11210IPT553000 Deep Learning in Biomedical Optical Imaging

Background Survey

https://forms.gle/VVQbKm8xv9avBd3d6

Week2
Neural Network Basics (Part I)

Instructor: Hung-Wen Chen @NTHU, Fall 2023
2023/09/18

SPONSORED BY

coursera deeplearning.ai NVIDIA DEEP LEARNING INSTITUTE

NVIDIA DEEP LEARNING INSTITUTE CERTIFIED INSTRUCTOR

Outline of Today's Lecture

- Course Overview
- Introduction to Deep Learning (Course 1 Week1)
- Neural Networks Basics (Course 1 Week2)
- Lab Practice: Coding Warm-up & GitHub

Course Overview

Syllabus

Course Overview

In this course, you will learn:

(1) the foundational knowledge of deep learning algorithms, build neural networks from scratch via Python, and furthermore get familiar with different frameworks such as Pytorch, Tensorflow, and Keras.

(2) The principle of different biomedical optical imaging technologies will be introduced as well.

(3) A final project on medical imaging is required.

(4) The opportunities of earning free official certificates from Nvidia Deep Learning Institute and DeepLearning.AI from Coursera as well, which might be helpful for your future career development.

Course Overview

Grading

- **Homework (35%):** Assignments to get familiar with basic deep learning programming.
  (Total 5 including one from Nvidia Workshop)
- **Midterm (30%):** Basic concepts of deep learning and related mathematical derivation.
- **Final Project Presentation (20%):** A project presentation of a research paper related to the deep learning application in medical imaging and the code implementation of the paper.
- **Creativity Report (15%):** A detailed analysis of your model implementation on a given image dataset.

References

- Goodfellow, Y. Bengio, and A. Courville, "Deep Learning," 2016.
- Francois Chollet (creator of Keras), "Deep Learning with Python," 2017.
- J. Schmidhube, "Deep Learning in Neural Networks: An Overview," Neural Networks 61: 85-117, 2015.
- Y. Bengio, Y. LeCun, and G. Hinton, "Deep Learning," Nature 521: 436-44, 2015.

Online Resources

- Goodfellow, Y. Bengio, and A. Courville, "Deep Learning," 2016.
  http://www.deeplearningbook.org
- Deep Learning Specialization by Andrew Ng
  https://www.youtube.com/c/Deeplearningai/playlists
- Stanford CS231n: Deep Learning for Computer Vision
  http://cs231n.stanford.edu/schedule.html
- AI for Medicine Specialization.
  https://www.coursera.org/specializations/ai-for-medicine
- NVIDIA Deep Learning Institute
  https://www.nvidia.com/en-us/training/
- NVIDIA NGC
  https://www.nvidia.com/zh-tw/gpu-cloud/containers/
- TensorFlow 2 quickstart for beginners.
  https://www.tensorflow.org/tutorials/quickstart/beginner
- Python Numpy Tutorial (with Jupyter and Colab)
  https://cs231n.github.io/python-numpy-tutorial/
- Python Basics for Data Science
  https://www.edx.org/course/python-basics-for-data-science
- Python Tutorial
  https://www.w3schools.com/python/
- 莫烦Python
  https://mofanpy.com/

Deep Learning Specialization

There are 5 Courses in this Specialization (focus on the first 4 courses)

Deep Learning Specialization

DeepLearning.AI

Become a Deep Learning expert. Master the fundamentals of deep learning and break into AI.

★★★★★ 4.8 270,766 ratings

Andrew Ng +2 more instructors TOP INSTRUCTORS

COURSE 1 — 4 weeks

Neural Networks and Deep Learning

4.9 103,180 ratings • 20,631 reviews

In the first course of the Deep Learning Specialization, you will study the foundational concept of neural networks and deep learning.

By the end, you will be familiar with the significant technological trends driving the rise of deep learning; build, train, and apply fully connected deep neural networks; implement

SHOW ALL

COURSE 2 — 3 weeks

Improving Deep Neural Networks: Hyperparameter Tuning, Regularization and Optimization

4.9 57,409 ratings • 6,595 reviews

In the second course of the Deep Learning Specialization, you will open the deep learning black box to understand the processes that drive performance and generate good results systematically.

SHOW ALL

COURSE 3 — 2 weeks

Structuring Machine Learning Projects

4.8 45,740 ratings • 5,219 reviews

In the third course of the Deep Learning Specialization, you will learn how to build a successful machine learning project and get to practice decision-making as a machine learning project leader.

SHOW ALL

COURSE 4 — 4 weeks

Convolutional Neural Networks

4.9 38,261 ratings • 5,018 reviews

In the fourth course of the Deep Learning Specialization, you will understand how computer vision has evolved and become familiar with its exciting applications such as autonomous driving, face recognition, reading radiology images, and more.

SHOW ALL

COURSE 5 — 3 weeks

Sequence Models

4.8 26,176 ratings • 3,089 reviews

In the fifth course of the Deep Learning Specialization, you will become familiar with NLP models and their exciting applications such as speech recognition, music synthesis, chatbots, machine translation, natural language understanding, and more that have become possible with the evolution of sequence algorithms thanks to deep learning.

SHOW ALL

$49/month on Coursera

https://www.coursera.org/specializations/deep-learning?utm_source=deeplearningai&utm_medium=institutions&utm_campaign=WebsiteCoursesDLSTopButton#courses

NVIDIA ›

SHOP DRIVERS SUPPORT

DEEP LEARNING INSTITUTE ONLINE COURSES INSTRUCTOR-LED WORKSHOPS EDUCATOR PROGRAMS ENTERPRISE SOLUTIONS RESOURCES

INSTRUCTOR-LED WORKSHOP

Fundamentals of Deep Learning

Request a workshop for your organization ›

Notify me when public workshops are available ›

https://www.nvidia.com/en-us/training/instructor-led-workshops/fundamentals-of-deep-learning/

Businesses worldwide are using artificial intelligence to solve their greatest challenges. Healthcare professionals use AI to enable more accurate, faster diagnoses in patients. Retail businesses use it to offer personalized customer shopping experiences. Automakers use it to make personal vehicles, shared mobility, and delivery services safer and more efficient. Deep learning is a powerful AI approach that uses multi-layered artificial neural networks to deliver state-of-the-art accuracy in tasks such as object detection, speech recognition, and language translation. Using deep learning, computers can learn and recognize patterns from data that are considered too complex or subtle for expert-written software.

In this workshop, you'll learn how deep learning works through hands-on exercises in computer vision and natural language processing. You'll train deep learning models from scratch, learning tools and tricks to achieve highly accurate results. You'll also learn to leverage freely available, state-of-the-art pre-trained models to save time and get your deep learning application up and running quickly.

Workshop Details

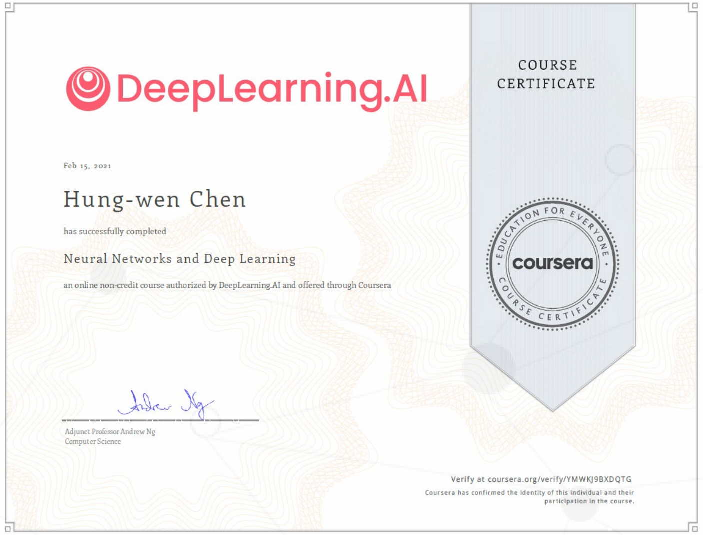
Duration: 8 hours

$\boxed{\textbf{Price: } \$500}$ for public workshops, contact us for enterprise workshops.

Prerequisites: An understanding of fundamental programming concepts in Python 3, such as functions, loops, dictionaries, and arrays; familiarity with Pandas data structures; and an understanding of how to compute a regression line.

Online Certificates

from DeepLearning.AI and NVIDIA



DeepLearning.AI

COURSE CERTIFICATE

Feb 15, 2021

Hung-wen Chen

has successfully completed

Neural Networks and Deep Learning

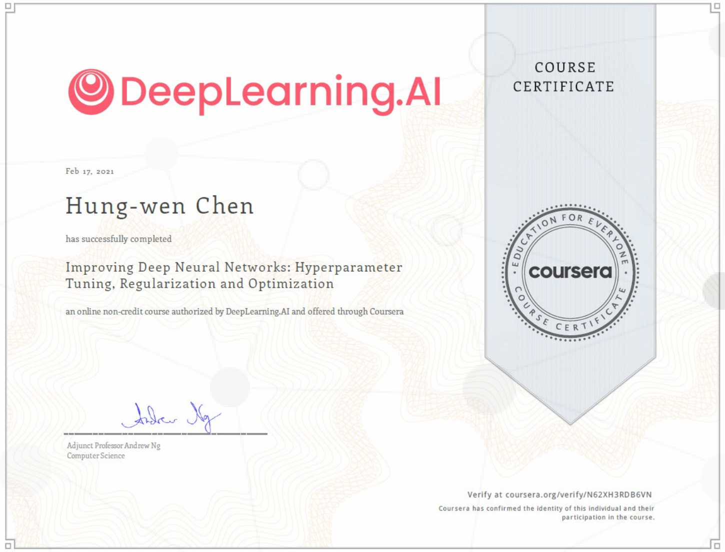
an online non-credit course authorized by DeepLearning.AI and offered through Coursera

EDUCATION FOR EVERYONE · coursera · COURSE CERTIFICATE

Adjunct Professor Andrew Ng
Computer Science

Verify at coursera.org/verify/YMWKJ9BXDQTG

Coursera has confirmed the identity of this individual and their participation in the course.



DeepLearning.AI

COURSE CERTIFICATE

Feb 17, 2021

Hung-wen Chen

has successfully completed

Improving Deep Neural Networks: Hyperparameter Tuning, Regularization and Optimization

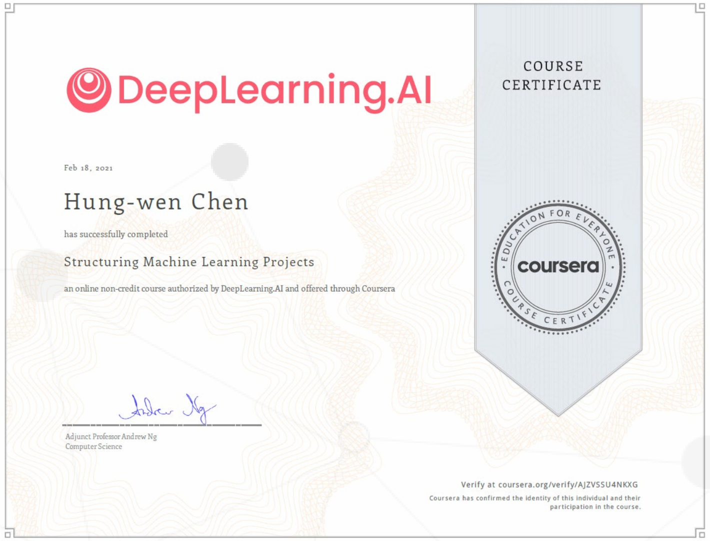
an online non-credit course authorized by DeepLearning.AI and offered through Coursera

EDUCATION FOR EVERYONE · coursera · COURSE CERTIFICATE

Adjunct Professor Andrew Ng
Computer Science

Verify at coursera.org/verify/N62XH3RDB6VN

Coursera has confirmed the identity of this individual and their participation in the course.



DeepLearning.AI

COURSE CERTIFICATE

Feb 18, 2021

Hung-wen Chen

has successfully completed

Structuring Machine Learning Projects

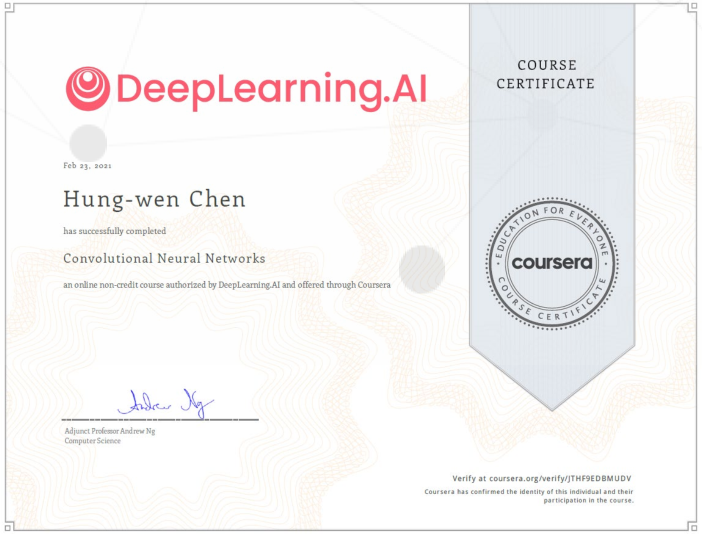
an online non-credit course authorized by DeepLearning.AI and offered through Coursera

EDUCATION FOR EVERYONE · coursera · COURSE CERTIFICATE

Adjunct Professor Andrew Ng
Computer Science

Verify at coursera.org/verify/AJZVSSU4NKXG

Coursera has confirmed the identity of this individual and their participation in the course.



DeepLearning.AI

COURSE CERTIFICATE

Feb 23, 2021

Hung-wen Chen

has successfully completed

Convolutional Neural Networks

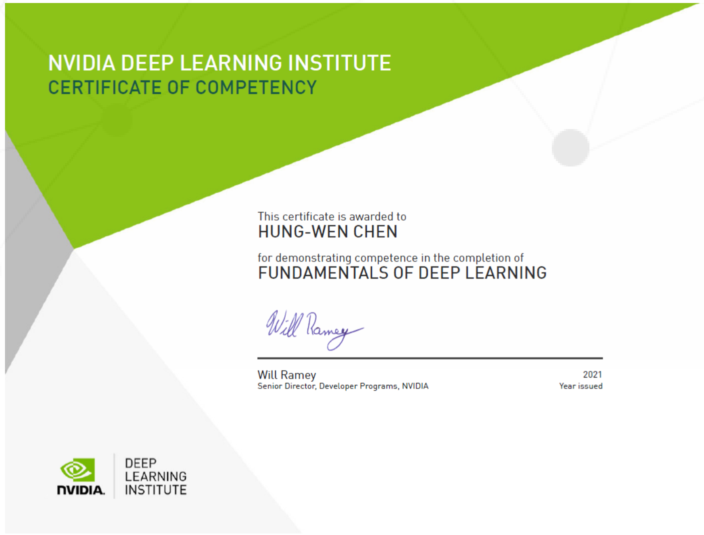
an online non-credit course authorized by DeepLearning.AI and offered through Coursera

EDUCATION FOR EVERYONE · coursera · COURSE CERTIFICATE

Adjunct Professor Andrew Ng
Computer Science

Verify at coursera.org/verify/JTHF9EDBMUDV

Coursera has confirmed the identity of this individual and their participation in the course.



NVIDIA DEEP LEARNING INSTITUTE
CERTIFICATE OF COMPETENCY

This certificate is awarded to
HUNG-WEN CHEN

for demonstrating competence in the completion of
FUNDAMENTALS OF DEEP LEARNING

Will Ramey
Senior Director, Developer Programs, NVIDIA

2021
Year issued

NVIDIA. DEEP LEARNING INSTITUTE

5 Courses

- Neural Networks and Deep Learning
- Improving Deep Neural Networks: Hyperparameter Tuning, Regularization and Optimization
- Structuring Machine Learning Projects
- Convolutional Neural Networks
- Sequence Models

Online Certificates

Deep Learning Specialization

Mar 17, 2021

Hung-Wen Chen

has successfully completed the online, non-credit Specialization

Deep Learning

Congratulations! You have completed all five courses of the Deep Learning Specialization. In this Specialization, you built neural network architectures such as Convolutional Neural Networks, Recurrent Neural Networks, LSTMs, Transformers and learned how to make them better with strategies such as Dropout, BatchNorm, Xavier/He initialization, and more. You mastered these theoretical concepts and their application using Python and TensorFlow and also tackled real-world case studies such as autonomous driving, sign language reading, music generation, computer vision, speech recognition, and natural language processing. You're now familiar with the capabilities, challenges, and consequences of deep learning and are ready to participate in the development of leading-edge AI technology.

Andrew Ng,
Founder,
DeepLearning.AI
Co-founder, Coursera

Verify this certificate at:
coursera.org/verify/specialization/YYFRHNXCK2QX

Introduction to Deep Learning

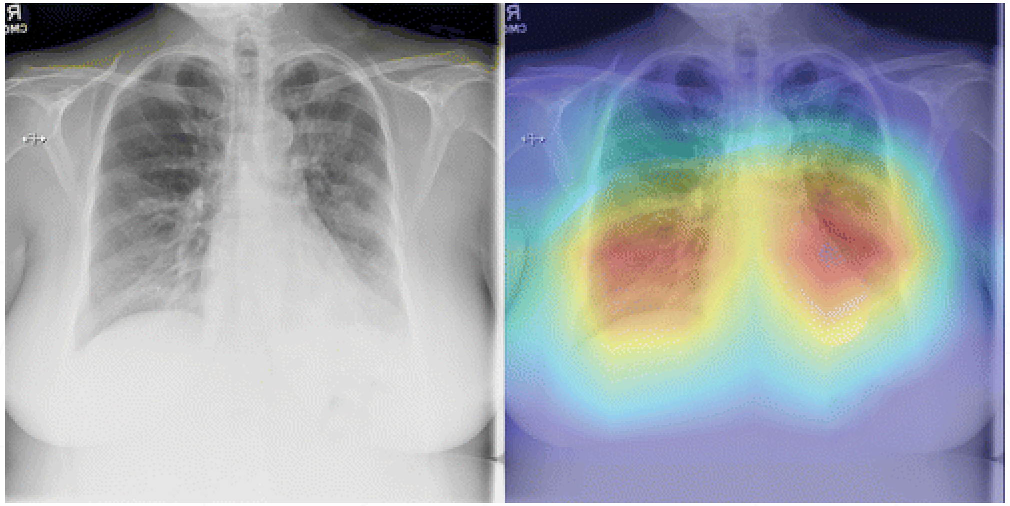
AI Data Scientist

https://www.itread01.com/content/1545465431.html

Applications

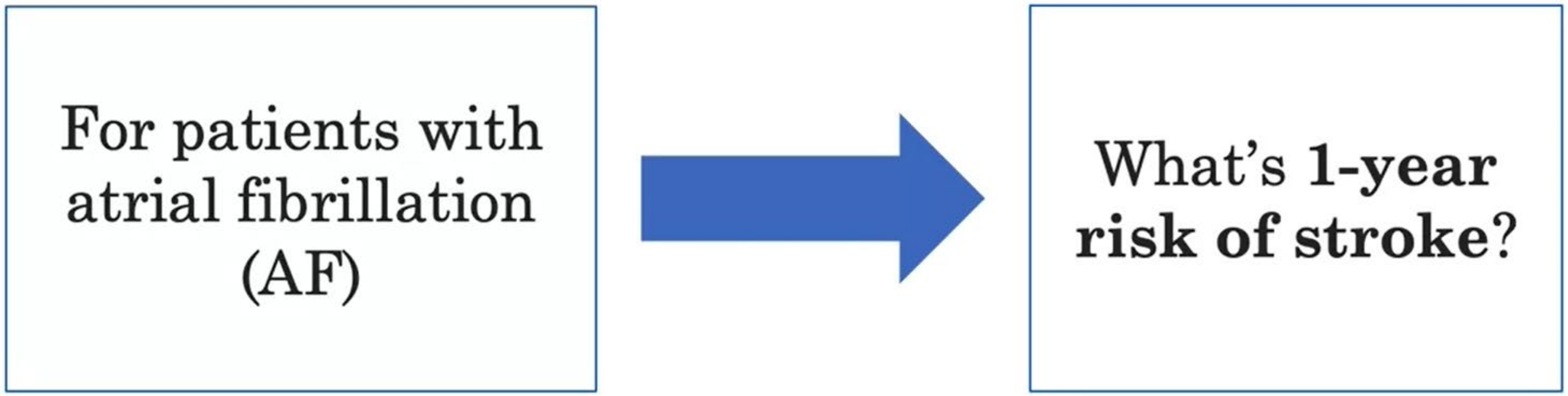
AI is transforming the practice of medicine.
It's helping doctors diagnose patients more accurately, make predictions about patients' future health, and recommend better treatments.

- Diagnose diseases from x-rays and 3D MRI brain images
- Predict patient survival rates more accurately
- Estimate treatment effects on patients using data from randomized trials
- Automate the task of labeling medical datasets using natural language processing



Predicting risk of a future event

Risk of Stroke



- AF is a common abnormal heart rhythm
- Stroke is when blood flow to an area of brain is cut off

CHADSVASc score

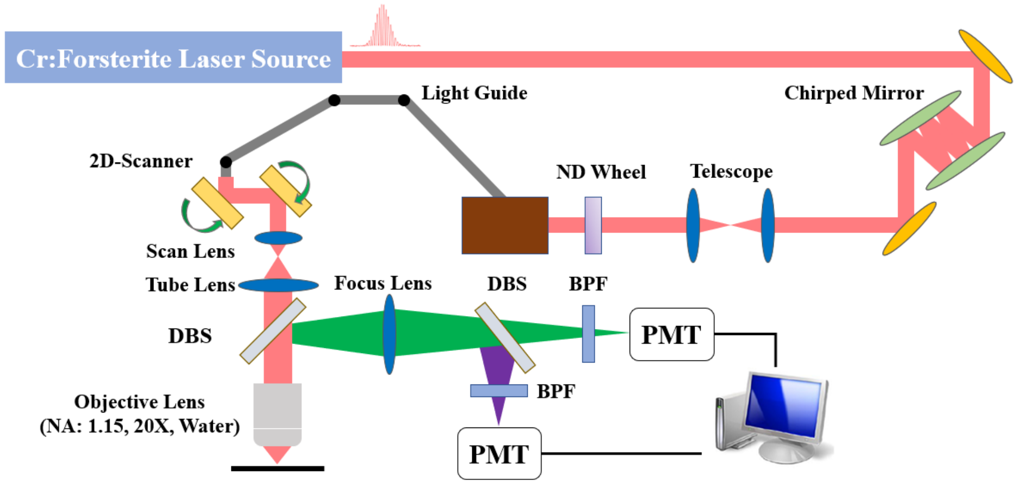
CHA_2DS_2-VASc ("chads vasc") score

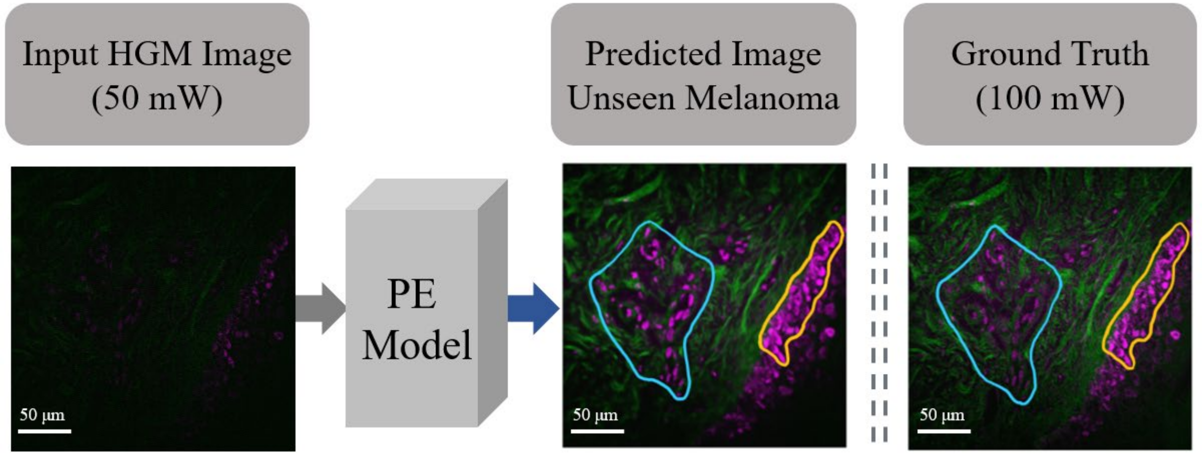
		Coefficient	Value 1=Yes, 0=No	Coefficient x Value
C	Congestive heart failure	1	0	0
H	Hypertension	1	1	1
A_2	Age 75 years or older	2	0	0
D	Diabetes mellitus	1	1	1
S_2	Stroke, TIA, or TE	2	0	0
V	Vascular disease	1	0	0
A	Age 65 to 74 years	1	1	1
Sc	Sex category (female)	1	0	0
	Score			3

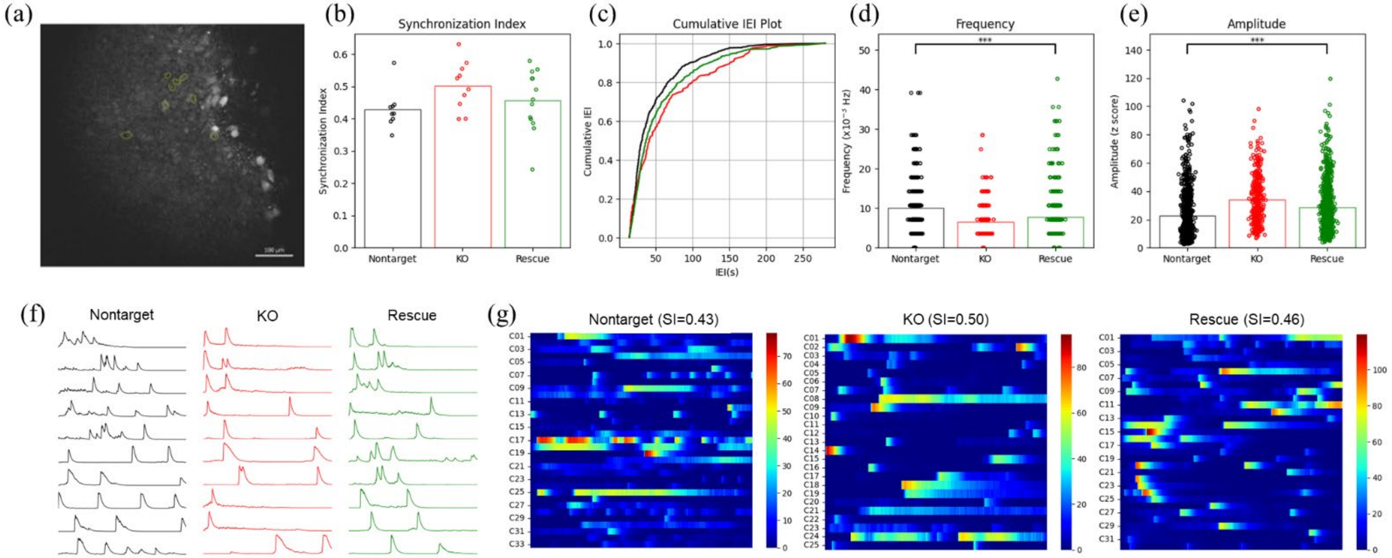
A 70-year-old male diagnosed with AF has *hypertension* and *diabetes.*

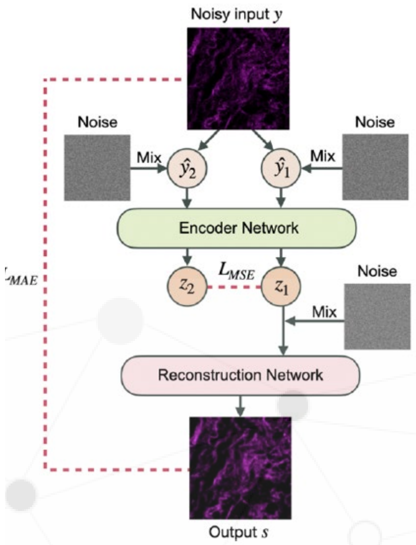
AI for Medicine

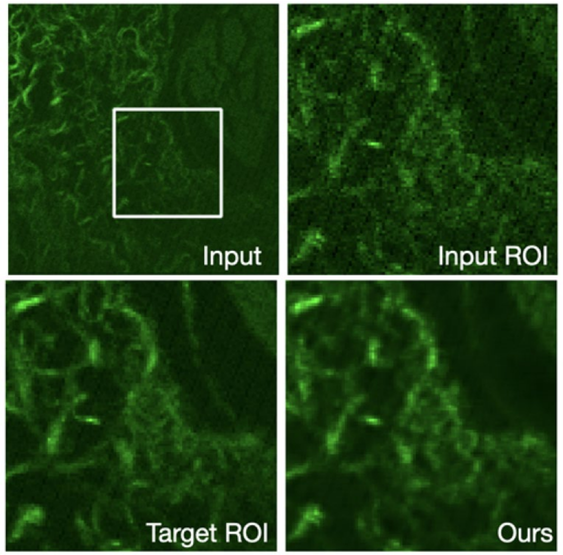
Biomedical Research in HW Group



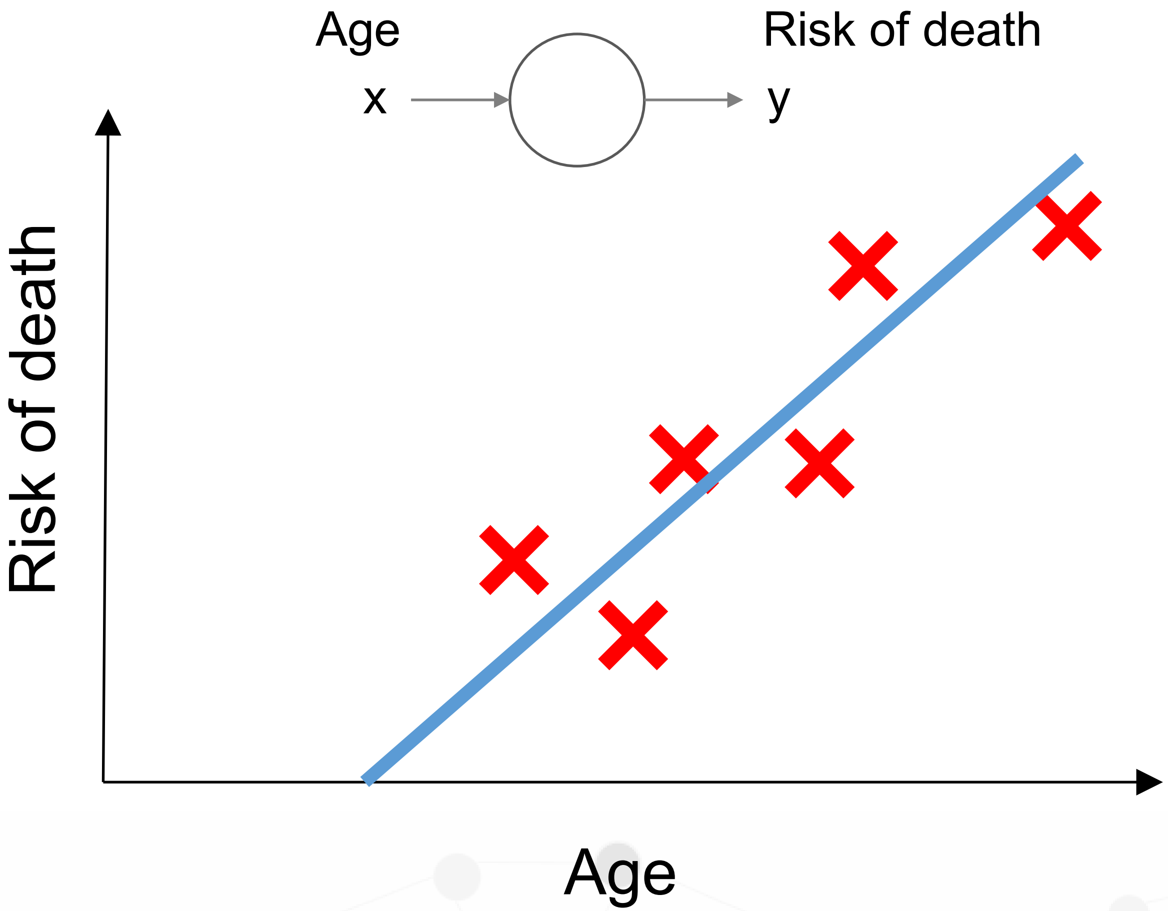




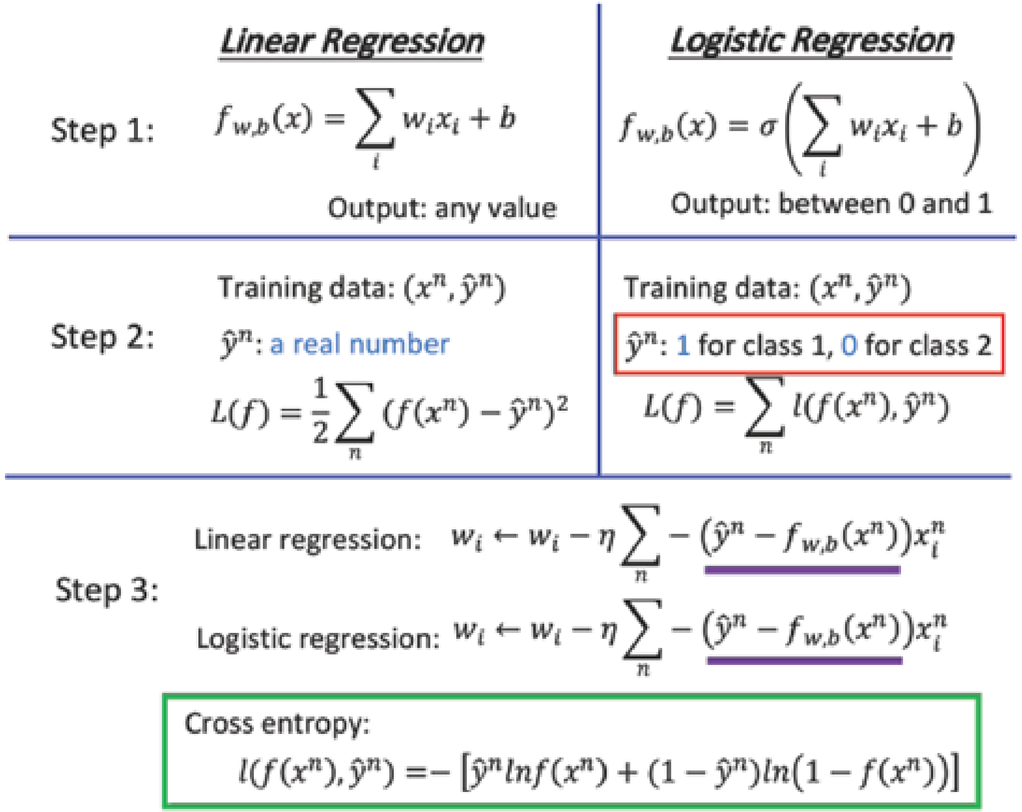




What is a Neural Network?



$$\hat{y} = wx + b$$



	Linear Regression	Logistic Regression
Step 1:	$f_{w,b}(x) = \sum_i w_i x_i + b$ Output: any value	$f_{w,b}(x) = \sigma\left(\sum_i w_i x_i + b\right)$ Output: between 0 and 1
Step 2:	Training data: $(x^n, \hat{y}^n)$ $\hat{y}^n$: a real number $L(f) = \frac{1}{2}\sum_n \left(f(x^n) - \hat{y}^n\right)^2$	Training data: $(x^n, \hat{y}^n)$ $\hat{y}^n$: 1 for class 1, 0 for class 2 $L(f) = \sum_n l\left(f(x^n), \hat{y}^n\right)$

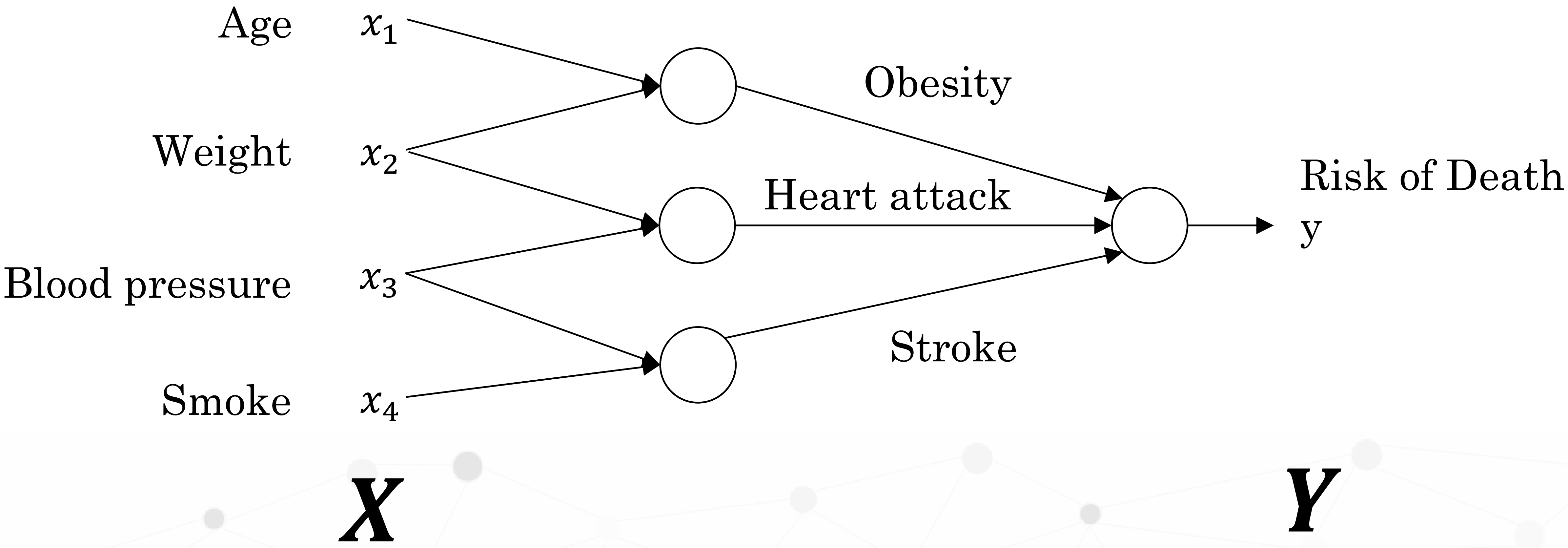
Step 3:

Linear regression: $w_i \leftarrow w_i - \eta \sum_n -\left(\hat{y}^n - f_{w,b}(x^n)\right)x_i^n$

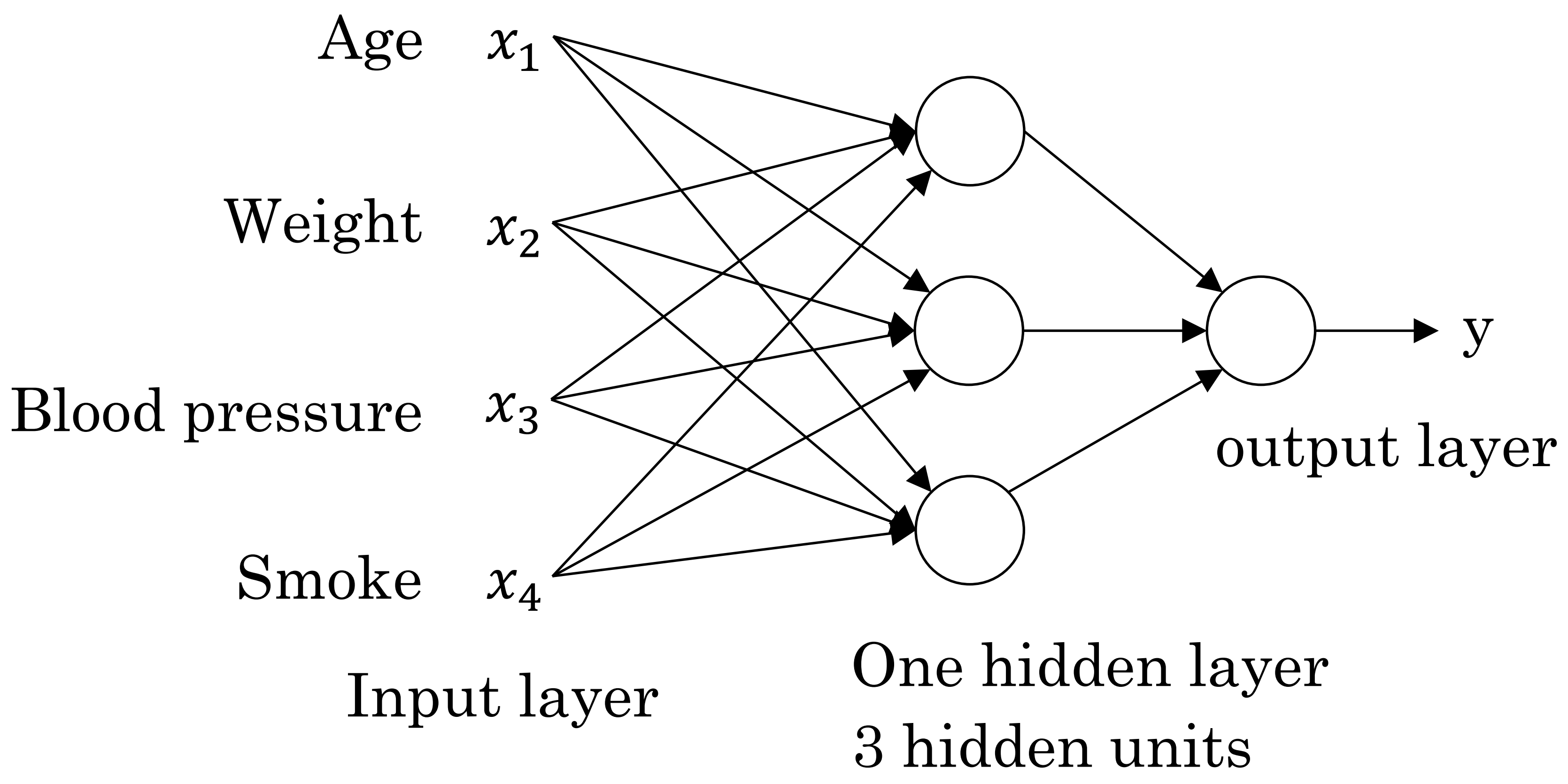
Logistic regression: $w_i \leftarrow w_i - \eta \sum_n -\left(\hat{y}^n - f_{w,b}(x^n)\right)x_i^n$

Cross entropy:
$$l\left(f(x^n), \hat{y}^n\right) = -\left[\hat{y}^n \ln f(x^n) + (1 - \hat{y}^n)\ln\left(1 - f(x^n)\right)\right]$$

Prognostic Model: Death Prediction



Layer Definition



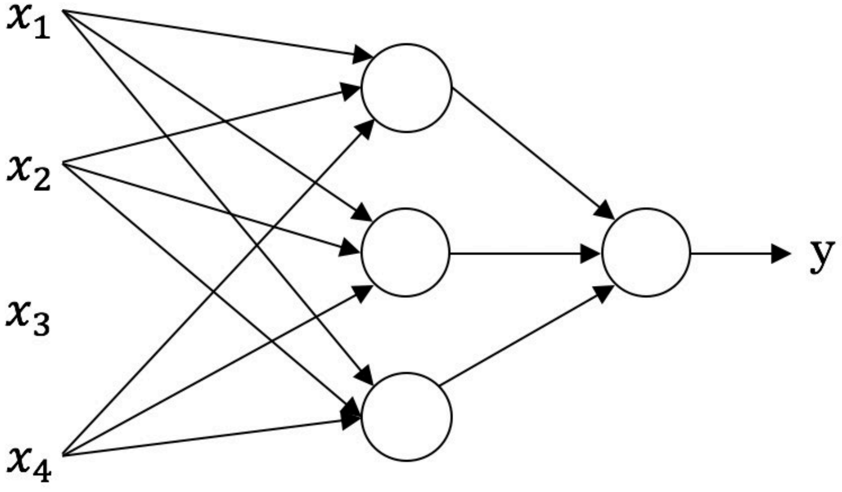
Supervised learning with enough data

$\boldsymbol{find\ f, where\ f(X) = Y}$ → $\hat{Y} = \boldsymbol{f(X)}$ and $\hat{Y} \approx \boldsymbol{Y}$

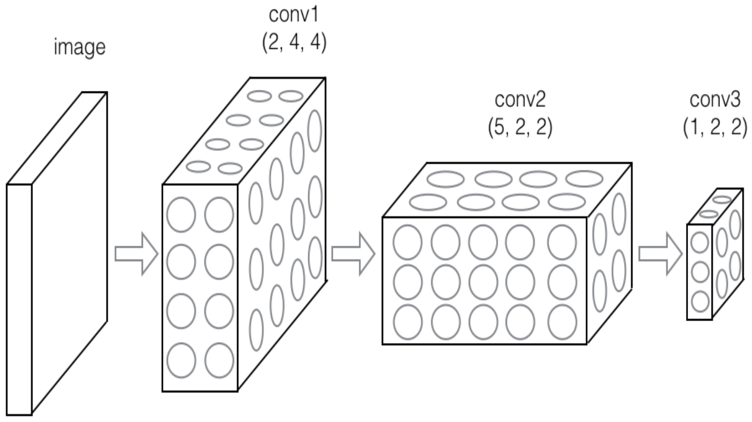
Supervised Learning

Input(x)	Output (y)	Application
Home features	Price	Real Estate
Ad, user info	Click on ad? (0/1)	Online Advertising (FB, YT)
Image	Object (1,…,1000)	Photo tagging
Audio	Text transcript	Speech recognition
English	Chinese	Machine translation
Medical Images	Position/Classification	Type of illness or Cancer border identification

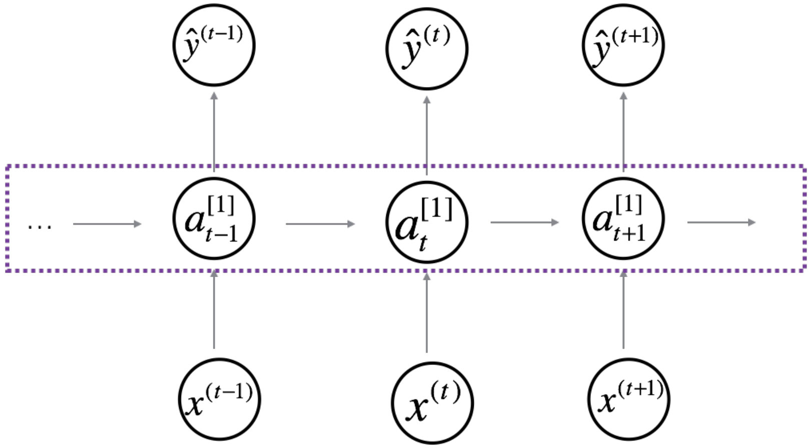
Neural Network examples



Standard NN



Convolutional NN



Recurrent NN

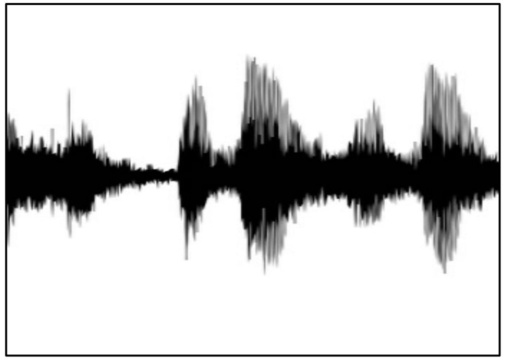
Supervised Learning

Structured Data

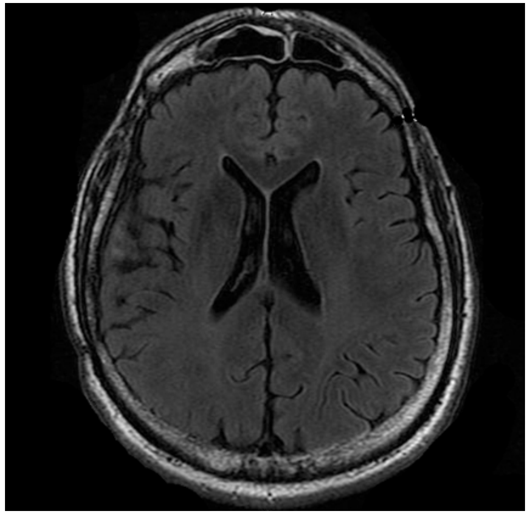
Age	Weight	⋯	Gender
50	72		Male
33	82		Female
18	66		Female
⋮	⋮		⋮
80	55		Male

User Age	Ad Id	⋯	Click
41	93242		1
80	93287		0
18	87312		1
⋮	⋮		⋮
27	71244		1

Unstructured Data



Audio

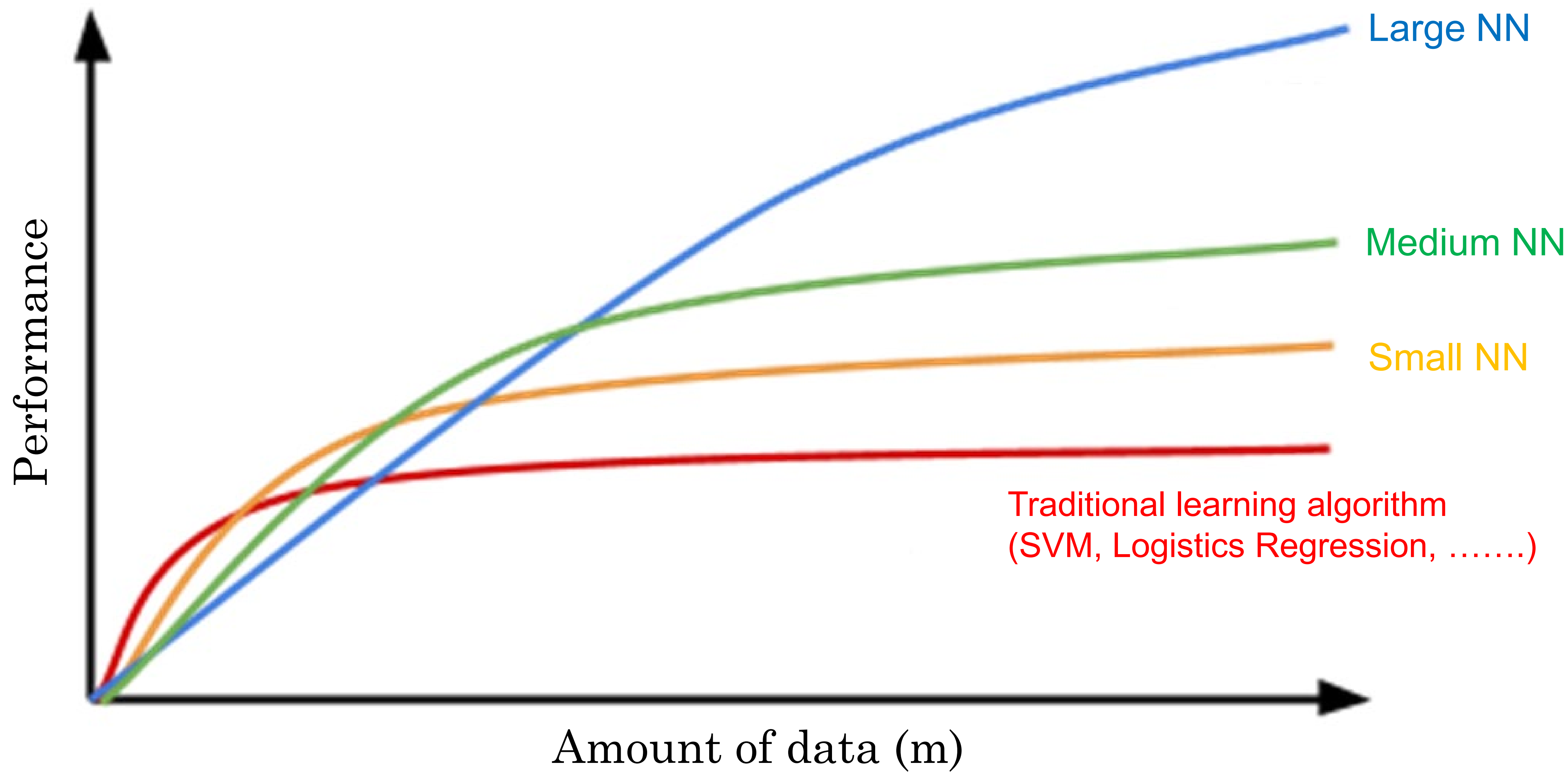


Four scores and seven years ago…

Text

Why is Deep Learning taking off?

Scale drives deep learning progress

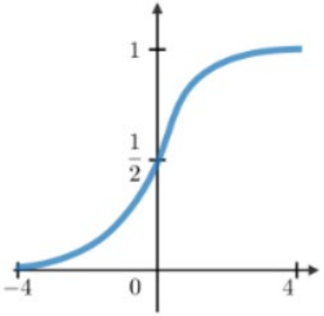


Scale drives deep learning progress

- Data
- Computation
- Algorithms

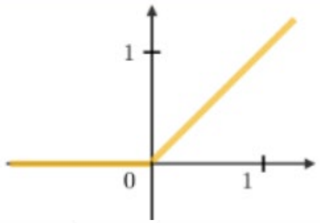
Sigmoid

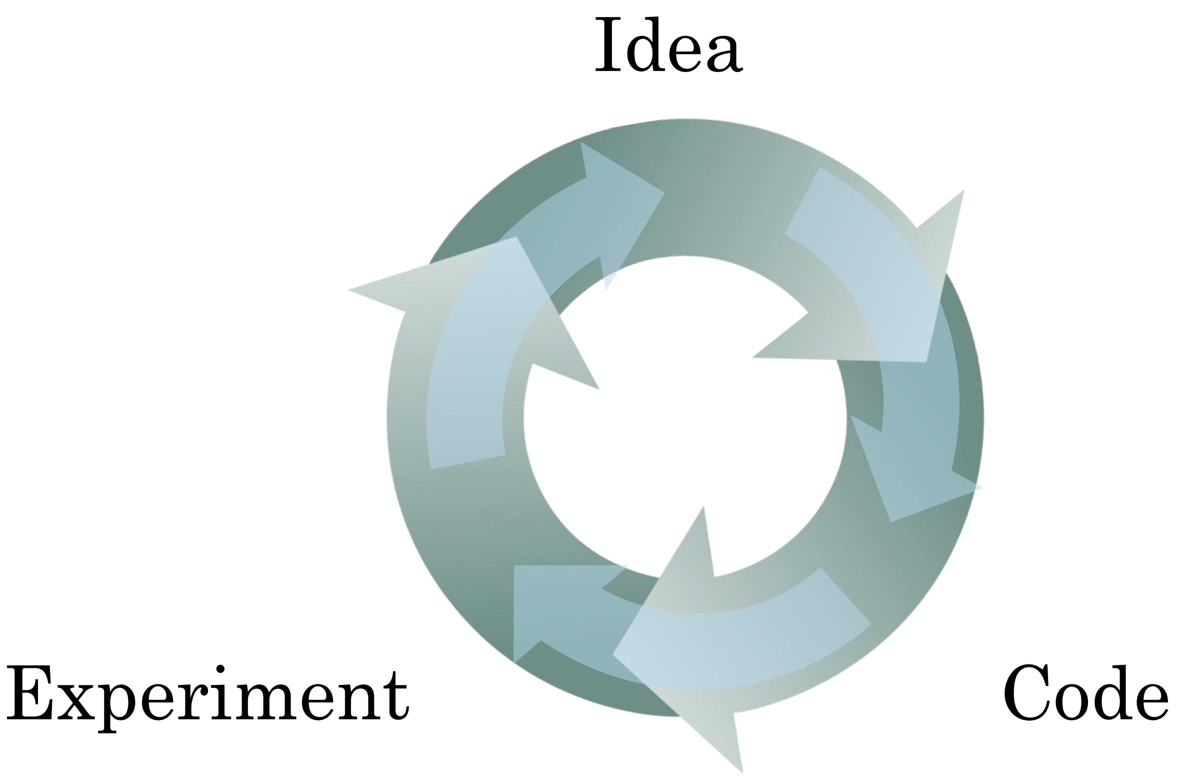
$$g(z) = \frac{1}{1+e^{-z}}$$



RELU

$$g(z) = \max(0, z)$$

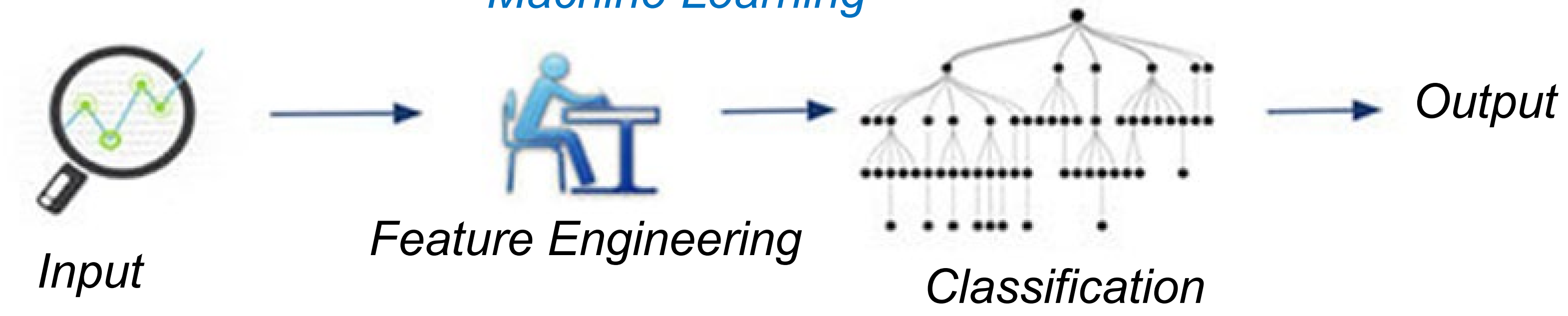




Why is Deep Learning taking off?

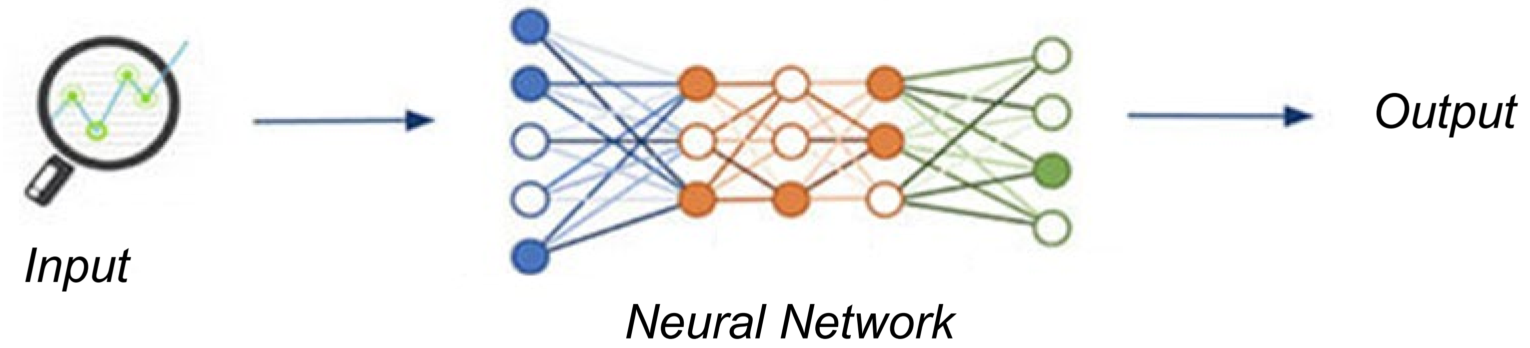
Machine learning vs Deep learning

Machine Learning



Traditional machine learning uses hand-crafted features, which is tedious and costly to develop.

Deep Learning



Deep learning learns hierarchical representation from the data itself, and scales with more data

Neural Networks Basics

Basics of Neural Network Programming

Binary Classification

64

64

1 (COVID19)

vs

0 (not COVID19)

64x64x1 = 4096

64

255	134	202	22
123	94	83	4
34	44	187	192
34	76	232	34
67	83	194	94

1

64

$$x = \begin{bmatrix} 255 \\ 133 \\ 202 \\ \vdots \end{bmatrix}$$

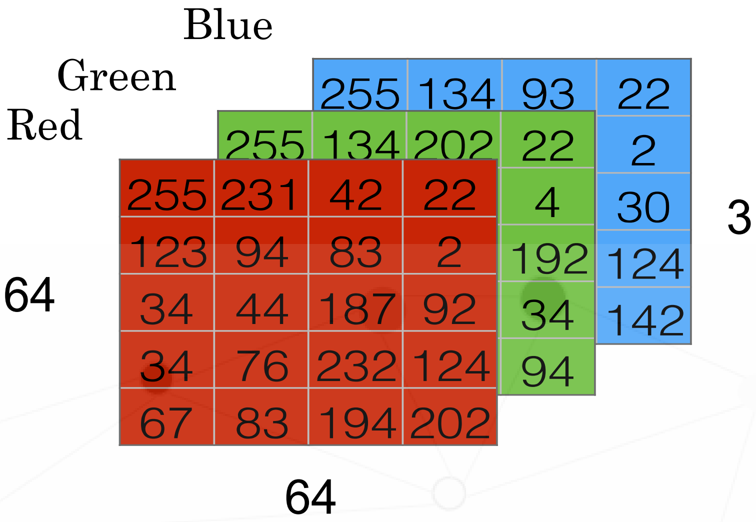
Dimension of the input features

n_x = 4096

x -> y

Binary Classification

→ 1 (cat) vs 0 (non cat)



$$x = \begin{array}{c}\left[\begin{array}{c} 255 \\ 231 \\ 42 \\ \vdots \\ 255 \\ 134 \\ 202 \\ \vdots \\ 255 \\ 134 \\ 93 \\ \vdots \end{array}\right]\end{array}\begin{array}{l} \left.\begin{array}{c} \\ \\ \\ \\ \end{array}\right\}\text{red} \\ \left.\begin{array}{c} \\ \\ \\ \\ \end{array}\right\}\text{green} \\ \left.\begin{array}{c} \\ \\ \\ \\ \end{array}\right\}\text{blue} \end{array}$$

64x64x3 = 12288

Dimension of the input features

n_x = 12288

x -> y

Notation

$(x\,,y)$ $\quad X \in \mathrm{R}^{n_x}$ $\quad Y \in \{0,1\}$ $\quad Y \in \mathrm{R}^1$

m training examples: $\{\left(x^{(1)}, y^{(1)}\right), (x^{(2)}, y^{(2)}),\ \ldots\ldots\ ,\ (x^{(m)}, y^{(m)})\}$

m = m_{train} $\qquad m_{test}$ = # of test examples

$$X = \begin{bmatrix} | & | & & | \\ x^{(1)} & x^{(2)} & \cdots & x^{(m)} \\ | & | & & | \end{bmatrix} \; \updownarrow n_x$$

⟵ *m columns* ⟶

$$Y = [\ y^{(1)}, y^{(2)}, y^{(3)}, \cdots\ y^{(m)}\ ]$$

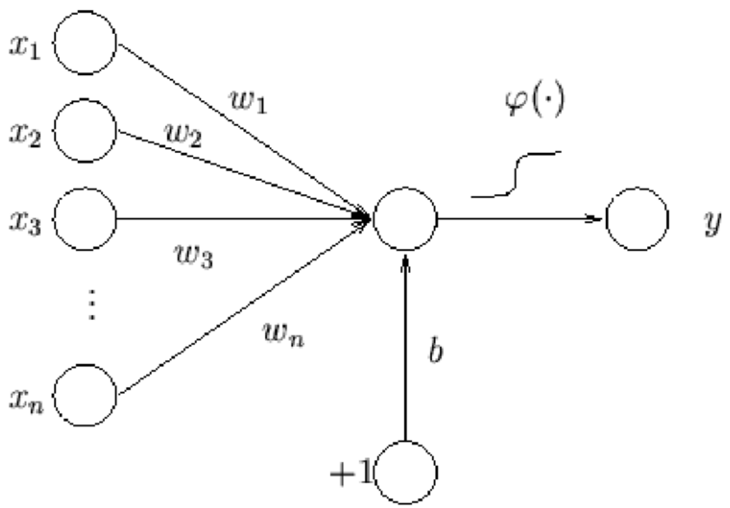
$Y \in \mathrm{R}^{1\times m}$

$Y.shape = (1, m)$

$X \in \mathrm{R}^{n_x \times m}$ $\qquad X.shape = (n_x, m)$

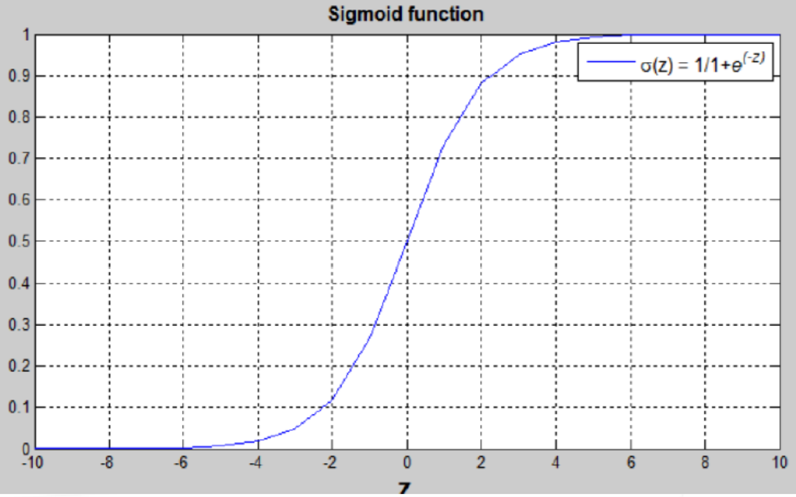
Logistic Regression

$Given\ x\ ,\ \hat{y} = P(y = 1|x)$, where $0 \leq \hat{y} \leq 1$



The parameters used in Logistic regression are:

- The input features vector: $x \in \mathbb{R}^{n_x}$, where n_x is the number of features
- The training label: $y \in 0,1$
- The weights: $w \in \mathbb{R}^{n_x}$, where n_x is the number of features
- The bias: $b \in \mathbb{R}$
- The output: $\hat{y} = \sigma(w^T x + b)$
- Sigmoid function: $s = \sigma(w^T x + b) = \sigma(z) = \frac{1}{1+e^{-z}}$



- If z is a large positive number, then $\sigma(z) = 1$
- If z is small or large negative number, then $\sigma(z) = 0$
- If $z = 0$, then $\sigma(z) = 0.5$

with enough data

$\boldsymbol{find\ f, where\ f(X) = Y}$ ➡ $\boxed{\hat{Y} = \boldsymbol{f(X)} \text{ and } \hat{Y} \approx \boldsymbol{Y}}$

How do we update *w* and *b*?

Logistic Regression cost function

$\hat{y} = \sigma(w^T x + b)$, where $\sigma(z) = \frac{1}{1+e^{-z}}$

Given $\{(x^{(1)}, y^{(1)}), \ldots, (x^{(m)}, y^{(m)})\}$, want $\hat{y}^{(i)} \approx y^{(i)}$.

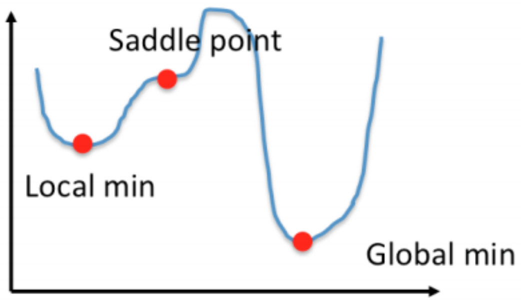
- Loss (error) function: $L(\hat{y}^{(i)}, y^{(i)}) = \frac{1}{2}(\hat{y}^{(i)} - y^{(i)})^2$

The loss function measures the discrepancy between the prediction ($\hat{y}^{(i)}$) and the desired output ($y^{(i)}$).
In other words, the loss function computes the error for a single training example.

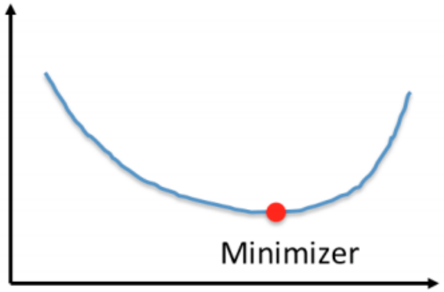
$L(\hat{y}^{(i)}, y^{(i)}) = -(y^{(i)} \log(\hat{y}^{(i)}) + (1 - y^{(i)}) \log(1 - \hat{y}^{(i)})$ **(Cross-entropy)**

- If $y^{(i)} = 1$: $L(\hat{y}^{(i)}, y^{(i)}) = -\log(\hat{y}^{(i)})$ where $\log(\hat{y}^{(i)})$ and $\hat{y}^{(i)}$ should be close to 1
- If $y^{(i)} = 0$: $L(\hat{y}^{(i)}, y^{(i)}) = -\log(1 - \hat{y}^{(i)})$ where $\log(1 - \hat{y}^{(i)})$ and $\hat{y}^{(i)}$ should be close to 0

Non-Convex



Convex



- Cost function

The cost function is the average of the loss function of the entire training set. We are going to find the parameters w *and* b that minimize the overall cost function.

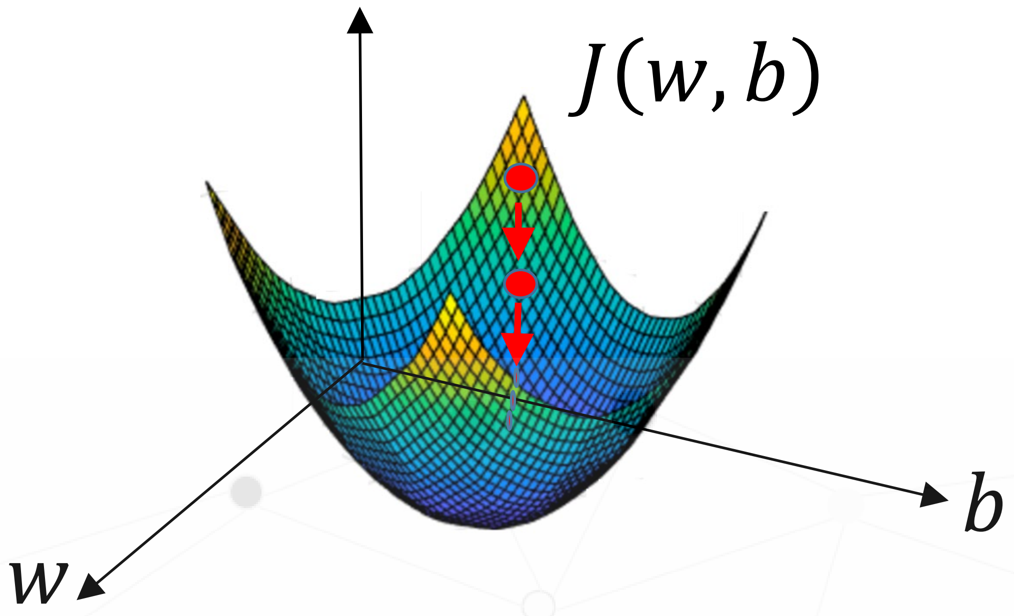
$$J(w, b) = \frac{1}{m}\sum_{i=1}^{m} L(\hat{y}^{(i)}, y^{(i)}) = -\frac{1}{m}\sum_{i=1}^{m}[(y^{(i)} \log(\hat{y}^{(i)}) + (1 - y^{(i)}) \log(1 - \hat{y}^{(i)})]$$

Gradient Descent

Logistic regression: $\hat{y} = \sigma(w^T x + b)$, $\sigma(z) = \frac{1}{1+e^{-z}}$

$$J(w,b) = \frac{1}{m}\sum_{i=1}^{m} \mathcal{L}(\hat{y}^{(i)}, y^{(i)}) = -\frac{1}{m}\sum_{i=1}^{m} y^{(i)} \log \hat{y}^{(i)} + (1 - y^{(i)}) \log(1 - \hat{y}^{(i)})$$

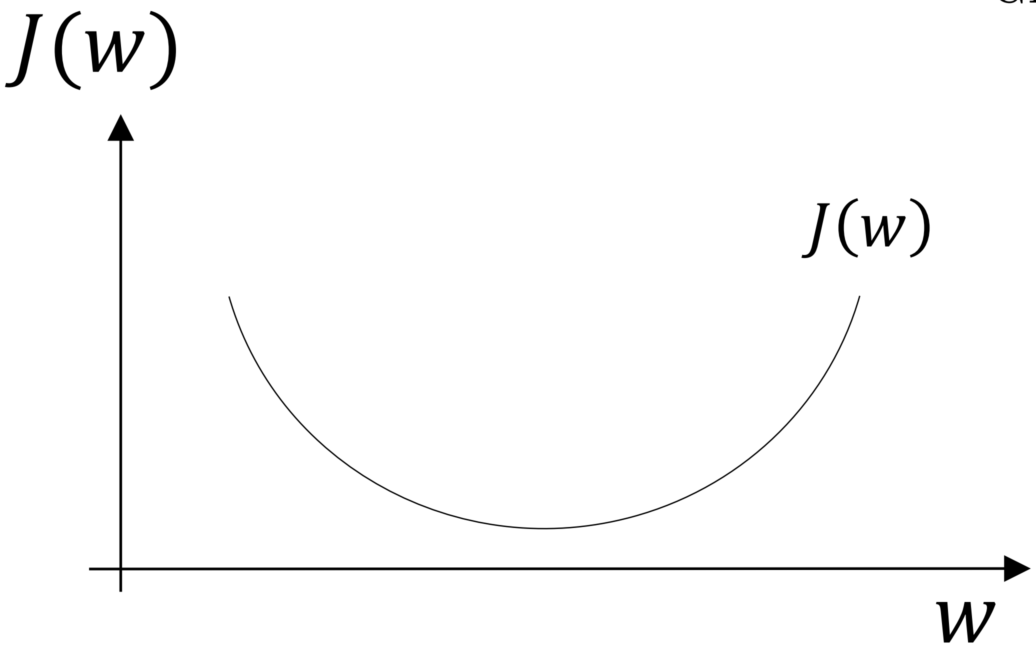
Want to find w, b that minimize $J(w,b)$



$$w := w - \alpha \frac{\partial J(w,b)}{\partial w}$$

$$b := b - \alpha \frac{\partial J(w,b)}{\partial b}$$

Gradient Descent



Learning rate > 0

Repeat {

$$w := w - \alpha \frac{dJ(w)}{dw}$$

}

variable dw in the coding assignment

w := w – αdw

$J(w, b)$

$$w := w - \alpha \frac{\partial J(w, b)}{\partial w}$$

$$b := b - \alpha \frac{\partial J(w, b)}{\partial b}$$

Cross Entropy vs Square Error

Logistic Regression + Square Error

Step 1: $f_{w,b}(x) = \sigma\left(\sum_i w_i^T x_i + b\right)$

Step 2: Training data: $(x^n, \hat{y}^n)$, $\hat{y}^n$: 1 for class 1, 0 for class 2

$$L(f) = \frac{1}{2}\sum_n \left(f_{w,b}(x^n) - \hat{y}^n\right)^2$$

Step 3:

$$\frac{\partial \left(f_{w,b}(x) - \hat{y}\right)^2}{\partial w_i} = 2\left(f_{w,b}(x) - \hat{y}\right)\frac{\partial f_{w,b}(x)}{\partial z}\frac{\partial z}{\partial w_i}$$

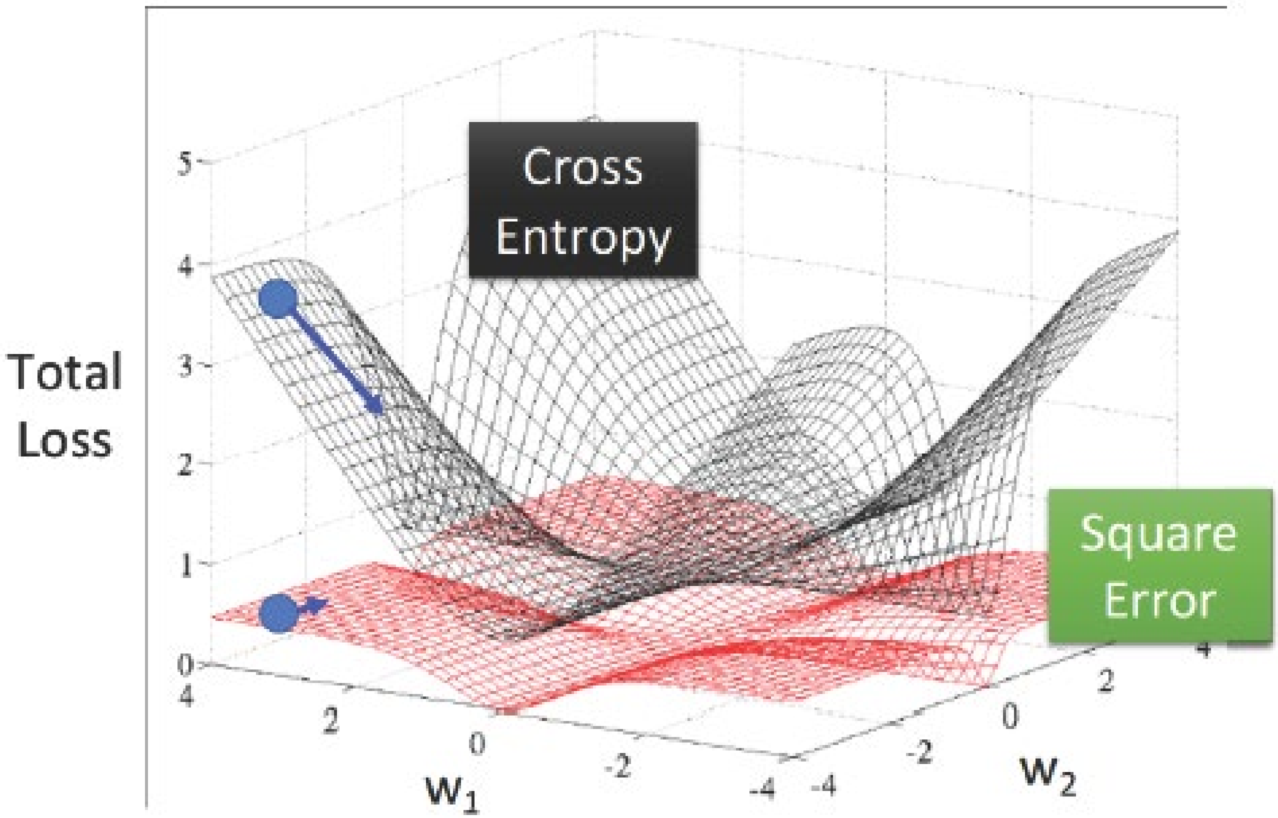
$$= 2\left(f_{w,b}(x) - \hat{y}\right) f_{w,b}(x)\left(1 - f_{w,b}(x)\right) x_i$$

$\hat{y}^n = 1$ If $f_{w,b}(x^n) = 1$ (close to target) ➡ $\partial L / \partial w_i = 0$

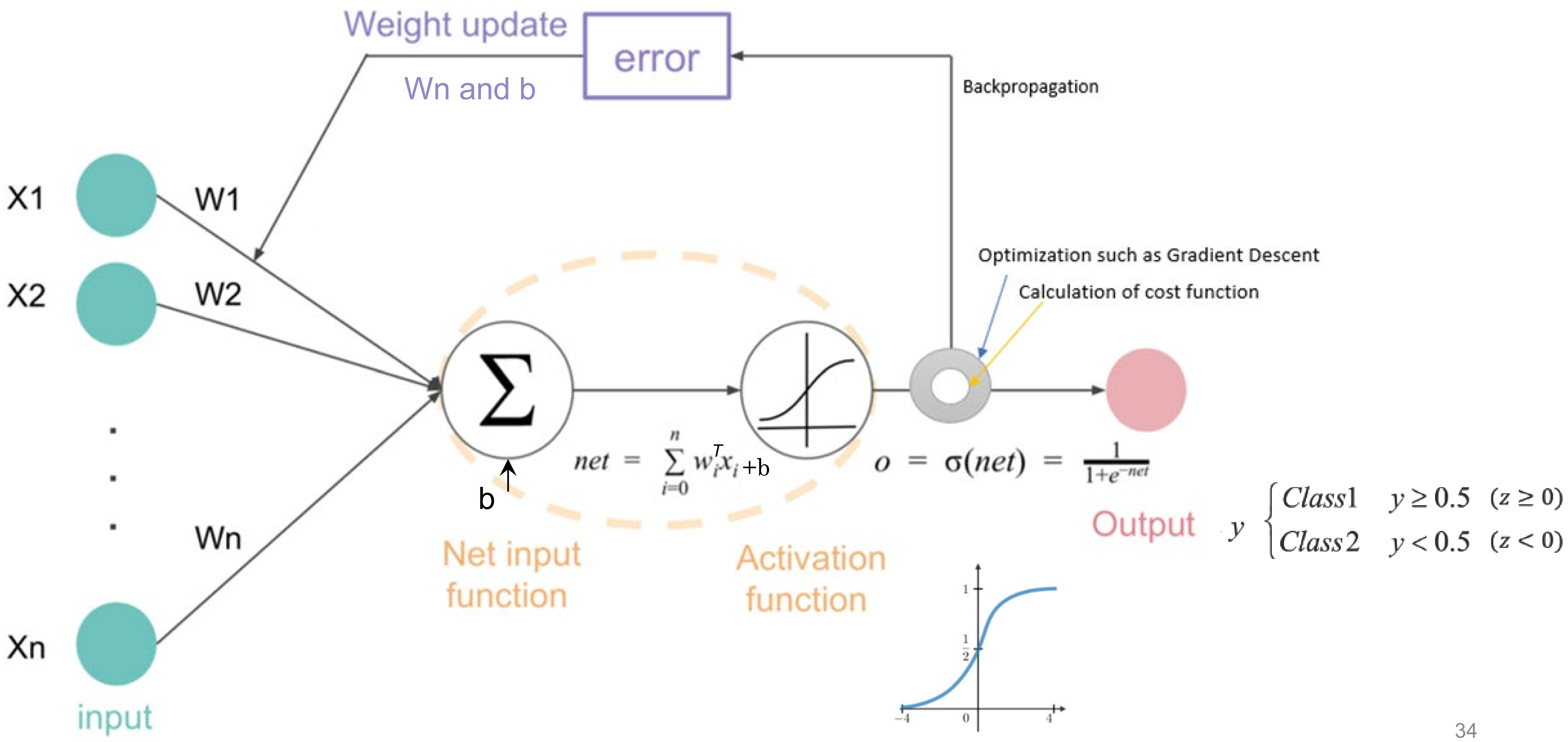
If $f_{w,b}(x^n) = 0$ (far from target) ➡ $\partial L / \partial w_i = 0$

$\hat{y}^n = 0$ If $f_{w,b}(x^n) = 1$ (far from target) ➡ $\partial L / \partial w_i = 0$

If $f_{w,b}(x^n) = 0$ (close to target) ➡ $\partial L / \partial w_i = 0$



Computations of a neural network

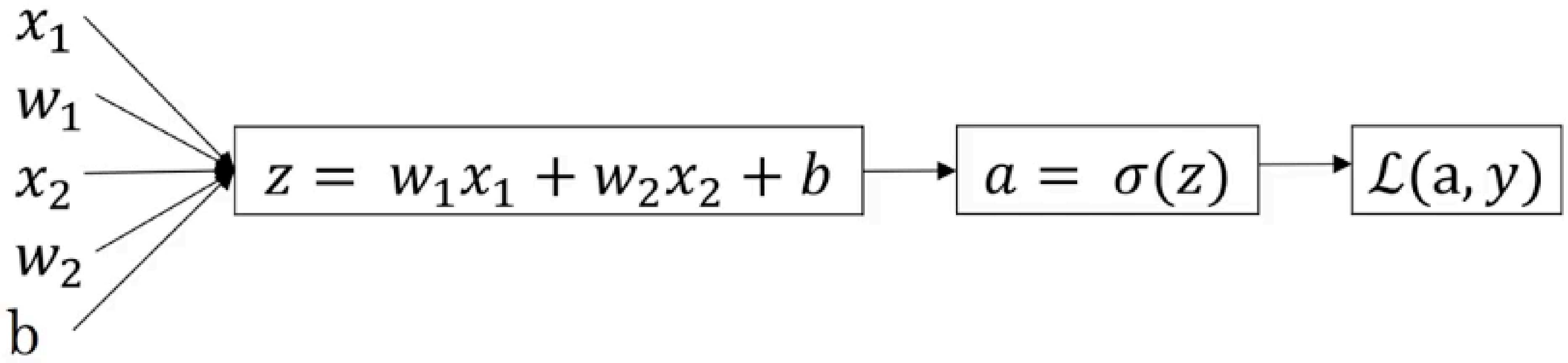


Computation Graph: Forward Propagation

Logistic regression: $\hat{y} = \ \sigma(w^T x + b) = z,$

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

$$J(w,b) = \frac{1}{m}\sum_{i=1}^{m}\mathcal{L}(\hat{y}^{(i)}, y^{(i)}) = -\frac{1}{m}\sum_{i=1}^{m} y^{(i)}\log\hat{y}^{(i)} + (1-y^{(i)})\log(1 \ -\hat{y}^{(i)})$$



Computation Graph: Backward Propagation

$$z = w^T x + b \qquad \hat{y} = a = \sigma(z) \qquad L(a,y) = -(y\log(a) + (1-y)\log(1-a))$$

x_1, w_1, x_2, w_2, b $\rightarrow$ $\boxed{z = w_1 x_1 + w_2 x_2 + b}$ $\rightarrow$ $\boxed{a = \sigma(z)}$ $\rightarrow$ $\boxed{\mathcal{L}(a,y)}$

$$w^T = [\, w_1 \;\; w_2 \,] \qquad \sigma(z) = \frac{1}{1+e^{-z}}$$

We want $\frac{\partial L(w,b)}{\partial b}$ and $\frac{\partial L(w,b)}{\partial w_i}$ $\xrightarrow{\text{chain rule}}$ $\frac{\partial L}{\partial w_i} = \frac{\partial L}{\partial a}\frac{\partial a}{\partial z}\frac{\partial z}{\partial w_i}$

$$\frac{\partial L}{\partial a} = -\frac{y}{a} + \frac{1-y}{1-a} \qquad \frac{d\sigma(z)}{dz} = \sigma(z)(1-\sigma(z))$$

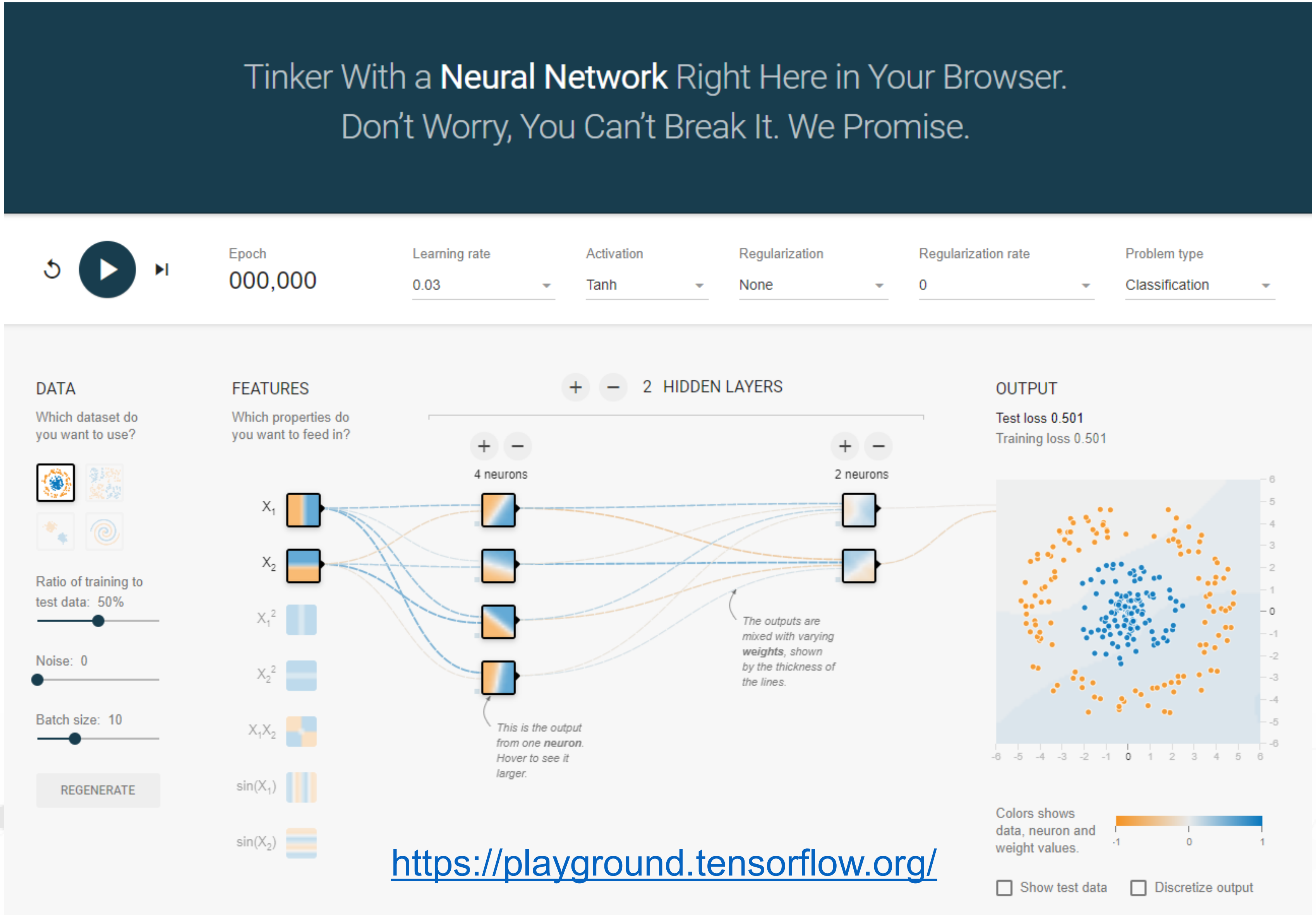
$$\frac{\partial L}{\partial w_i} = \frac{\partial L}{\partial z}\frac{\partial z}{\partial w_i} \qquad \frac{\partial L}{\partial z} = \frac{\partial L}{\partial a} \cdot \frac{da}{dz} \qquad \frac{\partial L}{\partial z} = \left(-\frac{y}{a} + \frac{1-y}{1-a}\right) \cdot a(1-a) = -y(1-a) + a(1-y) = -y + a$$

$$\frac{\partial L}{\partial w_1} = \frac{\partial L}{\partial z} \cdot \frac{\partial z}{\partial w_1} = \frac{\partial L}{\partial z} \cdot x_1 \qquad \frac{\partial L}{\partial w_2} = \frac{\partial L}{\partial z} \cdot \frac{\partial z}{\partial w_2} = \frac{\partial L}{\partial z} \cdot x_2 \qquad \frac{\partial L}{\partial b} = \frac{\partial L}{\partial z} \cdot \frac{\partial z}{\partial b} = \frac{\partial L}{\partial z}$$

Update $w_1 := w_1 - \alpha \frac{\partial L}{\partial w1}$, $w_2 := w_2 - \alpha \frac{\partial L}{\partial w2}$, $b := b - \alpha \frac{\partial L}{\partial b}$

* Automatic Differentiation in Machine Learning: a Survey https://arxiv.org/abs/1502.05767

Tensorflow Playground



https://playground.tensorflow.org/

Next:
Lab
Practice

Next Lecture

- Shallow Neural Networks (Course 1 Week 3)

- Deep Neural Networks (Course 1 Week 4)