CIS 678 Machine Learning

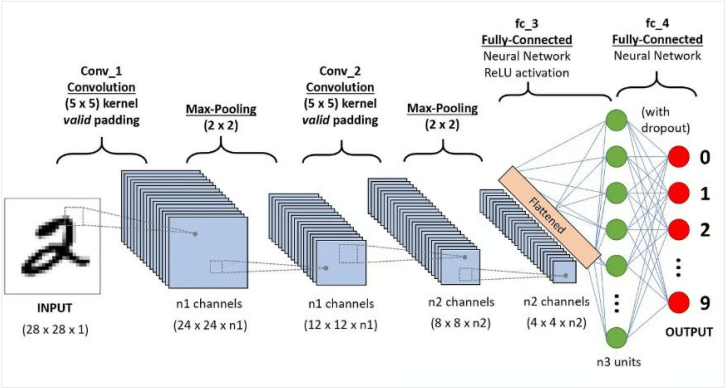
Convolutional Neural Networks (CNNs)

Popular Neural Network (NN) Architectures

- ***Feed Forward NNs***
- ***Convolutional NNs***
- Sequence models
  - LSTM
  - Transformers
- Generative AI
  - GANs
  - VAEs
- Graph Neural Networks

Popular Neural Network (NN) Architectures

- *Feed Forward NNs*
- *Convolutional NNs*
- Sequence models
  - LSTM
  - Transformers
- Generative AI
  - GANs
  - VAEs
- Graph Neural Networks



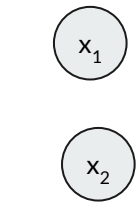
Quiz!

Question?

Q. Draw the diagram of a Feed Forward Neural Network with the properties given below, and estimate the minimum number of parameters your model would have:

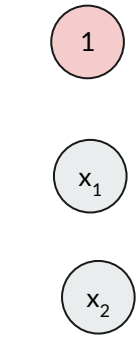
1. **Input layer**: 2 nodes (to consume 2 input features, $\{x_1, x_2\}$)
2. **Two (2) Hidden layers** with the following configuration:
   i) Hidden layer one: 2 nodes
   ii) Hidden layer two: 2 nodes
   One bias input node for each hidden layer in (2)
3. **Output layer**: 1 node (y)

FFNN Parameters



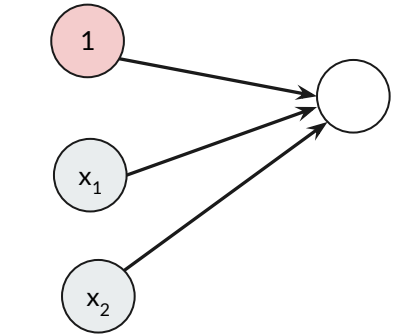
Input layer

FFNN Parameters

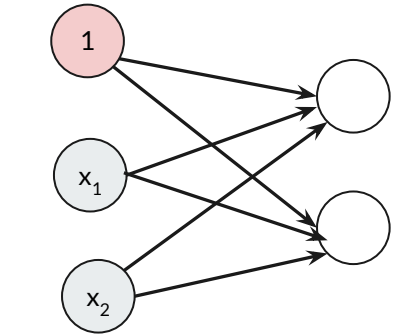


Input layer

FFNN Parameters



FFNN Parameters

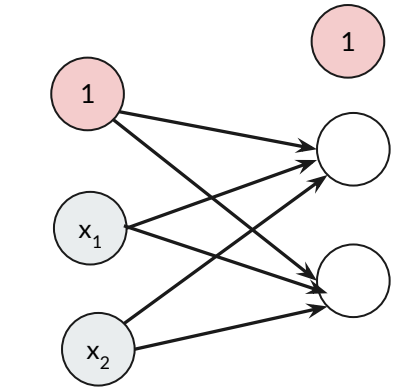


Input layer

Hidden layer(1)

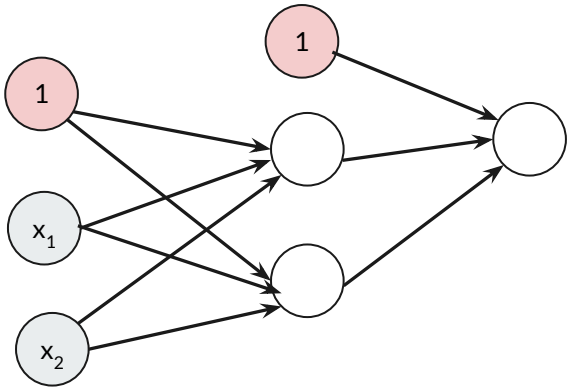
$(2 + 1) \times 2$

FFNN Parameters



(2 + 1) x 2

FFNN Parameters

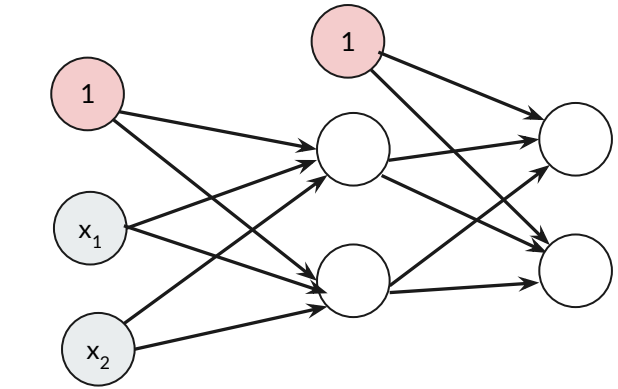


Input layer

Hidden layer(1)

$(2 + 1) \times 2$ +

FFNN Parameters

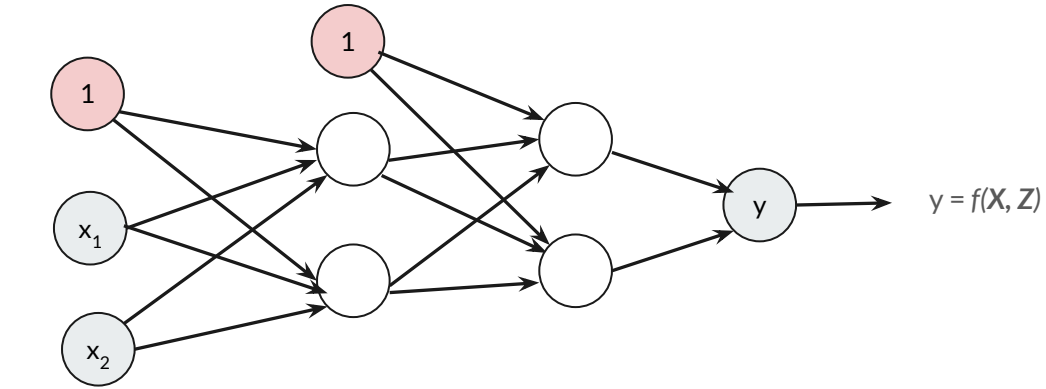


Input layer

Hidden layer(1)

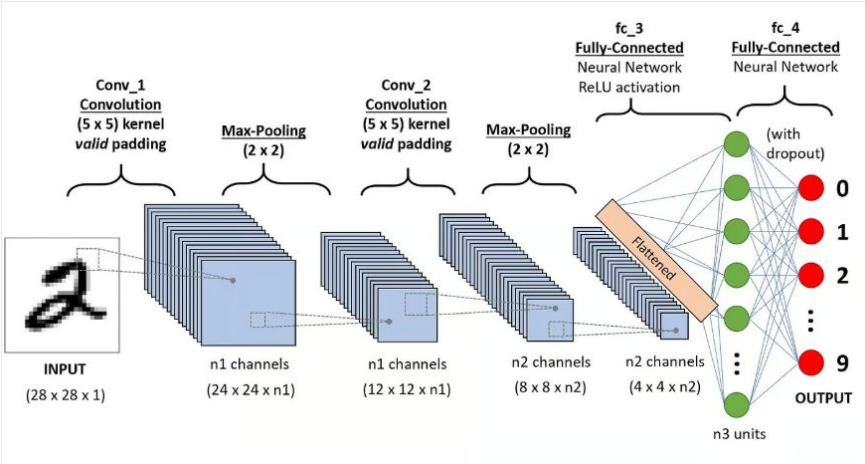
$(2 + 1) \times 2 \quad + \quad (2 + 1) \times 2$

FFNN Parameters



$$(2 + 1) \times 2 \quad + \quad (2 + 1) \times 2 \quad + \quad 2 \times 1 \quad = \quad 14$$

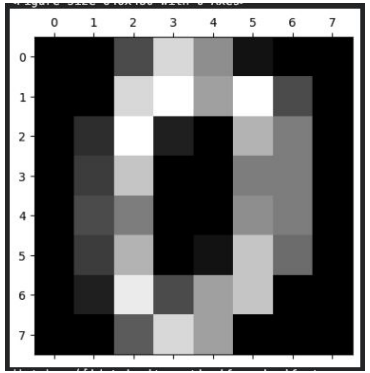
Convolutional Neural Networks (CNNs)



Digit Classification examples

MNIST Image Classification

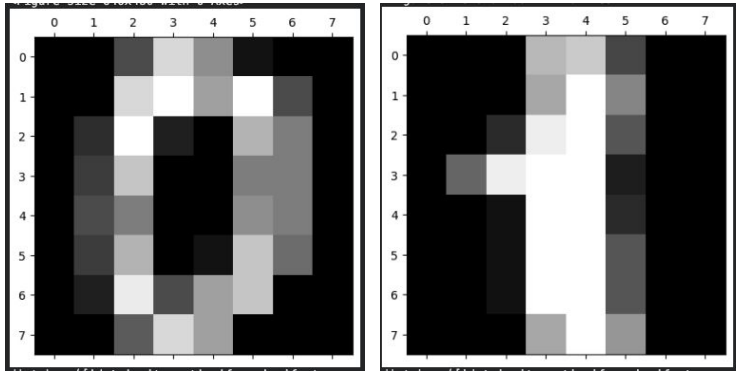
What does it look like?



Digit Classification examples

MNIST Image Classification

What does it look like?

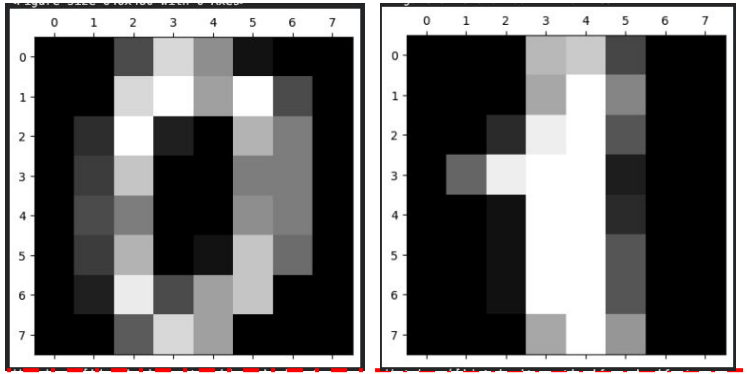


Digit Classification examples

[MNIST Image Classification]

Images of Size (2D Matrix): 8x8

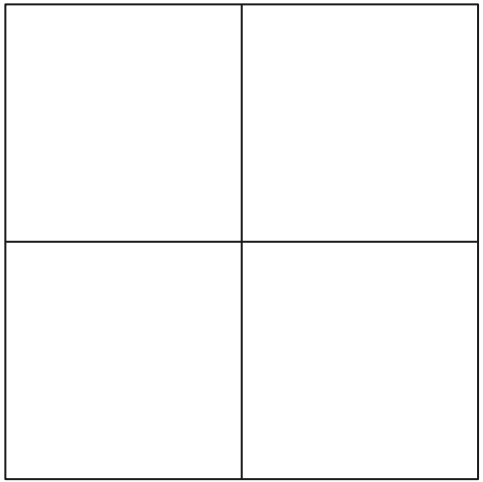
Corresponding Vector Representation of Size: 64



	pixel_0_0	pixel_0_1	pixel_0_2	pixel_0_3	pixel_0_4	pixel_0_5	pixel_0_6	pixel_0_7	pixel_1_0	pixel_1_1	...	pixel_6_6	pixel_6_7	pixel_7_0	pixel_7_1	pixel_7_2	pixel_7_3	pixel_7_4	pixel_7_5	pixel_7_6	pixel_7_7
0	0.0	0.0	5.0	13.0	9.0	1.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	6.0	13.0	10.0	0.0	0.0	0.0
1	0.0	0.0	0.0	12.0	13.0	5.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	11.0	16.0	10.0	0.0	0.0
2	0.0	0.0	0.0	4.0	15.0	12.0	0.0	0.0	0.0	0.0	...	5.0	0.0	0.0	0.0	0.0	3.0	11.0	16.0	9.0	0.0
3	0.0	0.0	7.0	15.0	13.0	1.0	0.0	0.0	0.0	8.0	...	9.0	0.0	0.0	0.0	7.0	13.0	13.0	9.0	0.0	0.0
4	0.0	0.0	0.0	1.0	11.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	2.0	16.0	4.0	0.0	0.0

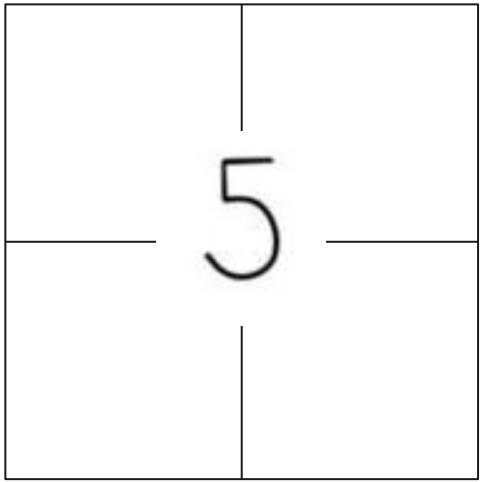
Why FF Networks don’t work well for Image Data?

- We ask you to write a “5”



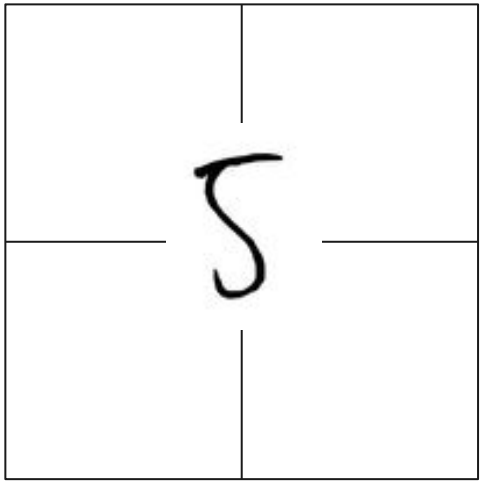
Why FF Networks don't work well for Image Data?

- We ask you to write a "5"
- **A perfect "5" at the center of the coordinate**



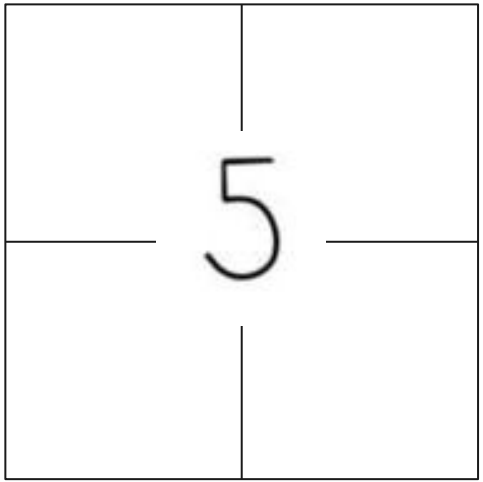
Why FF Networks don't work well for Image Data?

- We ask you to write a "5"
- A perfect "5" at the center of the coordinate
- **This is my "5", not good as yours but still at the center of the coordinate.**



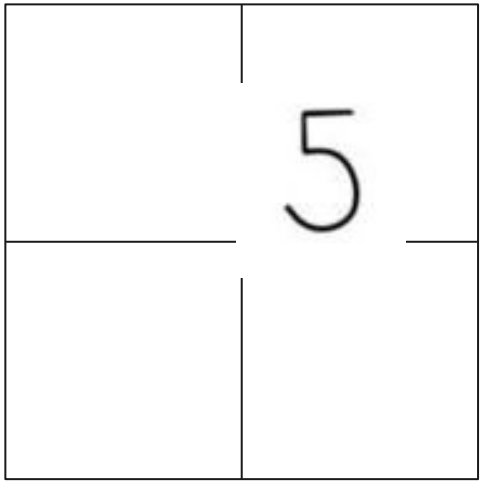
Why FF Networks don't work well for Image Data?

- We ask you to write a "5"
- A perfect "5" at the center of the coordinate
- Your "5" still will have a lots of variations



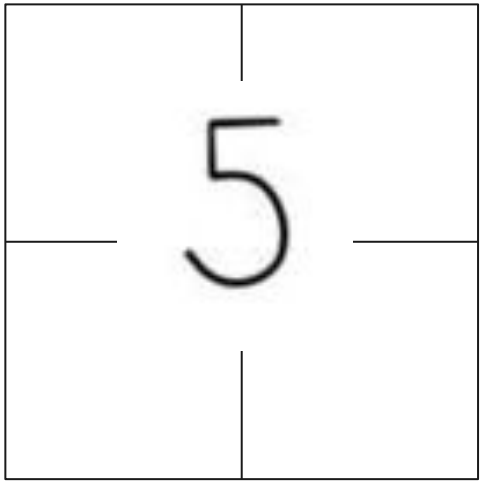
Why FF Networks don't work well for Image Data?

- We ask you to write a "5"
- A perfect "5" at the center of the coordinate
- Your "5" still will have a lots of variations
- **Translated "5"**



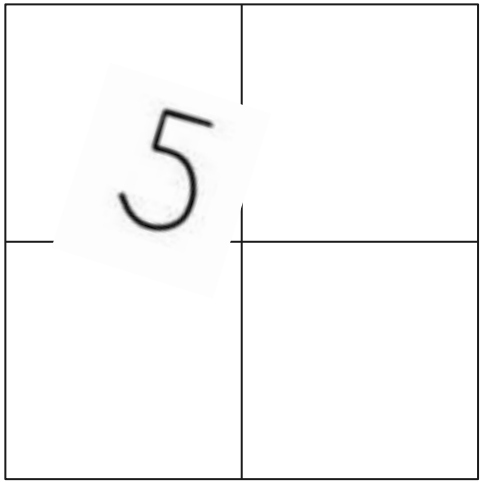
Why FF Networks don't work well for Image Data?

- We ask you to write a "5"
- A perfect "5" at the center of the coordinate
- Your "5" still will have a lots of variations
- **Scaled "5"**



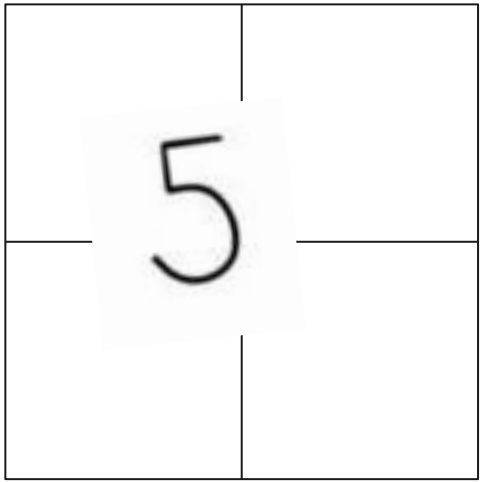
Why FF Networks don't work well for Image Data?

- We ask you to write a "5"
- A perfect "5" at the center of the coordinate
- Your "5" still will have a lots of variations
- **Translated and Rotated "5"**



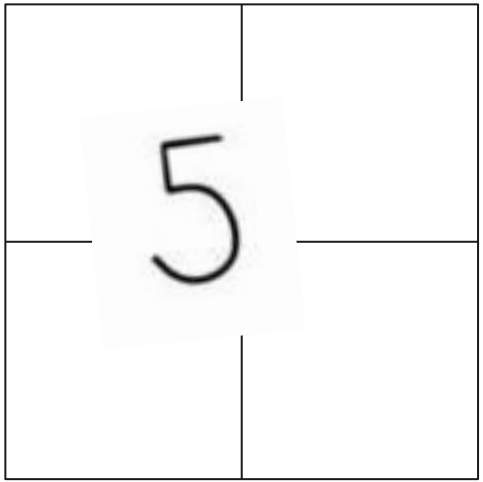
Why FF Networks don't work well for Image Data?

- We ask you to write a "5"
- A perfect "5" at the center of the coordinate
- Your "5" still will have a lots of variations
- **Translated, Rotated and Scaled "5"**



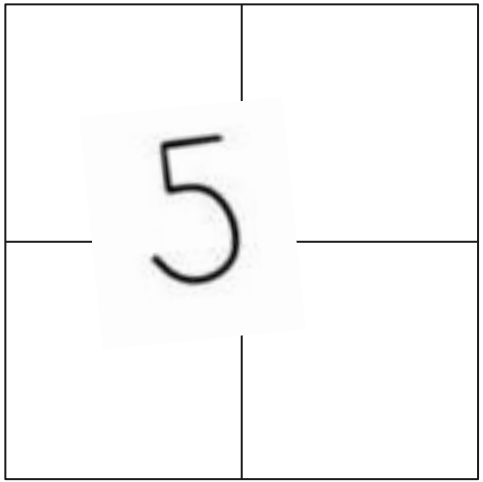
Why FF Networks don't work well for Image Data?

- We ask you to write a "5"
- A perfect "5" at the center of the coordinate
- **Your "5" still will have a lots of variations and each definition is unique in the vector space**



Why FF Networks don't work well for Image Data?

- We ask you to write a "5"
- A perfect "5" at the center of the coordinate
- Your "5" still will have a lots of variations and each definition is unique in the vector space
- **How many different "5"s we will need provide the model to learn a perfect model?**



Convolutional Neural Networks (CNNs)

- State of the Art for CV and some other problems
- Filters/Convolutional Kernels

Convolutional NNs

- State of the Art for CV and some other problems
- Filters/Convolutional Kernels

Examples:

- *Alexnet*
- *VGGNet*
- *ResNet*
- *GoogLeNet*
- ..

ref

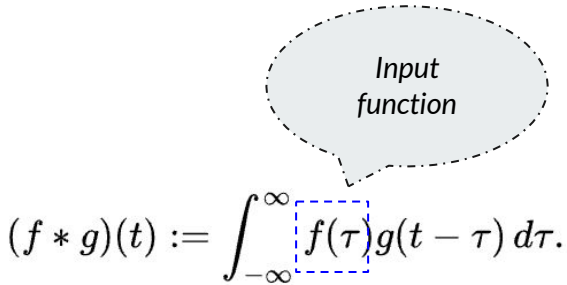
Convolution

- Is a mathematical operation

$$(f * g)(t) := \int_{-\infty}^{\infty} f(\tau) g(t - \tau)\, d\tau.$$

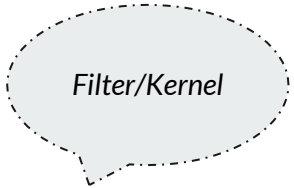
Convolution

- Is a mathematical operation



Convolution

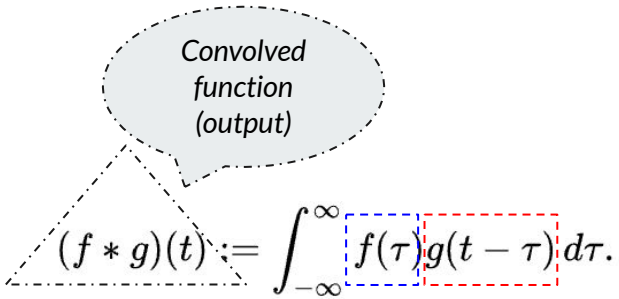
- Is a mathematical operation



$$(f * g)(t) := \int_{-\infty}^{\infty} f(\tau) g(t - \tau) \, d\tau.$$

Convolution

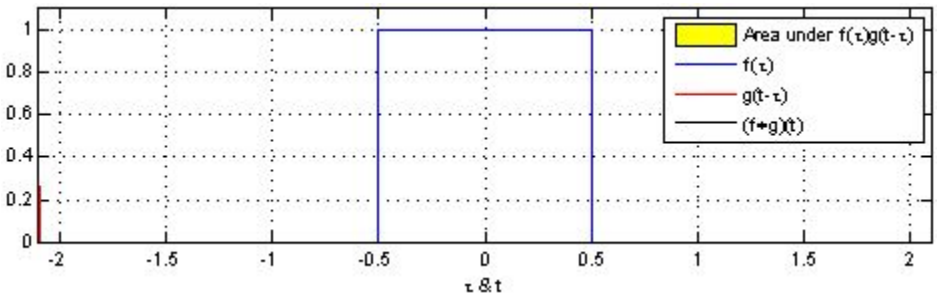
- Is a mathematical operation



Convolution

- Is a mathematical operation

$$(f * g)(t) := \int_{-\infty}^{\infty} f(\tau)\, g(t-\tau)\, d\tau.$$



Src

Convolution

Let's learn through a practical example

Moving average (Discrete case):

- Replaces each vector values pixel with an average of all the values in its neighborhood
- Moving average in 1D:

  [1, 1, 1] / 3

$f(\tau)$

2	5	0	1	3

1D vector: 5 elements

ref

Convolution

Let's learn through a practical example

Moving average (Discrete case):

- Replaces each vector values pixel with an average of all the values in its neighborhood
- Moving average in 1D:

  [1, 1, 1] / 3

$g(t-\tau)\,d\tau.$

1	1	1

1/3

Known as Filter/Kernel

$f(\tau)$

2	5	0	1	3

1D vector: 5 elements

ref

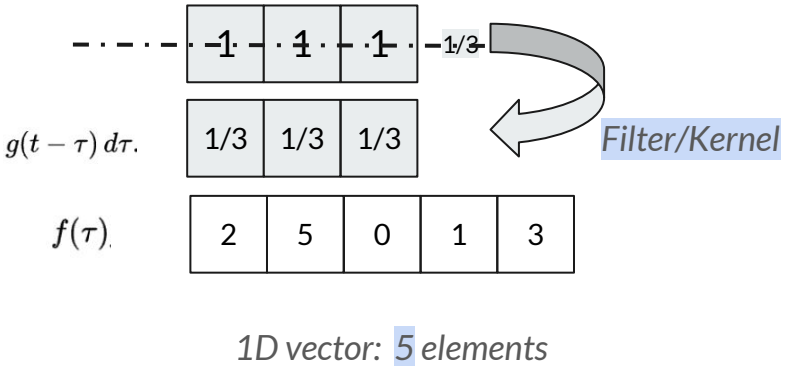
Convolution

Let's learn through a practical example

Moving average (Discrete case):

- Replaces each vector values pixel with an average of all the values in its neighborhood
- Moving average in 1D:

  [1, 1, 1] / 3



1D vector: 5 elements

ref

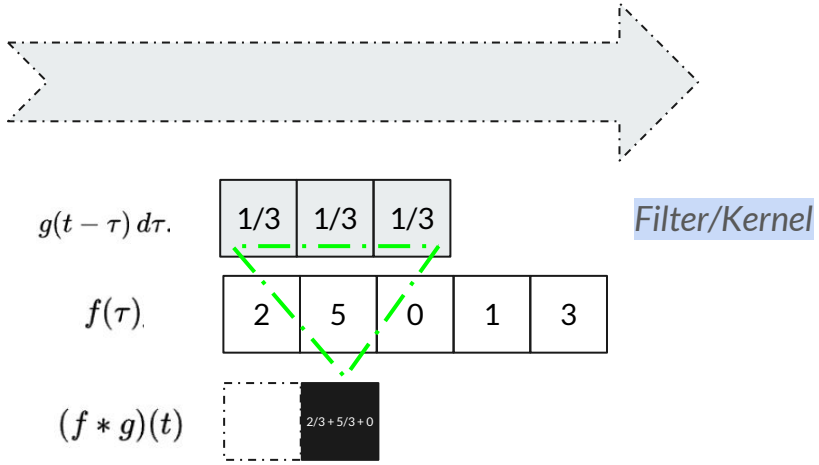
Convolution

Let's learn through a practical example

Moving average (Discrete case):

- Replaces each vector values pixel with an average of all the values in its neighborhood
- Moving average in 1D:

  [1, 1, 1] / 3



ref

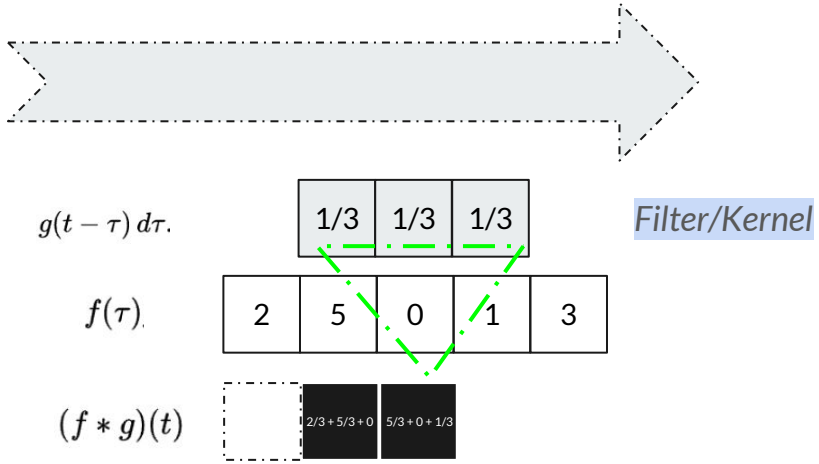
Convolution

Let's learn through a practical example

Moving average (Discrete case):

- Replaces each vector values pixel with an average of all the values in its neighborhood
- Moving average in 1D:

  [1, 1, 1] / 3



ref

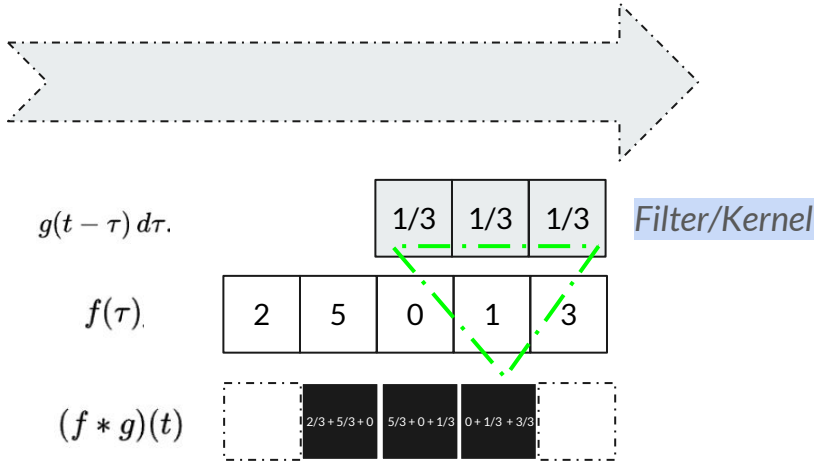
Convolution

Let's learn through a practical example

Moving average (Discrete case):

- Replaces each vector values pixel with an average of all the values in its neighborhood
- Moving average in 1D:

  [1, 1, 1] / 3



ref

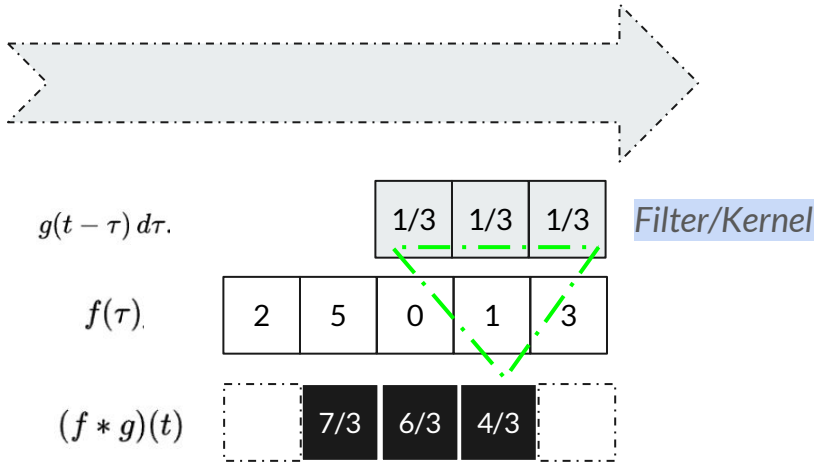
Convolution

Let's learn through a practical example

Moving average (Discrete case):

- Replaces each vector values pixel with an average of all the values in its neighborhood
- Moving average in 1D:

  [1, 1, 1] / 3

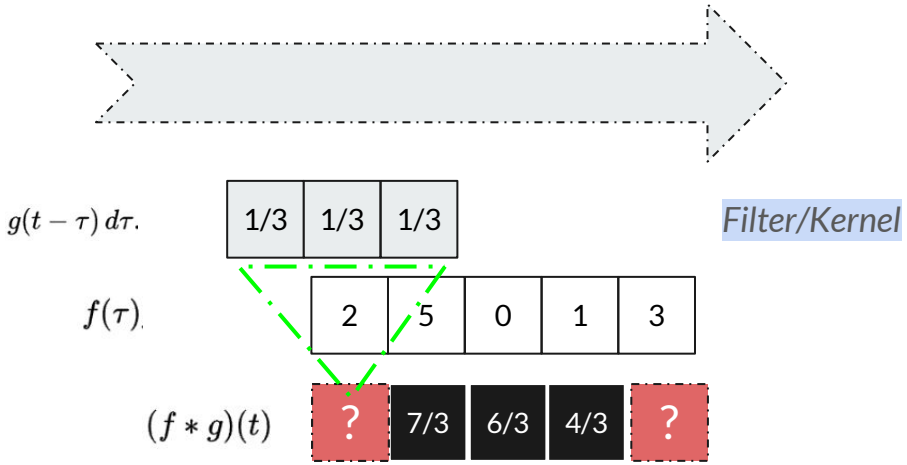


ref

Convolution - Concepts

How can we ensure that the input and output sizes are uniform?

Padding (0)

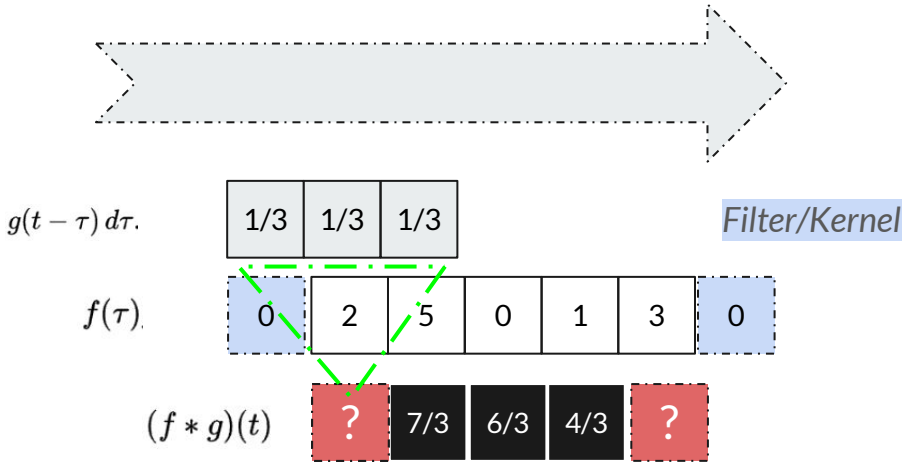


ref

Convolution - Concepts

How can we ensure that the input and output sizes are uniform?

Padding (0)

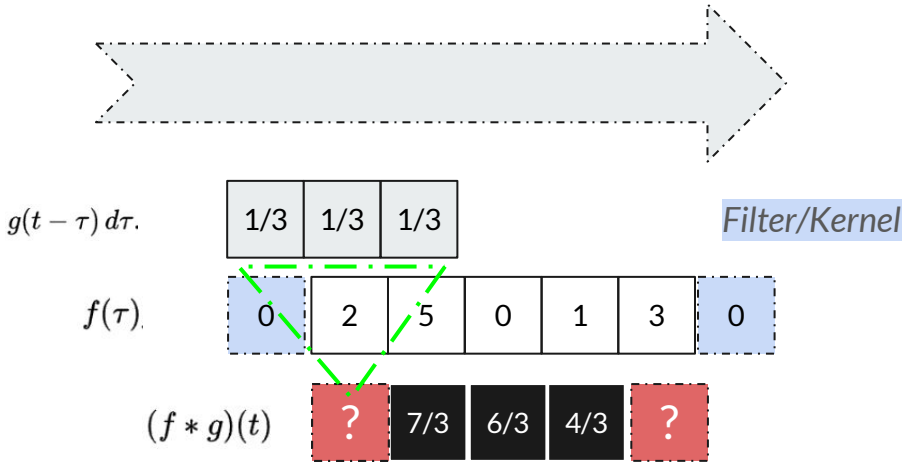


ref

Convolution - Concepts

Faster Operation?
Skip closely local context?

Stride (1)

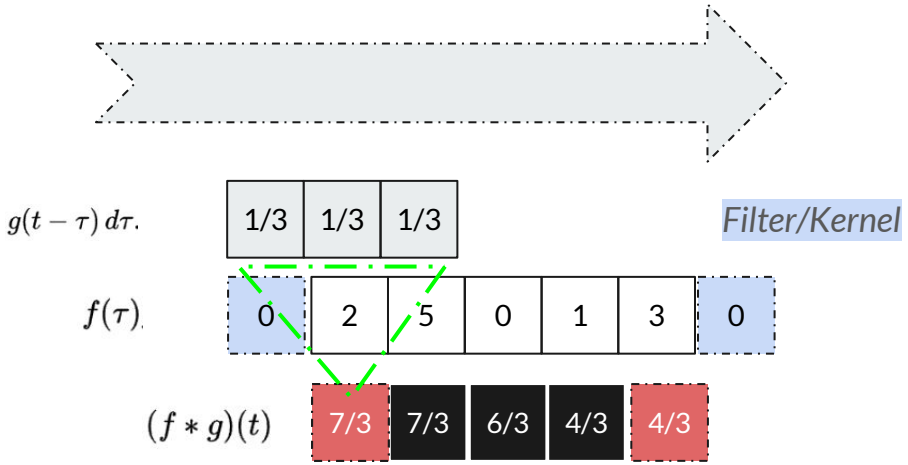


ref

Convolution - Concepts

Faster Operation?
Skip closely local context?

Stride (1)

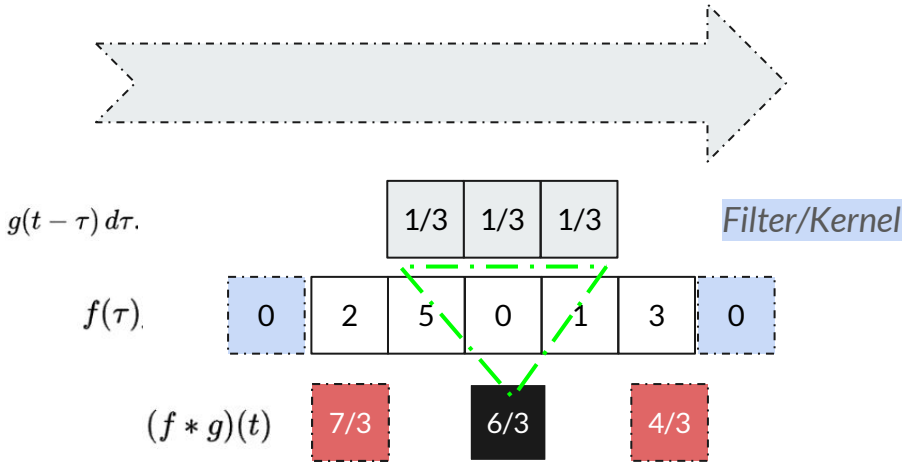


ref

Convolution - Concepts

Faster Operation?
Skip closely local context?

Stride (2)



ref

From 1D to 2D

This is a general concept and applicable to any dimensions!

Image Filters

Moving Average in 2D

- 2D convolution operation

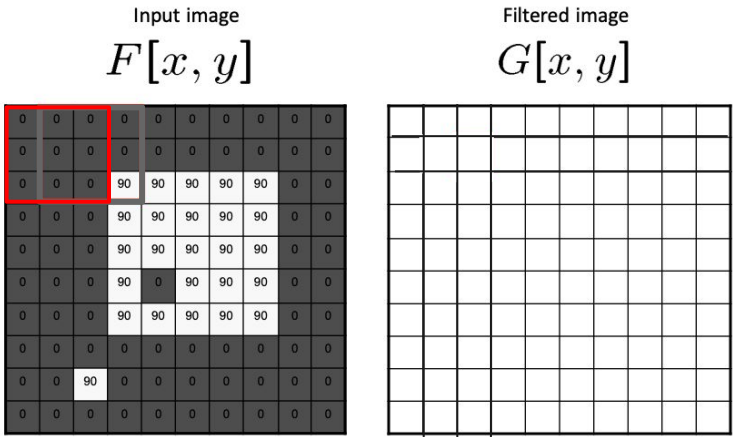


Image Filters

Moving Average in 2D

- 2D convolution operation

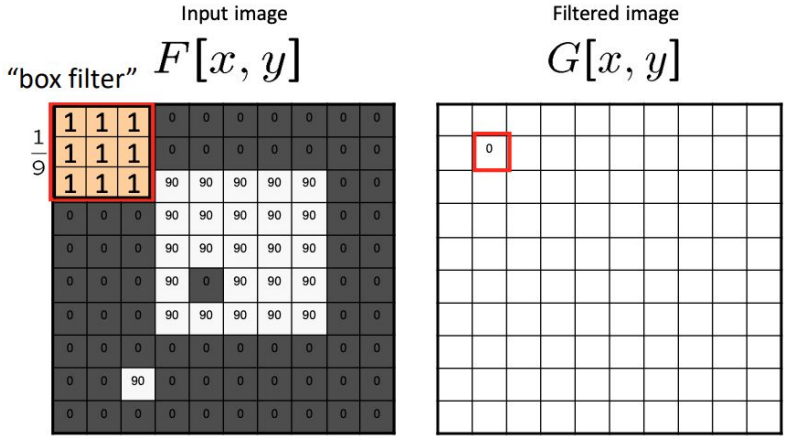


Image Filters

Moving Average in 2D

- 2D convolution operation

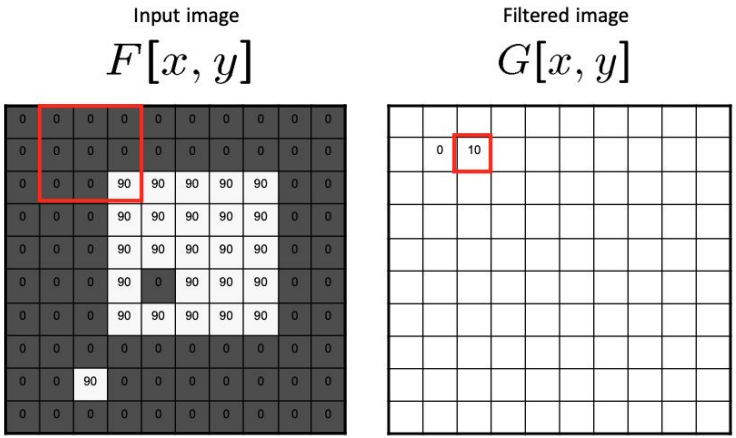


Image Filters

Moving Average in 2D

- 2D convolution operation
- Practical application (smoothing)

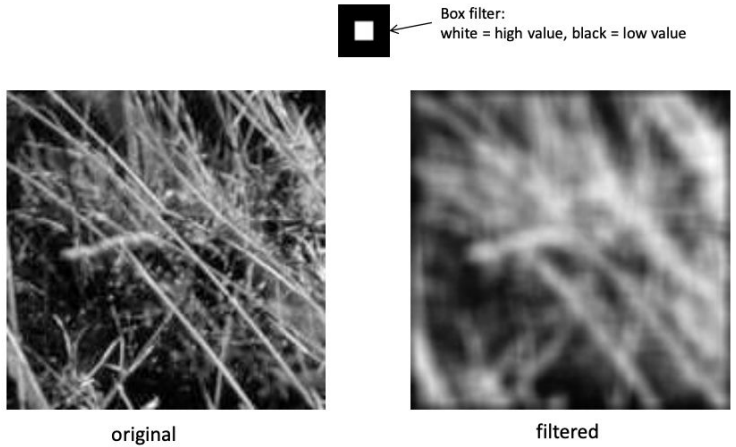


Image Filters

High Pass Filter

“If you take an image and blur it you only keep the "low frequencies". High pass makes the opposite, it only lets the "high frequencies" pass, or what most people call "**the details**". Any image can be deconstructed into these two components.”[ref]

−1	−1	−1
−1	8	−1
−1	−1	−1

Note: Only for demonstration

Image Filters

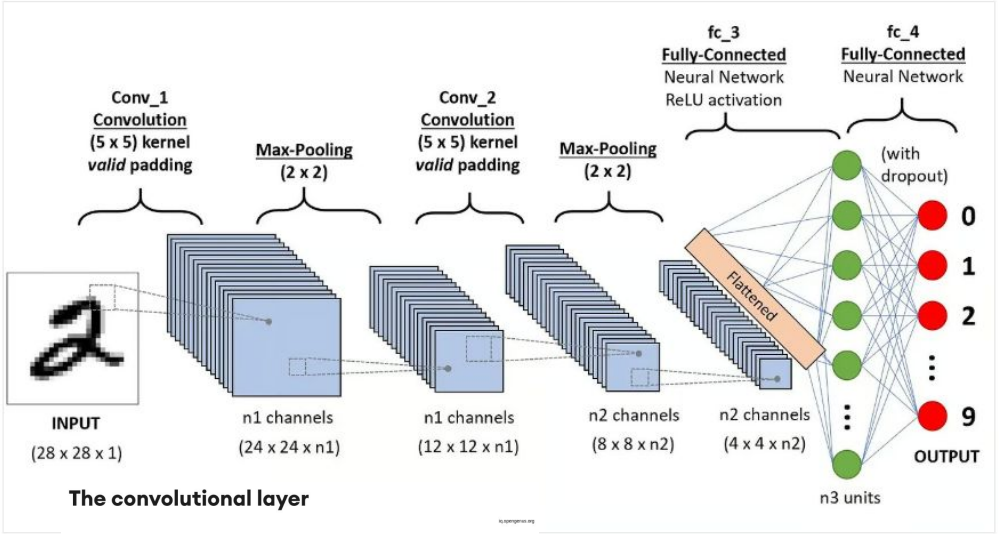
Other Filters:

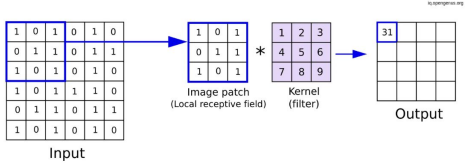
- Essentially these are convolution functions, we also call as convolution kernels

Prewitt filter $\mathbf{G_x} = \begin{bmatrix} -1 & 0 & +1 \\ -1 & 0 & +1 \\ -1 & 0 & +1 \end{bmatrix}$ and $\mathbf{G_y} = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ +1 & +1 & +1 \end{bmatrix}$

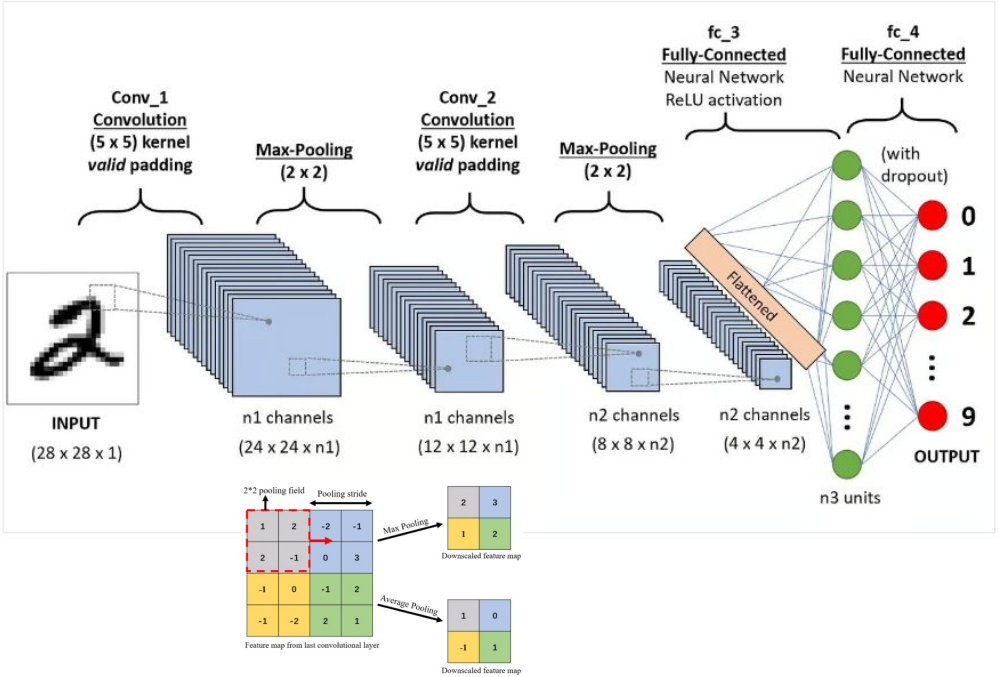
Sobel filter $\mathbf{G_x} = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix}$ and $\mathbf{G_y} = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix}$

Convolutional NNs





Convolutional NNs



Coding!

Image classification (Convolutional Neural Network)