Machine Learning Engineer Course
Day 18

- Convolutional Neural Network 1 -

DIVE INTO CODE

Thursday July 8, 2021

DIOP Mouhamed

Agenda

1. Check-in
2. How to proceed
3. Quick Review
4. Convolutional Neural Network
5. Assignment
6. Scratch CNN1 – Sample Code
7. Check-out

Check-in

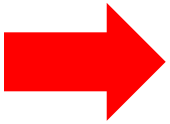
3 minutes Please post the following point to Zoom chat.

Q. What did you learn in the previous week?
(Anything is fine.)

How to proceed - Objective

What is the purpose?

1. Understand CNNs through Scratch
2. Exposure to linear models and different methods



Let’s learn the basics of CNN here

Quick Review (Deep Neural Network)

Shallow Neural Nets: three-layer neural net (with one intermediate layer)

Deep Neural Nets: One intermediate layer is sufficient to approximate an arbitrary function, but that one layer can be unrealistically large.

Generalization capability, vanishing of a gradient

What is CNN?

A convolutional neural network (CNN) is a network consisting of convolutional layers with a sparse structure.

It was invented to mimic the process by which humans recognize patterns from visual information.

In 1958, Huebel and Weisel at Harvard University discovered that there are cells in the visual cortex of cats that respond only when shown a line segment with a specific slope. These cells are called simple cells. Simple cells fire only when a specific pattern appears in a localized, effective area.

http://web2.chubu-gu.ac.jp/web_labo/mikami/brain/26/index-26.html

The Flow

① Learn about CNN's breakthrough

② Know the difference between a convolutional layer and a fully coupled layer.

③ Think about what a convolutional layer is.

CNN's Breakthrough

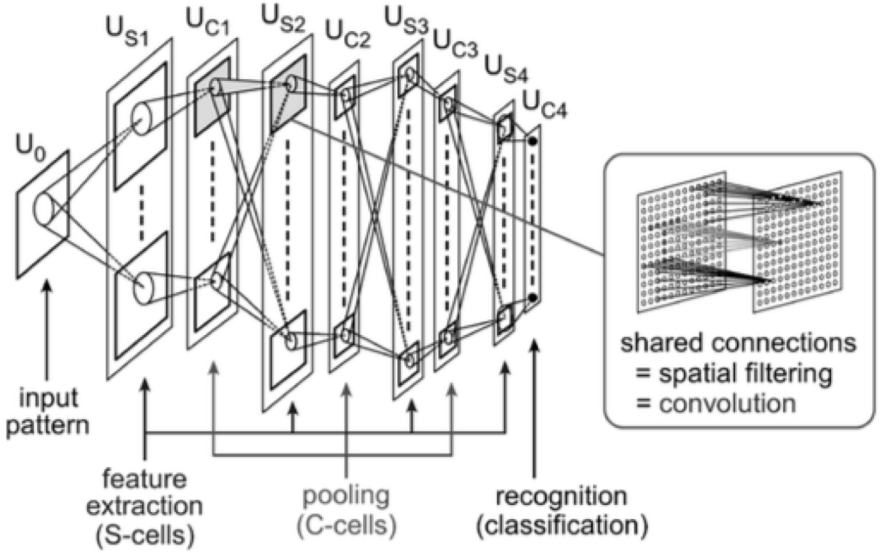
The origin of CNNs can be traced back to Neocognitron [1], which was published by Kunihiko Fukushima in 1982.

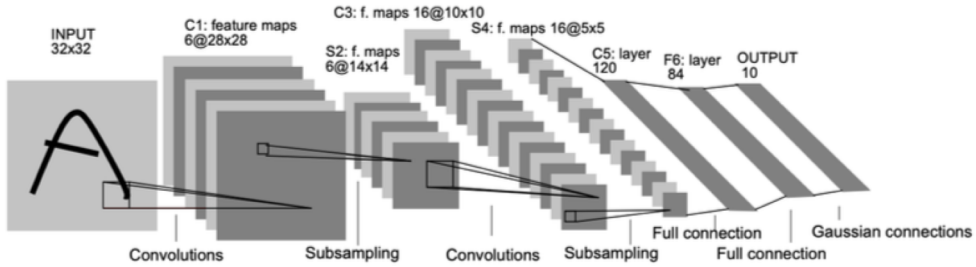
Both current CNNs and Neocognitron are the same hierarchical multilayer networks, with a layer for local feature extraction and a layer for misalignment correction.

Neocognitron uses a learning rule called the Add-if-Silent rule, while the CNN published by Yann LeCun et al. in 1998 (called LeNet [2]) used a generalization of the Delta rule (error back propagation method).

Later, in 2012, a CNN called AlexNet made a breakthrough by winning the ImageNet LSVRC-2012 competition.

[1] http://www.visionsociety.jp/vision/vol29-1/29-1_1.pdf
[2] http://yann.lecun.com/exdb/publis/pdf/lecun-01a.pdf





What is AlexNet?

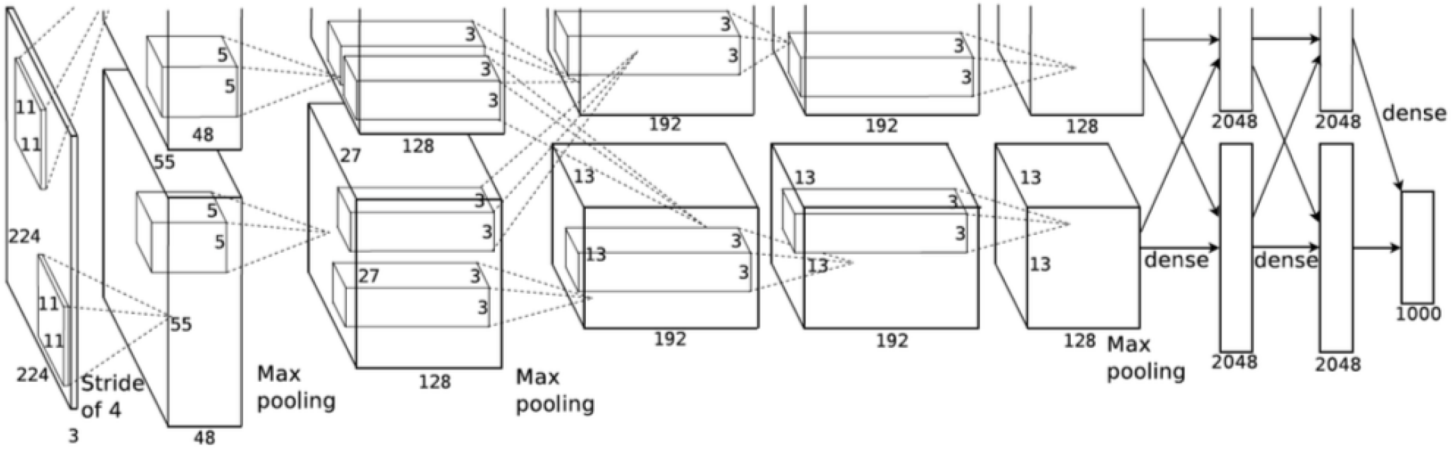
Large Scale Visual Recognition Challenge 2012 (ILSVRC2012)

http://image-net.org/challenges/LSVRC/2012/results.html#abstract

The architecture consists of 8 layers, with 62.3 million network parameters and 1.1 billion computational units in the forward path. The convolutional layer accounts for 90-95% of the computational cost and has 5% of all parameters [3]. (The total coupling layer accounts for 5-10% of the computational cost and 95% of the total parameters)

[3] https://arxiv.org/pdf/1404.5997.pdf

1. To add nonlinearity, Relu was employed instead of Tanh, which accelerated the speed by a factor of 6 with the same accuracy.
2. Dropout was used instead of normalization to deal with overfitting. When the dropout rate was 0.5, training time was doubled.
3. Duplicate pooling reduces the size of the network.



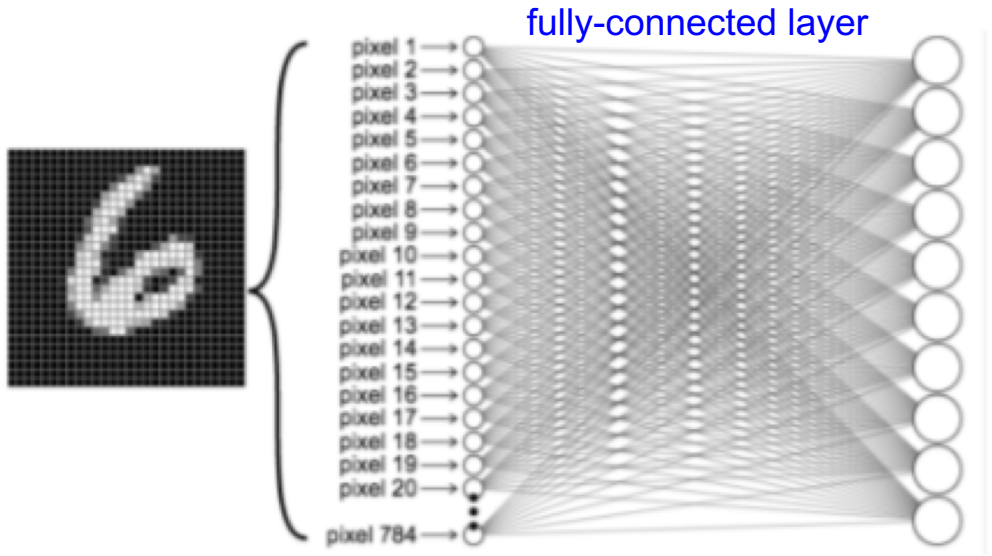
Difference between convolutional and fully-connected layers

Fully-connected layer

In neural networks, 2D data (number of samples, number of features) was passed through all the combined layers (called affine layer, Full Conected Layer, Dense Layer, etc.).

In contrast, convolutional neural networks pass 4-dimensional matrix data (height, width, number of channels, batch size) [1] through convolution.

[1] The order may differ depending on the framework.



Difference between convolutional and fully-connected layers

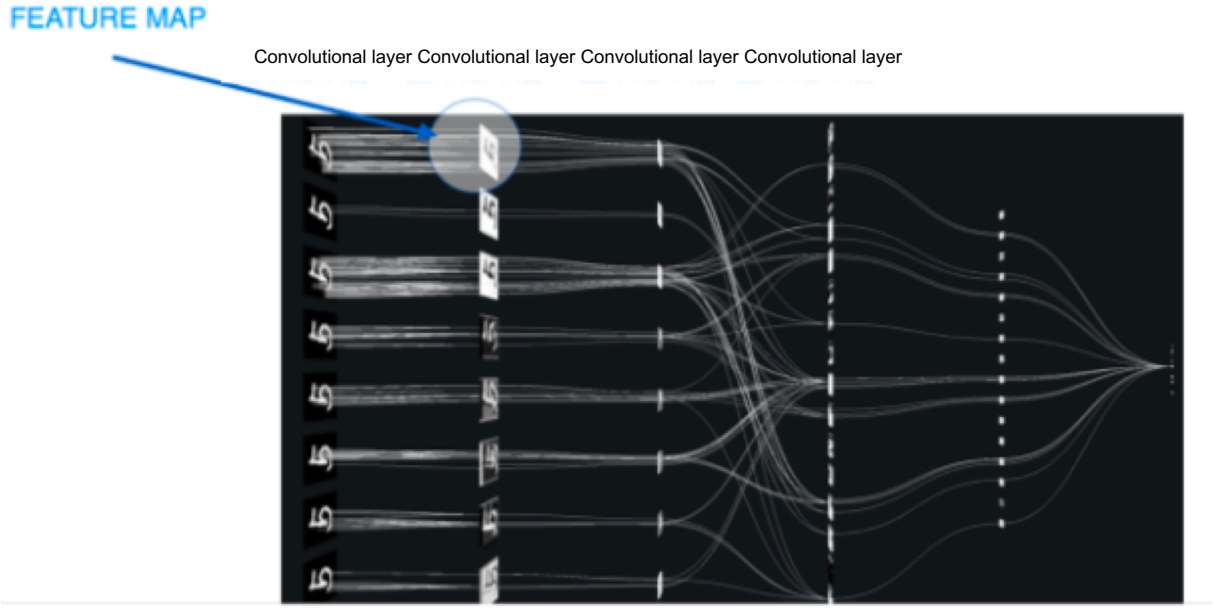
Convolutional layer (also called convolutional layer, Conv2d, etc.)

The output from the convolutional layer is commonly referred to as the **Feature Map.**

Let's look at the site on the right from various angles. https://terencebroad.com/works/cnn-vis

What are the axes of the channel of the 4D matrix?

Why does the output become smaller and smaller?



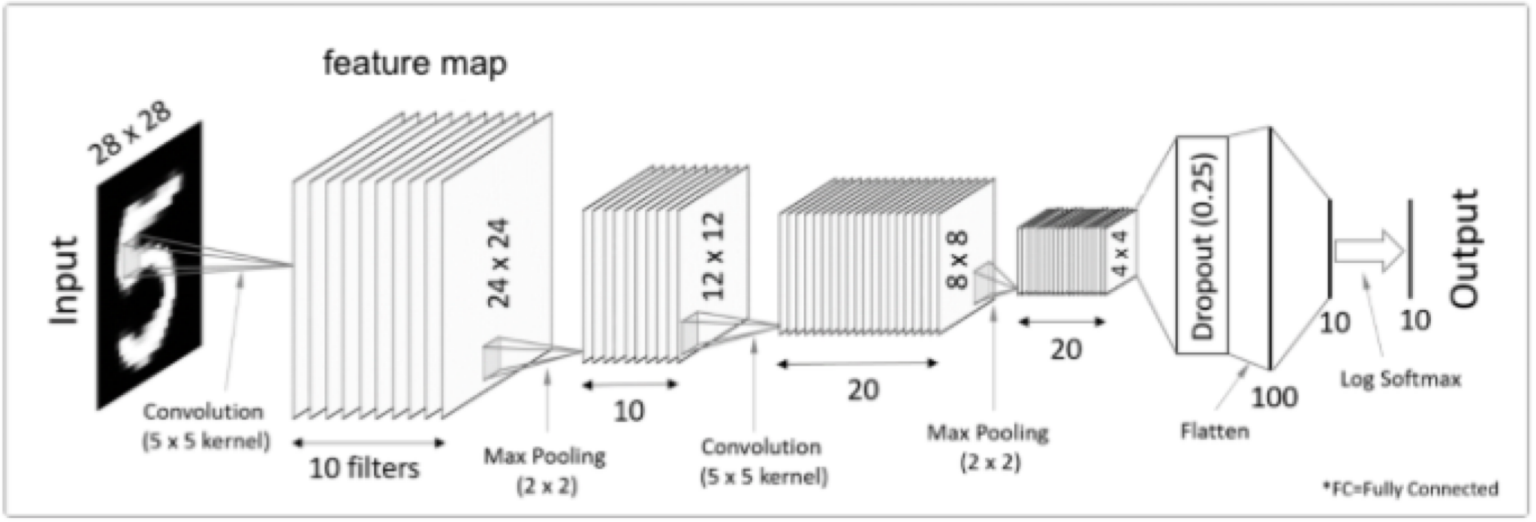
Difference between convolutional and fully-connected layers

Convolutional layer

The weight matrix that the convolutional layer locally combines with the input is called the filter or kernel. (Strictly speaking, a collection of kernels is defined as a filter.

This filter extracts the features of the input (patterns called edges and textures).

In two-dimensional convolution, one filter is offset horizontally and vertically across the input to exhaustively search the entire input. The following is a classical CNN architecture.

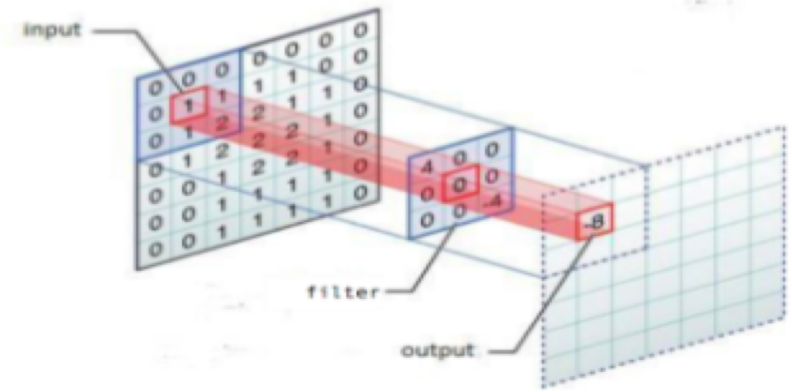


Difference between convolutional and fully-connected layers

Convolutional layer

A filter is a matrix of some kind that extracts features (patterns) by performing a sum-of-products operation on the overlapping parts of the input, with each filter offset horizontally and vertically (in pixels for images), and the result of each sum-of-products operation output (i**nstead of changing filters for each local area, one filter covers the whole area, so multiple pixels share the same weight values**). If the filter size is larger than 1×1, the output will be smaller in height and width than the input.

If you want to output the input keeping its aspect size, fill the outer pixel of the input with a specific value (usually zero) before the convolution operation (this is called zero padding). The filter by itself is a 3-dimensional matrix (filter height, filter width, channel number), but in one convolution layer, it is prepared as a 4-dimensional matrix (filter height, filter width, channel number, filter number). In other words, multiple filters are applied to a single input. Therefore, the number of filters can be viewed as the number of nodes in the NN.



What is a convolutional layer?

What is "convolution"?

In mathematics, there is an operation called **convolutional integration** for continuous values, and convolution in CNN is **convolutional sum** of products for discrete values.

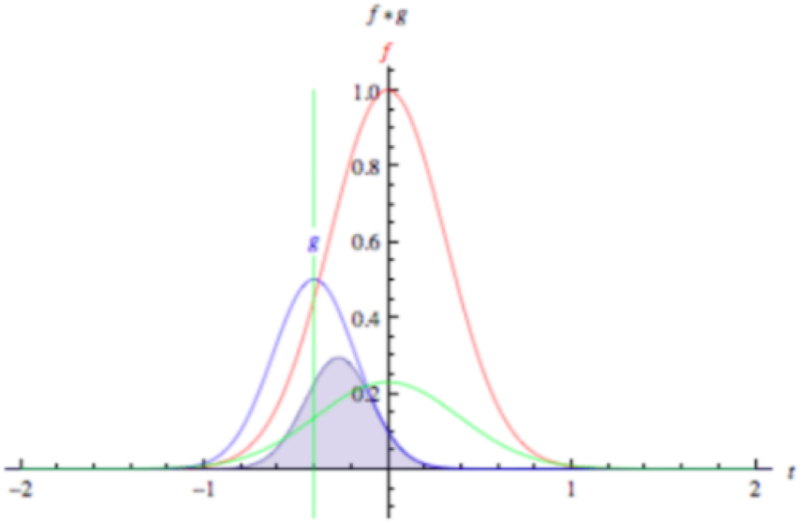
https://www.clg.niigata-u.ac.jp/~medimg/practice_medical_imaging/imgproc_scion/4filter/index.htm

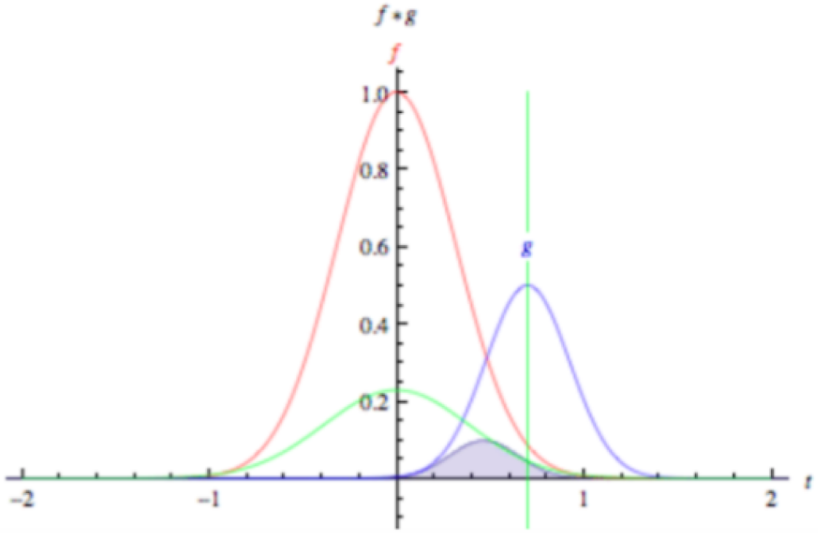
http://tdual.hatenablog.com/entry/2018/05/02/113110

The "convolution" of CNN can be captured as follows.

Sum of products: extract features

Convolution: brute force (measure the overlap)





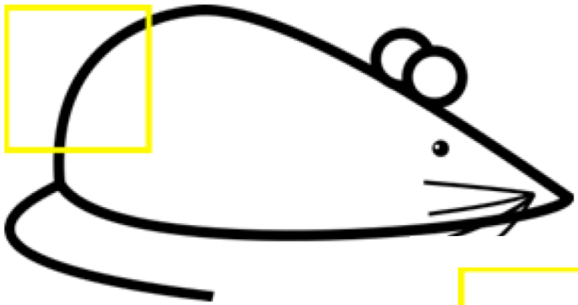
https://pathmind.com/wiki/convolutional-network

Extract features with filters

How does a filter detect features?

Before neural networks, the value of the filter was determined by human calculation.

For example, a filter suitable for detecting a receptive field like A in the figure below has a value like B.



Visualization of the filter on the image

Visualization of the receptive field

A: Need wild

0	0	0	0	0	0	30
0	0	0	0	50	50	50
0	0	0	20	50	0	0
0	0	0	50	50	0	0
0	0	0	50	50	0	0
0	0	0	50	50	0	0
0	0	0	50	50	0	0

Pixel representation of the receptive field

*

B: Filter

0	0	0	0	0	30	0
0	0	0	0	30	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	0	0	0	0

Pixel representation of filter

Multiplication and Summation = (50*30)+(50*30)+(50*30)+(20*30)+(50*30) = 6600 (A large number!)

https://adeshpande3.github.io/A-Beginner's-Guide-To-Understanding-Convolutional-Neural-Networks/

Extract features with filters

How does a filter detect features?

When filter B is locally integrated for receptive field A as shown below, the calculation result is "0".

(No features are extracted.)

Visualization of the filter on the image

A: Need wild

0	0	0	0	0	0	0
0	40	0	0	0	0	0
40	0	40	0	0	0	0
40	20	0	0	0	0	0
0	50	0	0	0	0	0
0	0	50	0	0	0	0
25	25	0	50	0	0	0

Pixel representation of receptive field

*

B: Filter

0	0	0	0	0	30	0
0	0	0	0	30	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	0	0	0	0

Pixel representation of filter

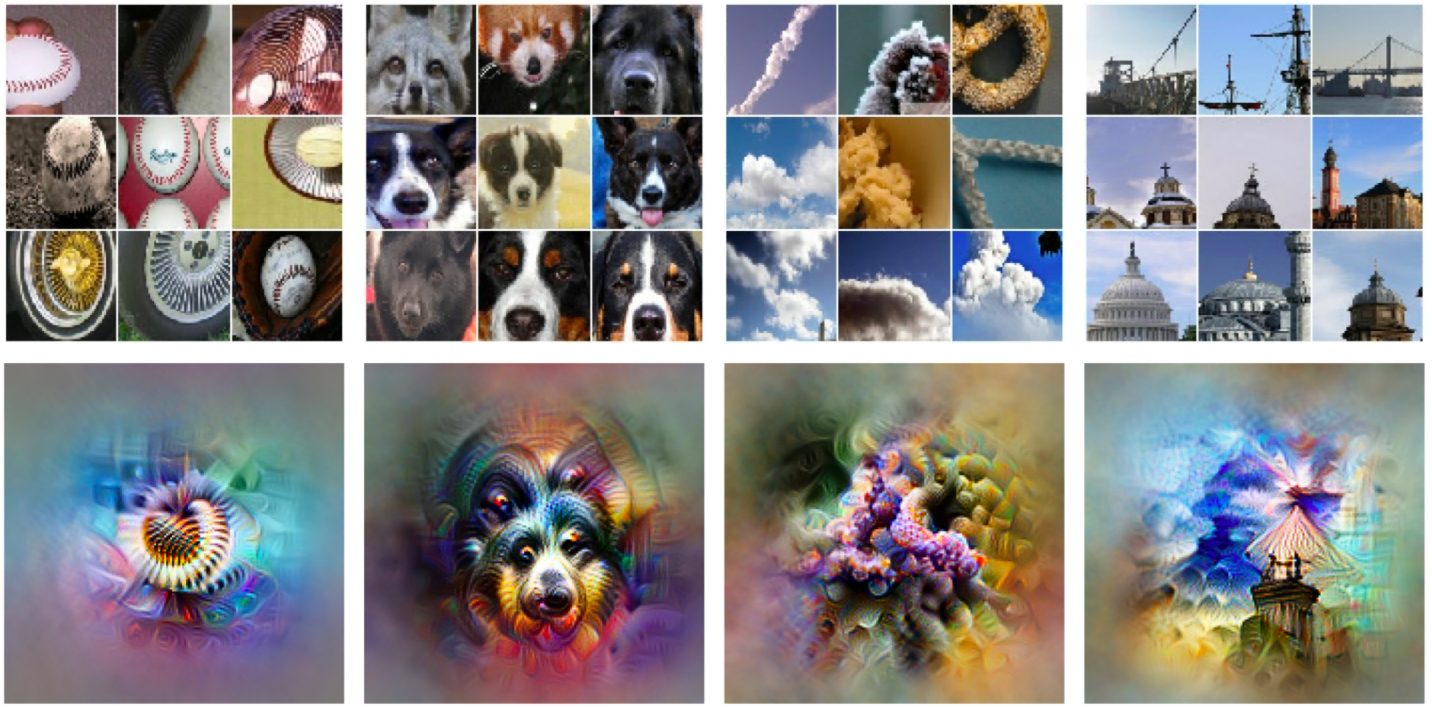
Multiplication and Summation = 0

https://adeshpande3.github.io/A-Beginner's-Guide-To-Understanding-Convolutional-Neural-Networks/

Feature Map

What CNN is looking at

Let's take a peek at CNN's optimized Feature Map.

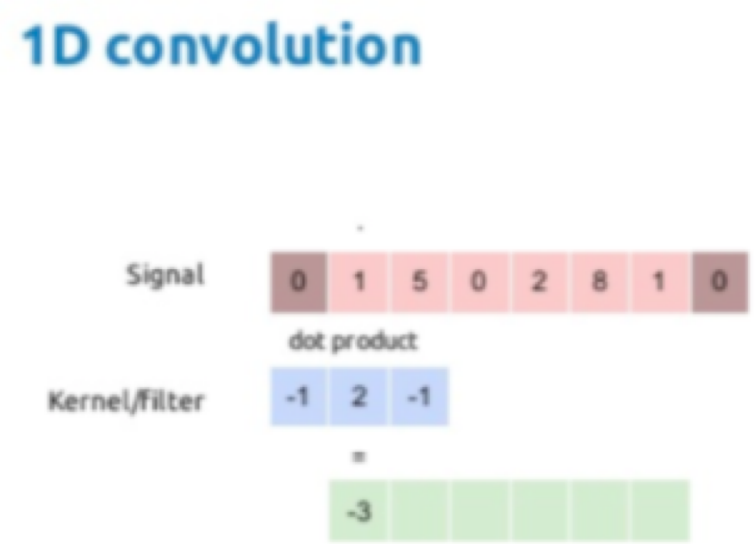


https://distill.pub/2017/feature-visualization/

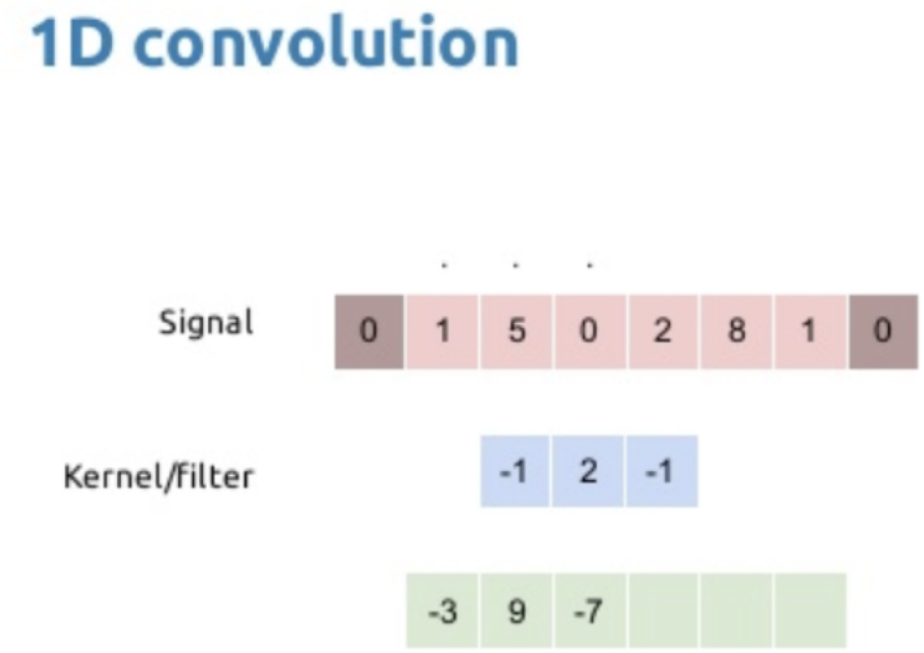
What is 1D convolution?

It's based on 2D convolution.
Let's start with 1D convolution.

Let's follow the filter in action.
The settings are: zero padding = 1, stride = 1.



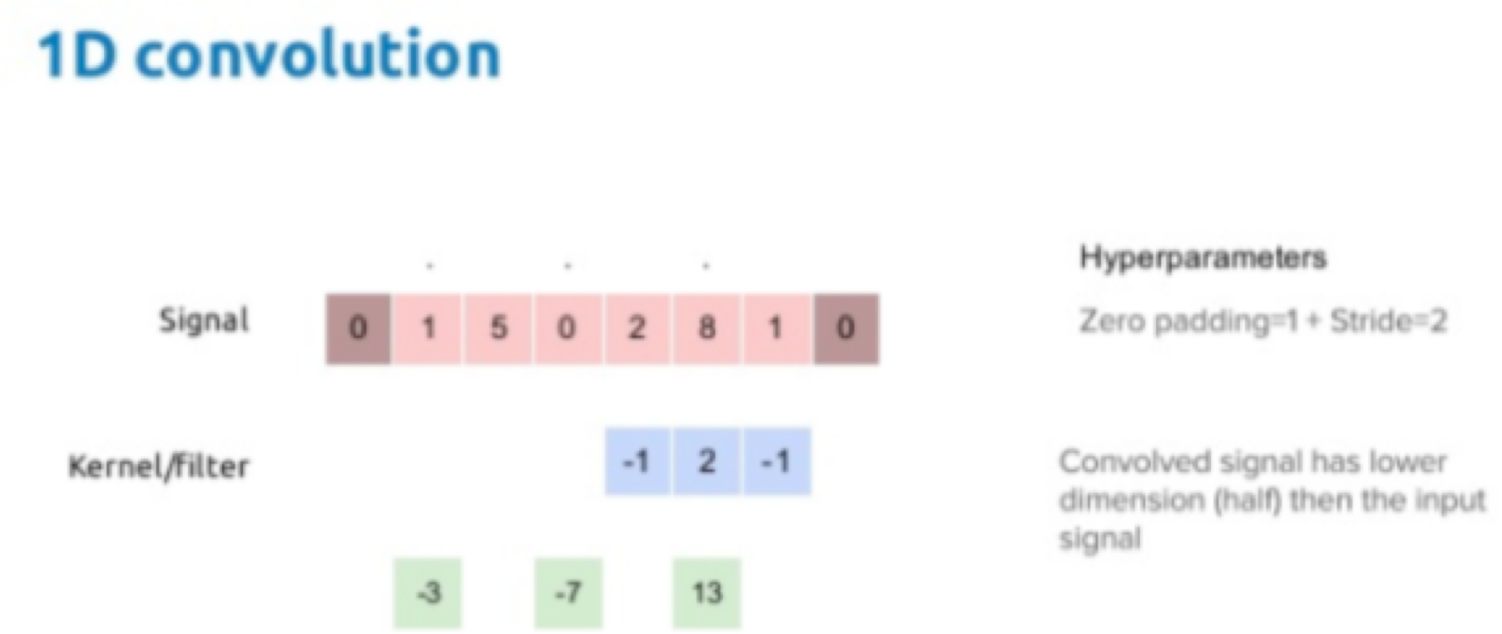
What is 1D convolution?



What is 1D convolution?

What is 1D convolution?

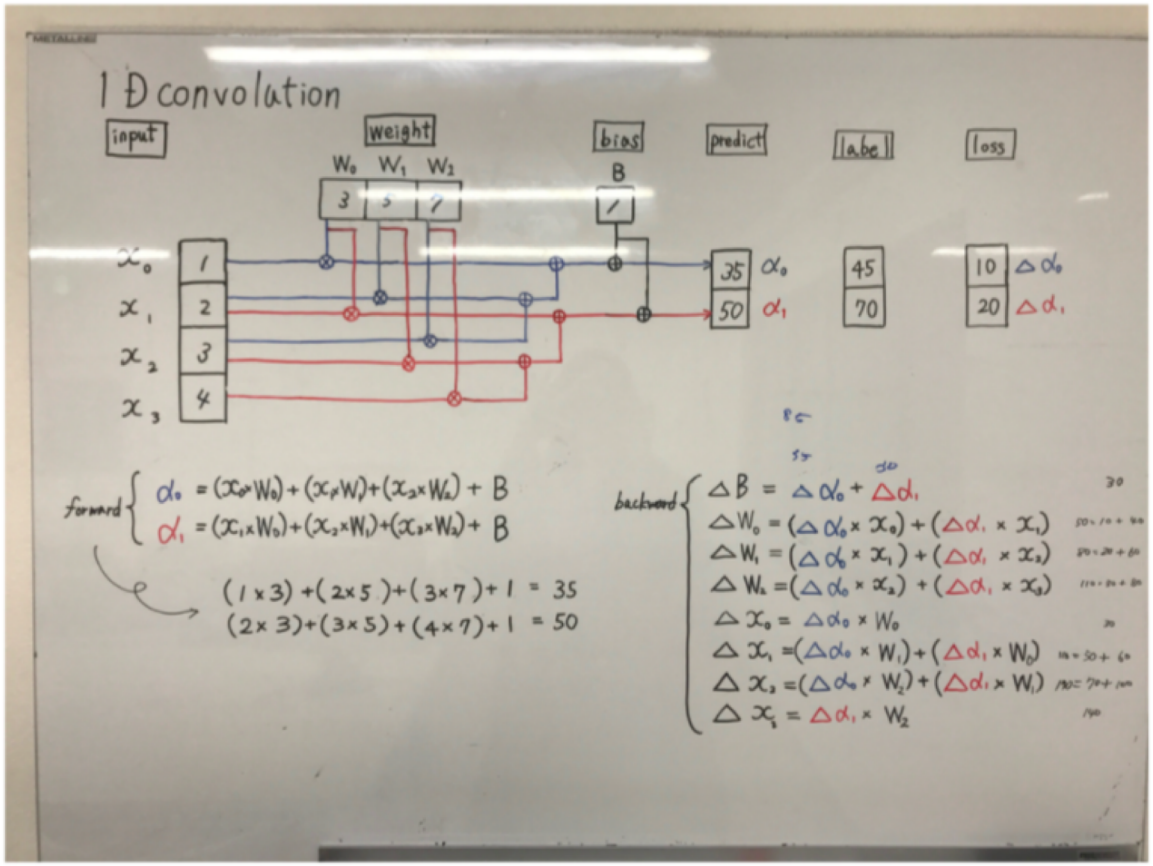
If stride = 2,
the output size becomes smaller.



Tips for conv1D

Today's DIVER

Let's run the unit test data through conv1D.



Summary

1. The convolutional layer extracts patterns from the local receptive field (effective range)

2. One filter that extracts patterns is an exhaustive search over the entire input data by offsetting local joins (weight sharing)

Sample code

How to solve problems

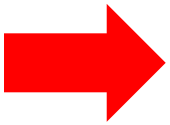
“Scratch Convulutional Neural Network 1”

[Problem 1] Creating a 1D convolutional layer class with the number of channels limited to 1
[Problem 2] Calculating the output size after 1D convolution
[Problem 3] Experiments with 1D convolutional layers on small arrays
[Problem 4] Creating a 1D convolutional layer class without limiting the number of channels
[Problem 6] (Advanced Problem) Dealing with mini-batches
[Problem 8] Learning and Estimation

Sprint 11 – Convolutional Neural Network 1

Explanation about this Sprint is given but please try it on your own first.

Sprint 11 – Convolutional Neural Network 1



Please work on your own after class and submit your assignments on DIVER.

Sprint 11 – Convolutional Neural Network 1

A Sample Code of this Sprint is given but please try it on your own.

Sprint 11 – Convolutional Neural Network 1

Please work on your own after class and submit your assignments on DIVER.

ToDo by next class

Next class will be Zoom : Thursday July 15, 2021 19:00 ~ 20:30

🚨ToDo: Convolutional Neural Network 2 (SimpleConv2d)
https://diver.diveintocode.jp/curriculums/1900

Check-out

3 minutes Please post the following point to Zoom chat.

Q. Current feelings and reflections
(joy, anger, sorrow, anticipation, nervousness, etc.)

Thank You For Your Attention