

Reading 10

★ Convolutional Neural Networks

- Primarily used for image classification.
- Scenario: Computer has to figure out whether the first play on a tic-tac-toe board is an "X" or "O"
 - ↳ Each pixel is 0 or 1.
 - ↳ 0 for white
 - ↳ 1 for black
 - ↳ 6x6 picture (pixels)
- Convolutional NN do 3 things to make image classification practical:
 1. Reduce the # of input nodes.
 2. Tolerate small shifts in where the pixels are in the image
 3. Take advantage of the correlations that we observe in complex images.

• The first thing a Convolutional Neural Network does is apply a Filter to the Input Image. ↑ or "kernel"

↳ It's simply a smaller square that is commonly 3 pixels by 3 pixels

↳ The intensity of each pixel in the filter is determined by Backpropagation.

↳ So before training we start with random pixel values

↳ and after training we end up with something more useful.

↳ To apply the Filter to the input image we overlay the Filter onto the image...

↳ And then we multiply together each overlapping pixel

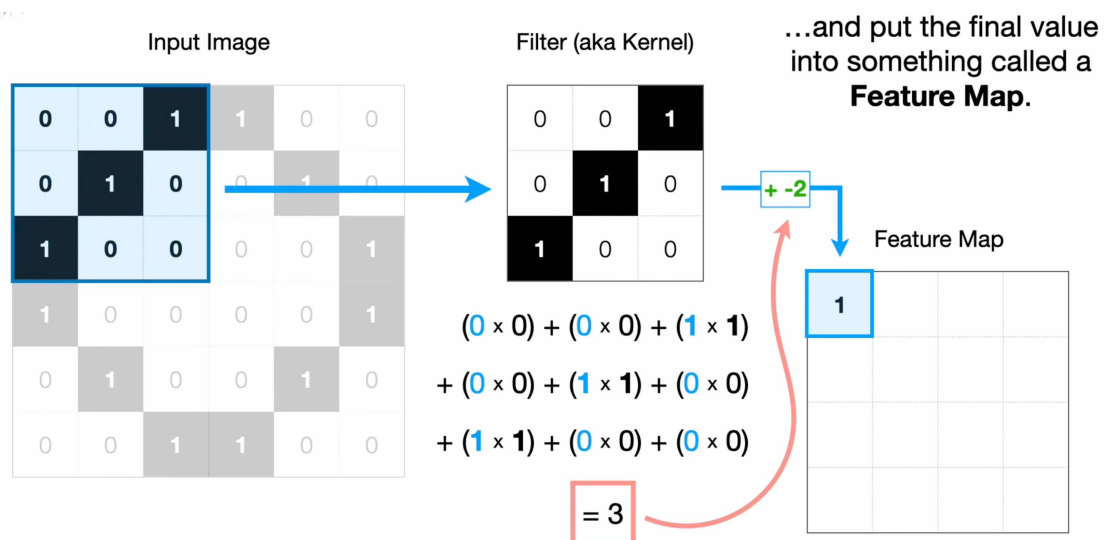
↳ Basically, a bit mask

↳ Then add each product together to get a value

↳ The sum of all the products is just a dot product.

↳ This dot product between the input and the Filter determines whether the filter is convolved with the input.

↳ then you add a bias to the output of the dot product and put the final value into a Feature Map.



• Then slide the pixel over (# of pixels shifted depends on network)

↳ Do this until whole image has been masked by the filter and hence the feature map is complete.

• The feature map helps us take advantage of any correlations there might be in the image.

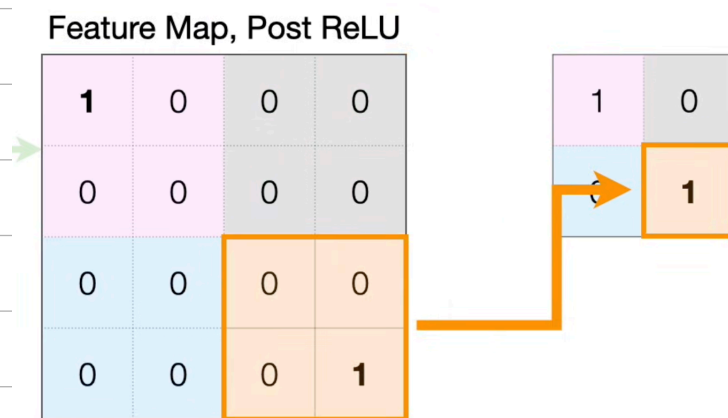
• Typically you run feature map through ReLU so all negative values go to 0, and all positive values stay the same.

• The next step is to apply another filter to a new feature map.

↳ In this one, the filter is to simply select the max value from the filtered portion.

↳ This filter usually does not overlap itself.

↳ This process is called max pooling



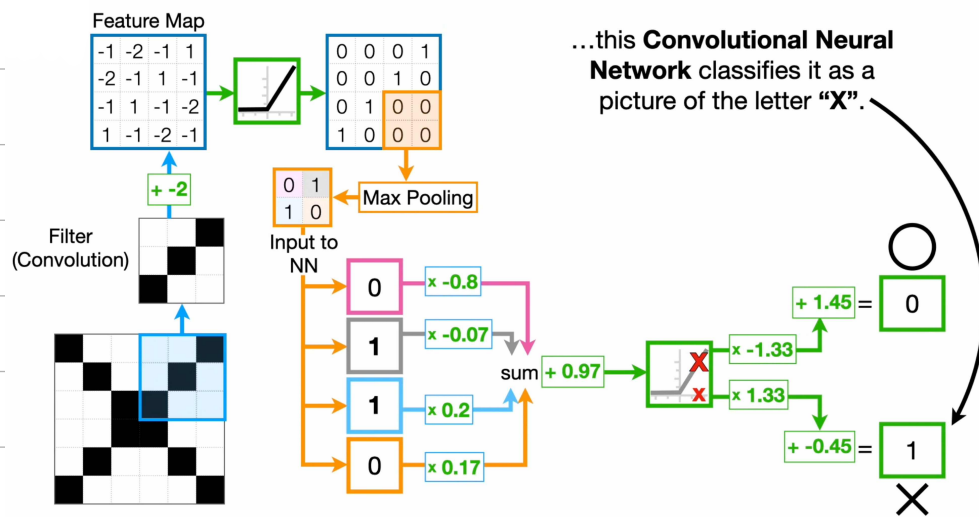
• So max pooling selects the spots where the Filter did the best job matching the input image.

• Alternatively you could calculate the average value for each region and that would be called Average or Mean Pooling.

• You then convert the pooled layer into a column of input nodes.

↳ You then plug the input nodes into a typical Neural Network.

• Full Process Diagram:



• So in this case we compressed from 36 input nodes to just 4.

• This is a simple exam but all Convolutional NN do the following:

1. Use filters a.k.a. Convolution

2. Applying activation function to the filter output (feature map in this example)

3. Pooling the output of the activation function.