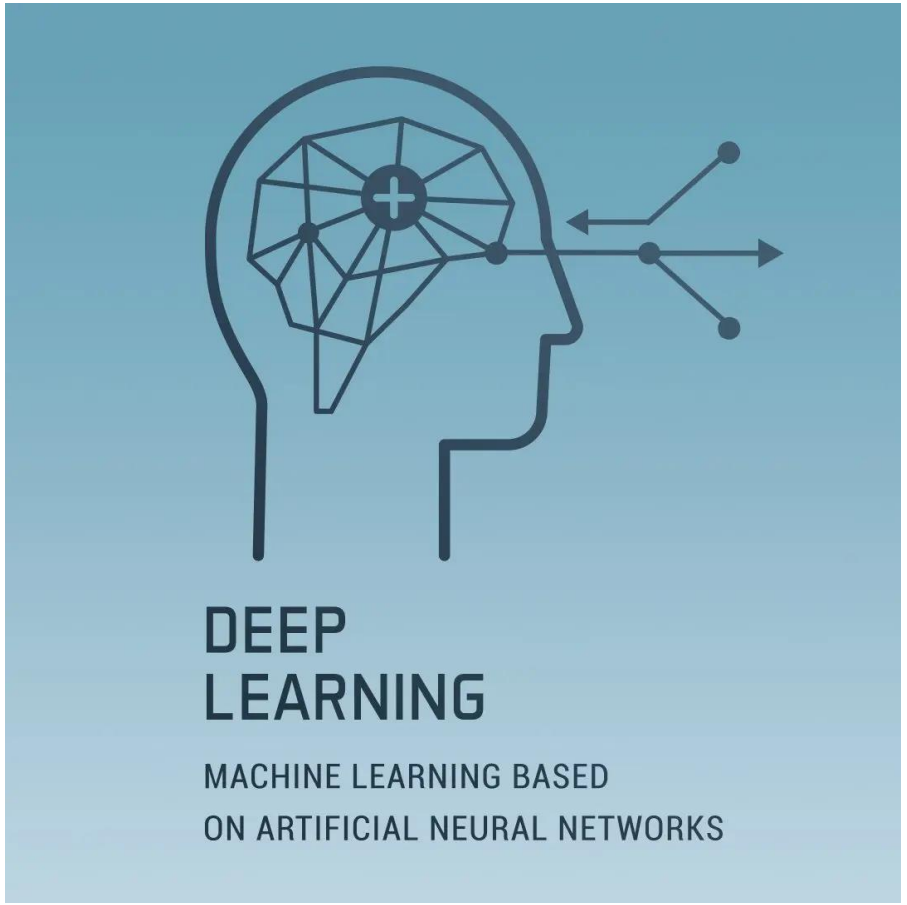
DEEP LEARNING

Content

- Deep Learning Basics
- Neural Network and Applications
- Components, Architecture, Activation Functions
- Working and Training of neural network
- Loss Functions and Gradient Descent
- Computer Vision
- Demo on OpenCV
- Demo on Building an NN from SK-learn
- Demo on Building an NN from Keras
- Case Study – Digit Recognizer using MNIST datasets

Deep Learning

- also known as **deep structured learning**
- part of a broader family of machine learning methods
- based on artificial neural networks
- learning can be supervised, semi-supervised or unsupervised.[1]



https://careers.edicomgroup.com/wp-content/uploads/2021/03/DeepLearning-2.jpg

What is a neural network?

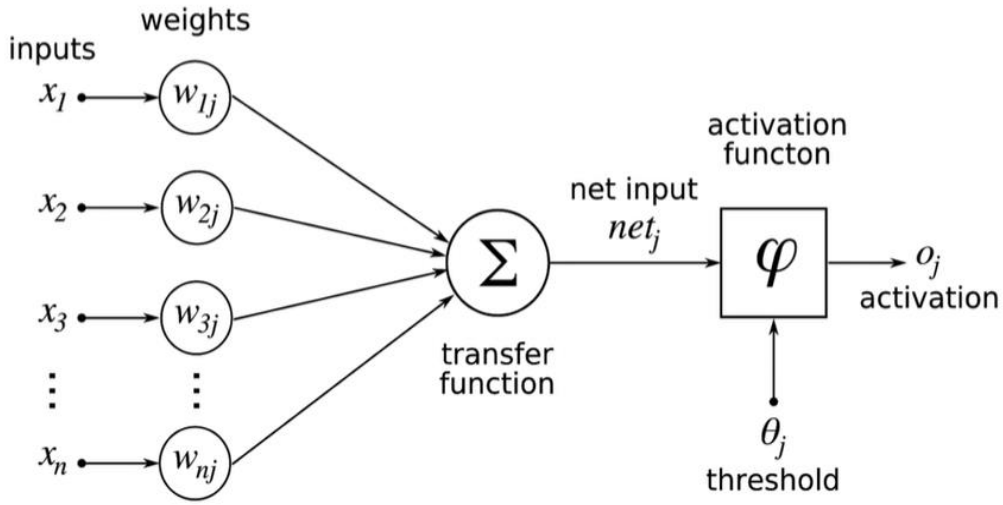
- Artificial replica of densely interconnected brain cells inside
- it can learn things, recognize patterns, and make decisions in a humanlike way.
- One don't have to program it to learn explicitly: ***it learns all by itself, just like a brain!***

Application of neural networks

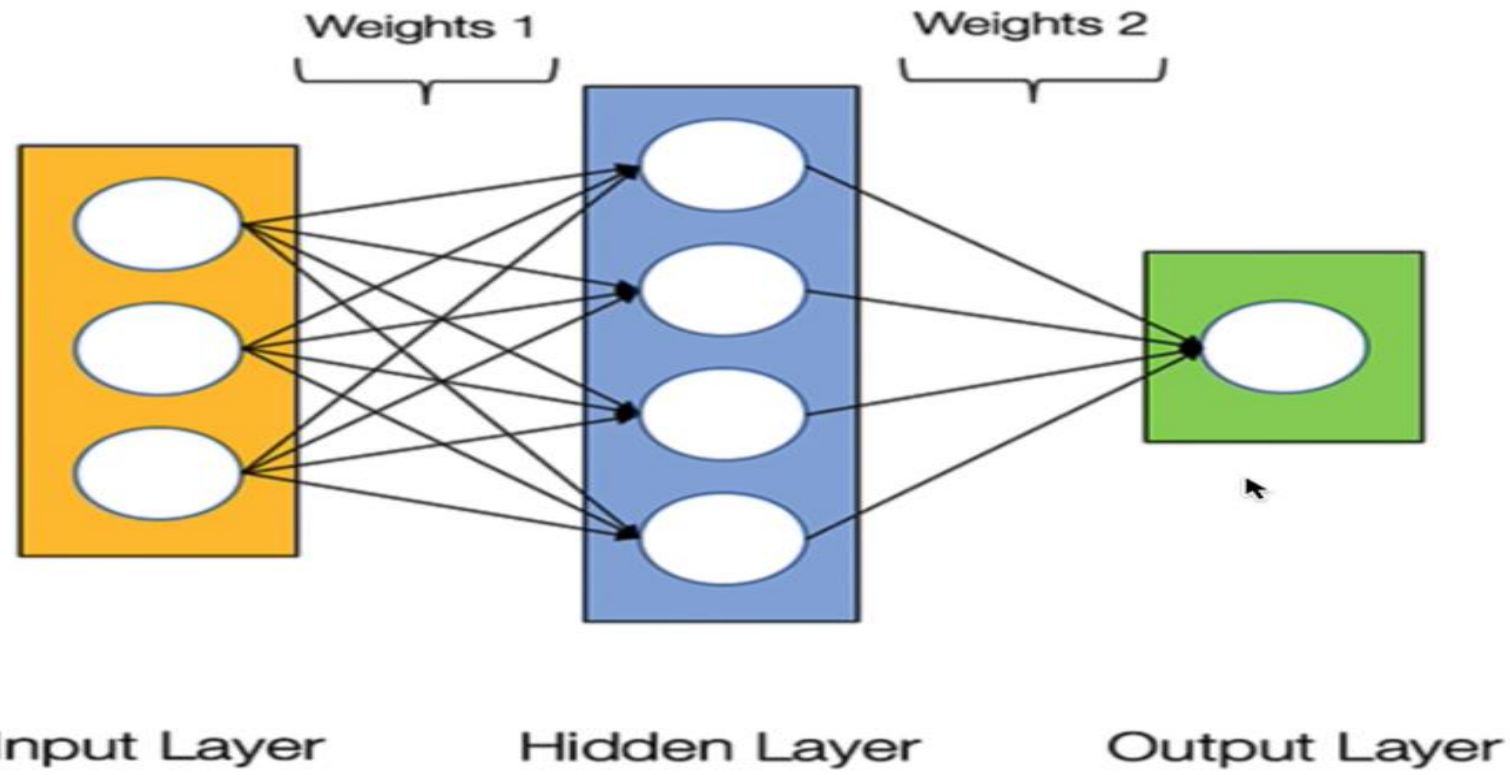
- Process modelling and control
- Machine Diagnostics
- Portfolio Management
- Target Recognition
- Medical Diagnosis
- Credit Rating
- Targeted Marketing
- Voice recognition
- Face recognition
- Financial Forecasting
- Fraud detection

Components

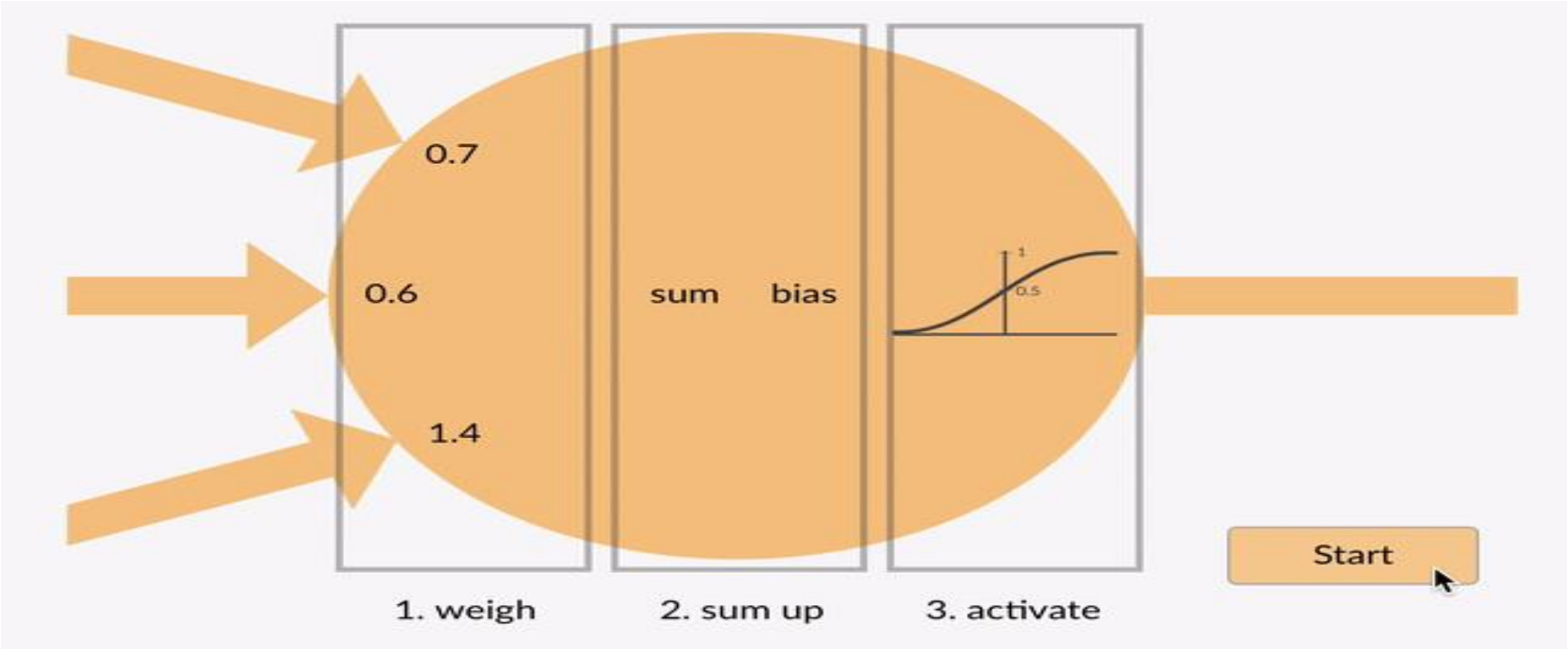
- Input Layer
- Hidden Layer
- Output Layer
- Weights and Biases between Layers
- Activation Function



2 Layer Architecture



Single Neuron



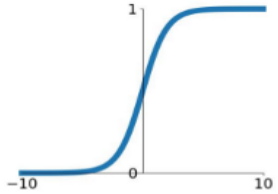
https://www.analyticsvidhya.com/blog/2020/02/cnn-vs-rnn-vs-mlp-analyzing-3-types-of-neural-networks-in-deep-learning/

Activation Function

Activation Functions

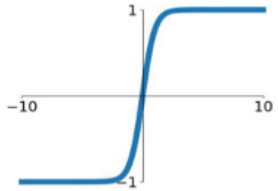
Sigmoid

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



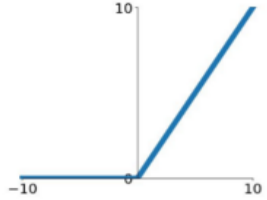
tanh

$$\tanh(x)$$



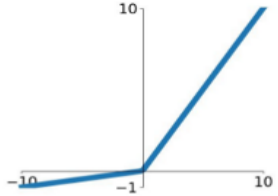
ReLU

$$\max(0, x)$$



Leaky ReLU

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

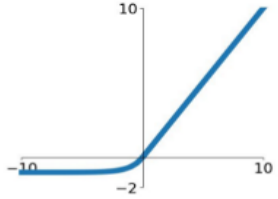


Maxout

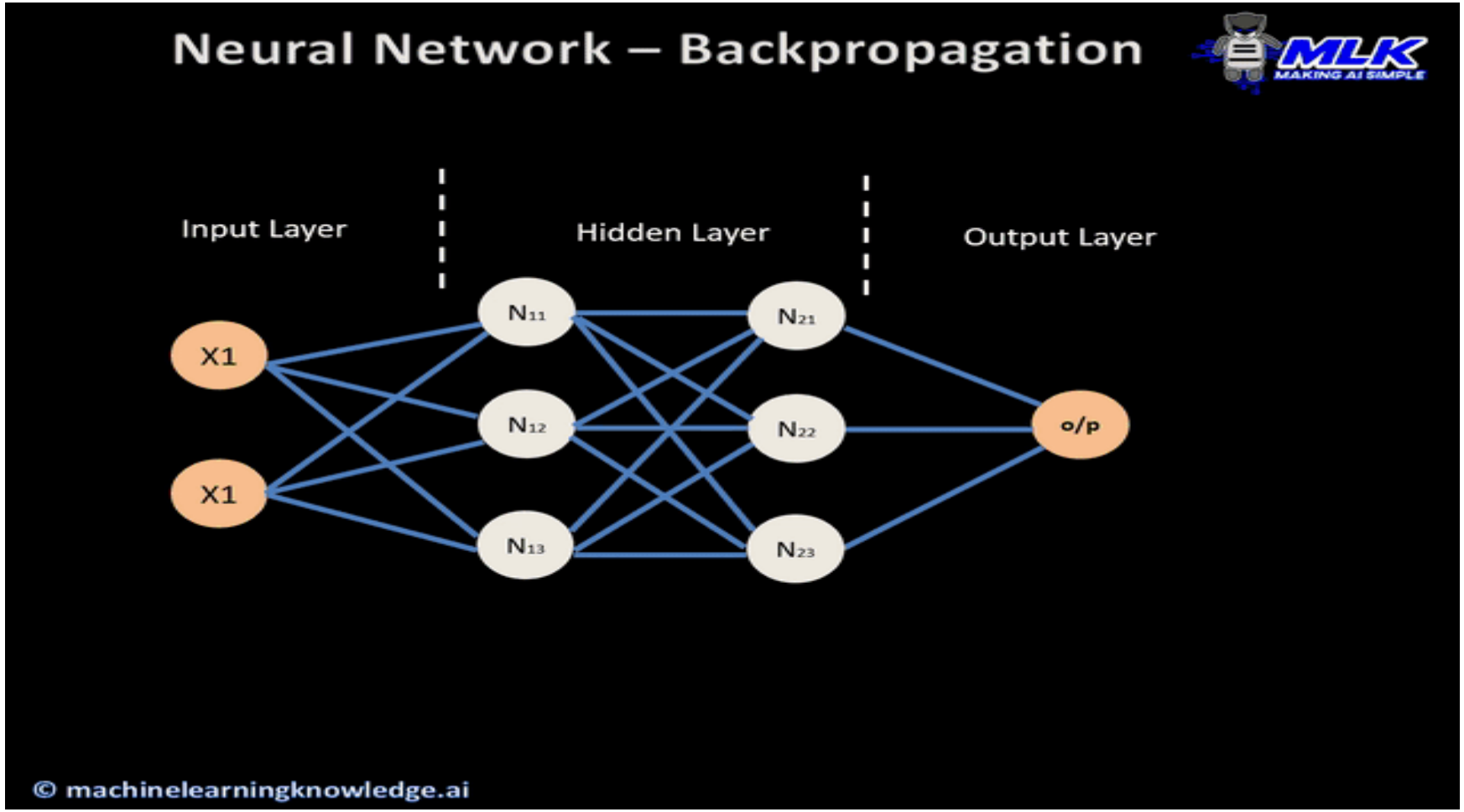
$$\max(w_1^T x + b_1, w_2^T x + b_2)$$

ELU

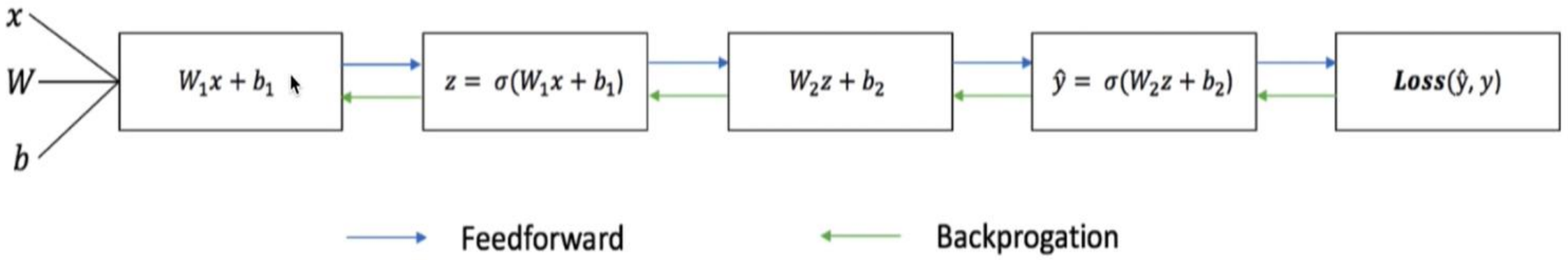
$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



Working of ANN

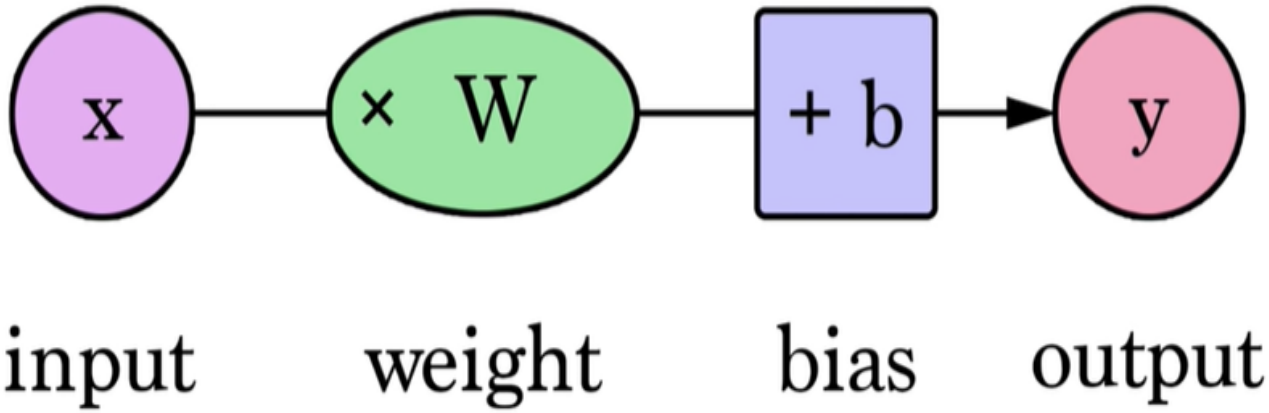


Sequential Graph



Training a Neural Network

- Output is designated by the following function:
  - $Y = \sigma(W_2\sigma(W_1x+b_1)+b_2)$
    - Weights are represented by W_1 and W_2
    - Biases represented by b_1 and b_2
- Two Steps in Training:
  - Feedforward
  - Backpropagation



Loss Functions

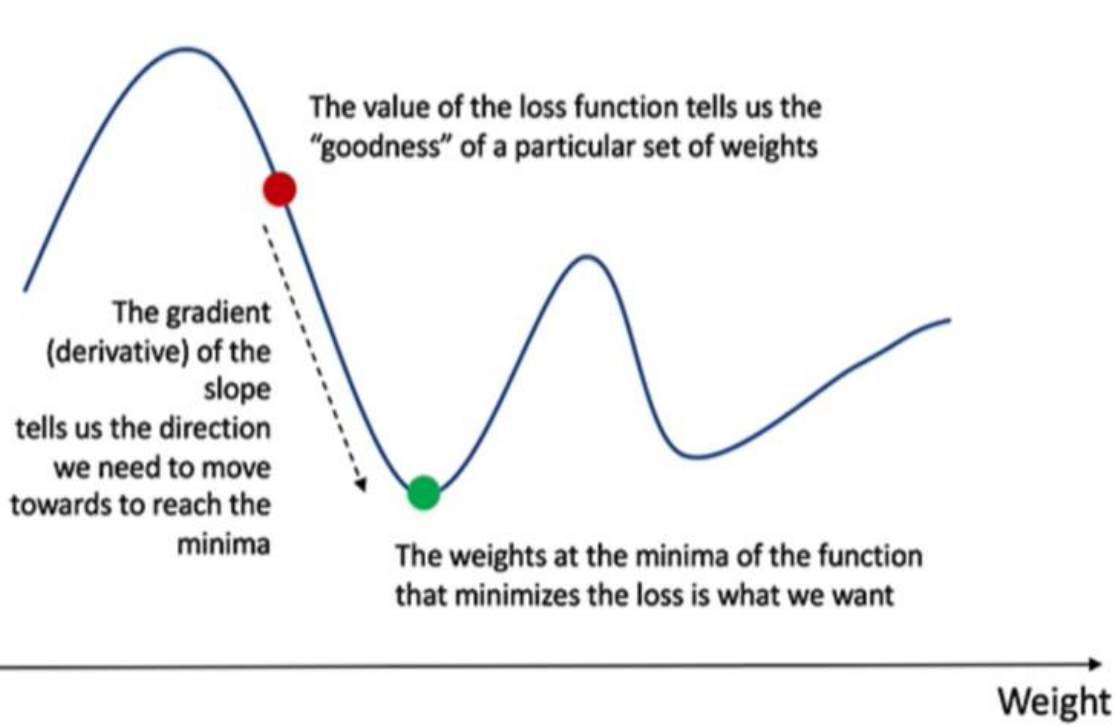
A loss function, that can be used to estimate the loss of the model so that the weights can be updated to reduce the loss on the next evaluation.

Gradient Descent

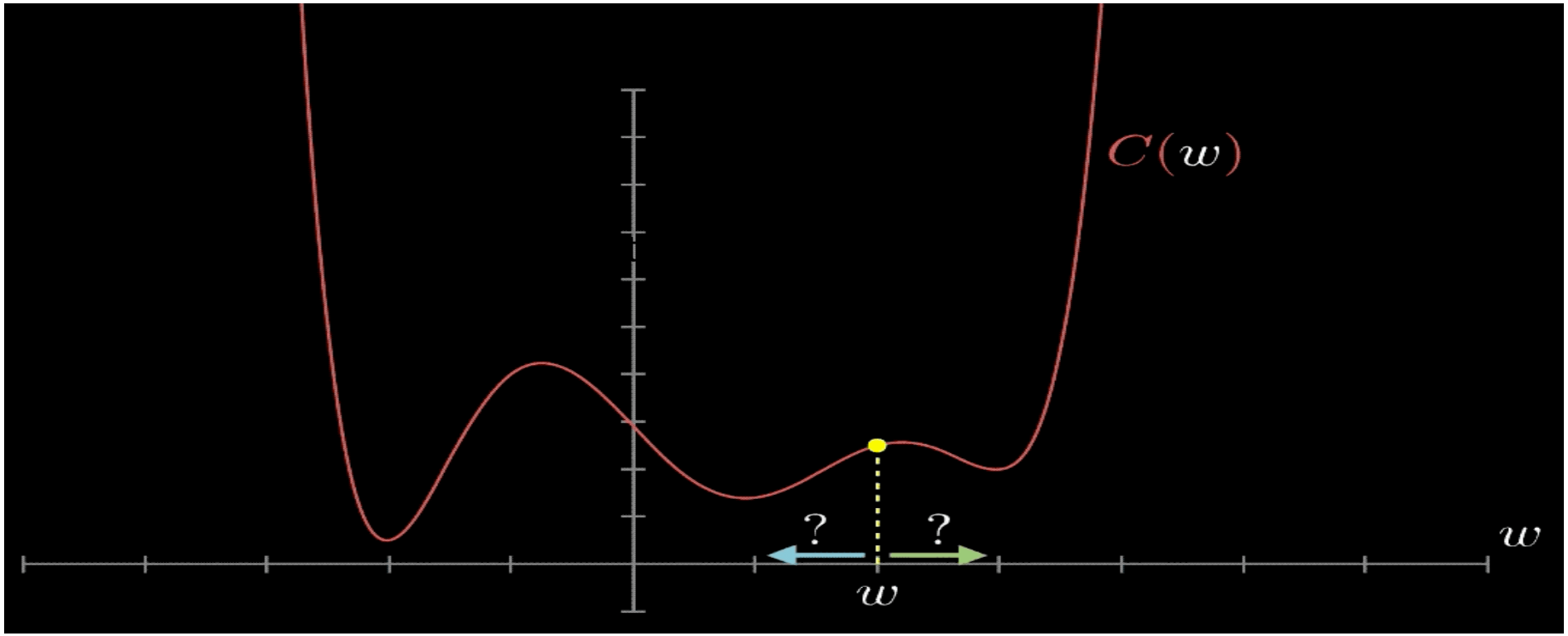
- Derivative of the Loss Function with respect to weights and biases

$$W_1 = W_1 - \alpha \frac{dL}{dW_1}$$

$$W_2 = W_2 - \alpha \frac{dL}{dW_2}$$



Gradient Descent



Ref: mlfromscratch.com

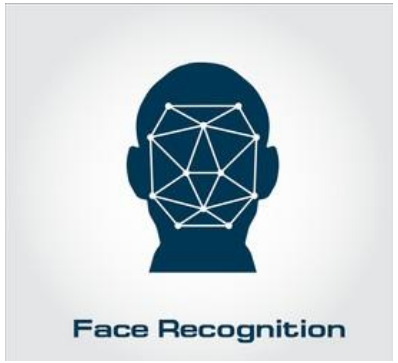
COMPUTER VISION

Gaining High-level understanding from digital images or videos

Microsoft SAP edunet foundation

Popular Python Libraries

SimpleCV

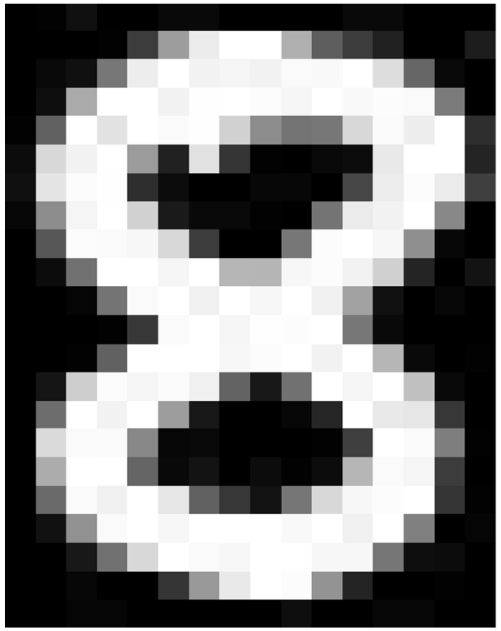


python

PYTESSERACT

What is an Image?

Data in the form of matrix(Rows and Columns) consisting of Pixels



0	2	15	0	0	11	10	0	0	0	0	9	9	0	0	0
0	0	0	4	60	157	236	255	255	177	95	61	32	0	0	29
0	10	16	119	238	255	244	245	243	250	249	255	222	103	10	0
0	14	170	255	255	244	254	255	253	245	255	249	253	251	124	1
2	98	255	228	255	251	254	211	141	116	122	215	251	238	255	49
13	217	243	255	155	33	226	52	2	0	10	13	232	255	255	36
16	229	252	254	49	12	0	0	7	7	0	70	237	252	235	62
6	141	245	255	212	25	11	9	3	0	115	236	243	255	137	0
0	87	252	250	248	215	60	0	1	121	252	255	248	144	6	0
0	13	113	255	255	245	255	182	181	248	252	242	208	36	0	19
1	0	5	117	251	255	241	255	247	255	241	162	17	0	7	0
0	0	0	4	58	251	255	246	254	253	255	120	11	0	1	0
0	0	4	97	255	255	255	248	252	255	244	255	182	10	0	4
0	22	206	252	246	251	241	100	24	113	255	245	255	194	9	0
0	111	255	242	255	158	24	0	0	6	39	255	232	230	56	0
0	218	251	250	137	7	11	0	0	0	2	62	255	250	125	3
0	173	255	255	101	9	20	0	13	3	13	182	251	245	61	0
0	107	251	241	255	230	98	55	19	118	217	248	253	255	52	4
0	18	146	250	255	247	255	255	255	249	255	240	255	129	0	5
0	0	23	113	215	255	250	248	255	255	248	248	118	14	12	0
0	0	6	1	0	52	153	233	255	252	147	37	0	0	4	1
0	0	5	5	0	0	0	0	0	14	1	0	6	6	0	0

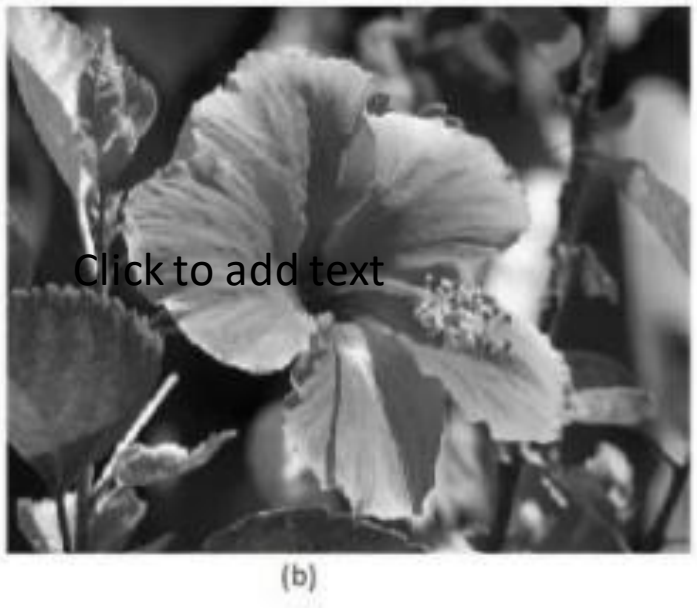
0	2	15	0	0	11	10	0	0	0	0	9	9	0	0	0
0	0	0	4	60	157	236	255	255	177	95	61	32	0	0	29
0	10	16	119	238	255	244	245	243	250	249	255	222	103	10	0
0	14	170	255	255	244	254	255	253	245	255	249	253	251	124	1
2	98	255	228	255	251	254	211	141	116	122	215	251	238	255	49
13	217	243	255	155	33	226	52	2	0	10	13	232	255	255	36
16	229	252	254	49	12	0	0	7	7	0	70	237	252	235	62
6	141	245	255	212	25	11	9	3	0	115	236	243	255	137	0
0	87	252	250	248	215	60	0	1	121	252	255	248	144	6	0
0	13	113	255	255	245	255	182	181	248	252	242	208	36	0	19
1	0	5	117	251	255	241	255	247	255	241	162	17	0	7	0
0	0	0	4	58	251	255	246	254	253	255	120	11	0	1	0
0	0	4	97	255	255	255	248	252	255	244	255	182	10	0	4
0	22	206	252	246	251	241	100	24	113	255	245	255	194	9	0
0	111	255	242	255	158	24	0	0	6	39	255	232	230	56	0
0	218	251	250	137	7	11	0	0	0	2	62	255	250	125	3
0	173	255	255	101	9	20	0	13	3	13	182	251	245	61	0
0	107	251	241	255	230	98	55	19	118	217	248	253	255	52	4
0	18	146	250	255	247	255	255	255	249	255	240	255	129	0	5
0	0	23	113	215	255	250	248	255	255	248	248	118	14	12	0
0	0	6	1	0	52	153	233	255	252	147	37	0	0	4	1
0	0	5	5	0	0	0	0	0	14	1	0	6	6	0	0

Image : Image with Pixels

https://mozanunal.com/images/pixel.png

Types of Images

Color Image



Grayscale Image

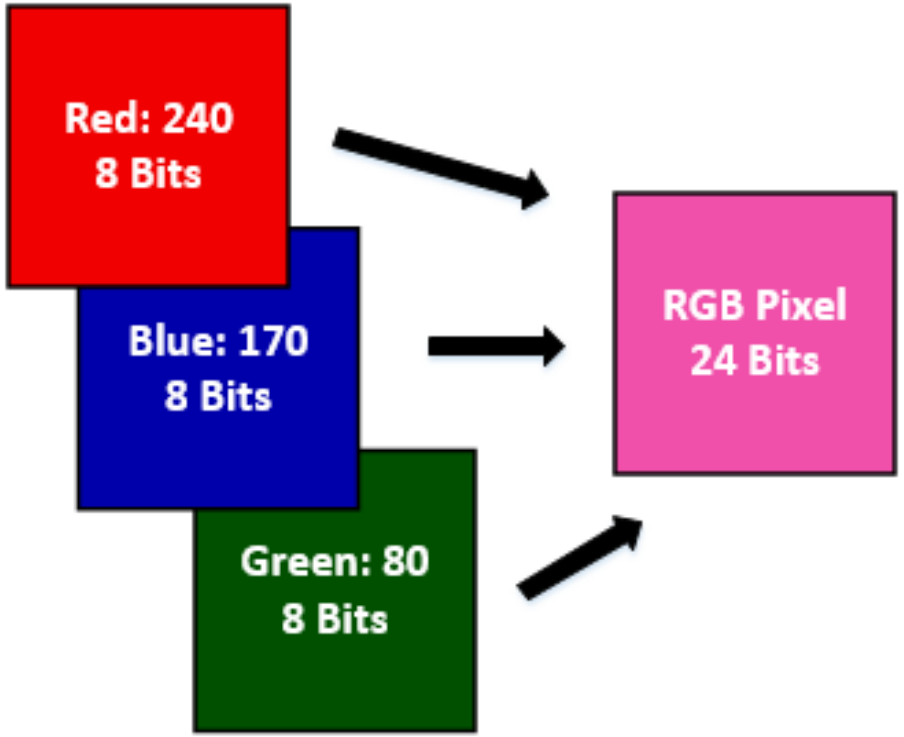
Binary Image

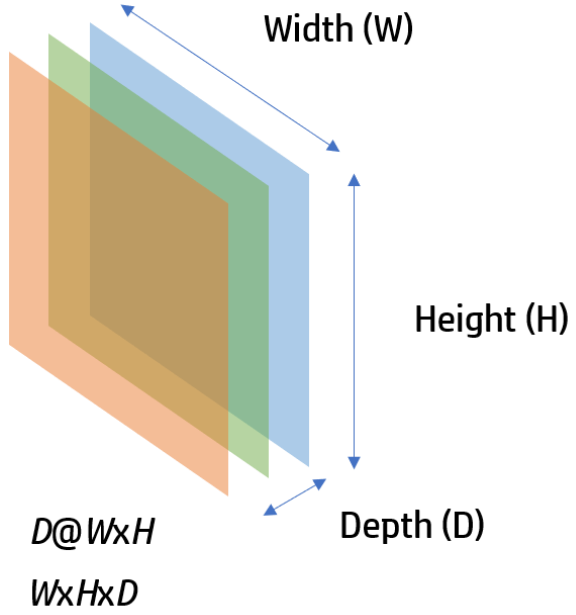
https://www.researchgate.net/profile/Sanskruti-Patel-2/publication/344249310/figure/fig2/AS:935972338425861@1600164603821/Figure2-a-RGB-image-b-Gray-Scale-image-c-Binary-image.jpg

Color Image

Width * Height * Depth

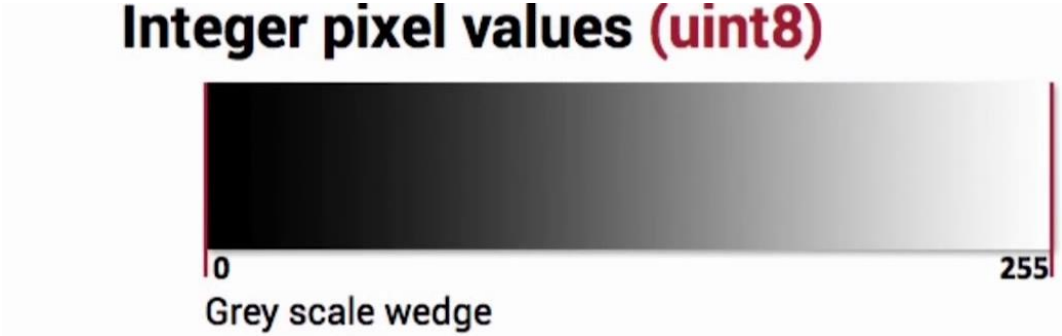
Depth : [Red, Green, Blue]





https://www.codeproject.com/KB/tips/1112774/rgbPixelEx.PNG

How to get Binary Image in Opencv?



https://i.ytimg.com/vi/tSBuwNpIUdI/maxresdefault.jpg

Main methods of OpenCV

1. cv2.imread() - method loads an image from the specified file
2. cv2.resize() – Resizing the pixels value of an image
3. cv2.cvtColor()- convert an image from one color space to another.
4. cv2.threshold() – Changing the pixels value with respect to a threshold

Threshold Function

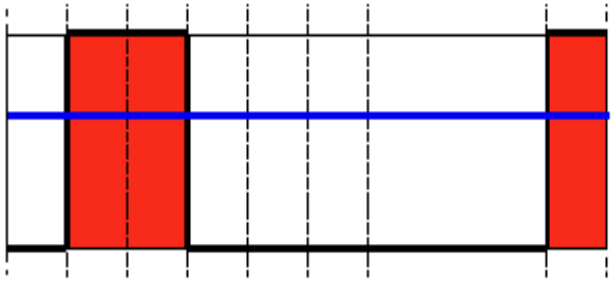
ret,binary = cv2.threshold(img,127,255,cv2.THRESH_BINARY)

Threshold Binary

- This thresholding operation can be expressed as:

$$\texttt{dst}(x,y) = \begin{cases} \texttt{maxVal} & \text{if } \texttt{src}(x,y) > \texttt{thresh} \\ 0 & \text{otherwise} \end{cases}$$

- So, if the intensity of the pixel $\texttt{src}(x,y)$ is higher than $\texttt{thresh}$, then the new pixel intensity is set to a $\texttt{MaxVal}$. Otherwise, the pixels are set to 0.



https://docs.opencv.org/2.4.13.7/doc/tutorials/imgproc/threshold/threshold.html#threshold-binary

Build ANN Using Scikit-learn

Microsoft

SAP

edunet foundation

Extensions to SciPy (Scientific Python) are called SciKits. SciKit-Learn provides machine learning algorithms.

- Algorithms for supervised & unsupervised learning
- Built on SciPy and Numpy
- Standard Python API interface
- Sits on top of c libraries, LAPACK, LibSVM, and Cython
- Open Source: BSD License (part of Linux)

Building Model using Scikit Package

Applying same dataset over MLPClassifier under Scikit Package

```
#With scikit learn - with multilayer perceptron classifier
from sklearn.neural_network import MLPClassifier
from sklearn.metrics import accuracy_score

sknet = MLPClassifier(hidden_layer_sizes=(8),learning_rate_init=0.001, max_iter=100)
```

```
#Fit the data to the classifier model
sknet.fit(Xtrain, ytrain)
preds_train = sknet.predict(Xtrain)
preds_test = sknet.predict(Xtest)

#Print the accuracy of the train and test datasets
print("Train accuracy of sklearn neural network: {}".format(round(accuracy_score(preds_train, ytrain),2)*100))
print("Test accuracy of sklearn neural network: {}".format(round(accuracy_score(preds_test, ytest),2)*100))
```

Microsoft SAP edunet foundation

Build ANN Using Keras

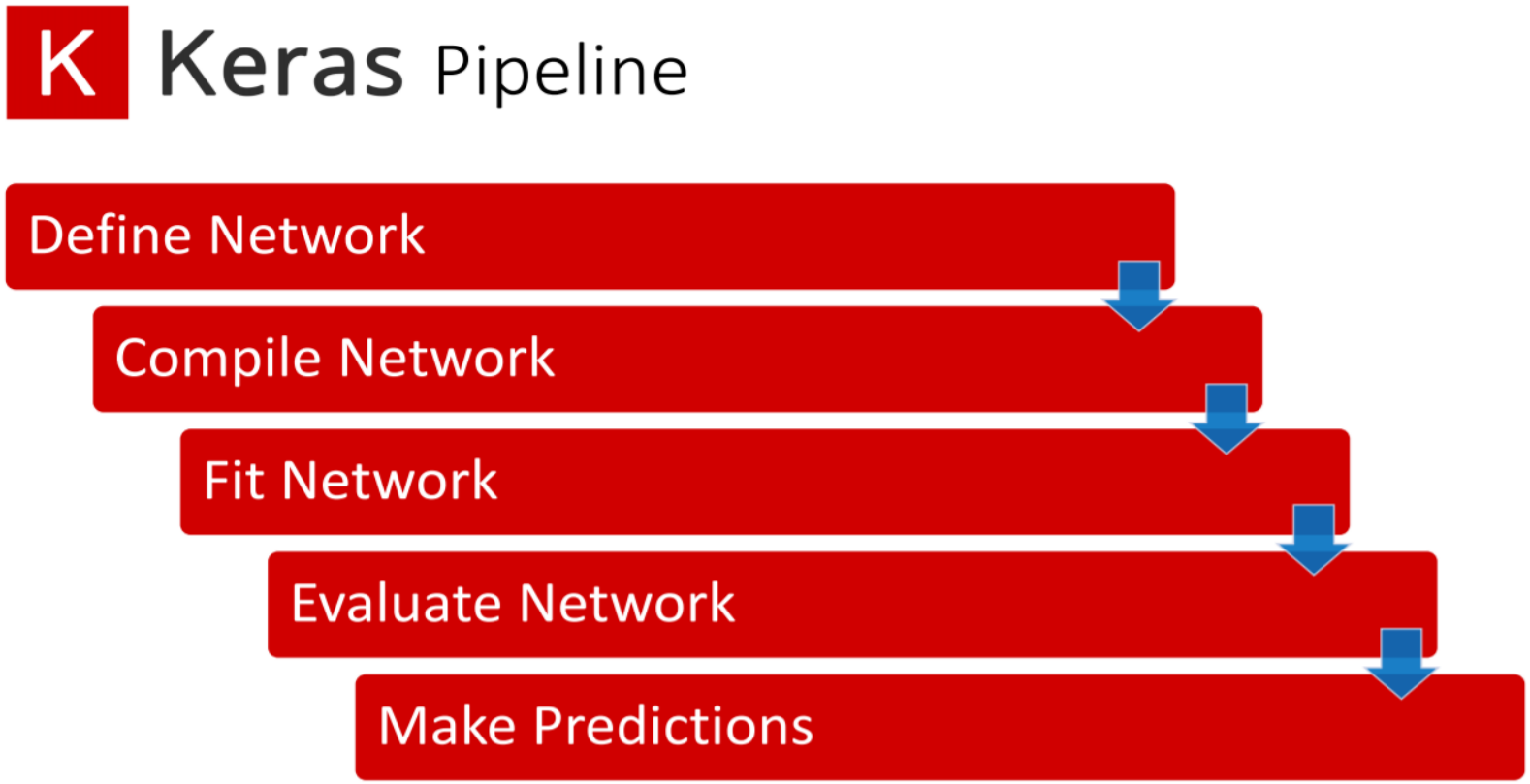
What is Keras

- High Level neural network API
- Written in Python
- Integration with TensorFlow, Theano & CNTK.
  - (MXNet backend for Keras on the way!)

Microsoft SAP edunet foundation

Why K Keras

- Fast prototyping
- Supports CNN, RNN & combination of both
- Modularity
- Easy extensibility
- Simple to get started, simple to keep going
- Deep enough to build serious models.
- Well-written document.
- Runs seamlessly on CPU and GPU.



Microsoft SAP edunet foundation

Building Model using K Keras

```
#With Keras

from tensorflow.keras import Sequential
from tensorflow.keras.layers import Dense
from tensorflow.keras.optimizers import Adam

#Define the model
model = Sequential()
model.add(Dense(8,input_shape=(13,)))
model.add(Dense(1, activation='sigmoid'))
#model.summary()
```

```
# compile the model
opt = Adam(learning_rate=0.001)
model.compile(optimizer=opt, loss='binary_crossentropy', metrics=['accuracy'])
```

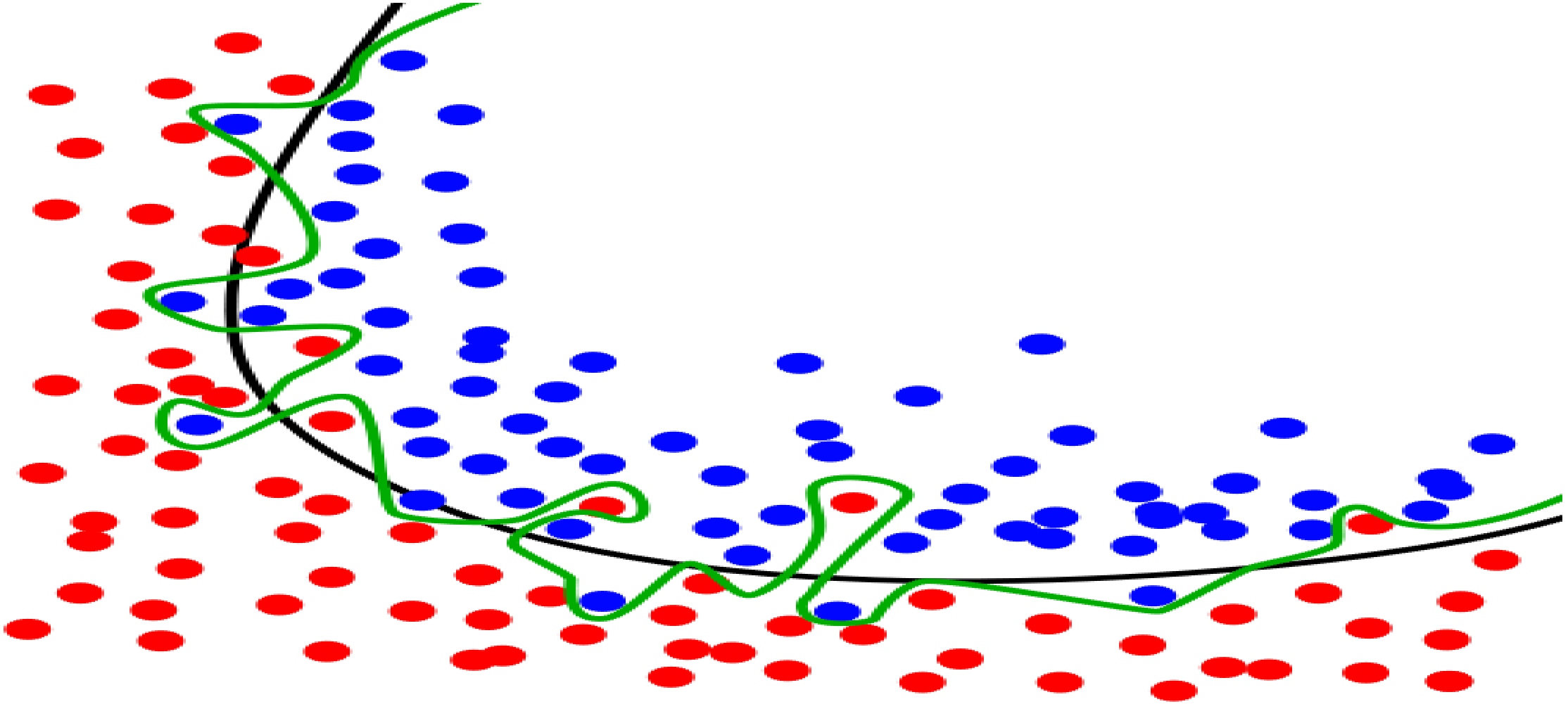
```
#Fitting the model to data - Training
model.fit(Xtrain, ytrain, epochs=10, verbose=1)
```

Overfitting

- In statistics, **overfitting** is "the production of an analysis that corresponds too closely or exactly to a particular set of data and may therefore fail to fit additional data or predict future observations reliably".
- An **overfitted model** is a statistical model that contains more parameters than can be justified by the data.[3]

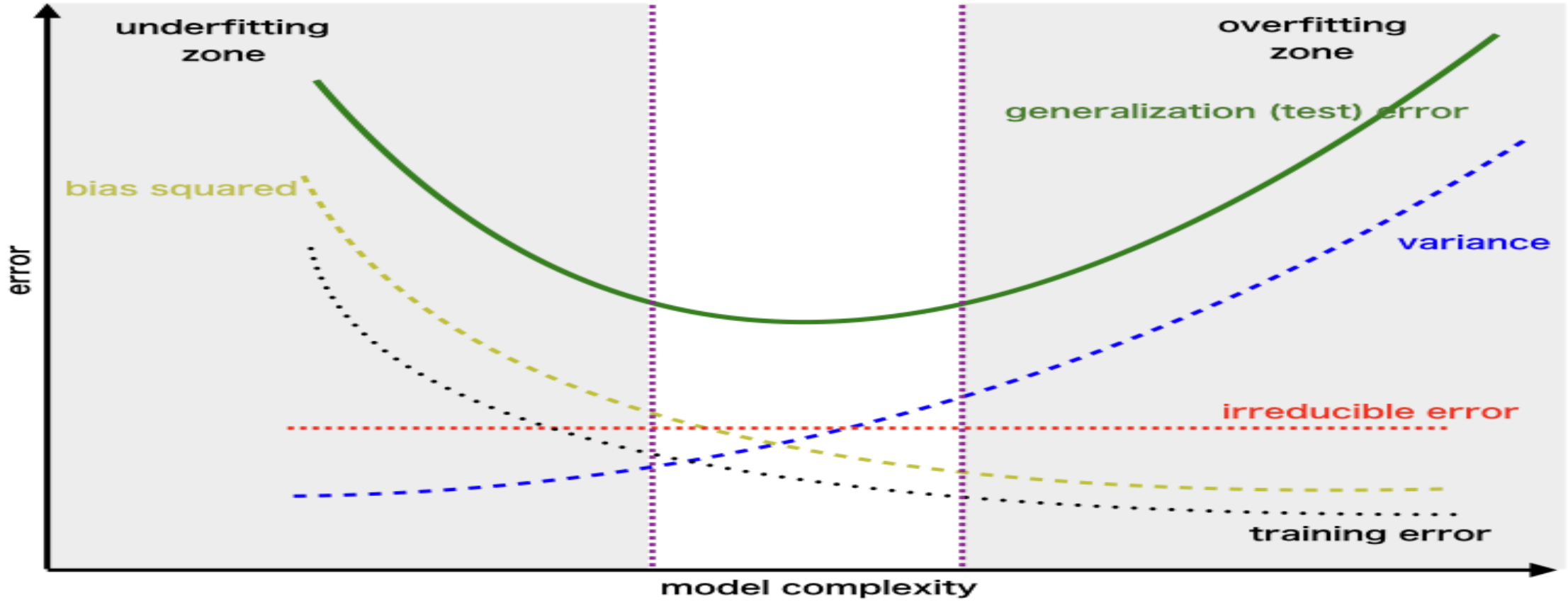
Underfitting

- Underfitting refers to a model that can neither model the training data nor generalize to new data.
- An underfit machine learning model is not a suitable model and will be obvious as it will have poor performance on the training data.[5]



Source - Chabacano, CC BY-SA 4.0 <https://creativecommons.org/licenses/by-sa/4.0>, via Wikimedia Commons [4]

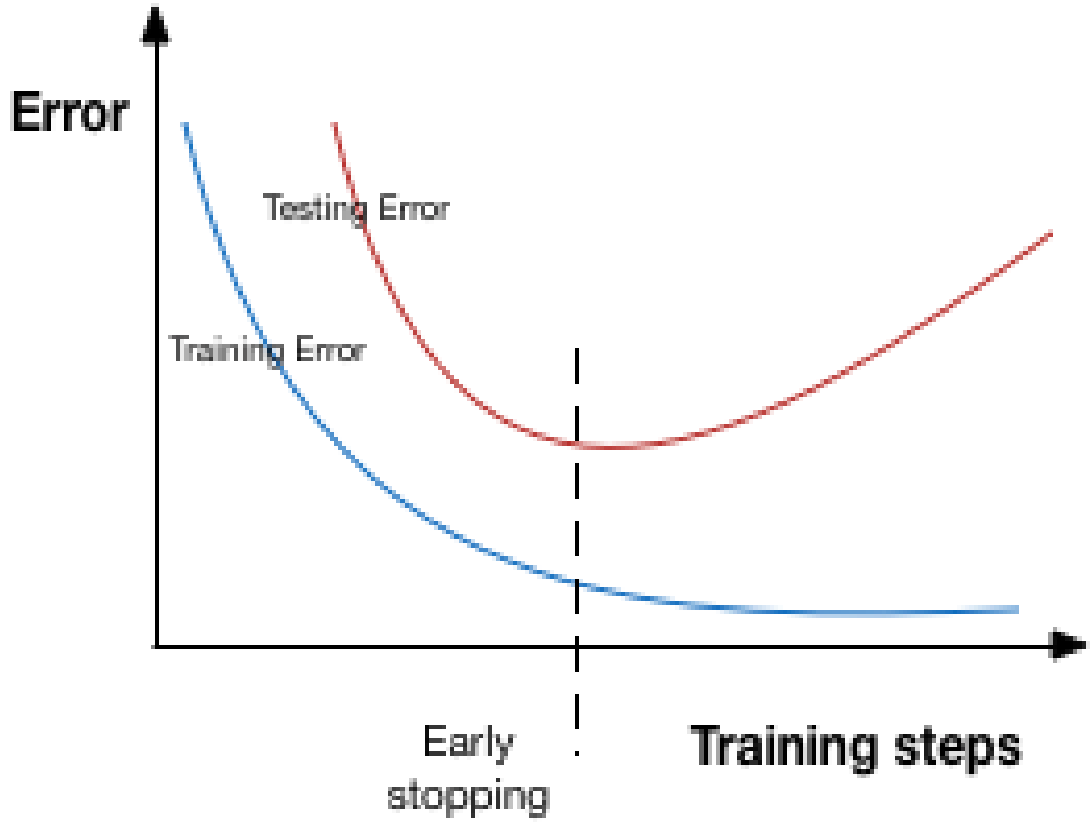
Bias - Variance Tradeoff



Source - https://www.geeksforgeeks.org/ml-bias-variance-trade-off/ [2]

Early Stopping

- One part of the training set as the validation set.
- When we see that the performance on the validation set is getting worse, we immediately stop the training on the model. This is known as early stopping.[7]



Source - https://www.analyticsvidhya.com/blog/2018/04/fundamentals-deep-learning-regularization-techniques/

Practical
Digit Recognizer using MNIST Dataset