AI Engineer Training: V
In the Era of Deep Learning

IT21 Learning
Alvin Jin

Weekly AI News

- Canadian government invests $25M in AI based Health research projects

- European researchers@CLAIRE call for EU-wide AI coordination

- Spark + AI Summit: AI might drive a complete architecture change — Software 2.0

Agenda

- Deep Learning in Computer Visions:
  - Convolutional Neural Networks

- Case Studies:
  - Handwritten Digits Recognition: LeNet

Images

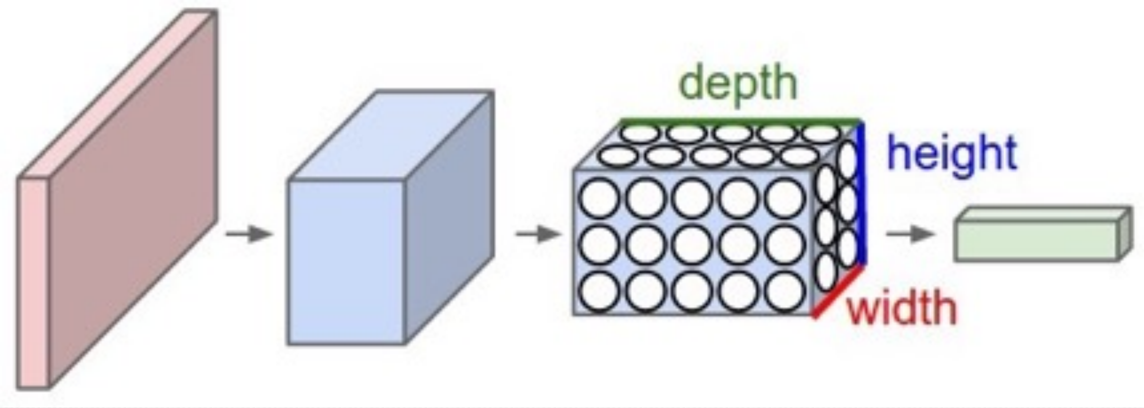
- An image is a multidimensional matrix of pixels.
- A pixel is considered the color/intensity of light appears in a given place in our image between 0-255
- Images have a depth – the number of channels.
  - Grayscale has a depth of 1
  - RGB has a depth of 3
- The layers of a CNN are arranged in 3D volume: width, height and depth.

Convolutional Neural Network

- CNNs operate convolutional, extracting features from local input, allowing for representation modularity and data efficiency

- CNNs chain up convolutional and pooling layers to help downsample the input samples, and don't use FC layers until the very last layers to obtain the final output classification.

fully connected

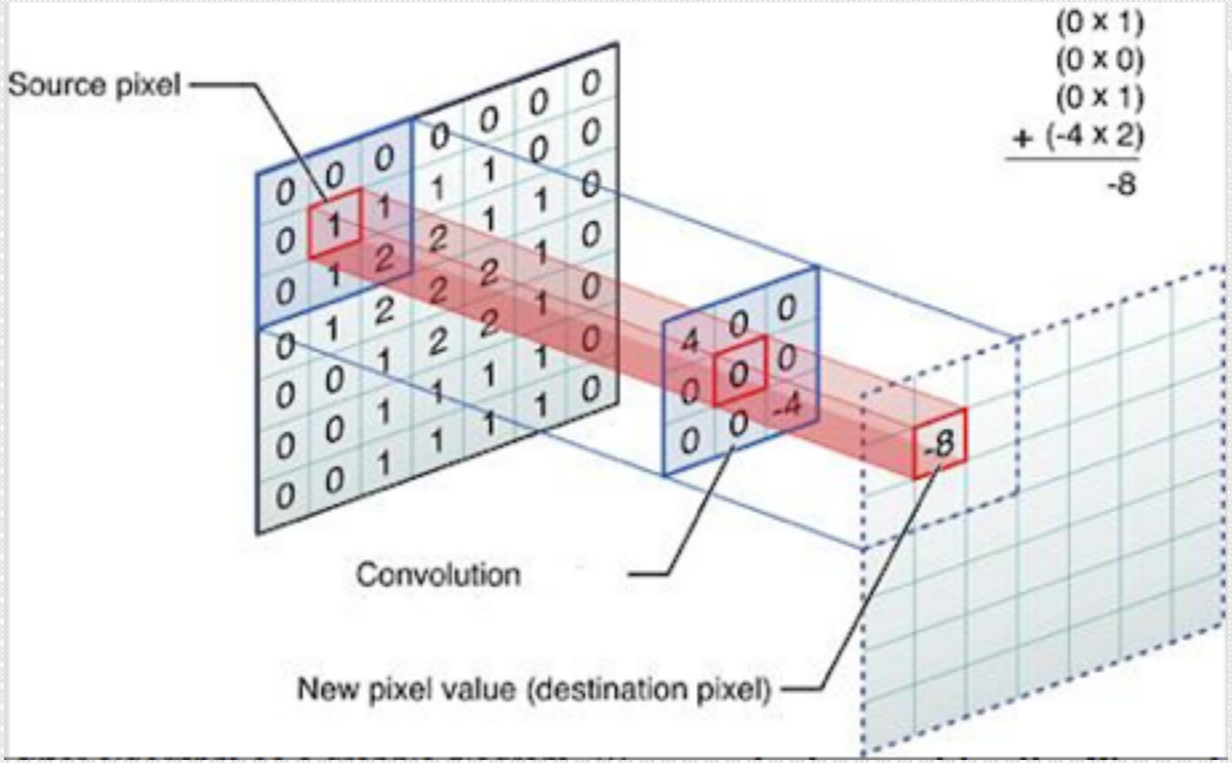


What's Convolution?

- Convolutions are fundamental building-blocks in image processing.
- A convolution is an element-wise matrix multiplication between a filter, and the area that filter covers of the input image.
- The neurons in a convolution layer are only connected to a small region of the layer before it, instead of all of the neurons in a fully-connected manner.

Why Convolution?

- The source pixel is replaced with a weighted sum of itself and nearby pixels.



Filters

- Filter/kernel is used for applying process functions to detect features or patterns.
- Each conventional layer applies a different set of filters. During training, a CNN automatically learns the values for these filters, which are initialized randomly.
- Filter is a tiny matrix that slides across, from left-to-right and top-to-bottom, of a larger image.
- Most filters are square matrices, use an odd kernel size to ensure a valid integer coordinate at the center of the image

Filter Depth

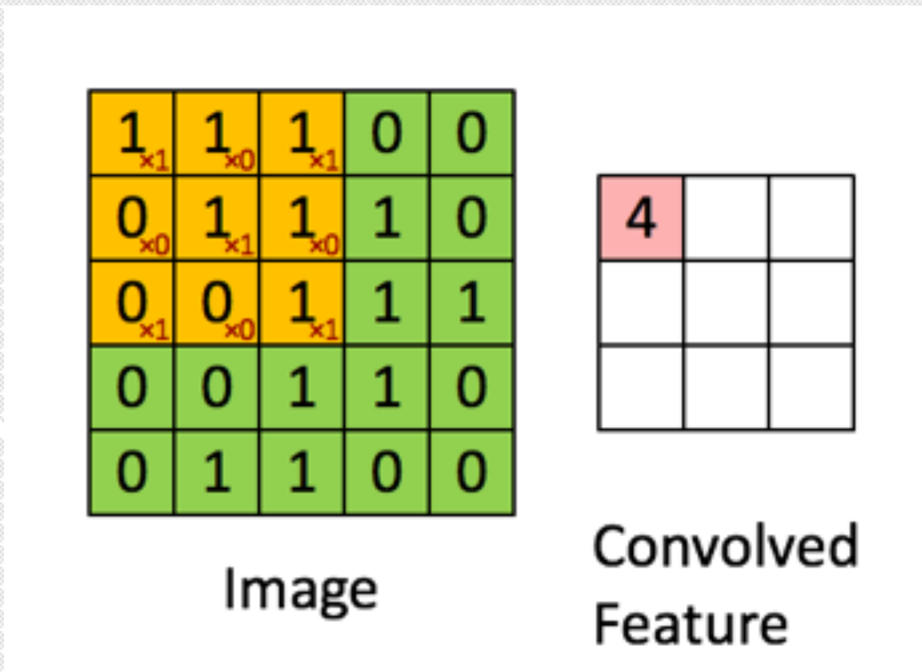
- For image inputs to CNNs, depth is the **number of channels**.
- For volumes deeper in CNNs, the depth is the **number of filters** applied in the previous layer.
- Each filter will produce a separate 2-dimensional **activation map**, which activates in the presence of features.

Sliding and Stride

- A small matrix slides from left-to-right and top-to-bottom across an image, and applying a convolution at each coordinate of the image.
- Smaller strides(1 or 2) will lead to overlapping receptive fields and larger output volumes.
- Conversely, larger strides will result in less overlapping receptive fields and smaller output volumes.

Convolution Demo

- For each activation map, each neuron connects to only a small region of the input volume, and shares the same connection weights(filter).

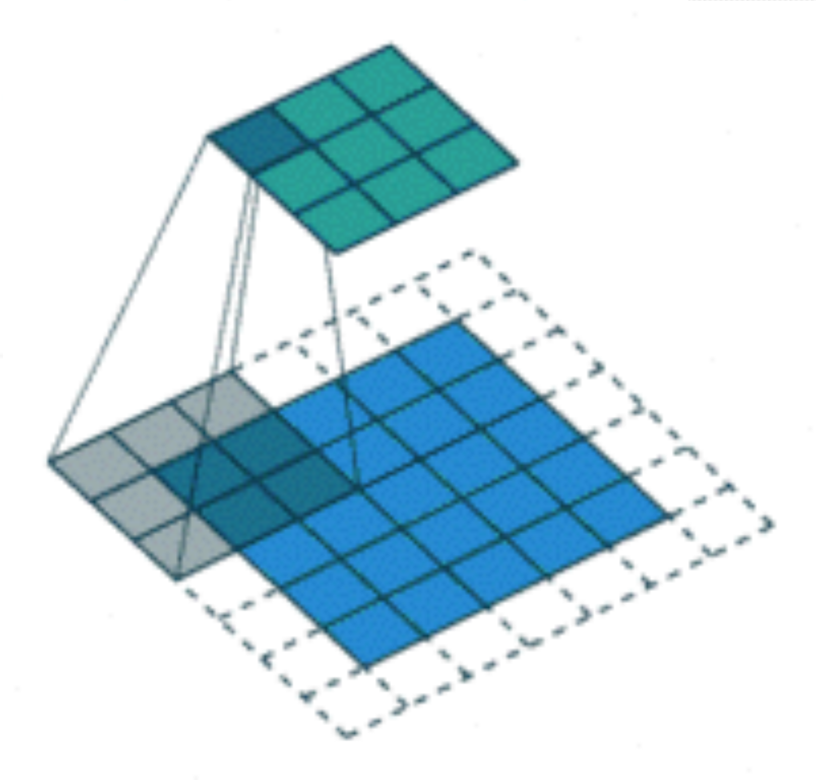


Zero Padding

- We want to preserve as much information about the original input volume so that we can extract those low level features.
- Zero padding pads the input volume with zeros around the border.
- Keep output volume with the same spatial dimensions as the input volume:

$$\textit{Zero Padding} = \frac{(K-1)}{2}$$

Stride and Padding Demo



Control Output Volume Size

- Stack activation maps along the depth dimension and produce the output volume.

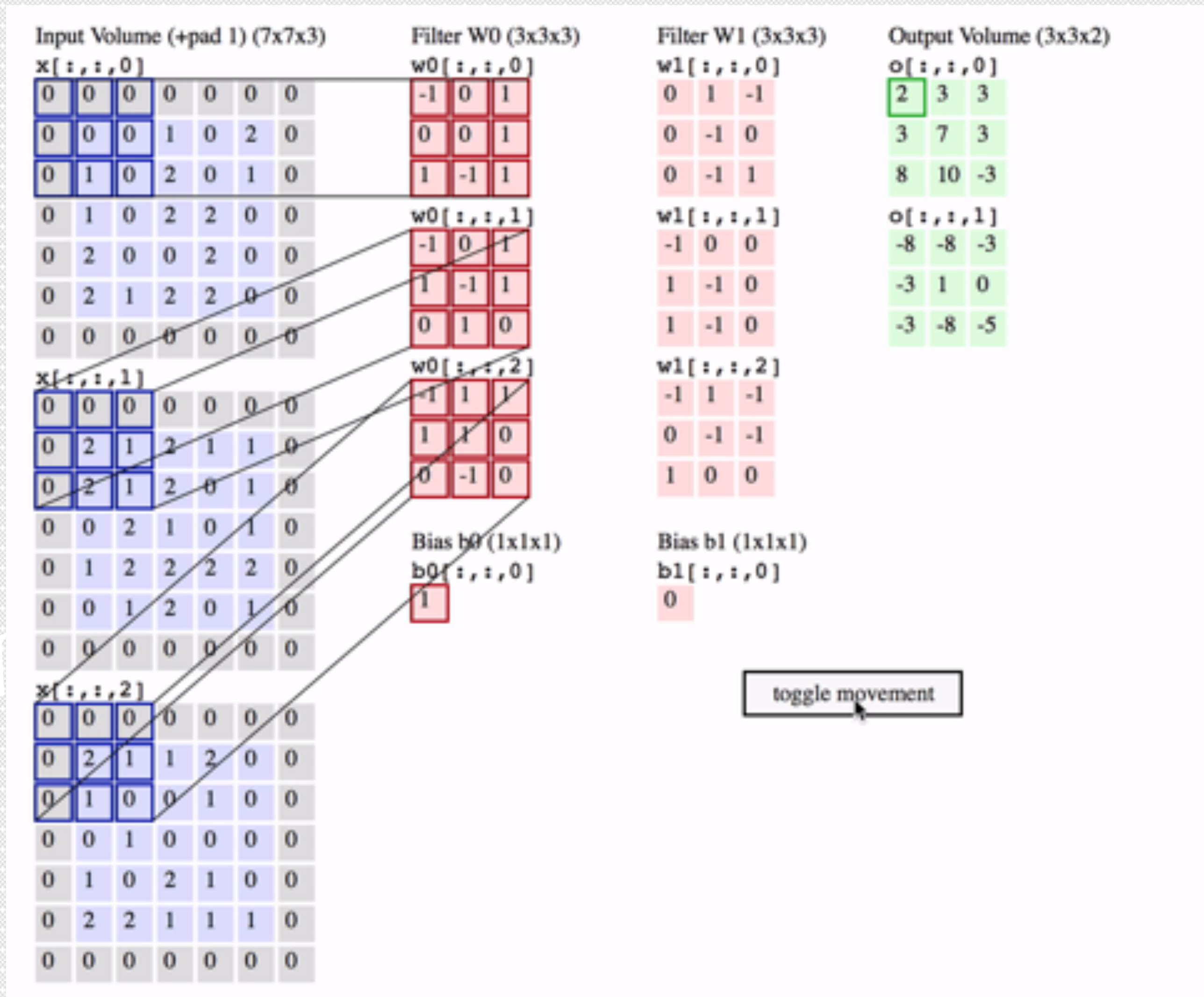
- CONV layers can be used to reduce the spatial dimensions of the input volumes by changing the stride of the filters.

$$O = \frac{(W - K + 2P)}{S} + 1$$

Convolution Demo

W=5, K=3, S=2, P=1

(5 - 3 + 2)/2 + 1 = 3

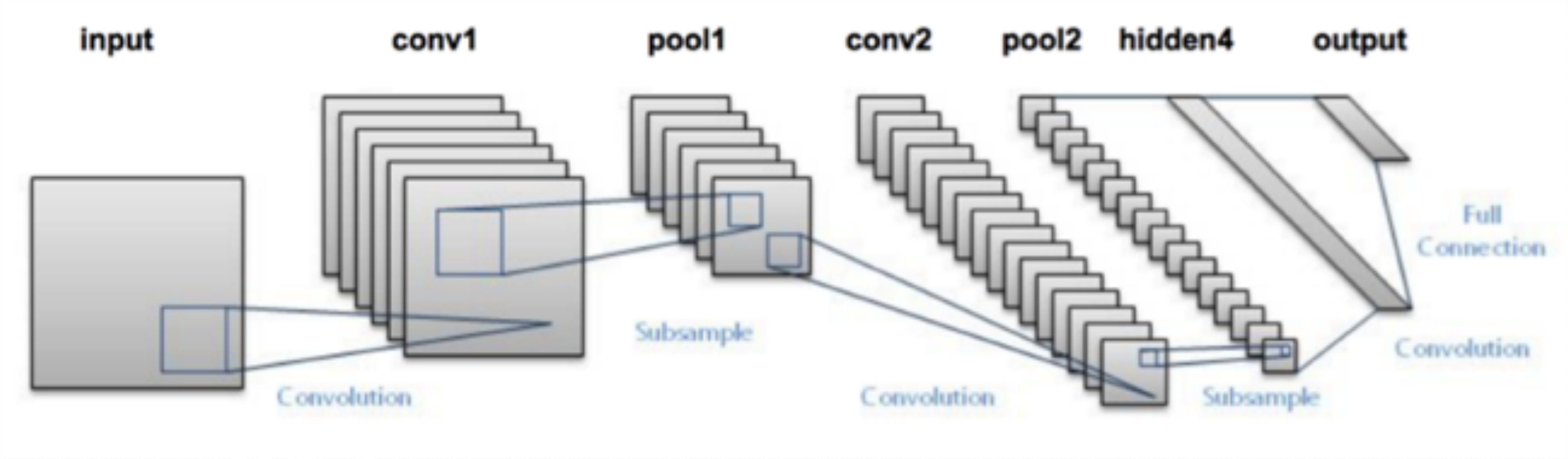


Layer Types

- Fully-Connected Layer - FC
- Convolutional Layer - CONV
- Pooling Layer - POOL
- Dropout Layer - DO
- Batch normalization - BN

Case Studies

- LeNet-5 architecture(1998) designed for MNIST



Convolutional Layers

- The CONV layer parameters consist of a set of K learnable filters, where each filter has a width and a height, and are always square.
- These filters are small but extend throughout the full depth of the volume.
- The network learns filters that activate, when they see a specific type of feature at a given spatial location in the input volume.

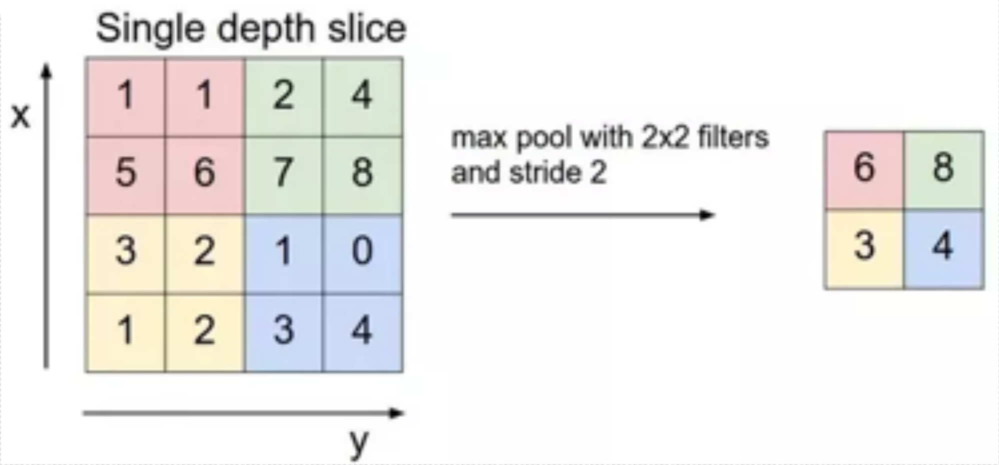
Why not FC in CNN?

- Connecting neurons in the current layer to all neurons in the previous layer generates too many weights, making it impossible to train deep networks on large spatial dimensions.

- Instead, CNNs choose to connect each neuron to only a local region of the previous, which is called the **receptive field** of the neuron. This local connectivity saves a huge amount of parameters in CNNs.

Pooling Layers

- Periodically insert a Pooling layer between successive Conv layers in a CNN.
- By representing each 2x2 block with one number, it allowed for translational invariance, the feature could be detected and lead to the same output.
- o = (w-k)/s+1



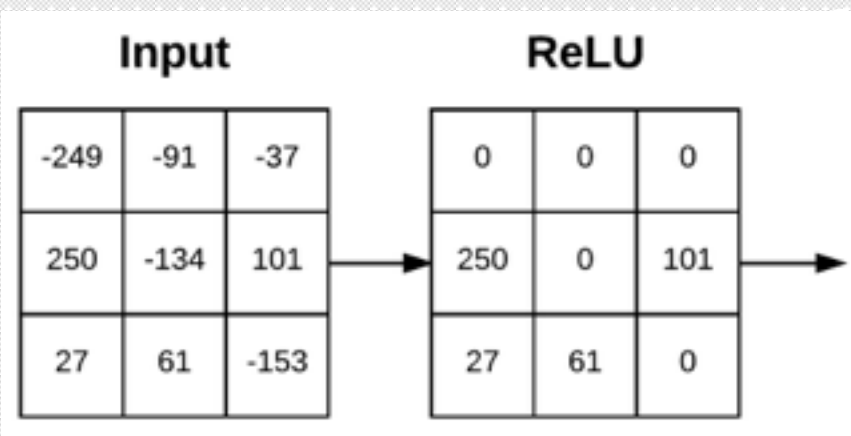
Pooling Layers

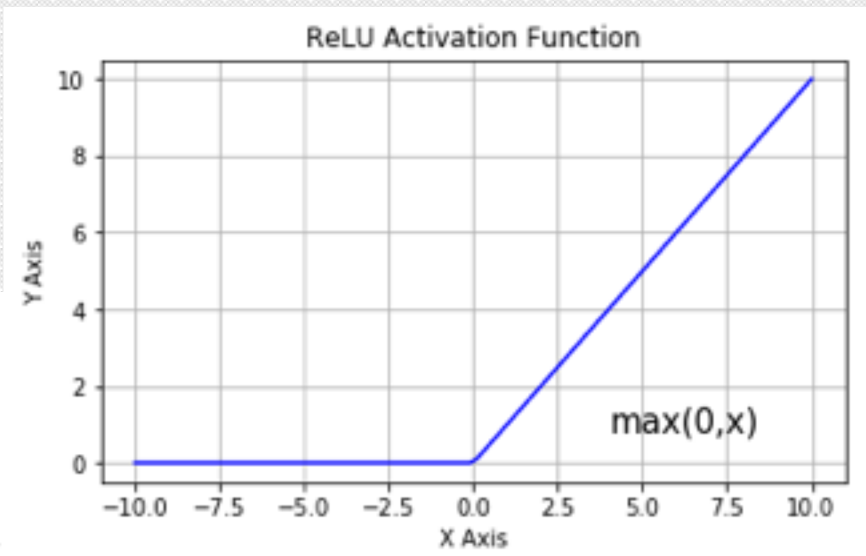
- The amount of parameters or weights is reduced by 75%, thus lessening the computation cost, and control overfitting.
- Max pooling is done in the middle of the network to reduce spatial size, and slowly strips off spatial relationship to create translational invariance.
- Average pooling is normally used as the final layer of the network to avoid using FC layers entirely.

ReLU Functions

- Rectified Linear Units(2010):
  - Zero out negative values. Widely used in CNN.

$$f(x) = \begin{cases} 0 & \text{for} \quad x < 0 \\ x & \text{for} \quad x \geq 0 \end{cases}$$

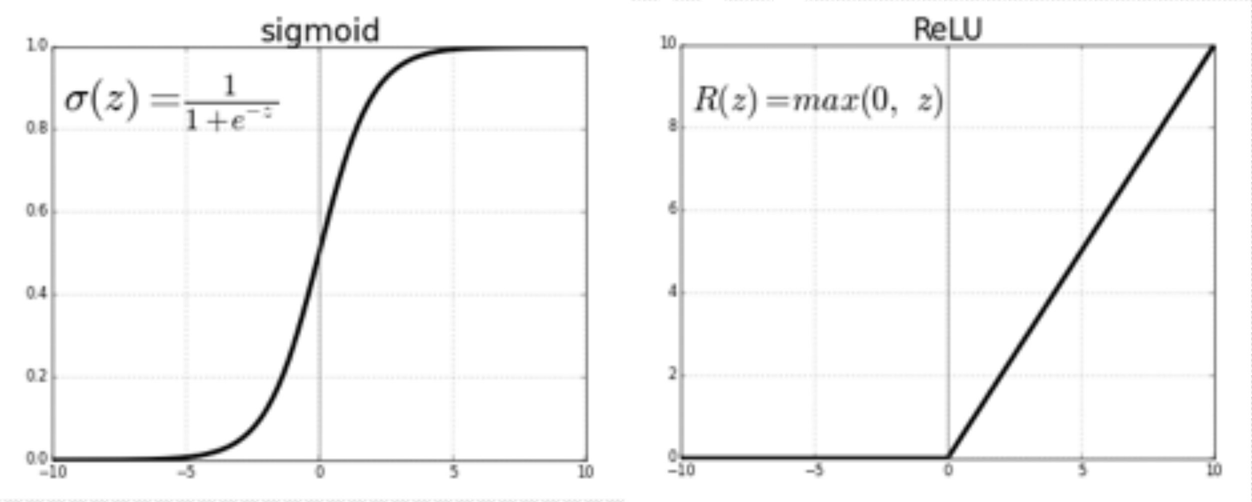




Sigmoid vs. ReLU

- Sigmoid squashes all values between 0 and 1, neuron outputs and gradients can vanish entirely.

- ReLU trains a lot faster due to computational efficiency, and alleviate the vanishing gradient problem



What does CNN learn?

- Detect edges from raw pixel data in the first layer.
- Use these edges to detect shapes in the second layer.
- Use these shapes to detect higher-level features in the highest layers of the network.
- The last layer in a CNN uses these higher-level features to make predictions regarding the contents of the image.

Q & A