Module
Artificial Intelligence Fundamental

Section
Deep Learning

Deep Learning

An introduction

Why Deep Learning?

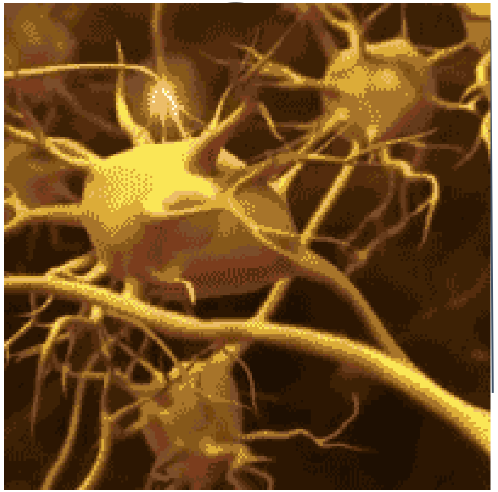
- Real applications and others
- But, why now?

Coba, renungkan bagaimana kesadaran itu? Dan bagaimana struktur saraf pada otak kita bekerja?

- NN dimulai dari kisah bahwa manusia berusaha untuk meniru cara neuron pada manusia bekerja
- Dengan membuat tiruan, yaitu ANN
- Dikemudian hari ANN menjadi cikal bakal DL

Coba, renungkan bagaimana keberadaan kita sebagai manusia dan apa saja yang telah kita perbuat?

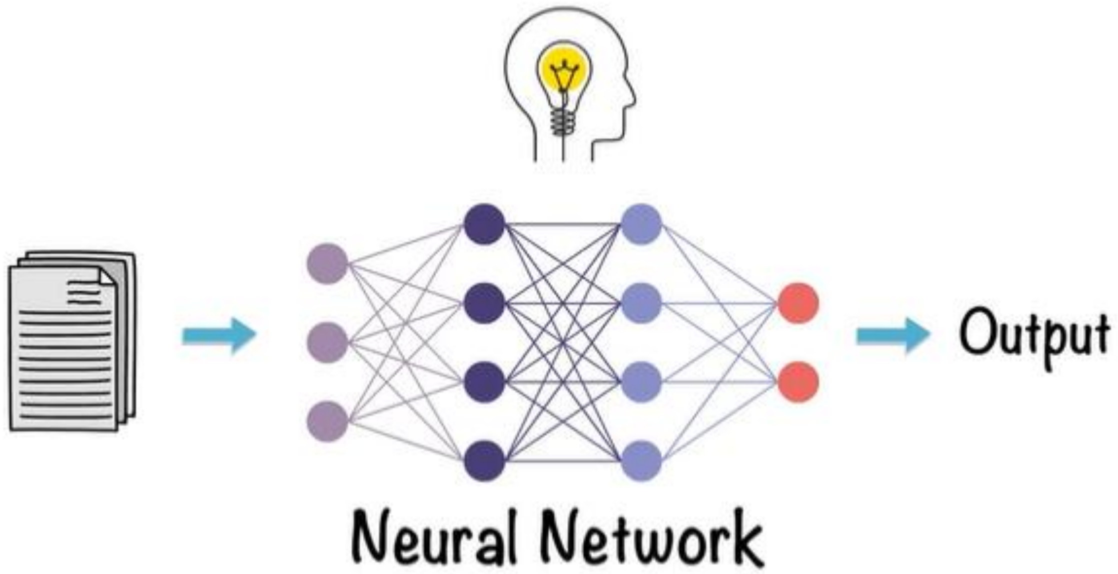
- 13.8 miliar tahun yang lalu alam semesta terbentuk
- 4.5 miliar tahun yang lalu bumi mulai tercipta
- 300.000 tahun yang lalu peradaban manusia modern
- Lalu bagaimana dengan *“a brief history of AI and DL”* ?



Session I

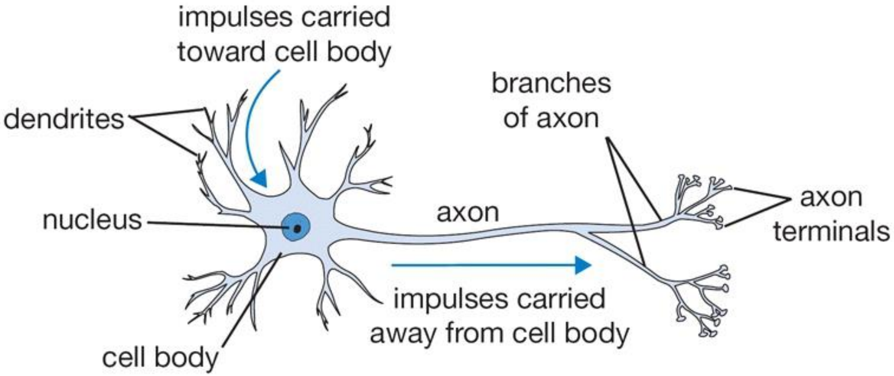
Pengenalan Deep Learning

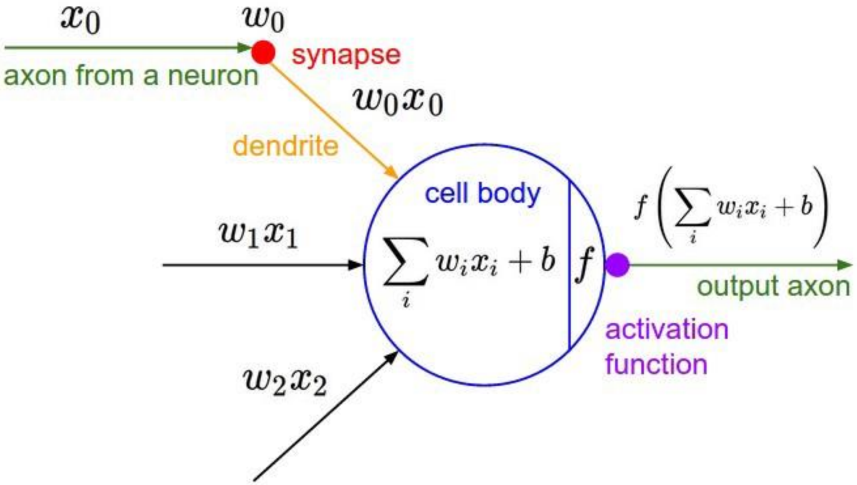
Neural Network vs Deep Learning



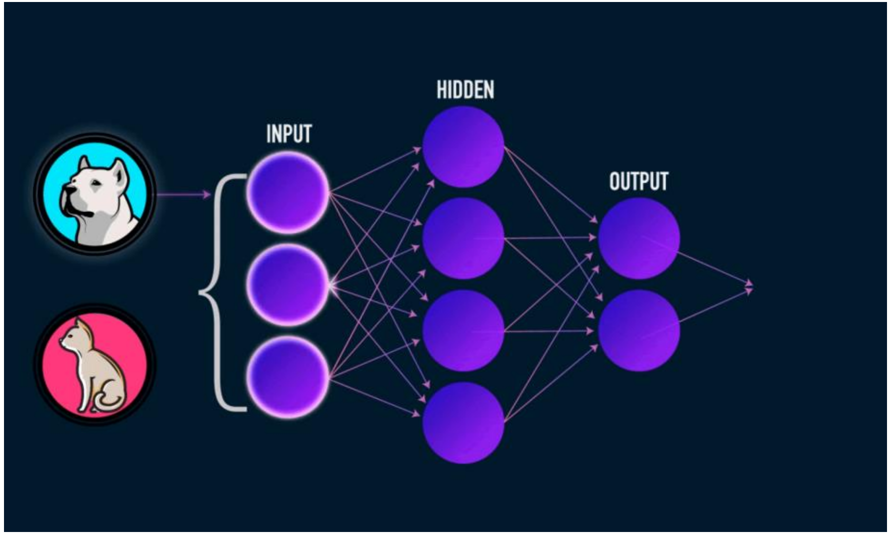
https://youtu.be/bfmFfD2RIcg

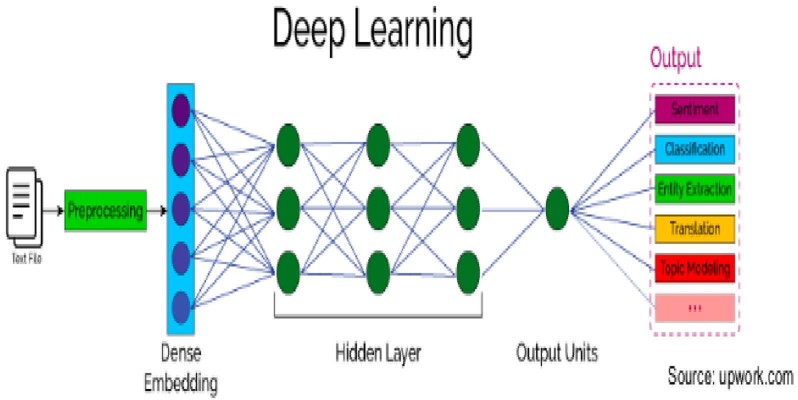
Neural Network



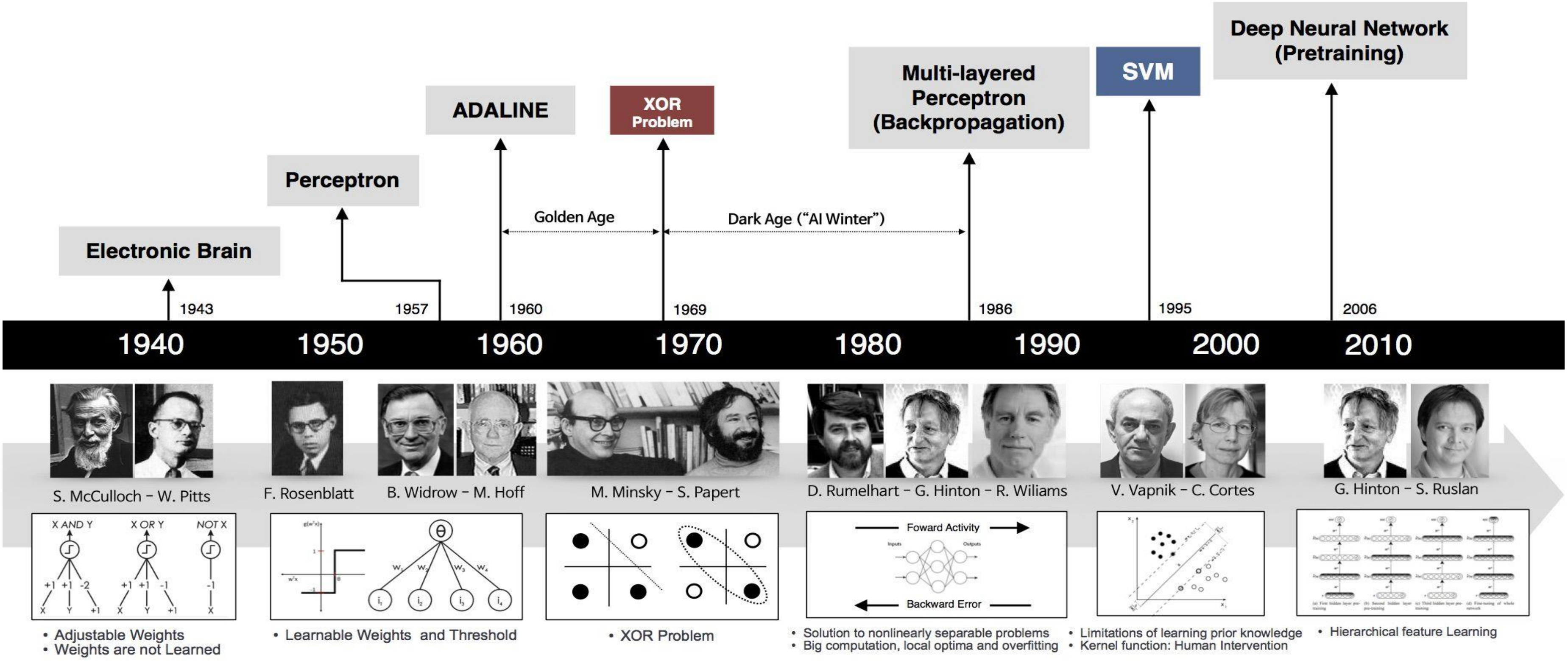


Deep Learning





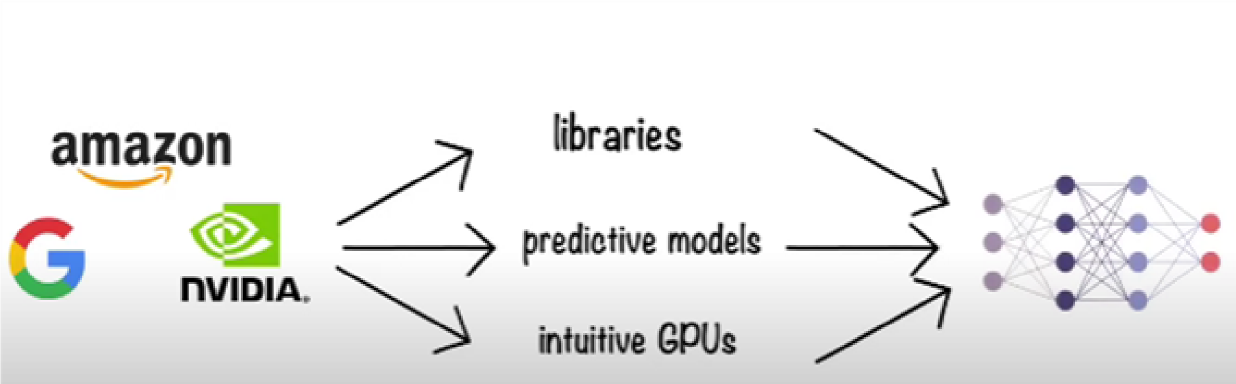
“Deep” Learning

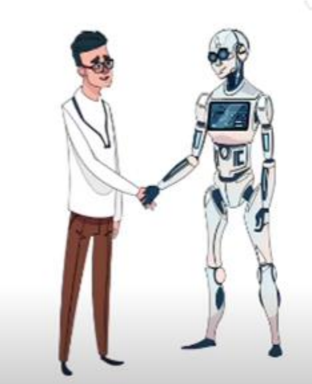


Perkembangan & Model DL

Perkembangan Deep Learning

The growth in this field has been foreseen by the big names



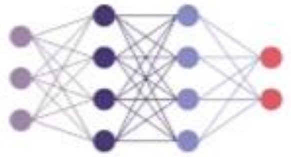


Melatih Model Deep Learning

From hours to weeks or even more!

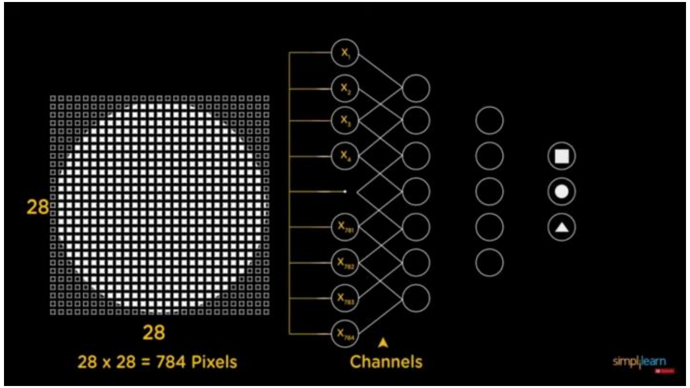
Bergantung pada faktor-faktor seperti:

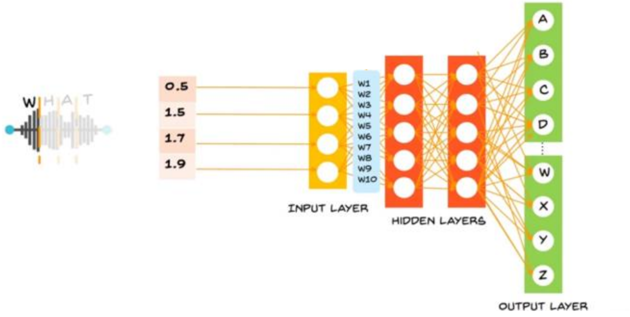
- perangkat keras yang tersedia,
- pengoptimalan,
- jumlah lapisan dalam jaringan saraf,
- arsitektur *neural network*,
- ukuran dataset, dan
- banyak lagi.



But, time is a reasonable trade-off when compared to its scope

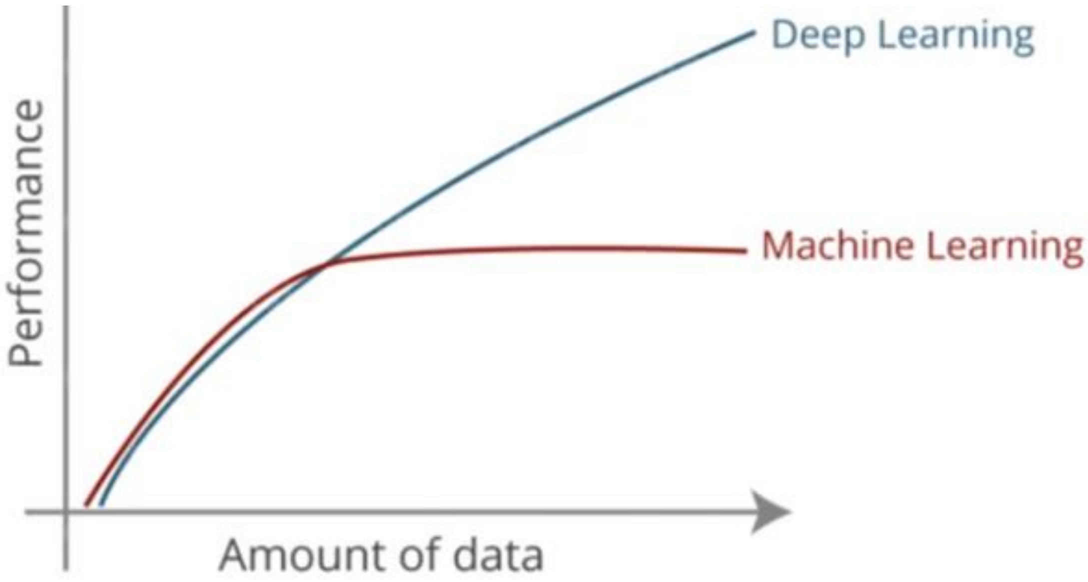
Penggunaan Deep Learning





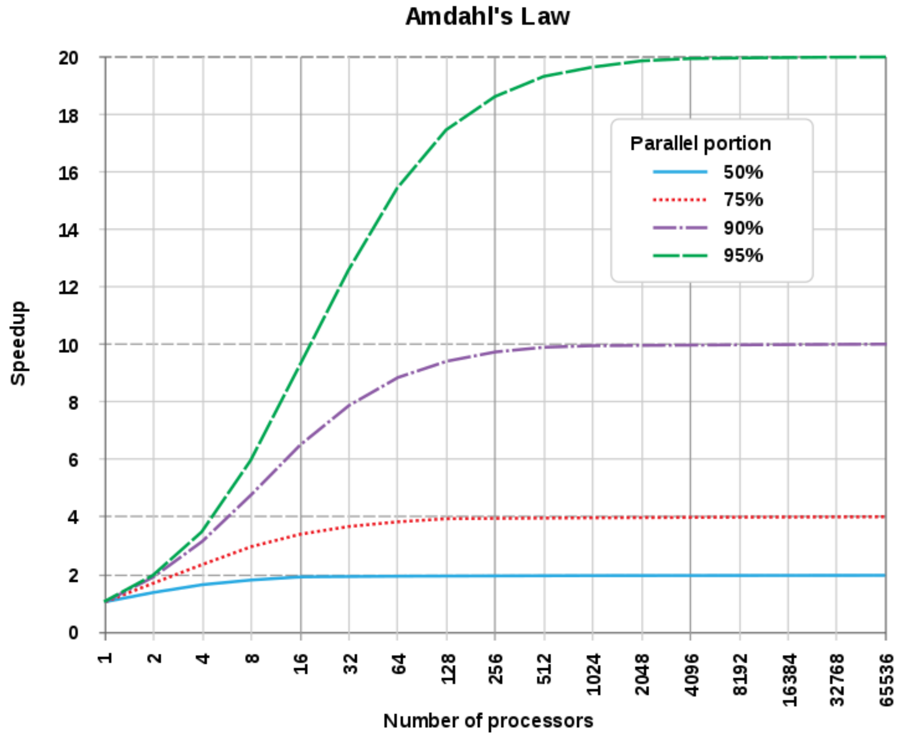
- ❑Ketika mempunyai data tak terstruktur yang sangat besar dan banyak.
- ❑Neural Network dapat mengolah data yang tidak memiliki label.
- ❑Bekerja sangat baik dengan data berbentuk *speech*, *images*, dan *video*.

ML vs DL



By: Dr. Andrew Ng

Amdahl's Law for Parallel Computing Accelerates



Persamaan untuk menghitung *Speed-up parallel computing* menggunakan hukum Amdahl's

$$Speedup = \frac{1}{(1-p) + p/N}$$

Formulasi atau hukum ini banyak dipakai dalam bidang komputasi paralel untuk meramalkan peningkatan kecepatan maksimum pemrosesan data (secara teoritis) jika jumlah prosesor di dalam komputer paralel tersebut ditambah.

GPU
THROUGHPUT TINGGI LATENSI RENDAH

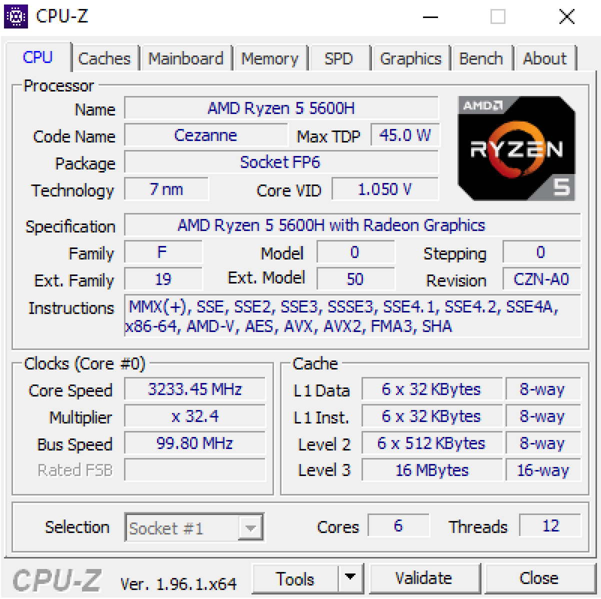
CPU
THROUGHPUT RENDAH LATENSI TINGGI

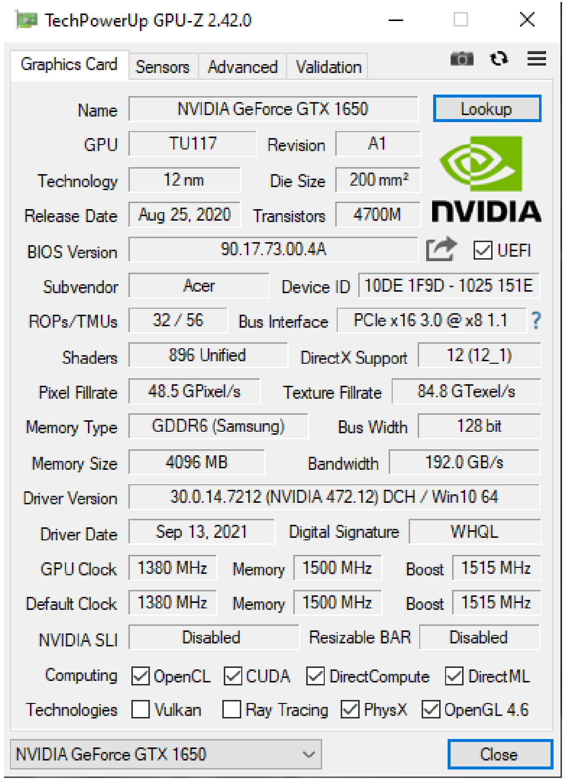
Referensi: "Hands-On GPU Programming with Python and CUDA", karangan Dr. Brian Tuomanen

Amdahl’s Law for Parallel Computing Accelerates

Kampus Merdeka INDONESIA JAYA

orbit FUTURE ACADEMY Skills For Future Jobs





Contoh perbandingan jumlah core:

1. GPU Nvidia GeForce GTX 1650 memiliki 896 CUDA cores
2. CPU AMD Ryzen 5 5600H memiliki 6 cores

Apakah harus selalu menggunakan Deep Learning?

Jawabannya adalah **Tidak!**

- Deep Learning sangat boros di komputasional

- Untuk melakukan sebuah *task*, deep learning memproses data yang banyak dengan menggunakan GPU dan memakan waktu cukup lama.

- Untuk permasalahan dengan data yang tidak memerlukan perhitungan complex, penggunaaan **Machine learning** dengan komputasi yang lebih ringan **sangat disarankan**

- Deep Learning untuk pekerjaan prediksi dengan jumlah data yang kecil bisa tidak seakurat algoritma Machine Learning. Akurasi yang sama atau lebih tinggi bisa dihasilkan bila deep learningnya dilatih dengan baik.

Kekurangan Deep Learning

- Jumlah data yang dibutuhkan dalam pelatihan cukup besar, sehingga membutuhkan akses ke komputer yang berkekuatan tinggi yang dilengkapi dengan GPU atau TPU (High Performance Computing - HPC).

- Deep-neural network sulit menentukan *tuning* terbaik dikarenakan menggunakan teknik **Black Box** yang dimana bobot akan di-update di setiap layernya.

Machine learning as a service (MLaaS)

Kampus Merdeka INDONESIA JAYA
orbit FUTURE ACADEMY Skills For Future Jobs

Machine learning as a service (MLaaS) adalah definisi umum dari berbagai platform berbasis cloud yang mencakup sebagian besar masalah infrastruktur seperti pra-pemrosesan data, pelatihan model, dan evaluasi model, dengan prediksi lebih lanjut. Hasil prediksi dapat dijembatani dengan REST API.

CLOUD MACHINE LEARNING SERVICES COMPARISON

	Amazon ML and SageMaker	Microsoft Azure AI Platform	Google AI Platform (Unified)	IBM Watson Machine Learning
Classification	✓	✓	✓	✓
Regression	✓	✓	✓	✓
Clustering	✓	✓	✓	✗
Anomaly detection	✓	✓	✗	✗
Recommendation	✓	✓	✓	✗
Ranking	✓	✓	✗	✗
Data Labeling	✓	✓	✓	✓
MLOps pipeline support	✓	✓	✓	✓
Built-in algorithms	✓	✓	✓	✗
Supported frameworks	TensorFlow, MXNet, Keras, Gluon. Pytorch, Caffe2, Chainer, Torch	TensorFlow, scikit-learn, PyTorch, Microsoft Cognitive Toolkit, Spark ML	TensorFlow, scikit-learn, XGBoost, Keras	TensorFlow, Keras, Spark MLlib, scikit-learn, XGBoost, PyTorch, IBM SPSS, PMML

altexsoft
software r&d engineering

Deep Learning Framework

Kampus Merdeka INDONESIA JAYA
orbit FUTURE ACADEMY Skills For Future Jobs

TensorFlow
mxnet
GLUON
theano
PYTORCH
K
Chainer
Caffe
Caffe2

Table 1 Some of the popular deep learning implementation tools

Tools	Platform	Support	Interface
Caffe (Williams and Zipser 1989)	Windows, Linux, Mac OSX	CNN, RNN	Python, C++, Matlab, Cuda
Tensorflow (Salakhutdinov and Hinton 2009)	Windows, Linux, Mac OSX, Android	Almost support all deep learning techniques	Python
Theano (Younes 1999)	Windows, Linux, Mac OSX	Almost support all deep learning techniques	Python, Cuda
Torch (Microsoft 2016)	Windows, Linux, Mac OSX	Almost support all deep learning techniques	Lua
Keras (Delakis and Garcia 2008)	Windows, Linux, Mac OSX	Almost support all deep learning techniques	Cross-platform, Cuda
PyTorch (Xu and Su 2015)	Linux, Mac OSX	Almost support all deep learning techniques	Python, C, Cuda

Recent Advances in Deep Learning Techniques and Its Applications: An Overview
DOI : 10.1007/978-981-15-6329-4_10

Most Popular Framework Deep Learning

TensorFlow

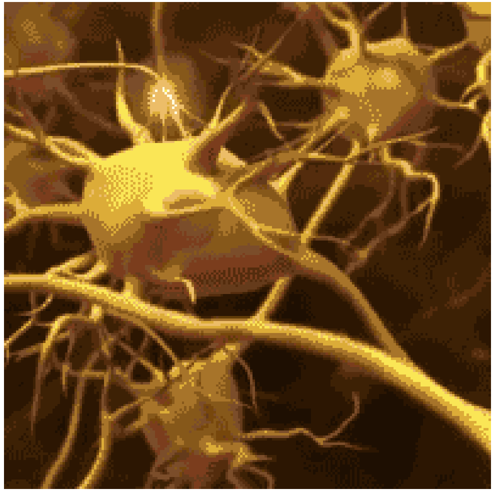
- Versi terbaru di TF saat ini adalah versi 2
- Sudah dikembangkan di bahasa pemrograman JavaScript yang bernama Tensorflow JS

PyTorch

- Mendapatkan Merger dari Framework DL yaitu CAFFE2
- Integrated dengan Framework DL yaitu YOLO V5 (You Only Look Once)

DL4J

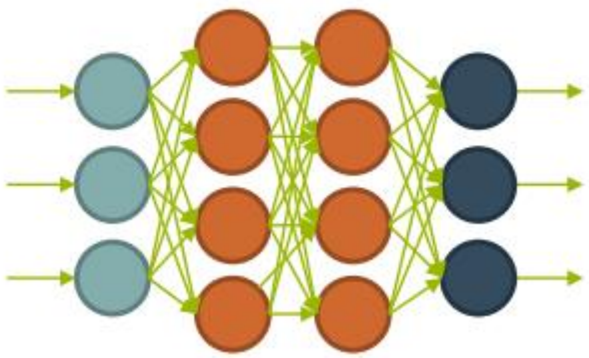
- Bisa diintegrasikan di Hadoop and Apache Spark (Data Engineering)
- Hanya Support dalam bahasa pemrograman JAVA



Session II

Artificial Neural Network

Artificial Neural Network



Artificial Neural Networks (ANN) atau Jaringan Syaraf Tiruan (JST)

Adalah paradigma pemrosesan informasi yang terinspirasi dari cara kerja sistem saraf (**otak**) manusia dalam memproses informasi.

ANN terdiri dari sejumlah besar elemen pemrosesan (**neuron**) yang saling berhubungan dan bekerja secara bersama untuk memecahkan masalah tertentu.

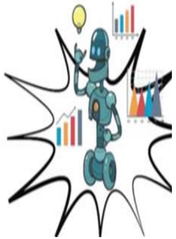
Kelebihan Neural Network

Kemampuannya berguna untuk robotika dan sistem pengenalan pola

Keluaran ANN tidak sepenuhnya dibatasi oleh masukan dan hasil yang diberikan kepada mereka pada awalnya oleh sistem pakar

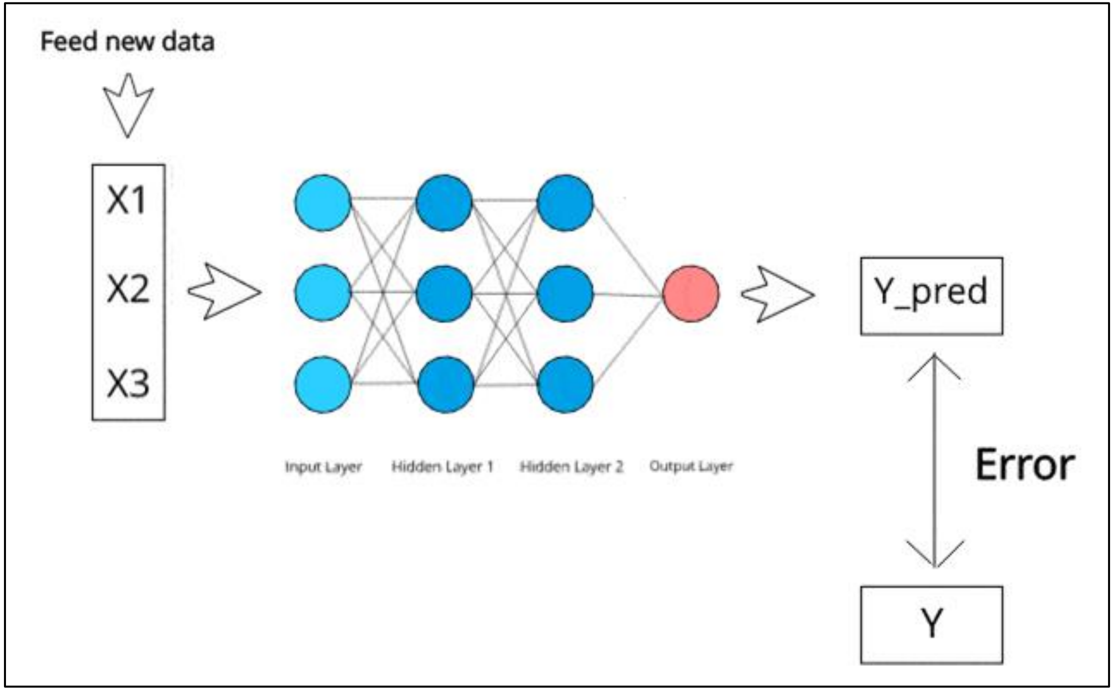
JST memiliki potensi toleransi kesalahan yang tinggi

ANN mampu men-debug atau mendiagnosis jaringan sendiri

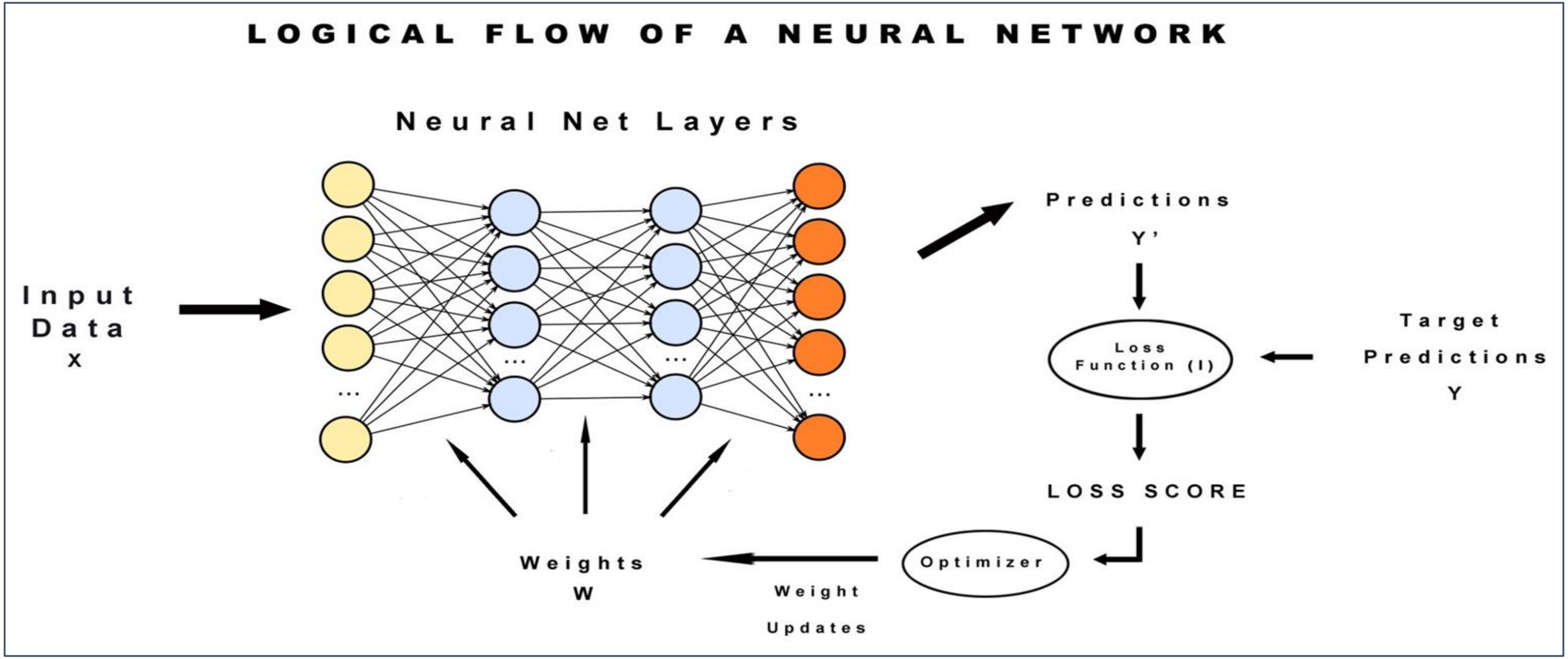


Sistem non-linear yang memiliki kemampuan untuk menemukan jalan pintas untuk mencapai solusi komputasi yang mahal

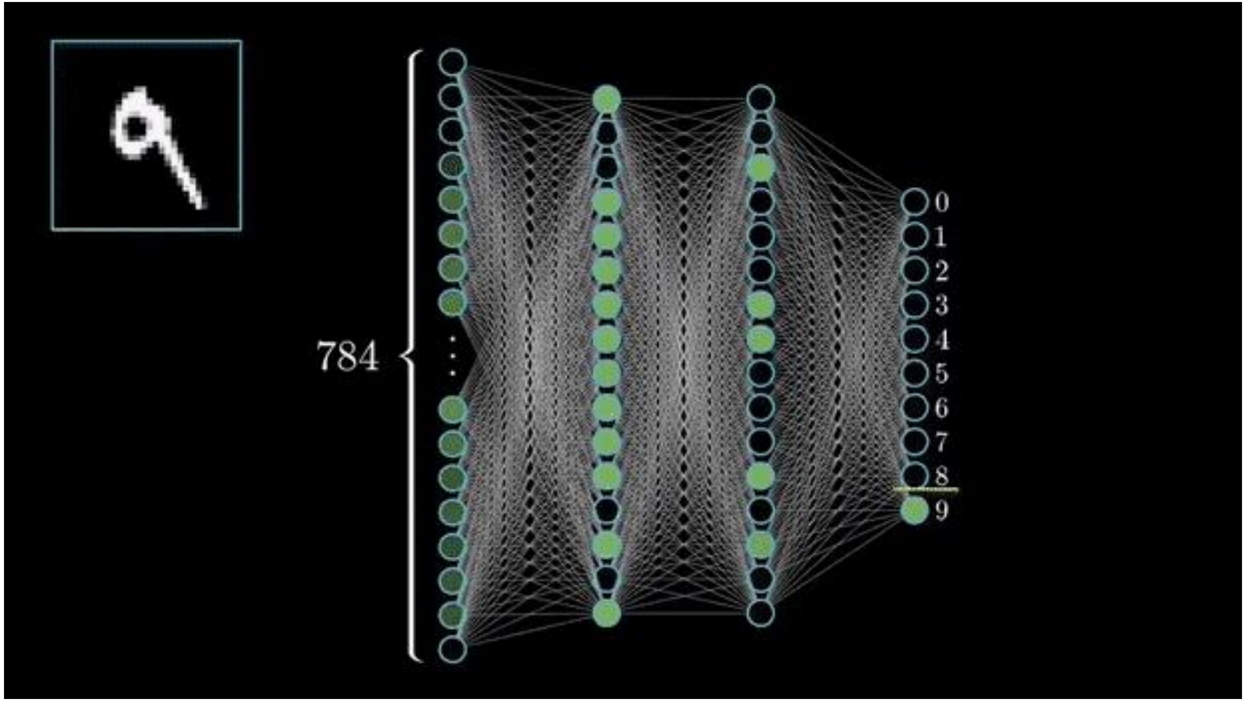
Cara Kerja ANN



Cara Kerja ANN

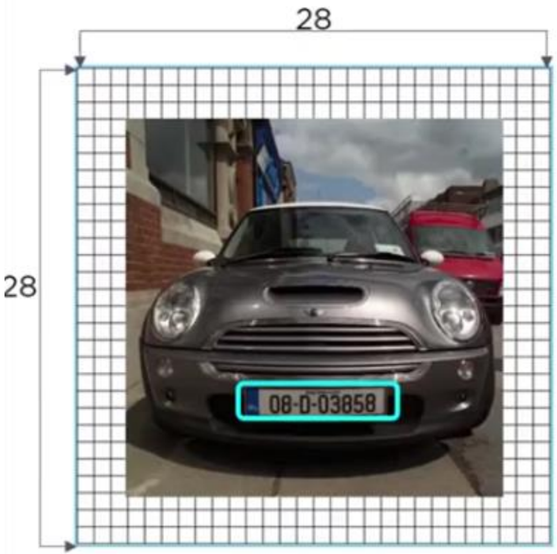


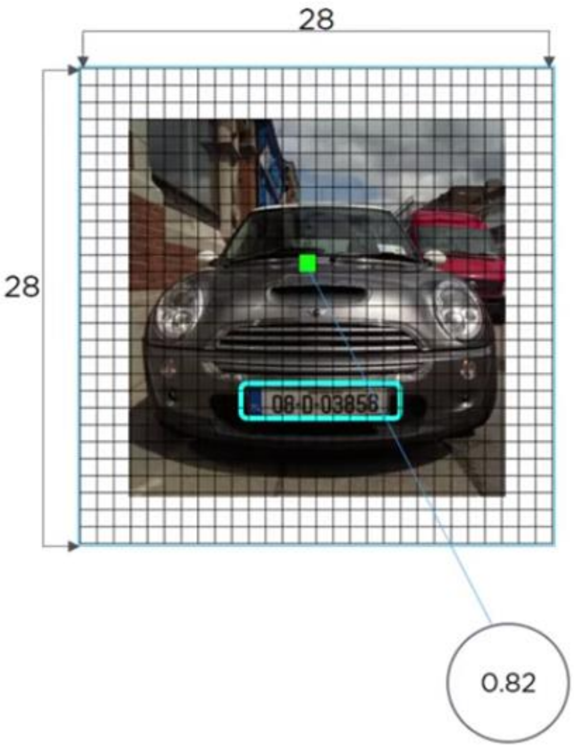
Cara Kerja ANN

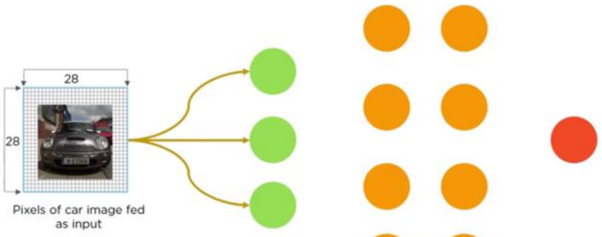


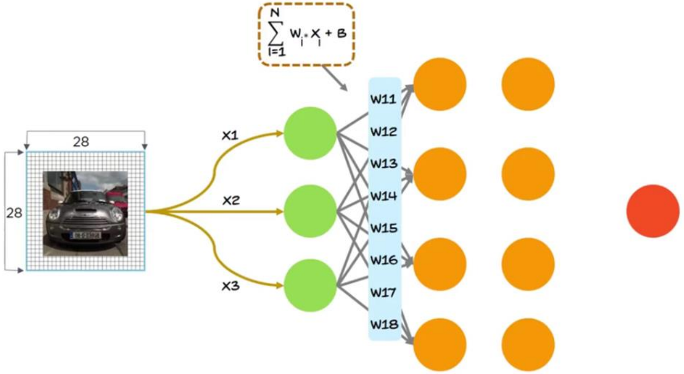
https://youtu.be/aircAruvnKk

Neuron


28
28
08-D-03858

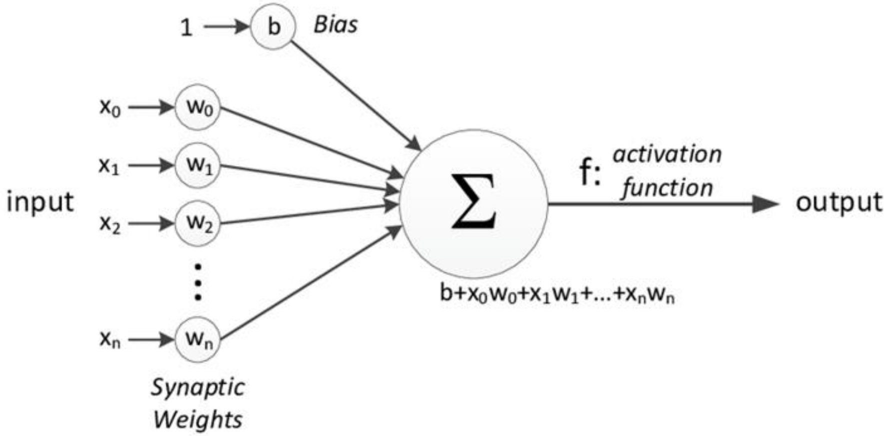

28
28
0.82


28
28
Pixels of car image fed as input


$\sum_{i=1}^{N} W_i * X_i + B$
28
28
X1
X2
X3
W11
W12
W13
W14
W15
W16
W17
W18

Formula Matematis ANN

Neural network satu lapis disebut Perceptron. Perceptron memberikan output tunggal.



x0, x1, x2, x3...x(n) mewakili berbagai masukan (variabel bebas). Masing-masing input ini dikalikan dengan bobot koneksi atau sinapsis.

Bobot direpresentasikan sebagai w0, w1, w2, w3….w(n). Bobot menunjukkan kekuatan node tertentu.

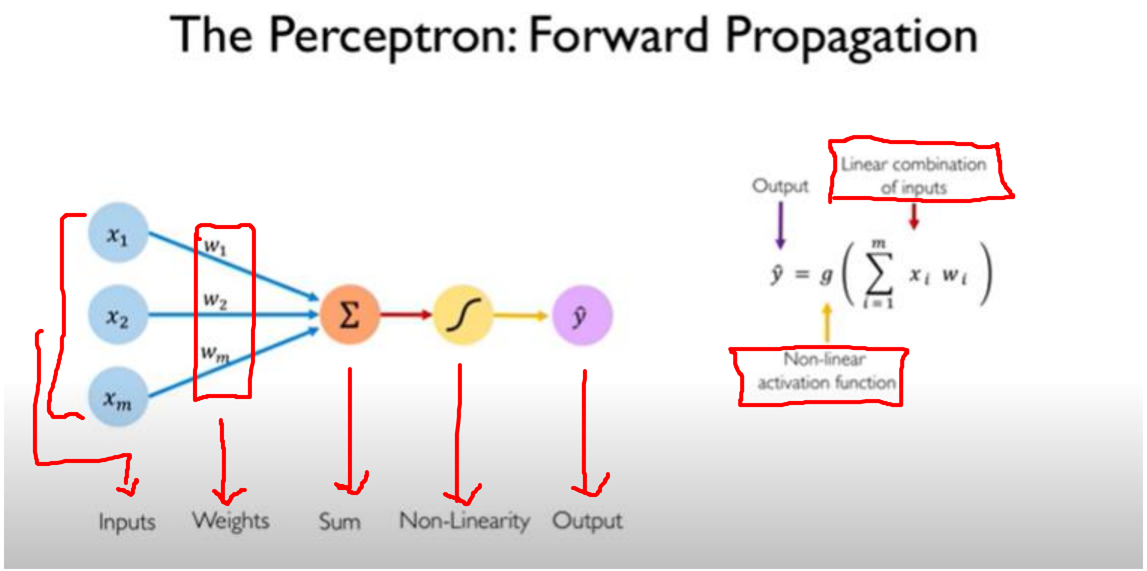
b adalah nilai bias. Nilai bias memungkinkan Anda untuk menggeser fungsi aktivasi ke atas atau ke bawah.

$$z = b + \sum_{i=1}^{m} x_i w_i$$

$$z = b + x^T w$$

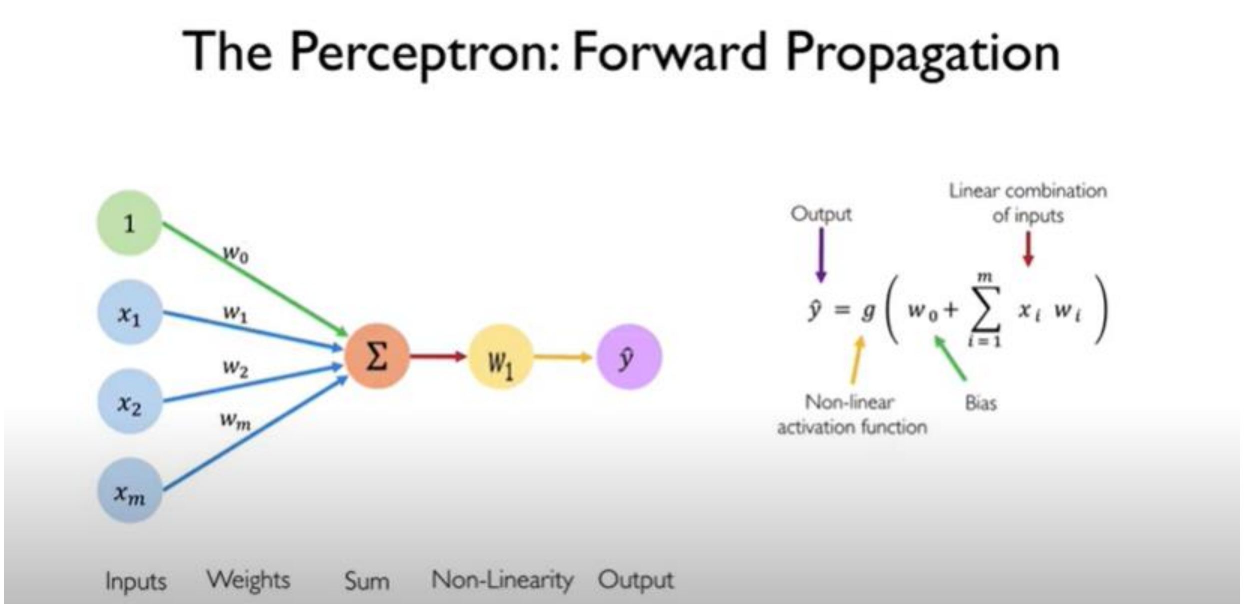
$$a = f(z)$$

Perceptron



The Perceptron: Forward Propagation
x_1
x_2
x_m
w_1
w_2
w_m
Σ
$\hat{y}$
Inputs
Weights
Sum
Non-Linearity
Output
Output
Linear combination of inputs
$\hat{y} = g\left(\sum_{i=1}^{m} x_i w_i\right)$
Non-linear activation function

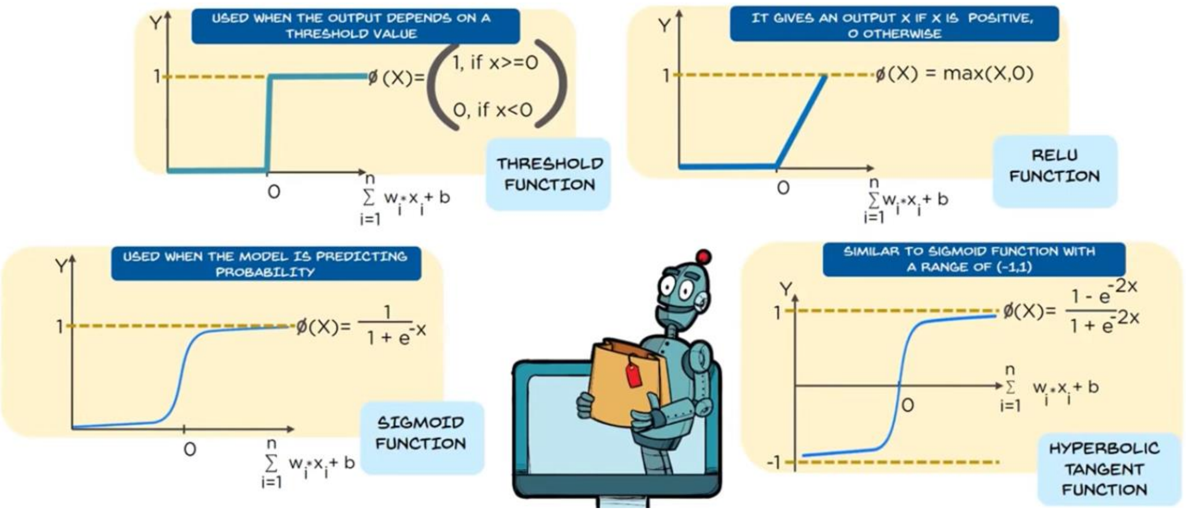
Perceptron



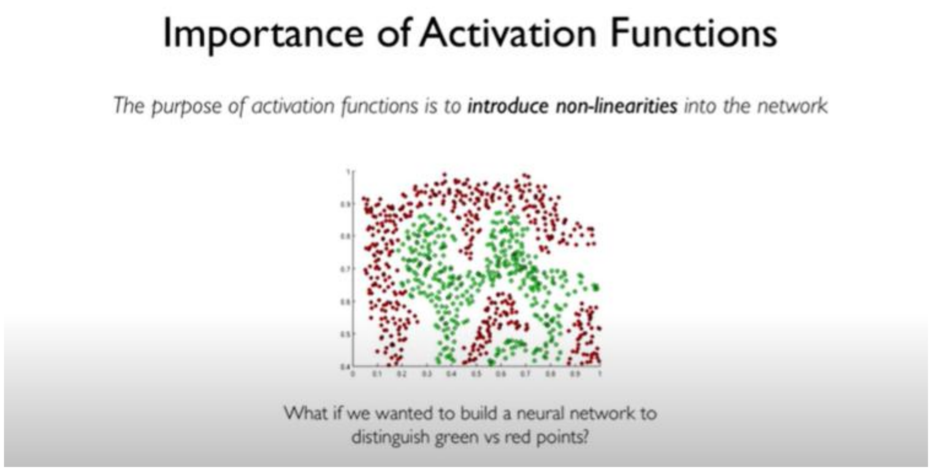
The Perceptron: Forward Propagation
1
w_0
x_1
w_1
w_2
x_2
w_m
x_m
Σ
W_1
$\hat{y}$
Inputs
Weights
Sum
Non-Linearity
Output
Output
Linear combination of inputs
$\hat{y} = g\left(w_0 + \sum_{i=1}^{m} x_i \, w_i \right)$
Non-linear activation function
Bias

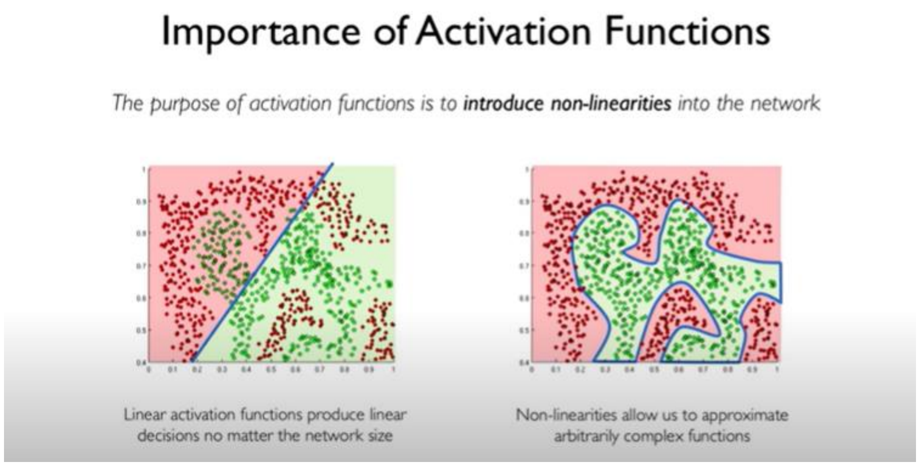
Activation Function

Activation function membantu menyelesaikan permasalah-permasalahan yang bersifat non-trivial dalam suatu neural network dengan cara mengambil sebuah nilai dan melakukan operasi matematika. Fungsi-fungsi matematika yang umum digunakan pada ANN adalah ReLu dan Sigmoid

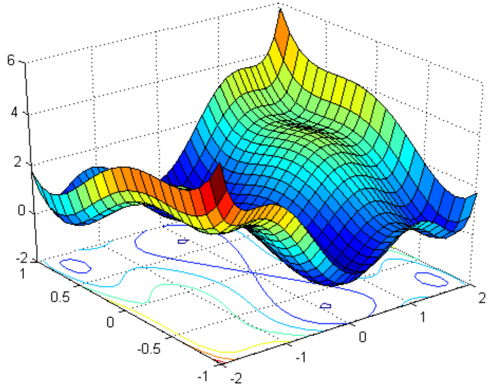


Activation Function





Loss Function



Loss function digunakan untuk mengukur seberapa bagus performa dari *neural network* dalam melakukan prediksi terhadap target.

Loss Function atau Cost Function merupakan fungsi yang menggambarkan **kerugian** yang terkait dengan semua kemungkinan yang dihasilkan oleh model.

Mean Square Error (MSE):

$$error = \frac{1}{n}\sum_{i=1}^{n}(target_i - prediksi_i)^2$$

Loss Function

Loss Optimization

We want to find the network weights that ***achieve the lowest loss***

$$W^* = \underset{W}{\operatorname{argmin}} \frac{1}{n}\sum_{i=1}^{n} \mathcal{L}(f(x^{(i)}; W), y^{(i)})$$

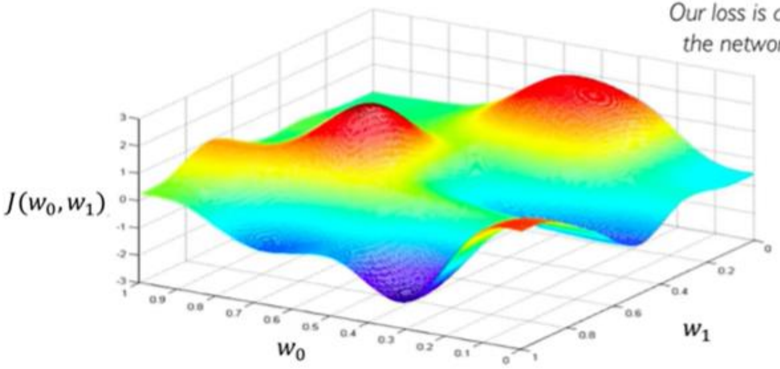
$$W^* = \underset{W}{\operatorname{argmin}} J(W)$$

Remember:
$W = \{W^{(0)}, W^{(1)}, \cdots\}$

Loss Optimization

$$W^* = \underset{W}{\operatorname{argmin}} J(W)$$

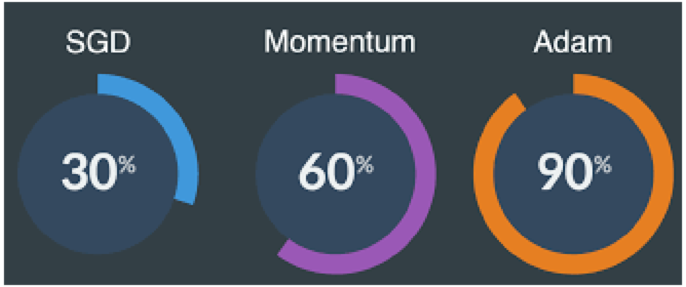
Remember:
Our loss is a function of the network weights!



Optimizer

Optimizer adalah fungsi pengoptimalan yang dapat digunakan untuk memperbarui *weight network* secara iteratif berdasarkan data training.

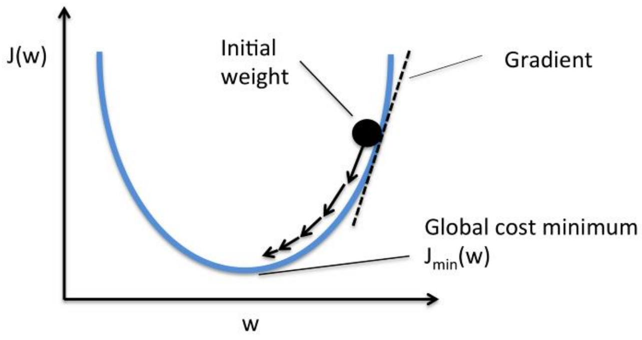
Contoh optimizer yang sering digunakan adalah gradient descent, SGD, momentum, dan ADAM (*Adaptive Moment Estimation) optimizer*.



Back Propagation

- **Back-propagation**–adalah proses memperbaharui bobot dari network untuk mereduksi error dalam hasil prediksi

- **Gradient descent,** adalah proses menyesuaikan parameter model untuk turun melalui *loss function*.



Gradient descent

Rumus utama untuk memperbaiki suatu bobot w berdasarkan error E adalah:

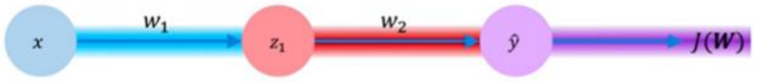
$$w_{new} = w_{old} - \alpha \frac{\partial E}{\partial w}$$

Rumus ini juga berlaku untuk memperbaiki nilai bias:

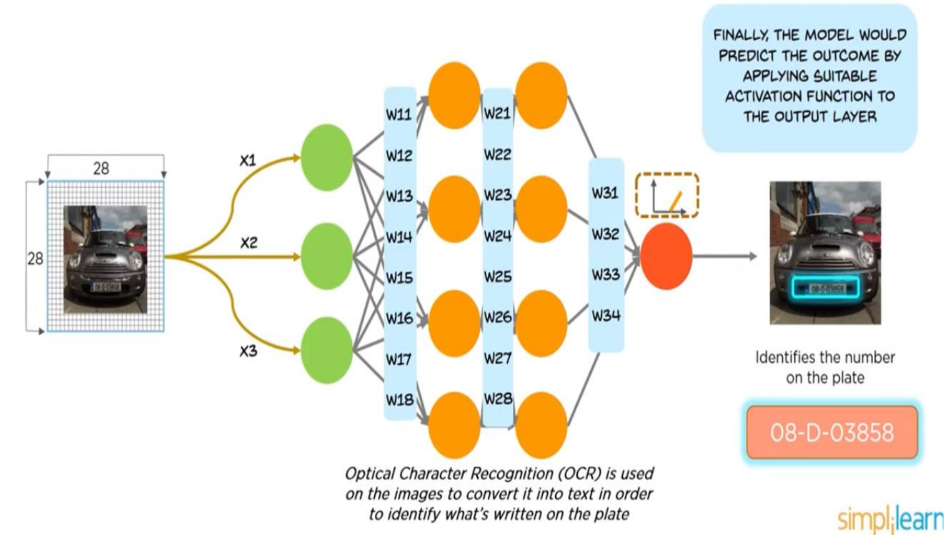
$$b_{new} = b_{old} - \alpha \frac{\partial E}{\partial b}$$

Back Propagation

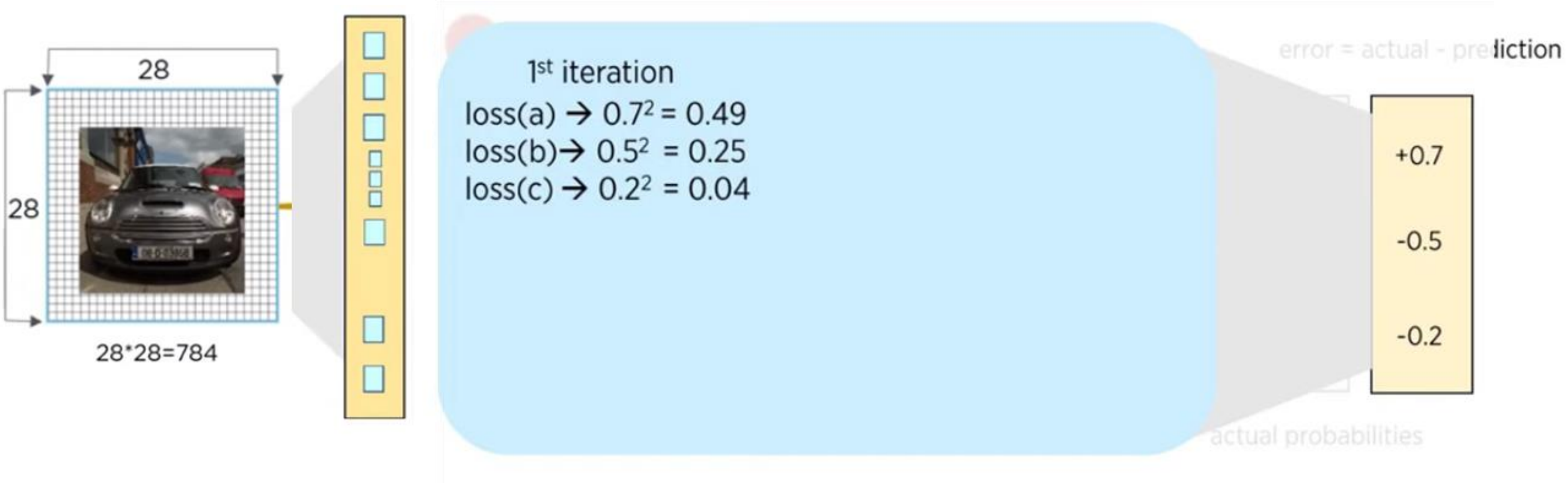
Computing Gradients: Backpropagation



$$\frac{\partial J(\boldsymbol{W})}{\partial w_1} = \frac{\partial J(\boldsymbol{W})}{\partial \hat{y}} * \frac{\partial \hat{y}}{\partial z_1} * \frac{\partial z_1}{\partial w_1}$$

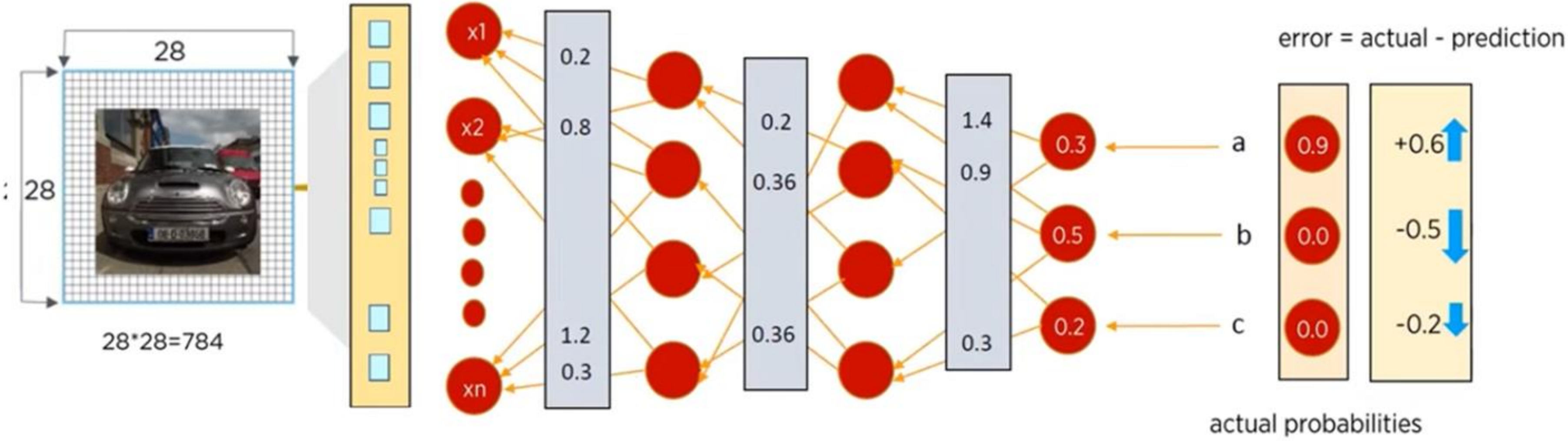


Back Propagation

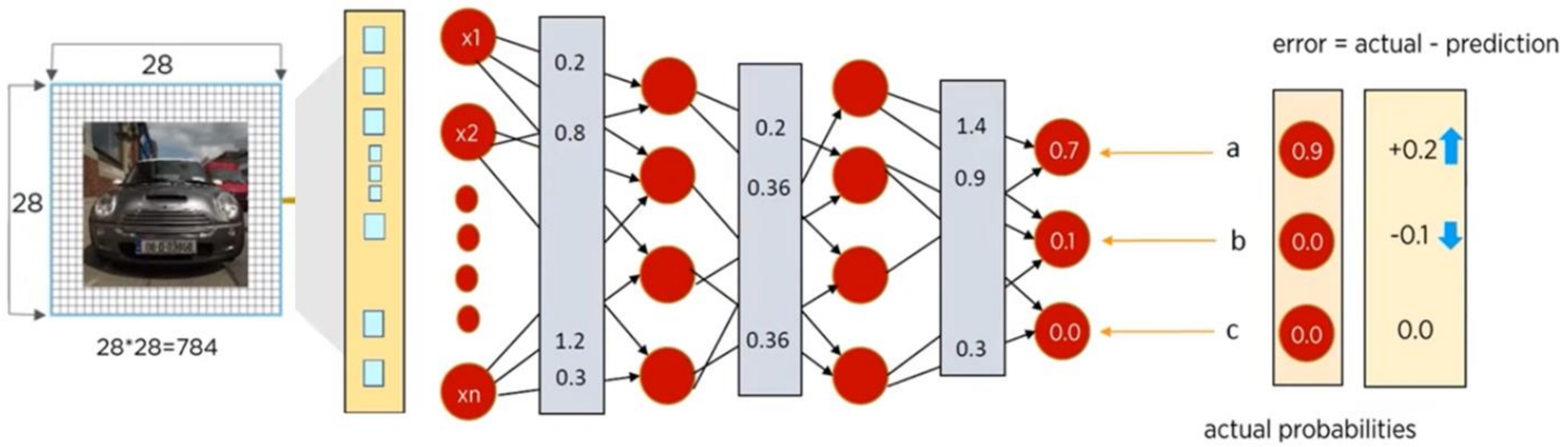


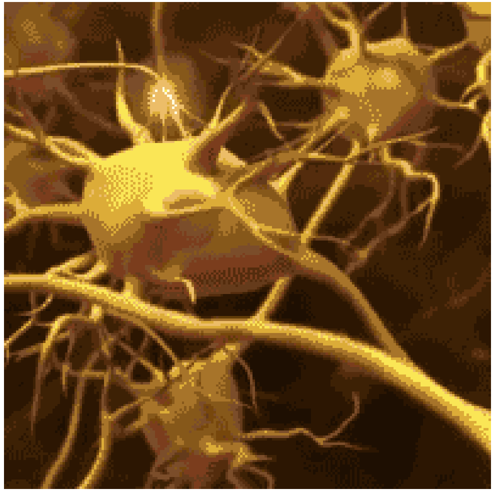
Back Propagation

Updating weight (1)



Back Propagation



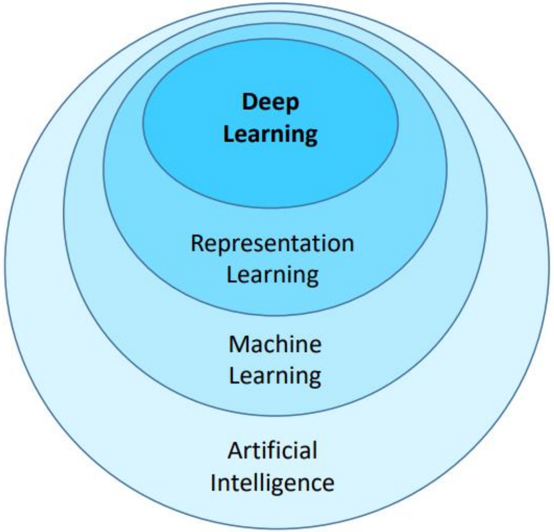


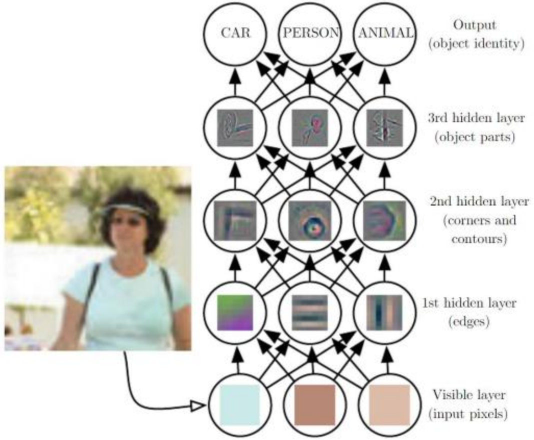
Session III

Deep Learning a.k.a Feature Learning

Deep Learning is **Representation Learning**

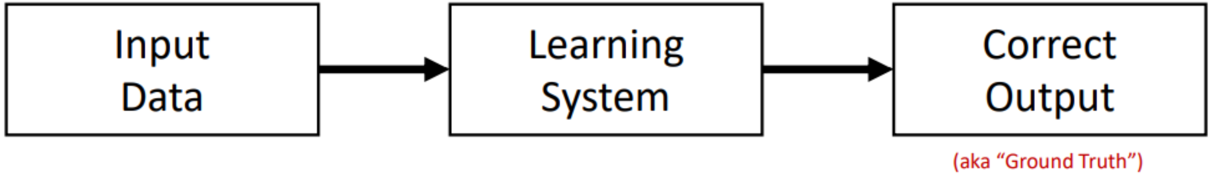
(aka Feature Learning)



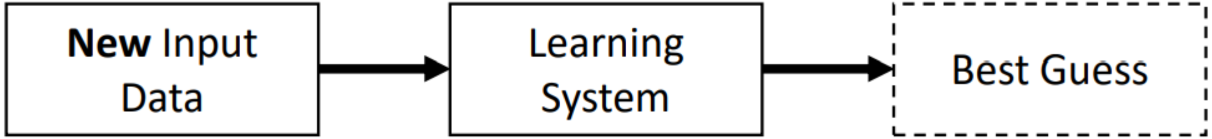


Deep Learning: Training and Testing

Training Stage:



Testing Stage:



How Neural Networks Learn: Backpropagation

Forward Pass:

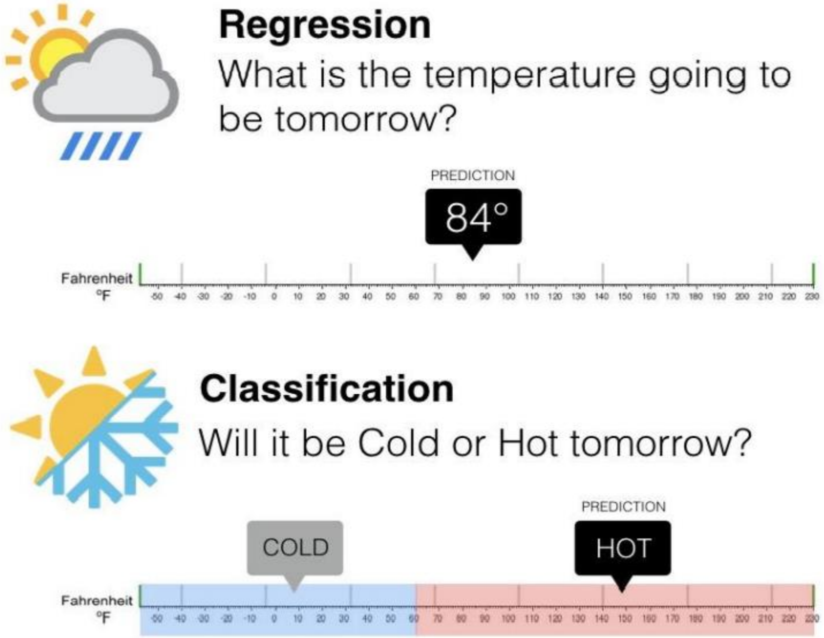
Input Data → Neural Network → Prediction

Backward Pass (aka Backpropagation):

Neural Network ← Measure of Error

Adjust to Reduce Error

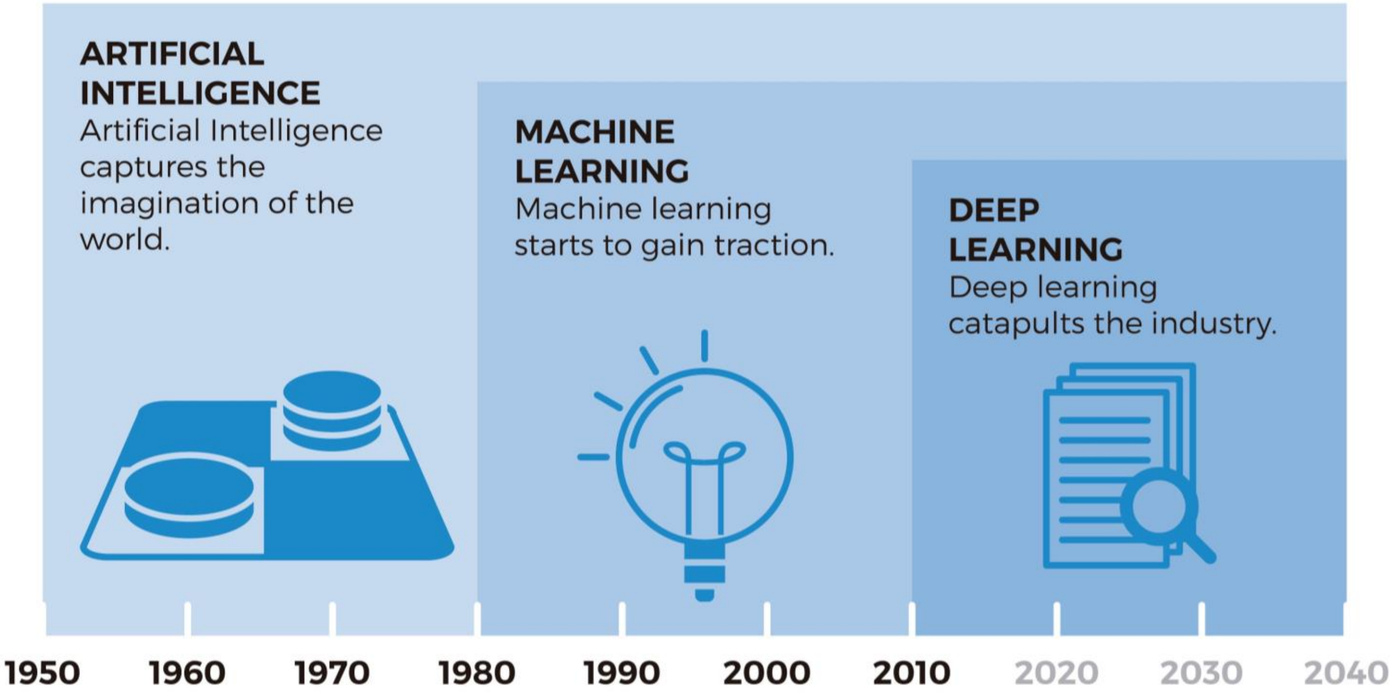
Regression vs Classification



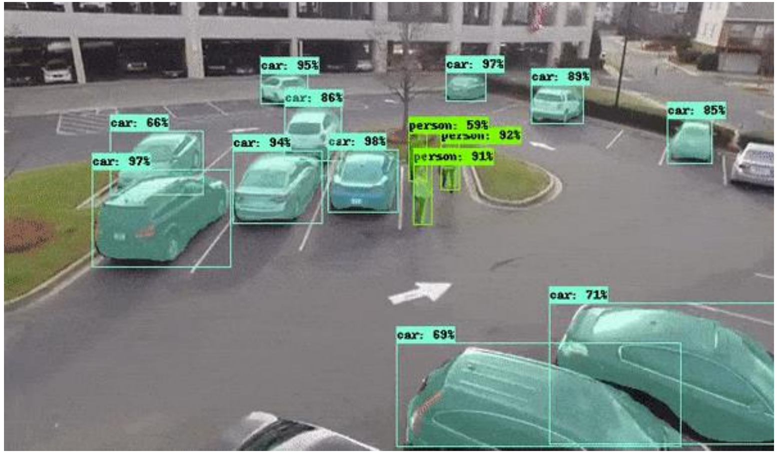
The Challenge of Deep Learning:

Efficient Teaching + Efficient Learning

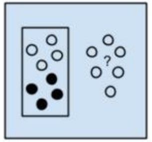
- Humans can learn from very few examples
- Machines (in most cases) need thousands/millions of examples


ARTIFICIAL INTELLIGENCE
Artificial Intelligence captures the imagination of the world.
MACHINE LEARNING
Machine learning starts to gain traction.
DEEP LEARNING
Deep learning catapults the industry.
1950
1960
1970
1980
1990
2000
2010
2020
2030
2040

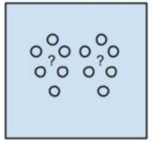
Deep Learning membantu komputer untuk mengenali objek. Tapi, bagaimana computer cara melihat?



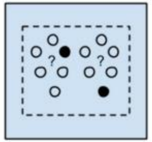
Computer Vision is Machine Learning



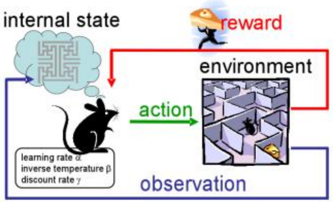
Supervised Learning



Unsupervised Learning



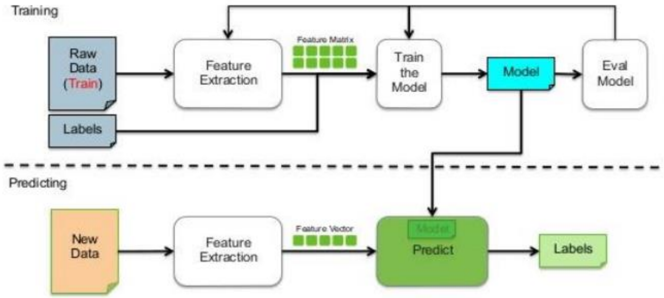
Semi-Supervised Learning



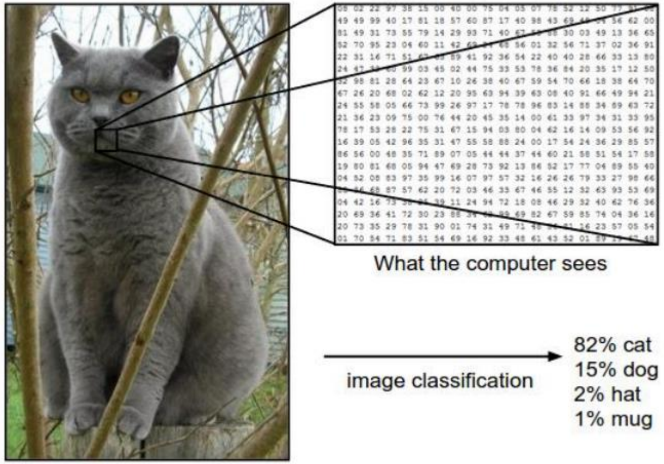
Reinforcement Learning

Computer Vision

Standard supervised learning pipeline:

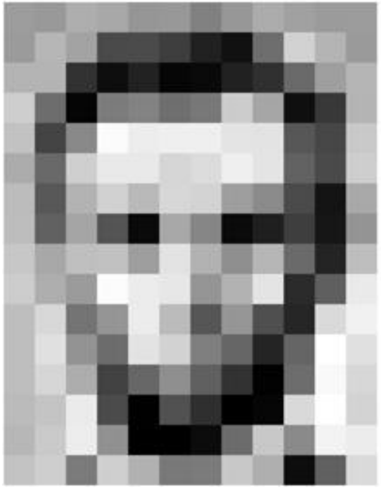


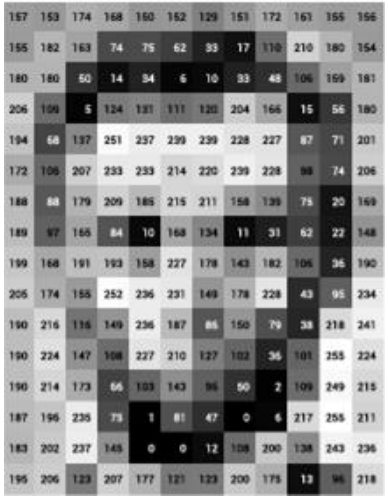
Images are Numbers



- **Regression:** The output variable takes continuous values
- **Classification:** The output variable takes class labels
  - Underneath it may still produce continuous values such as probability of belonging to a particular class.

Image are Numbers

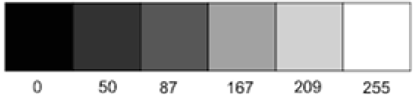




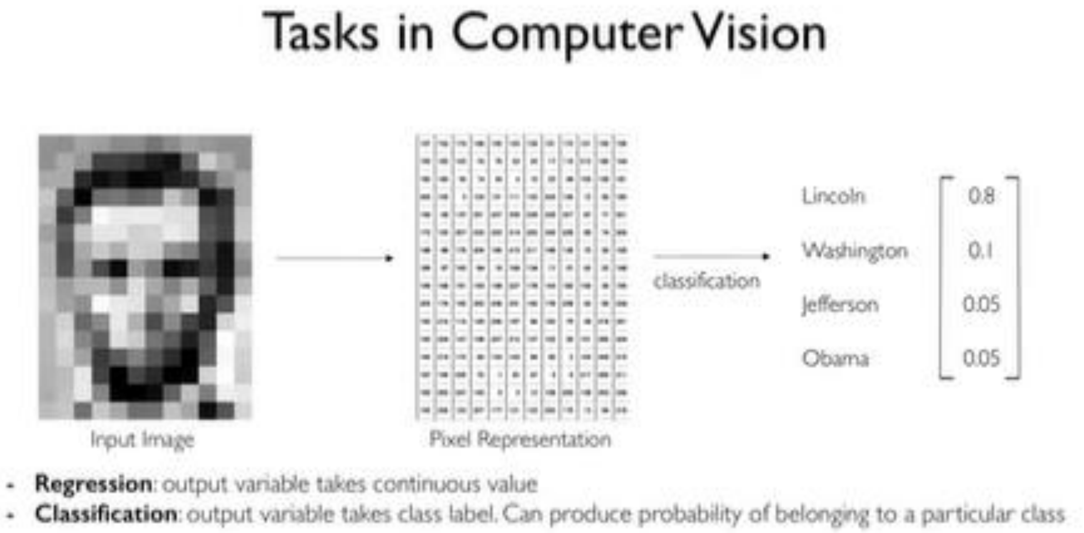
157	153	174	168	150	152	129	151	172	161	155	156
155	182	163	74	75	62	33	17	110	210	180	154
180	180	50	14	34	6	10	33	48	106	159	181
206	109	5	124	131	111	120	204	166	15	56	180
194	68	137	251	237	239	239	228	227	87	71	201
172	105	207	233	233	214	220	239	228	98	74	206
188	88	179	209	185	215	211	158	139	75	20	169
189	97	165	84	10	168	134	11	31	62	22	148
199	168	191	193	158	227	178	143	182	106	36	190
205	174	155	252	236	231	149	178	228	43	95	234
190	216	116	149	236	187	86	150	79	38	218	241
190	224	147	108	227	210	127	102	36	101	255	224
190	214	173	66	103	143	96	50	2	109	249	215
187	196	235	75	1	81	47	0	6	217	255	211
183	202	237	145	0	0	12	108	200	138	243	236
195	206	123	207	177	121	123	200	175	13	96	218

- Why we need math?
- An image is just a matrix of numbers [0,255]!

Source: ai.stanford.edu, mit



Source: processing.org



- An image is just a matrix of numbers [0,255]!
- But why must 255?

Kenapa 255?

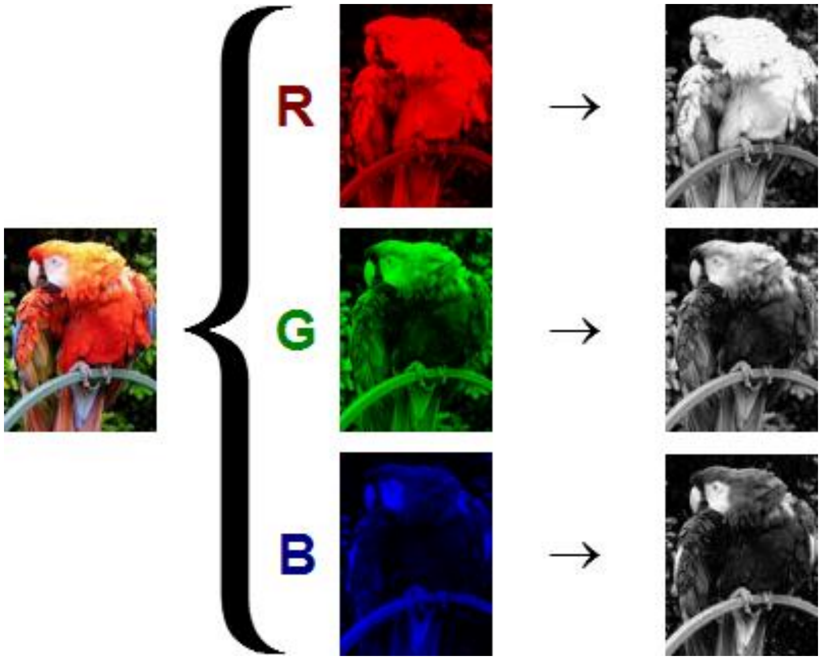
1 bit	
0	1

Number of bits	Different patterns	No. of patterns	No. of patterns
1	0 1	2^1	2
2	00 01 10 11	2^2	4
3	000 001 010 100 011 101 110 111	2^3	8

1 byte = 8 bits

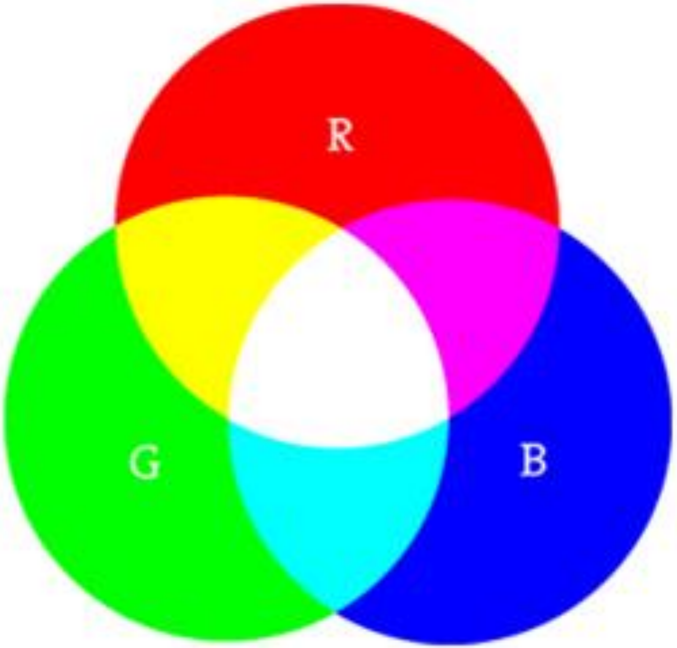
2^8 = 256

Range: 0 to 255



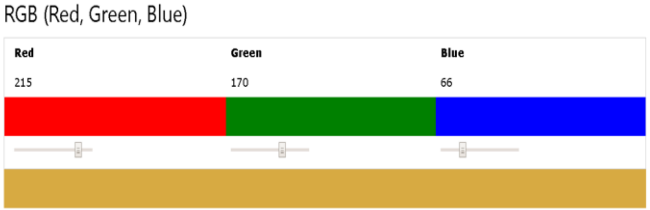
Color channel separation

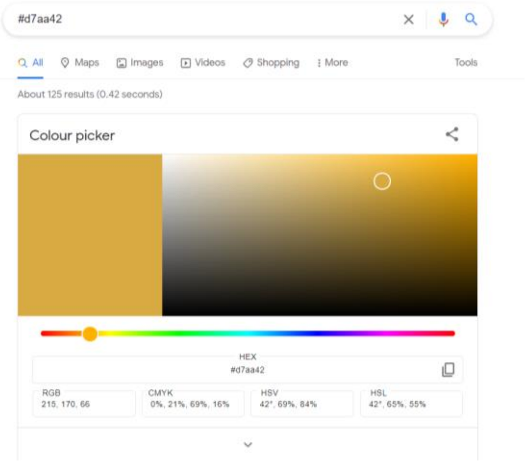
Source: Wikipedia



Source: processing.org

Let's try by yourself!





https://www.w3schools.com/colors/colors_rgb.asp

Is Computer Vision Hard?

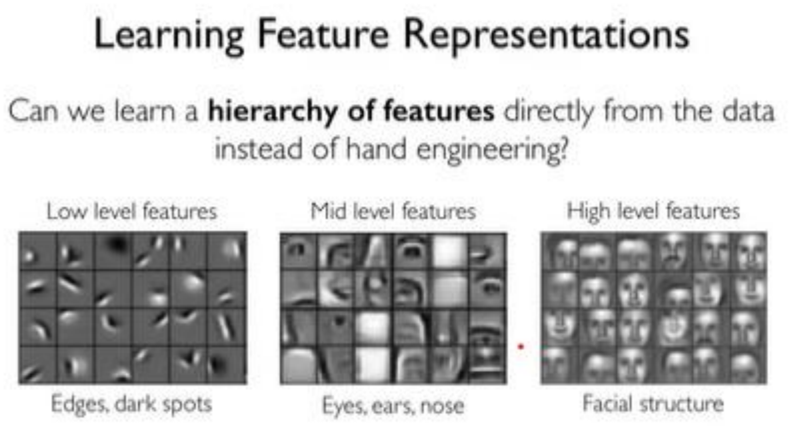
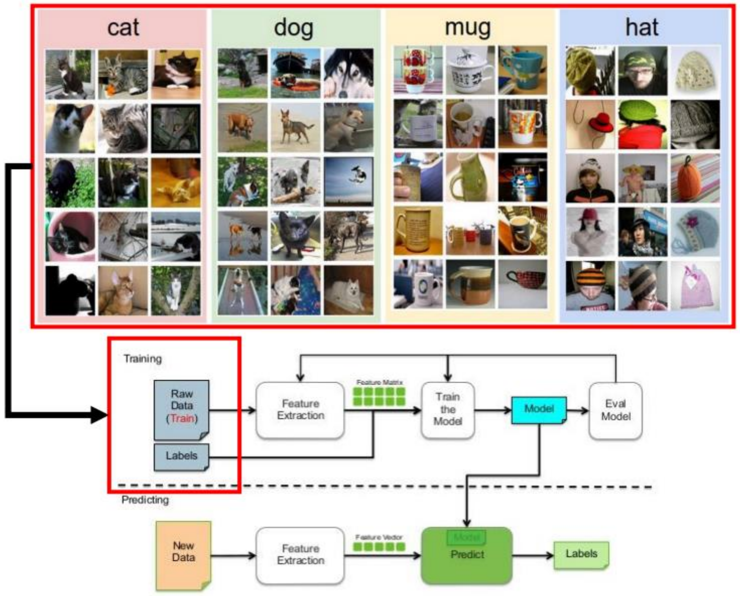
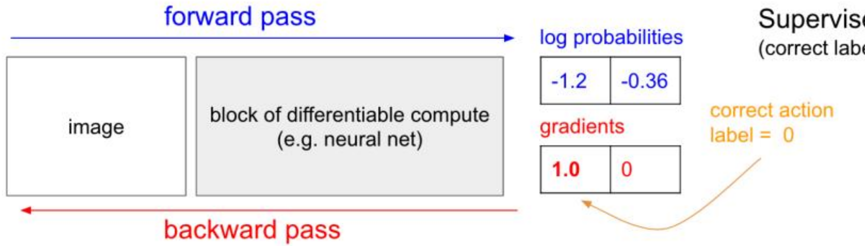


Image Classification Pipeline



Think like Machine

Reminder: "Learning" is Optimization of a Function



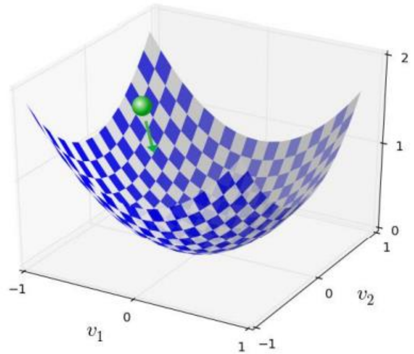
Supervised Learning
(correct label is provided)

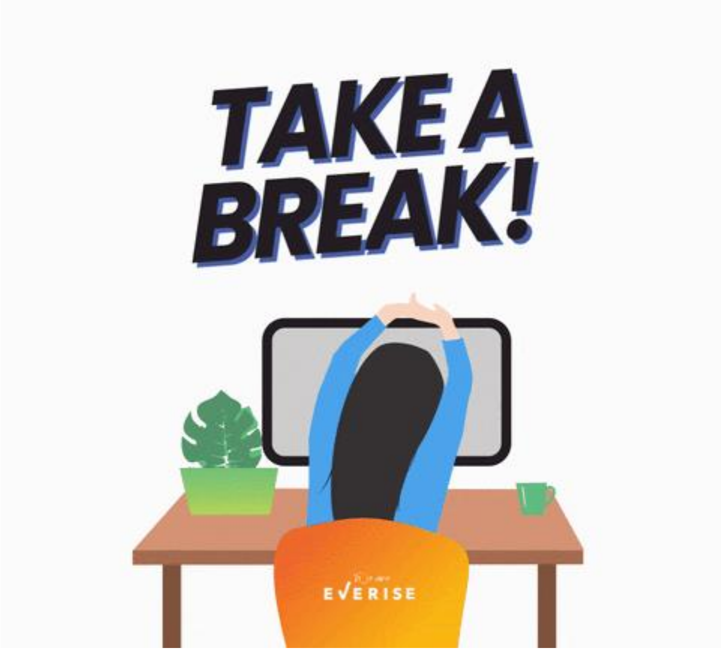
Ground truth for "6":

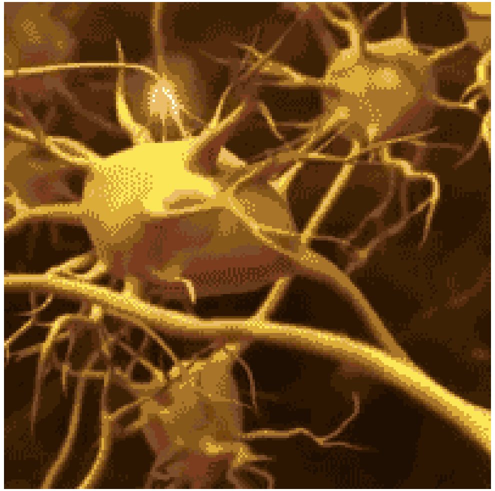
$$y(x) = (0,0,0,0,0,0,1,0,0,0)^T$$

"Loss" function:

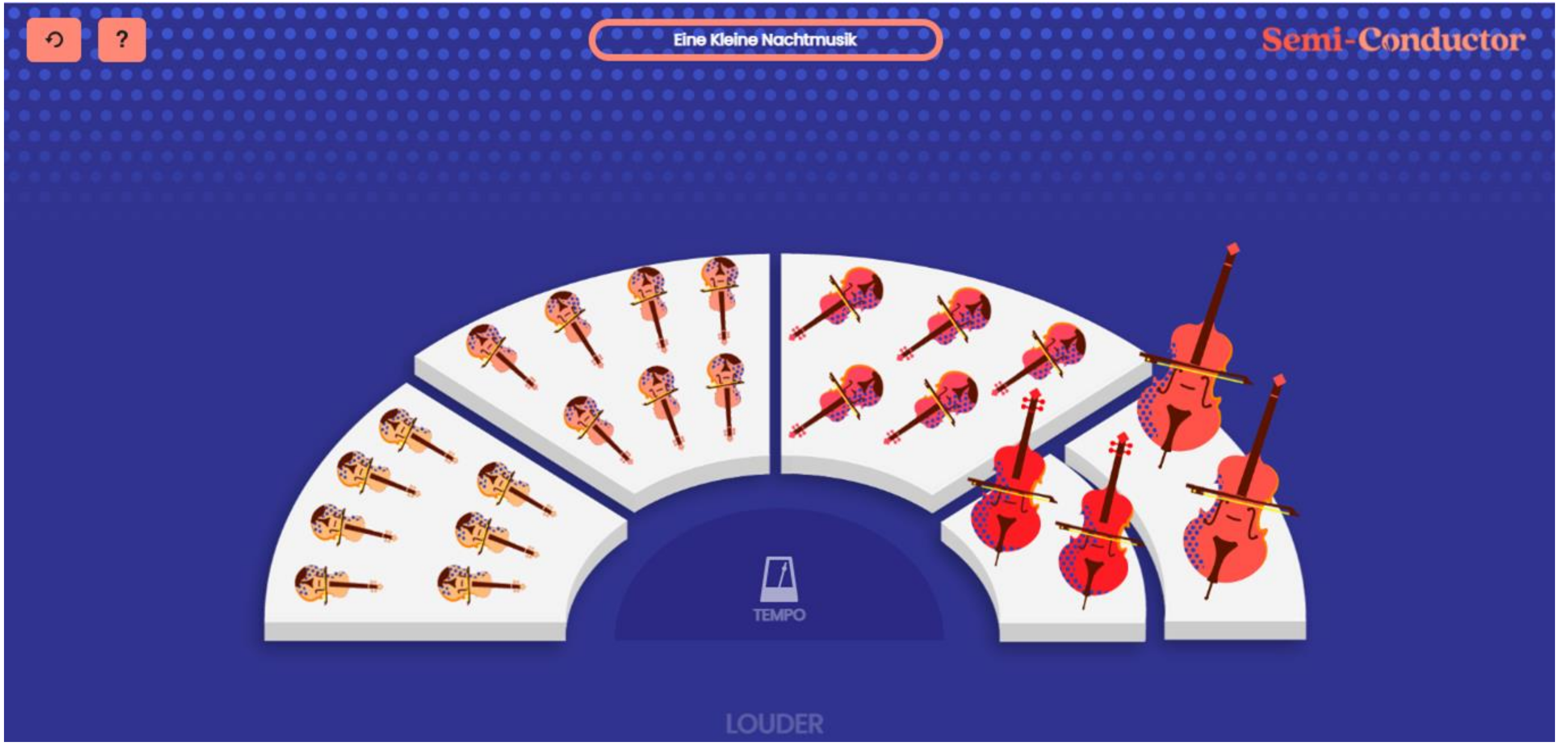
$$C(w,b) \equiv \frac{1}{2n} \sum_x \|y(x) - a\|^2$$



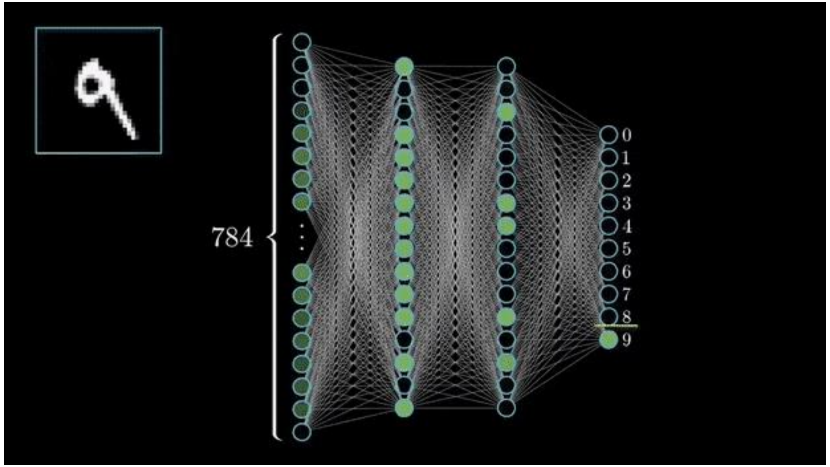

TAKE A
BREAK!
EVERISE

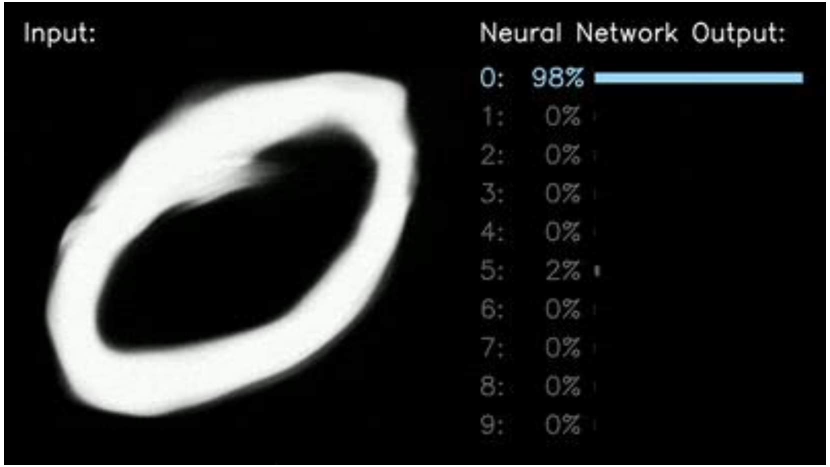


Session IV
Deep Learning; Example

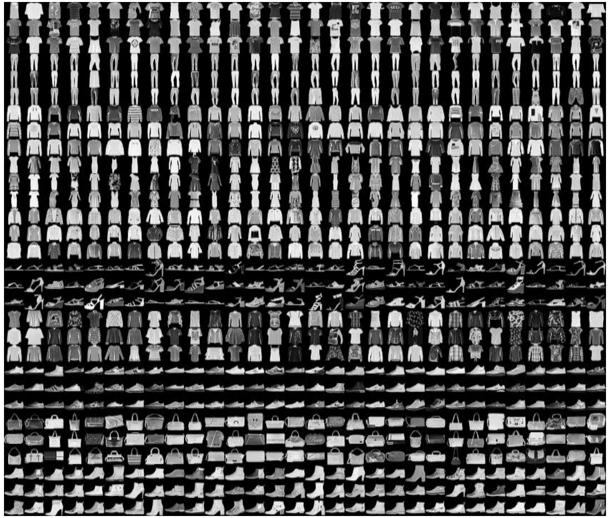

Eine Kleine Nachtmusik
Semi-Conductor
TEMPO
LOUDER

Digit Classification
MNIST Dataset

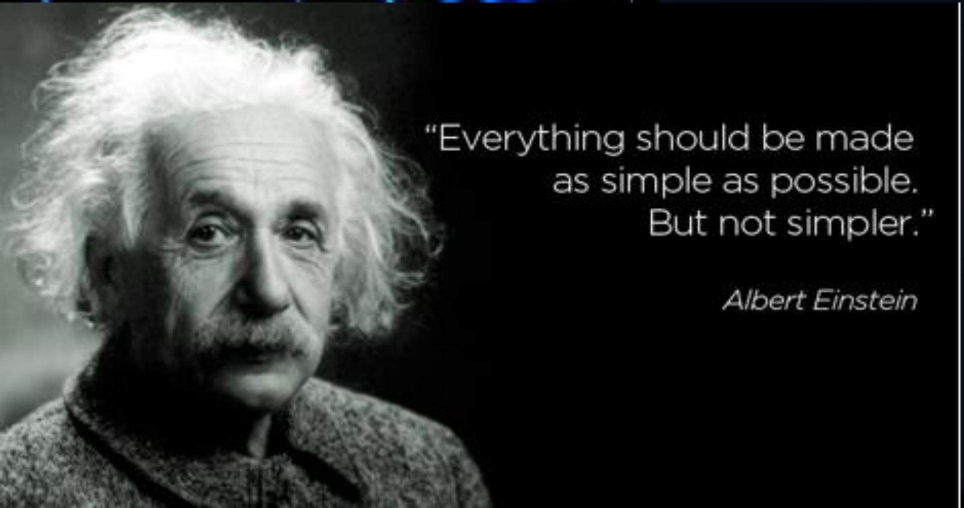


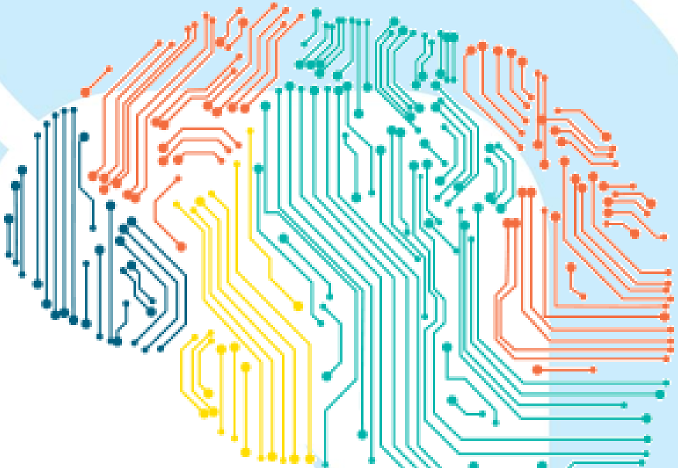


The Fashion MNIST Dataset



orbit FUTURE ACADEMY | Skills For Future Jobs





TERIMA KASIH

THANK YOU

Orbit Future Academy

PT Orbit Ventura Indonesia
Center of Excellence (Jakarta Selatan)
Gedung Veteran RI, Lt.15
Unit Z15-002, Plaza Semanggi
Jl. Jenderal Sudirman Kav.50, Jakarta
12930, Indonesia

- Jakarta Selatan/Pusat
- Jakarta Barat/BSD
- Kota Bandung
- Kab. Bandung
- Jawa Barat

Hubungi Kami

Director of Sales & Partnership
ira@orbitventura.com
+62 858-9187-7388

Social Media

- Facebook: Orbit Future Academy
- Twitter: @OrbitFutureAcademyIn1
- Instagram: OrbitFutureAcademy
- YouTube: Orbit Future Academy