Version: Sunday, October 1, 2023

Introduction to Machine Learning for Chemists: Visualization, Data Processing, Analysis

Romuald Poteau
Stella Christodoulou

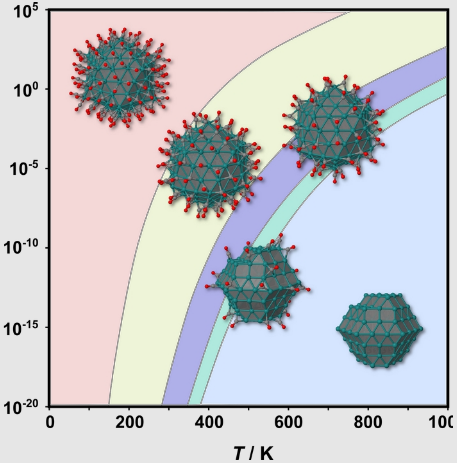
UNIVERSITÉ TOULOUSE III PAUL SABATIER Université de Toulouse

LPCNO Laboratoire de Physique & Chimie des Nano-Objets

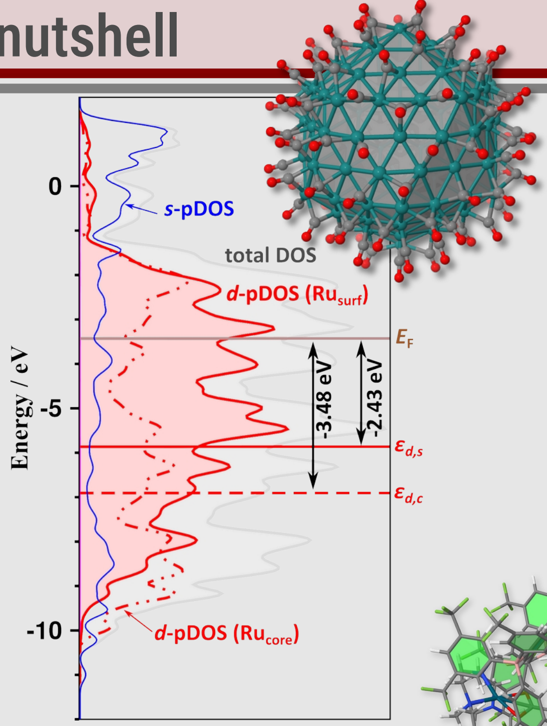
Scientific activity in a nutshell

Applied Quantum Chemistry

DFT (Gaussian)
DFT with PBC (VASP)
(basic tools freely available)

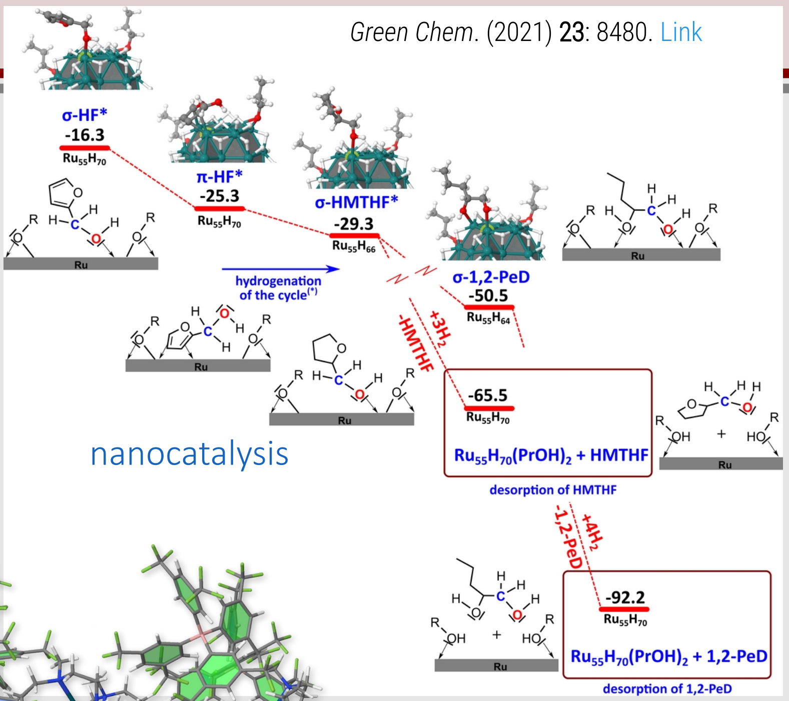


3D structure and stability of colloidal MNPs



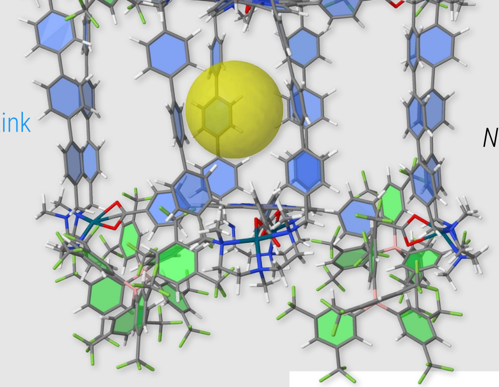
electronic structure of MNPs

Chem. Mater. (2022) **34**: 2931-2944. Link

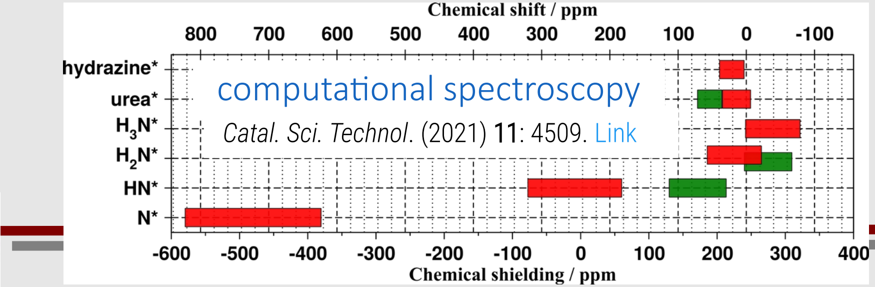


Green Chem. (2021) **23**: 8480. Link

nanocatalysis



Nanoscale Horiz. (2022) **7**: 607. Link



computational spectroscopy

Catal. Sci. Technol. (2021) **11**: 4509. Link

“holistic” approach ← physical chemistry

Version: Sunday, October 1, 2023

LPCNO

Outline of this lecture

1. General introduction

2. Short selection of [simple] applications of supervised learning to chemistry

3. Tutorials / Live demonstrations ← Jupyter notebooks

github repository

Python in the
[Physical] Chemistry Lab

[PytChem]

https://github.com/rpoteau/PytChem

Université de Toulouse

Version: Sunday, October 1, 2023

General context

Context

Artificial Intelligence (AI)

intelligence demonstrated by machines, as opposed to the natural intelligence displayed by humans or animals

Goals

reasoning & (basic) problem solving

knowledge representation

planning: making choices and hierarchy of events

learning (*i.e*. machine learning)

natural language processing

perception of the world from sensors

ability to move and manipulate objects

simulating human affects

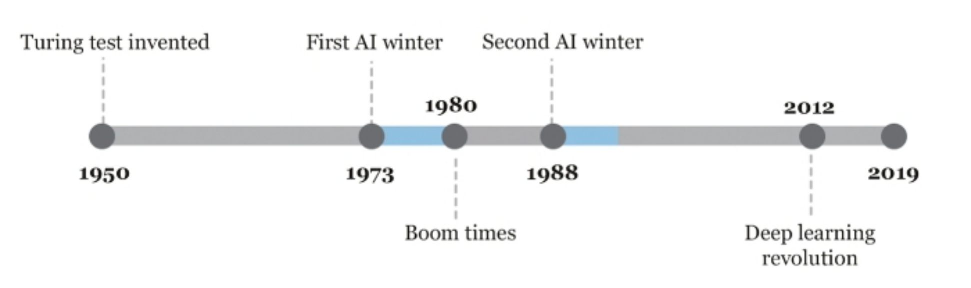
long-term goal: ability to solve an arbitrary problem

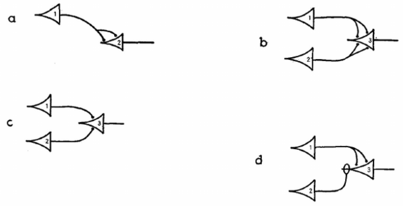
N.B. in some fields "artificial intelligence" means "machine learning with neural networks"

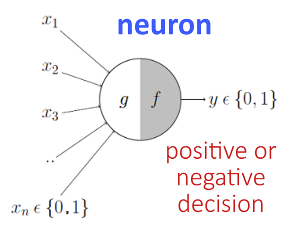
Context

Artificial Intelligence & Deep learning timelines

Since the first McC&P mathematical model for a neuron & the pioneering work of Turing, AI has a long history, with two "winters"



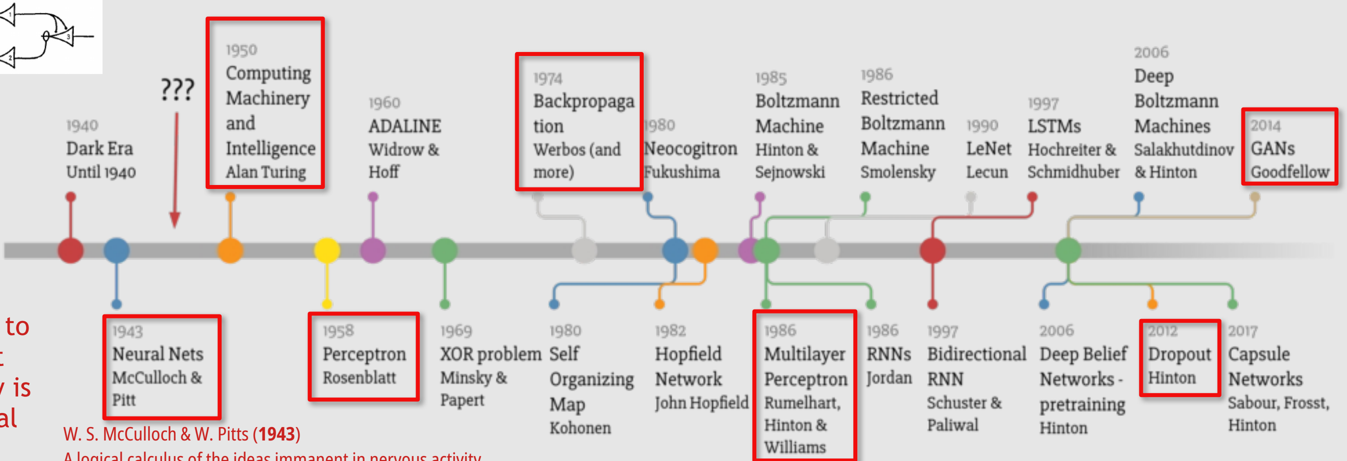




neuron

positive or negative decision

were the first to suggest that neural activity is computational



???
1940 Dark Era Until 1940
1950 Computing Machinery and Intelligence Alan Turing
1960 ADALINE Widrow & Hoff
1974 Backpropagation Werbos (and more)
1980 Neocogitron Fukushima
1985 Boltzmann Machine Hinton & Sejnowski
1986 Restricted Boltzmann Machine Smolensky
1990 LeNet Lecun
1997 LSTMs Hochreiter & Schmidhuber
2006 Deep Boltzmann Machines Salakhutdinov & Hinton
2014 GANs Goodfellow
1943 Neural Nets McCulloch & Pitt
1958 Perceptron Rosenblatt
1969 XOR problem Minsky & Papert
1980 Self Organizing Map Kohonen
1982 Hopfield Network John Hopfield
1986 Multilayer Perceptron Rumelhart, Hinton & Williams
1986 RNNs Jordan
1997 Bidirectional RNN Schuster & Paliwal
2006 Deep Belief Networks - pretraining Hinton
2012 Dropout Hinton
2017 Capsule Networks Sabour, Frosst, Hinton

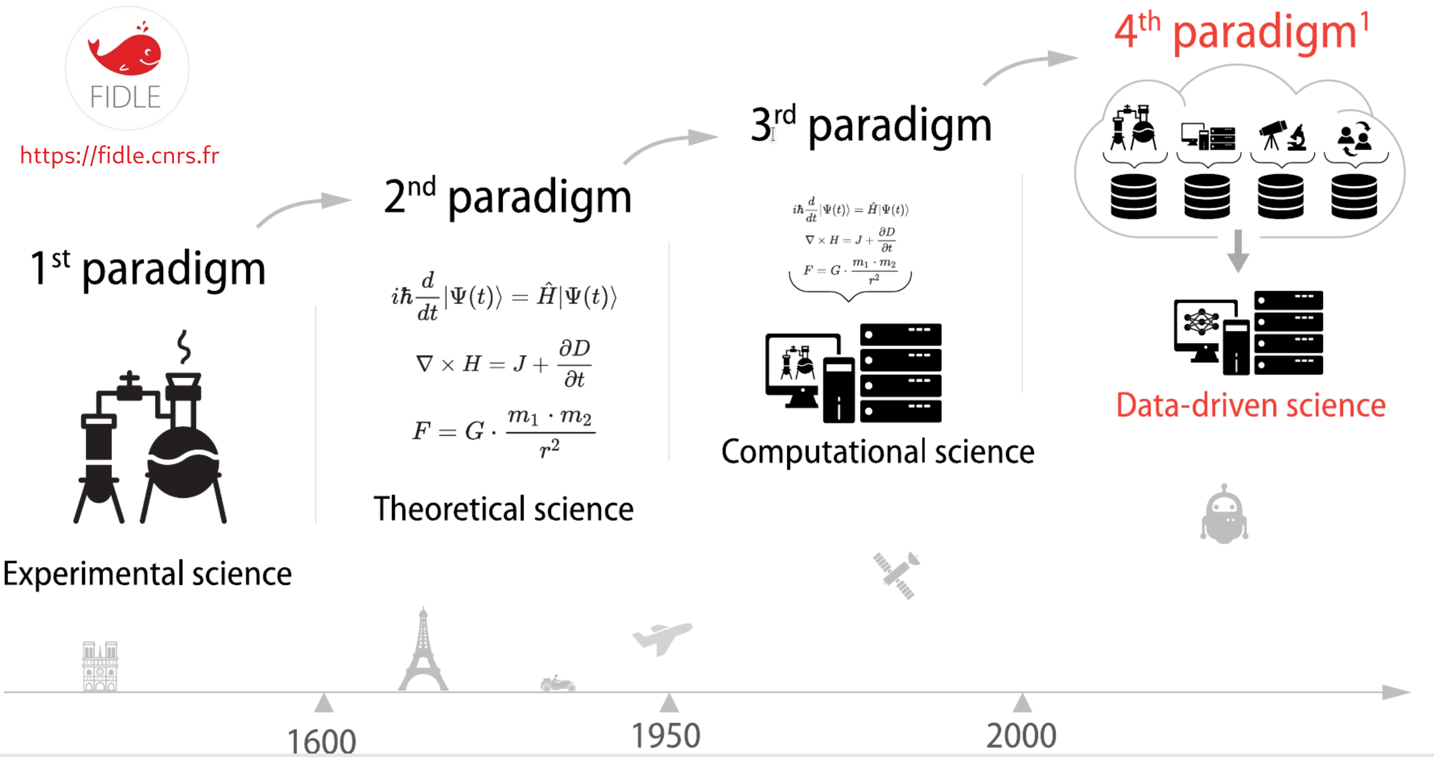
Two neural networks contest with each other

W. S. McCulloch & W. Pitts (**1943**)
A logical calculus of the ideas immanent in nervous activity
Bull. Math. Biophys. **5**, 115-133

Version: Sunday, October 1, 2023

Context

Artificial Intelligence and Scientific Research



https://fidle.cnrs.fr

1st paradigm

Experimental science

2nd paradigm

$$i\hbar\frac{d}{dt}|\Psi(t)\rangle = \hat{H}|\Psi(t)\rangle$$

$$\nabla \times H = J + \frac{\partial D}{\partial t}$$

$$F = G \cdot \frac{m_1 \cdot m_2}{r^2}$$

Theoretical science

3rd paradigm

$$\underbrace{\begin{gathered} i\hbar\frac{d}{dt}|\Psi(t)\rangle = \hat{H}|\Psi(t)\rangle \\ \nabla \times H = J + \frac{\partial D}{\partial t} \\ F = G \cdot \frac{m_1 \cdot m_2}{r^2} \end{gathered}}$$

Computational science

4th paradigm[1]

Data-driven science

NEW!

“a new forum for data-driven approaches to scientific discoveries”

experimental and computational work

all topics related to the acceleration of discovery (screening, robotics, databases and advanced data analytics)

broadly defined, but anchored in chemistry

[1] Hey, T.; Tansley, S.; Tolle, K. The Fourth Paradigm: Data-Intensive Scientific Discovery; The Fourth Paradigm: Data-Intensive Scientific Discovery; Microsoft Research, 2009

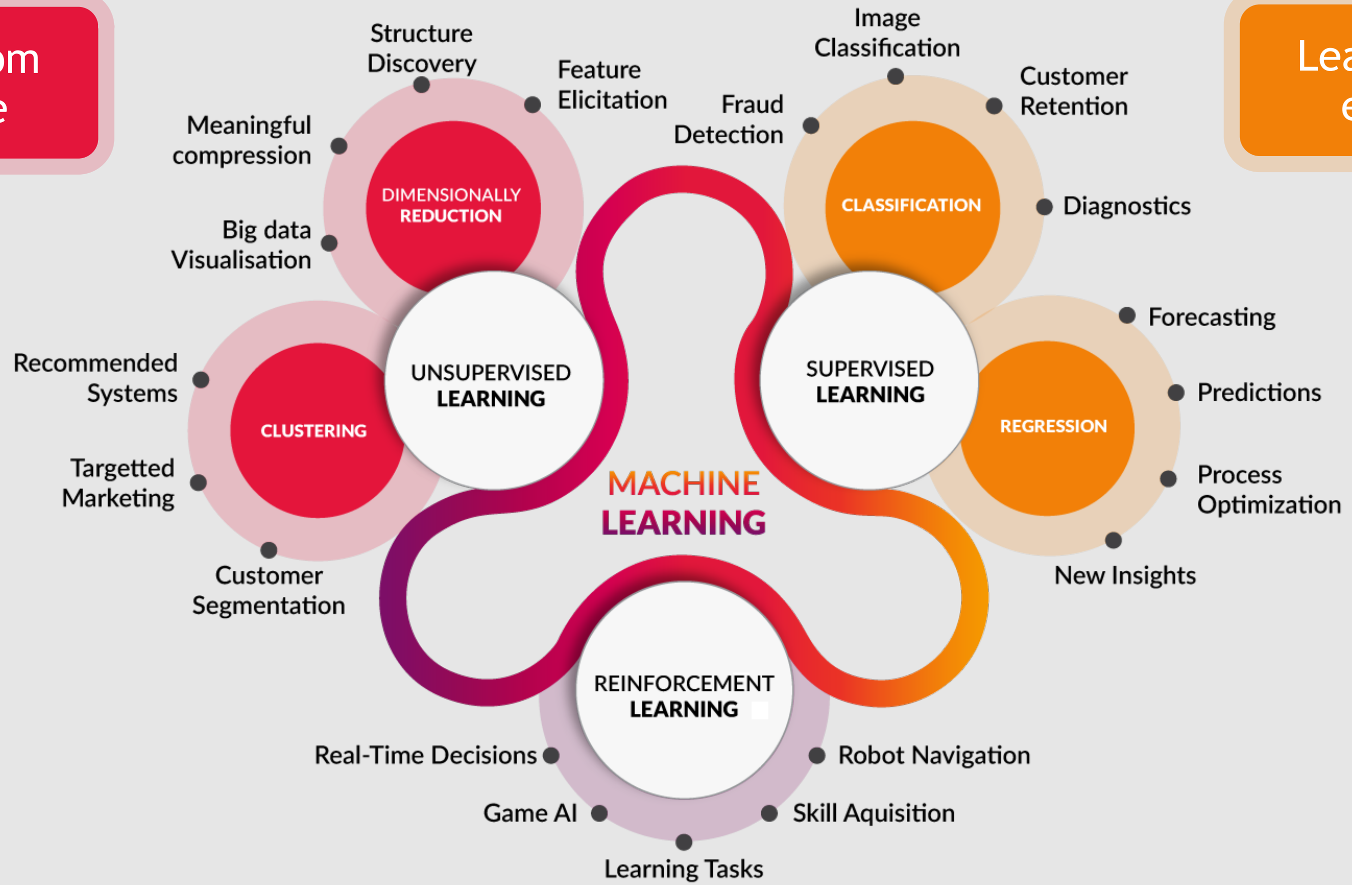
Version: Sunday, October 1, 2023

Machine Learning

Machine learning

Learning from data alone

Learning from examples



MACHINE LEARNING

UNSUPERVISED LEARNING

DIMENSIONALLY REDUCTION

Structure Discovery

Feature Elicitation

Meaningful compression

Big data Visualisation

CLUSTERING

Recommended Systems

Targetted Marketing

Customer Segmentation

SUPERVISED LEARNING

CLASSIFICATION

Image Classification

Customer Retention

Fraud Detection

Diagnostics

REGRESSION

Forecasting

Predictions

Process Optimization

New Insights

REINFORCEMENT LEARNING

Real-Time Decisions

Robot Navigation

Game AI

Skill Aquisition

Learning Tasks

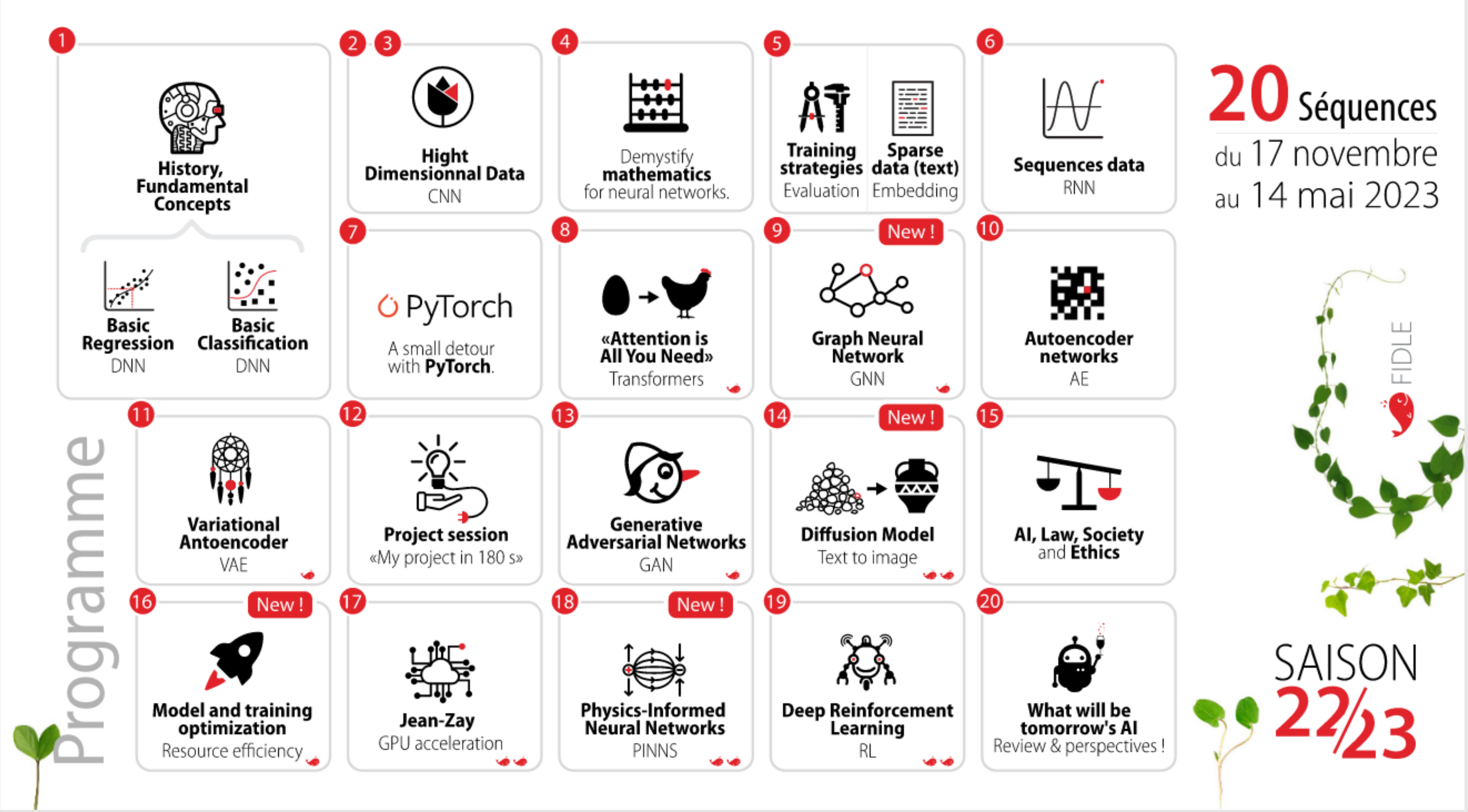
Version: Sunday, October 1, 2023

https://towardsdatascience.com/coding-deep-learning-for-beginners-types-of-machine-learning-b9e651e1ed9d

Introduction to deep learning

FIDLE

https://fidle.cnrs.fr



Programme

1. **History, Fundamental Concepts**
   - **Basic Regression** DNN
   - **Basic Classification** DNN
2. 3. **Hight Dimensional Data** CNN
4. Demystify **mathematics** for neural networks.
5. **Training strategies** Evaluation — **Sparse data (text)** Embedding
6. **Sequences data** RNN
7. PyTorch — A small detour with **PyTorch**.
8. **«Attention is All You Need»** Transformers
9. New ! **Graph Neural Network** GNN
10. **Autoencoder networks** AE
11. **Variational Antoencoder** VAE
12. **Project session** «My project in 180 s»
13. **Generative Adversarial Networks** GAN
14. New ! **Diffusion Model** Text to image
15. **AI, Law, Society** and **Ethics**
16. New ! **Model and training optimization** Resource efficiency
17. **Jean-Zay** GPU acceleration
18. New ! **Physics-Informed Neural Networks** PINNS
19. **Deep Reinforcement Learning** RL
20. **What will be tomorrow's AI** Review & perspectives !

20 Séquences

du 17 novembre au 14 mai 2023

FIDLE

SAISON 22/23

Machine learning

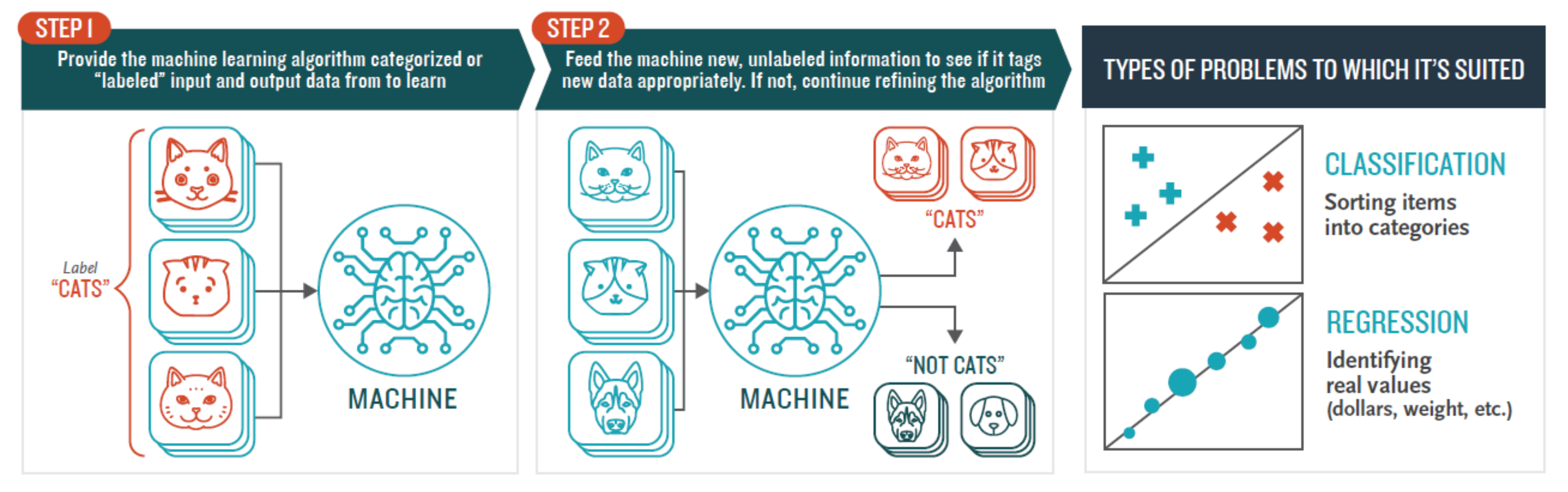
Supervised learning

Data are provided along with the desired output (*i.e.* labelled data)

Learning from examples

<u>Example of cats detection</u>:

- collect thousands of images of cats
- draw a bounding box around each cat
- feed the entire dataset to the machine so it can learn all by itself



STEP 1
Provide the machine learning algorithm categorized or "labeled" input and output data from to learn
Label
"CATS"
MACHINE
STEP 2
Feed the machine new, unlabeled information to see if it tags new data appropriately. If not, continue refining the algorithm
"CATS"
"NOT CATS"
MACHINE
TYPES OF PROBLEMS TO WHICH IT'S SUITED
CLASSIFICATION
Sorting items into categories
REGRESSION
Identifying real values (dollars, weight, etc.)

https://towardsdatascience.com/wtf-is-machine-learning-a-quick-guide-39457e49c65b

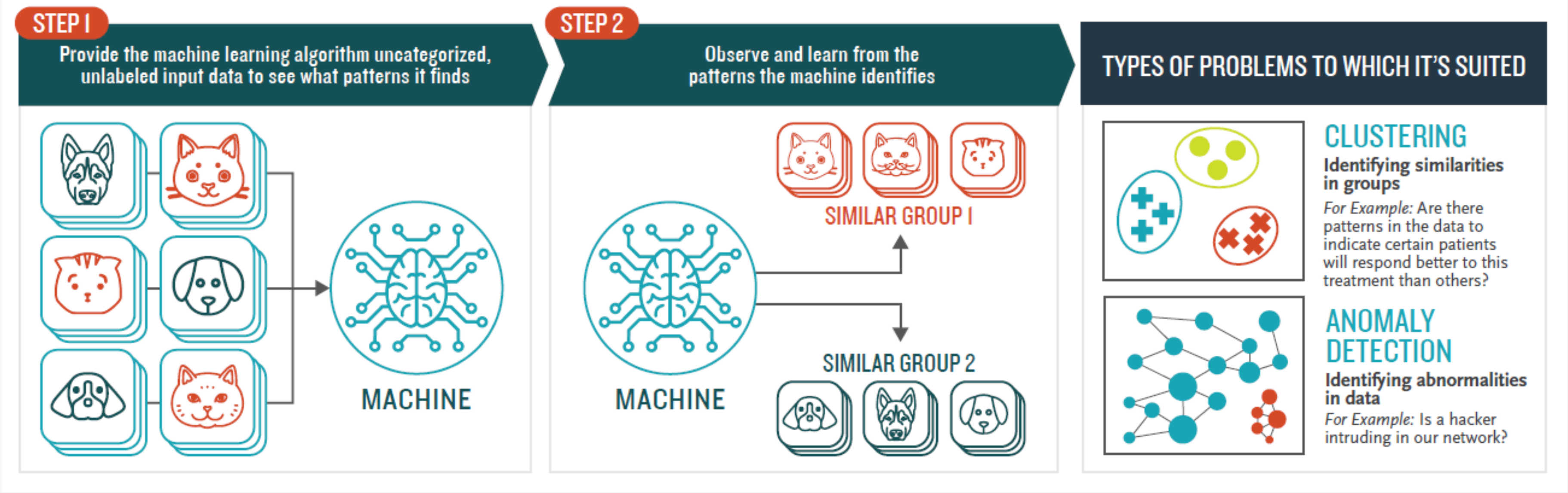
Version: Sunday, October 1, 2023

Machine learning

Unsupervised learning

Learning from data alone

- Just provide data
- Let the machine find out (or cluster) the patterns in the dataset



STEP 1

Provide the machine learning algorithm uncategorized, unlabeled input data to see what patterns it finds

STEP 2

Observe and learn from the patterns the machine identifies

TYPES OF PROBLEMS TO WHICH IT'S SUITED

CLUSTERING

Identifying similarities in groups

For Example: Are there patterns in the data to indicate certain patients will respond better to this treatment than others?

ANOMALY DETECTION

Identifying abnormalities in data

For Example: Is a hacker intruding in our network?

Version: Sunday, October 1, 2023

LPCNO

https://towardsdatascience.com/wtf-is-machine-learning-a-quick-guide-39457e49c65b

How to develop home-made ML tools?

R

MATLAB®

mathematica

python™

Python is a high-level, interpreted, object-oriented, general-purpose programming language

Core philosophy:

- Beautiful is better than ugly.
- Explicit is better than implicit.
- Simple is better than complex.
- Complex is better than complicated.
- Readability counts.

~ 250 additional libraries are available
- data science
- machine learning
- modern scientific computation
- visualization

Python programming language

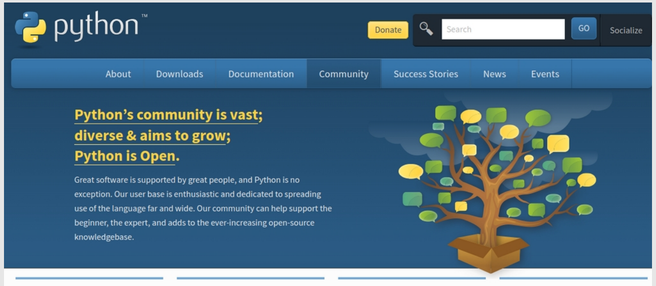
Version: Sunday, October 1, 2023

Using Python as an everyday tool for scientific calculation and computation

(it can even replace excel... by far)

→ basic knowledge of programming languages (variables, arrays, loops, conditional tests...)

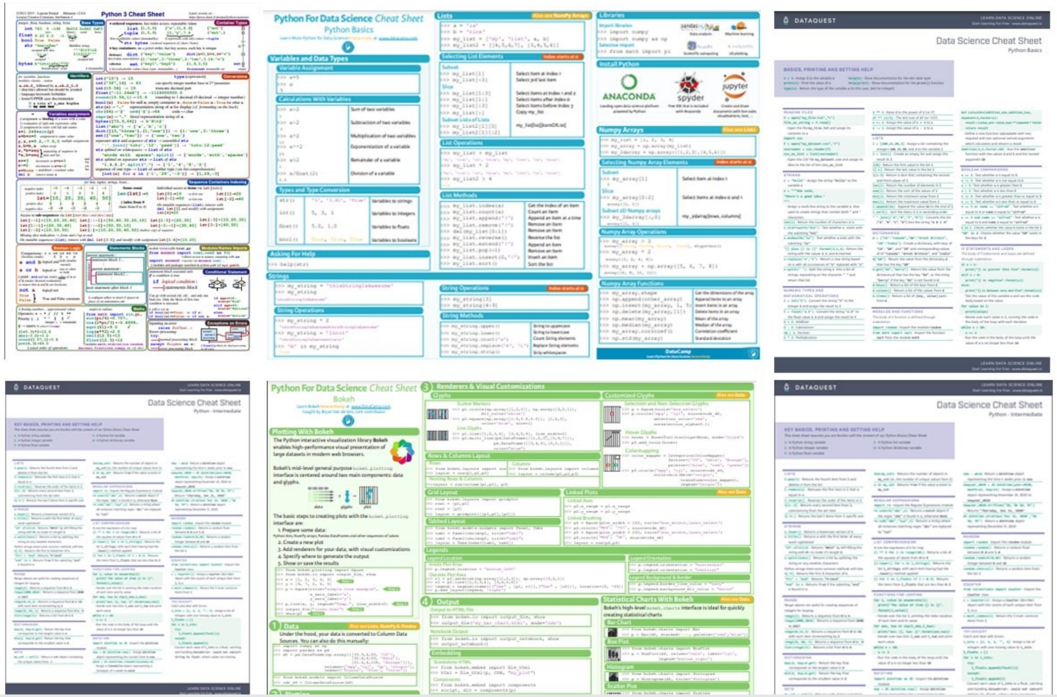
→ enthusiastic and vast community



Python's community is vast;
diverse & aims to grow;
Python is Open.

Great software is supported by great people, and Python is no exception. Our user base is enthusiastic and dedicated to spreading use of the language far and wide. Our community can help support the beginner, the expert, and adds to the ever-increasing open-source knowledgebase.

→ cheatsheets & cut/paste



Uneasy?

- yes and no
- important initial investment
- worth the effort

ML using the Python programming language

https://scikit-learn.org/

Machine learning in Python with scikit-learn

https://keras.io/

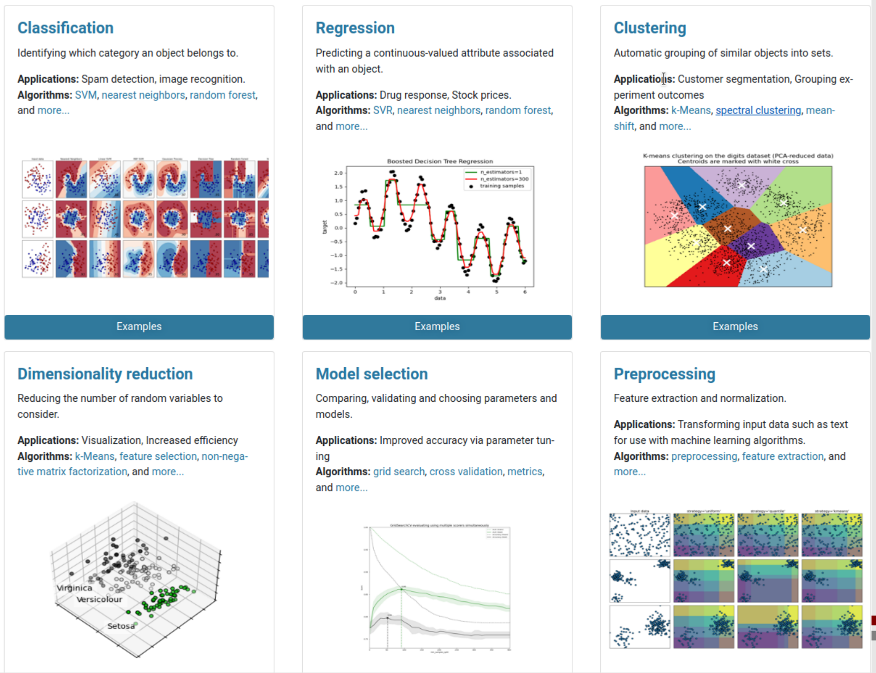
Simple and efficient tools for predictive data analysis

Accessible to everybody, and reusable in various contexts

Built on NumPy, SciPy, and matplotlib

INRIA took leadership of the project and made the first public release on February 2010

3-clause BSD License (permissive free software license, compatible with the GNU GPL)



Classification

Identifying which category an object belongs to.

Applications: Spam detection, image recognition.
Algorithms: SVM, nearest neighbors, random forest, and more...

Examples

Regression

Predicting a continuous-valued attribute associated with an object.

Applications: Drug response, Stock prices.
Algorithms: SVR, nearest neighbors, random forest, and more...

Examples

Clustering

Automatic grouping of similar objects into sets.

Applications: Customer segmentation, Grouping experiment outcomes
Algorithms: k-Means, spectral clustering, mean-shift, and more...

K-means clustering on the digits dataset (PCA-reduced data)
Centroids are marked with white cross

Examples

Dimensionality reduction

Reducing the number of random variables to consider.

Applications: Visualization, Increased efficiency
Algorithms: k-Means, feature selection, non-negative matrix factorization, and more...

Model selection

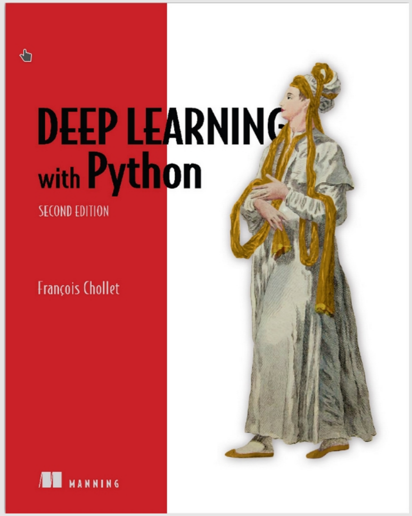
Comparing, validating and choosing parameters and models.

Applications: Improved accuracy via parameter tuning
Algorithms: grid search, cross validation, metrics, and more...

Preprocessing

Feature extraction and normalization.

Applications: Transforming input data such as text for use with machine learning algorithms.
Algorithms: preprocessing, feature extraction, and more...



High Level Deep Learning Application Programming Interface (API)

By François Cholet (Google)

Part on TensorFlow since 2017

MIT license (permissive free software license)

how to start?

```
import numpy as np
import tensorflow as tf
from tensorflow import keras
```

https://www.tensorflow.org/

Google Brain's second-generation system

Supported by Google

Low level API

Apache license (yet another permissive free software license)

Python programming language or R software

Version: Sunday, October 1, 2023

PyTorch

https://pytorch.org/

From Torch library
Supported by Facebook
BSD licence
(permissive free software license)

An open source machine learning framework that accelerates the path from research prototyping to production deployment

Widely used in the field of AI research

The ML galaxy

Keras

TensorFlow

PyTorch

scikits learn
machine learning in Python

seaborn

jupyter

python™

ANACONDA®

matplotlib

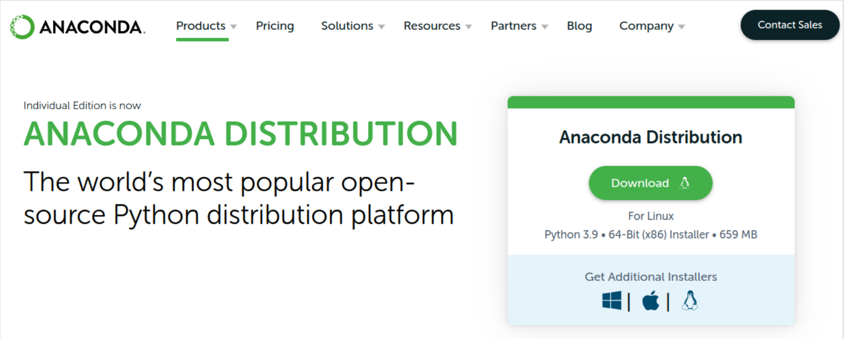
pandas

SciPy

NumPy

Python programming language

ANACONDA®



Individual Edition is now

ANACONDA DISTRIBUTION

The world's most popular open-source Python distribution platform

Anaconda Distribution

Download

For Linux

Python 3.9 • 64-Bit (x86) Installer • 659 MB

Get Additional Installers

jupyter
SciPy
SPYDER
NumPy
DASK
bokeh
scikit learn
TensorFlow
Numba
Panel
pandas
matplotlib
HoloViews
PYTORCH
CONDA

Anaconda is a distribution of the Python and R programming languages for scientific computing (data science, machine learning applications, large-scale data processing, predictive analytics, etc.), that aims to simplify package management and deployment

JupyterLab is a web-based interactive development environment for notebooks, code, and data

Users can arrange workflows in **data science**, **scientific computing**, and **machine learning**

you can save your everyday data manipulation / visualization as you do in your chemistry laboratory notebooks