



Neural Network and Its Applications

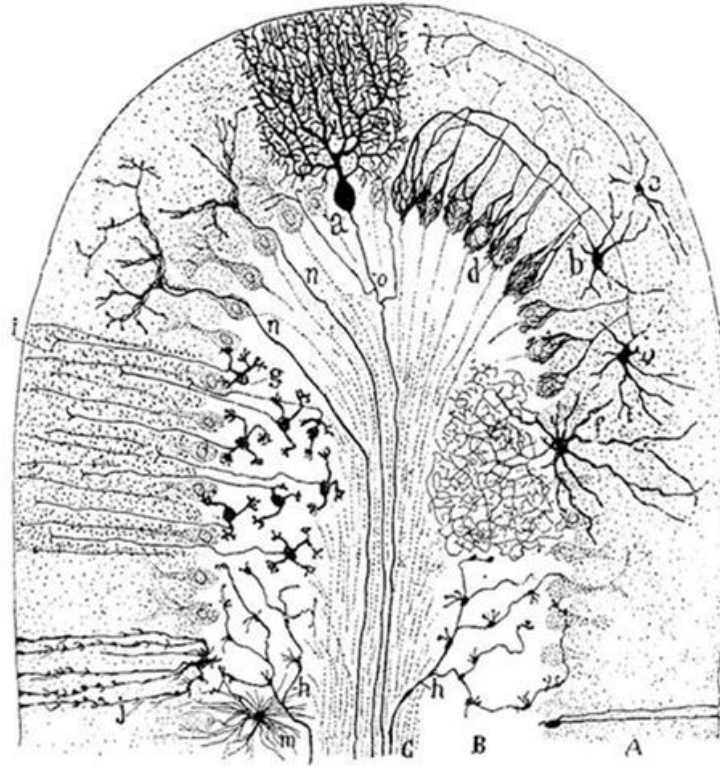
Albert C. Yang, M.D., Ph.D.

Institutes of Brain Science / Digital Medicine Center
National Yang-Ming University

May 21, 2020

accyang@gmail.com

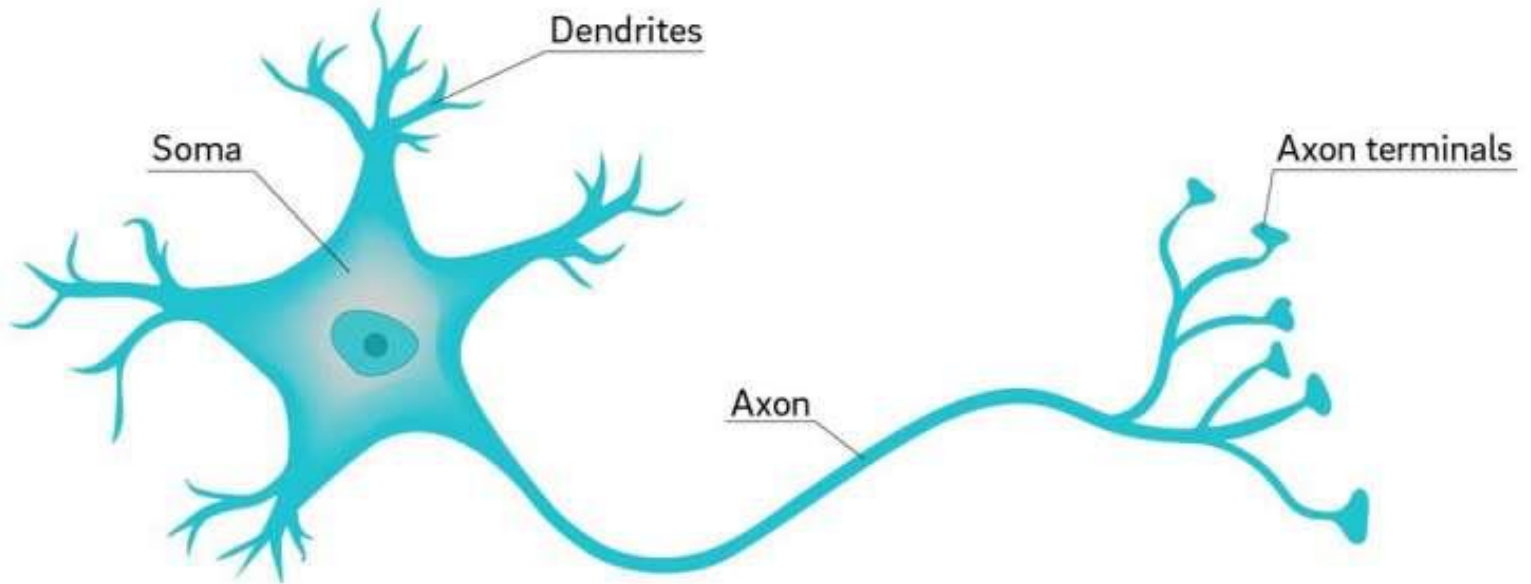
Neural Network



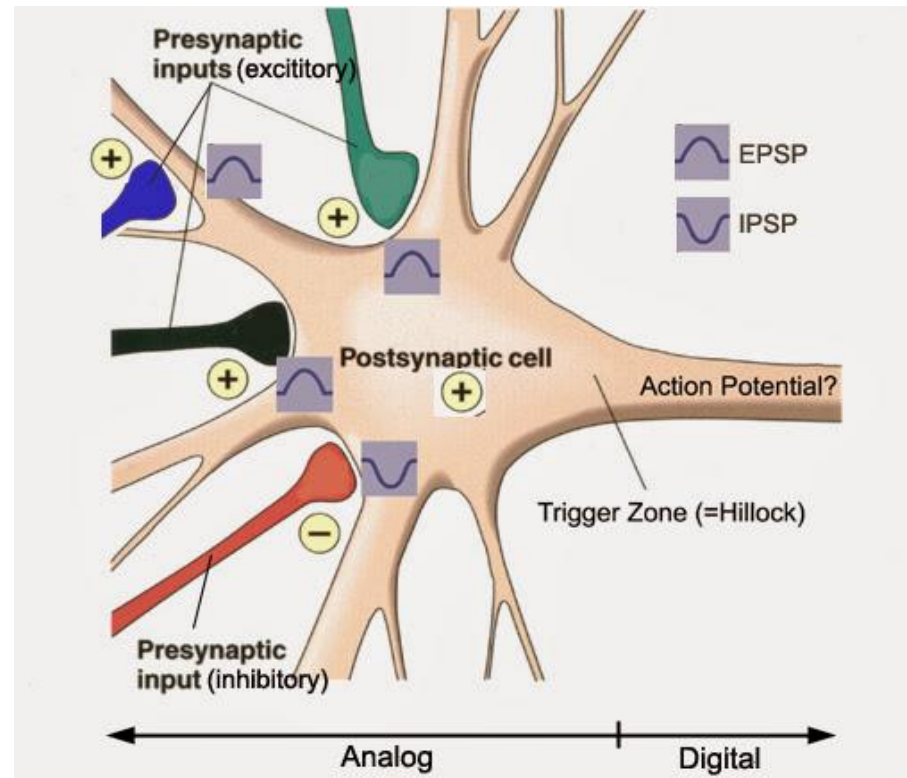
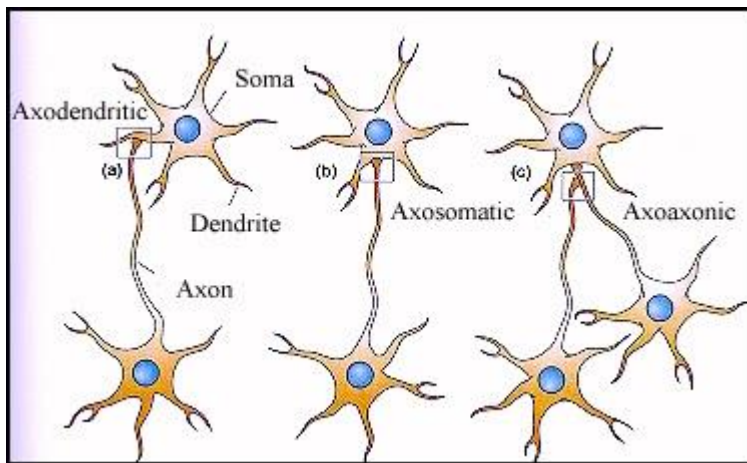
Santiago Ramón y Cajal
1877 AC

Neuron

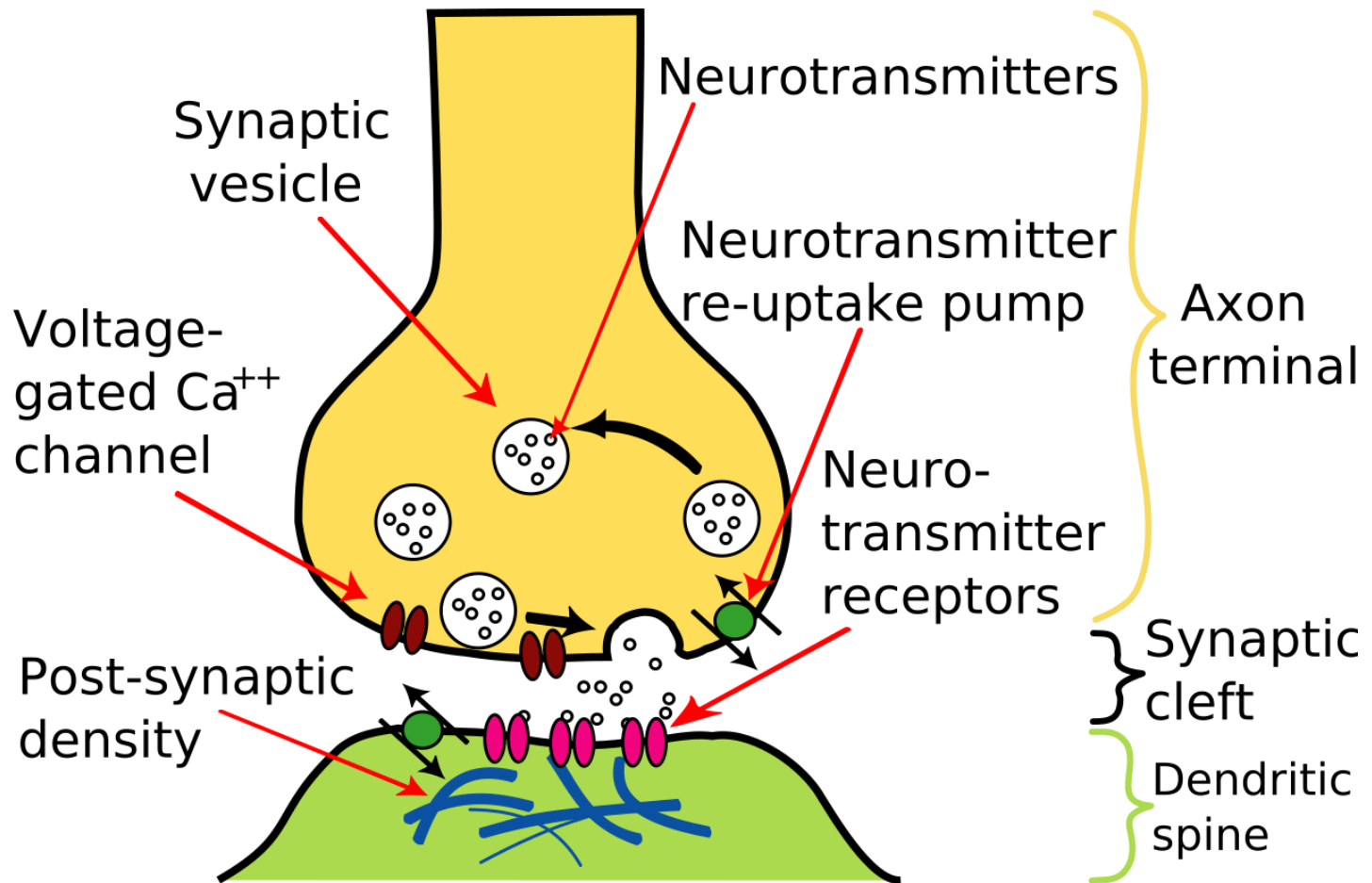
Neuron



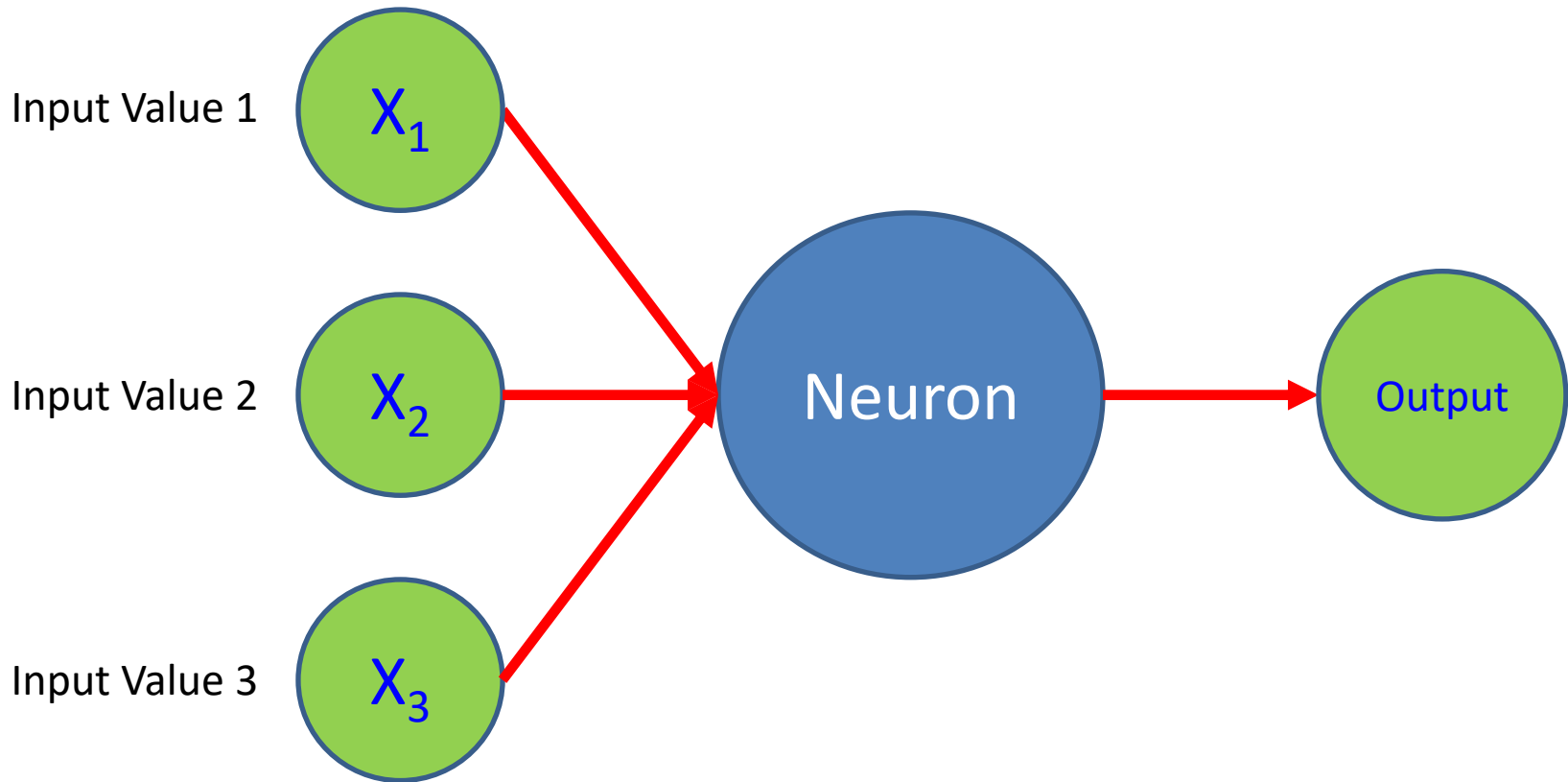
Neural Signal Transmission



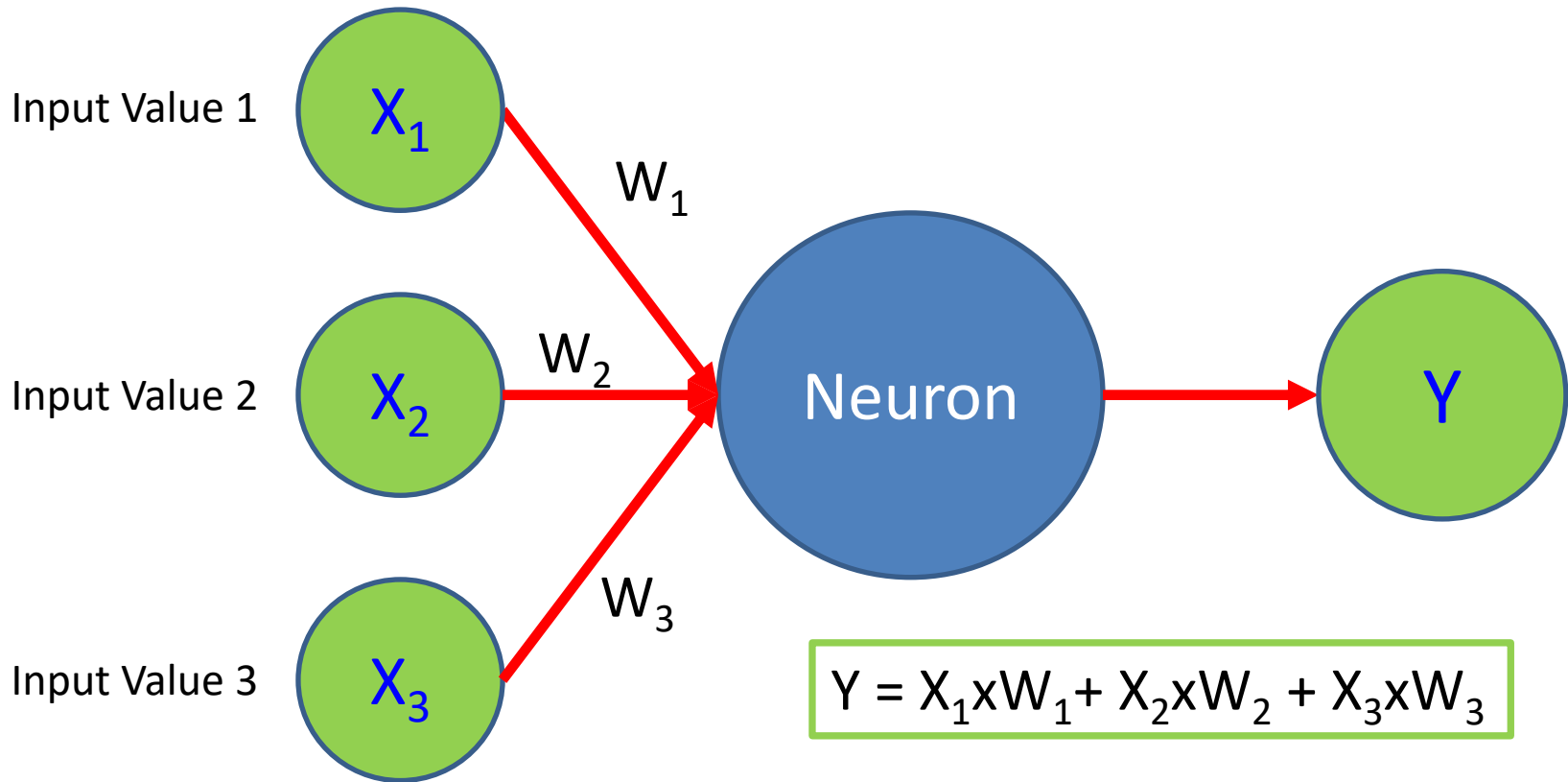
Synapse



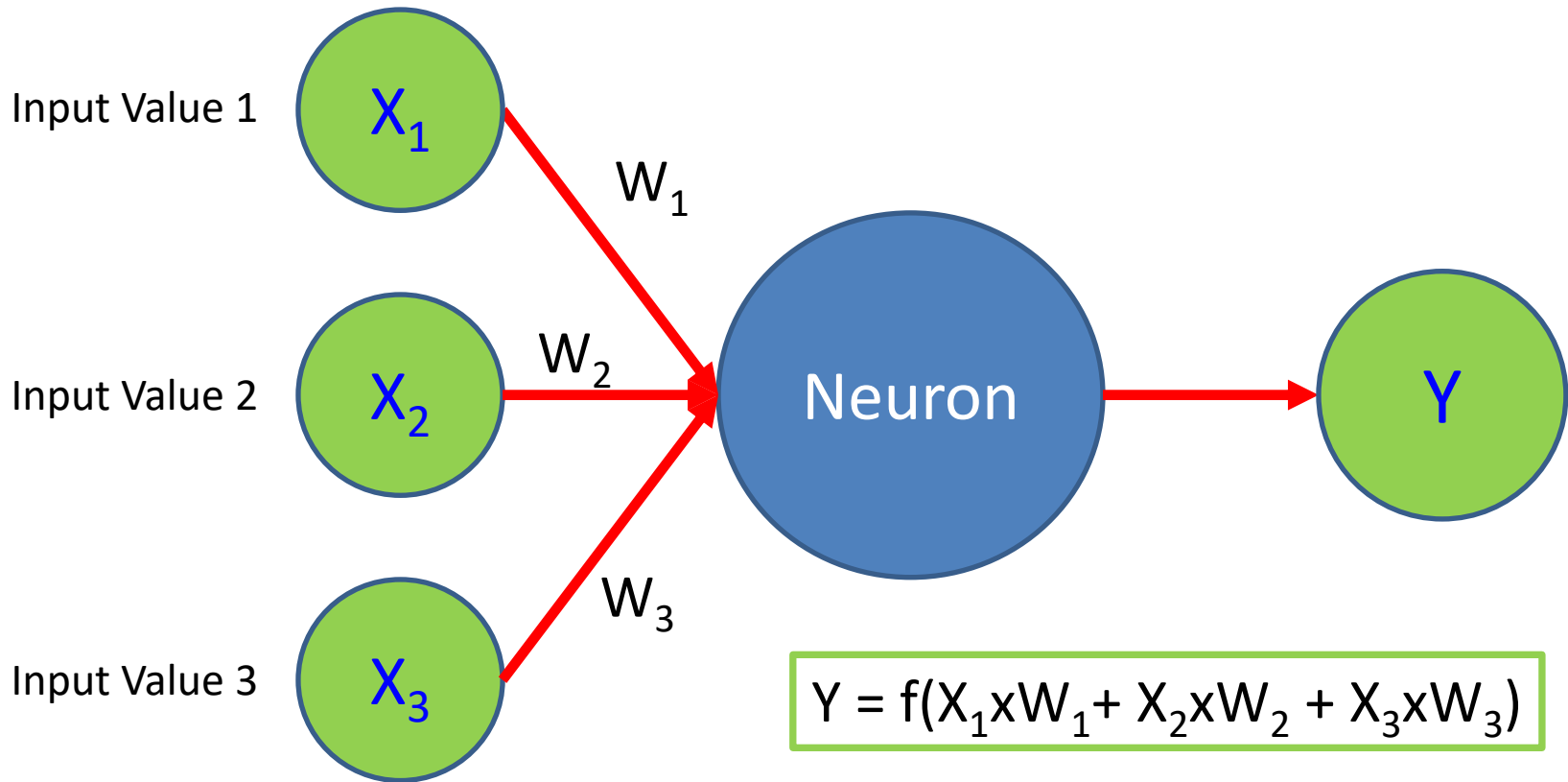
Neuron as Information Processor



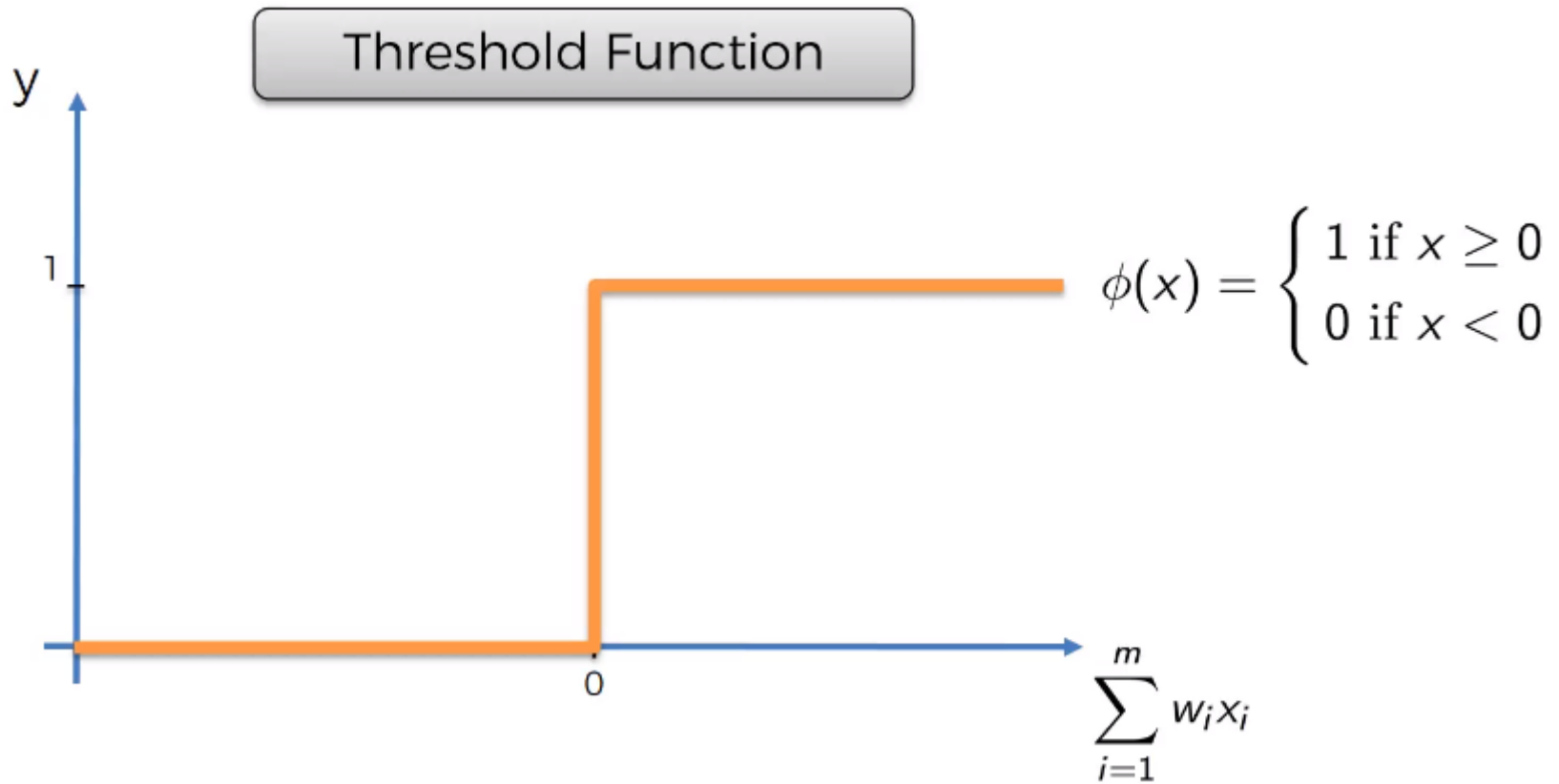
Neuron as Information Processor



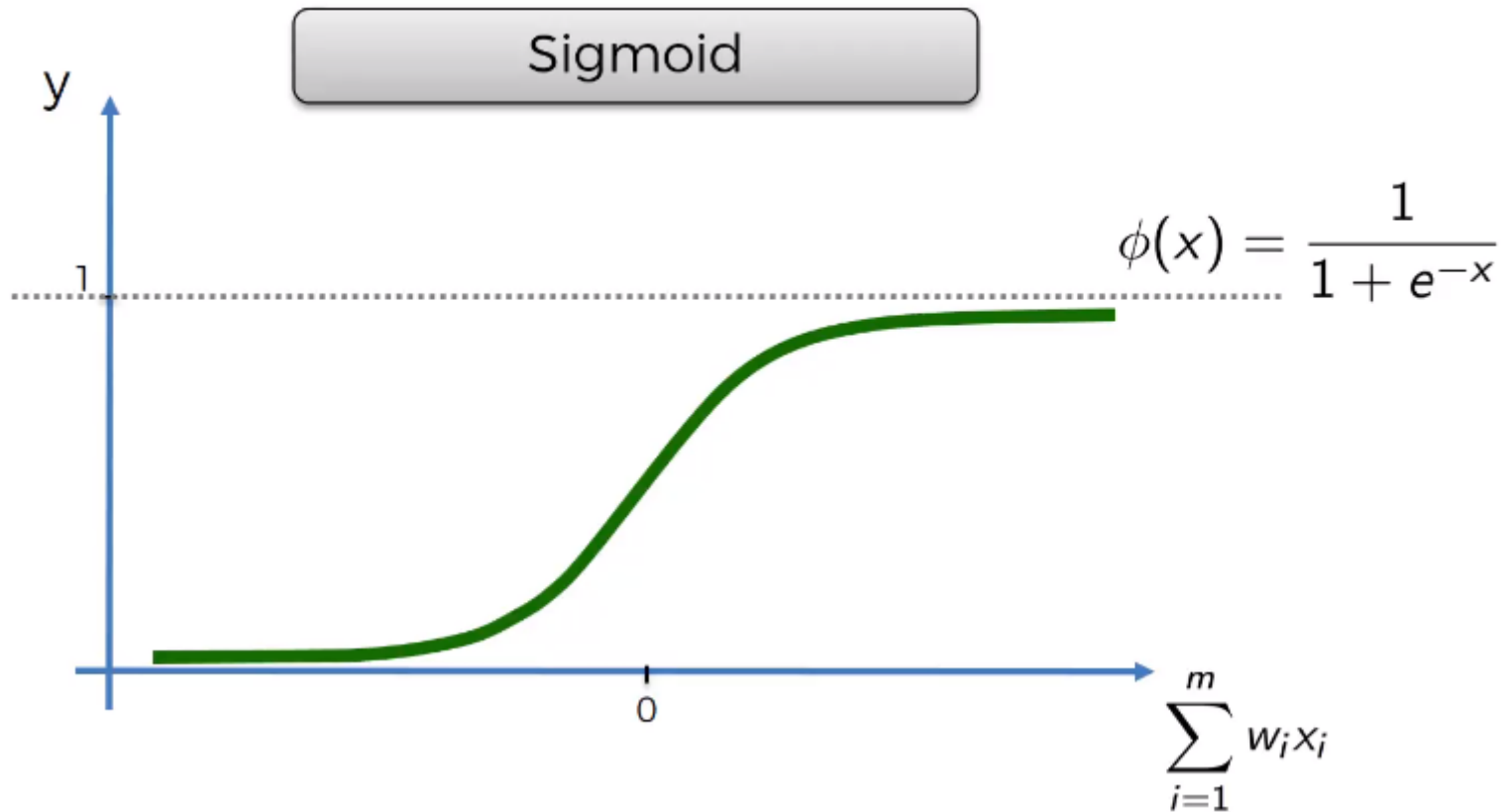
Neuron Activation – All or None



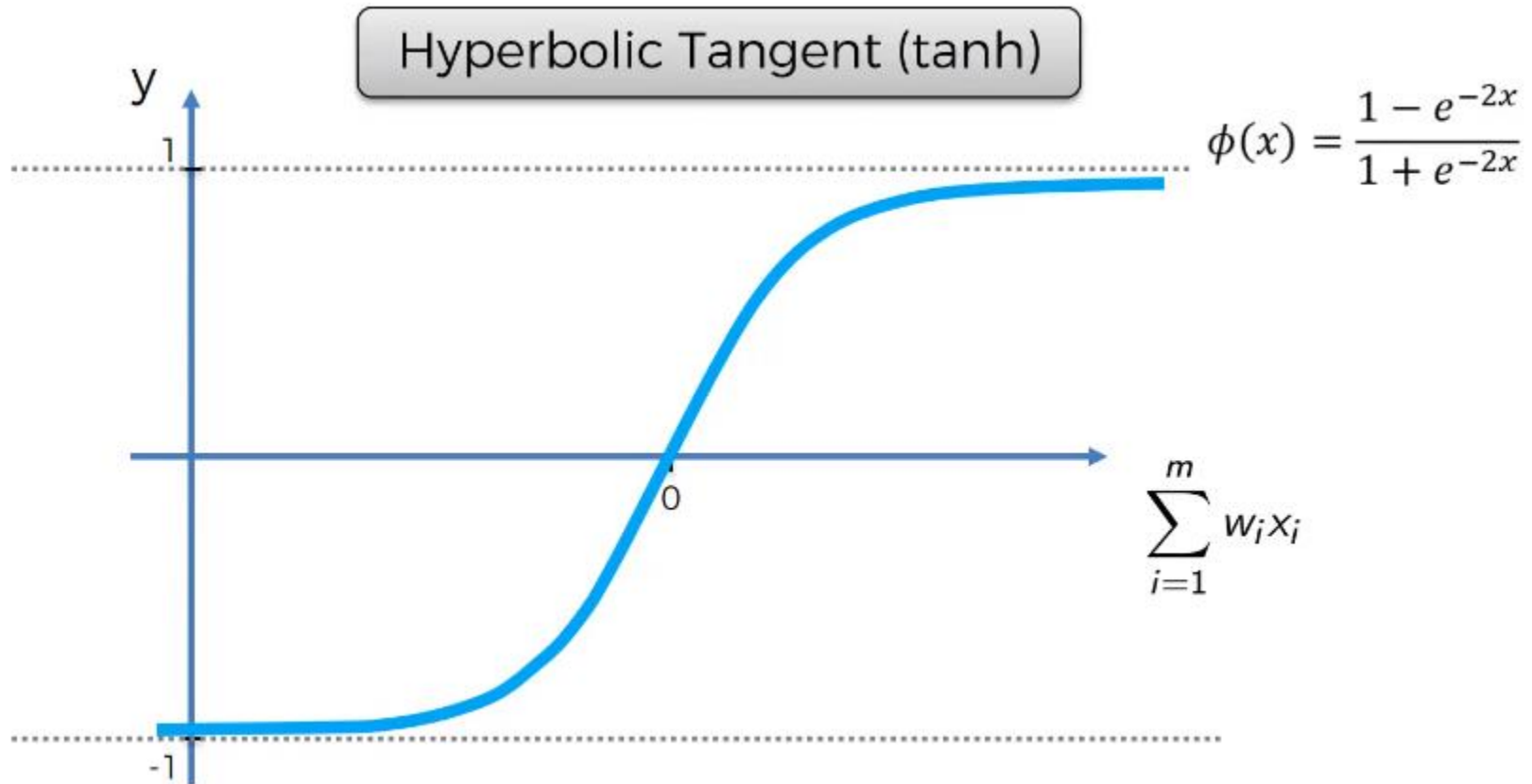
Neuron Activation



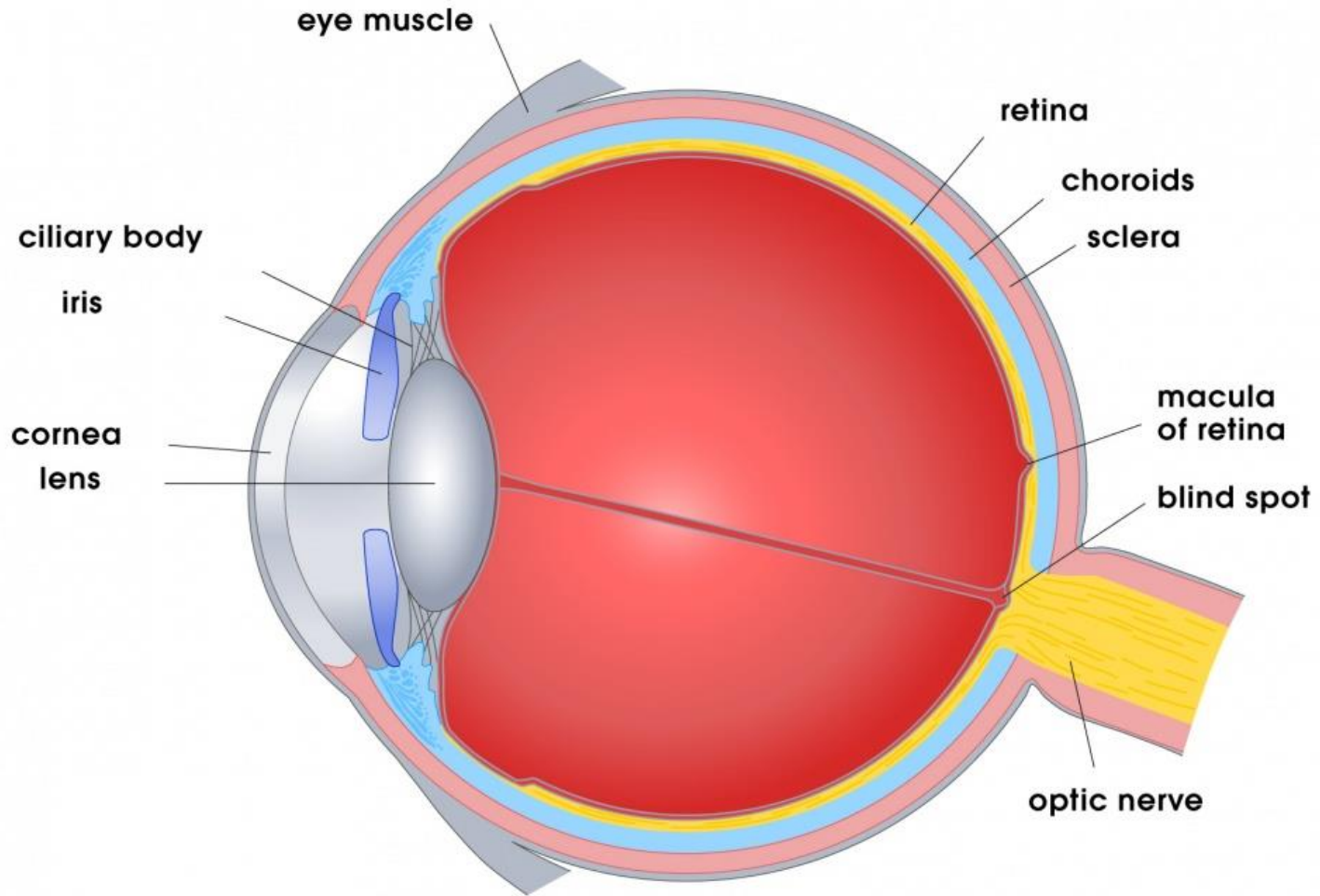
Neuron Activation



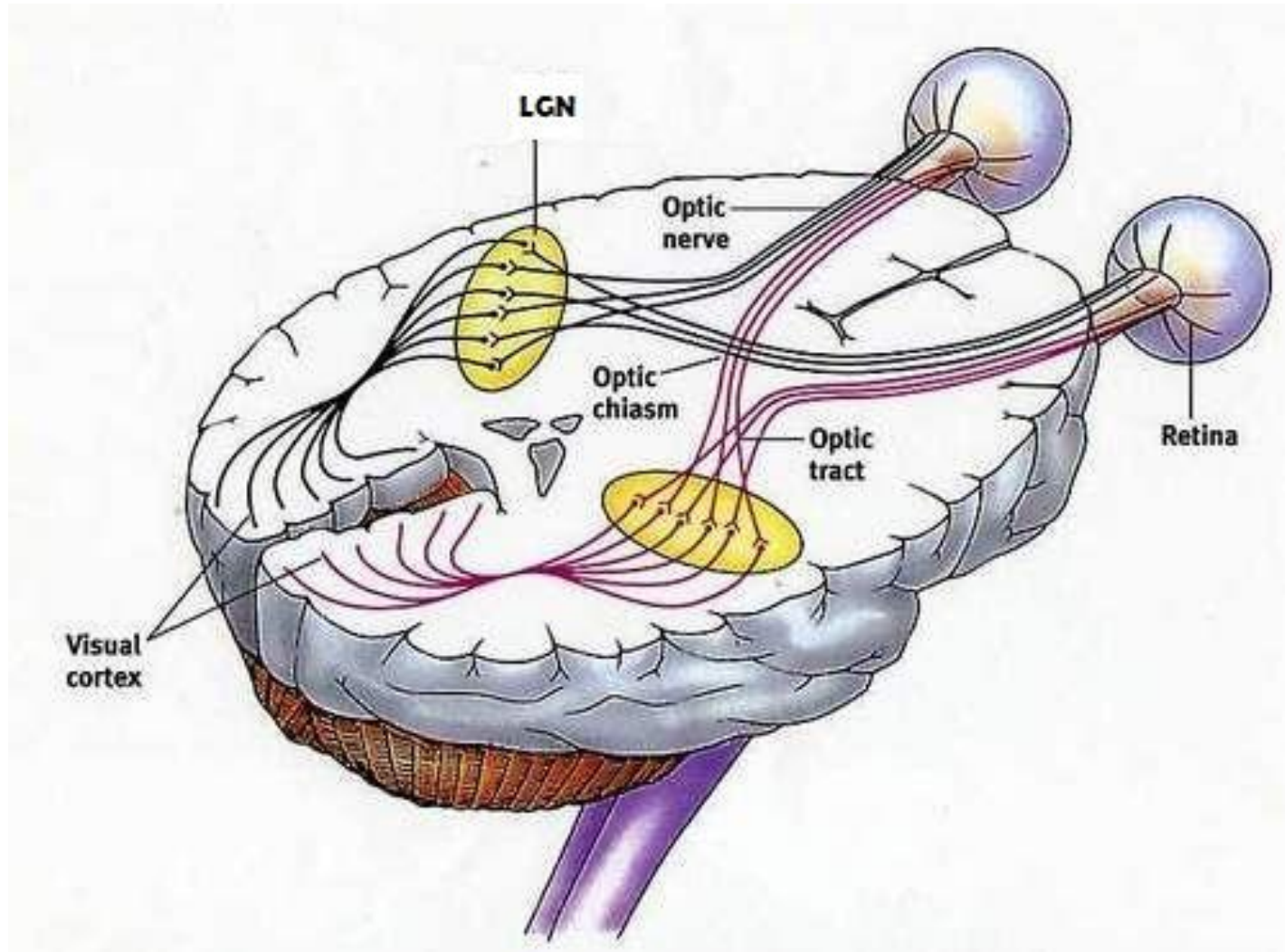
Neuron Activation



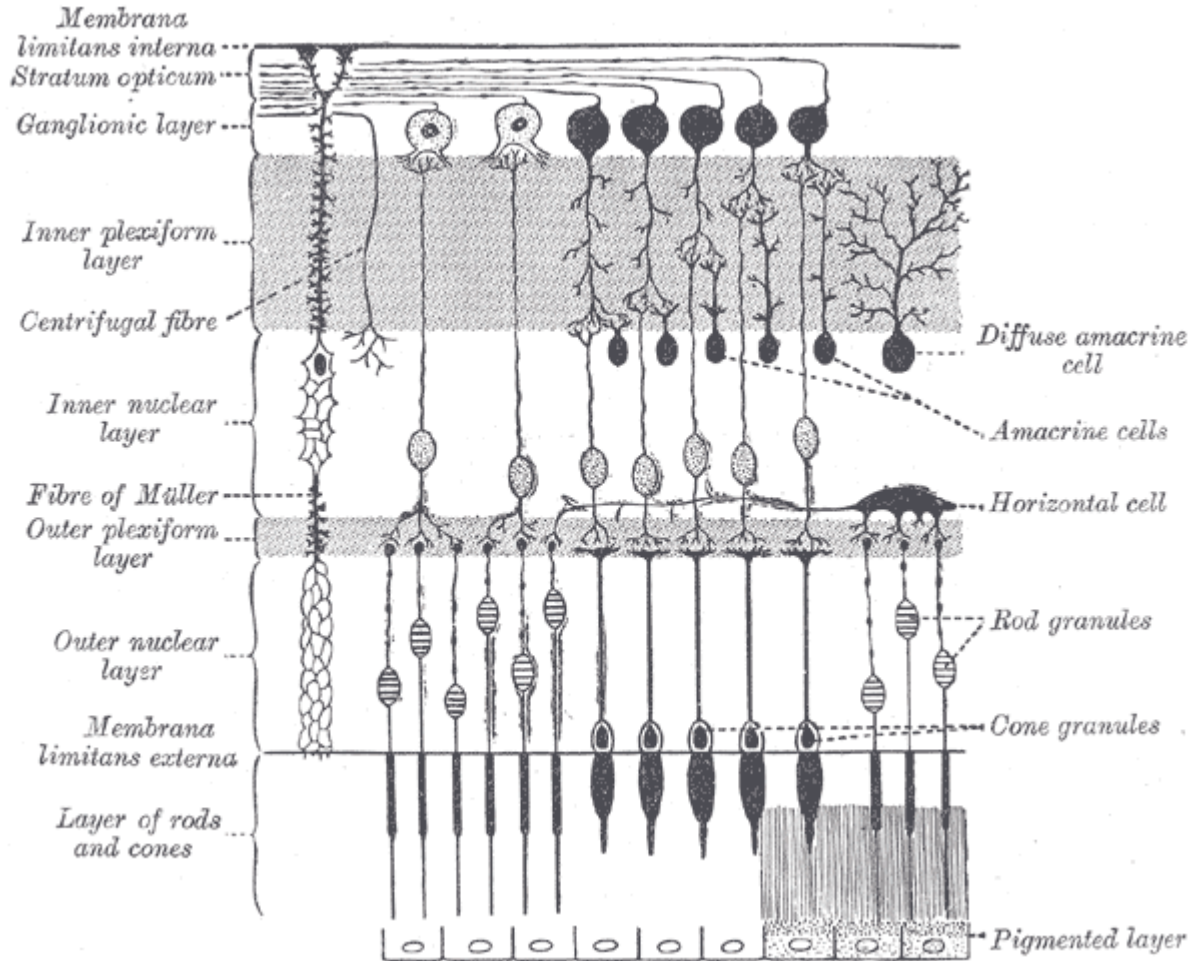
Human Vision System



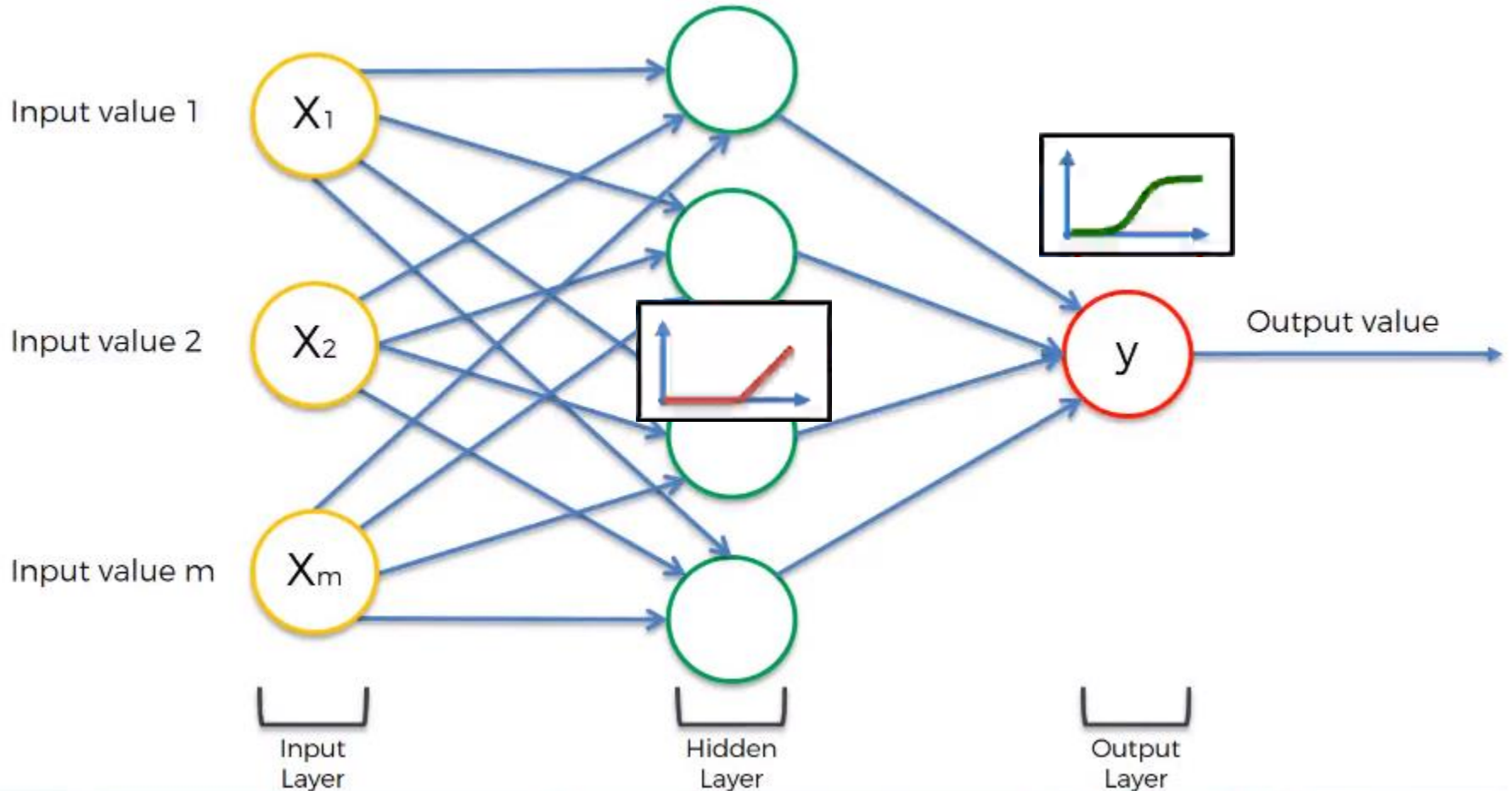
Human Vision System



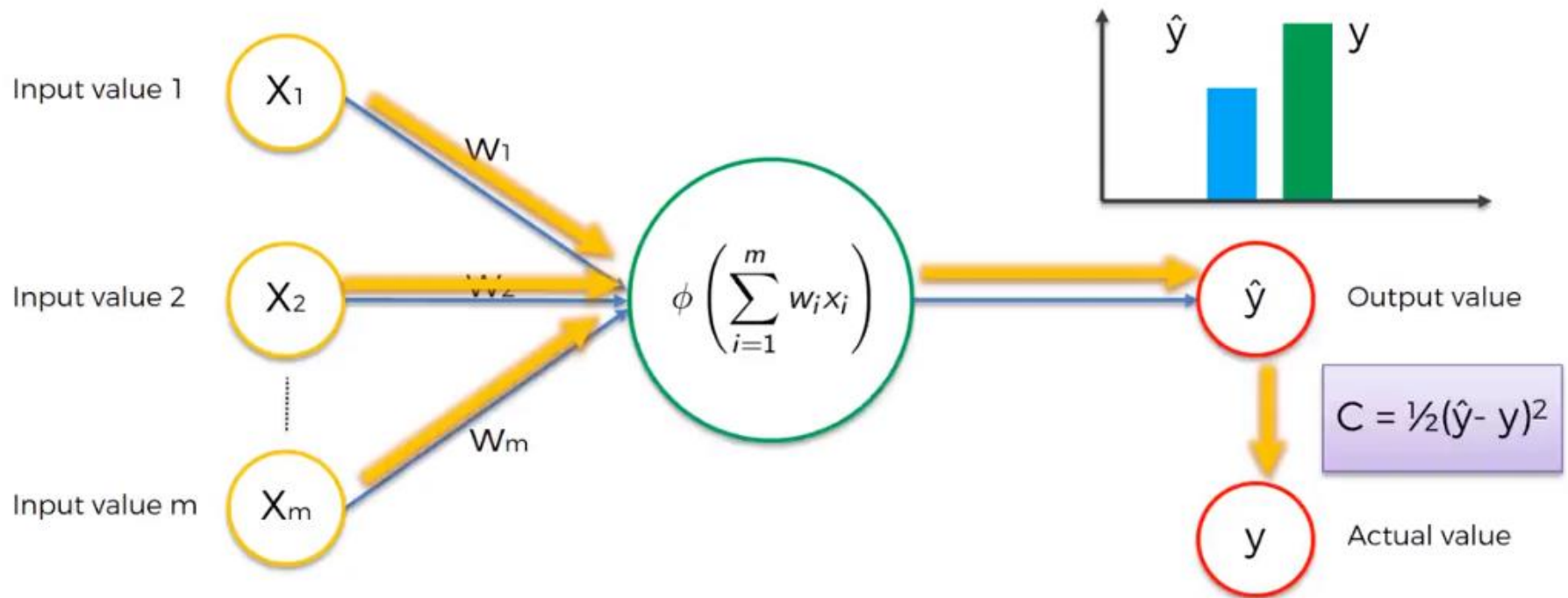
Human Vision System



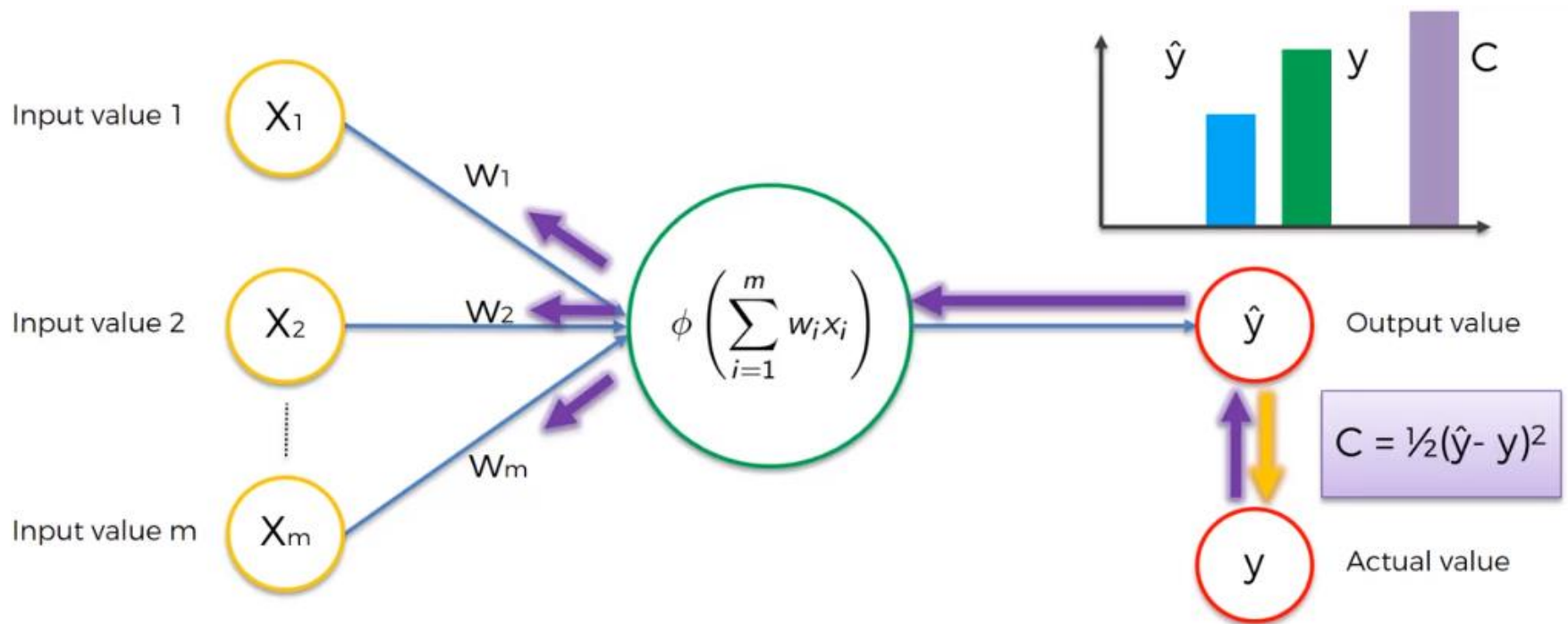
Simple Neural Network



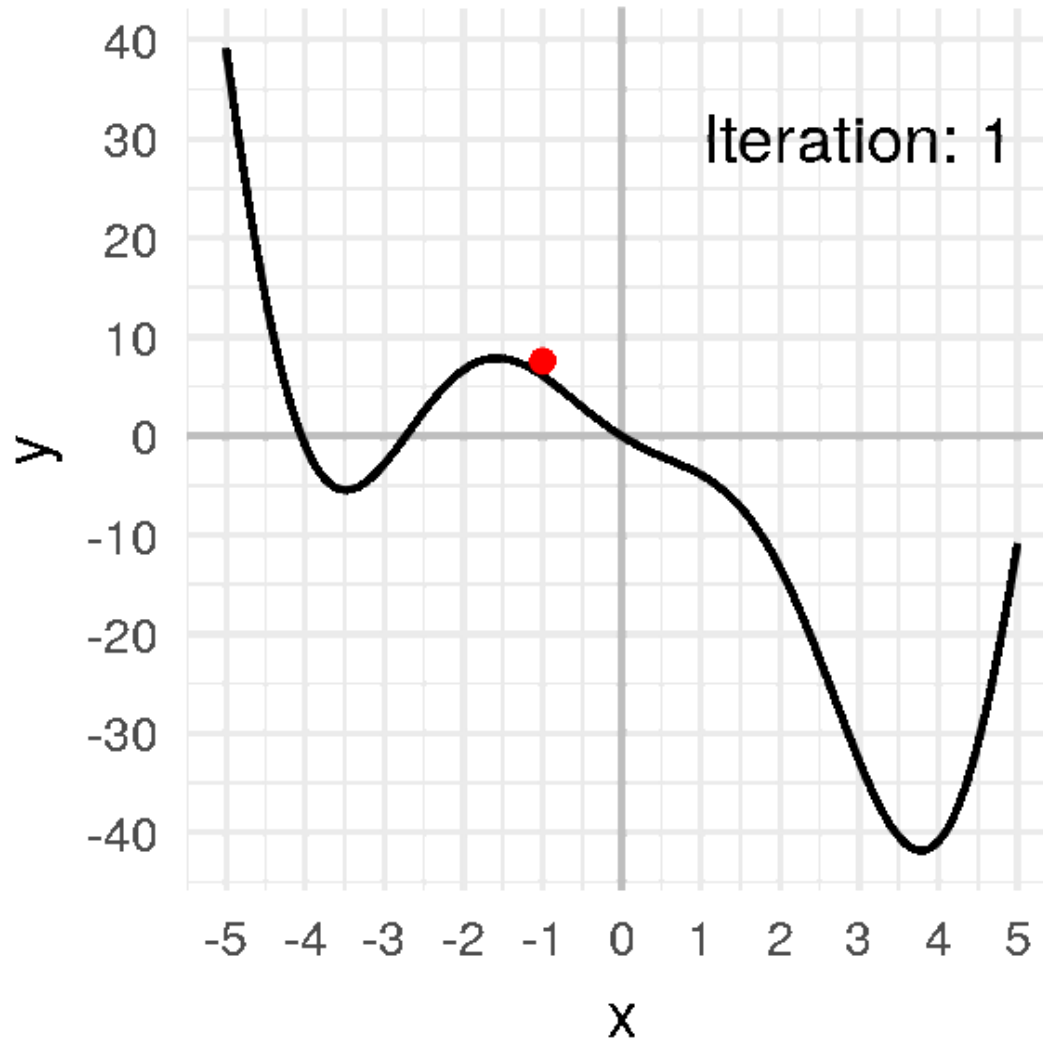
How do Neural Networks Learn?



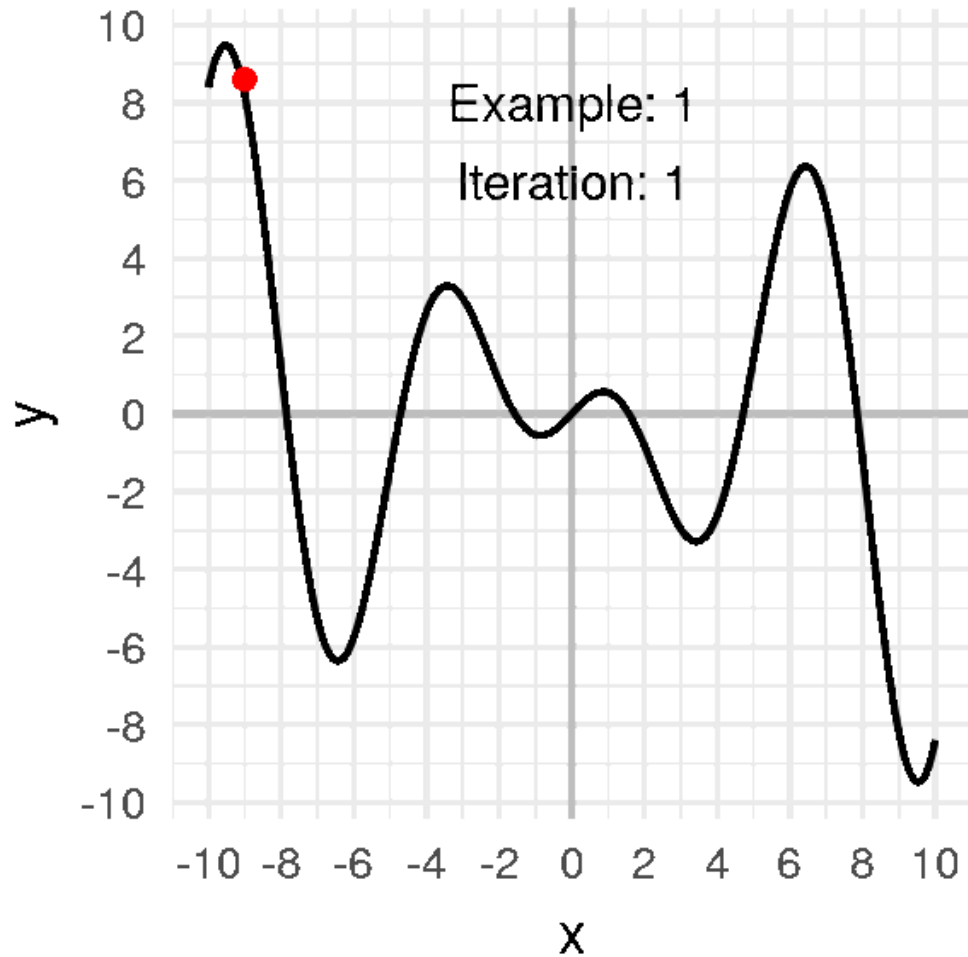
How do Neural Networks Learn?



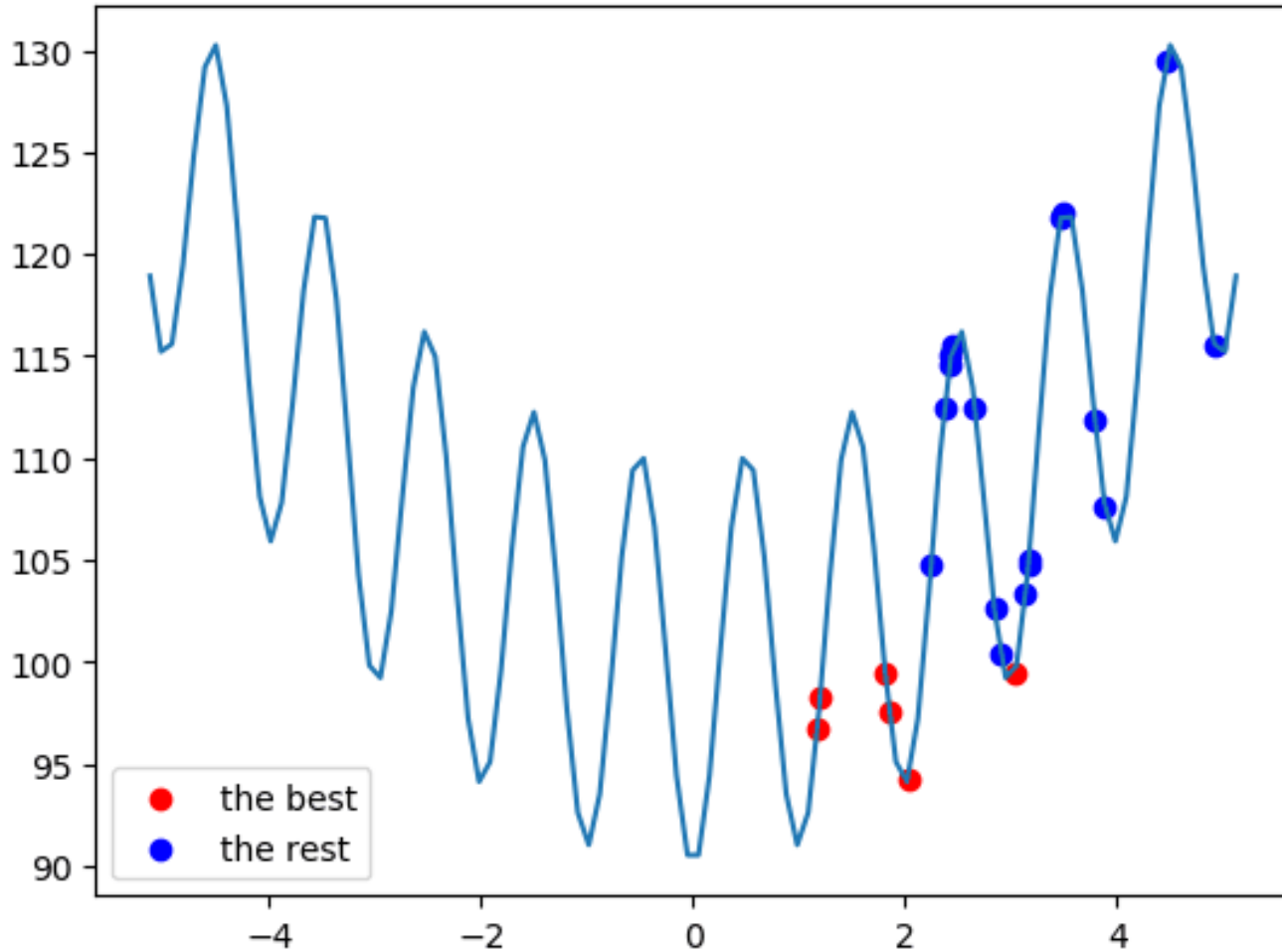
Gradient Descent



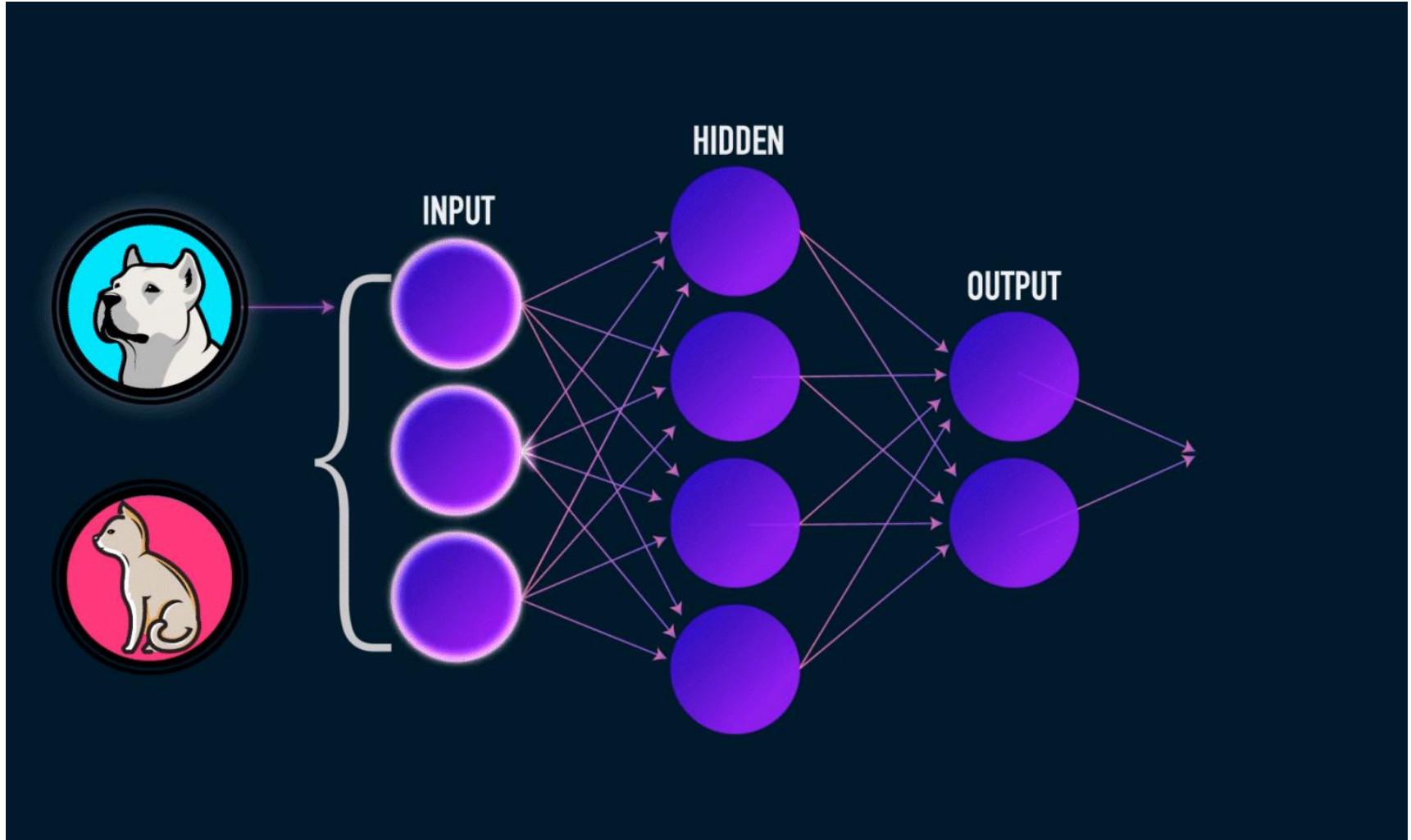
Gradient Descent



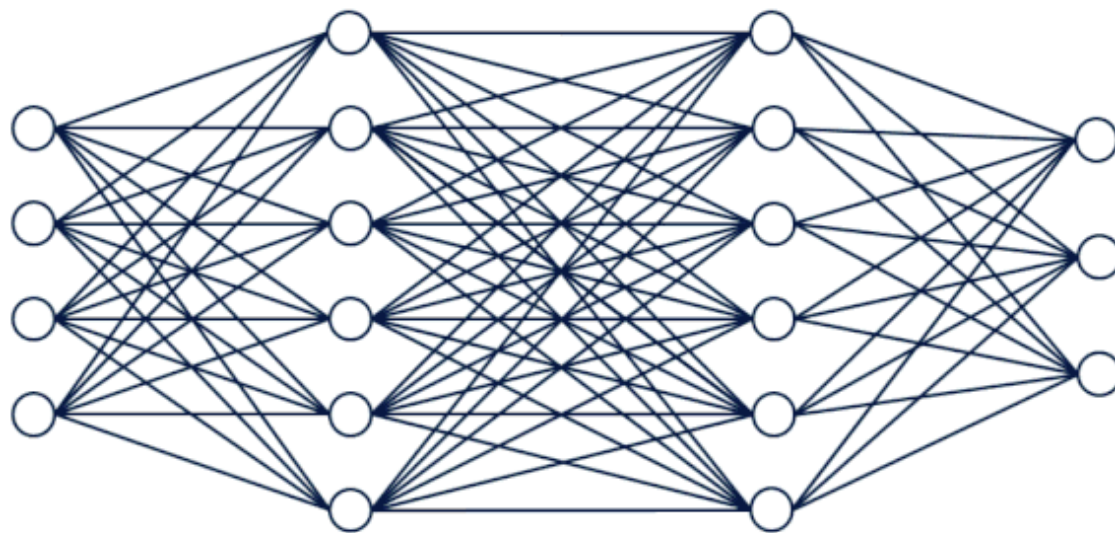
Stochastic Gradient Descent



Simple Neural Network

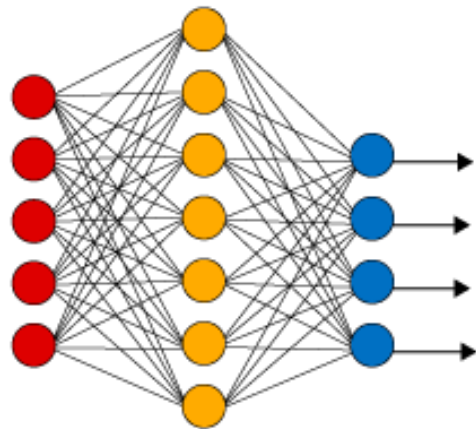


Deep Neural Network

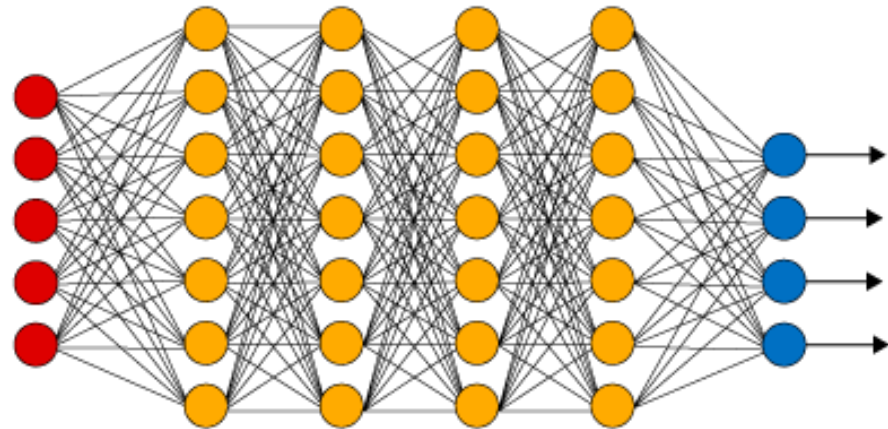


Simple vs. Deep Neural Network

Simple Neural Network



Deep Learning Neural Network

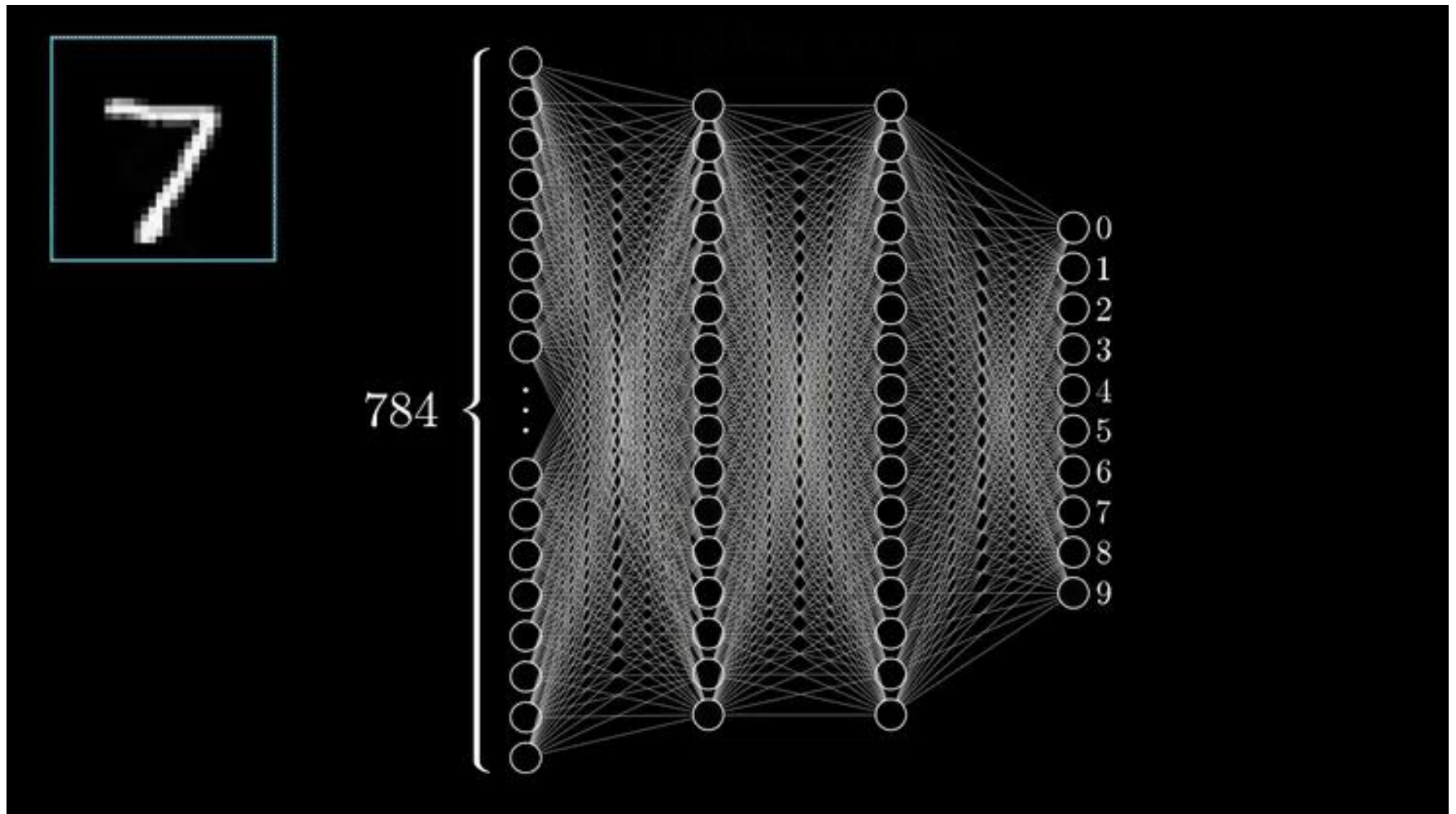


● Input Layer

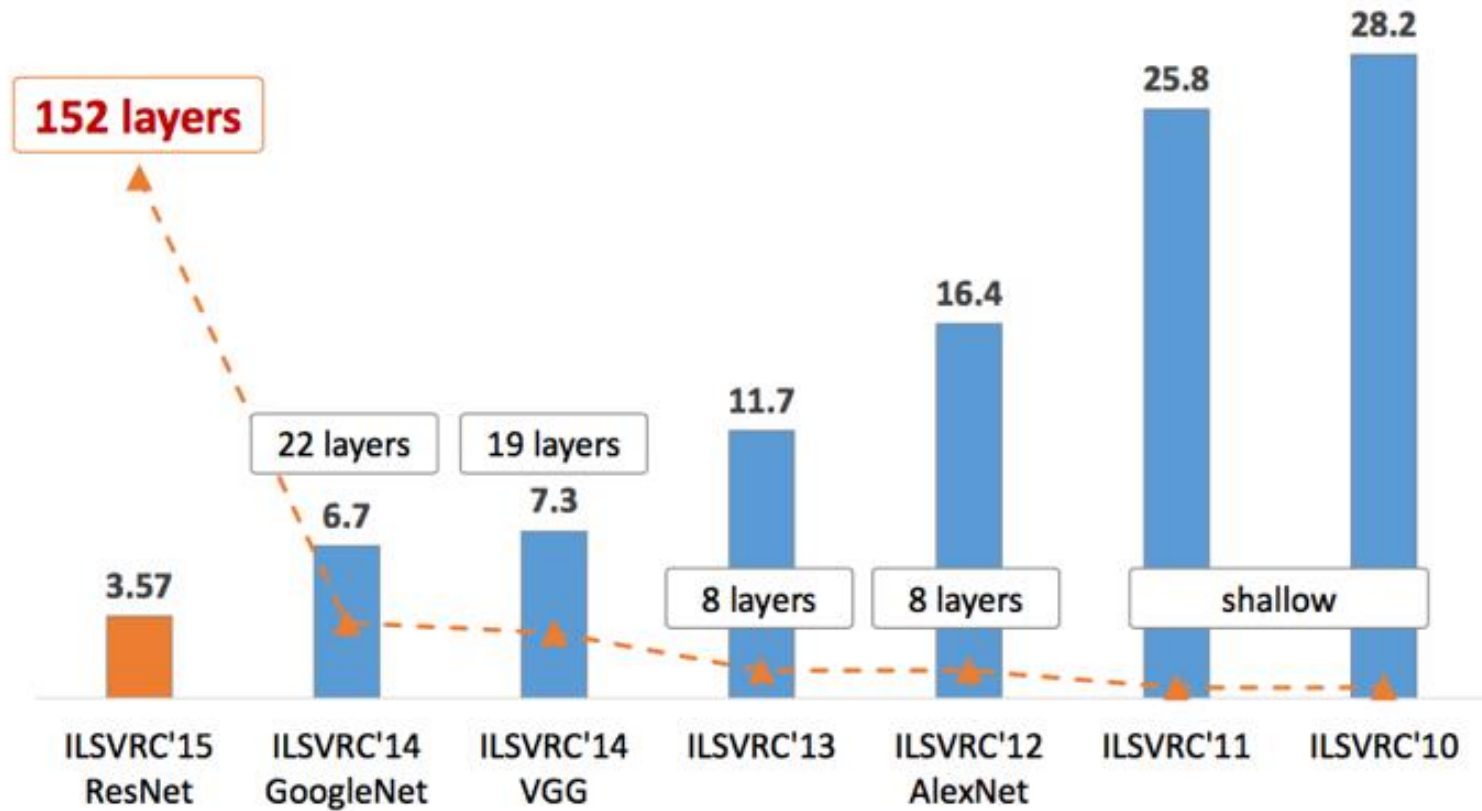
● Hidden Layer

● Output Layer

Deep Neural Network

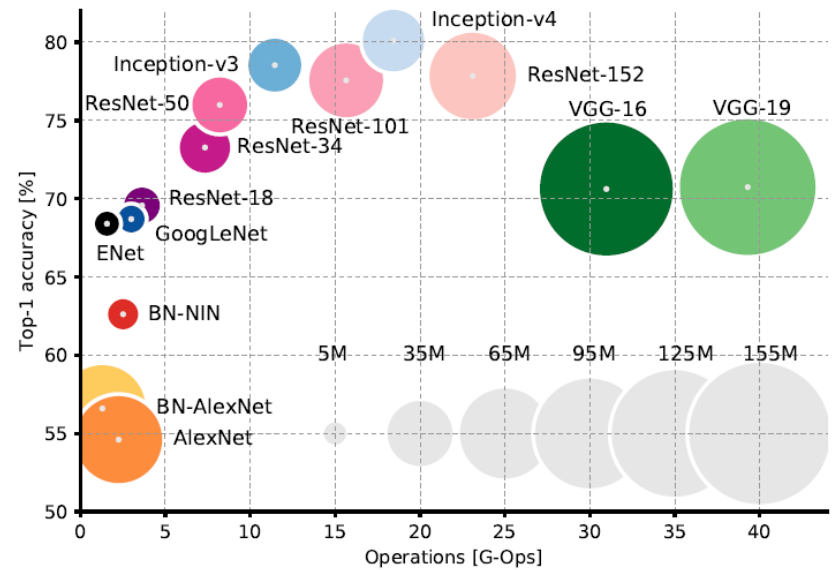
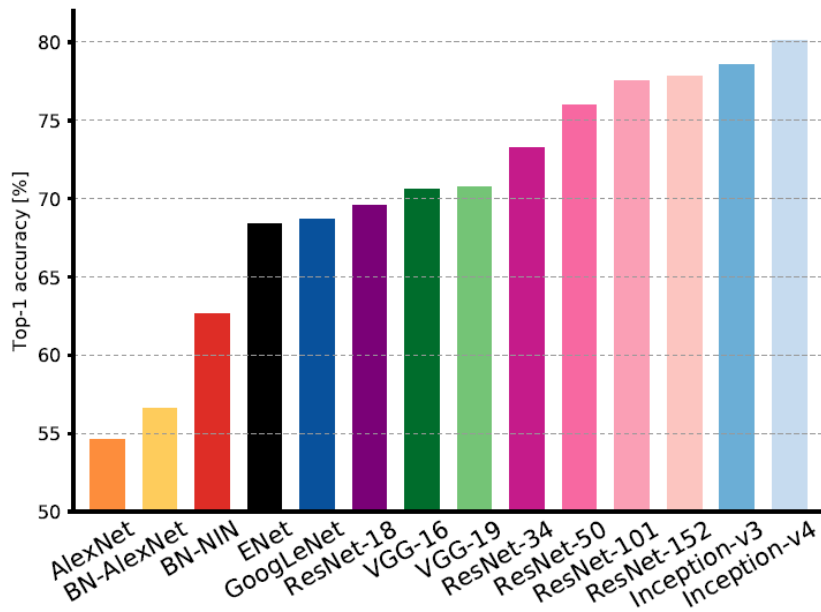


Evolution of Deep Neural Network



<https://iq.opengenus.org/evolution-of-cnn-architectures/>

Evolution of Deep Neural Network



<https://arxiv.org/abs/1605.07678>

Simple Neural Network Toolbox



Products Solutions Academia Support **Community** Events

Get MATLAB



File Exchange

Search File Exchange

File Exchange

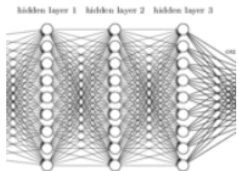


MATLAB Central ▾ Files Authors My File Exchange Contribute About

Trial software

File Exchange now awards reputation points!

Learn how you can earn [reputation points](#) and be on the [leaderboard](#).



Simple Neural Network

version 1.1 (7.17 MB) by Vahe Tshitoyan

A fully connected customizable neural network with an example.
<https://github.com/vtshitoyan/simpleNN>

★★★★☆ 8 Ratings

102 Downloads

Updated 10 Feb 2019



[view license on GitHub](#)

+ Follow

Download from GitHub

Overview

Functions

A fully connected neural network with many options for customisation.

Basic training:

```
modelINN = learnNN(X, y);
```

Prediction:

```
p = predictNN(X_valid, modelINN);
```

One can use an arbitrary number of hidden layers, different activation functions (currently tanh or sigm), custom regularisation parameter, validation sets, etc. The code does not use any matlab toolboxes, therefore, it is perfect if you do not have the statistics and machine learning toolbox, or if you have an older version of matlab. I use the conjugate gradient algorithm for minimisation borrowed from Andrew Ngs machine learning course. See the github and comments in the code for more documentation.

Requires

MATLAB

MATLAB Release Compatibility

Created with R2015b

Compatible with any release

Platform Compatibility

Windows macOS Linux

Categories

<https://www.mathworks.com/matlabcentral/fileexchange/64247-simple-neural-network>

Installing SNN Toolbox

- Download the zip file
- Unzip the file and rename the folder
- Add path in the matlab

PSQI Data Revisited

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	SG Diagnos	PSQI_5.1	PSQI_5.2	PSQI_5.3	PSQI_5.4	PSQI_5.5	PSQI_5.6	PSQI_5.7	PSQI_5.8	PSQI_5.9	PSQI_5.10	sleep quality	sleep latency	sleep duration	sleep efficiency	disturbance	medication	me dysfunction	PSQI_sum
2	PI	2	3	0	2	0	0	0	2	0	0	2	2	3	3	1	0	2	13
3	PI	0	0	0	0	0	0	0	1	0	0	1	0	0	0	1	0	0	2
4	PI	2	2	0	2	2	0	2	0	0	0	2	2	3	2	1	0	1	11
5	PI	3	3	2	0	0	0	3	3	0	2	2	3	3	0	2	2	1	13
6	PI	3	2	1	0	0	0	0	1	0	0	2	3	0	0	1	3	1	10
7	PI	3	2	2	2	3	3	3	3	3	3	3	3	2	3	3	3	3	20
8	PI	0	3	3	3	2	2	2	0	2	0	2	0	0	0	3	0	2	5
9	PI	2	2	2	1	1	2	2	3	0	0	2	3	0	0	2	0	1	8
10	PI	0	3	2	0	0	0	3	2	0	0	2	0	3	2	2	3	3	15
11	PI	2	2	2	0	0	1	2	3	0	0	1	3	0	0	2	3	1	10
12	PI	3	2	0	0	0	2	2	2	1	0	2	3	0	2	1	0	0	8
13	PI	3	2	2	0	0	0	2	2	0	0	1	3	3	0	1	3	1	12
14	PI	1	3	3	0	3	0	1	3	0	0	2	1	1	1	2	2	1	10
15	PI	3	3	3	0	0	0	2	3	2	0	3	3	3	3	2	3	2	19
16	PI	0	0	1	0	1	0	0	0	0	0	0	0	0	0	1	0	0	1
17	PI	3	3	2	0	0	1	2	2	0	0	2	2	1	3	2	0	2	12
18	PI	1	1	1	0	0	0	0	3	0	0	2	1	0	1	1	3	0	8
19	PI	2	3	2	3	0	2	2	0	0	0	2	2	0	0	2	3	3	12
20	PI	1	2	2	0	0	2	0	2	1	0	3	1	1	3	1	0	1	10
21	PI	3	1	0	0	0	0	2	3	2	0	3	3	1	0	1	3	0	11
22	PI	0	3	3	1	2	2	2	2	3	0	3	0	3	0	2	3	0	11
23	PI	3	3	2	2	3	0	0	2	3	0	2	3	3	2	2	3	2	17
24	PI	3	2	2	1	1	1	1	1	2	0	2	3	3	3	2	3	2	18
25	PI	3	2	3	0	0	0	0	0	0	0	2	2	1	0	1	0	1	7
26	PI	1	2	2	0	0	0	2	2	0	0	1	1	0	0	1	3	3	9
27	PI	3	2	3	0	1	2	2	3	1	0	2	3	3	2	2	3	0	15
28	PI	3	3	3	0	0	0	1	1	0	0	3	3	3	3	1	2	1	16
29	PI	1	1	1	1	1	0	0	0	0	0	2	1	0	0	1	0	1	5
30	PI	3	3	3	0	0	0	3	2	0	0	1	2	1	0	2	0	1	7
31	PI	1	2	0	0	0	3	0	0	2	0	1	1	1	0	1	0	2	6
32	PI	3	1	0	0	0	0	0	3	3	0	2	3	1	1	1	3	3	14

Import PSQI Data into Matlab

- data =
readtable('sleep_questionnaire_dataset.xlsx');

```
>> data = readtable('sleep_questionnaire_dataset.xlsx');
```

Warning: Table variable names were modified to make them valid MATLAB identifiers. The original names are saved in the VariableDescriptions property.

Predictors and Response

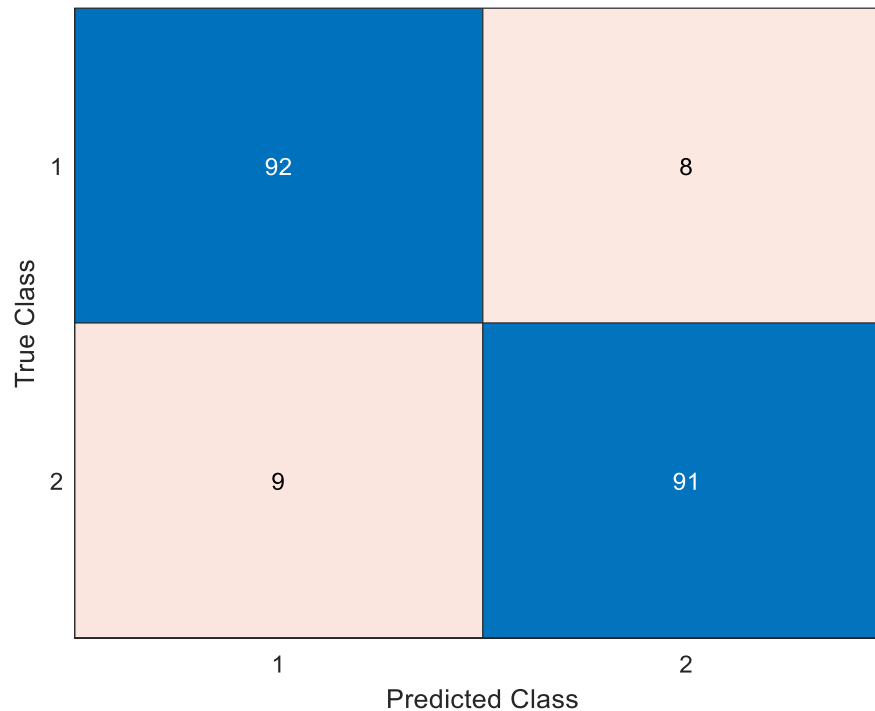
- `predictors = table2array(data(:,2:end));`
- `response =
double(categorical(data.PSGDiagnosis));`

Train Simple Neural Network Model

- `modelNN = learnNN(predictors, response);`
- `p = predictNN(predictors, modelNN);`

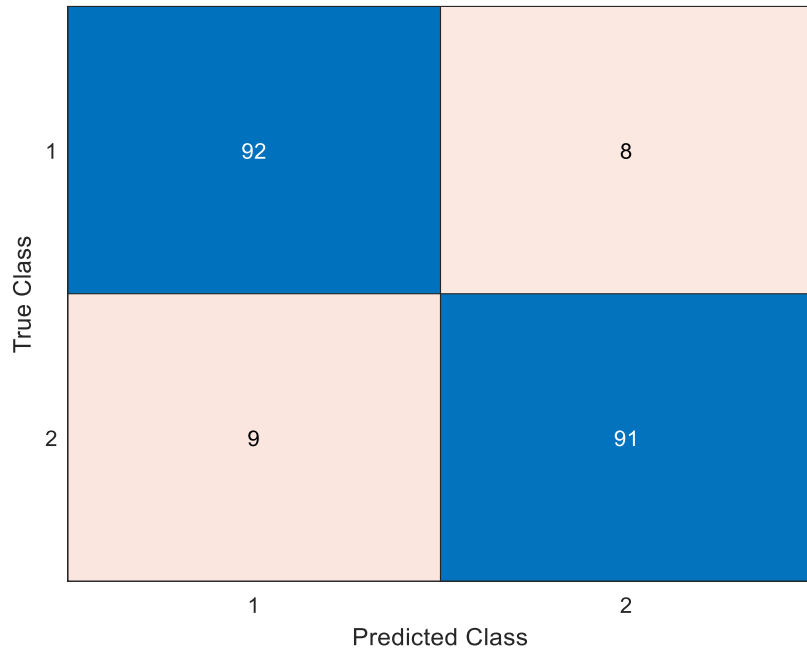
Confusion Matrix

- `c = confusionmat(response,p);`
- `confusionchart(c)`



Simple Neural Network vs. Linear Regression

Simple Neural Network



Linear Regression

