

Neural Networks

Vishal Patel

Spring 2025

Feature Engineering

CUSTOMER ID	PURCHASE DATE
1001	02-12-2015:05:20:39
1001	05-13-2015:12:18:09
1001	12-20-2016:00:15:59
1002	01-19-2014:04:28:54
1003	01-12-2015:09:20:36
1003	05-31-2015:10:10:02
...	...



CUSTOMER ID	x_1	x_2	...	x_j
1001
1002
1003
...

1. Number of transactions (Frequency)
2. Days since the last transaction (Recency)
3. Days since the earliest transaction (Tenure)
4. Avg. days between transaction
5. # of transactions during weekends
6. % of transactions during weekends
7. # of transactions by day-part (breakfast, lunch, etc.)
8. % of transactions by day-part
9. Days since last transaction / Avg. days between transactions
10. ...

Domain
Knowledge

Create
Features

Train
Models

Training Data



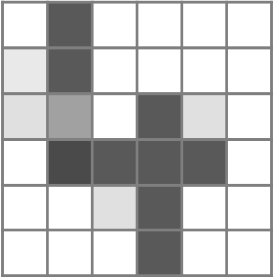
~~Domain Knowledge~~

~~Create Features~~

Train Models

Feature Engineering?

Training Data



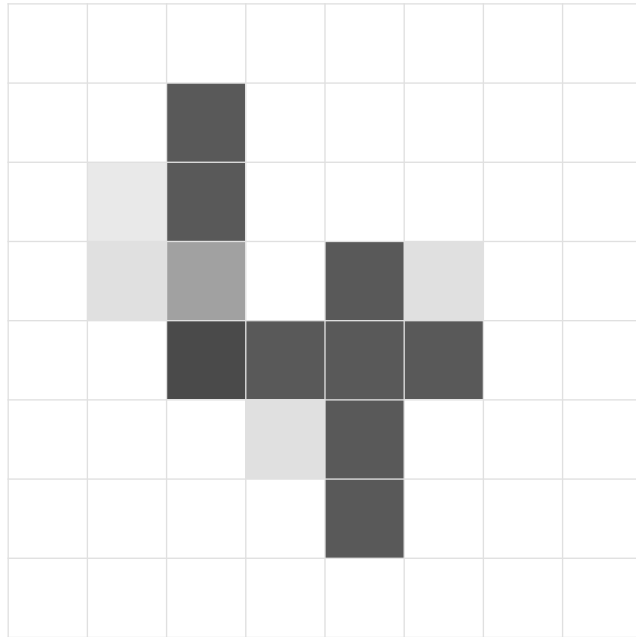
x_{11}	x_{12}	x_{13}	x_{1j}
x_{21}	x_{22}	x_{23}	x_{2j}
x_{31}	x_{32}	x_{33}	x_{3j}
.
.
.
x_{n1}	x_{n2}	x_{n3}	x_{nj}

y_1
y_2
y_3
.
.
.
y_n

$$j = 36$$

$$y_1 = 4$$

Digit Recognition Program



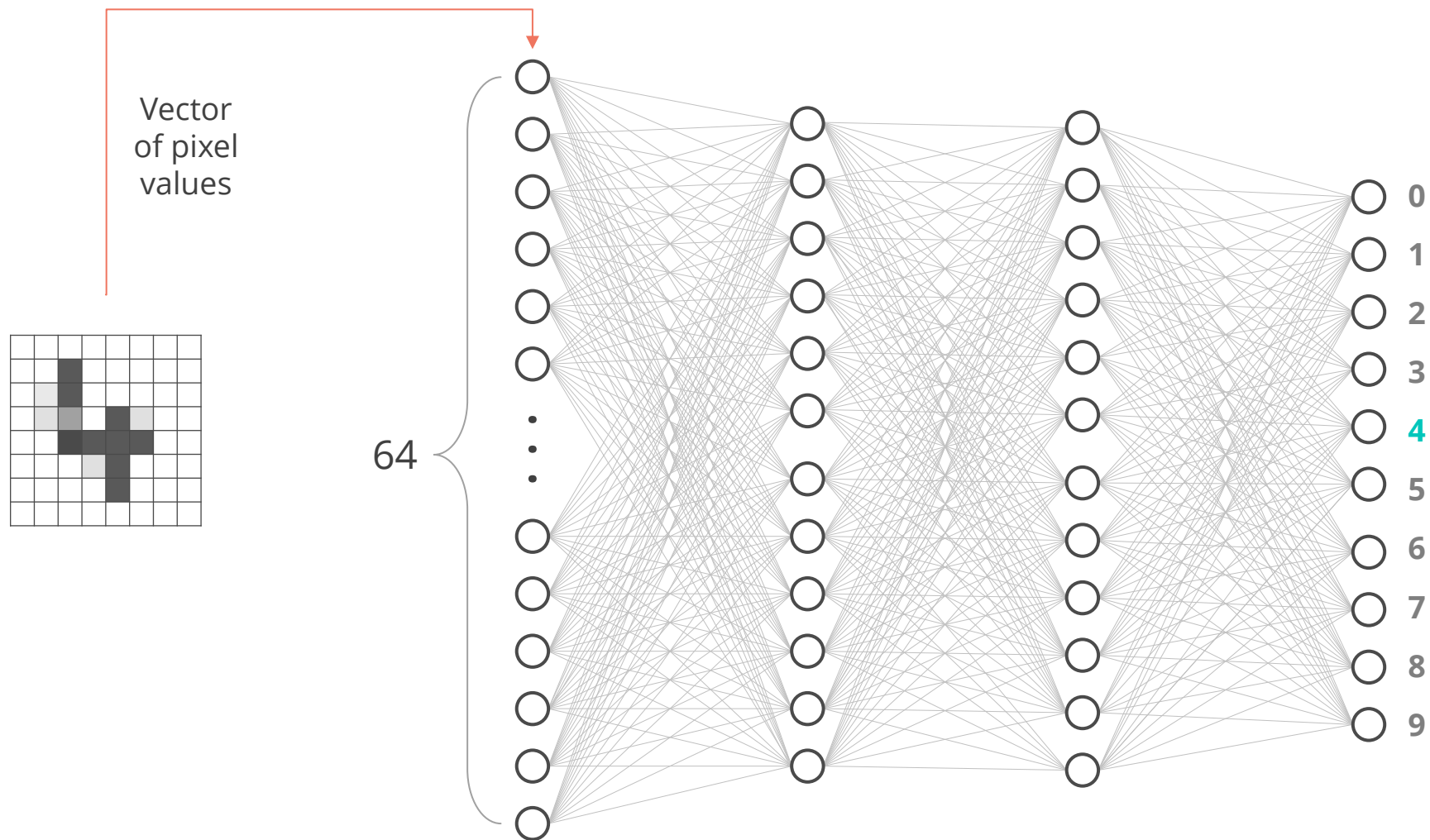
Input (8x8 grid)



0
1
2
3
4
5
6
7
8
9

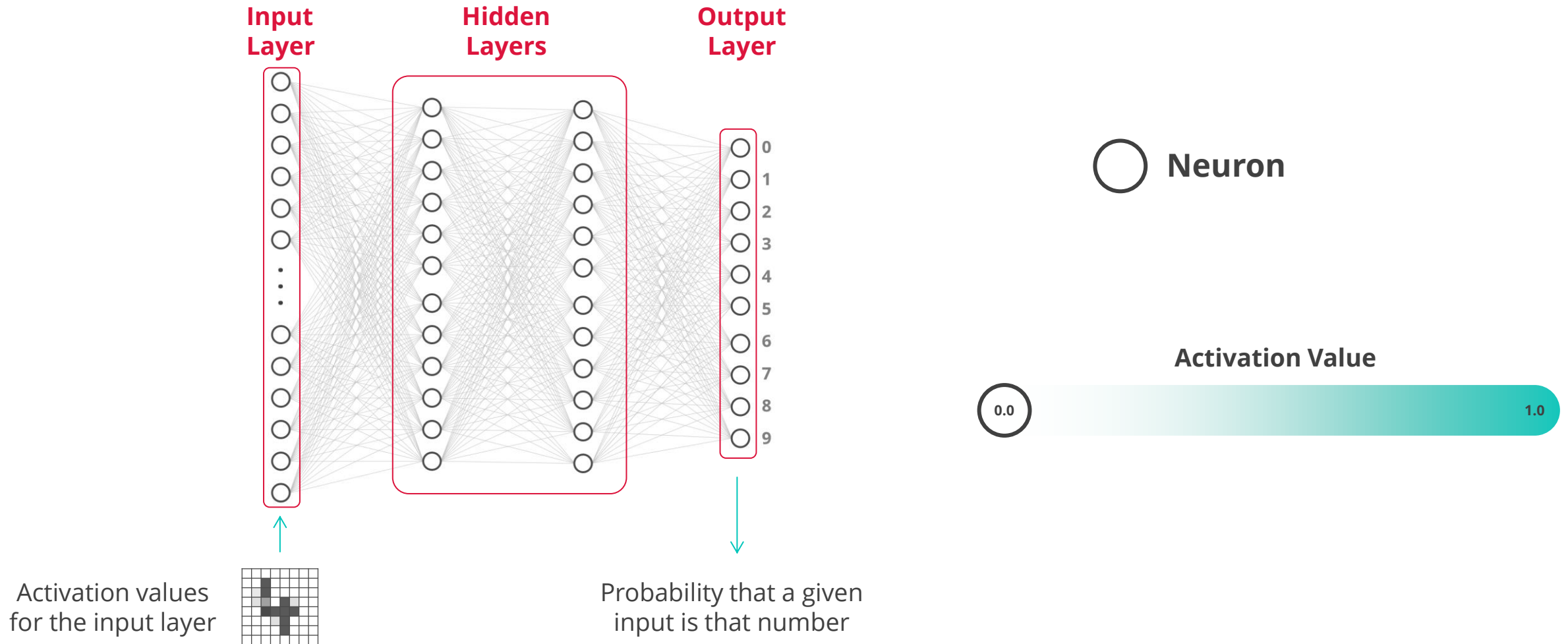
Output

Neural Network

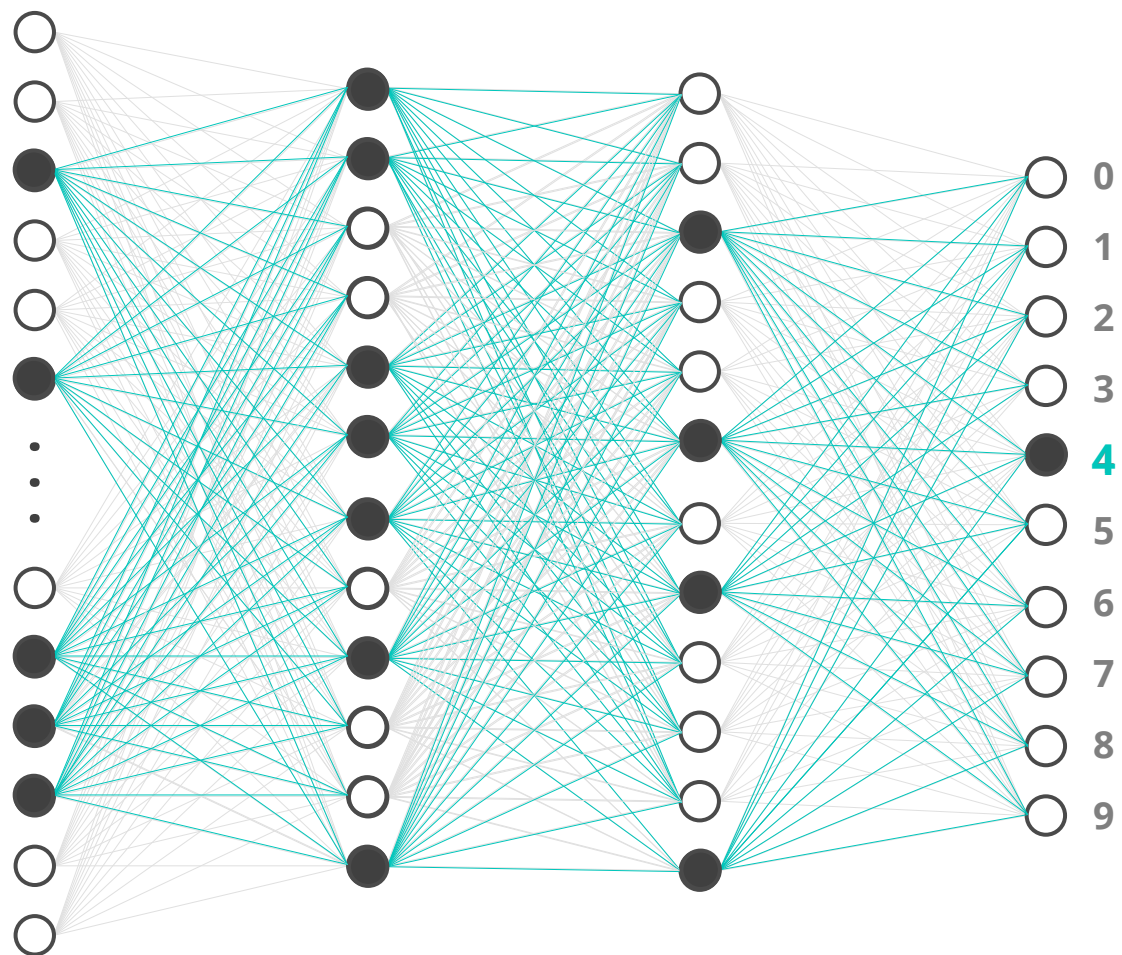
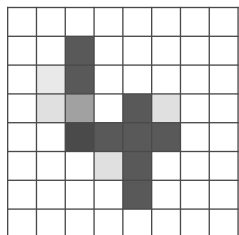


Multilayer Perceptron

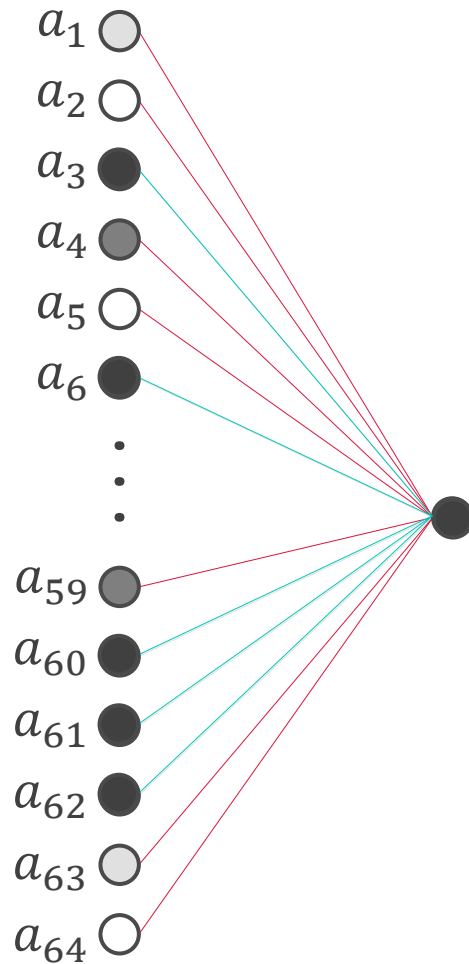
Terminology



The Layers



Activation Function



$$w_1 = -0.17$$

$$w_2 = -2.09$$

$$w_3 = +3.25$$

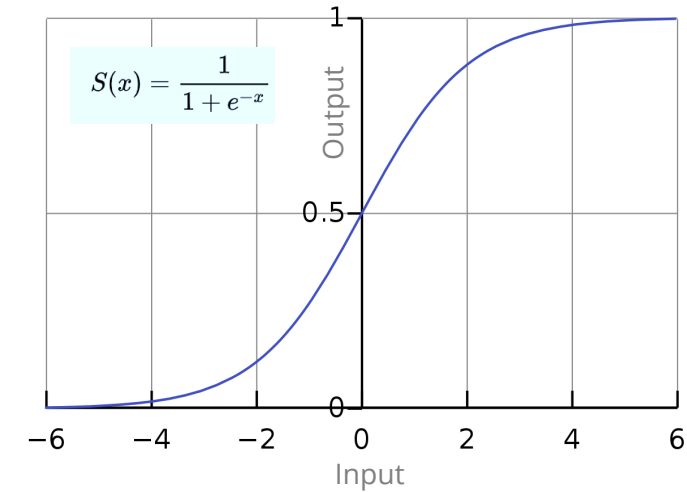
$$w_4 = -0.05$$

$$w_5 = -2.99$$

$$w_6 = +1.11$$

\vdots

Sigmoid function



The logistic curve

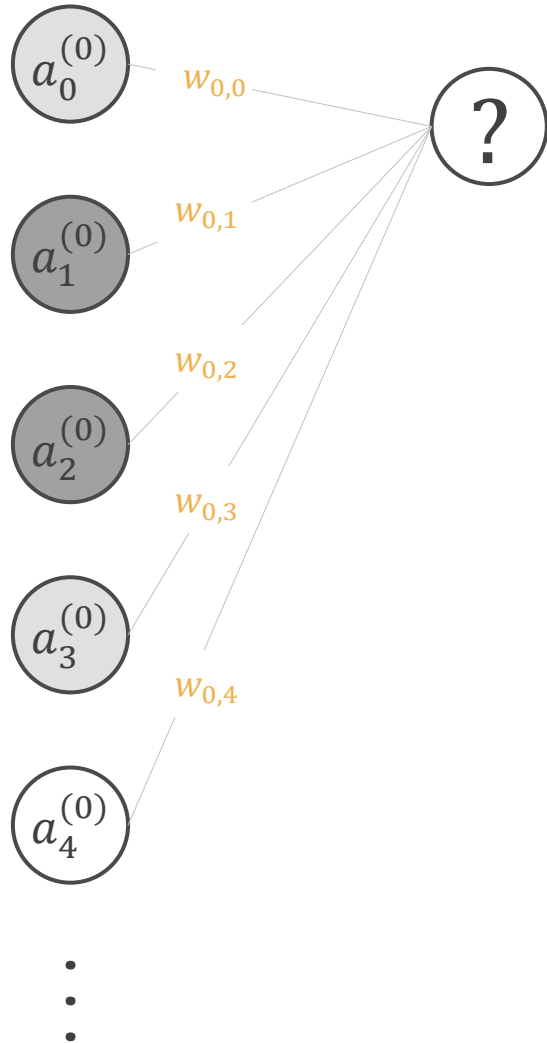
Activation value of a neuron:

$$\sigma(w_1 a_1 + w_2 a_2 + w_3 a_3 + \dots + w_n a_n + bias)$$

↑
Sigmoid function

Note: A generalized sigmoid function, called softmax, is used for the output layer.

Activation Function



$$a_0^{(1)} = \sigma \left(w_{0,0} a_0^{(0)} + w_{0,1} a_1^{(0)} + \dots + w_{0,n} a_n^{(0)} + b_0 \right)$$

$$\sigma \left(\begin{bmatrix} w_{0,0} & w_{0,1} & \dots & w_{0,n} \\ w_{1,0} & w_{1,1} & \dots & w_{1,n} \\ \dots & \dots & \dots & \dots \\ w_{k,0} & w_{k,1} & \dots & w_{k,n} \end{bmatrix} \begin{bmatrix} a_0^{(0)} \\ a_1^{(0)} \\ \dots \\ a_n^{(0)} \end{bmatrix} + \begin{bmatrix} b_0 \\ b_1 \\ \dots \\ b_n \end{bmatrix} \right)$$

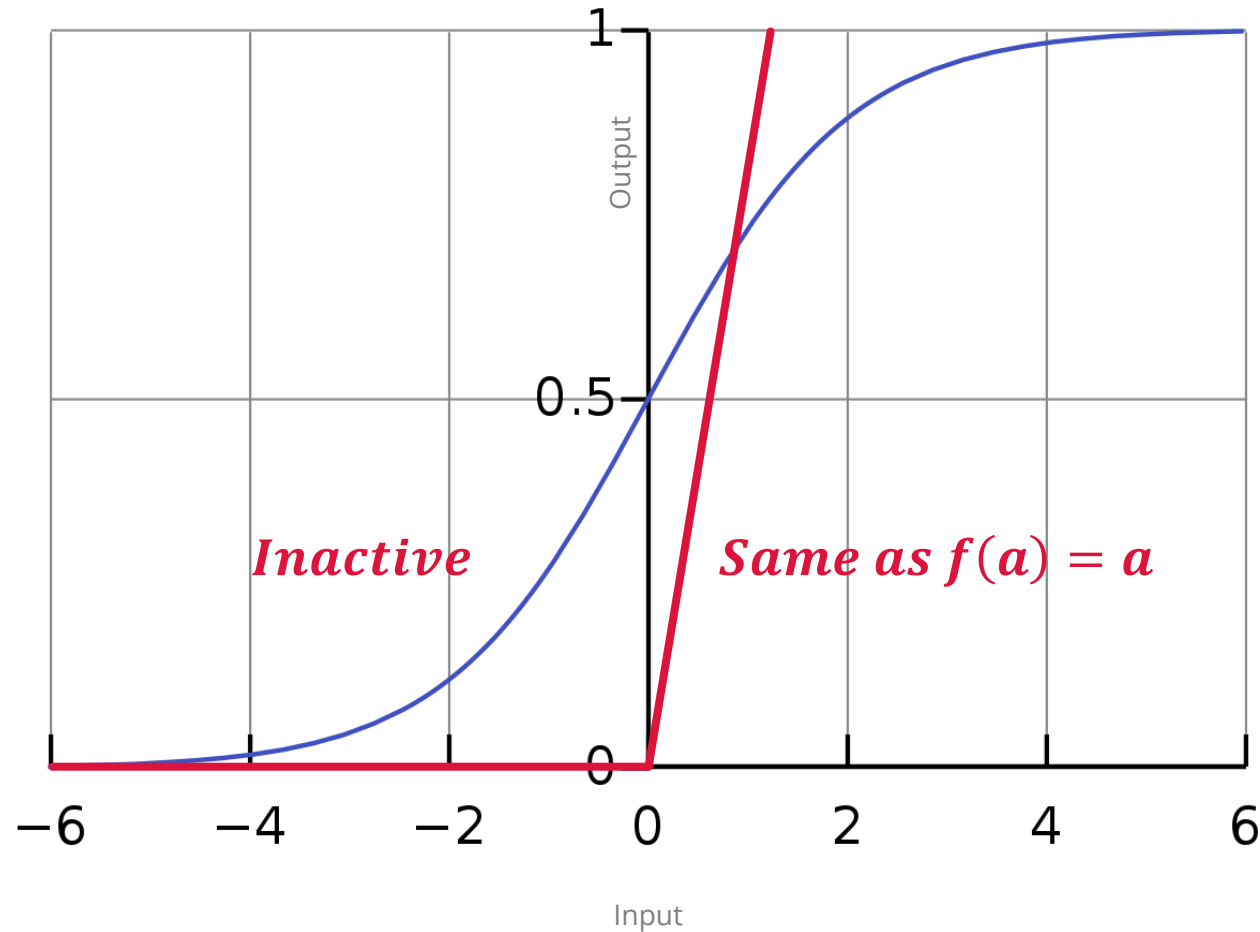
$$a^{(1)} = \sigma(Wa^{(0)} + b)$$

Rectified Linear Unit

Sigmoid

$$\sigma(a) = \frac{1}{1 + e^{-a}}$$

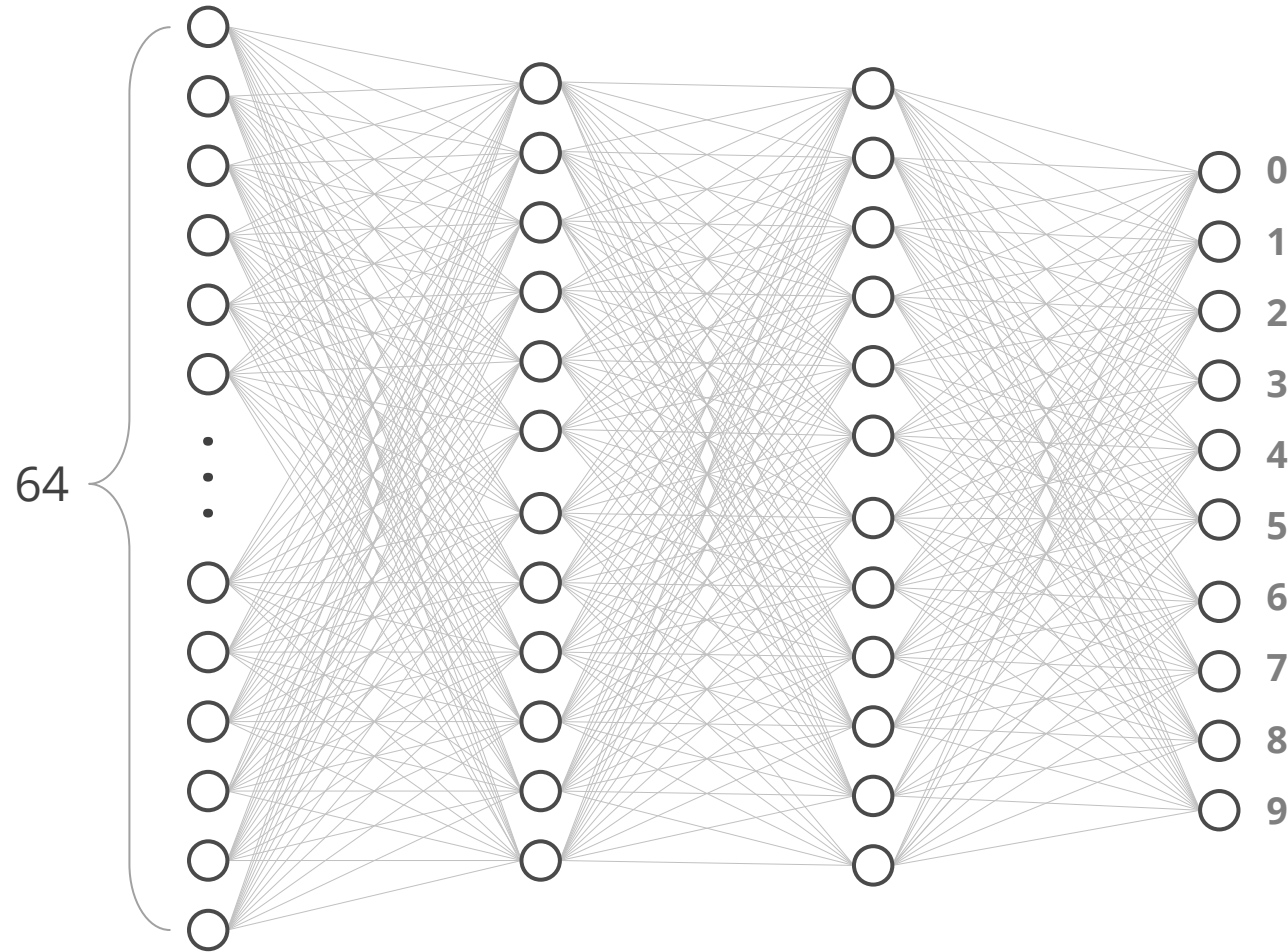
Slow learner



ReLU

$$ReLU(a) = \max(0, a)$$

Neural Network: Model Parameters



$$64 * 12 + 12 * 12 + 12 * 10$$

weights

$$12 + 12 + 10$$

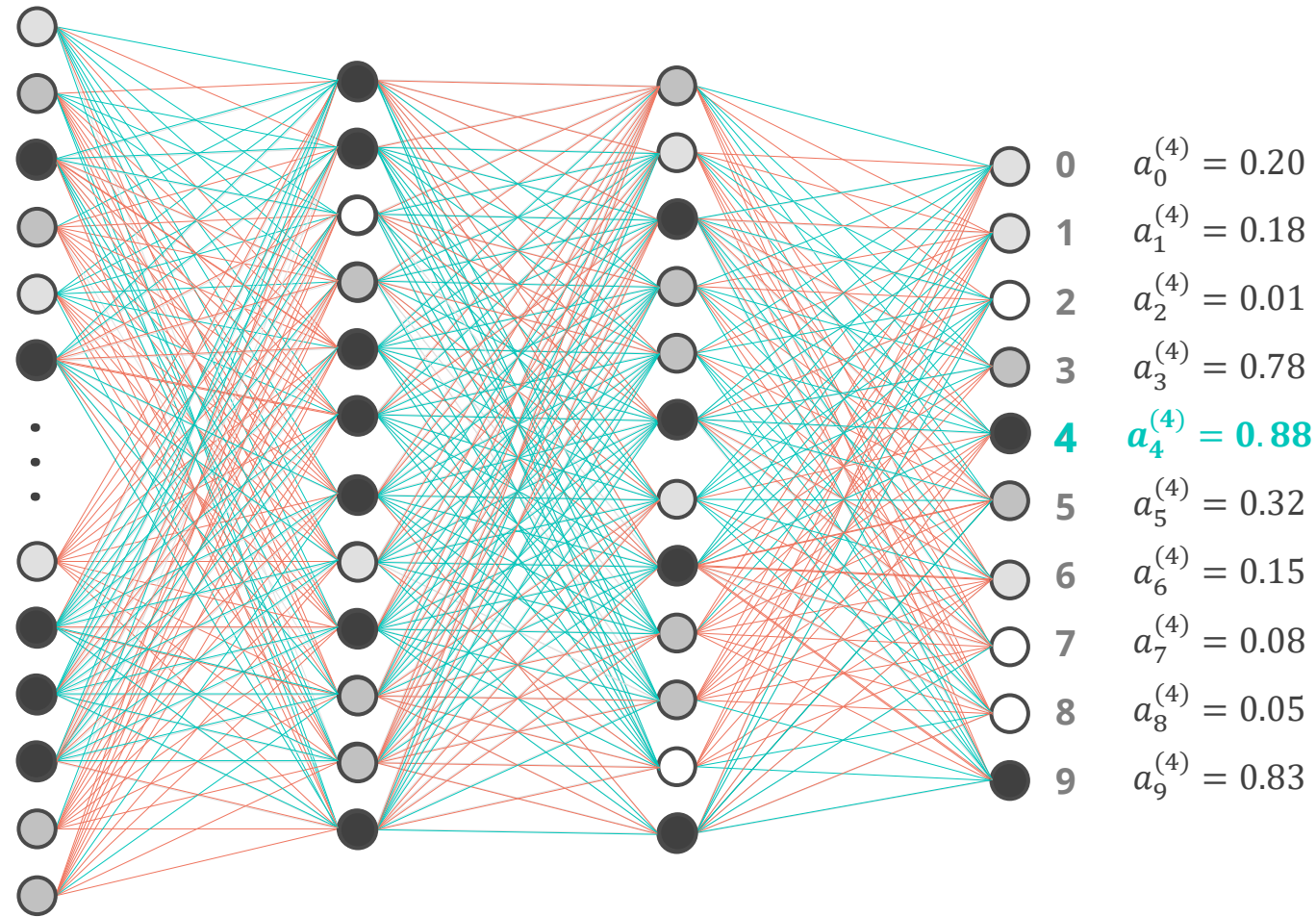
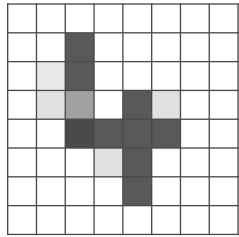
biases

4,394

GPT-4 has ~1.8 trillion parameters!

Learning → Finding the right weights and biases

The Cost Function



Weights and biases

Input

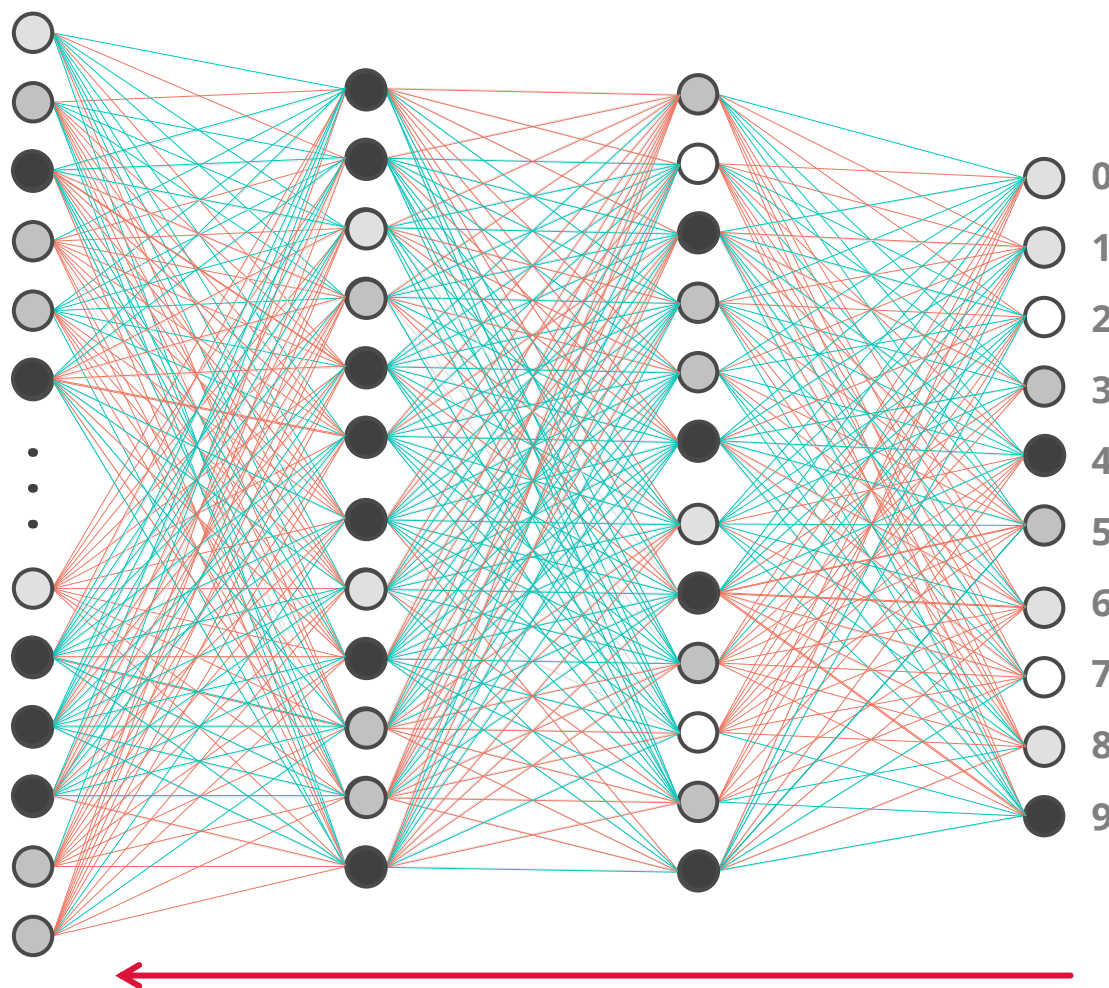
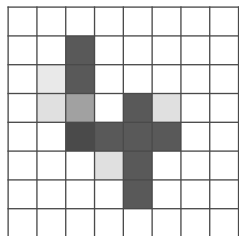
Training data

Cost

Output

Mean Squared Error (Quadratic cost),
Cross-entropy cost, etc.

Backpropagation



Actual Output	Desired Output	Adj.
0.20	0	-
0.18	0	-
0.01	0	-
0.78	0	-
0.88	1	+
0.32	0	-
0.15	0	-
0.08	0	-
0.05	0	-
0.83	0	-

Resources

3Blue1Brown series S3 • E1

But what is a Neural Network? | Deep learning, chapter 1

8,866,833 views • Oct 5, 2017

 227K  2K  SHARE  SAVE ...

Introduction | Deep Learning Tutorial 1 (Tensorflow Tutorial, Keras & Python)

56,034 views • Premiered Jul 11, 2020



codebasics ✓
277K subscribers

TensorFlow, Keras and deep learning, without a PhD

MIT 6.S191 (2020): Convolutional Neural Networks




195,647 views • Premiered Feb 21, 2020



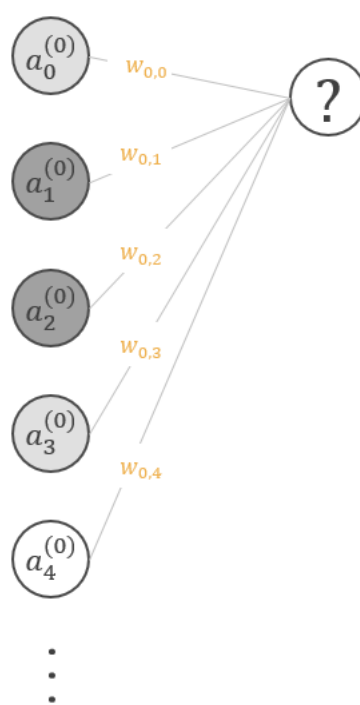
Alexander Amini
102K subscribers

Deep Learning: CS 182
Spring 2021

44 videos • 14,902 views • Last updated on Mar 13, 2021

   ...

Lectures for UC Berkeley CS 182: Deep Learning.



$$a_0^{(1)} = \sigma \left(w_{0,0}a_0^{(0)} + w_{0,1}a_1^{(0)} + \dots + w_{0,n}a_n^{(0)} + b_0 \right)$$

$$\sigma \left(\begin{bmatrix} w_{0,0} & w_{0,1} & \dots & w_{0,n} \\ w_{1,0} & w_{1,1} & \dots & w_{1,n} \\ \dots & \dots & \dots & \dots \\ w_{k,0} & w_{k,1} & \dots & w_{k,n} \end{bmatrix} \begin{bmatrix} a_0^{(0)} \\ a_1^{(0)} \\ \dots \\ a_n^{(0)} \end{bmatrix} + \begin{bmatrix} b_0 \\ b_1 \\ \dots \\ b_n \end{bmatrix} \right)$$

$$\mathbf{a}^{(1)} = \sigma(\mathbf{W}\mathbf{a}^{(0)} + \mathbf{b})$$



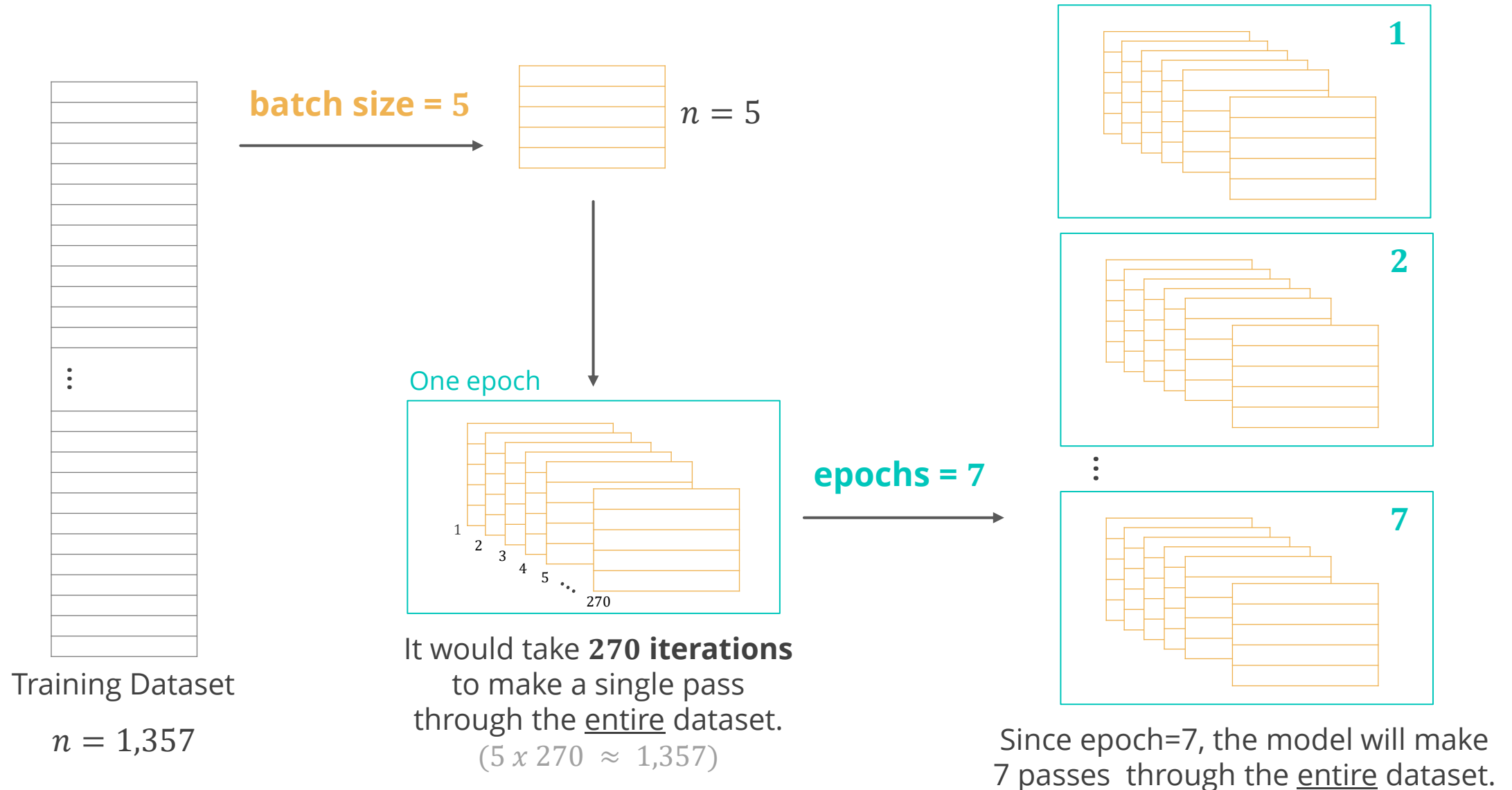
```
import tensorflow as tf
```


Neural Network Tutorial

`13_digits_recognition_model.ipynb`

`14_intro_neural_net.ipynb`

batch, iterations, and epoch



<https://www.youtube.com/watch?v=D-YHC8b6Hjk>

Fully Connected MLP

- Computationally expensive
- Spatial information is lost
- Sensitive to location of objects within an image



Convolutional Neural Networks (CNNs)



convolve

[kənˈvɒlv] 🔊

VERB *rare*

convolve (verb) · **convolves** (third person present) · **convolved** (past tense) · **convolved** (past participle) · **convolving** (present participle)

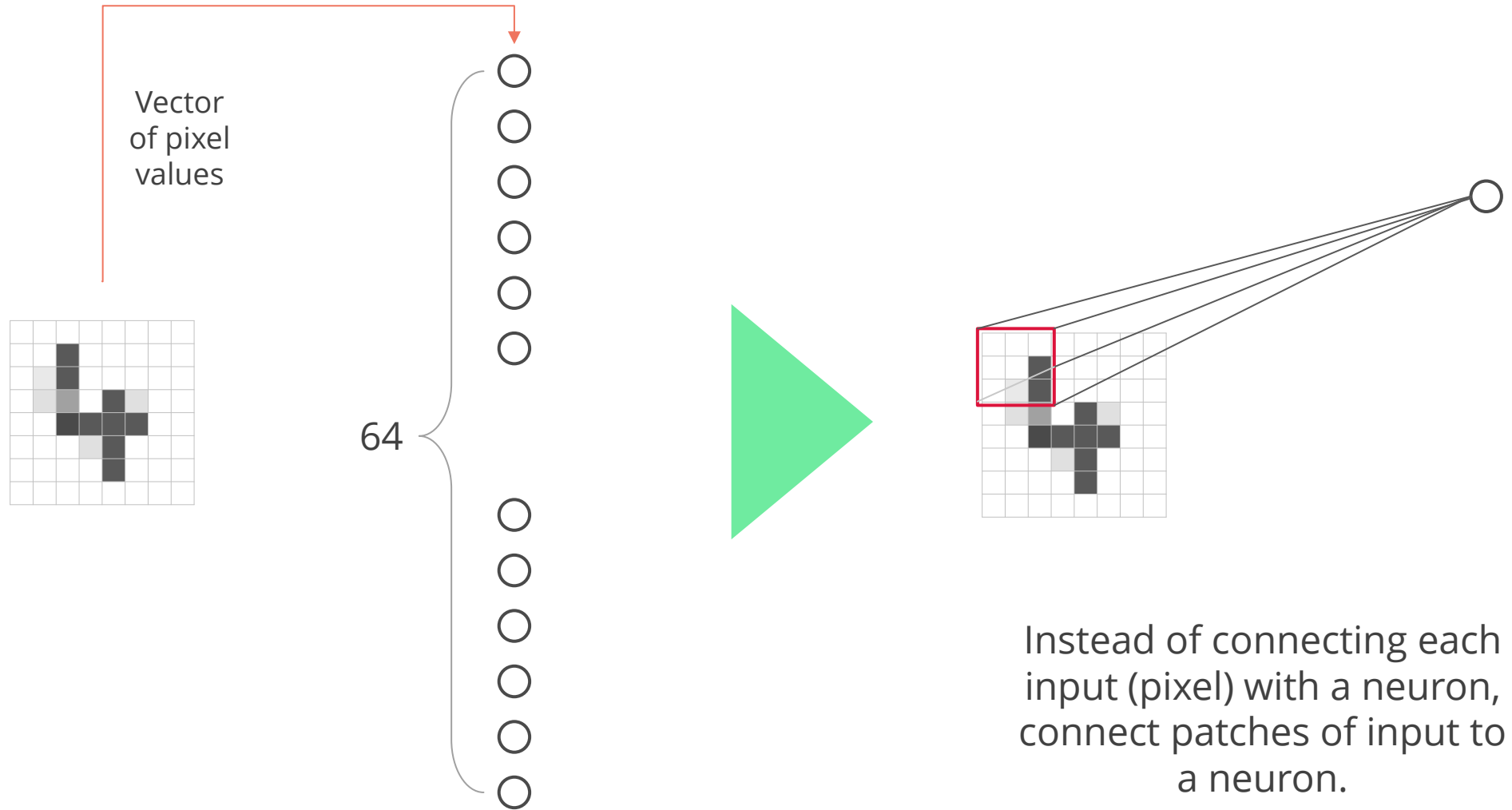
roll or coil together; entwine.

- *mathematics*
combine (one function or series) with another by forming their convolution.

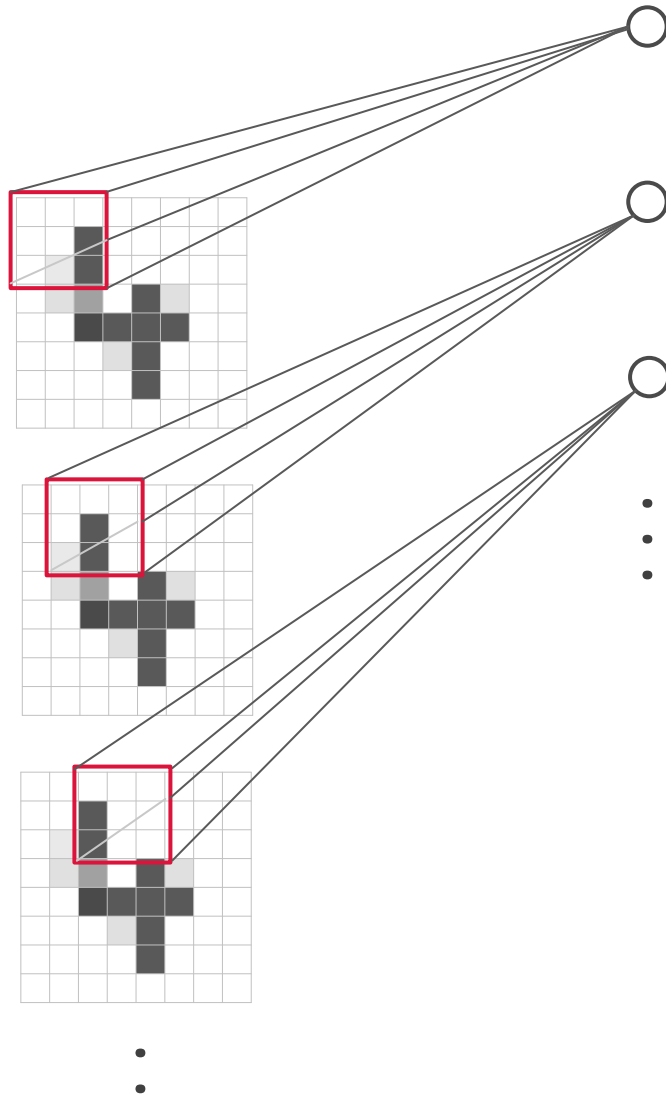
ORIGIN

late 16th century (in the sense 'enclose in folds'): from Latin *convolvere* 'roll together', from *con-* 'together' + *volvere* 'roll'.

Spatial Structure



Convolution



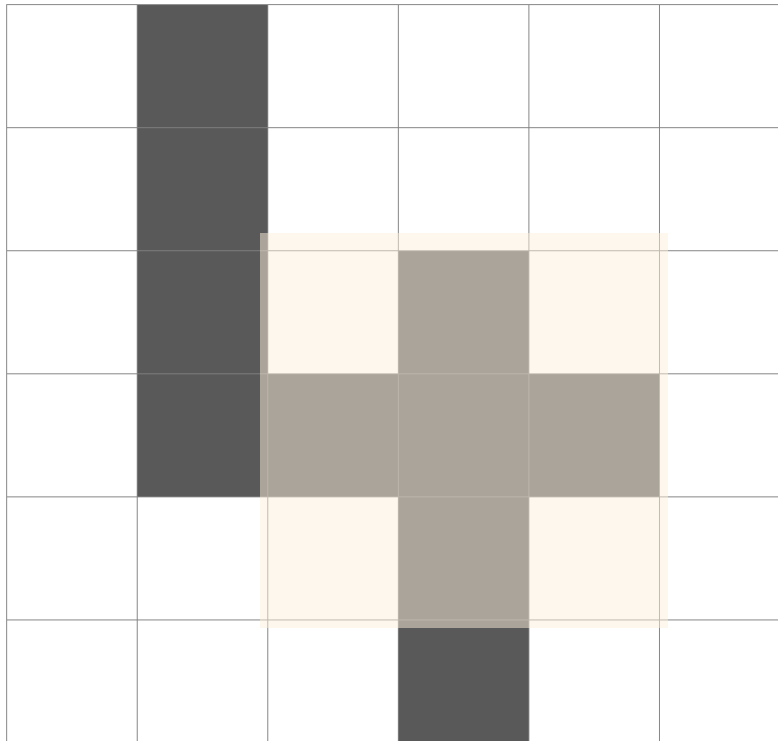
Filter = A **patch** of size 3x3 (in this example)

Instead of taking a straight sum of all pixels within a patch, a set of weights are created which are then used to take a weighted sums.

Multiple filters (i.e., set of weights) are used to extract different features.

Feature Extraction

A Filter to Detect a Feature



Let's create a filter to identify the **cross**.

A Filter to Detect a Feature

-1	1	-1	-1	-1	-1
-1	1	-1	-1	-1	-1
-1	1	-1	1	-1	-1
-1	1	1	1	1	-1
-1	-1	-1	1	-1	-1
-1	-1	-1	1	-1	-1

Input

-1	1	-1
1	1	1
-1	1	-1

Filter

A Filter to Detect a Feature

-1	1	-1	-1	-1	-1
-1	1	-1	-1	-1	-1
-1	1	-1	1	-1	-1
-1	1	1	1	1	-1
-1	-1	-1	1	-1	-1
-1	-1	-1	1	-1	-1

Input

-1	1	-1
1	1	1
-1	1	-1

Filter

5			

$$\begin{aligned} &(-1 * -1) + (1 * 1) + (-1 * -1) \\ &+ (-1 * 1) + (1 * 1) + (-1 * 1) \\ &+ (-1 * -1) + (1 * 1) + (-1 * -1) \end{aligned}$$

$$1 + 1 + 1 - 1 + 1 - 1 + 1 + 1 + 1 = 5$$

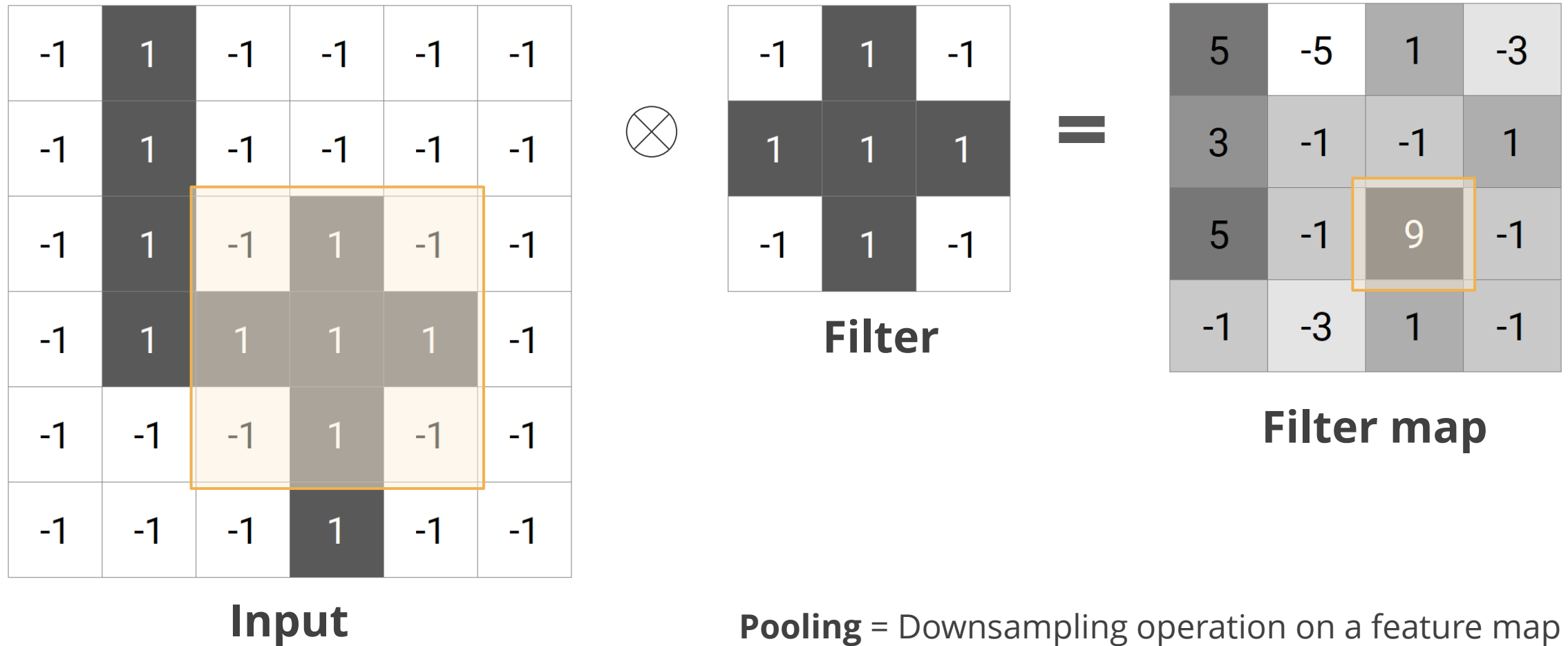
A Filter to Detect a Feature

-1	1	-1	-1	-1	-1
-1	1	-1	-1	-1	-1
-1	1	-1	1	-1	-1
-1	1	1	1	1	-1
-1	-1	-1	1	-1	-1
-1	-1	-1	1	-1	-1

-1	1	-1
1	1	1
-1	1	-1

5	-5		

A Filter to Detect a Feature



CNN Tutorial

15_image_classification_cnn.ipynb