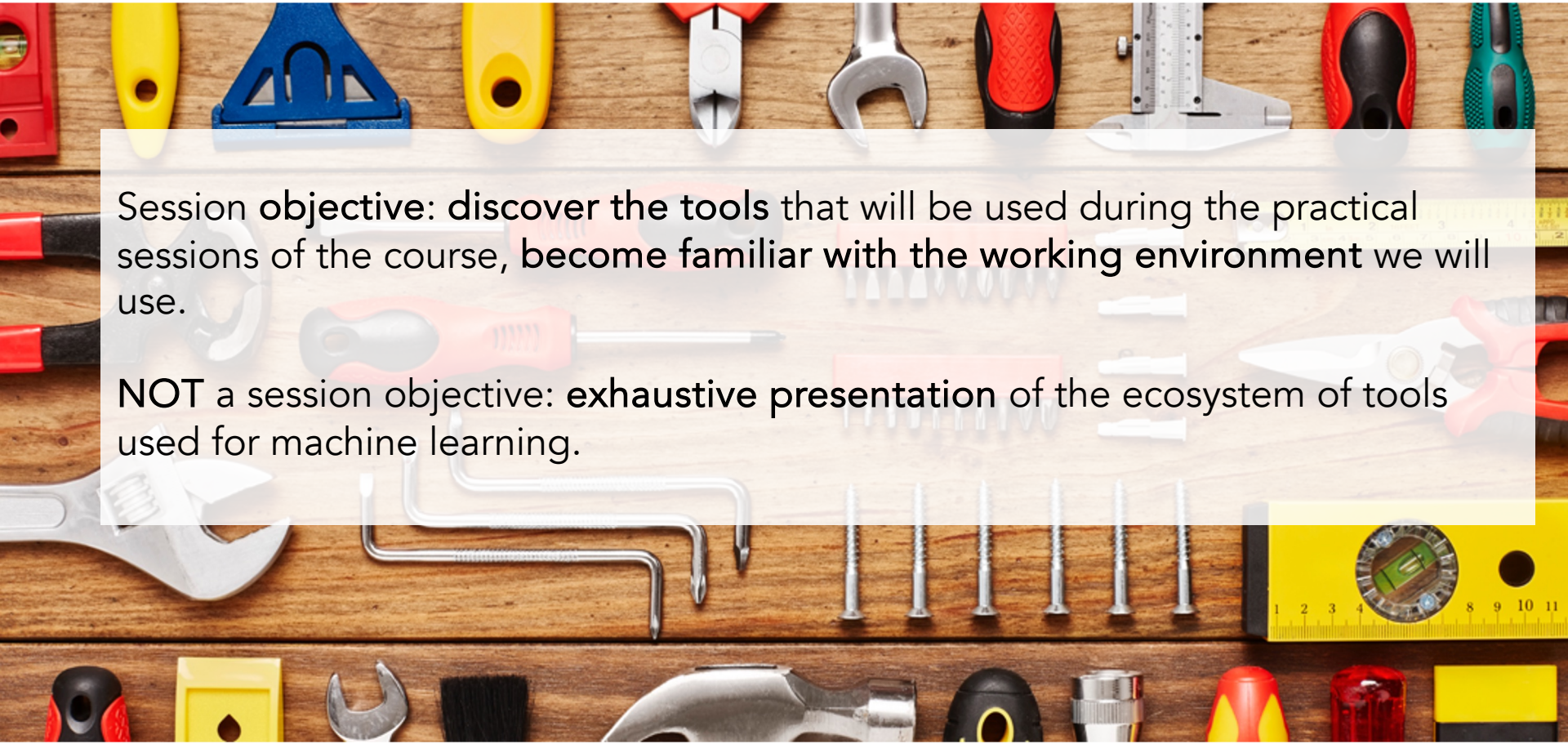
Introduction to Machine-Learning Software Tools

Daniel Nieto (d.nieto@ucm.es)

Second IPARCOS Workshop on
Machine Learning and Applications to Physics
MLAP23

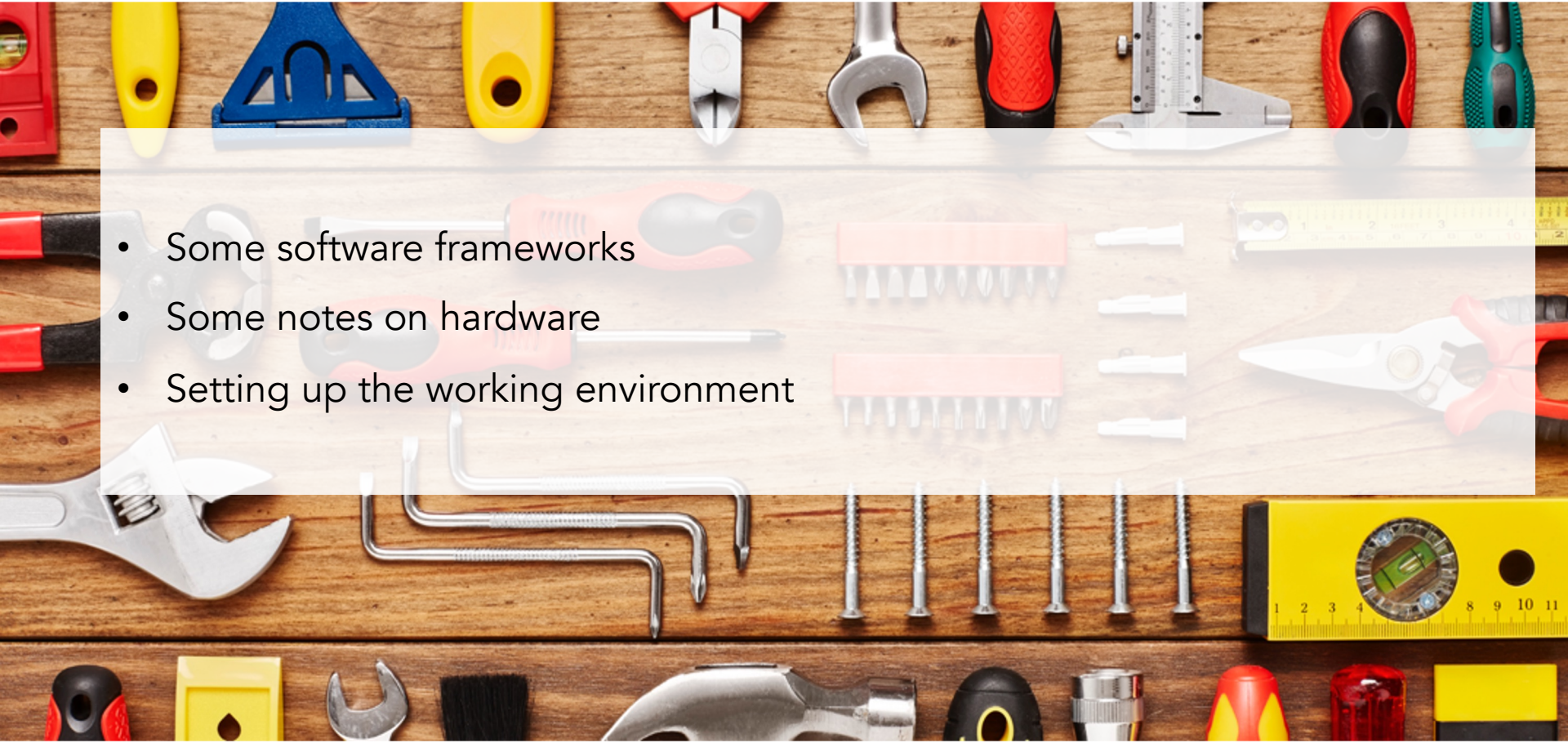
Introduction to machine-learning software tools



Session **objective**: **discover the tools** that will be used during the practical sessions of the course, **become familiar with the working environment** we will use.

NOT a session objective: **exhaustive presentation** of the ecosystem of tools used for machine learning.

Introduction to machine-learning software tools



- Some software frameworks
- Some notes on hardware
- Setting up the working environment

A zoo of frameworks

A zoo of frameworks

Google (U. Montreal)

TensorFlow

Facebook (NYU/UC Berkeley)

PyTorch

mxnet

Apache (MIT, U. Wash, …)

scikit learn

Community-driven project

飞桨 PaddlePaddle

Baidu

And many more…

R

R Foundation (GNU Project)

Mathematica

Matlab

Google

A zoo of frameworks

Google (U. Montreal)

TensorFlow

Facebook (NYU/UC Berkeley)

PyTorch

mxnet

Apache (MIT, U. Wash, …)

scikit learn

Community-driven project

飞桨 PaddlePaddle

Baidu

And many more…

R

R Foundation (GNU Project)

Mathematica

Matlab

Google

https://en.wikipedia.org/wiki/Comparison_of_deep-learning_software

A zoo of frameworks

Google (U. Montreal)

TensorFlow

mxnet

Apache (MIT, U. Wash, …)

And many more…

Facebook (NYU/UC Berkeley)

PyTorch

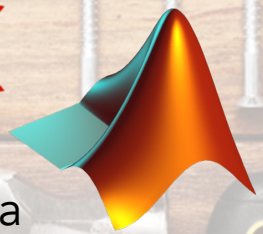
scikit learn

Community-driven project

飞桨 PaddlePaddle

Baidu

Matlab



Mathematica

R

R Foundation (GNU Project)

Google

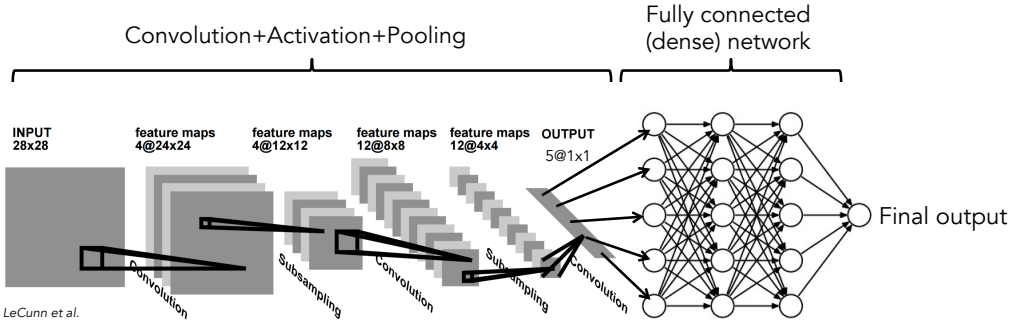
https://en.wikipedia.org/wiki/Comparison_of_deep-learning_software

A zoo of frameworks

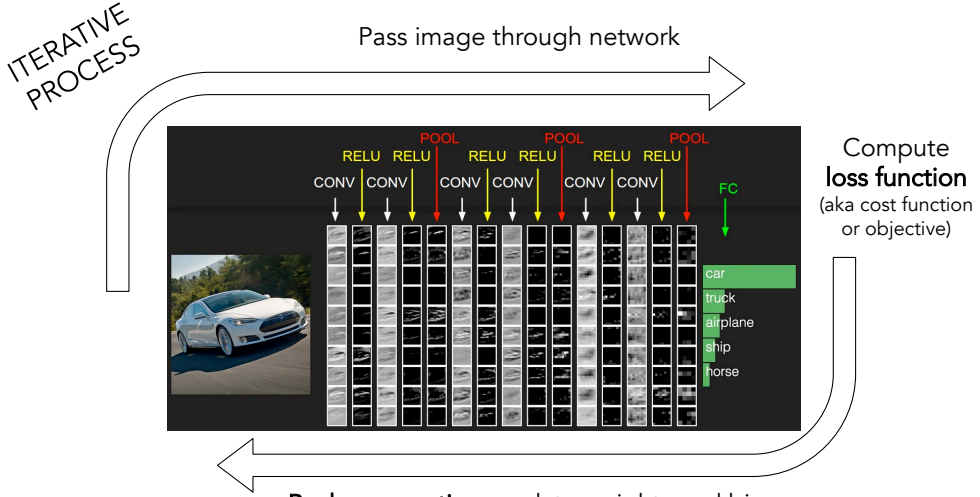
Some notes on hardware

(particularly for deep learning)

CNN model



Training process

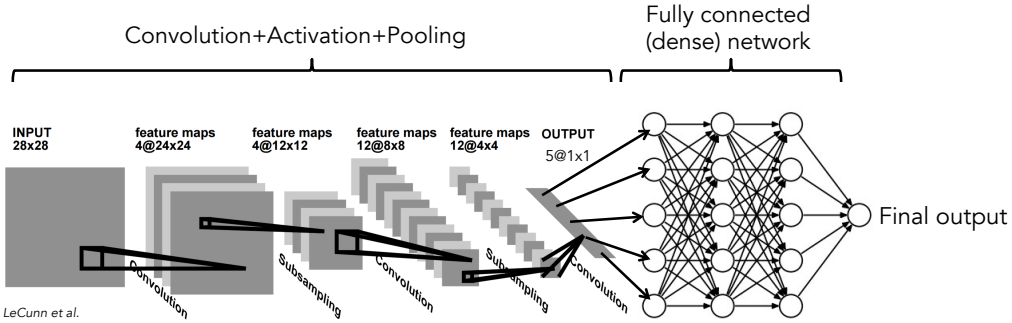


Refs: http://cs231n.github.io/

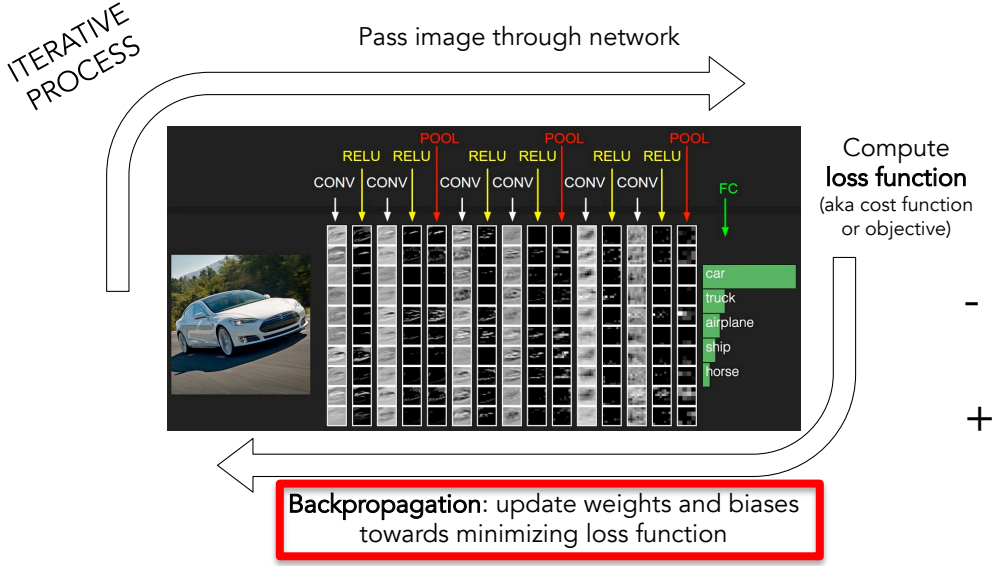
Some notes on hardware

(particularly for deep learning)

CNN model



Training process



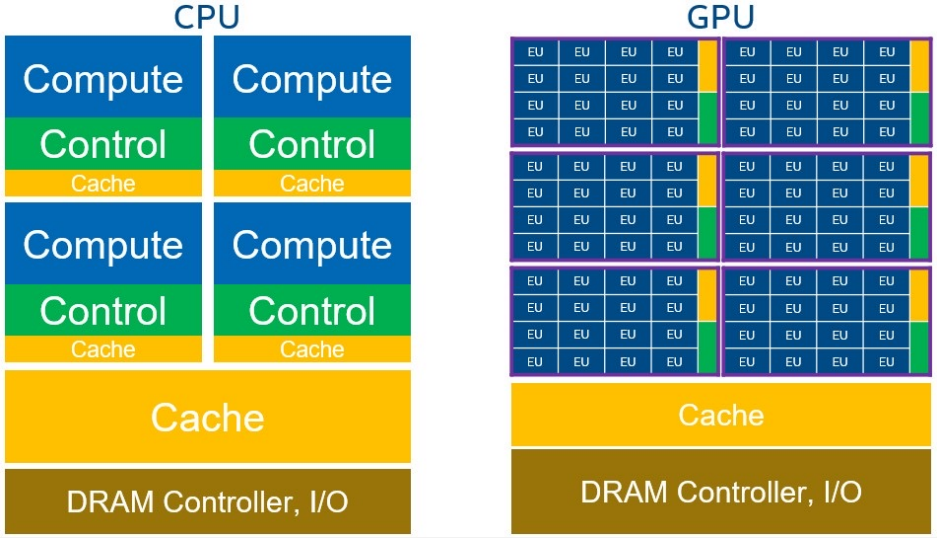
- Computationally expensive
+ Highly parallelizable

Refs: http://cs231n.github.io/

Some notes on hardware

(particularly for deep learning)

CPUs vs GPUs



	Cores	Clock Speed	Memory	Price	Speed
CPU (Intel Core i7-7700k)	4 (8 threads with hyperthreading)	4.2 GHz	System RAM	$385	~540 GFLOPs FP32
GPU NVIDIA RTX 2080 Ti	3584	1.6 GHz	11 GB GDDR 6	$1099	~13 TFLOPs FP32 ~114 TFLOPs FP16

CPU (Central Processing Unit):

- Few cores, but very fast and versatile.
- Low latency.
- Sequential tasks.

GPU (Graphics Processing Unit):

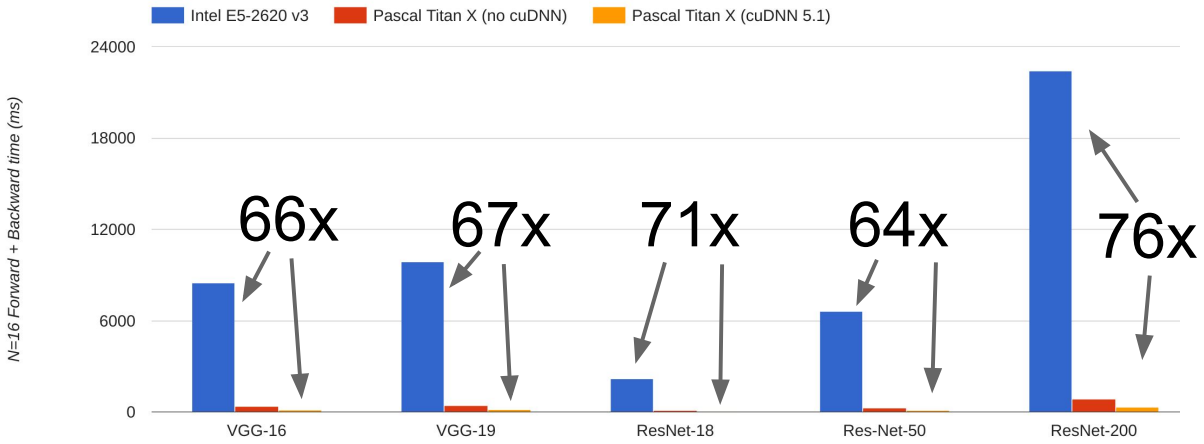
- Many cores, but slower and optimized for certain operations (e.g., matrix multiplication).
- Parallelizable tasks.

Refs. http://cs231n.stanford.edu/

Some notes on hardware

(particularly for deep learning)

CPUs vs GPUs



Data from https://github.com/jcjohnson/cnn-benchmarks

CPU (Central Processing Unit):

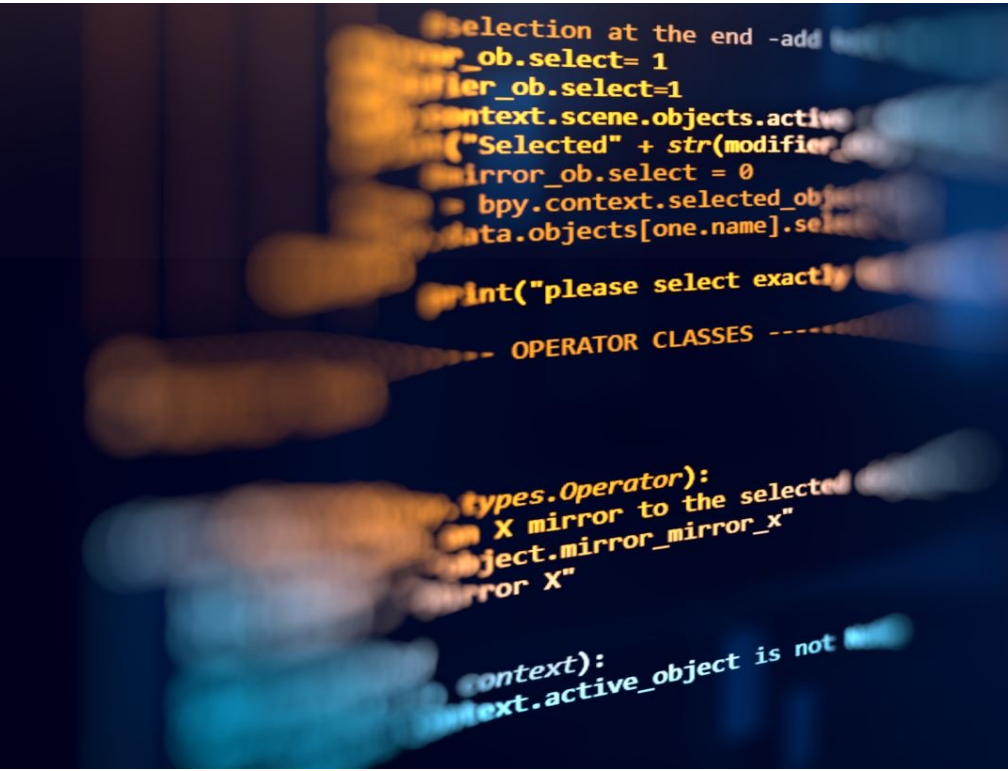
- Few cores, but very fast and versatile.
- Low latency.
- Sequential tasks.

GPU (Graphics Processing Unit):

- Many cores, but slower and optimized for certain operations (e.g., matrix multiplication).
- Parallelizable tasks.

Refs. http://cs231n.stanford.edu/

Programming on GPUs



CUDA (Nvidia only):
- Available for C, C++, Fortran, Python, and MATLAB.
- Optimized APIs: cuDNN, cuFFT, cuBLAS.

OpenCL (GPUs, FPGAs, CPUs...):
- Open standard, with C and C++ bindings.
- Slower on Nvidia GPUs but useful for other brands (e.g., AMD).

Refs. http://cs231n.stanford.edu/

Beyond GPUs

Field-Programmable Gate Arrays (FPGAs):

FPGAs are reconfigurable hardware devices that can be customized for specific machine learning workloads. They offer flexibility and efficiency, allowing for hardware-level optimization of algorithms.

ASICs (Application-Specific Integrated Circuits):

ASICs are custom-designed chips built for specific applications, including machine learning. Examples include Google's Edge TPU and the AI accelerator chips found in some smartphones, providing dedicated hardware for ML tasks.

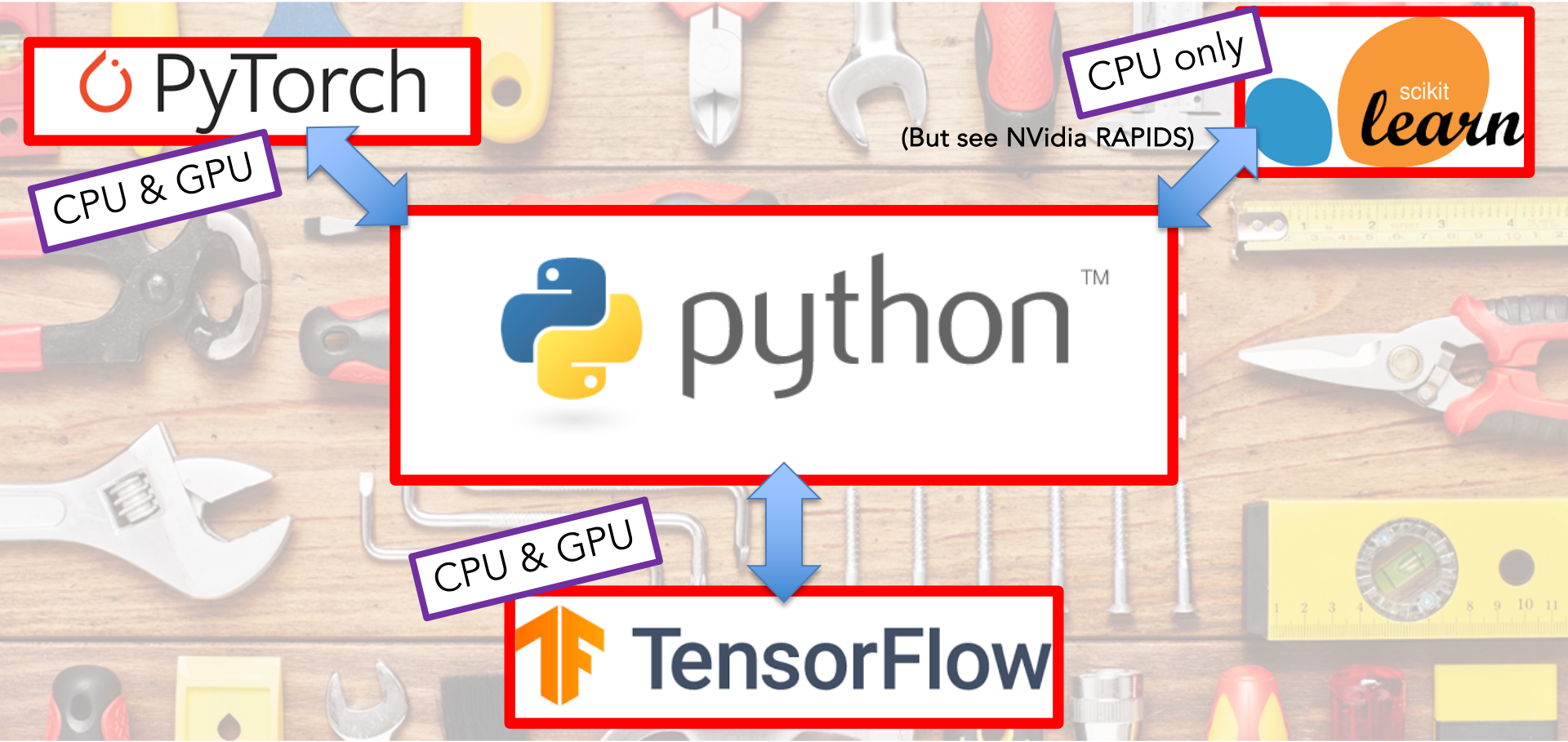
Tensor Processing Units (TPUs):

TPUs are custom-designed accelerators by Google specifically for machine learning workloads. They are optimized for matrix multiplication, a common operation in neural networks, and are integrated into Google's Cloud services for accelerated ML tasks.

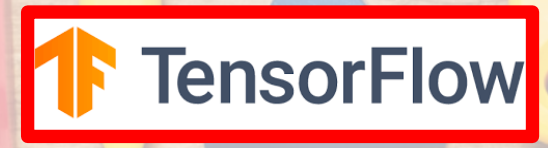
Neuromorphic Processors:

Neuromorphic processors are designed to mimic the architecture of the human brain, enabling efficient processing of spiking neural networks. These processors are geared towards tasks like pattern recognition and cognitive computing.

A zoo of frameworks



A zoo of frameworks



- Low-level library, but also features high-level layers
- Capability to run on GPU (CUDA, cuDNN)
  -> deep learning
- Python, Java, C++, Go, Haskell, C#, Julia, R, Ruby, Rust, Scala APIs

- Low-level library, but also features high-level layers
- Capability to run on GPU (CUDA, cuDNN)
  -> deep learning
- Python, C++ APIs

- High-level library
- Wide variety of implemented algorithms
- Ease of use
- Built on NumPy, SciPy & matplotlib
- Python API

Some examples of (free) cloud services for ML

- (Free) Access to computing resources: CPU, GPU, storage, development interfaces
- Free resources are limited, but they can help us start prototyping our own ML projects and assess our hardware needs.
- Specific programs for higher education, including online courses and certifications.
- Some examples:

Google Cloud
300$ free credit
Access to GPUs, TPUs

Amazon Web Services
12-month free access
SageMaker: access to GPUs

Microsoft Azure
200$ free credit
Access to GPUs

Setting up the working environment

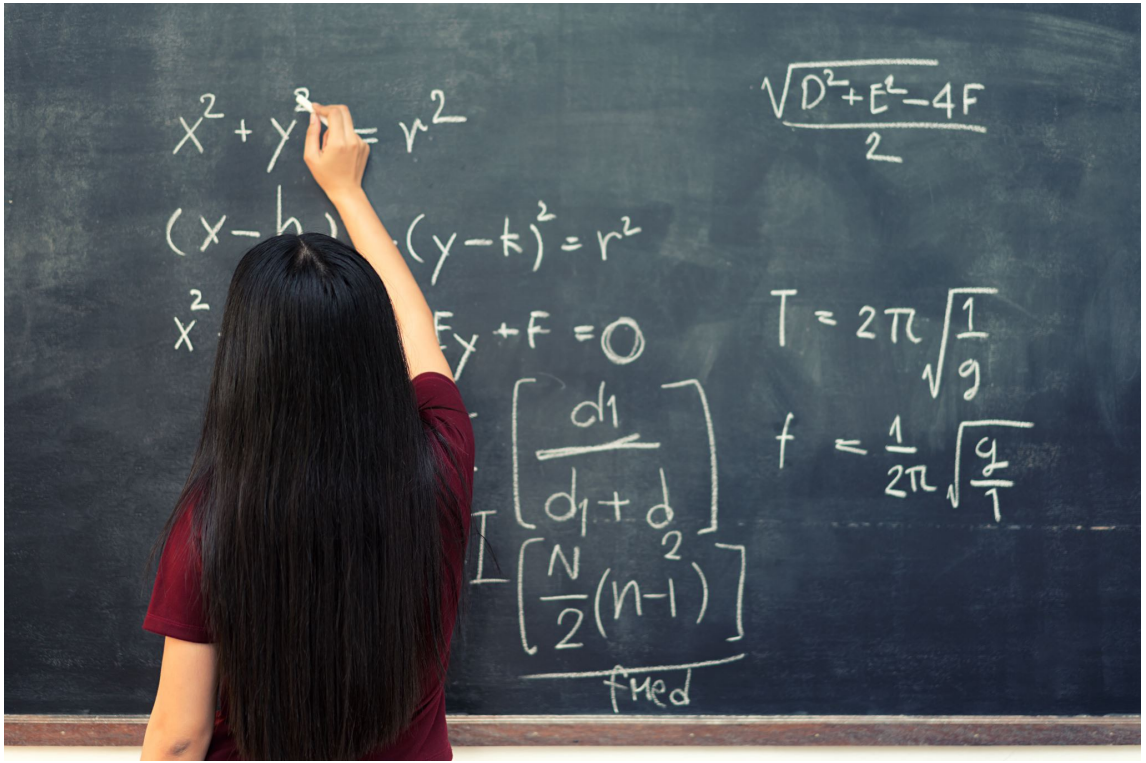
- Google Colaboratory
- Flash tutorials
  - Python
  - Scikit-Learn
  - TensorFlow

Google Colaboratory

Google Colaboratory

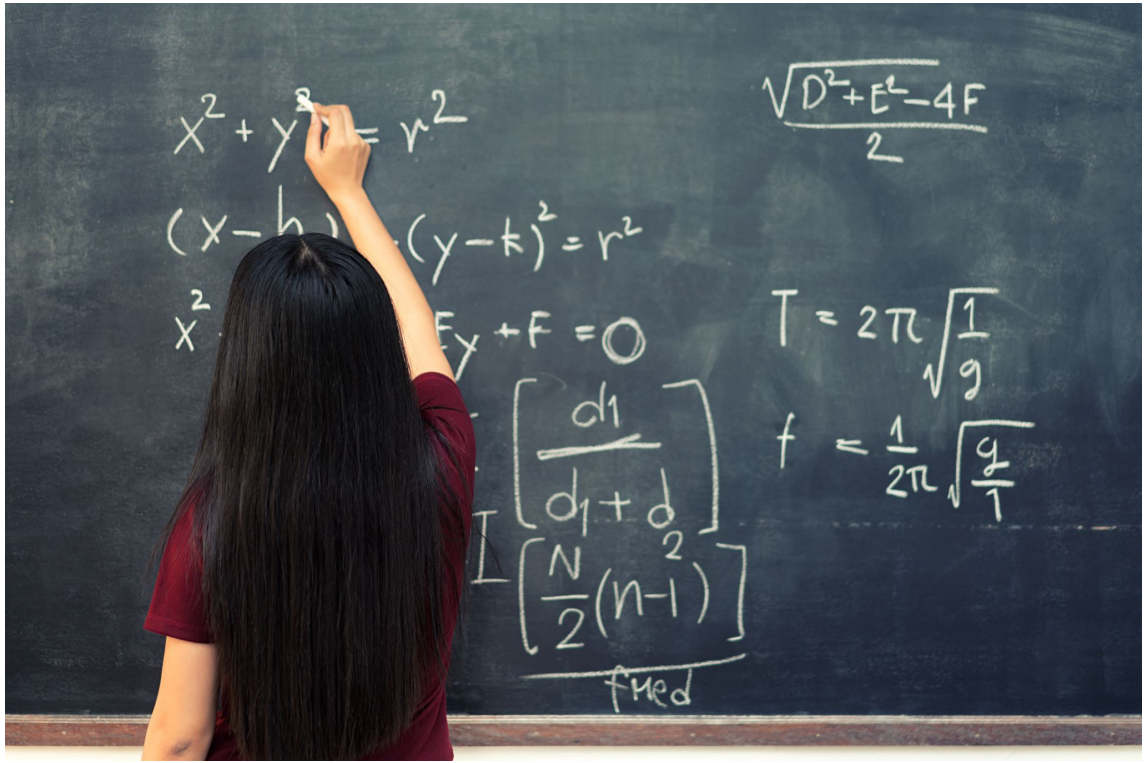
Flash tutorials

- Python
- Scikit-Learn
- TensorFlow



Flash tutorials

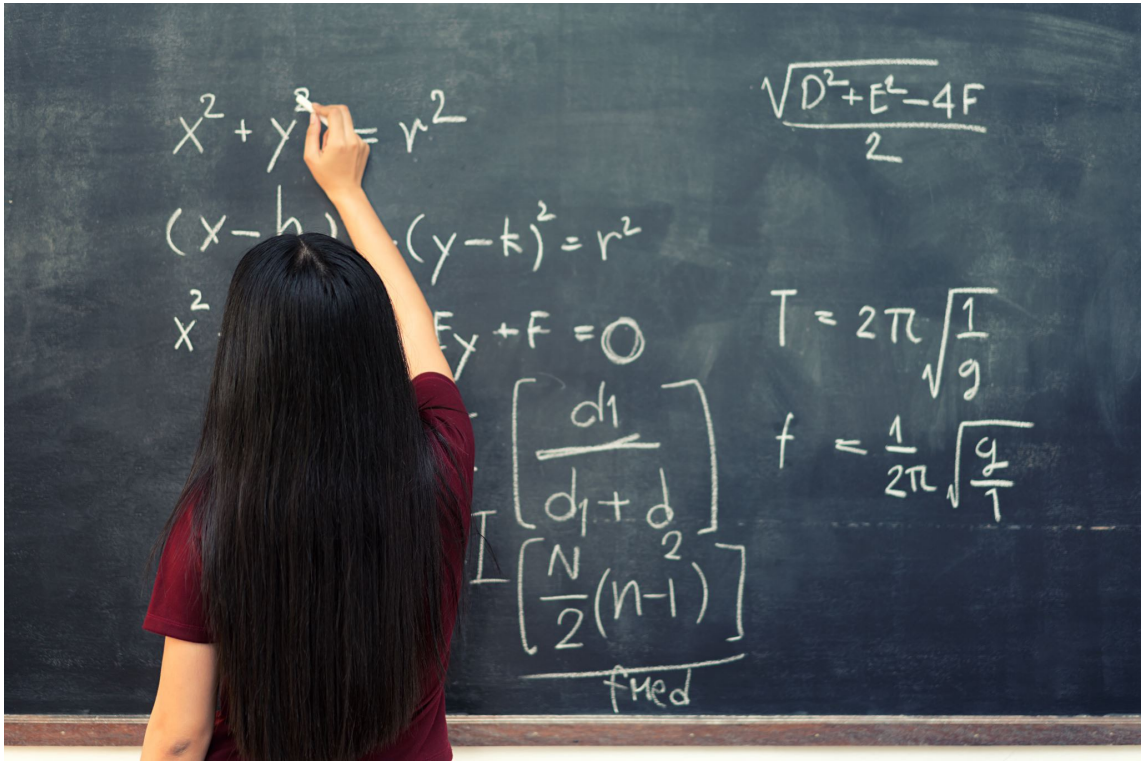
- Python
- Scikit-Learn
- TensorFlow



https://colab.research.google.com/github/cs231n/cs231n.github.io/blob/master/python-colab.ipynb

Flash tutorials

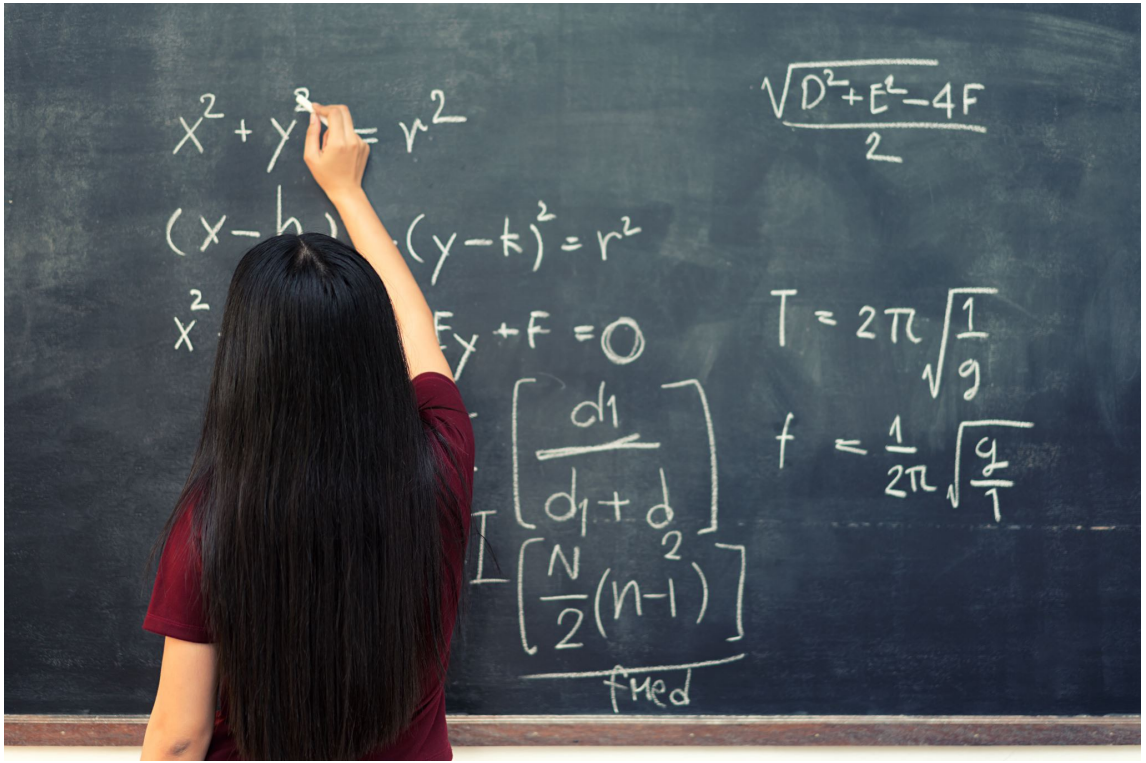
- Python
- Scikit-Learn
- TensorFlow



https://colab.research.google.com/drive/1P3JVw8ecEMV9VugEEH24poYCEZkkNLTK

Flash tutorials

- Python
- Scikit-Learn
- TensorFlow



https://colab.research.google.com/github/tensorflow/docs/blob/master/site/en/tutorials/keras/classification.ipynb

Some References & Resources

Courses
https://github.com/LHCfitNikhef/ML4PA
http://cs231n.stanford.edu/
https://www.coursera.org/learn/machine-learning
https://www.tensorflow.org/resources/learn-ml

Cloud services
https://colab.research.google.com
https://cloud.google.com
https://cloud.google.com/automl
https://aws.amazon.com
https://azure.microsoft.com

General programming (Python)
https://wiki.python.org/moin/BeginnersGuide

ML frameworks
https://www.tensorflow.org
https://scikit-learn.org
https://pytorch.org
https://www.paddlepaddle.org/
https://mxnet.apache.org/
https://docs.microsoft.com/en-us/cognitive-toolkit

Data visualization
https://matplotlib.org/tutorials/index.html
https://seaborn.pydata.org/tutorial.html