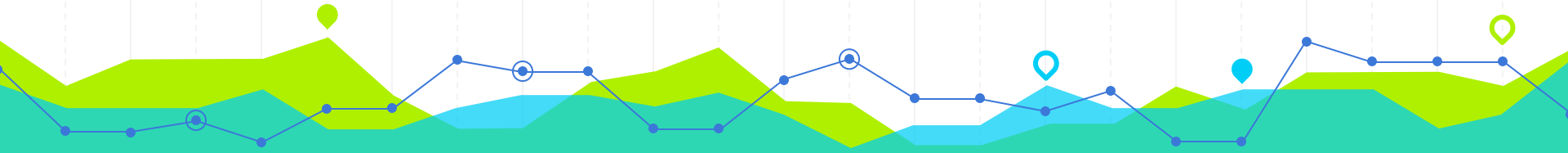
TensorFlow

Keras

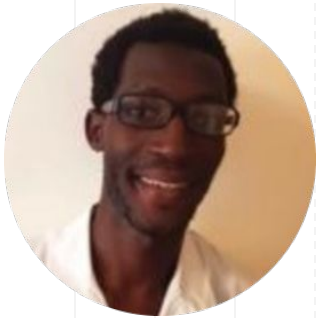


DEEP LEARNING

sample

SAMPLE

ABOUT THE SPEAKER



aws certified Solutions Architect Associate

aws certified Cloud Practitioner

aws certified SysOps Administrator Associate

Dr Assan Sanogo

Data Scientist x Pharmacist

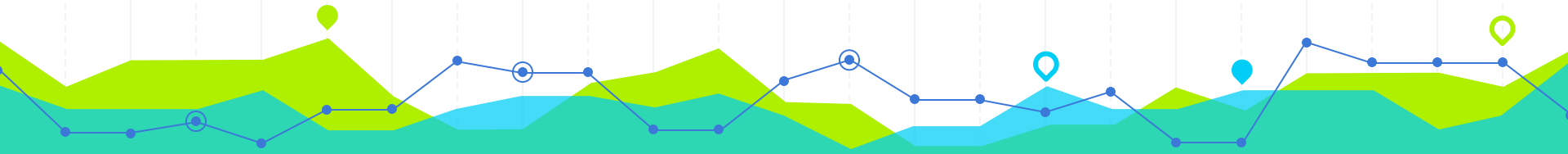
Independent Data scientist

Develops data sciences program

Coaches >100 junior data scientists

COURSE STRUCTURE

1. What is Deep Learning?
2. What is Tensorflow & Keras
3. Image Data
4. The structure of a Neural network
5. Convolutional layers
6. Pooling Layers
7. Activation
8. Loss function
9. Optimizer
10. Callbacks
11. Training
12. Evaluation



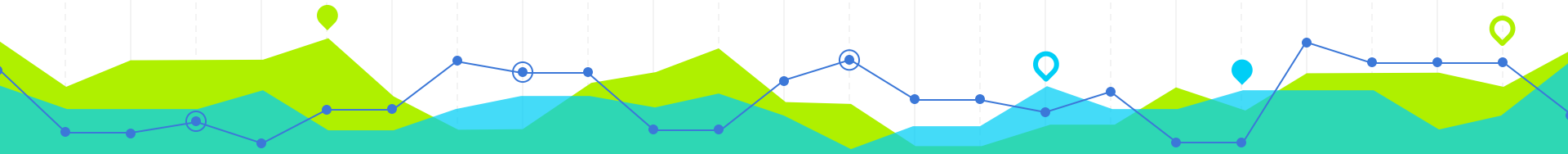
1 What is Deep Learning?

Let's start with the basics

1. WHAT IS DEEP LEARNING?

- **Deep Learning** is a subfield of machine learning concerned with algorithms inspired by the structure and function of the brain called **artificial neural networks**.
  - **Deep Learning** is the key technology behind **autonomous cars**, enabling them to recognize a cat or dog, or to distinguish digits
  - **Deep learning** requires **large amounts** of labeled data. For example, autonomous car development requires **millions of images** and thousands of hours of recording.
  - **Deep learning** requires a significant **computing power**. High-performance GPUs offer a parallel architecture that is efficient for this type of algorithm.

Deep Learning has set the standard in recognition and detection accuracy



What is Keras & Tensorflow?

Simplified modular deep learning

2

2. WHAT IS TENSORFLOW.KERAS?

- Keras is the **high-level API of TensorFlow**
- **highly-productive interface** for solving deep learning problems.
- provides **essential abstractions** and building blocks
- allows to develop deep learning solutions **very fast**

keras and tf.keras are two separate projects. (although they share many commonalities)

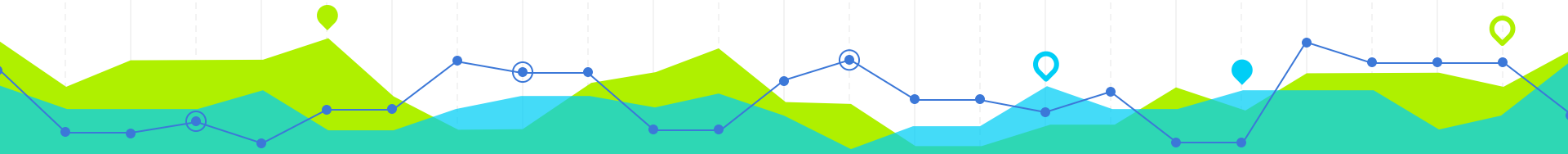


Image data

Images are tensors

3

SAMPLE

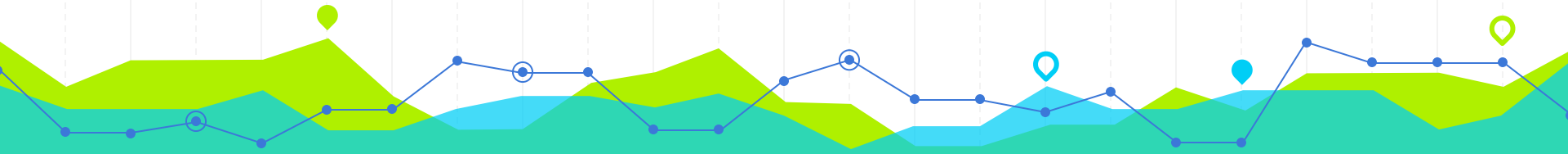
TensorFlow

Keras

3. THE DATA

- Dataset : Pictures of Dogs and Cats
- 25 000 Images (.jpg)
- Various Image size

The images vary in quality in simplicity (occlusion, multiple animals, human presence etc.)

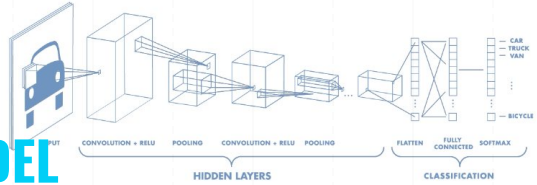


The Structure of a DL model

A matter of building blocks

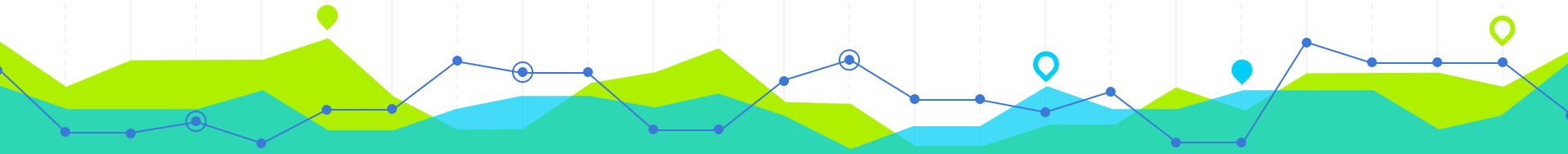
4

SAMPLE



A. STRUCTURE OF A CONVNET MODEL

- The main purpose of the **convolutional layers** is to extract meaningful features from the input images.
- To identify these features, CNNs analyze images thanks to “groups of pixels” called **filters.**
- Filters act as **feature detectors** from the original image.
- During the training process, the CNN slides the filter **over all locations** of the image and calculates the dot product for each feature at a time.
- **The results** of these calculations are stored in a so-called feature map (sometimes also named **activation map**).
- A feature map displays **where a certain feature** was identified in the original image.
- Then, the values from the **feature map** are transformed with an **activation function**
- These feature maps are used as **input** of the next layer



5 Layers

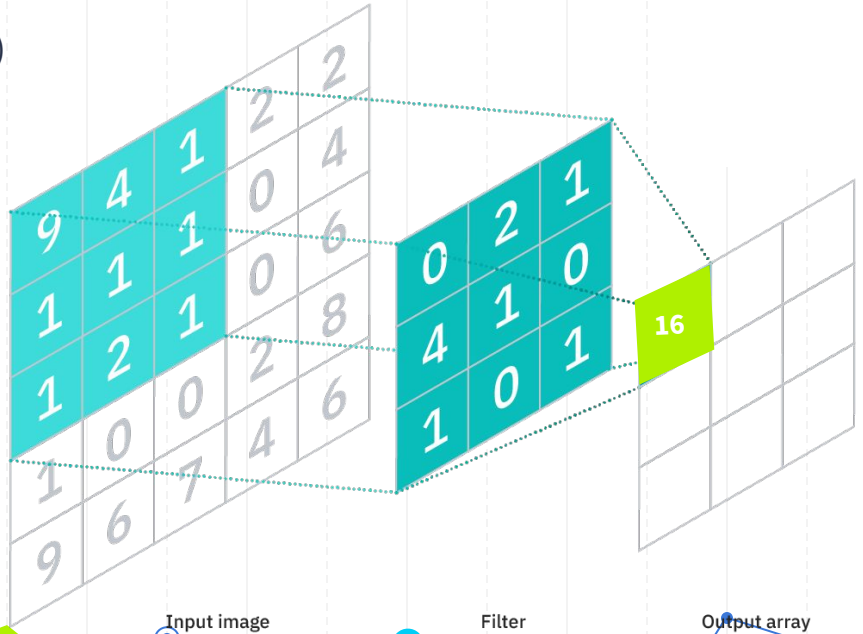
Let’s dig in the building blocks

TensorFlow

Keras

A. WHAT IS A CONVOLUTION ?

- The **learned** filter (dark turquoise)
- The image (light turquoise)
- Output (green)
- Image size reduction
- Learnable filters



Output = 9*0 + 4*2 + 1*1 +
1*4 + 1*1 + 1*0 +
1*1 + 2*0 + 1*1 = 16

TensorFlow Keras

B. WHAT IS A MAX POOLING LAYER?

- The image (light turquoise)
- Image size **reduction** *(padding)*
- No Learnable filters

1	3	5	1
76	32	12	6
26	07	8	26
16	17	68	32

2x2 Pooling

76	12
26	68

TensorFlow Keras

C. WHAT IS AN AVERAGE POOLING LAYER?

- The image (light turquoise)
- Image size r**eduction** *(padding)*
- **No Learnable filters**

1	3	5	1
76	32	12	6
26	37	68	26
16	37	8	32

28	6
29	29

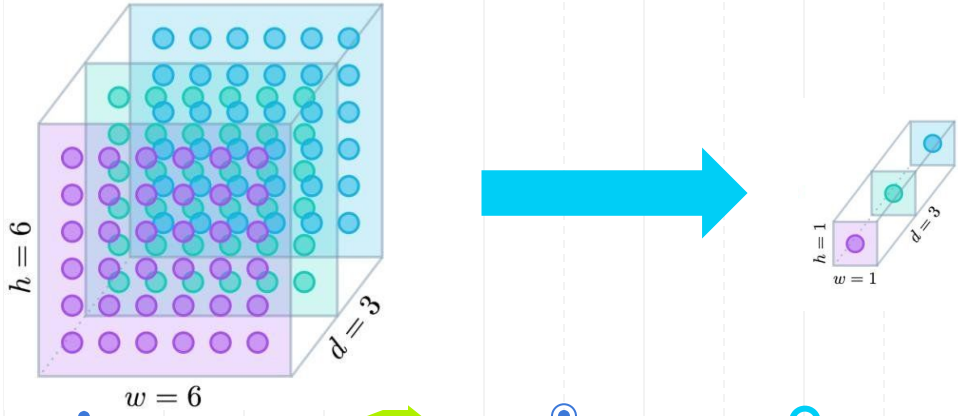
SAMPLE

TensorFlow

Keras

D. WHAT IS A GLOBAL AVERAGE POOLING LAYER?

- Transfert learning
- **Mapping of model feature maps to a vector**

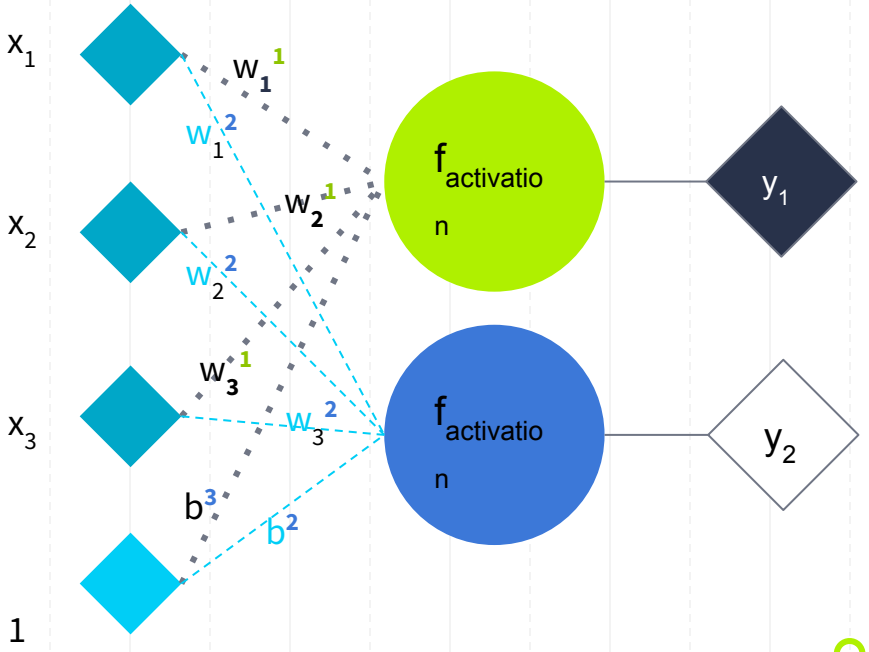


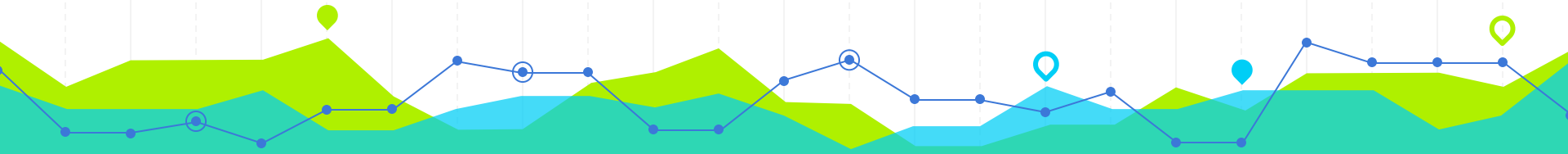
TensorFlow

Keras

D. WHAT IS A DENSE LAYER ?

- Dense layers are fully connected to previous layers
- **Concentrates the number of learnable parameters**
- Output after **Activation**
- **Numerous** notations





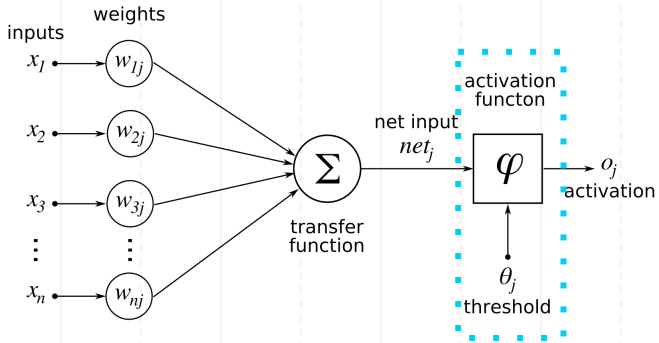
Activation functions

Set activate and Fire!

6

WHAT IS AN ACTIVATION FUNCTION ?

- Function added into an artificial neural network
  in order to make the network **learn complex patterns in the data**.
- Activation function are either **linear** or **nonlinear** (sigmoid, tanH, reLu)
- In analogy to **biological neurons**, the activation function
  represents the final step when “activated” neurons fire.



TensorFlow

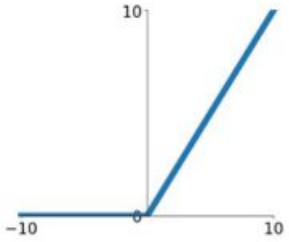
Keras

WHAT IS AN ACTIVATION FUNCTION ?

- reLU
- Sigmoid
- TanH

ReLU

$$\max(0, x)$$



ReLU, Sigmoid & tanH are NON LINEAR activation functions

TensorFlow

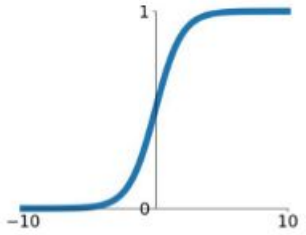
Keras

WHAT IS AN ACTIVATION FUNCTION ?

- reLU
- Sigmoid
- TanH

Sigmoid

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



ReLU, Sigmoid & tanH are NON LINEAR activation functions

TensorFlow

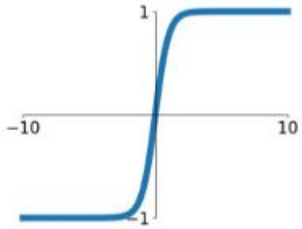
Keras

WHAT IS AN ACTIVATION FUNCTION ?

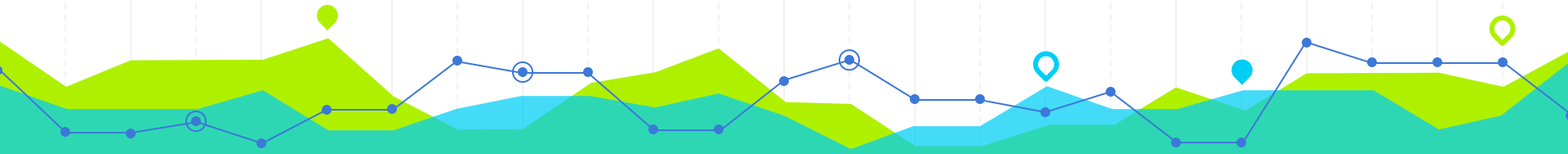
- reLU
- Sigmoid
- TanH

tanh

$$\tanh(x)$$



ReLU, Sigmoid & tanH are NON LINEAR activation functions



7 Loss function

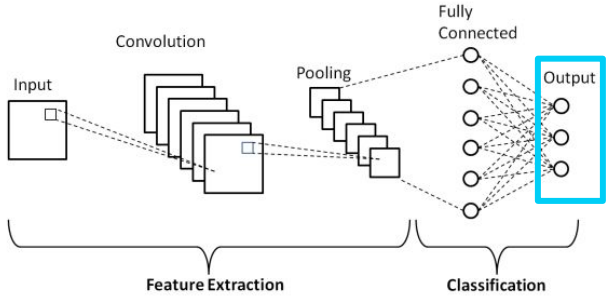
The objective function has to be minimized

TensorFlow Keras

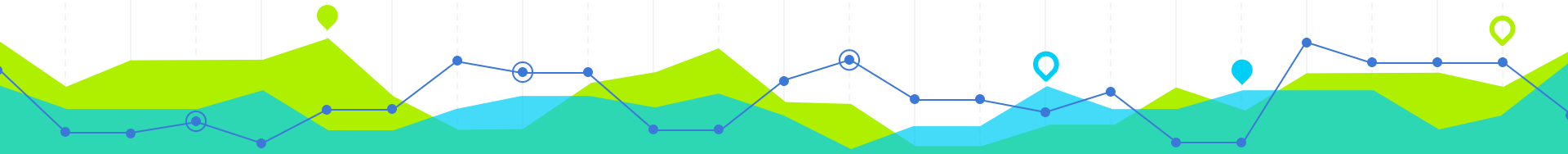
WHICH LOSS FUNCTION ?

- **Binary classification (n=2 classes)** → **Last activation** function : **Sigmoid**
  - Email spam detection (spam or not)
  - Churn prediction (churn or not)
- **Categorical classification (n>2 classes)** → Last activation function : **Softmax**
  - Digits classification (1 to 10)
  - Vehicles classification (cars, motorcycles, trucks)

WHICH LOSS FUNCTION ?



- **Binary classification (n=2 classes)** → **Last activation** function : **Sigmoid**
  - Email spam detection (spam or not)
  - Churn prediction (churn or not)
- **Categorical classification (n>2 classes)** → Last activation function : **Softmax**
  - Digits classification (1 to 10)
  - Vehicles classification (cars, motorcycles, trucks)



Optimizers

minimize the objective function

8

TensorFlow

Keras

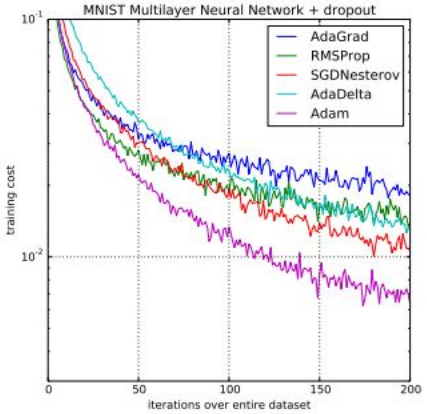
WHICH OPTIMIZER ?

- **SGD: stochastic gradient descent**

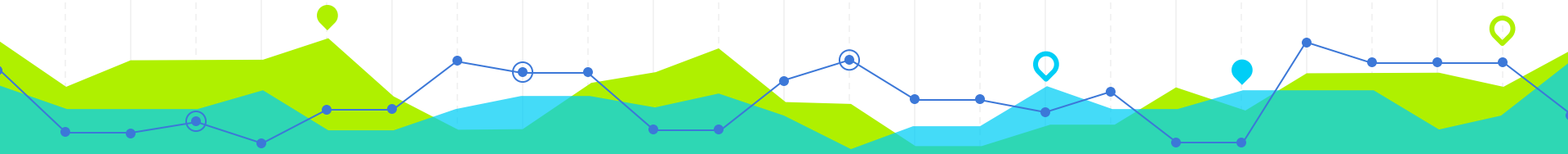
  Efficient way to fit linear classifiers and regressors under convex loss functions

- **Adam : adaptive moment estimation**

  The Adam optimization algorithm is an **extension** to **stochastic gradient descent**



Comparison of Adam to Other Optimization Algorithms Training a Multilayer Perceptron
Taken from Adam: A Method for Stochastic Optimization, 2015.



Callbacks

Save the model progress

9

TensorFlow Keras

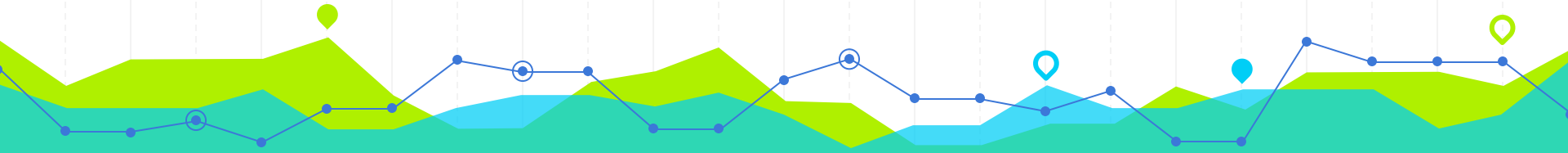
WHAT IS A CALLBACK ?

- A callback is **an object that can perform actions at various stages of training**
  - @ the start or end of an epoch
  - before or after a single batch.
- Callbacks for **monitoring metrics** : **tensorboard**
- Callbacks for **saving model** : **modelcheckpoint**
- Callbacks to **avoid overfitting**: **earlystopping.**

SAMPLE

TensorFlow

Keras



Training 10

Learning will happen step by step &
1 epoch at a time

TensorFlow Keras

A. TRAIN A NEURAL NETWORK - VOCAB

- **Epochs number: an hyperparameter** which describes **the number of times** the neural network will process **the whole dataset.**

- **Steps:** an hyperparameter which is calculated by using **the total number of observations and the batch size**

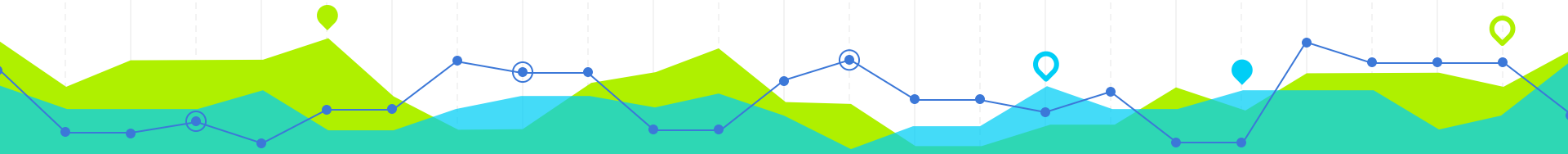
- **Batch size :** an hyperparameter that sets the **number of samples** to feed the neural network **before updating** the **learnable model parameters.**

- **Fine-tuning :** Fine-tuning is **one approach to transfer learning where you change the model output to fit the new task and train only the final layers of the model**.

TensorFlow Keras

B. TRAIN A NEURAL NETWORK STEP BY STEP?

- Prepare the data (preprocessing)
- Choose a pretrained model
- Define optimizers
- Choose the objective function
- Compile the model
- Define the metrics
- Launch the training
- Save the model



11 Evaluation

Time to check the model performance

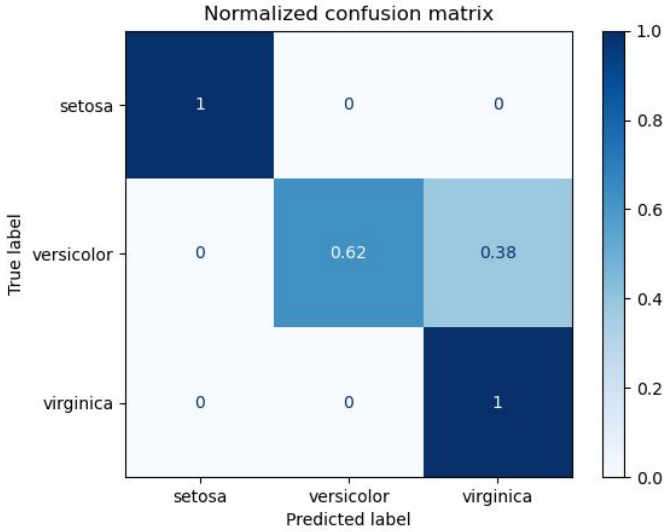
SAMPLE

TensorFlow

Keras

TRAINED MODEL : EVALUATION

- Confusion matrix
- Precision-recall

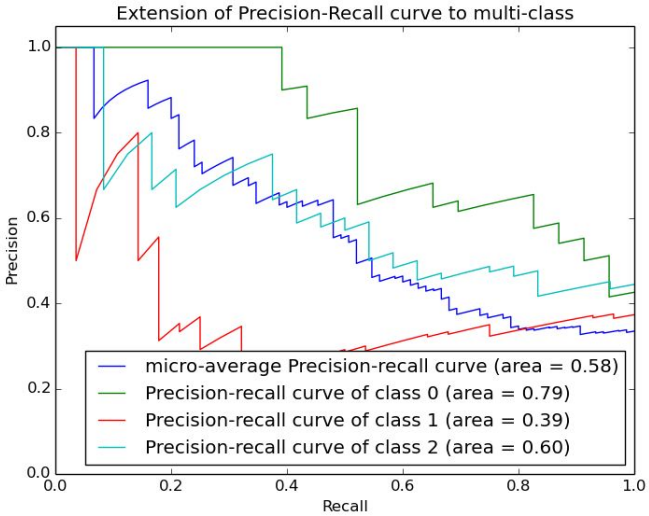


TensorFlow

Keras

TRAINED MODEL : EVALUATION

- **Confusion matrix**
- **Precision-recall**



THANKS!

Any questions?

You can ask for the **full** version

@ **predicteev.consulting@gmail.com**