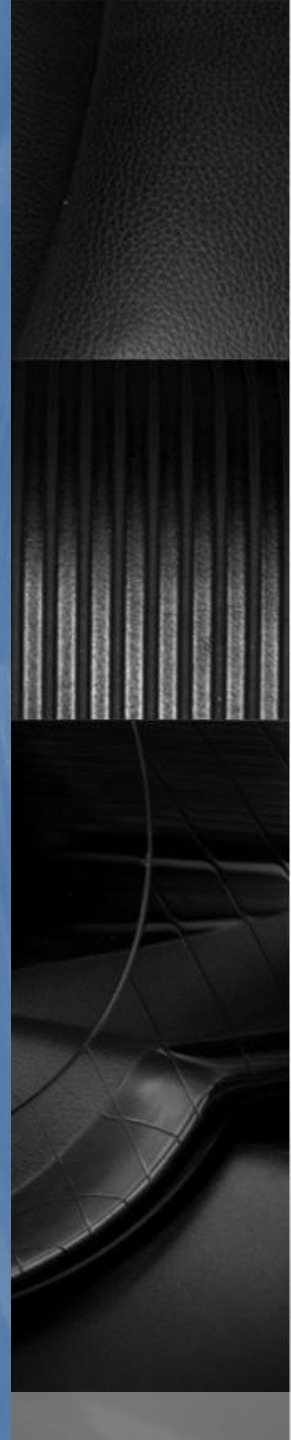
# COMP4431 Artificial Intelligence

## Deep Learning

Raymond Pang
Department of Computing
The Hong Kong Polytechnic University



# Quiz 4 (last) Arrangement (held on 20 Mar)

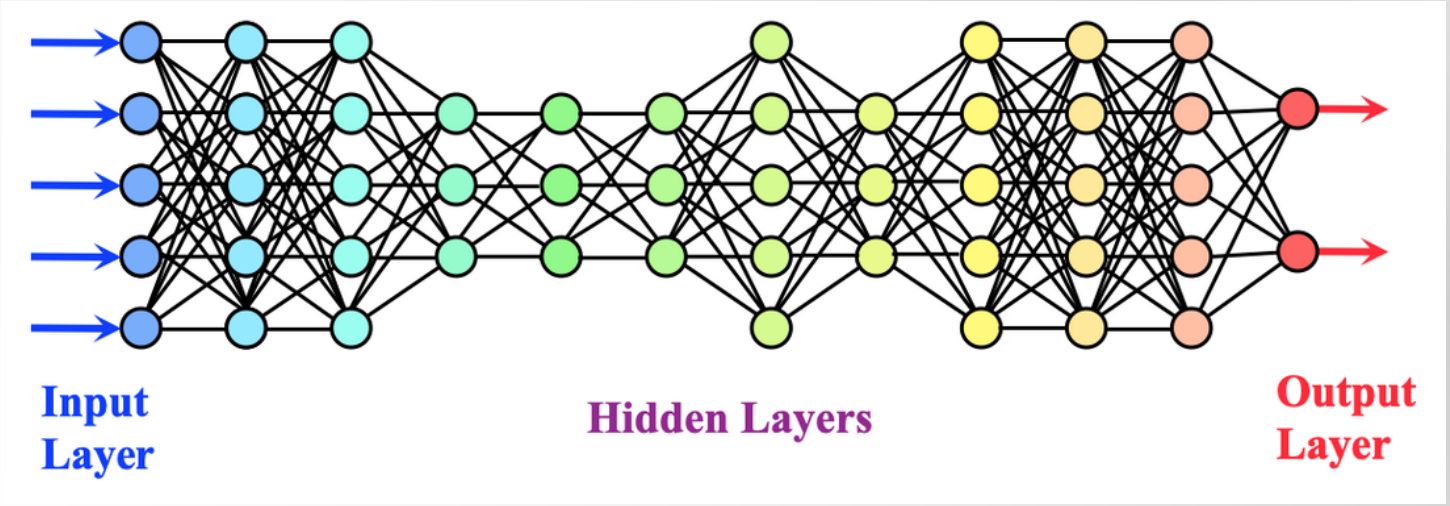
- Quiz is open book and notes
- Scope related to **Lecture 6 and 7**
- Use of mobile phone, Internet and other electronic devices are NOT ALLOWED!!
- Except a **Calculator**!! Please bring one !!
- Quiz will be held at the end of the lecture session
- Quiz time is **20 mins**
- Lecture will end at 7:40pm, Quiz starts at **7:50pm**

# Overview

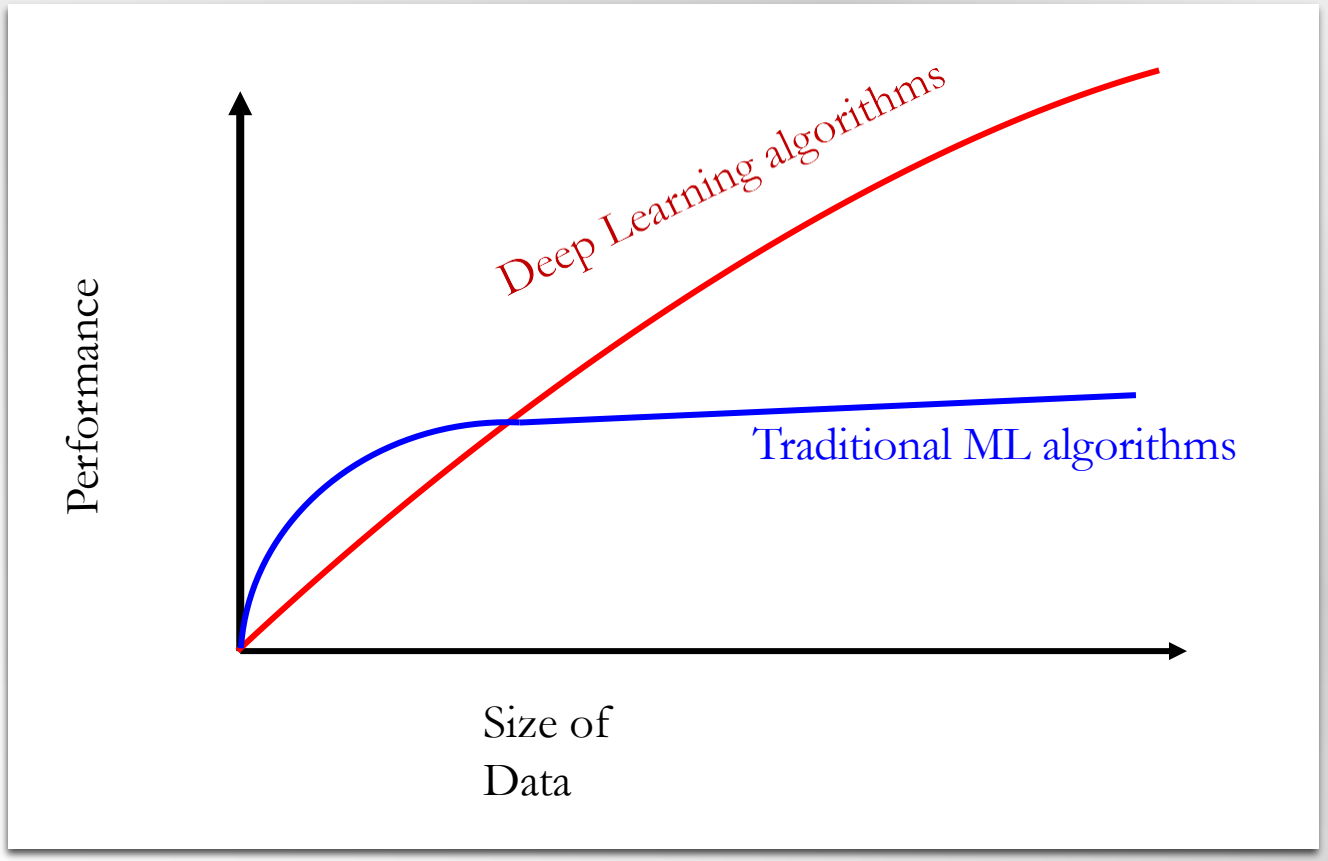
- Basics of Deep Learning
- Convolutional Neural Networks (CNN)
- Training and Hyper-parameters

# Deep Learning

- Based on Neural Network model, but "Deep"er than classical ones
- As computation power increase, it allows more hidden layers to be trained

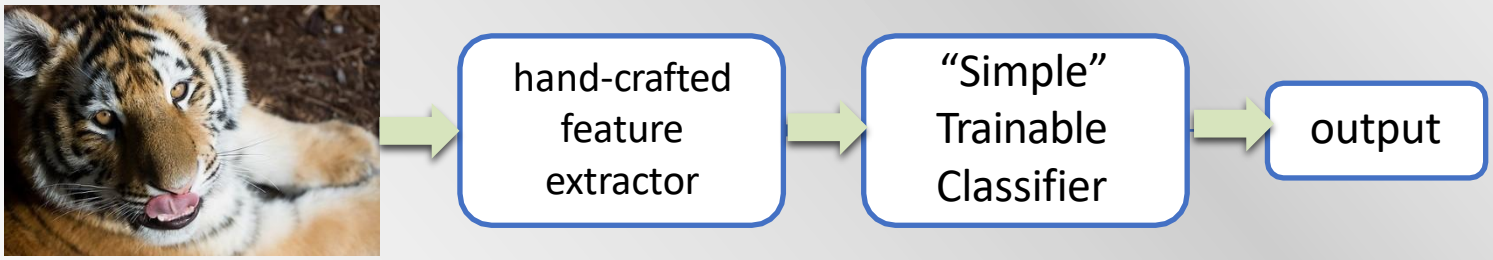


# Deep learning Vs. traditional ML



# Deep Learning

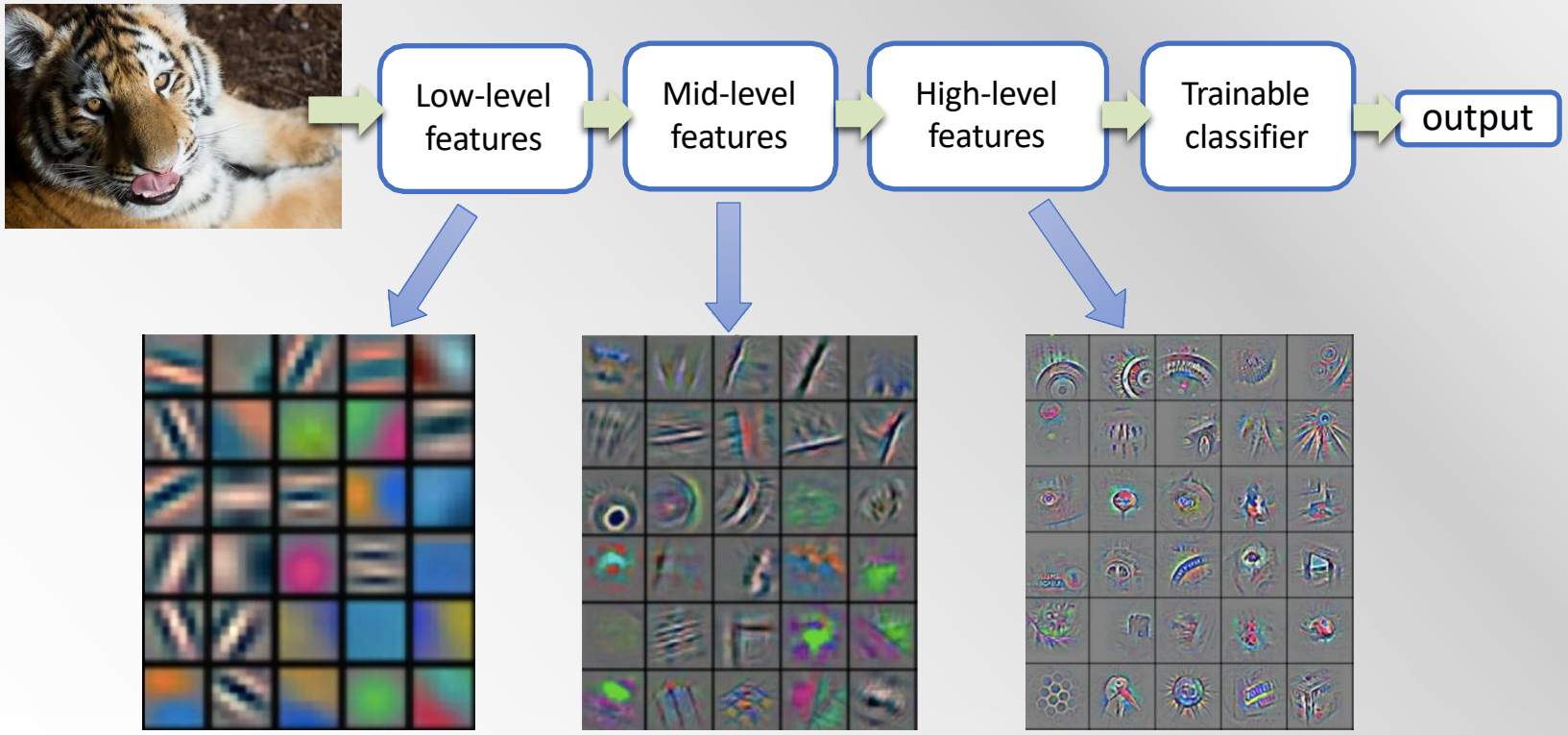
- Traditional pattern recognition models use hand-crafted features and relatively simple trainable classifier.




- This approach has the following limitations:
- It is very tedious and costly to develop hand-crafted features
- The hand-crafted features are usually highly dependents on one application, and cannot be transferred easily to other applications

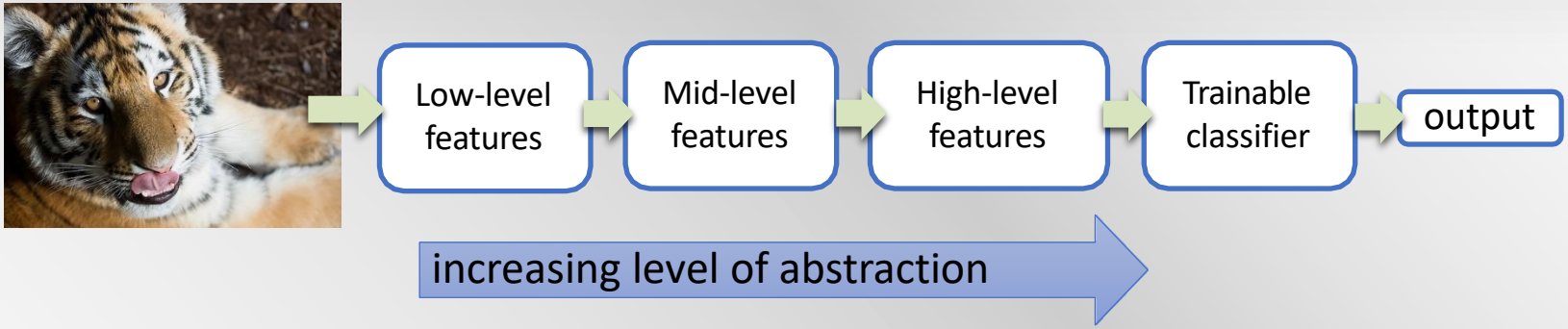
# Deep Learning

- Deep learning (representation learning) seeks to learn rich hierarchical representations (i.e. features) automatically through multiple stage of feature learning process.




Feature visualization of convolutional net trained on ImageNet (Zeiler and Fergus, 2013)

# Learning Hierarchical Representations

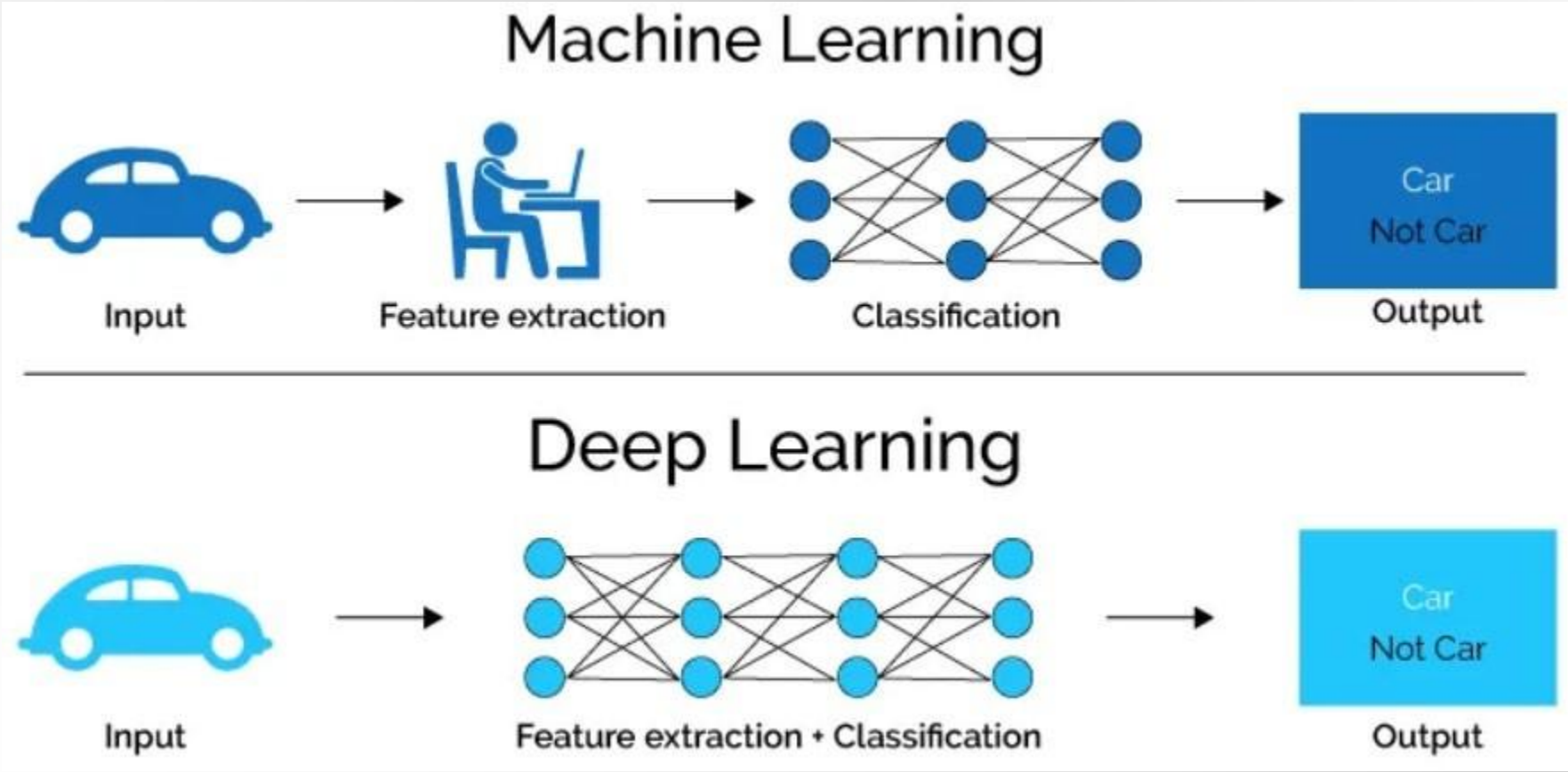



- Hierarchy of representations with increasing level of abstraction. Each stage is a kind of trainable nonlinear feature transform
- Image recognition

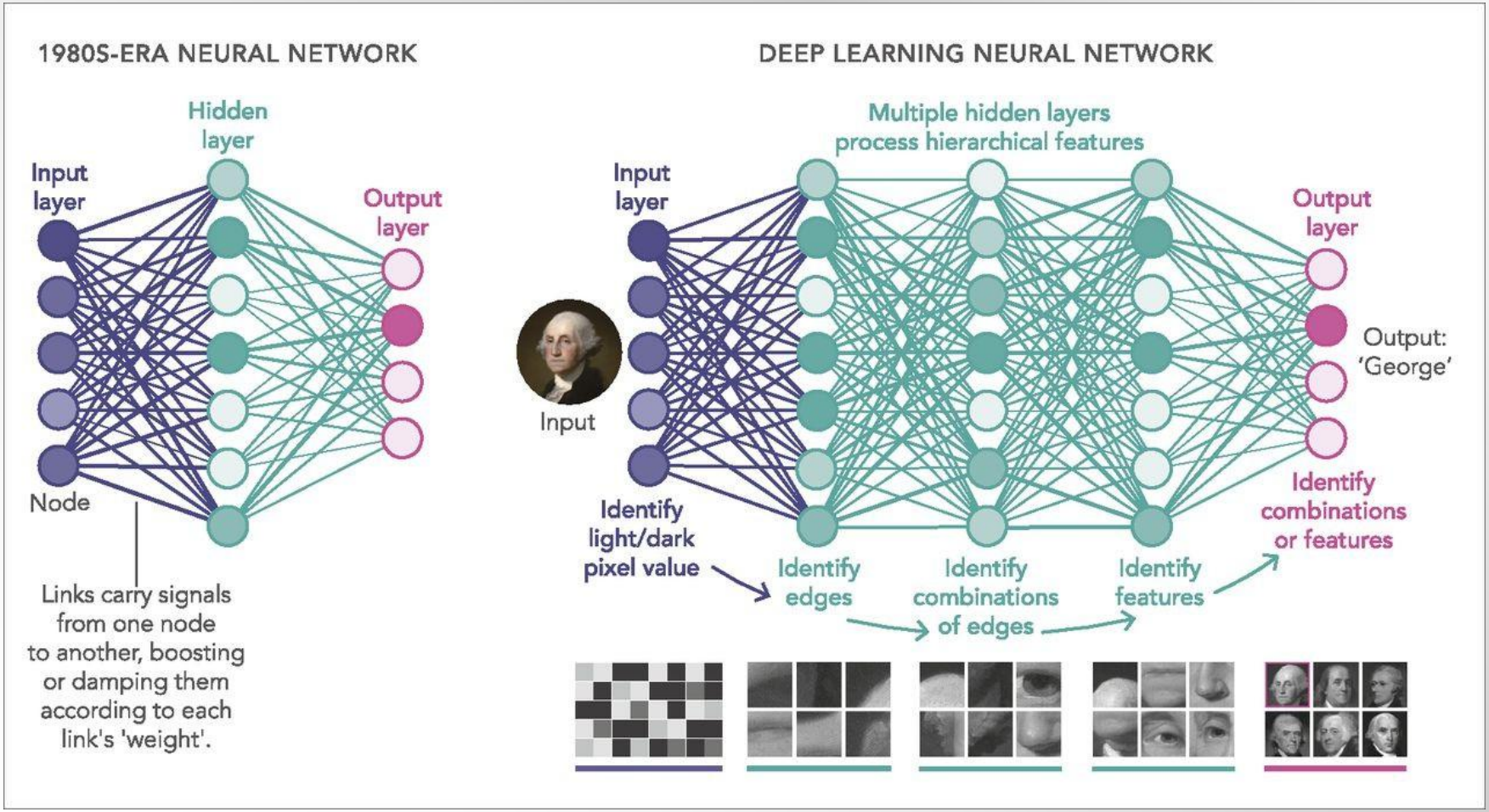
  Pixel → edge → texton → motif → part → object
- Text

  Character → word → word group → clause → sentence → semantics → story

# Machine Learning vs Deep Learning



# Classical ANN and Deep NN



# Convolution Neural Network (CNN)

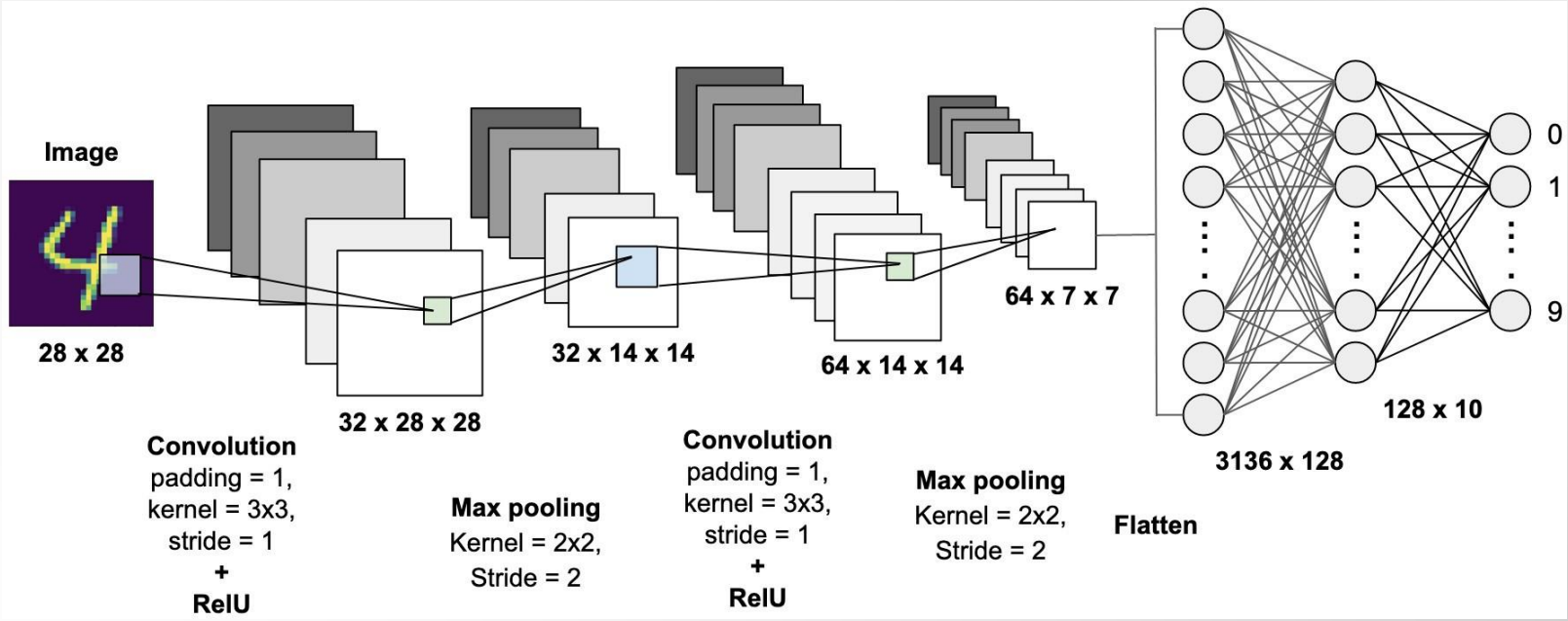
# Convolution Neural Network (CNN)

- Convolutional Neural Networks are a powerful artificial neural network technique.
- These networks preserve the spatial structure of the problem and were developed for object recognition tasks such as handwritten digit recognition.
- They are popular because people are achieving state-of-the-art results on difficult computer vision and natural language processing tasks

# Convolution Neural Network (CNN)

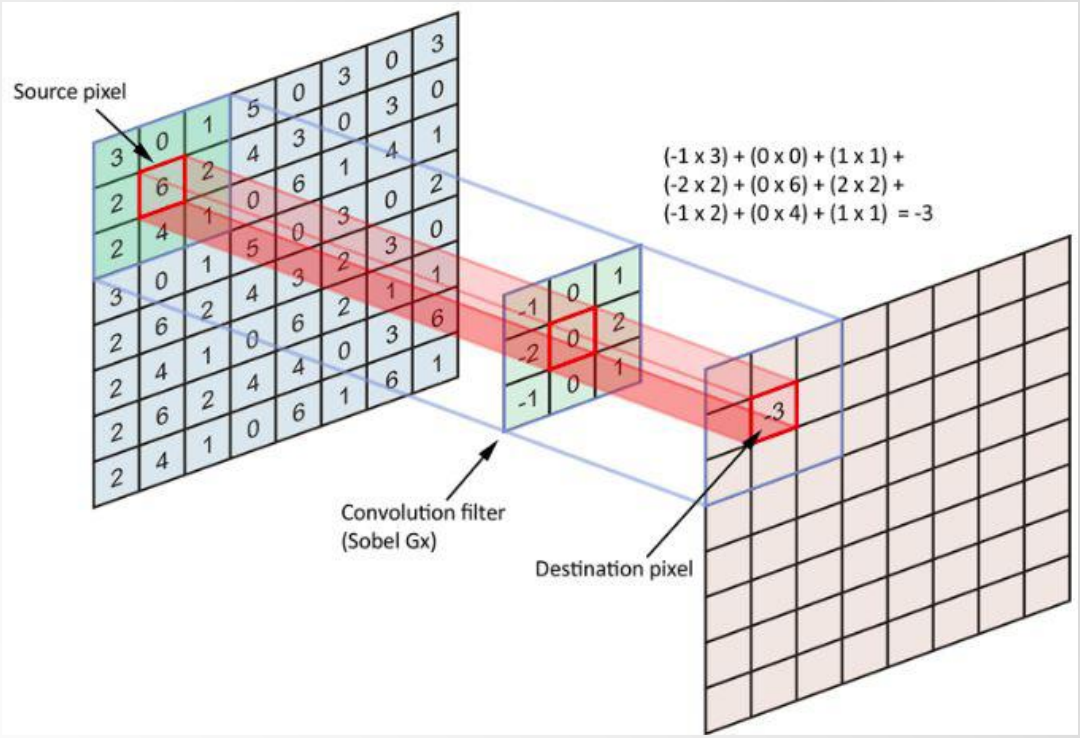
- There are three types of layers in a Convolutional Neural Network:
  - Convolutional Layers.
  - Pooling Layers.
  - Fully-Connected Layers.

# CNN Architecture



# Convolutional Layers

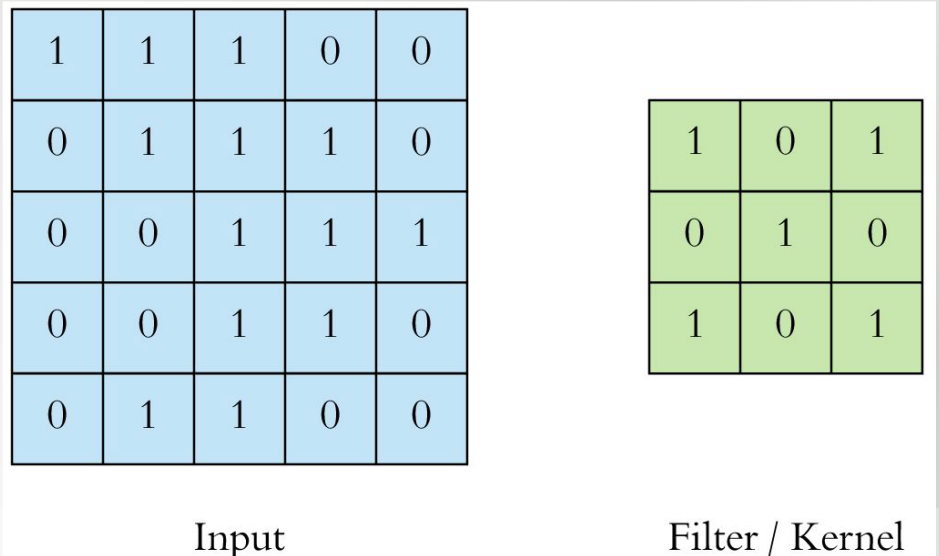
- Convolutional layers contain Filters/Kernels which extract feature maps from the previous layer



# Convolution Layer

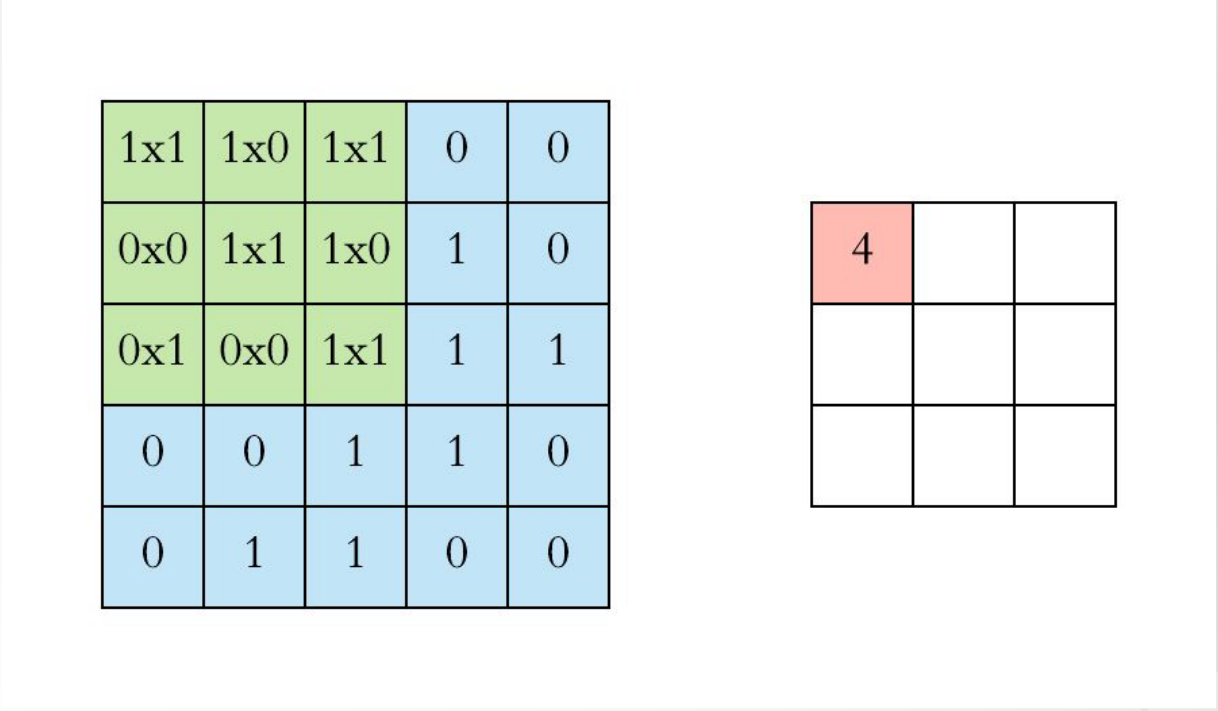
- Image Dimensions = 5 (Height) x 5 (Breadth) x 1 (Number of channels, e.g. RGB)
  - The blue matrix is our 5x5x1 input image.
- The element involved in carrying out the convolution operation of a Convolutional Layer is called the Kernel/Filter, K, represented in the green, which is a 3x3x1 matrix.
- Kernel/Filter K

| | | |
|---|---|---|
| 1 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 1 |



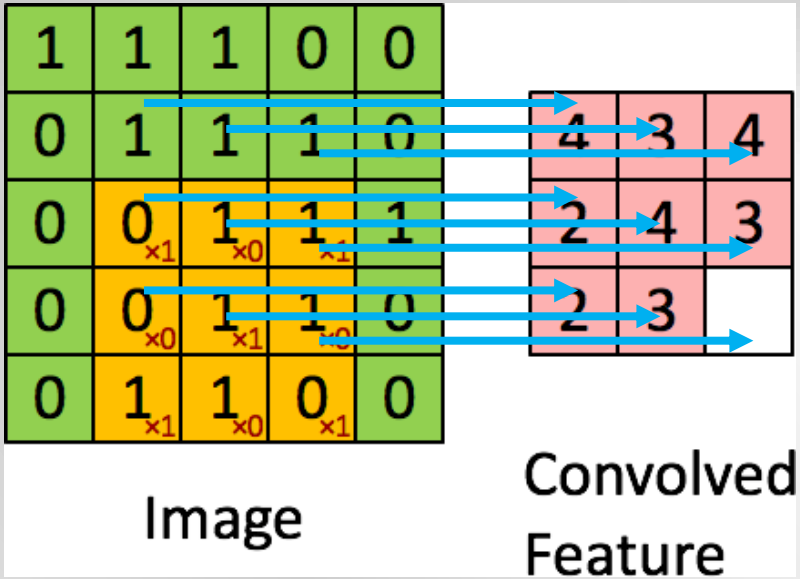
# Convolution Kernel

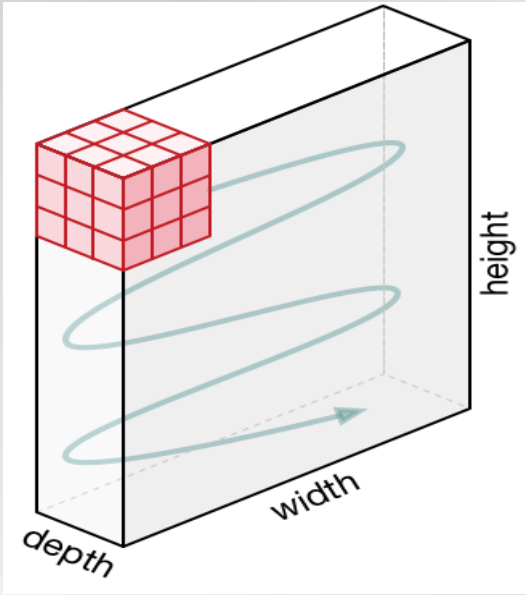
- A convolution is to multiply two matrices with the size of kernel
- Overlay the kernel  (3x3x1), on the image, and multiple, result in a single value as output



# Movement of the Kernel

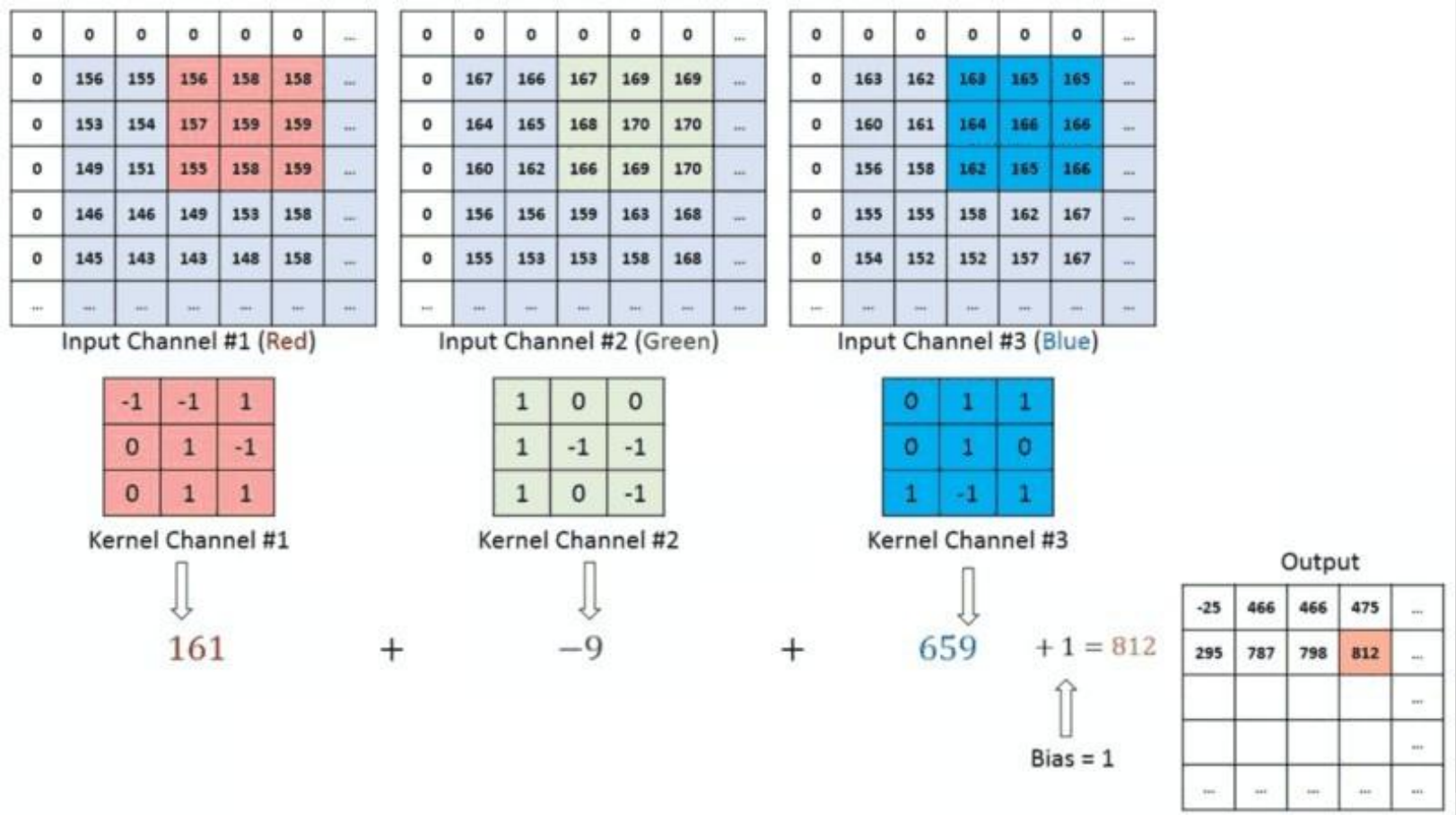
- The Kernel shifts 9 times because of Stride Length = 1 (Non-Strided)
- Every time performing a matrix multiplication operation between K and the portion P of the image over which the kernel is hovering.
- It repeats the process until the entire image is traversed.



# Movement of the Kernel

- For color image with 3 channels, to obtain a single channel feature map



# Convolutional on Image

- Multiple kernels are learned to apply on the same input
- Multiple feature maps

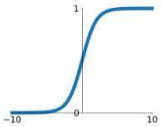
Input

Feature Maps

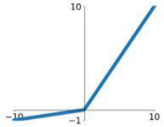
# Activation Functions

- Selected activation function will rectified the feature map

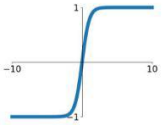
**Sigmoid**
$\sigma(x) = \frac{1}{1+e^{-x}}$



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

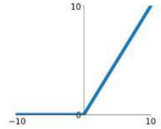


**tanh**
$\tanh(x)$

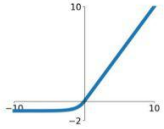


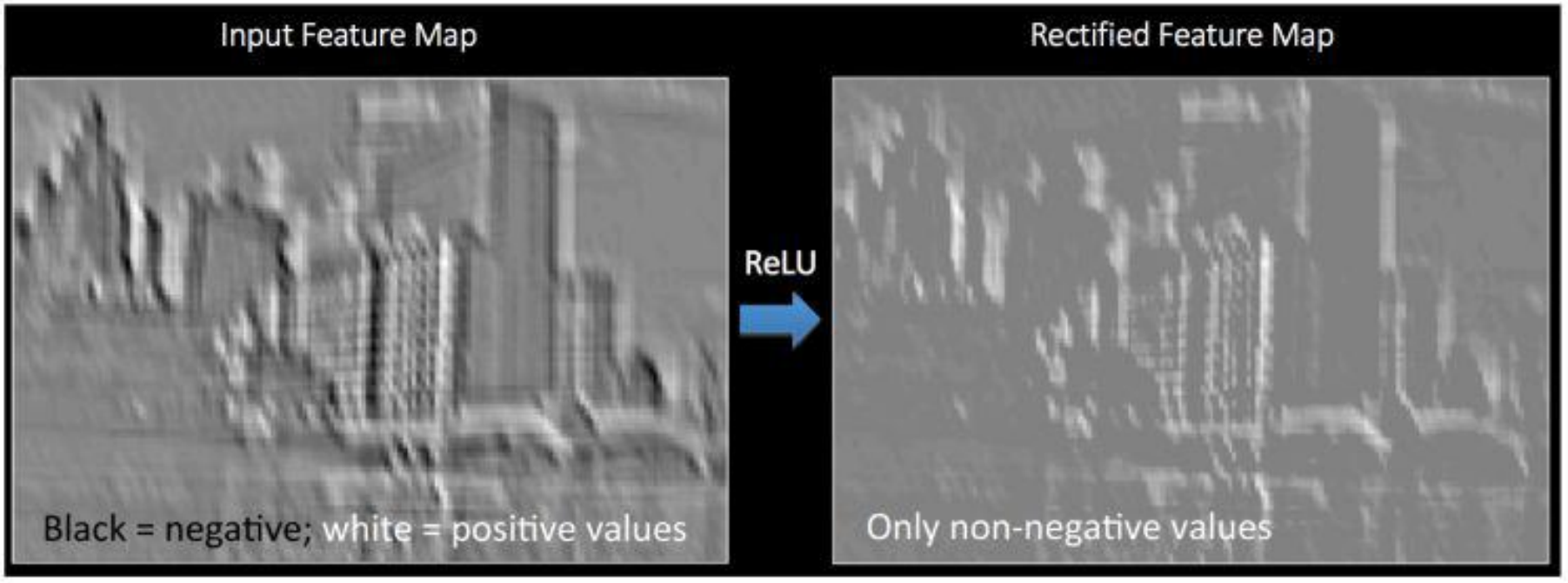
**Maxout**
$\max(w_1^T x + b_1, w_2^T x + b_2)$

**ReLU**
$\max(0, x)$



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



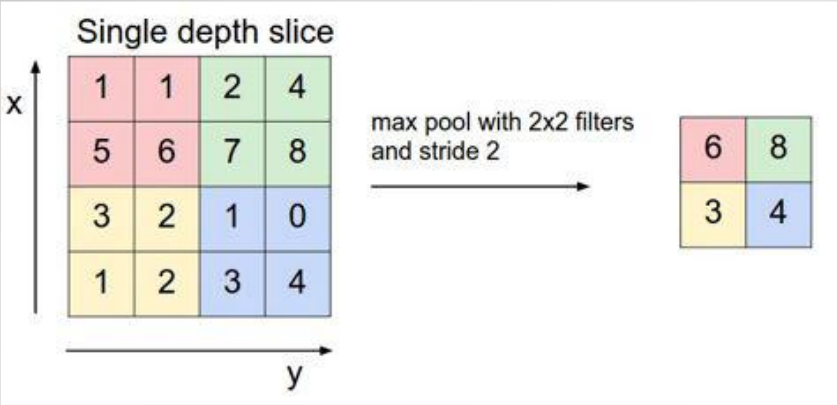


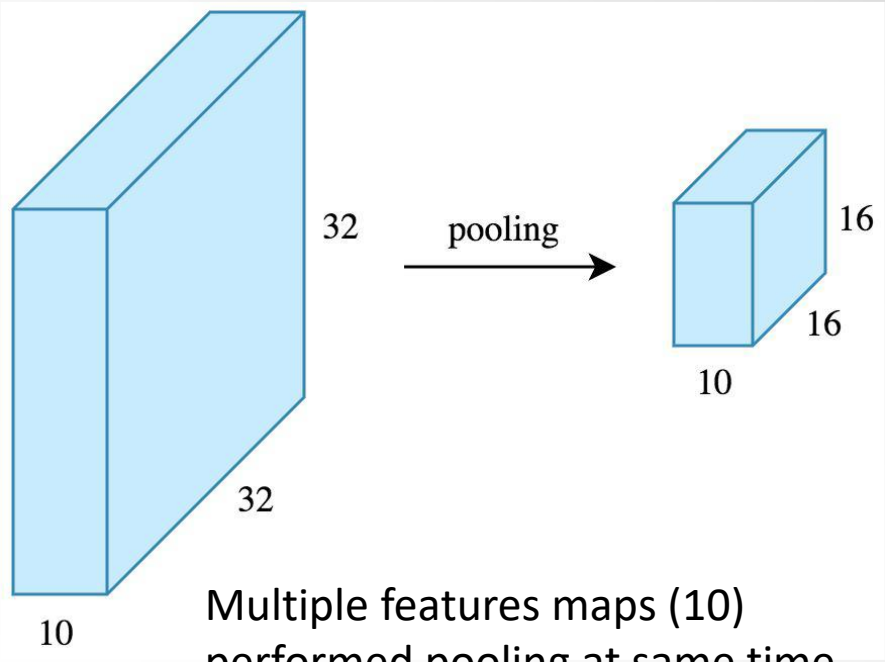
# Pooling Layers

- The pooling layers down-sample the previous layers feature map.
- Pooling layers follow a sequence of one or more convolutional layers and are intended to consolidate the features learned and expressed in the previous layers feature map.
- As such, pooling may be consider a technique to compress or generalize feature representations and generally reduce the overfitting of the training data by the model.

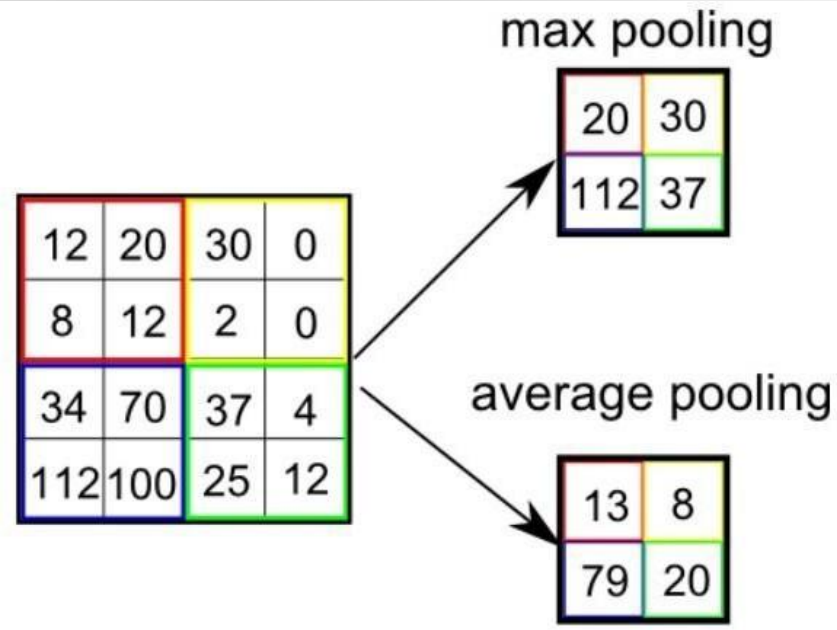
# Pooling Layer

- Max pooling
- Average pooling
- Sum pooling

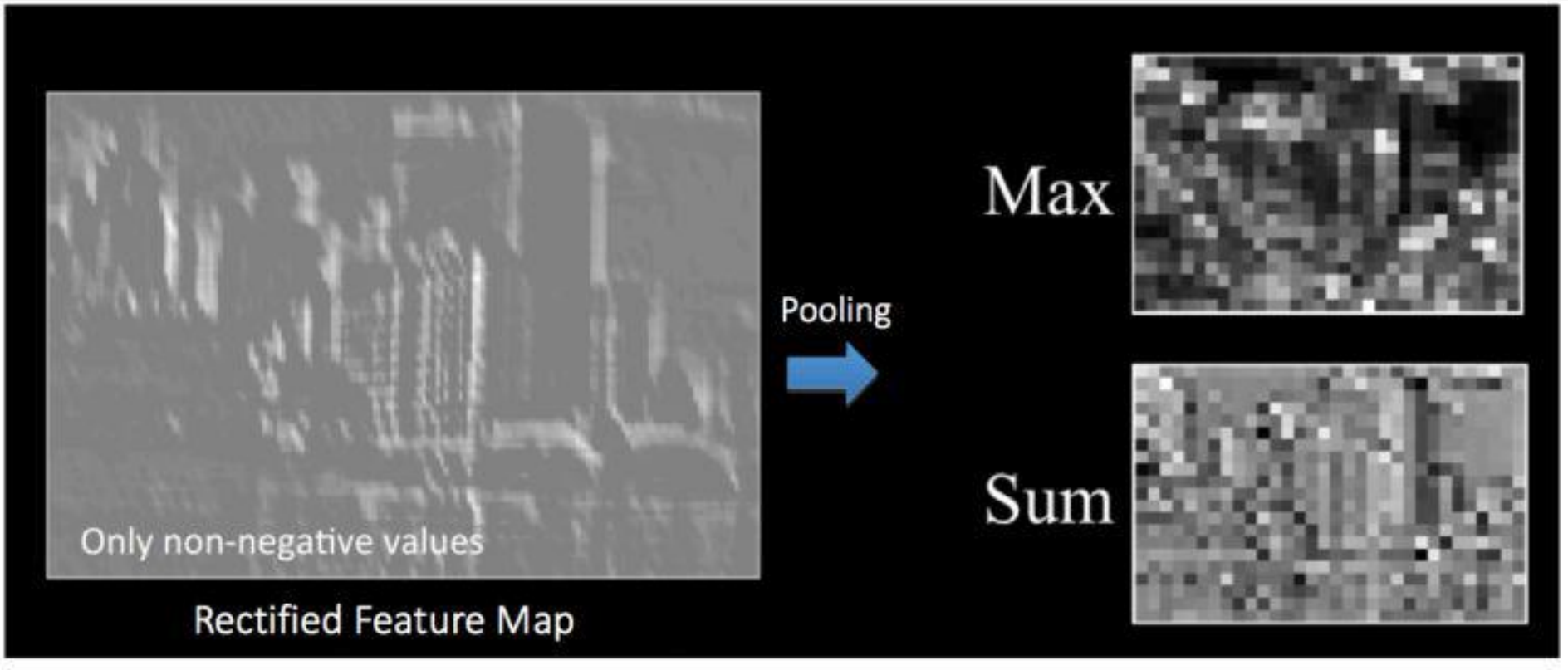



Multiple features maps (10) performed pooling at same time



# Pooling - Max-Pooling & Sum-Pooling

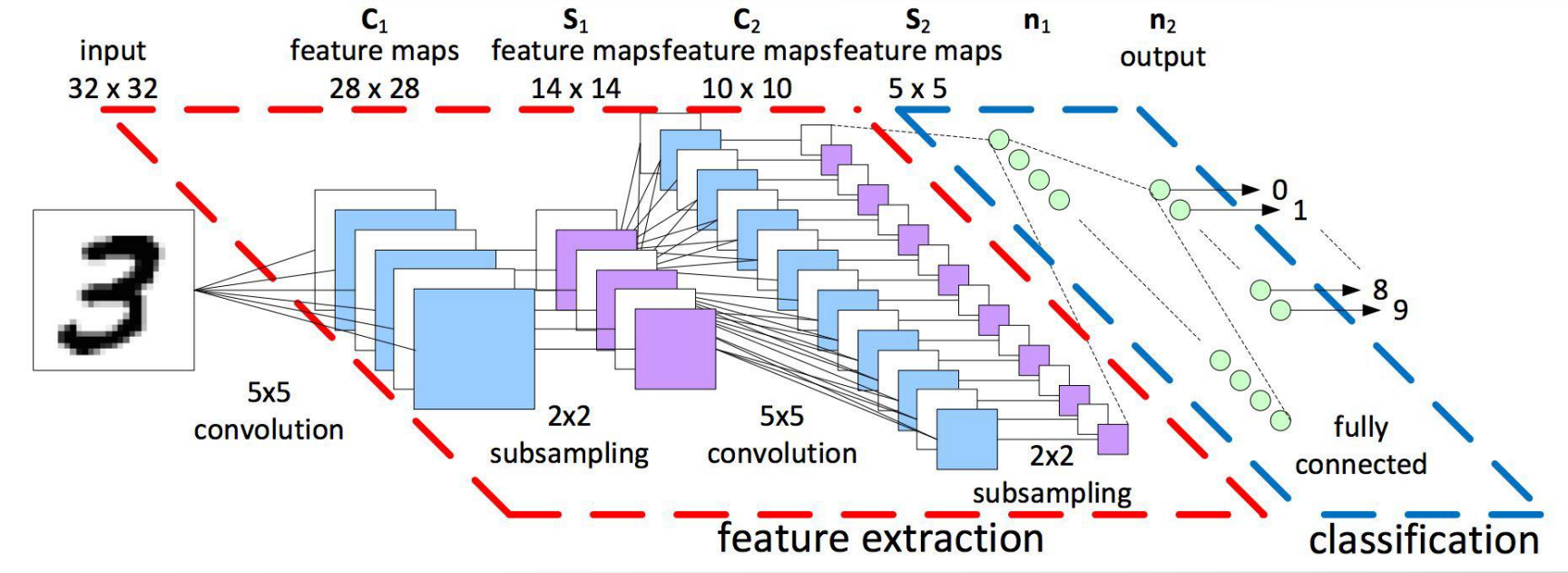


# Fully Connected Layers

- Fully connected layers are the normal flat feedforward neural network layer.
- These layers may have a nonlinear activation function or a softmax activation in order to output probabilities of class predictions.
- Fully connected layers are used at the end of the network after feature extraction and consolidation has been performed by the convolutional and pooling layers.
- They are used to create final nonlinear combinations of features and for making predictions by the network.

# MNIST digit recognition With LeNet

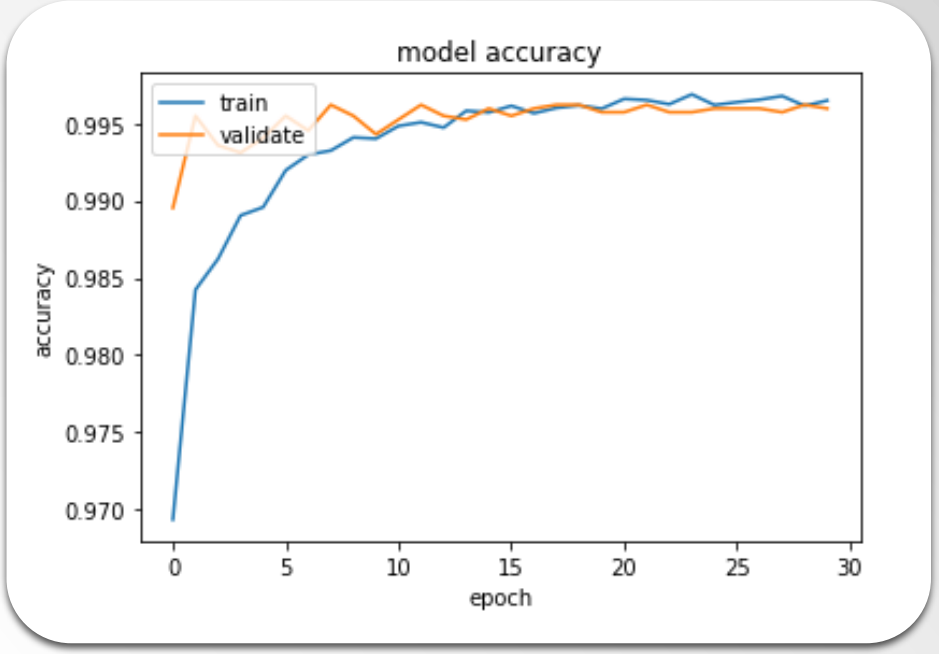
- LeNet layer structure
  - Early layers learned the features
  - Later layers responsible for classification

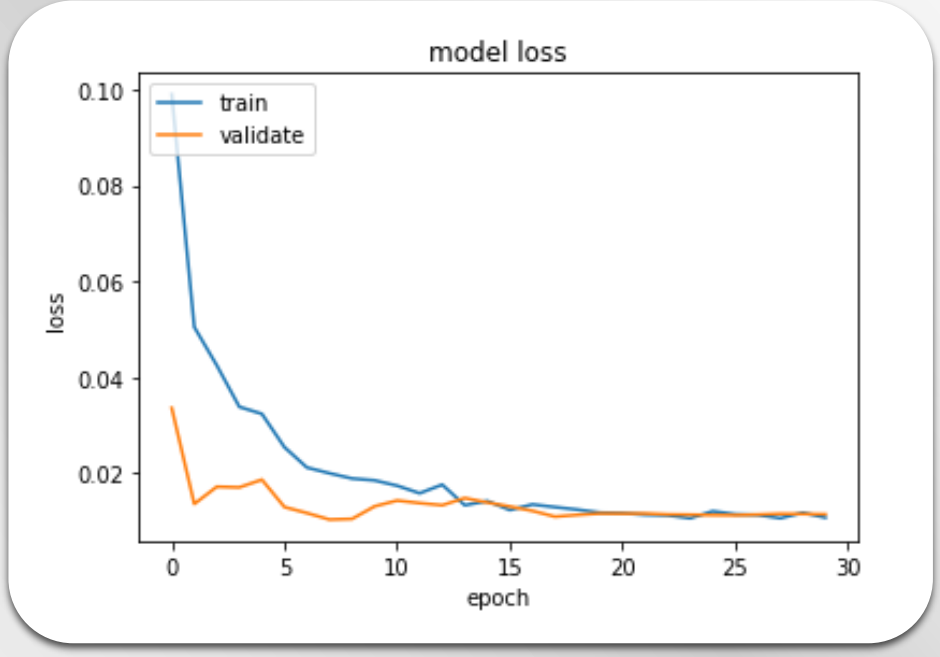



https://www.kaggle.com/code/roblexnana/mnist-digit-recognition-with-lenet-on-keras

# MNIST digit recognition With LeNet

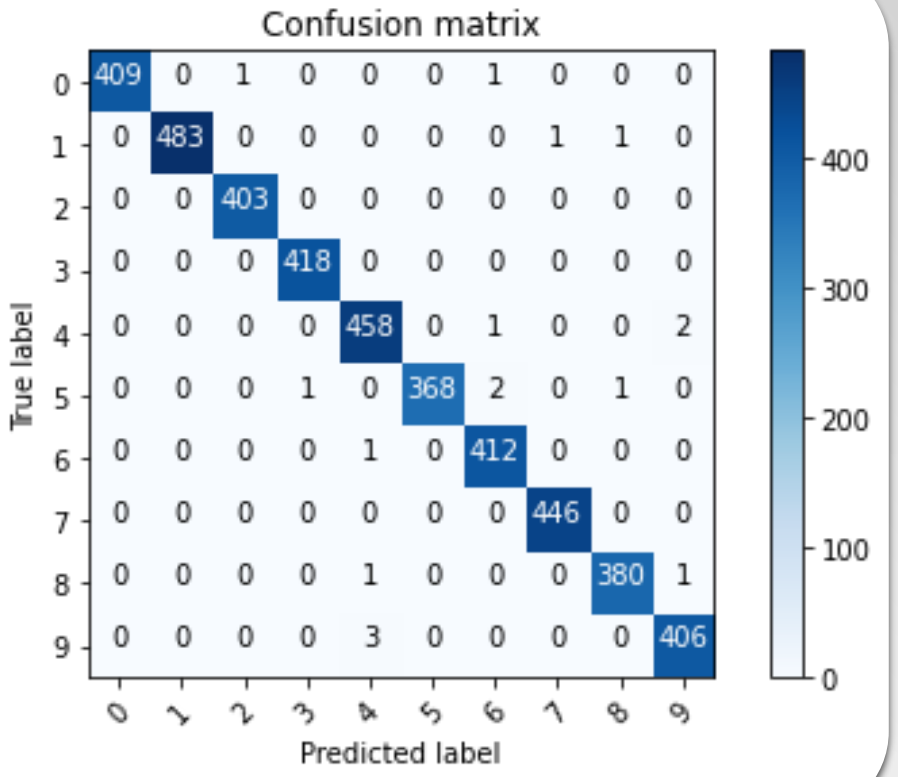
- Training Results

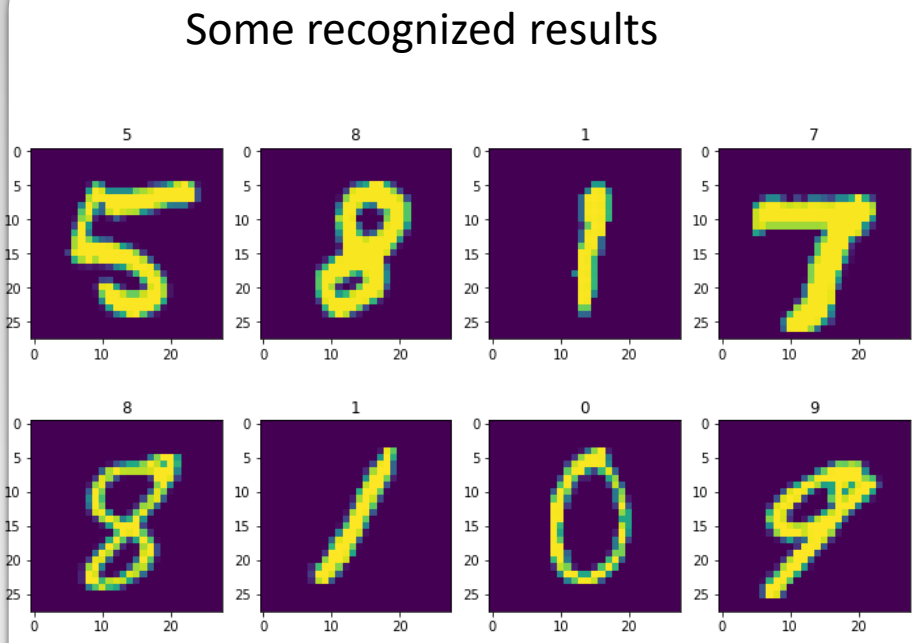


# MNIST digit recognition With LeNet

- Performance



# Hyper-parameters

- hyperparameters are parameters that are set before the learning process begins and are not learned from the data.
- They control the learning/training process (the process of finding the best weights) itself.
- The optimal values for hyperparameters are usually found through experimentation or techniques like grid or random search.

https://medium.com/the-research-nest/explained-hyperparameters-in-deep-learning-9b1e0f3b9029

# Common Hyperparameters in Deep Learning

- Optimizer
  - Algorithms like Adam, SGD (Stochastic Gradient Descent),
- Learning rate
- Batch size
- Number of training epochs
- Dropout Rate
- Regularization coefficient
- Number of hidden units
- Activation Functions

# Optimizer: Stochastic gradient descent

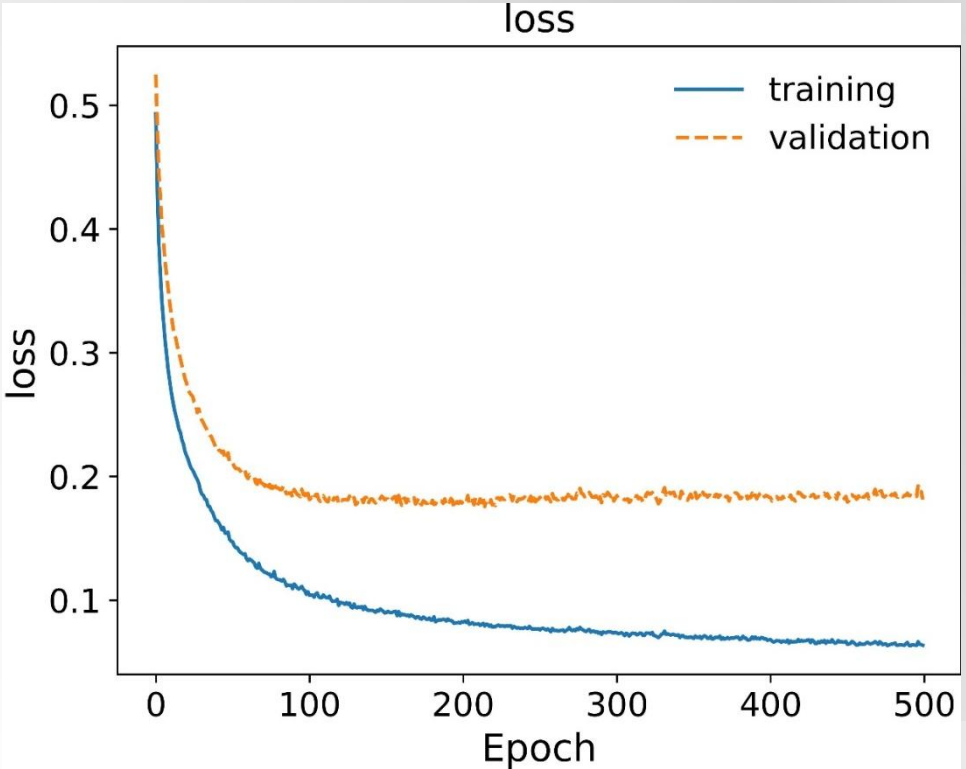
- Given one training sample $(x^{(i)}, y^{(i)})$
- Compute the output of the neural network $f_{\vec{W}}(x^{(i)})$
- Training objective: minimize the prediction error (loss) – there are different ways to define error. The following is an example:

$$E = \frac{1}{2}(y^{(i)} - f_{\vec{W}}(x^{(i)}))^2$$

- Estimate how much each weight $w_k$ in $\vec{W}$ contributes to the error: $\frac{\partial E}{\partial w_k}$
- Update the weight $w_k$ by $w_k = w_k - \alpha \frac{\partial E}{\partial w_k}$. Here $\alpha$ is the learning rate.

# Loss function

- measures how different the model's actual output is from the desired output.
- Loss vs Epoch Graph
  - Whether the training converge
  - Check if Overfitting



# Batch Size

- The number of training examples used in each iteration of the optimization.
- The choice of the batch size can significantly impact the performance of the optimization algorithm.
  - Typically, it's between 1 to a few hundred.
- The higher the batch size, the more memory space you'll need.

# Epoch and Iteration

- An epoch is when all the training data is used once
  - The total number of iterations of all the training data in one cycle for training the machine learning model.

- Iteration: The total number of batches required to complete one Epoch is called an iteration.

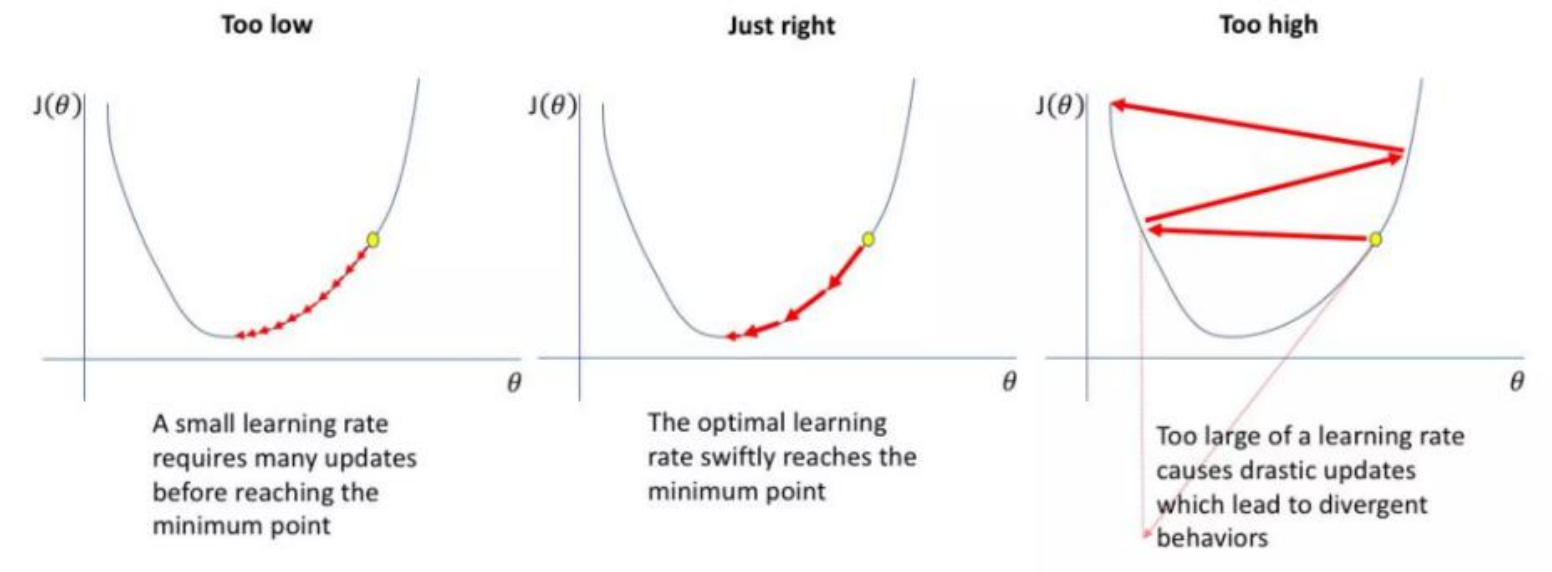
# Example on Batch size, Epoch,and Iteration

- Say you have a dataset of 10 examples (or samples). You have a batch size of 2, and you've specified you want the algorithm to run for 3 epochs.

- Therefore, in each epoch, you have 5 batches (10/2 = 5). Each batch gets passed through the algorithm, therefore you have 5 iterations per epoch. Since you've specified 3 epochs, you have a total of 15 iterations (5*3 = 15) for training.

# Learning Rate

- The size of each step during optimization
- Choosing the right learning rate is important for efficient and effective training.
- Large learning rates
  - Reduce in time
  - may cause divergence

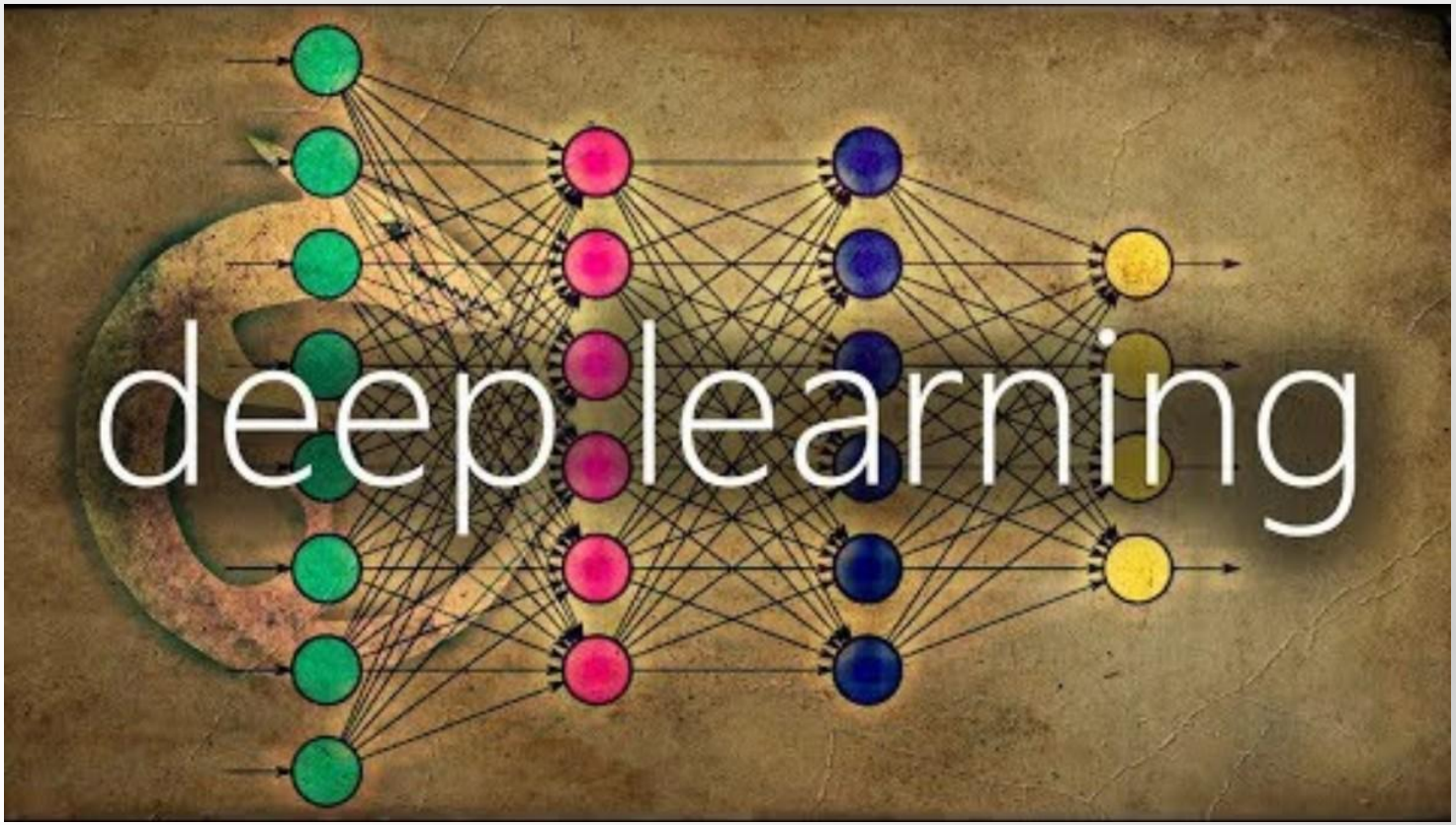



https://www.baeldung.com/cs/learning-rate-batch-size

What is Backpropagation? | IBM

# Regularization coefficient

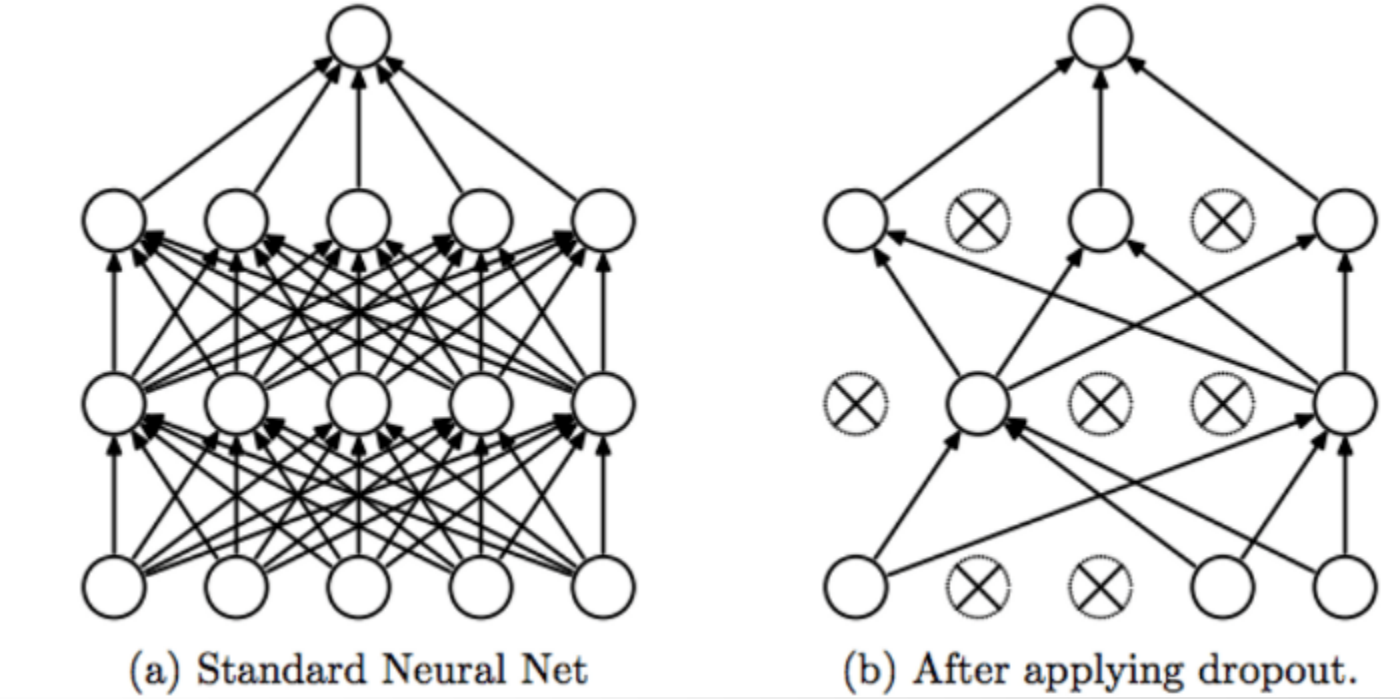
- hyperparameter used in L1 or L2 regularization to control the complexity of the model




https://www.youtube.com/watch?v=iuJgyiS7BKM

# Dropout

- Another regularization technique where randomly selected neurons are ignored during training to prevent overfitting.



# Summary

- Basics of Deep Learning
- Convolutional Neural Networks (CNN)
- Training and Hyper-parameters