- Programming skills.
- Using descriptive statistics to get a big-picture view of their data.
- Analyzing interesting trends found in the data.
- Creating visualizations and dashboards to help the company interpret and make decisions with the data.
- Presenting the results of a technical analysis to business clients or internal teams.
- Evaluating statistical models to determine the validity of analyses.
- Using machine learning to build better predictive algorithms.
- Testing and continuously improving the accuracy of machine learning models.
- Continuously Perform scientific Watch.
- Building data visualizations to summarize the conclusion of an advanced analysis.

JOB SPECIFICATION

TASK	DATA ENGINEER	DATA ANALYST	DATA SCIENTIST
• Managing data Warehousing	✓✓	✗	✗
• Ensure Data accessibility & consumption via API.	✓✓	✗	✗
• Continuously monitoring and testing the system to ensure optimized performance.	✓✓	✗	✗
• Extraction Transformation & Load	✓✓	✓	✓
• Cleaning and organizing raw data.	✓	✓	✓
• Programming skills.	✓	✓	✓
• Using descriptive statistics to get a big-picture view of their data.	✗	✓✓	✓
• Analyzing interesting trends found in the data.	✗	✓✓	✓
• Creating visualizations and dashboards to help the company interpret and make decisions with the data.	✗	✓✓	✓
• Presenting the results of a technical analysis to business clients or internal teams.	✗	✓✓	✓
• Evaluating statistical models to determine the validity of analyses.	✗	✓	✓✓
• Using machine learning to build better predictive algorithms.	✗	✓	✓✓
• Testing and continuously improving the accuracy of machine learning models.	✗	✗	✓✓
• Continuously Perform scientific Watch.	✗	✗	✓✓
• Building data visualizations to summarize the conclusion of an advanced analysis.	✗	✗	✓✓

JOB SPECIFICATION

TASK	You're Here ↓		
	DATA ENGINEER	DATA ANALYST	DATA SCIENTIST
• Managing data Warehousing	✓✓	✗	✗
• Ensure Data accessibility & consumption via API.	✓✓	✗	✗
• Continuously monitoring and testing the system to ensure optimized performance.	✓✓	✗	✗
• Extraction Transformation & Load	✓✓	✓	✓
• Cleaning and organizing raw data.	✓	✓	✓
• Programming skills.	✓	✓	✓
• Using descriptive statistics to get a big-picture view of their data.	✗	✓✓	✓
• Analyzing interesting trends found in the data.	✗	✓✓	✓
• Creating visualizations and dashboards to help the company interpret and make decisions with the data.	✗	✓✓	✓
• Presenting the results of a technical analysis to business clients or internal teams.	✗	✓✓	✓
• Evaluating statistical models to determine the validity of analyses.	✗	✓	✓✓
• Using machine learning to build better predictive algorithms.	✗	✓	✓✓
• Testing and continuously improving the accuracy of machine learning models.	✗	✗	✓✓
• Continuously Perform scientific Watch.	✗	✗	✓✓
• Building data visualizations to summarize the conclusion of an advanced analysis.	✗	✗	✓✓

NEW (?) DATA SCIENCE

As a **STATISTICIAN**.

- Deep comprehension of algorithms
- Scientific watch

What's really new for a **DATA SCIENTIST**.

- Programming skills.
- Better understanding of all data-pipeline technology.
- Manage bigger data.
- New questions about ethics.

⇒ Learn to "self-learn".

INTRODUCTION

ORGANIZATION & EVALUATION

OBJECTIVES - TOOLS

Tools you know

ML Python
Libraries



Viz' Python
Libraries



Python
Environment



Other Tools &
Frameworks



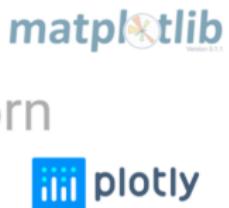
OBJECTIVES - TOOLS

Tools you'll discover

ML Python
Libraries



Viz' Python
Libraries



Python
Environment

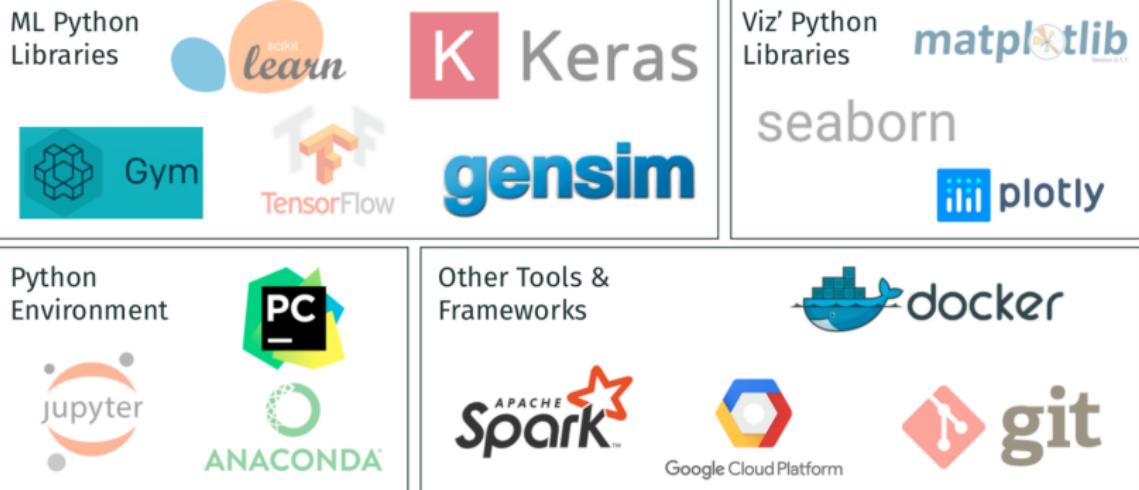


Other Tools &
Frameworks



OBJECTIVES - TOOLS

Tools you'll discover



Question :

- What are the best tools to used? In which circumstances?
- Is it valuable to use CPU/GPU? Own laptop/cloud resources?
Distributed architecture?

OBJECTIVES - DATA & ALGORITHM

DATA

- Structured Data
- Time Series & Signal
- Image

ALGORITHM

- Machine Learning : *SVM, RF, Regression.*
- Deep Learning : *MLP,CNN.*

OBJECTIVES - DATA & ALGORITHM

DATA

- Structured Data
- Time Series & Signal
- Image

ALGORITHM

- Machine Learning : SVM, RF,
Regression.
- Deep Learning : MLP,CNN.

- Text (*Cdiscount*)
- Video Game (*PacMan-like*)
- Note Data (*MovieLense*)

- Text cleaning : *TFIDF, stemming.*
- More Deep Learning : RNN,
Words Embedding.
- Reinforcement learning :
Policy Gradient / Q-learning.
- Recommendation System :
NMF.

FIVE LABS

1. Python complements. Introduction to **Spark**.
2. Cloud computing (**GCP**) and 'containerization' (**Docker**).
3. Reinforcement Learning. (**AI Gym**).
4. NLP on *Cdiscount*. (**Gensim**)
5. Recommendation system on *Movielens* with **Spark/MLLib**.

EVALUATION

⇒ ACT LIKE A DATA SCIENTIST :

1. Select a new algorithm you want to understand.
2. Select the data and the infrastructure on which you want to make it run.
3. Make it run.
4. Share it in order to make it easily reproducible for everyone.

⇒ TWO PARTS EVALUATION :

- A Github repository where you share your code.
 - How clear is your code ?
 - How easy it is to reproduce the results, following your instruction.
 - **Deadline** : January 11th, 2019.
- Oral Presentation.
 - In-Deep explanation of the chosen algorithm.
 - Choice of the tools /infrastructure used.
 - Difficulty you've met.
 - **Date** : January 14th, 2019.

EVALUATION

⇒ SOME PROPOSITIONS OF SUBJECT

- Generative Adversial Network.
- Object segmentation & Localisation
- Siamese Network.
- One-Shot Learning.
- Recurrent Neural Network (Text generation)
- Words Embedding Comparizon. (Glove/Fasttestx/Bert/Elmo)
- Words Embedding to handle small learning dataset.
- RL : Actor Critic.
- RL : Application on real data (Pacman).
- Google CLoud : Comparizon of AutoML with other services.

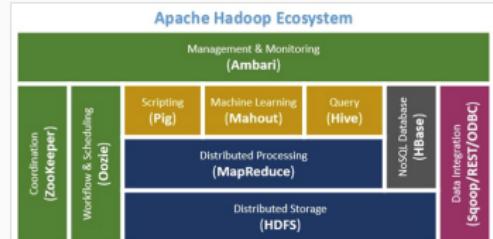
⇒ YOU'RE FREE TO CHOOSE THE INFRASTRUCTURE YOU WANT.

⇒ GROUP OF 4 TO 5.

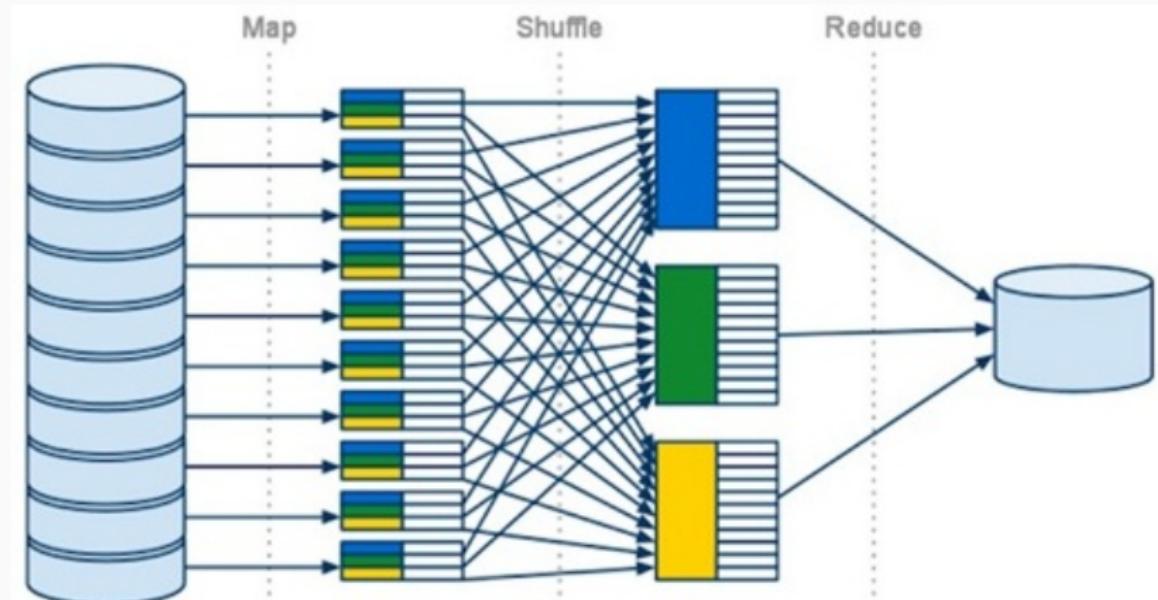
HADOOP, MAPREDUCE, SPARK

HADOOP, MAPREDUCE

- Environment : *Google* and now *Apache* (2009)
- *Hadoop Distributed File System* (HDFS)
- Heterogeneous distributed data
- Hardware fault tolerance
- Scalable (node multiplication)
- Parallelization : *Map Reduce*
- strictly Disk-Based
- (*Key, Value*) Communication
- Move the algorithm, not the data
- *Immutable* data
- *Streaming*



HADOOP, MAPREDUCE



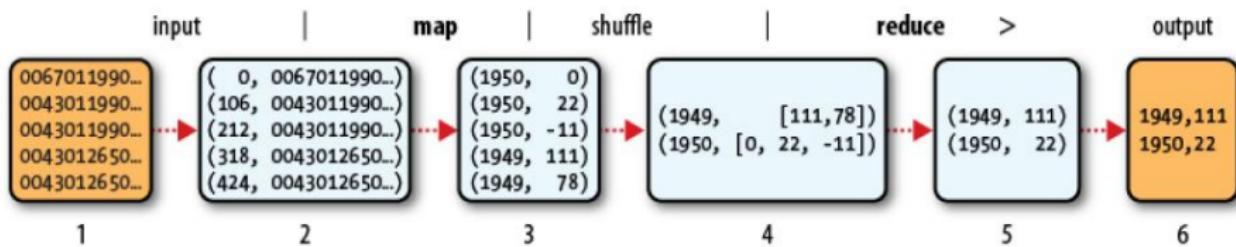
Hadoop Distributed File System (HDFS) & MapReduce

MAPREDUCE

ALGORITHM :

1. Input \Rightarrow list (k1, v1)
2. Map() \Rightarrow list (k2, v2)
3. *Shuffle, Combine* \Rightarrow (k2, list(v2))
 - Implicit
 - Identical key to same reduce node.
4. Reduce() \Rightarrow list (k3, v3)

EXAMPLE : Max per year.



MOBILE CENTER CLASSIFICATION ($\approx k$ -means)

- Definition of an euclidean **distance**
- Forgy's algorithm (1965)
 - Initialization of k centers
 - *MapReduce*'s steps **Itération**
 - Map : Affectation of each individual (**value**) to it's closest center (**key**).
 - Reduce : Compute means of individuals with same keys.
 - Update of the k centers
- **PROBLEM** : disk access to each iteration
- **SOLUTION** *Spark's Resilient Distributed Dataset*, (Zaharia et al., 2012)

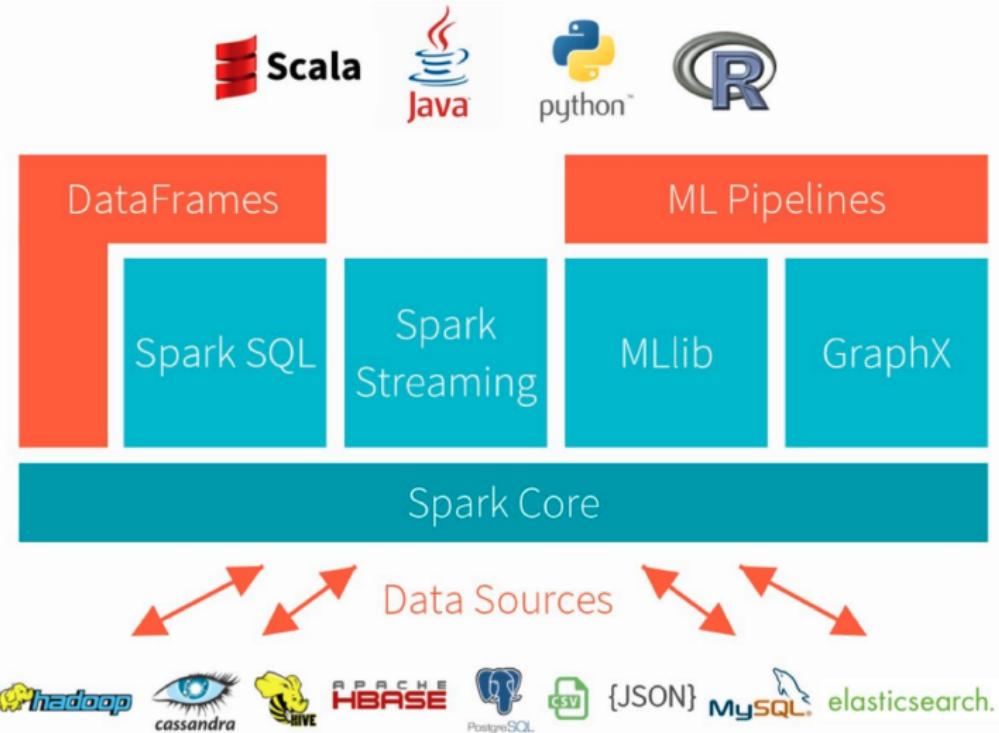
Spark's RDD supports in-memory processing computation

- Stores the state of memory as an object across the jobs.
- Data sharing in memory is 10 to 100 times faster than Disk.

Characteristics

- RDDs can contain any type of Python, Java, or Scala objects, including user-defined classes.
- Handle to type of operation :
 - Transformation allow to create a new RDD from an existing one.
 - Action apply computation on the RDD and return a value.
- Transformation are lazy, while Action aren't.

SPARK'S TECHNOLOGY AND ITS ECOSYSTEM



RDD vs DATAFRAME

Limit of RDD performances :

- Do not handle structure data.
 - Make not difference between and RDD of list and a RDD of dictionnaries.
- Do not handle easy access of part of the data (by columns name) etc..
- Lead to non-optimized performances.

DataFrame

- Based on Spark SQL API.
- Benefit all RDDs performances.
- Structured data (Columns, named etc..)
- Enable optimization and better computation performances.

Spark 2.4

- MLLib
 - RDD Based library.
 - Available but deprecated.
- SparkML
 - DataFrame Based library.
 - Official library (Not as furnished as MLLib).
 - Pipeline object.

Launch the notebook with pyspark :

```
pyspark_notebook.sh
```

Four notebooks :

1. (Optional Python notebooks.
2. Basic notions & RDD.
3. MLlib library.
4. Spark SQL & DataFrame.
5. SparkML & Pipeline.

REFERENCES I

RÉFÉRENCES

IDC. The digitization of the world. <https://www.seagate.com/files/www-content/our-story/trends/files/idc-seagate-dataage-whitepaper.pdf>, 2018.