Data analysis with artificial intelligence

Julio Cesar da Silva

CNRS NEEL Institute, Grenoble, France

French CRG Beamline FAME-PIX - ESRF, Grenoble, France

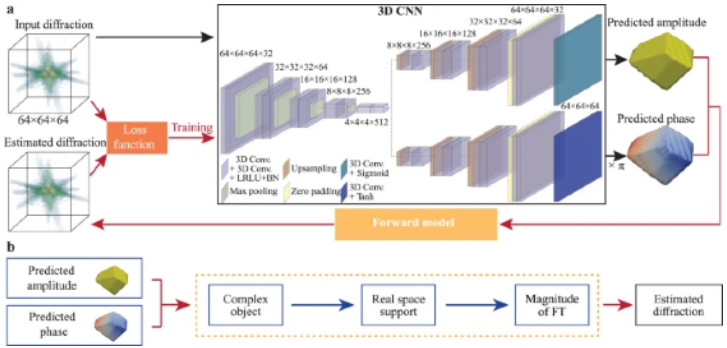
julio-cesar.da-silva@neel.cnrs.fr


812superfast

I WANT YOU

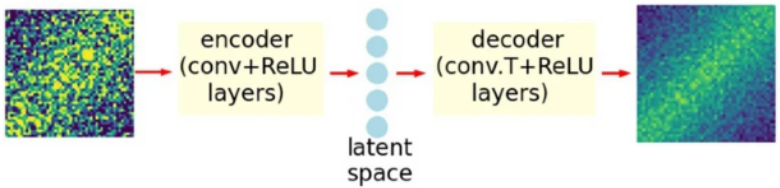
Machine learning on coherent X-ray techniques

AutoPhaseNN for Bragg CDI



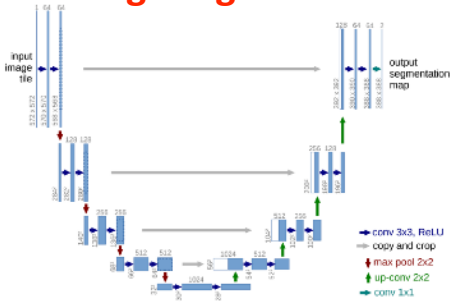
Y. Yao *et al., npj Comput Mater* 8, 124 (2022)

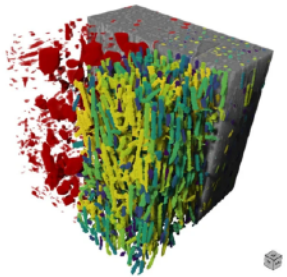
CNN Encoder-Decoder for XPCS



T. Konstantinova *et al. Sci Rep* 11, 14756 (2021)

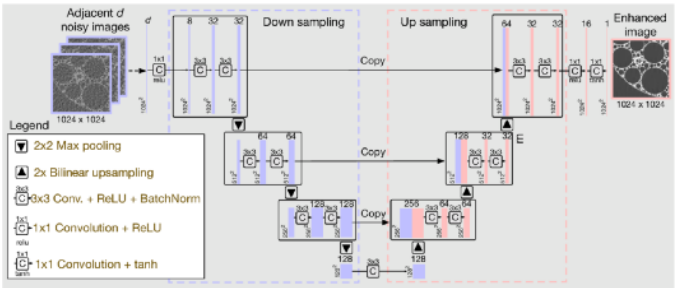
3D image segmentation or denoising with U-NET





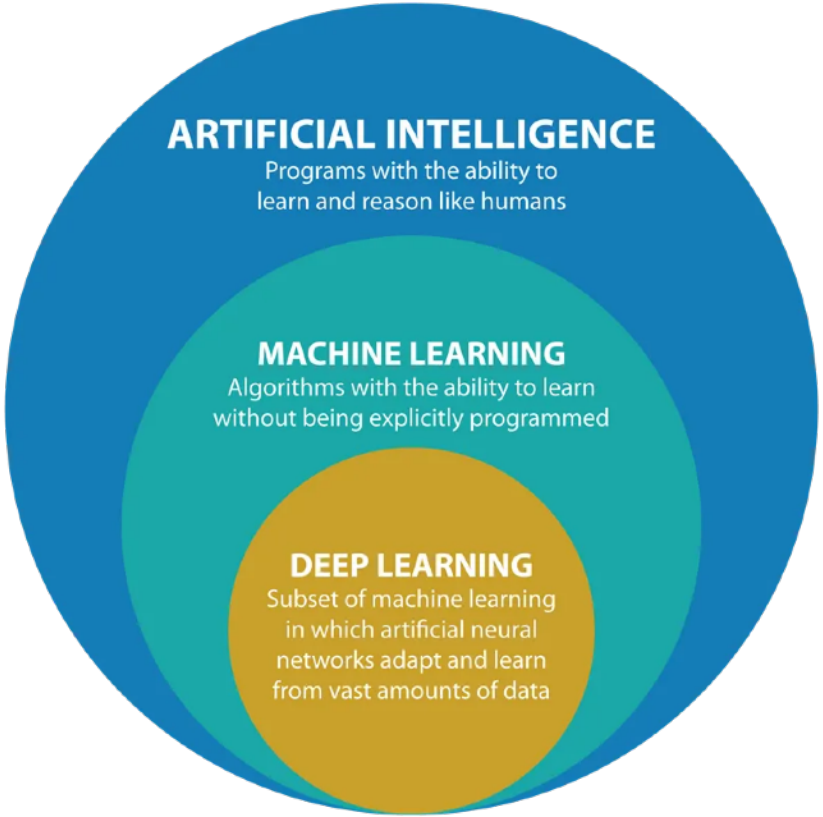
O. Ronneberger et al. (2015). MICCAI 2015. Lecture Notes in Computer Science, vol 9351. Springer

Tomographic Reconstruction with TomoGAN



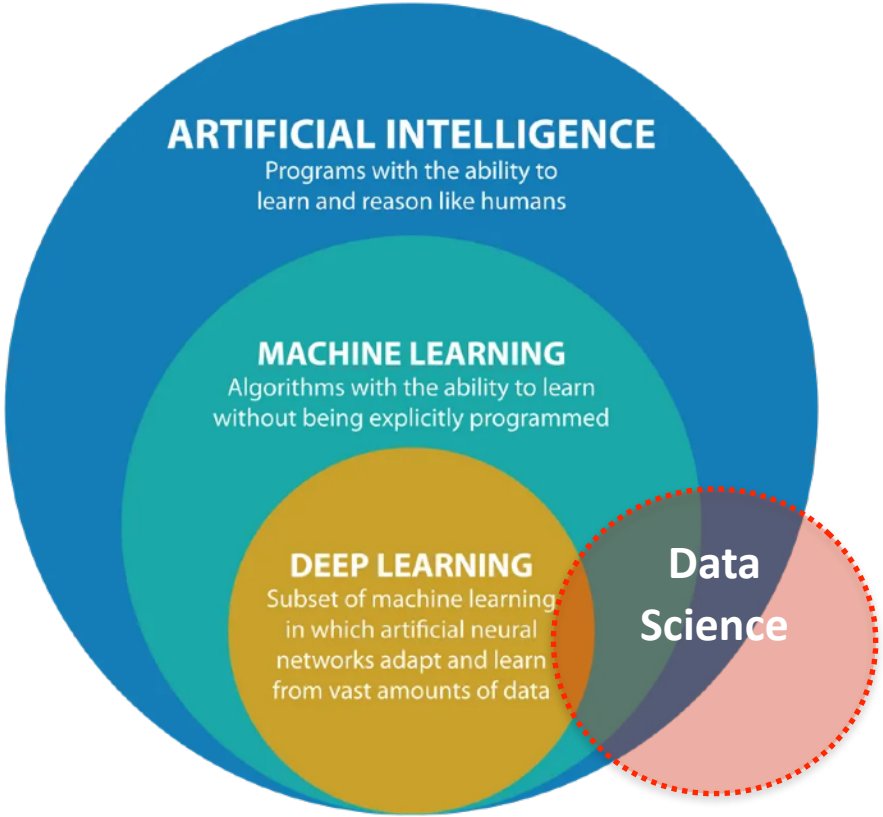
Z. Liu, et al. J. Opt. Soc. Am. A 37, 422-434 (2020)

Artificial intelligence, machine learning, deep learning



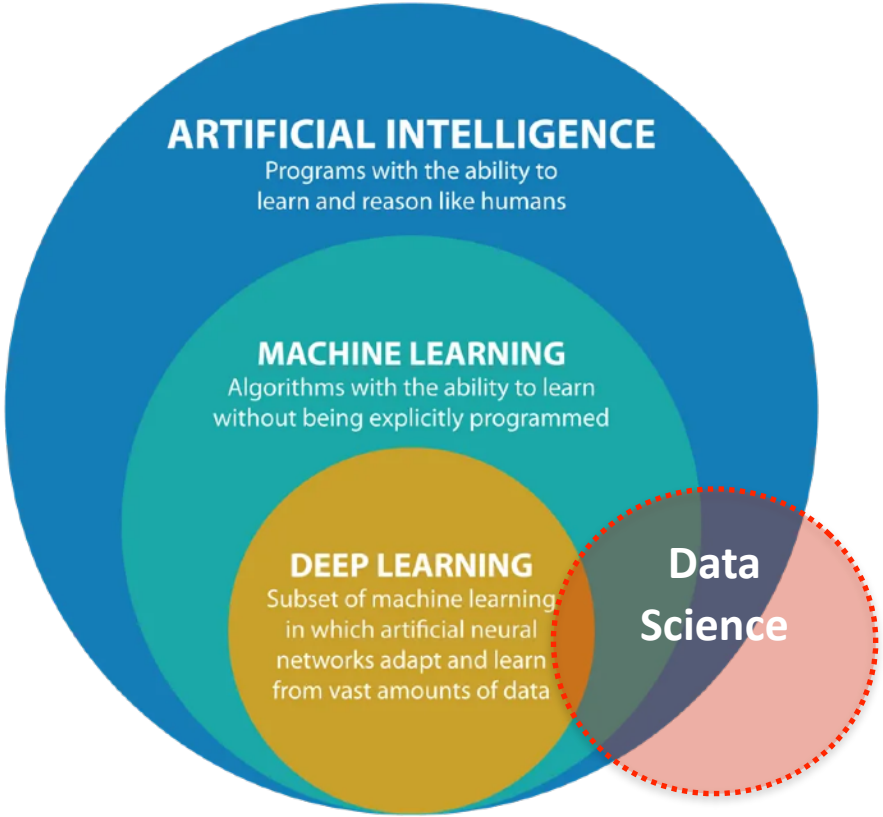
Deep Learning with Python, François Chollet, 2nd edition

Artificial intelligence, machine learning, deep learning



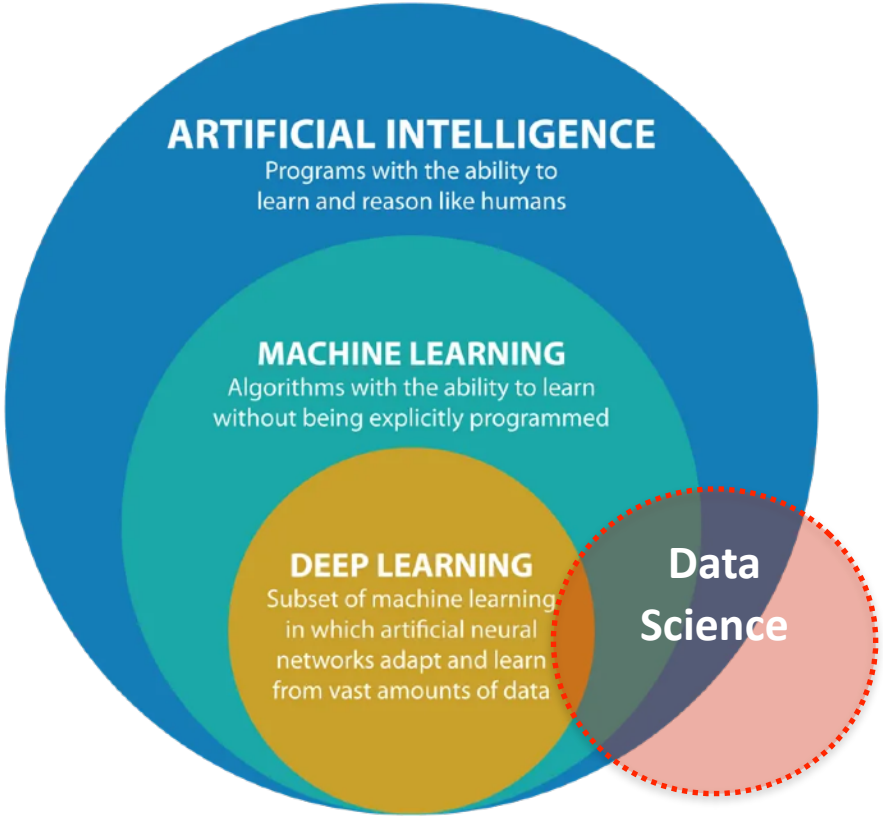
Deep Learning with Python, François Chollet, 2nd edition

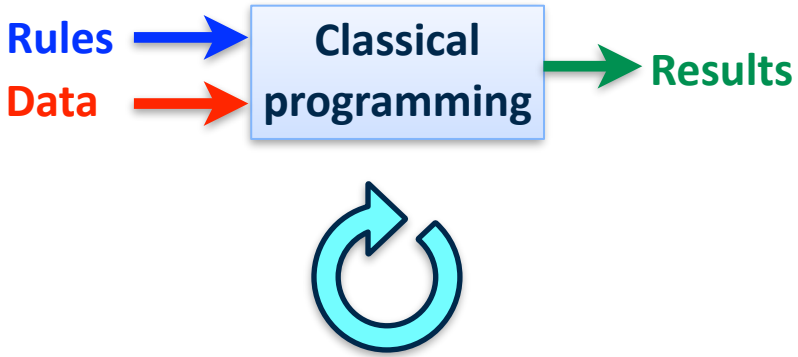
Artificial intelligence, machine learning, deep learning



Deep Learning with Python, François Chollet, 2nd edition

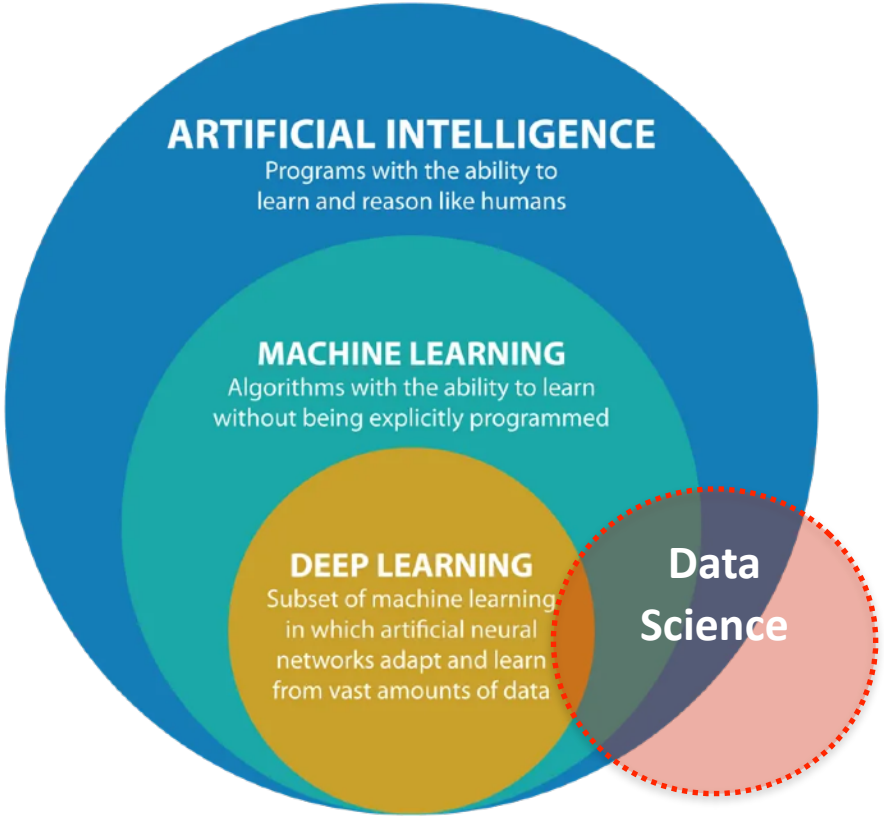
Artificial intelligence, machine learning, deep learning

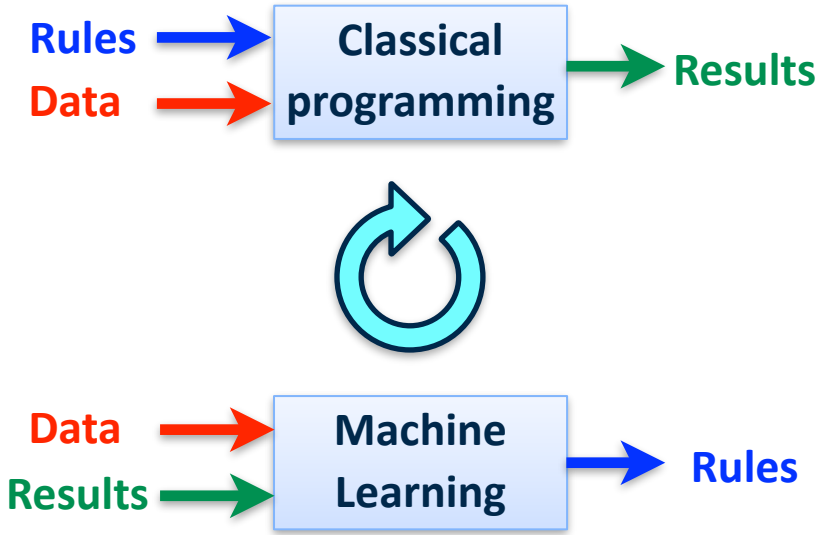




Deep Learning with Python, François Chollet, 2nd edition

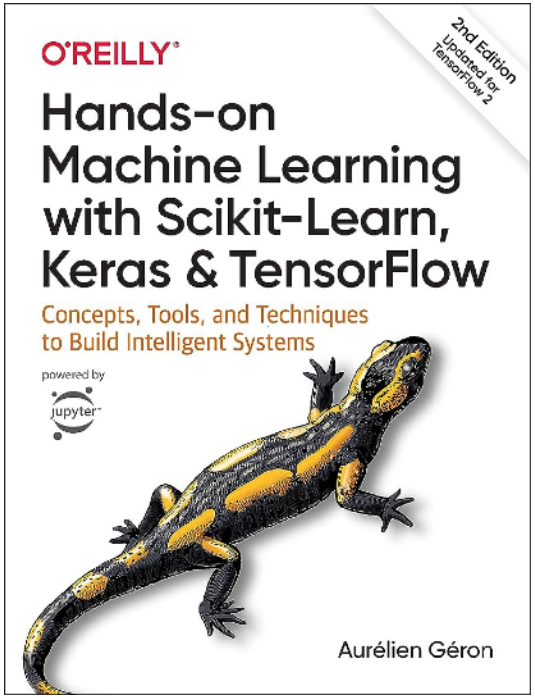
Artificial intelligence, machine learning, deep learning

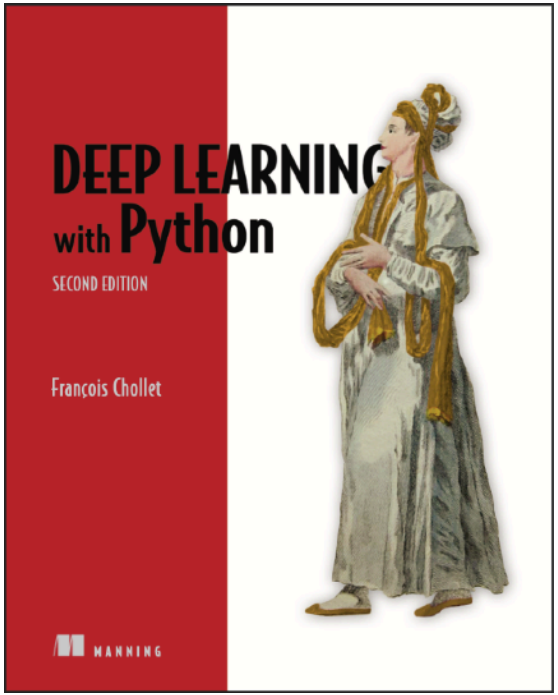




Deep Learning with Python, François Chollet, 2nd edition

References, literature, sources of knowledge





python™

scikit learn

Gaël Varoquaux

Keras TensorFlow 2.0

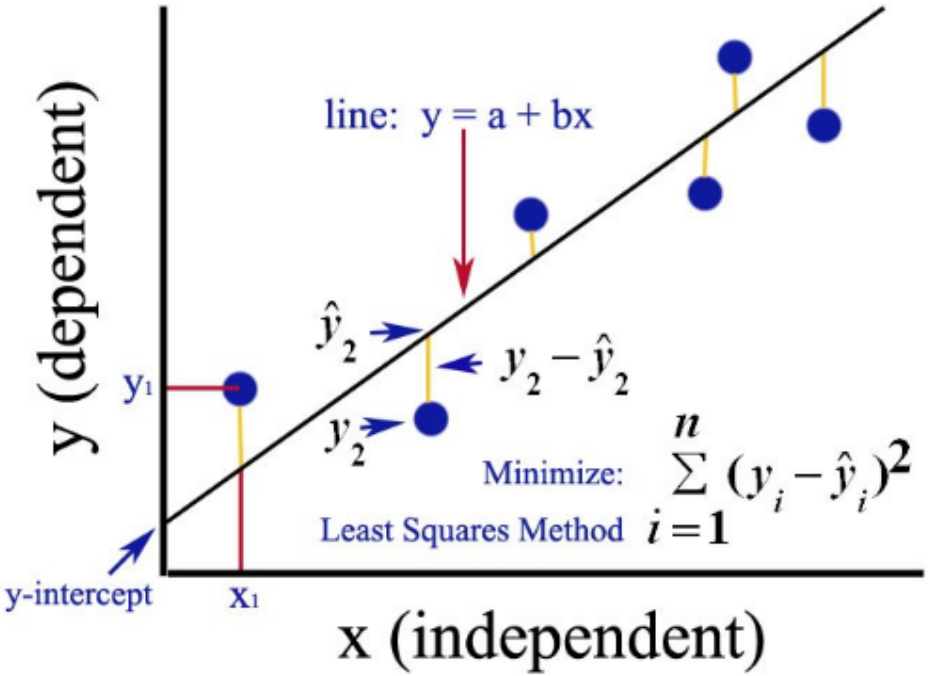
torch PyTorch

Google Colaboratory CO

kaggle

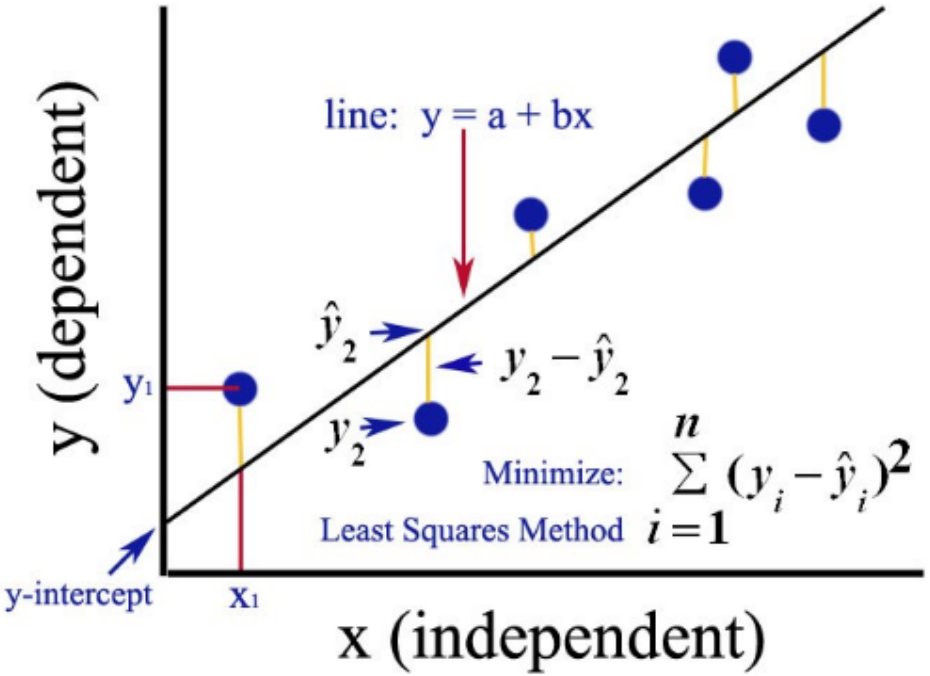
Also, AI competition

The linear regression



Hands-on Machine Learning with Scikit-learn, Keras & Tensorflow, Aurélien Géron, 2nd edition

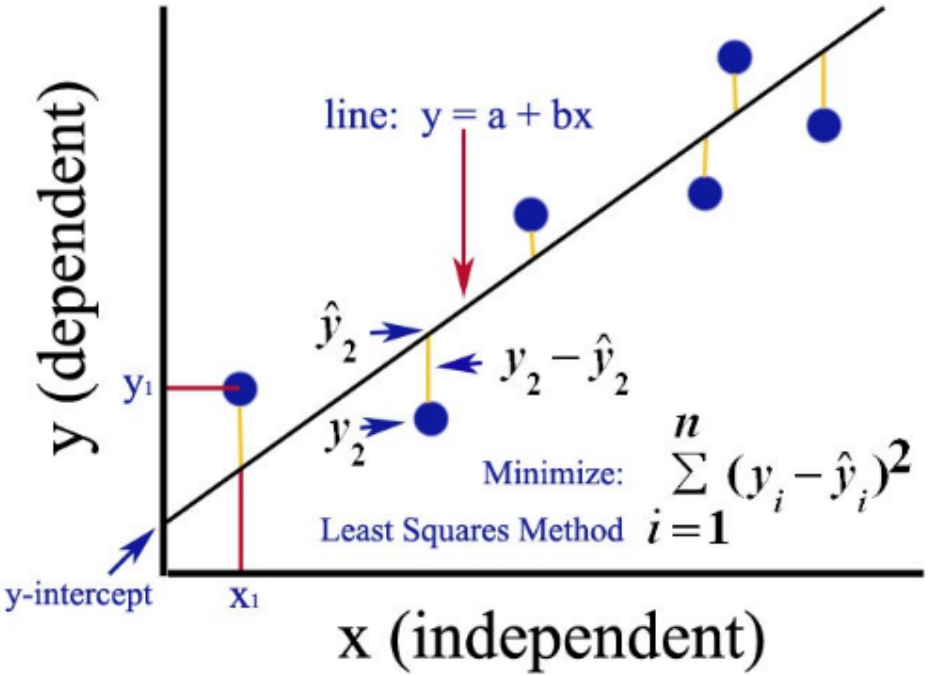
The linear regression



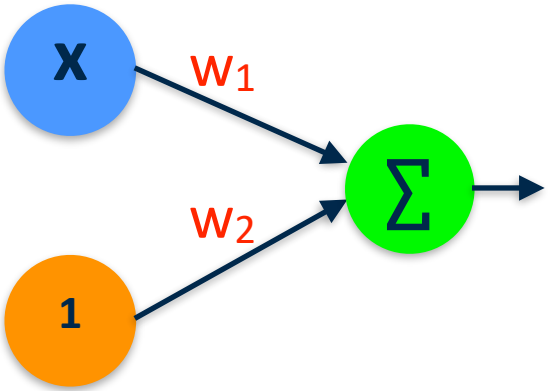
Let's just rethink it in this way:

Hands-on Machine Learning with Scikit-learn, Keras & Tensorflow, Aurélien Géron, 2nd edition

The linear regression

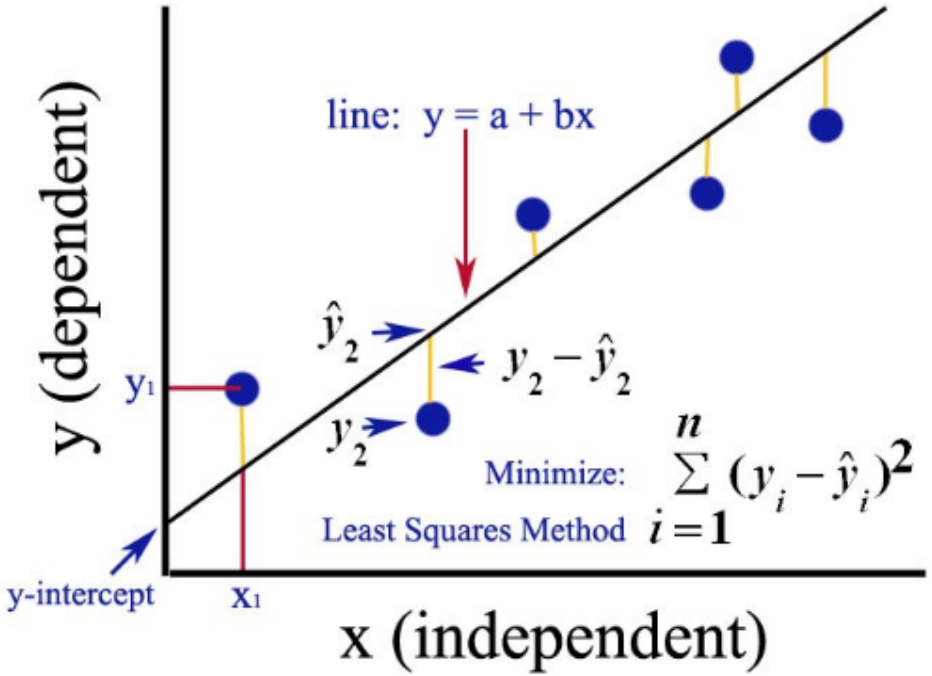


Let's just rethink it in this way:

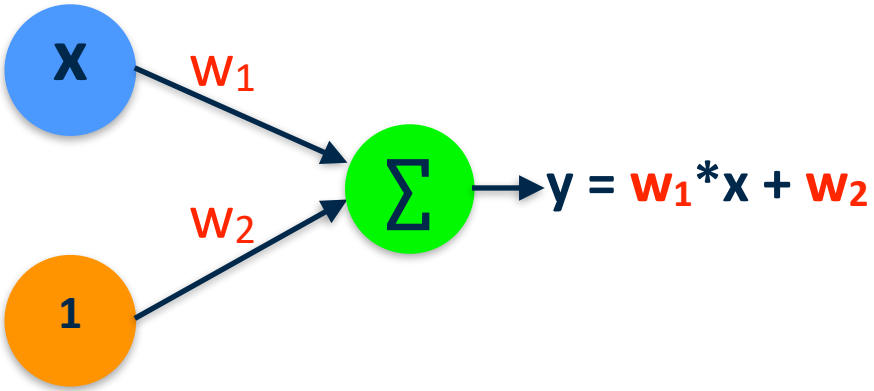


Hands-on Machine Learning with Scikit-learn, Keras & Tensorflow, Aurélien Géron, 2nd edition

The linear regression

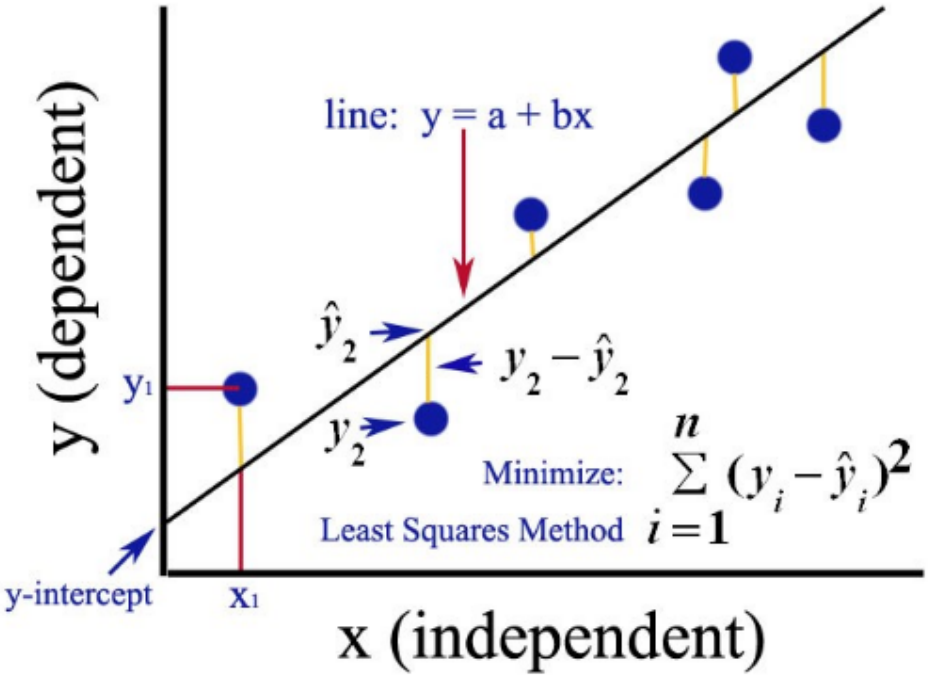


Let's just rethink it in this way:

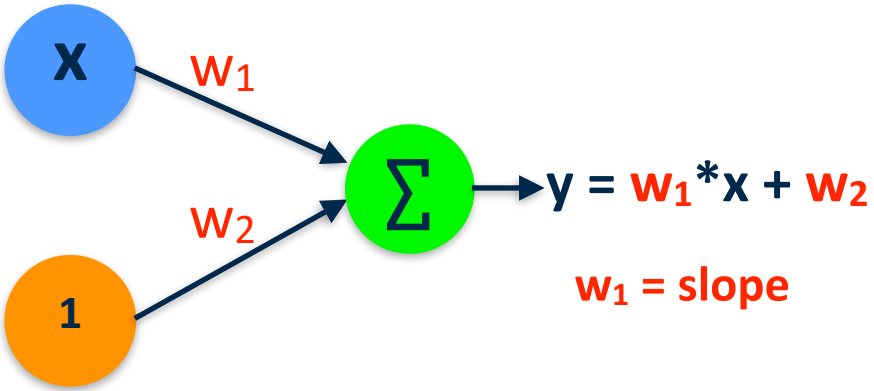


Hands-on Machine Learning with Scikit-learn, Keras & Tensorflow, Aurélien Géron, 2nd edition

The linear regression

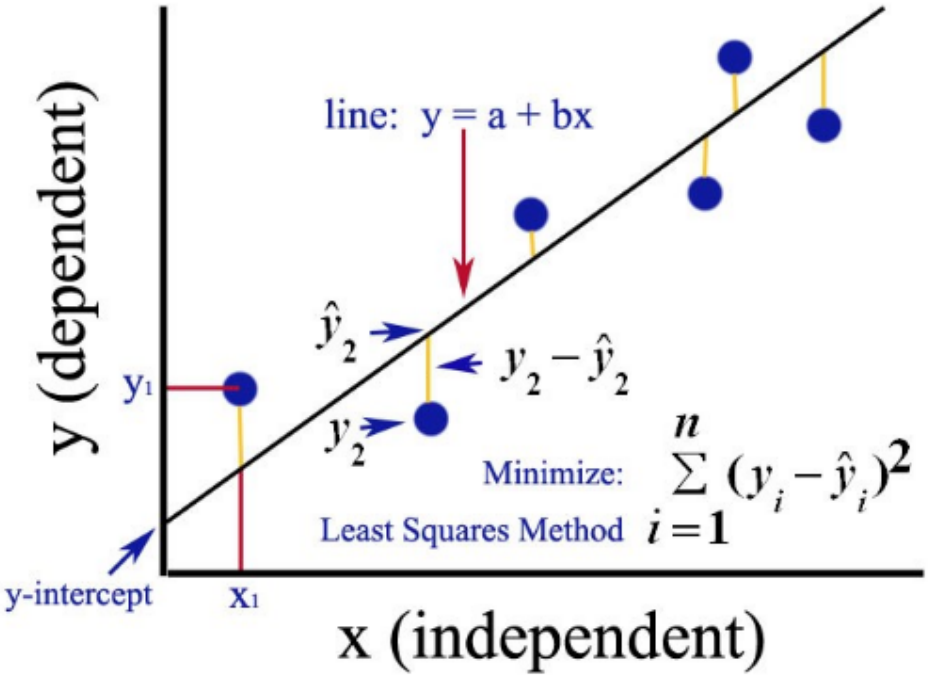


Let's just rethink it in this way:

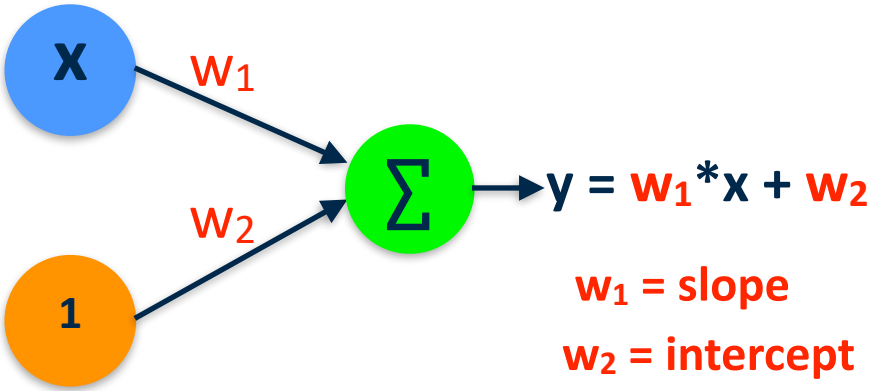


Hands-on Machine Learning with Scikit-learn, Keras & Tensorflow, Aurélien Géron, 2nd edition

The linear regression



Let's just rethink it in this way:



Hands-on Machine Learning with Scikit-learn, Keras & Tensorflow, Aurélien Géron, 2nd edition

Jupyter notebook
Linear models

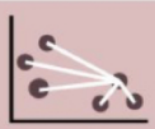
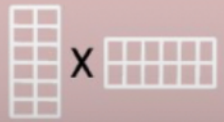
The machine-learning eco-system

Unsupervised learning

Clustering

- Hierarchical 
- K-means 

Dimensional reduction

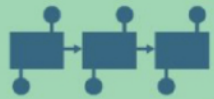
- PCA 
- tSNE 
- NMF 

Supervised Learning

"Machine Learning"

- SVM 
- KNN 
- Regression 
- Random forest 

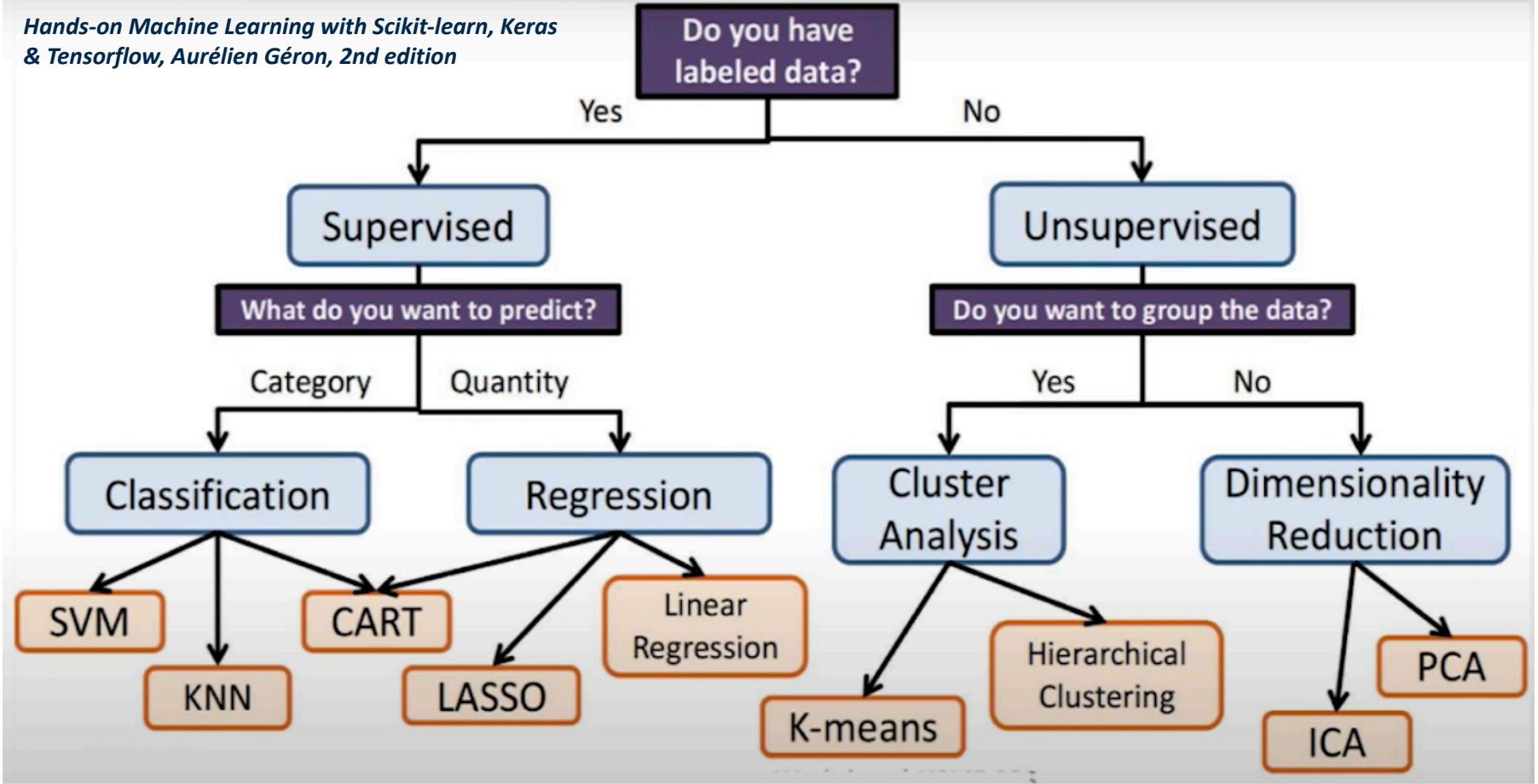
"Deep Learning"

- CNN 
- RNN 

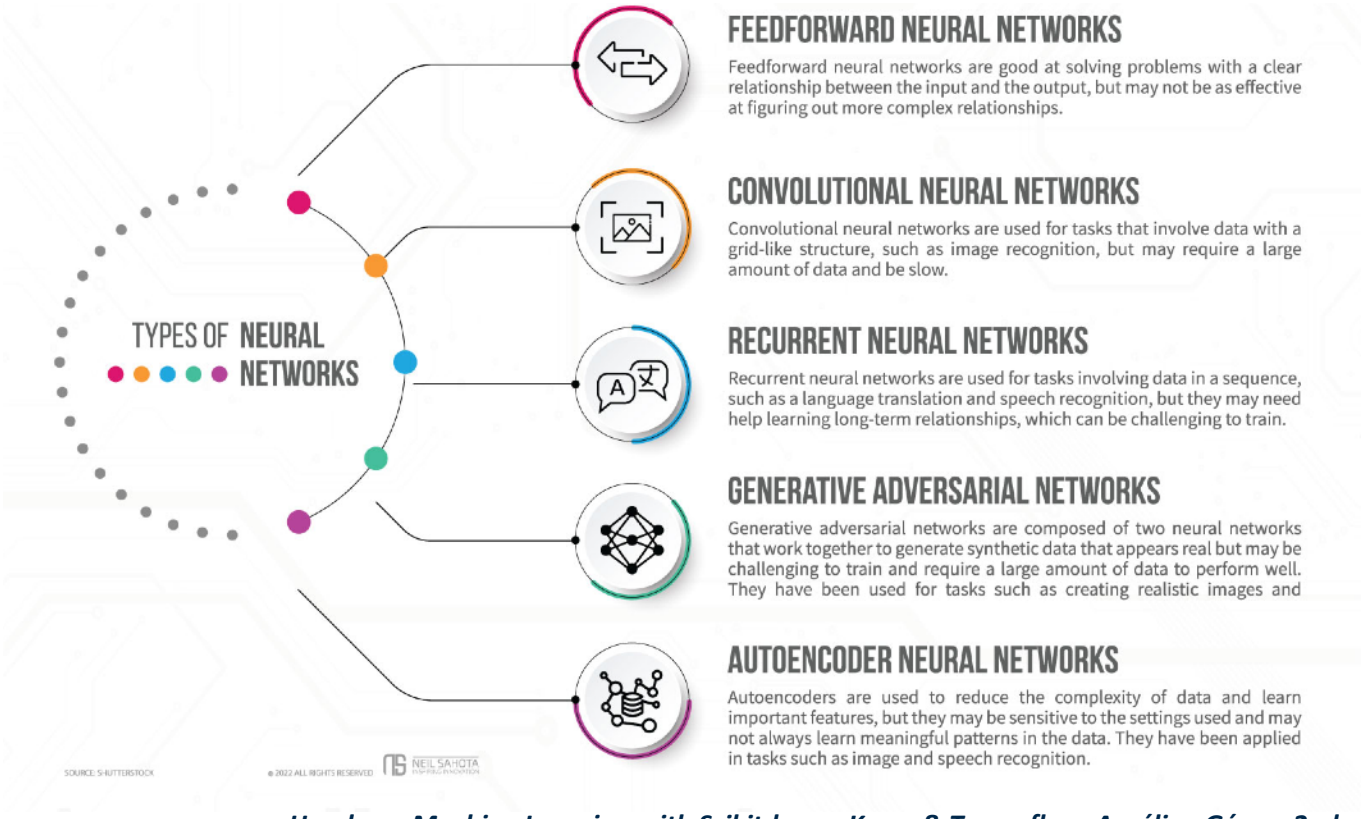
Hands-on Machine Learning with Scikit-learn, Keras & Tensorflow, Aurélien Géron, 2nd edition

The (« non-deep") machine-learning eco-system

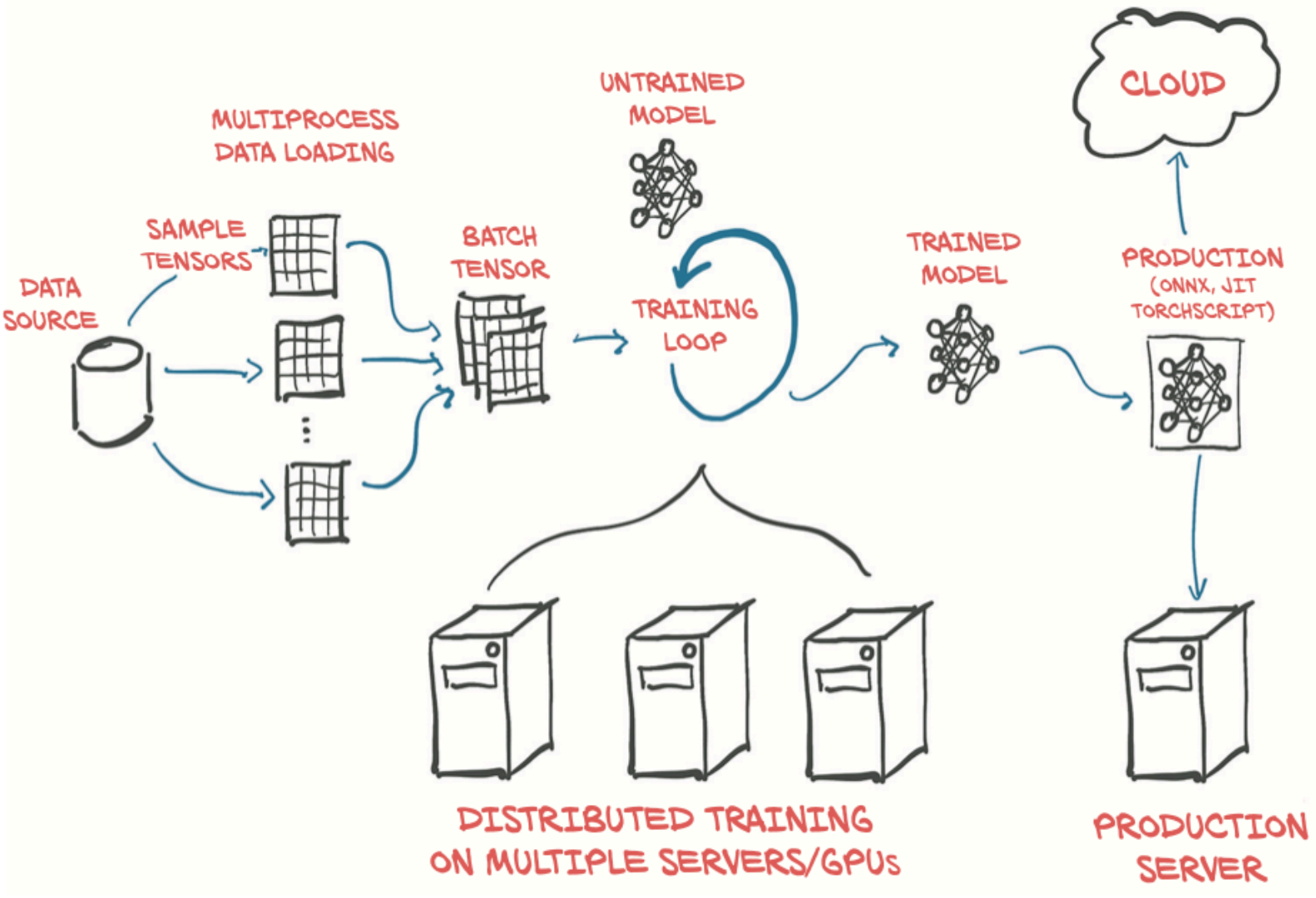
Hands-on Machine Learning with Scikit-learn, Keras & Tensorflow, Aurélien Géron, 2nd edition



Main types of Deep Learning networks



Hands-on Machine Learning with Scikit-learn, Keras & Tensorflow, Aurélien Géron, 2nd edition


CLOUD
UNTRAINED MODEL
MULTIPROCESS DATA LOADING
SAMPLE TENSORS
DATA SOURCE
BATCH TENSOR
TRAINING LOOP
TRAINED MODEL
PRODUCTION (ONNX, JIT TORCHSCRIPT)
DISTRIBUTED TRAINING ON MULTIPLE SERVERS/GPUs
PRODUCTION SERVER

The laws of Machine Learning

Deep Learning with Python, François Chollet, 2nd edition

The laws of Machine Learning

Deep Learning with Python, François Chollet, 2nd edition

The laws of Machine Learning

Deep Learning with Python, François Chollet, 2nd edition

The laws of Machine Learning

1) Only use Machine Learning if the classical approaches fail to solve your problems.

Deep Learning with Python, François Chollet, 2nd edition

The laws of Machine Learning

1) Only use Machine Learning if the classical approaches fail to solve your problems.

2) A neural network can only produce outputs that resemble the data it was trained on.

Deep Learning with Python, François Chollet, 2nd edition

The laws of Machine Learning

1) Only use Machine Learning if the classical approaches fail to solve your problems.

2) A neural network can only produce outputs that resemble the data it was trained on.

3) Data preparation is the most important part of the machine and deep learning approaches.

Deep Learning with Python, François Chollet, 2nd edition

The biological neural network

Neuron

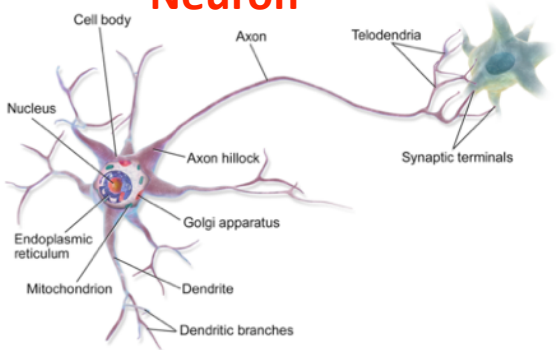


Image by Bruce Blaus (Creative Commons 3.0)
https://en.wikipedia.org/wiki/Neuron.

Biological neural network

Drawing of a cortical lamination by S. Ramon y Cajal (public domain).
https://en.wikipe dia.org/wiki/Cerebral_cortex

The biological neural network

Neuron

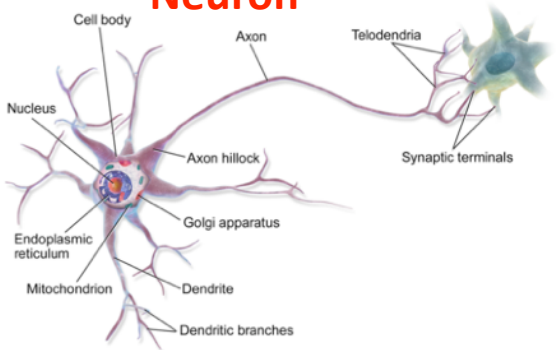


Image by Bruce Blaus (Creative Commons 3.0)
https://en.wikipedia.org/wiki/Neuron.

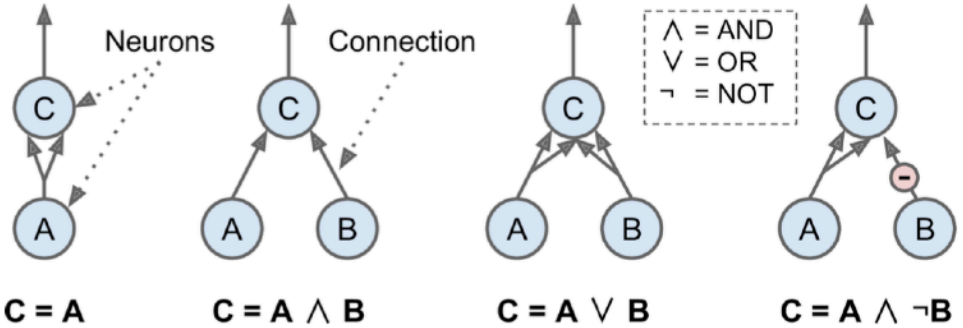
Biological neural network

Drawing of a cortical lamination by S. Ramon y Cajal (public domain).
https://en.wikipe dia.org/wiki/Cerebral_cortex

McCulloch and Walter Pitts proposed a model for biological neurons,which later became **artificial neurons**.

*McCulloch, Warren S.; Pitts, Walter (1943). Bulletin of Mathematical Biophysics. **5** (4): 115–133.*

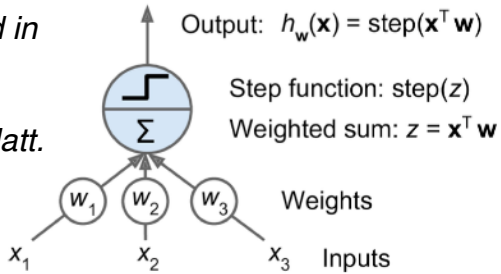
ANNs performing simple logical computations



TLU, Perceptron, Multi-layer perceptron

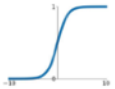
Threshold logic unit (TLU)

invented in 1957 by Frank Rosenblatt.

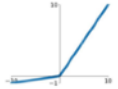


Activation Functions

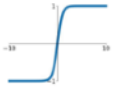
Sigmoid
$\sigma(x) = \frac{1}{1+e^{-x}}$



Leaky ReLU
$\max(0.1x, x)$

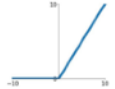


tanh
$\tanh(x)$

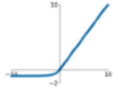


Maxout
$\max(w_1^T x + b_1, w_2^T x + b_2)$

ReLU
$\max(0, x)$



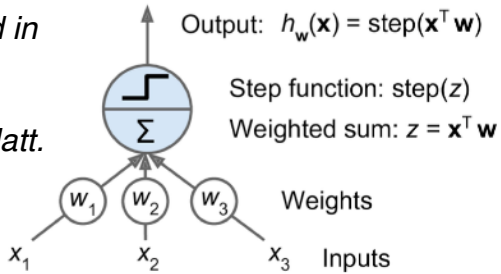
ELU
$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$

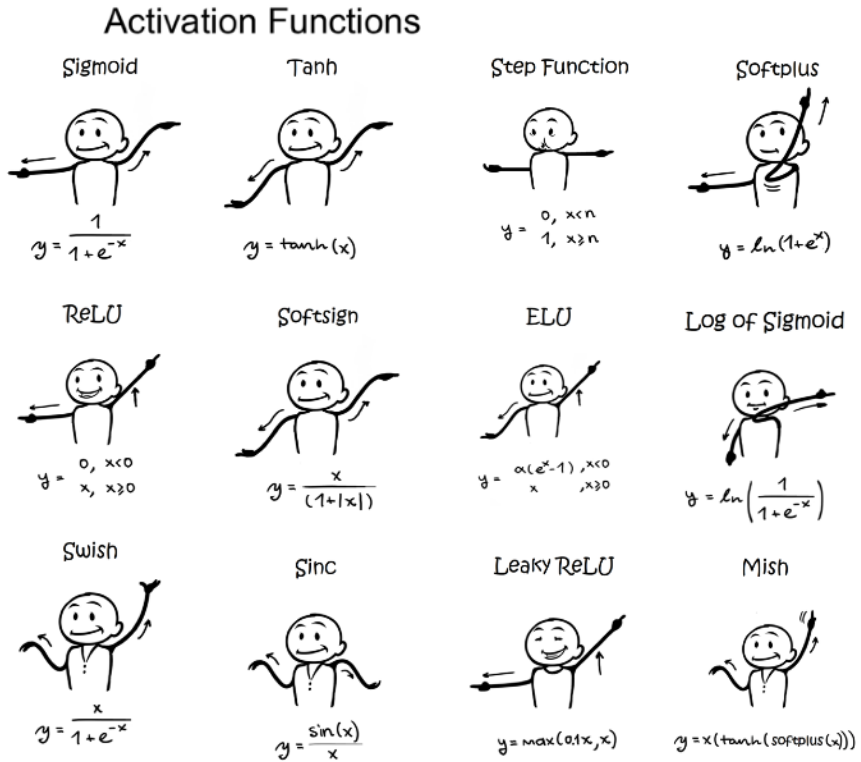


TLU, Perceptron, Multi-layer perceptron

Threshold logic unit (TLU)

invented in 1957 by Frank Rosenblatt.

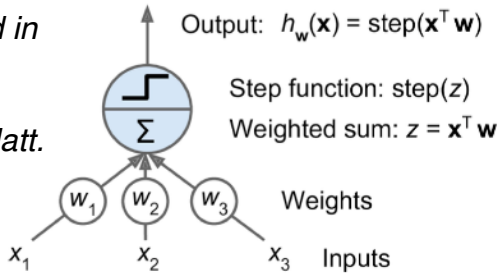




TLU, Perceptron, Multi-layer perceptron

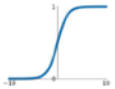
Threshold logic unit (TLU)

invented in 1957 by Frank Rosenblatt.

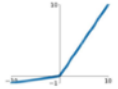


Activation Functions

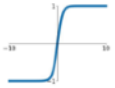
Sigmoid
$\sigma(x) = \frac{1}{1+e^{-x}}$



Leaky ReLU
$\max(0.1x, x)$

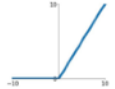


tanh
$\tanh(x)$

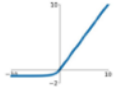


Maxout
$\max(w_1^T x + b_1, w_2^T x + b_2)$

ReLU
$\max(0, x)$



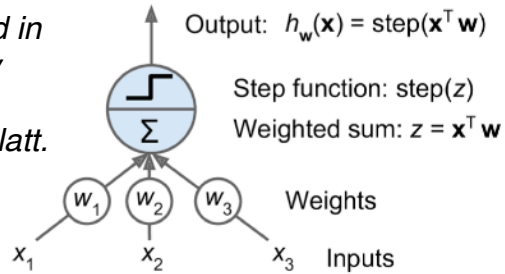
ELU
$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$



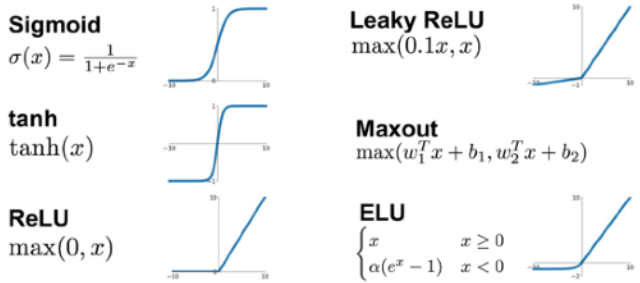
TLU, Perceptron, Multi-layer perceptron

Threshold logic unit (TLU)

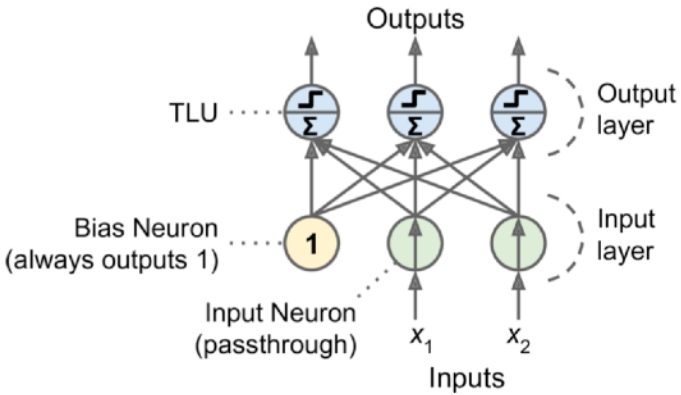
invented in 1957 by Frank Rosenblatt.



Activation Functions



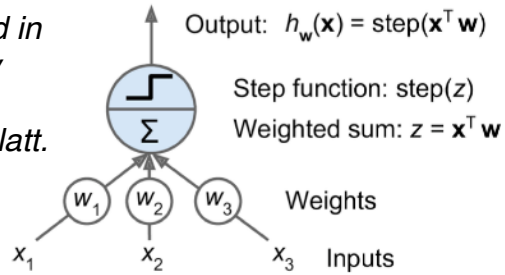
Perceptron (single layer of TLUs)



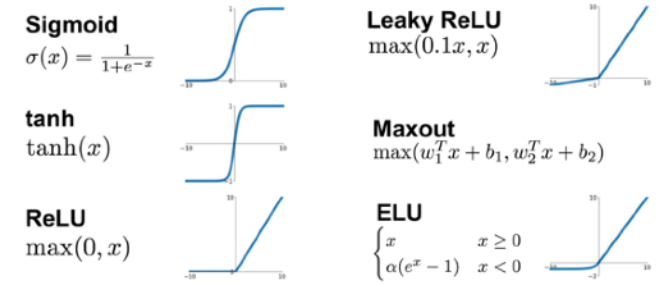
TLU, Perceptron, Multi-layer perceptron

Threshold logic unit (TLU)

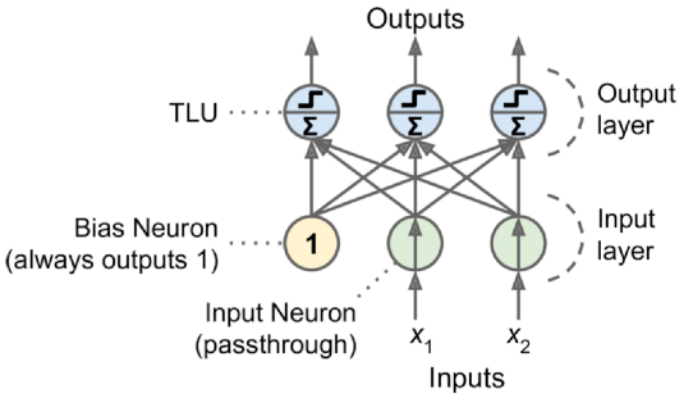
invented in 1957 by Frank Rosenblatt.



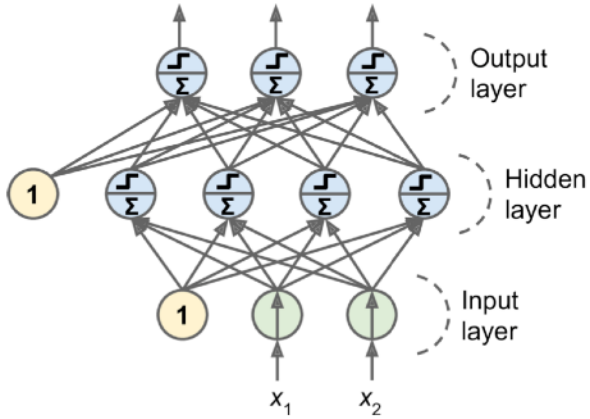
Activation Functions



Perceptron (single layer of TLUs)



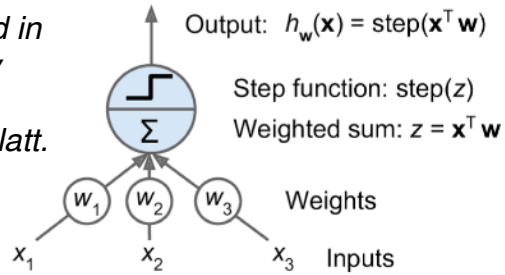
Multi-layer perceptron (deep neural network)



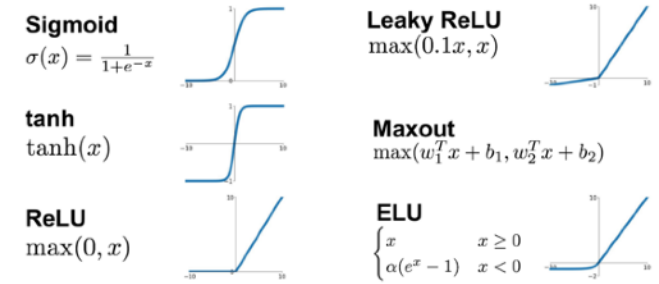
TLU, Perceptron, Multi-layer perceptron

Threshold logic unit (TLU)

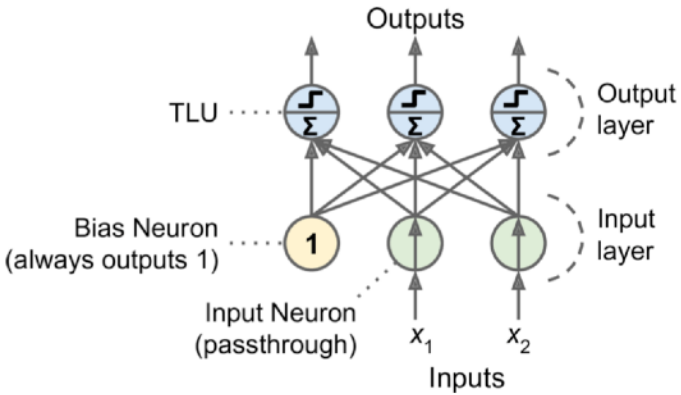
invented in 1957 by Frank Rosenblatt.



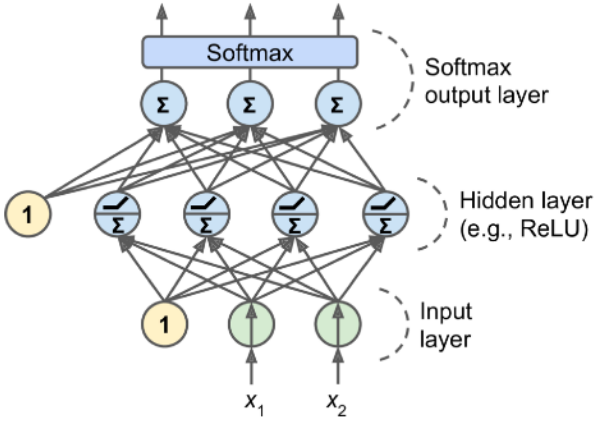
Activation Functions



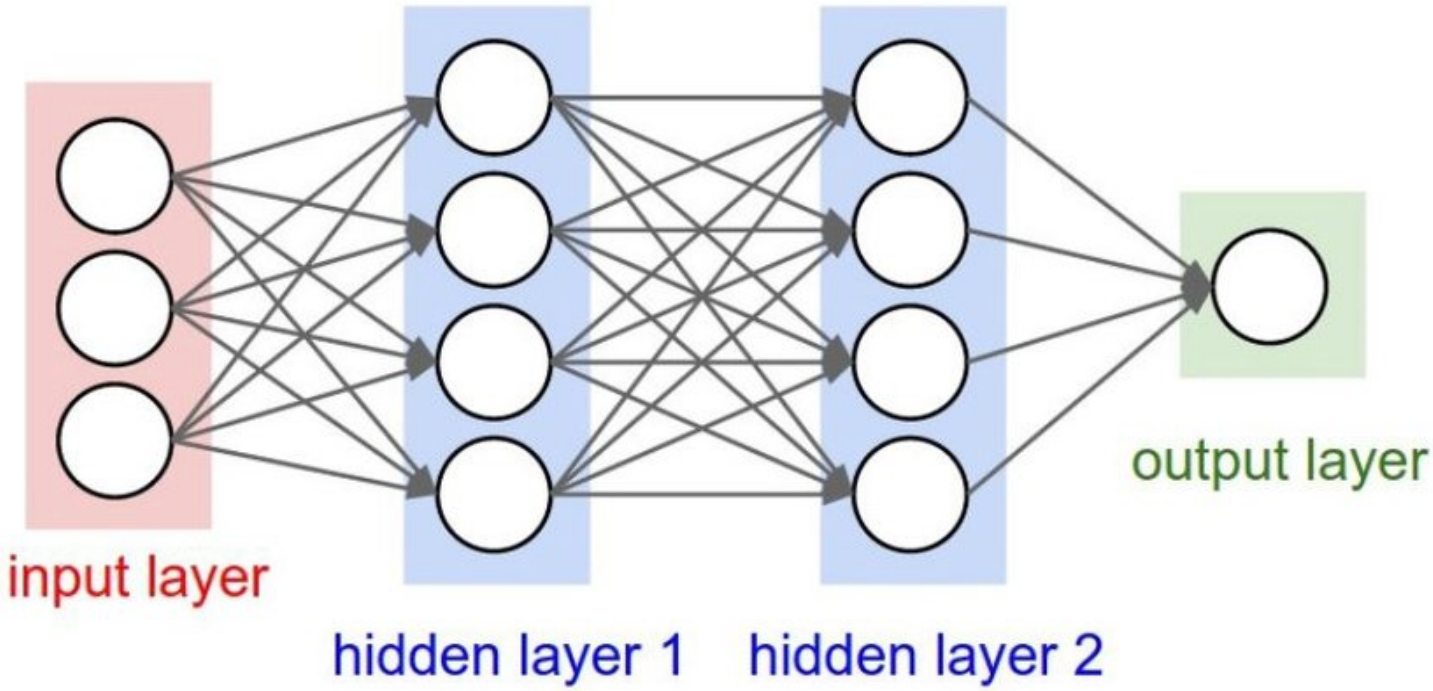
Perceptron (single layer of TLUs)



Multi-layer perceptron (deep neural network)

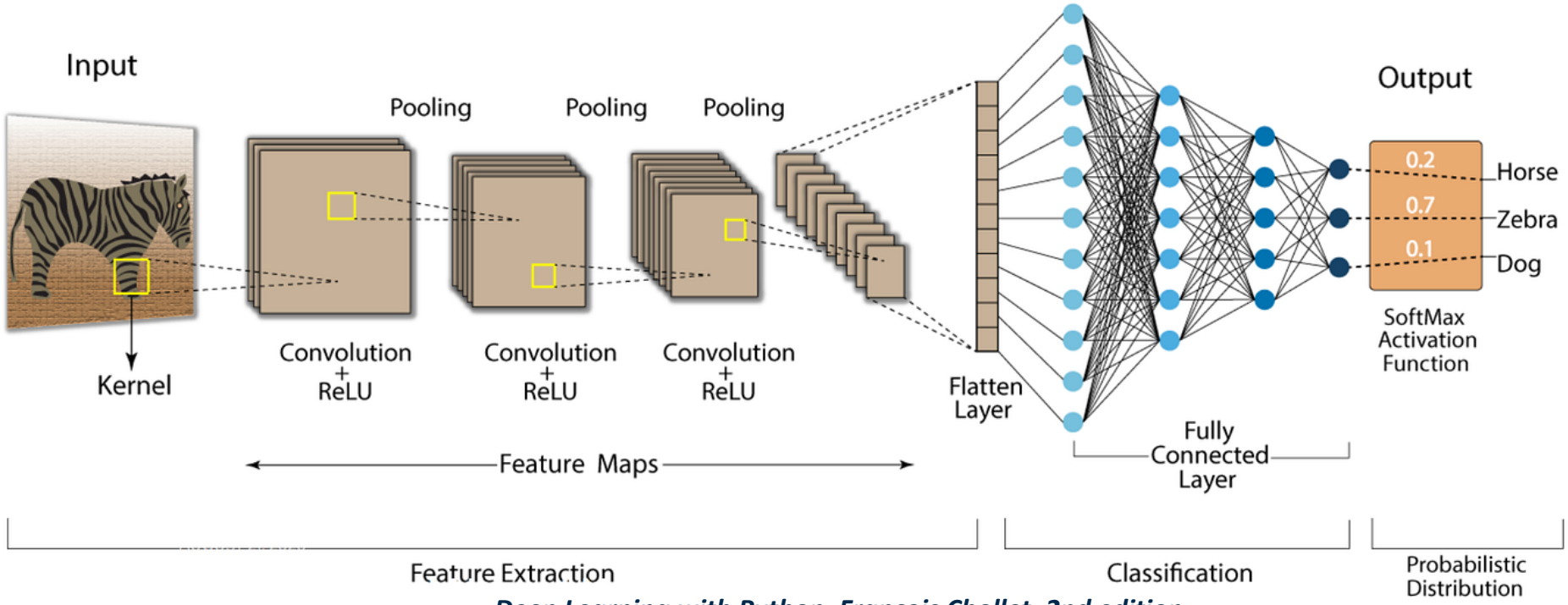


Dense Neural Networks



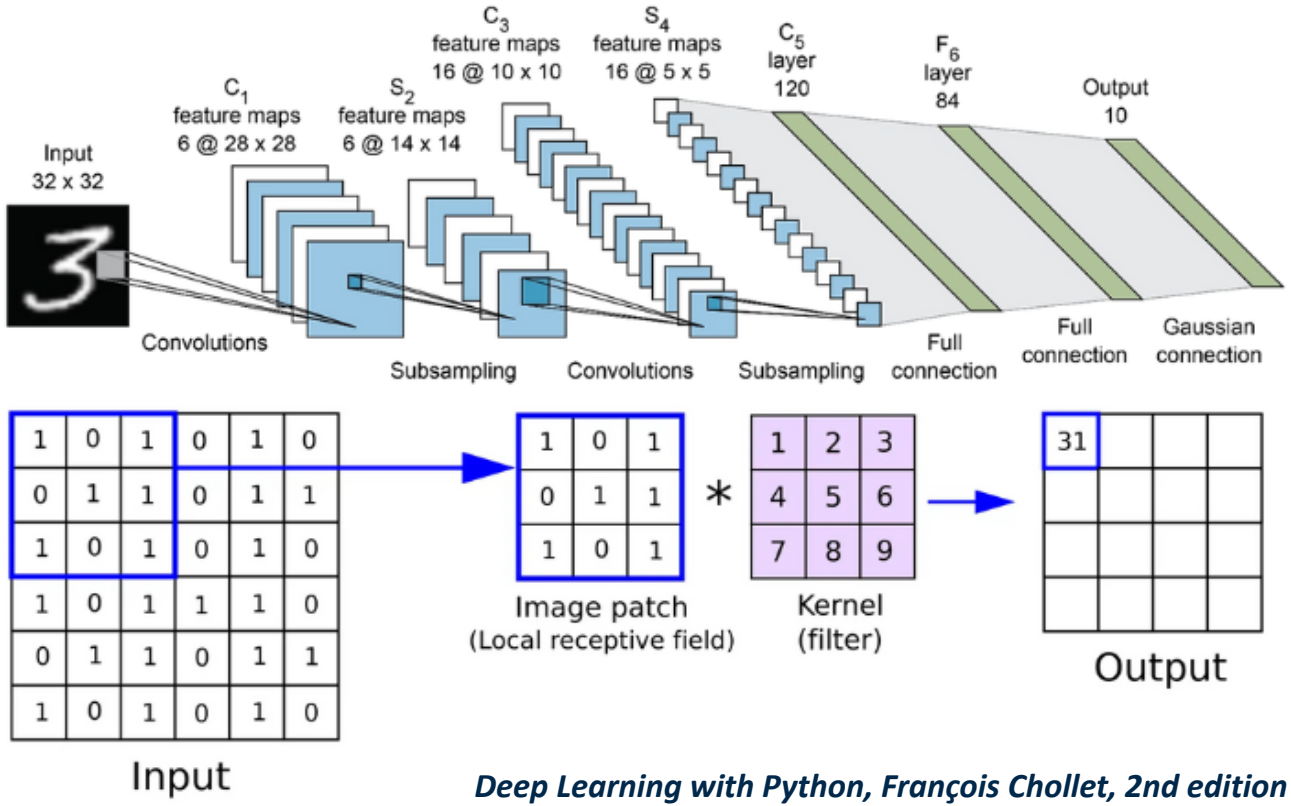
Jupyter notebook
Clothes classification

Convolutional Neural Networks (CNN)



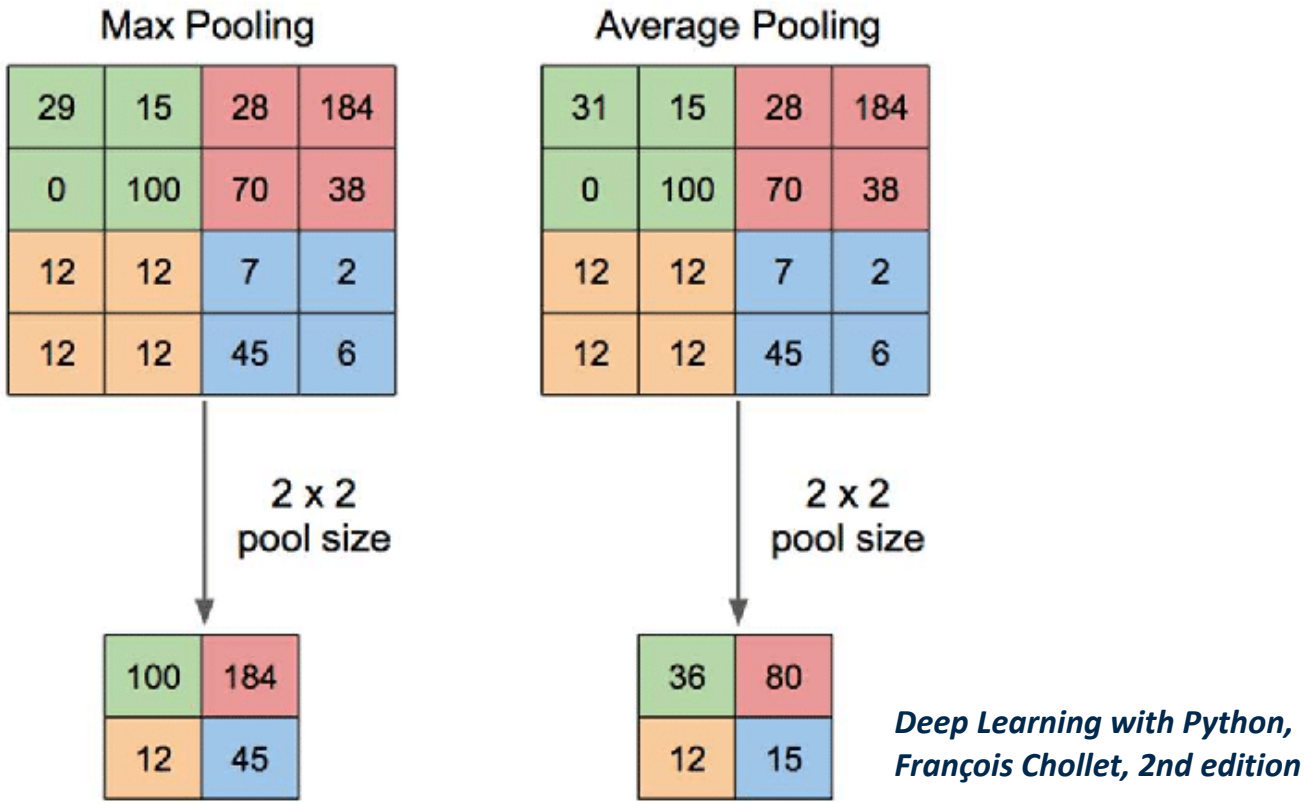
Deep Learning with Python, François Chollet, 2nd edition

The convolution step



Deep Learning with Python, François Chollet, 2nd edition

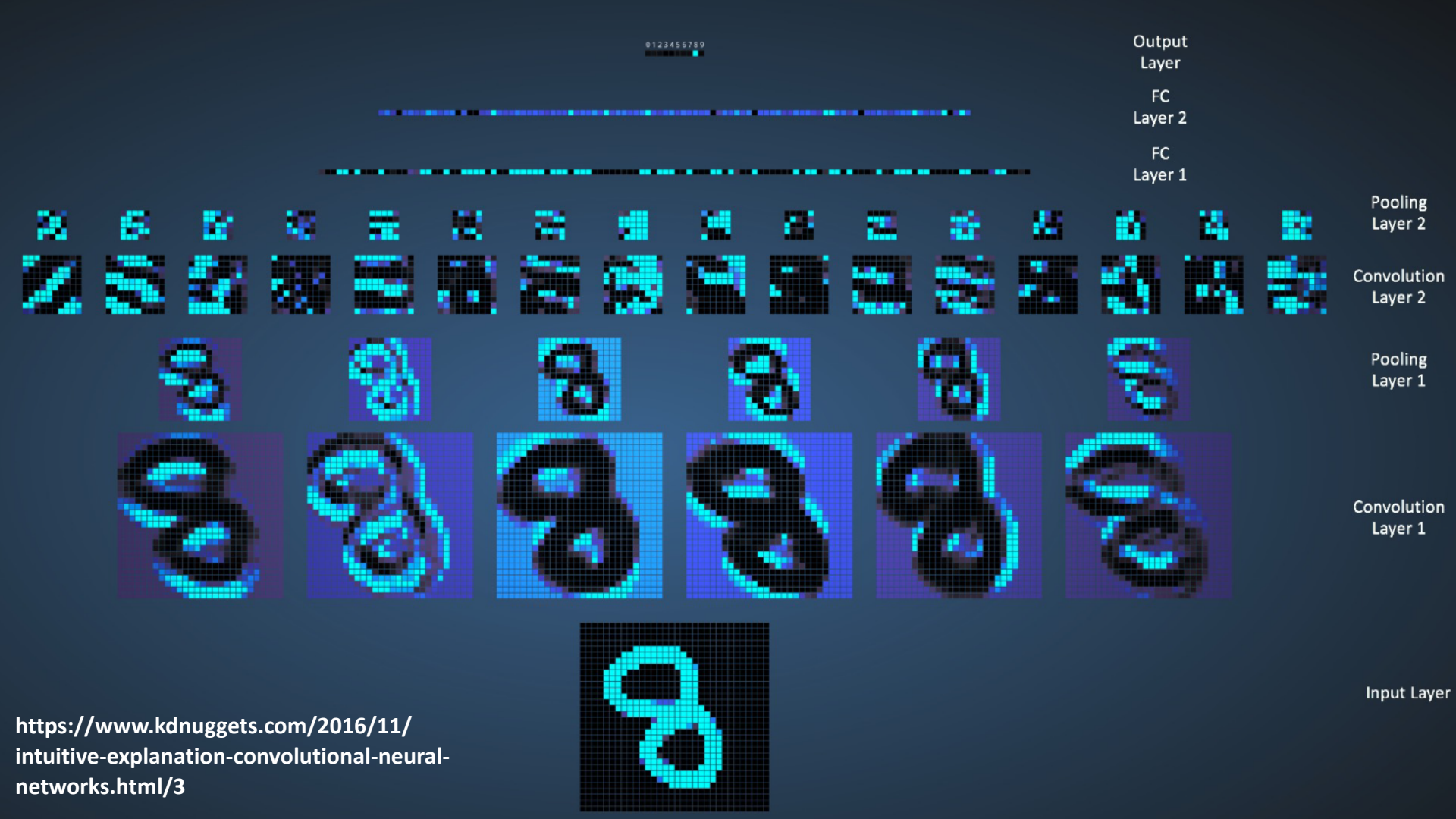
Max Pooling / Average Pooling



Deep Learning with Python, François Chollet, 2nd edition


TIPO EXCELSIUS


TIPO EXCELSIUS

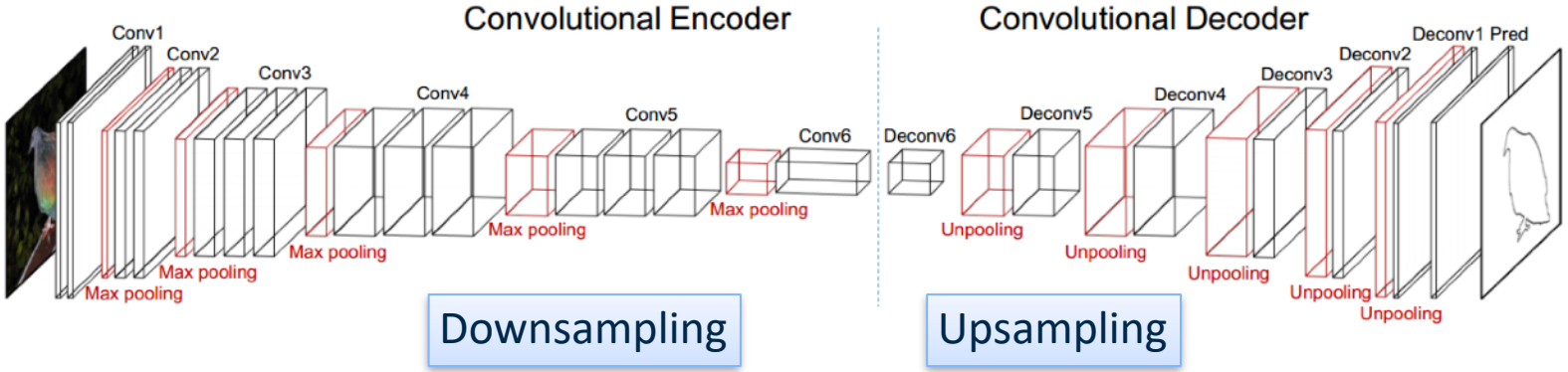


https://www.kdnuggets.com/2016/11/intuitive-explanation-convolutional-neural-networks.html/3

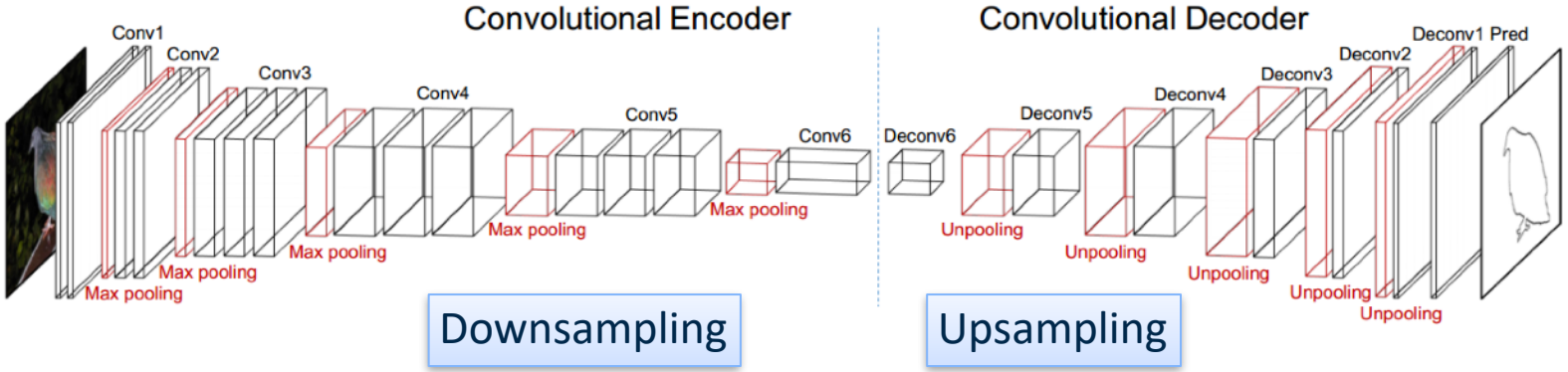
Jupyter notebook
Clothes classification with CNN

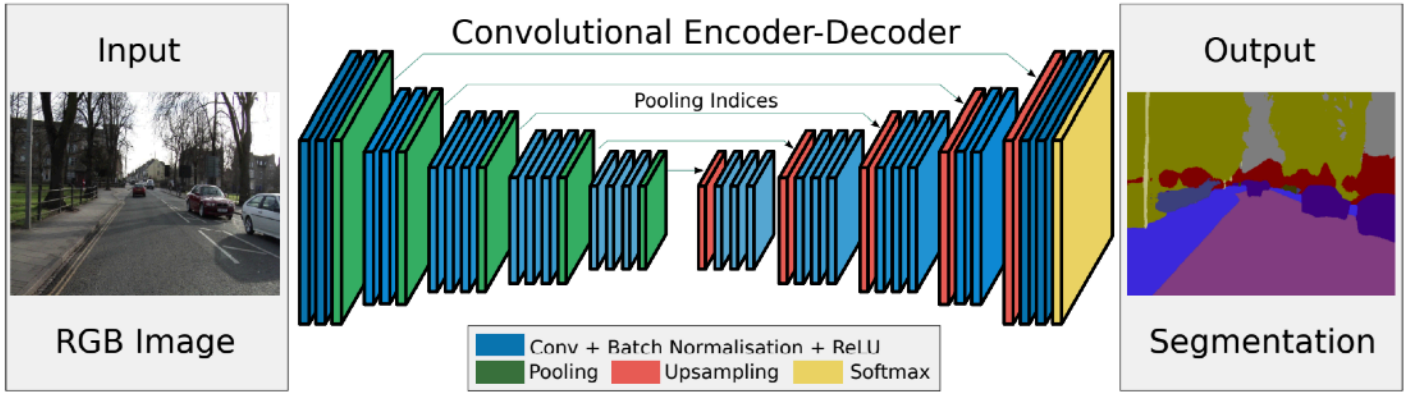
Jupyter notebook
Dogs and cats with and without data augmentation

Encoder-Decoder network

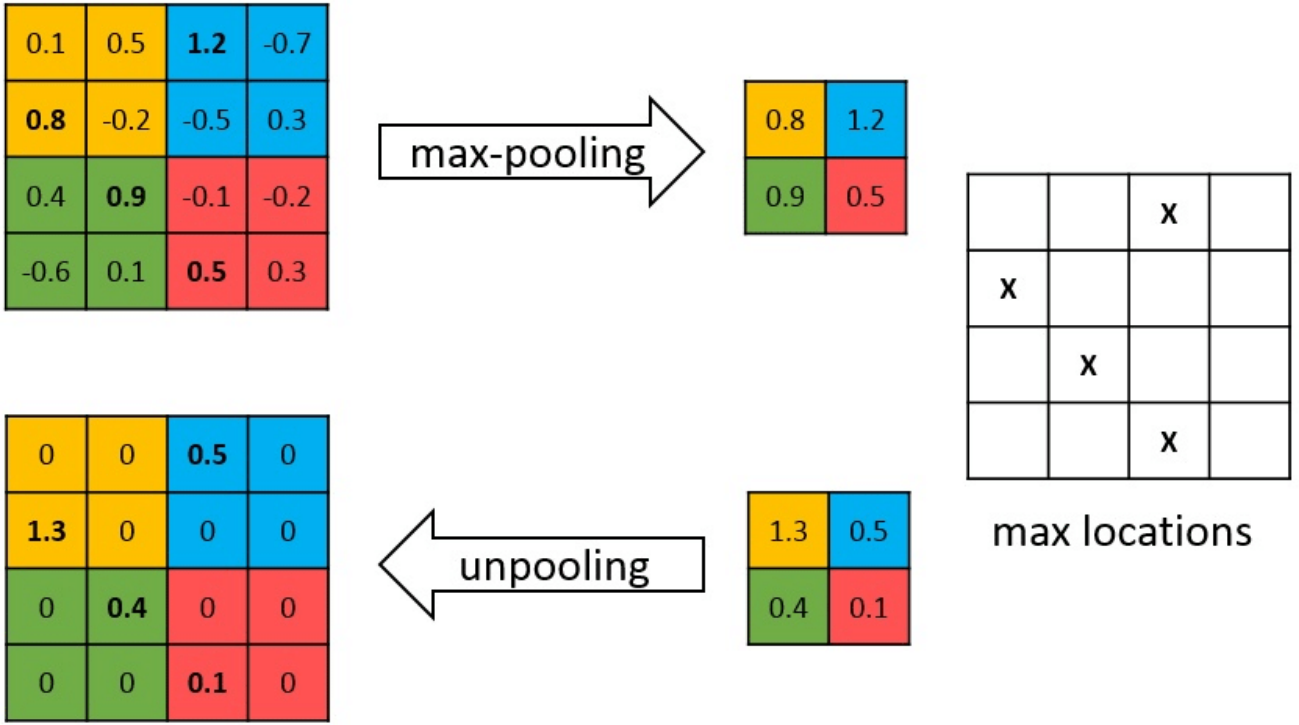


Encoder-Decoder network



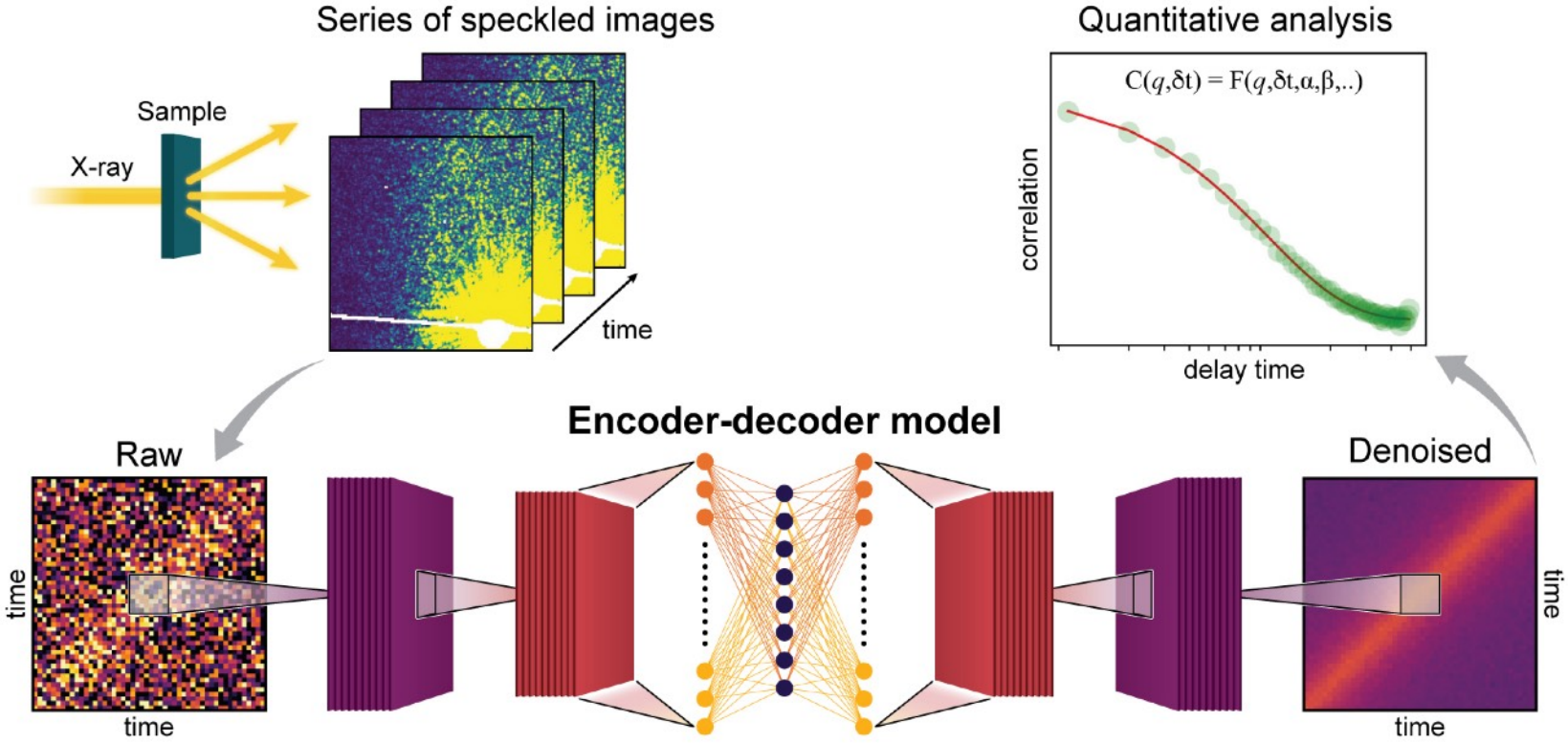


Max Pooling / Unpooling



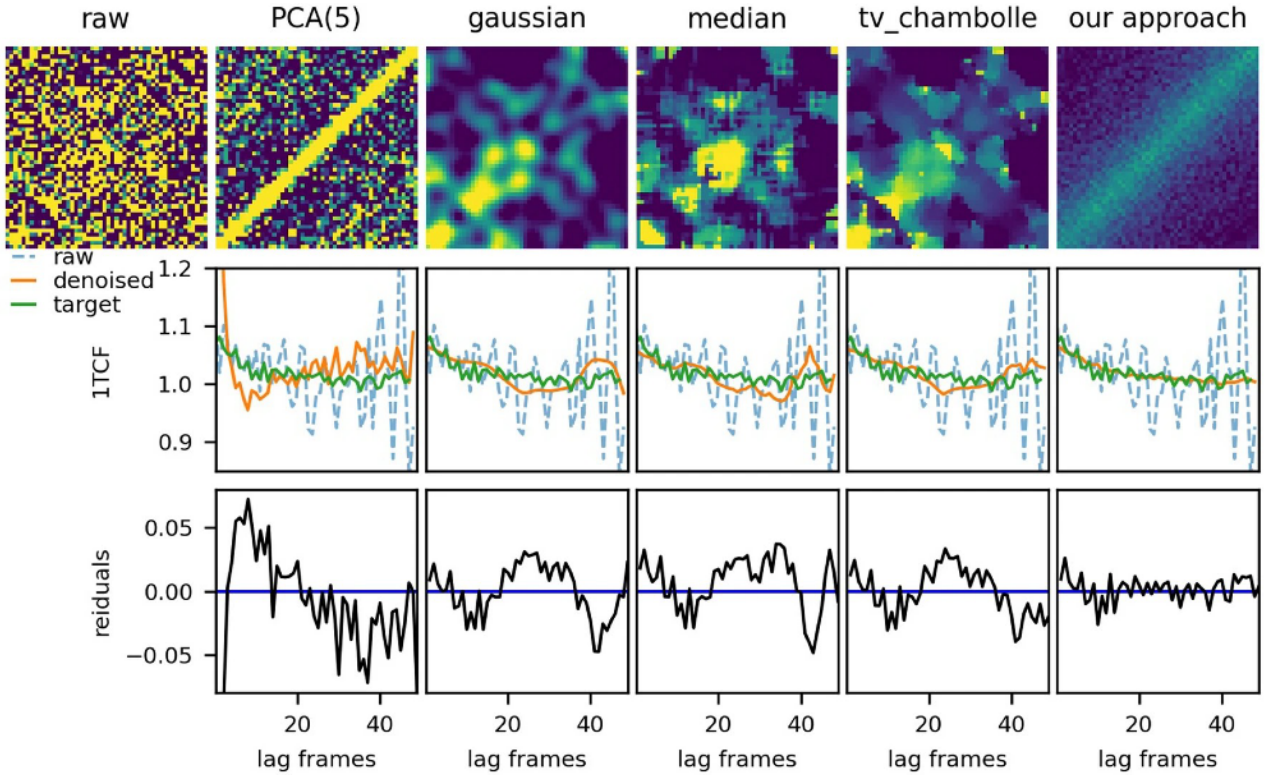
Deep Learning with Python, François Chollet, 2nd edition

Encoder - decoder for XPCS data denoising



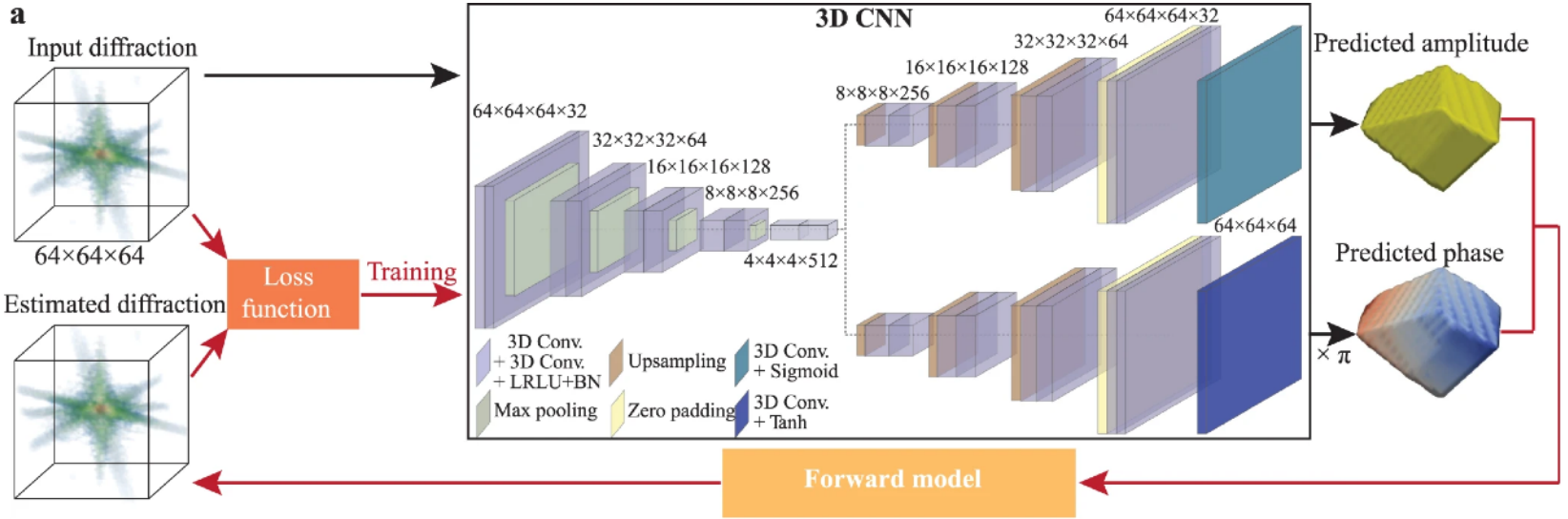
T. Konstantinova *et al. Sci Rep* 11, 14756 (2021)

Encoder - decoder for XPCS data denoising



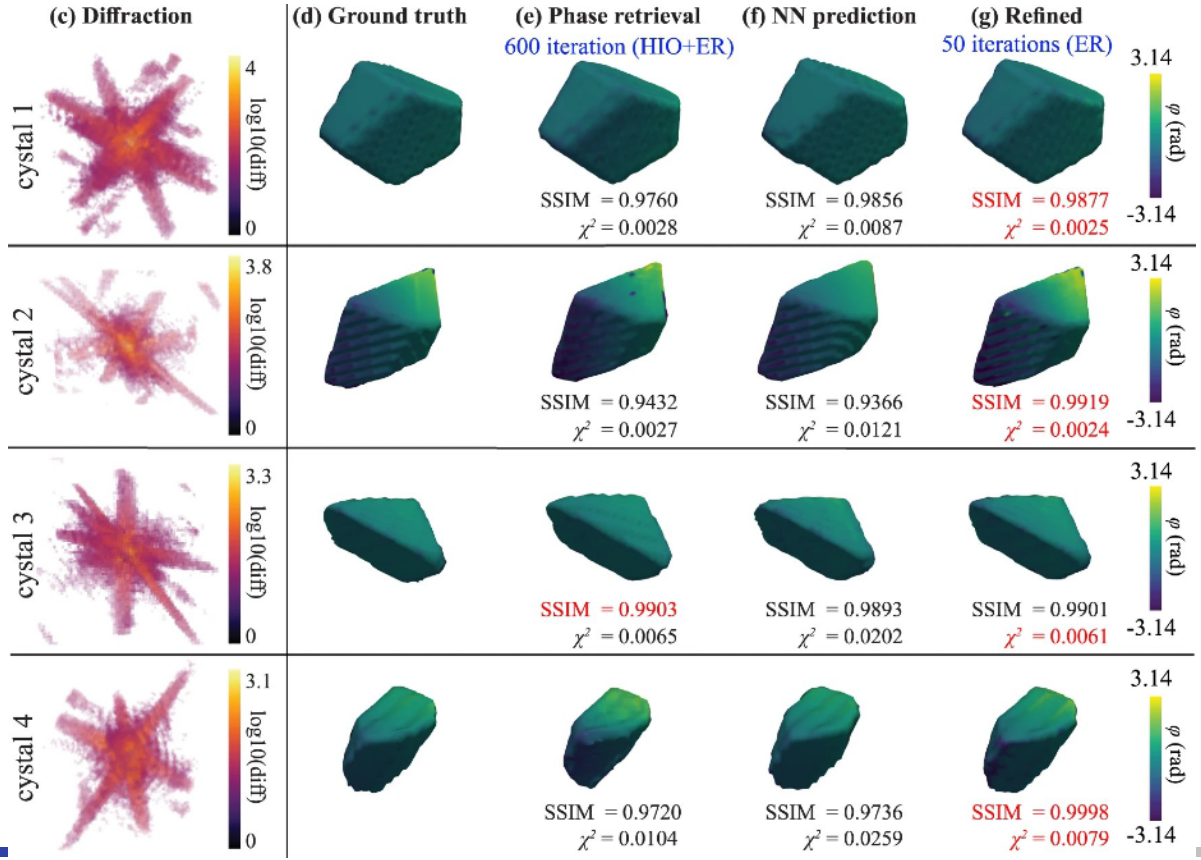
T. Konstantinova *et al. Sci Rep* 11, 14756 (2021)

AutoPhaseNN for Bragg CDI



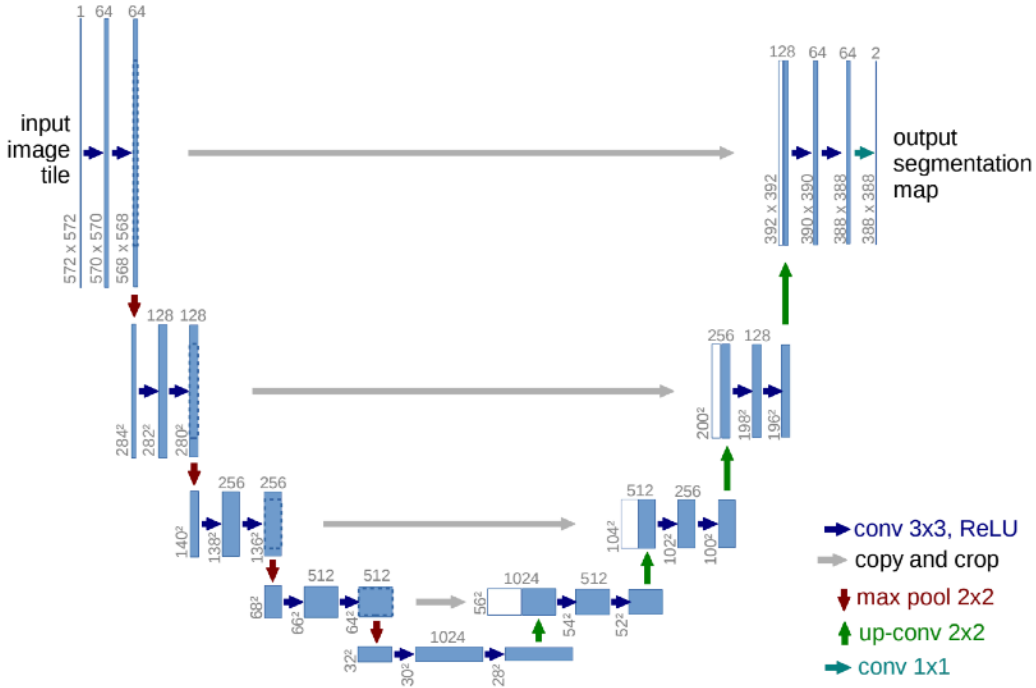
Y. Yao *et al.*, *npj Comput Mater* 8, 124 (2022)

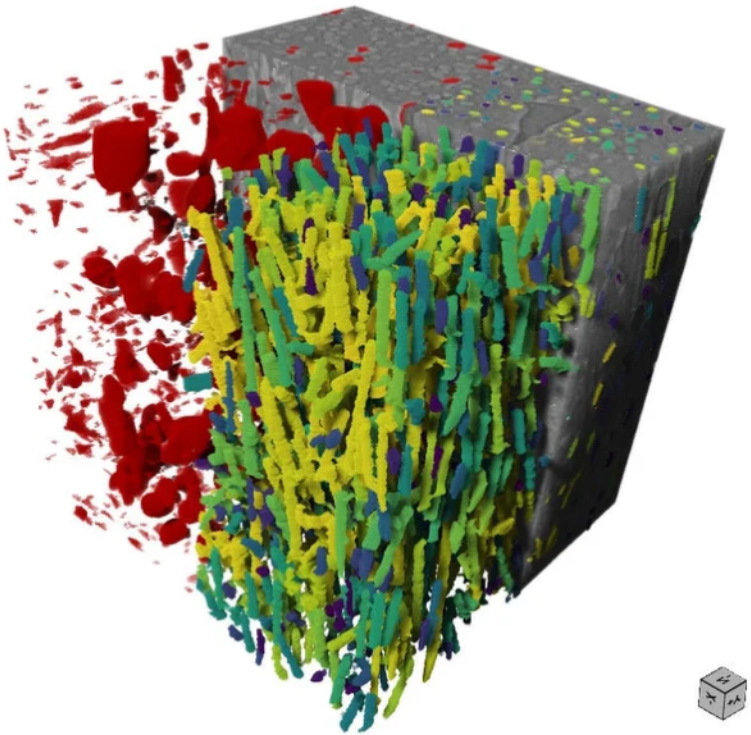
AutoPhaseNN for Bragg CDI



Y. Yao *et al.*, *npj Comput Mater* 8, 124 (2022)

The U-NET Architecture





O. Ronneberger et al. (2015). MICCAI 2015. Lecture Notes in Computer Science, vol 9351. Springer

Deep learning for 2D and 3D image processing

dragonfly

Non-Commercial Licensing

We proudly support academic research...

DRAGONFLY FREE 30-DAY TRIAL

Deep Learning for Imaging Scientists

This video provides the background details for a better understanding of Dragonfly's Deep Learning Tool and how to train deep models for denoising, image segmentation, and super-resolution. Topics include the comparison of image processing to linear regression, fitting and applying functions, perceptron and neural networks as functions, as well as model training and optimization.

References and data examples:

Go to Grant Sanderson's YouTube Channel, 3Brown1Blue, for a deeper explanation on neural networks and how they are trained.

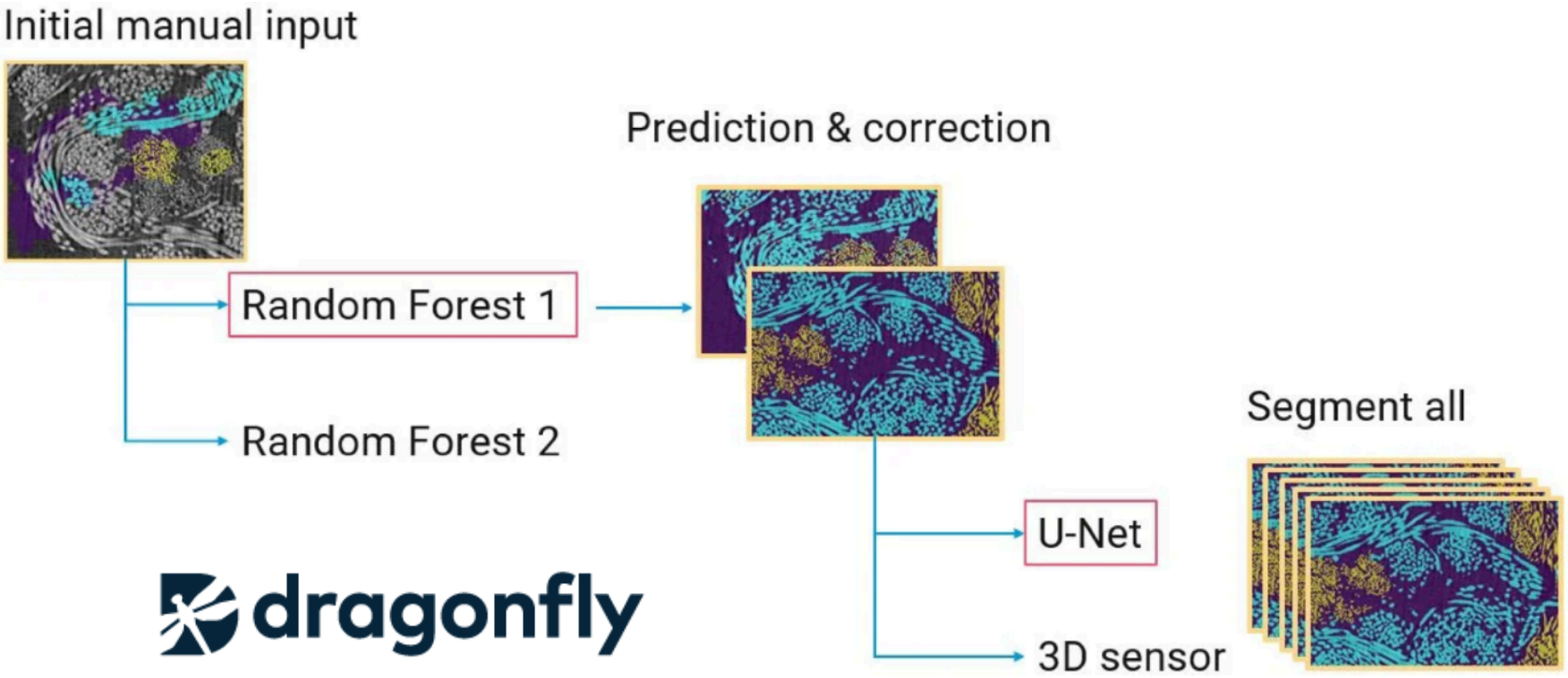
Pharmaceutical tablets example from Ma et al. 2020.

Rat neurons example from Eustaquio et al. 2018.

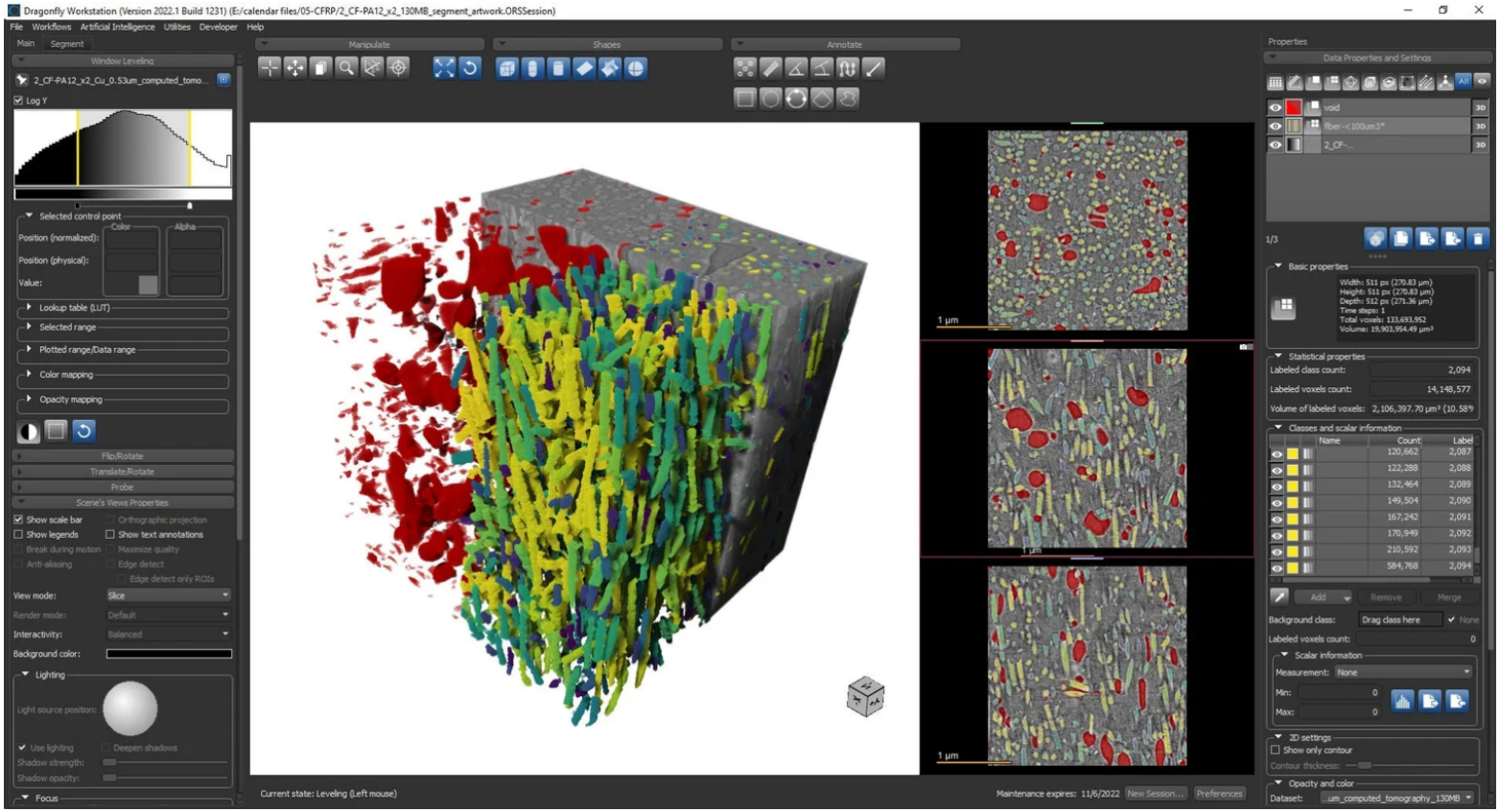
https://www.theobjects.com/dragonfly/deep-learning-getting-started.html


Deep Learning Tool
Model:
U-Net_millet
Train All Layers
Freeze All Layers
Duplicate Layer
Delete Selected
Select New Layer Type
Filter
Attribute
Value
Add New Layer
dragonfly
Graph Layout:
Vertical
Horizontal
Arrange Layers by:
Position
Size
Reorganize Graph
Fit Graph to View
Input Layer
Output layer
Back to Model Overview
Go to Editing
Go to Training
Reset
Reload
Save
Close

Deep learning for 2D and 3D image processing

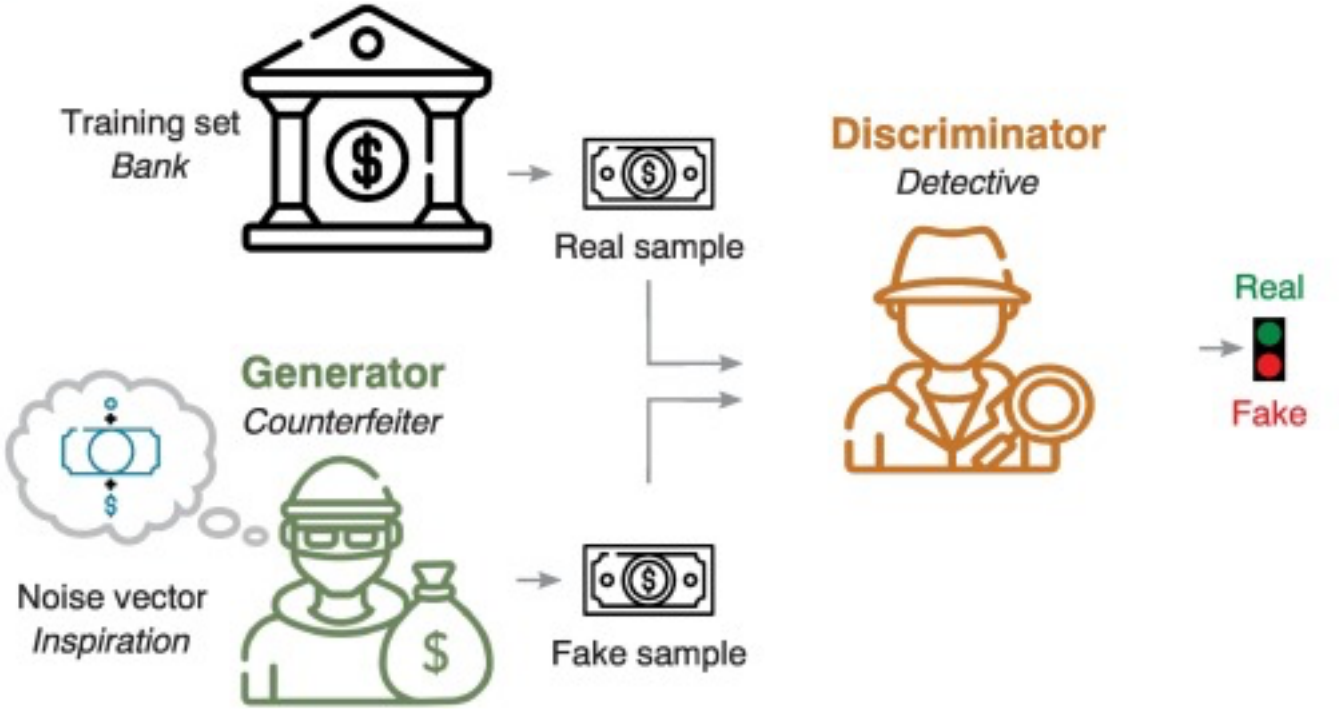


Deep learning for 2D and 3D image processing

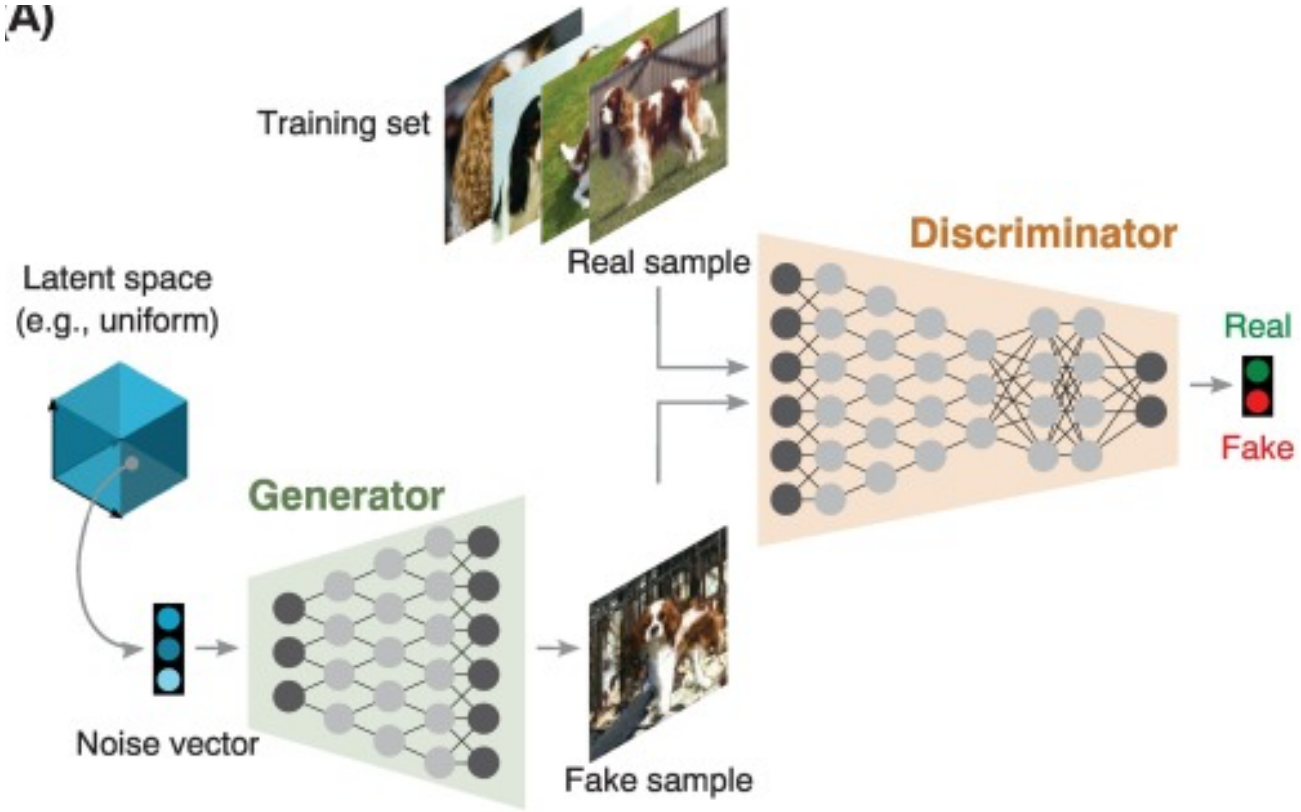


dragonfly

GAN - Generative Adversarial Network



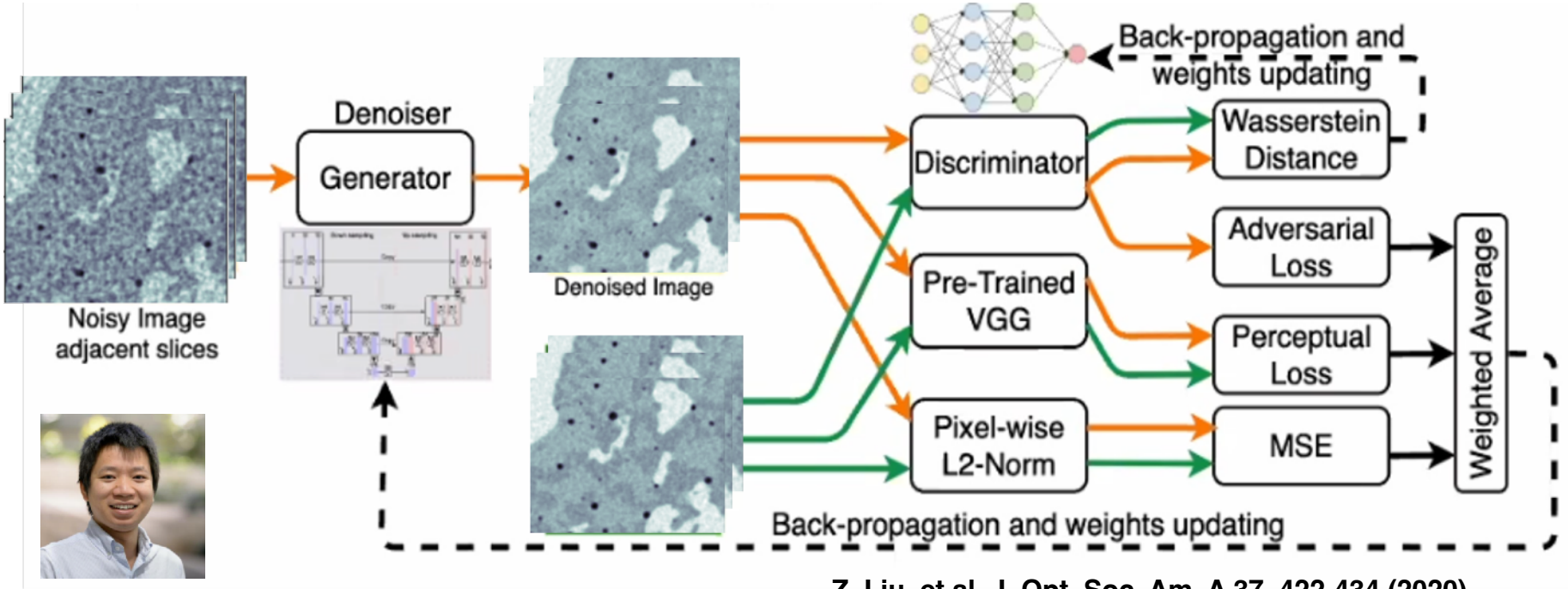
GAN - Generative Adversarial Network



Jupyter notebook
Fake Faces with GAN

Fake Faces - some results after 30 epochs

The architecture of TomoGAN



Z. Liu, et al. J. Opt. Soc. Am. A 37, 422-434 (2020)

Zhengchun Liu

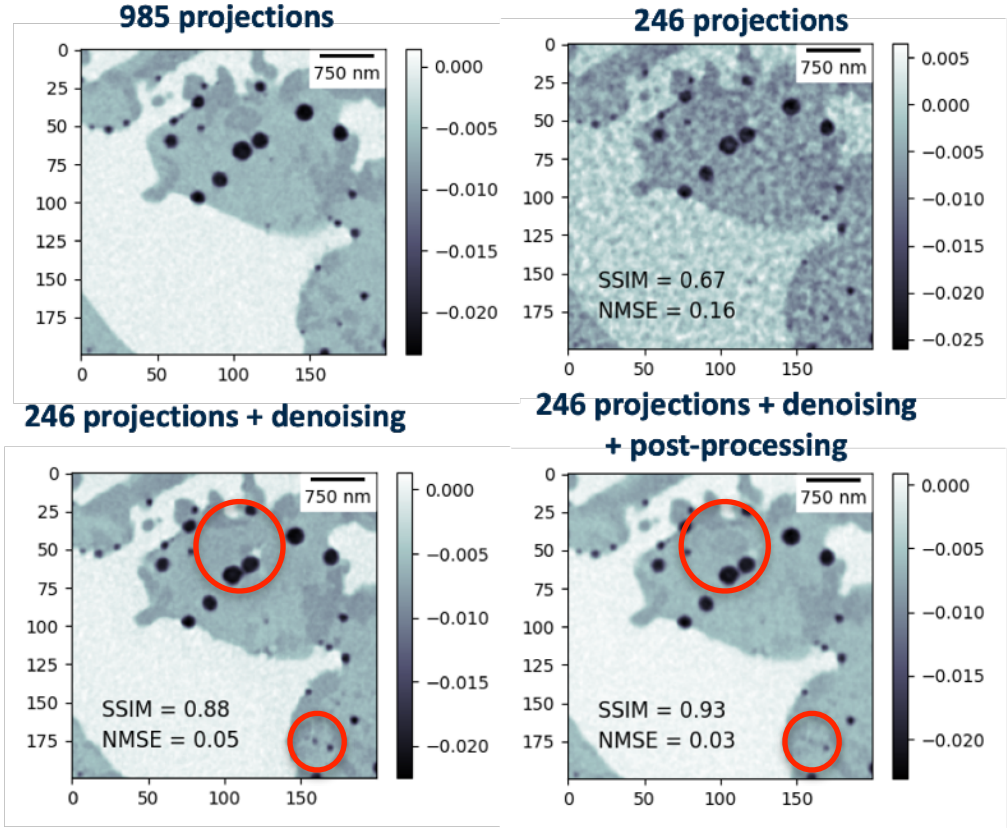
Argonne NATIONAL LABORATORY

TomoGAN network: https://github.com/lzhengchun/TomoGAN

Tutorial on denoising: https://github.com/lzhengchun/dn-tutorial/tree/main

Jupyter notebook
Denoising with TomoGAN

Application of TomoGAN

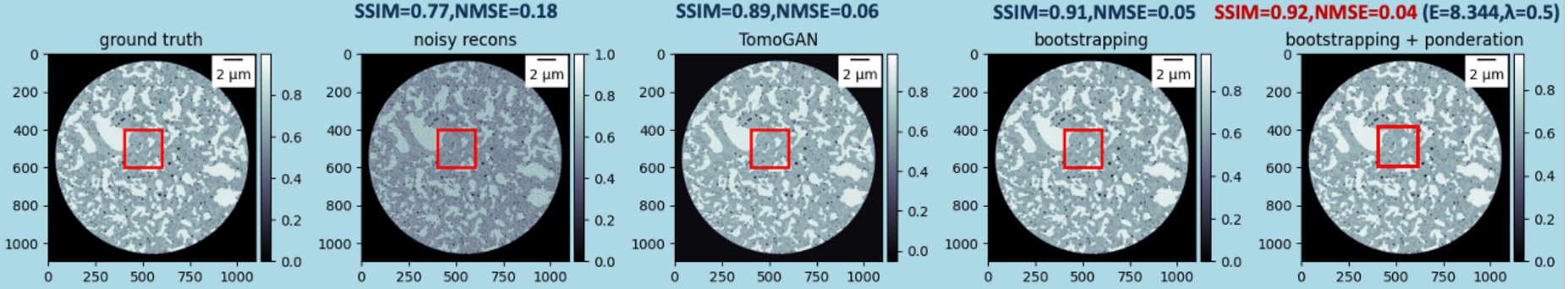


M. Redhouane Boudjehem *et al., PhD thesis work (2024), in preparation*

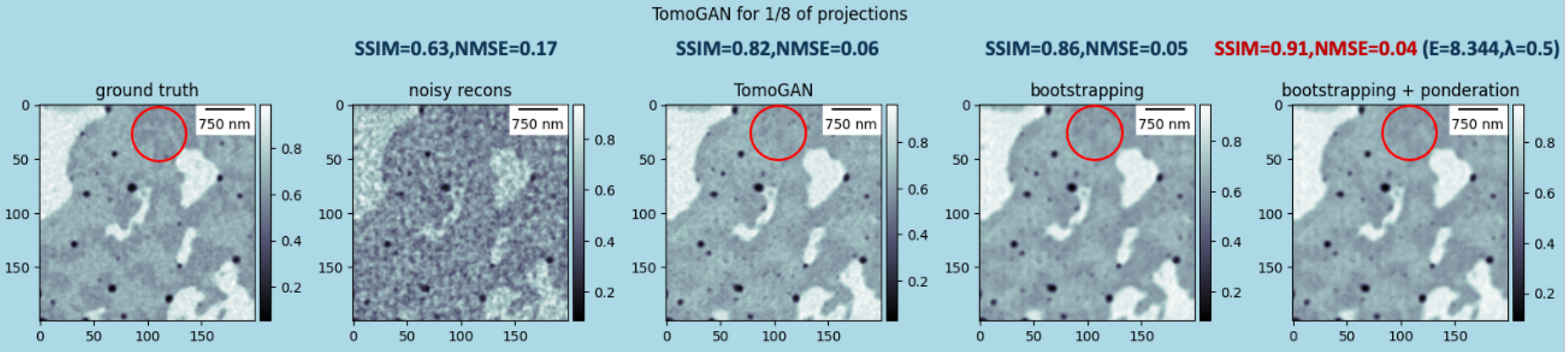
Application of TomoGAN

SSIM: Structural SIMilarity
NMSE: Normalized mean square error

TomoGAN for 1/8 of projections



TomoGAN for 1/8 of projections



M. Redhouane Boudjehem *et al., PhD thesis work (2024), in preparation*

Thank you for your attention

Contact info:

julio-cesar.da-silva@neel.cnrs.fr

Website:

sites.google.com/view/jcesardasilva

Data analysis tools:

https://github.com/jcesardasilva

Flash me for more info ▼

QR code