Before & After

The AI Winter

Session #2: After the AI Winter

Microsoft Student Partners

Evangelist Younjoon Chung

Goals

- Brief history of A.I.
- Introduction to deep learning
- Build your own network!

Acknowledgement

- Andrew Ng's ML class
  - https://class.coursera.org/ml-003/lecture
  - http://www.holehouse.org/mlclass (note)
- 모두의 딥러닝
  - https://hunkim.github.io/ml
- http://www.andreykurenkov.com/writing/a-brief-history-of-neural-nets-and-deep-learning/
- 김지환 교수님 System programming

Review:
machine learning

- Limitations of explicit programming
  - Spam filter: many rules
  - Automatic driving: too many rules

- Machine learning: "Field of study that gives computers the ability to learn **without being explicitly programmed**" Arthur Samuel (1959)

Machine learning: formal definition

- A computer program is said to learn from experience ***E*** with respect to some class of tasks ***T*** and performance measure ***P*** if its performance at tasks in ***T***, as measured by ***P***, improves with experience **E**. Tom M. Mitchell

- ***E:*** experience
- ***T:*** Task
- ***P:*** performance measure

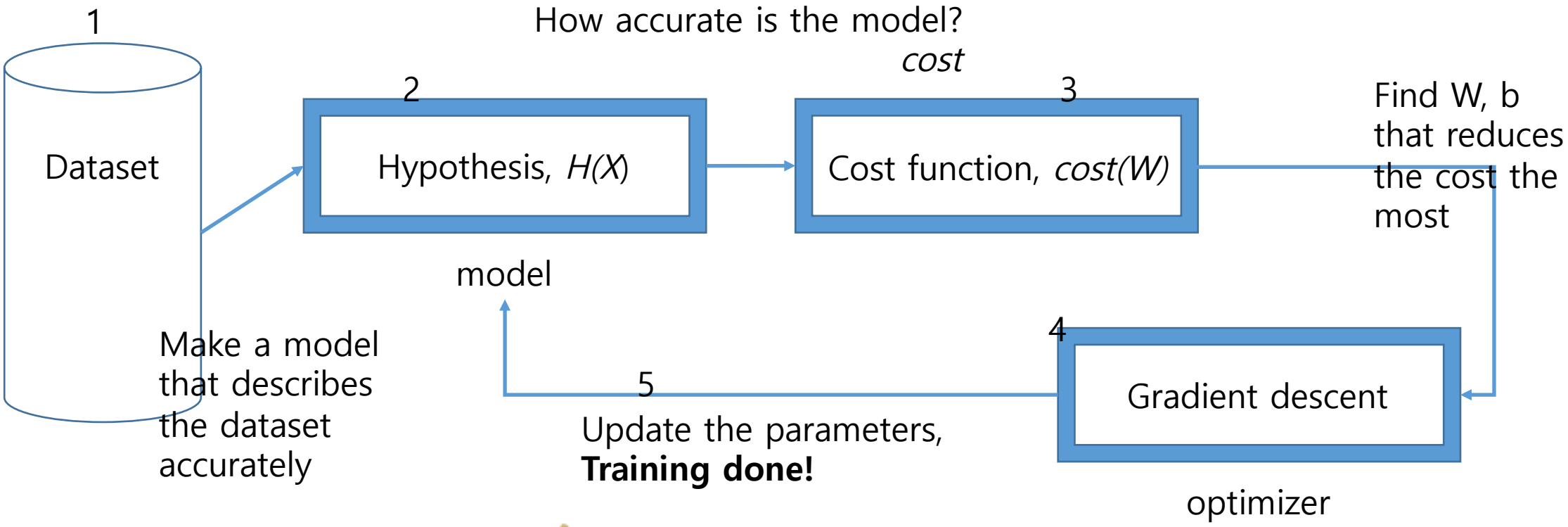
Review:
types of supervised learning

- Predicting final exam score based on time spent
  - **Regression**

- Pass/non-pass based on time spent
  - **Binary classification**

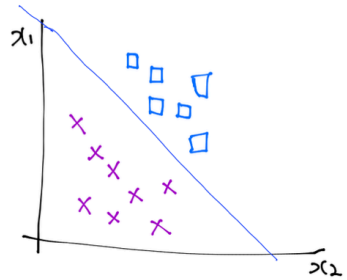
- Letter grade(A, B, C, D, and F) based on time spent
  - **Multi-label classification**

Review: regression training in a nutshell



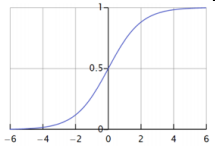
So module! Many simple!
Wow.

Review:
binary classification training in a nutshell



Linear model is prone to outliers. Why?

vs



How accurate is the model?

cost

1

Dataset

2

H(X)

Linear model

$\mathbb{R}$

3

Sigmoid

Activation function

(0, 1)

4

Cost function, *cost(W)*

Find W, b that reduces the cost the most

5

Gradient descent

optimizer

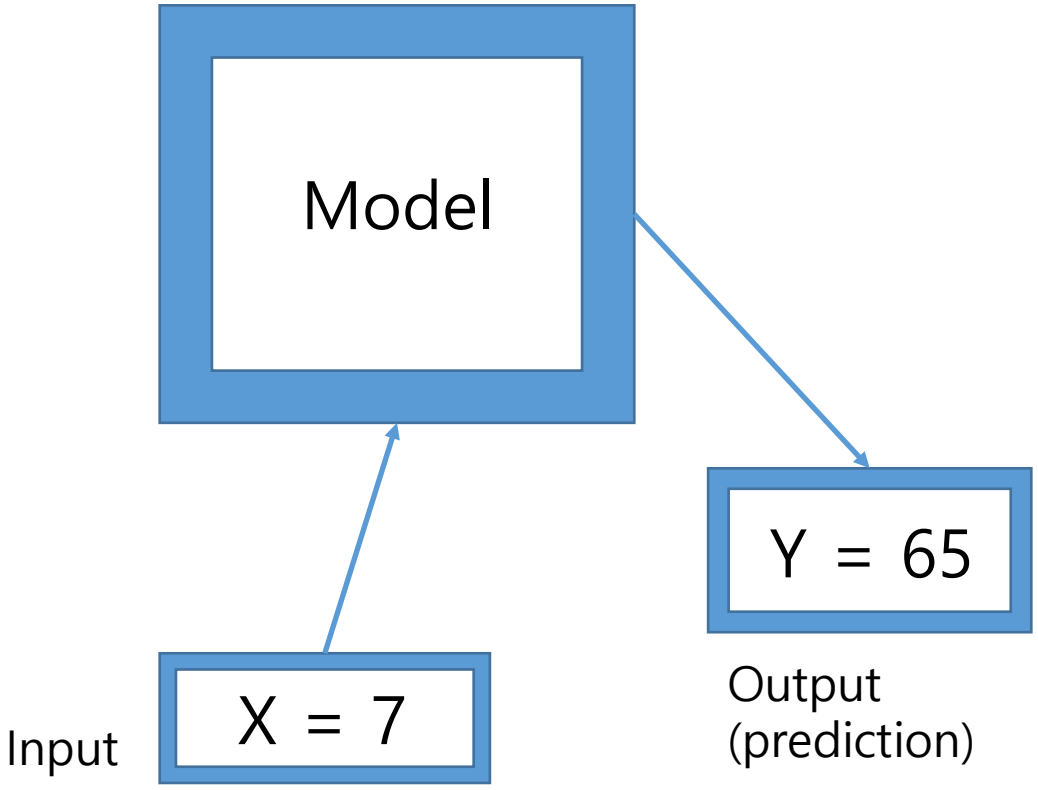
6

Update the parameters,
Training done!

Make a model that describes the dataset accurately

So module! Many simple!
Wow.

Issues in machine learning: performance evaluation



Is this good?

In supervised learning:
compare prediction
with the given label

Which input dataset should we use for evaluation?

Issues in machine learning: evaluation using training set?

Size	Price
2104	400
1600	330
2400	369
1416	232
3000	540
1985	300
1534	315

100% correct (accuracy)

Can memorize

Issues in machine learning: training and test sets

Size	Price
2104	400
1600	330
2400	369
1416	232

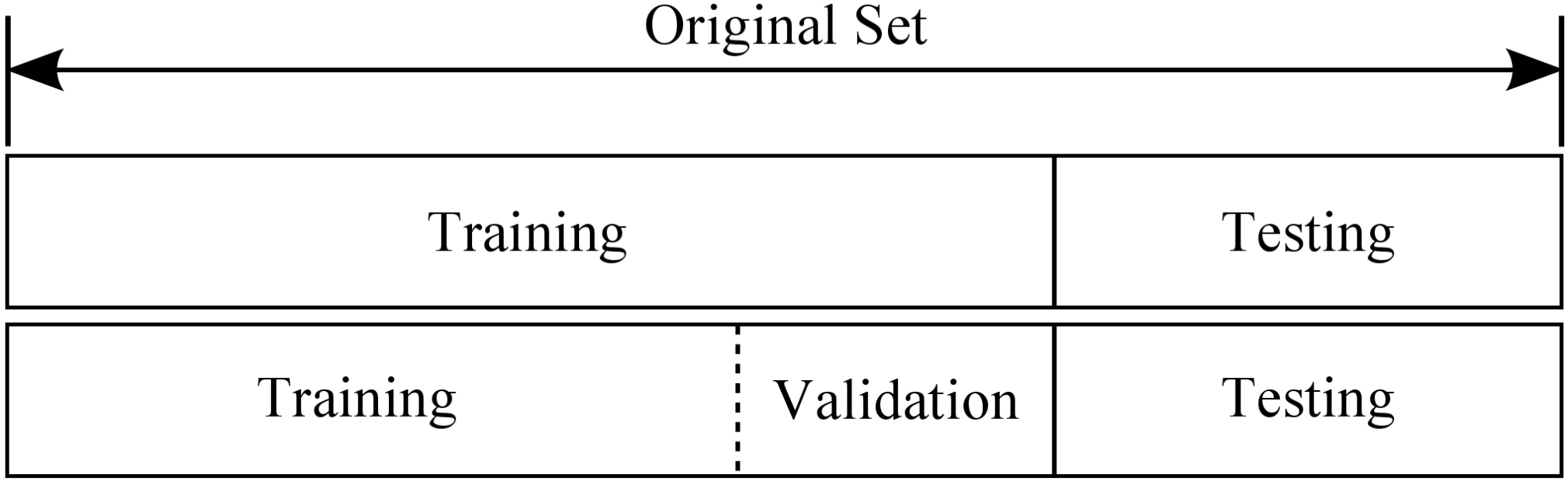
Training data

Size	Price
3000	540
1985	300
1534	315

Test data
(not visible during the training step)

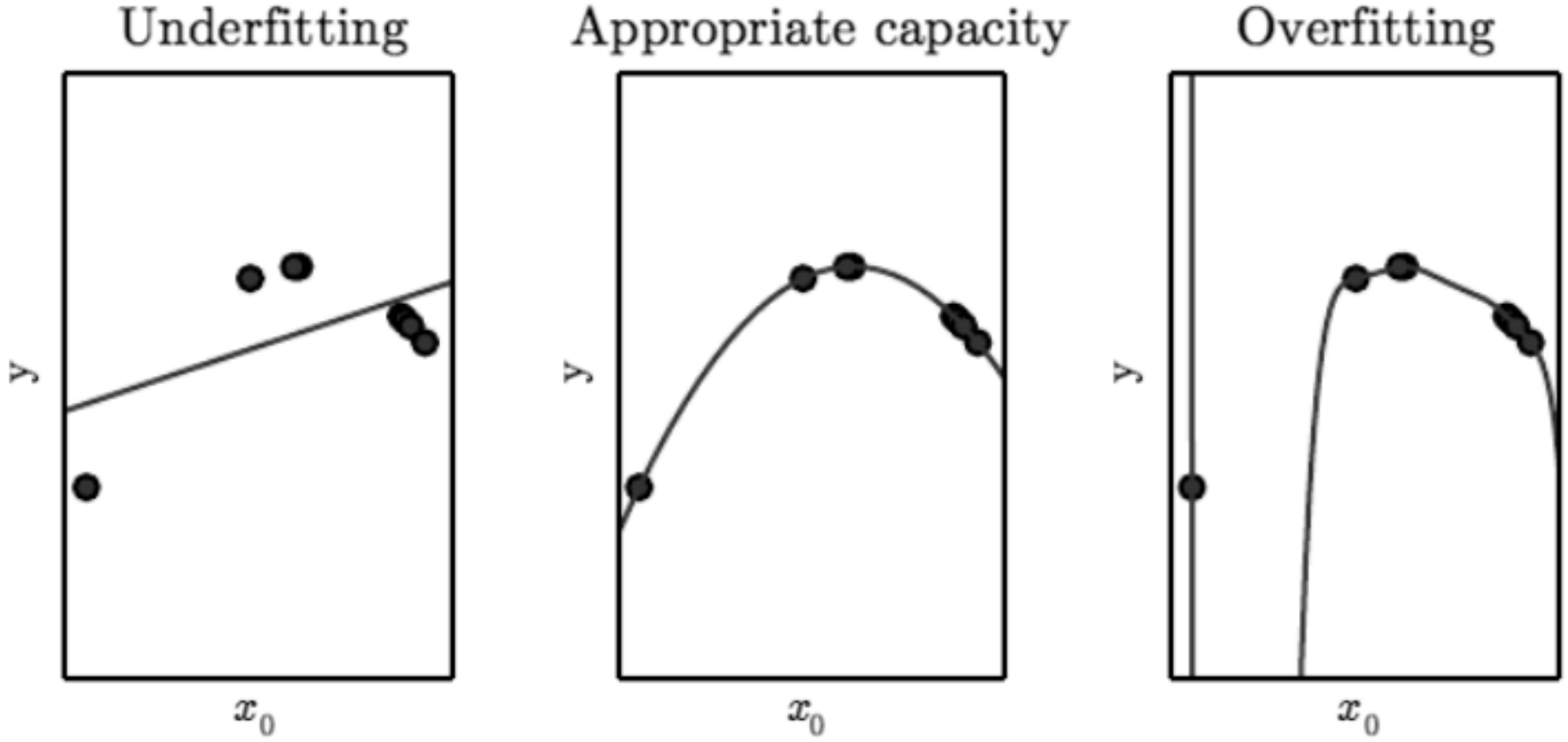
1. Split the training dataset into two(ex. 70%, 30%)
2. Use the first set for actual training(ex. 70%)
3. Use the second set for testing accuracy on unseen data

Issues in machine learning: training and test sets



Issues in machine learning: overfitting

This doesn't generalize well!

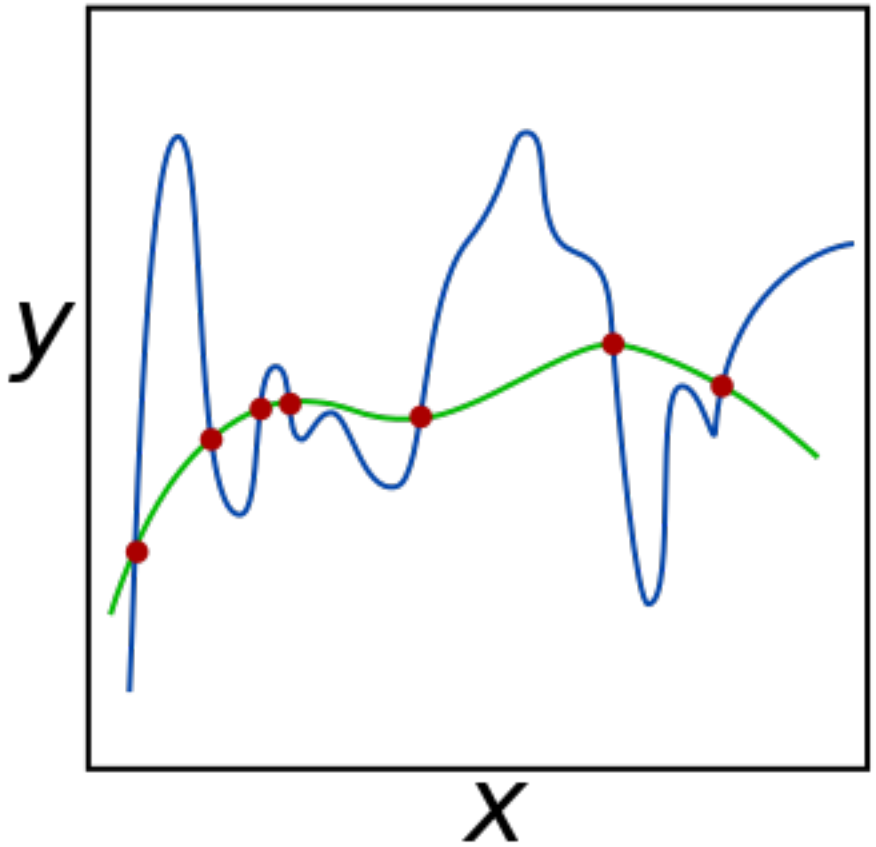


Issues in machine learning: solutions for overfitting

- More training data
- Reduce the number of features
- Regularization

Issues in machine learning: regularization

- Let's not have too big numbers in the weight



Occam's razor

Issues in machine learning: regularization

- Keep all the features, but reduce the magnitude/values of weights
- Works well when we have a lot of features

Issues in machine learning: regularization

- Lambda determines how fit (larger value means smaller weight)

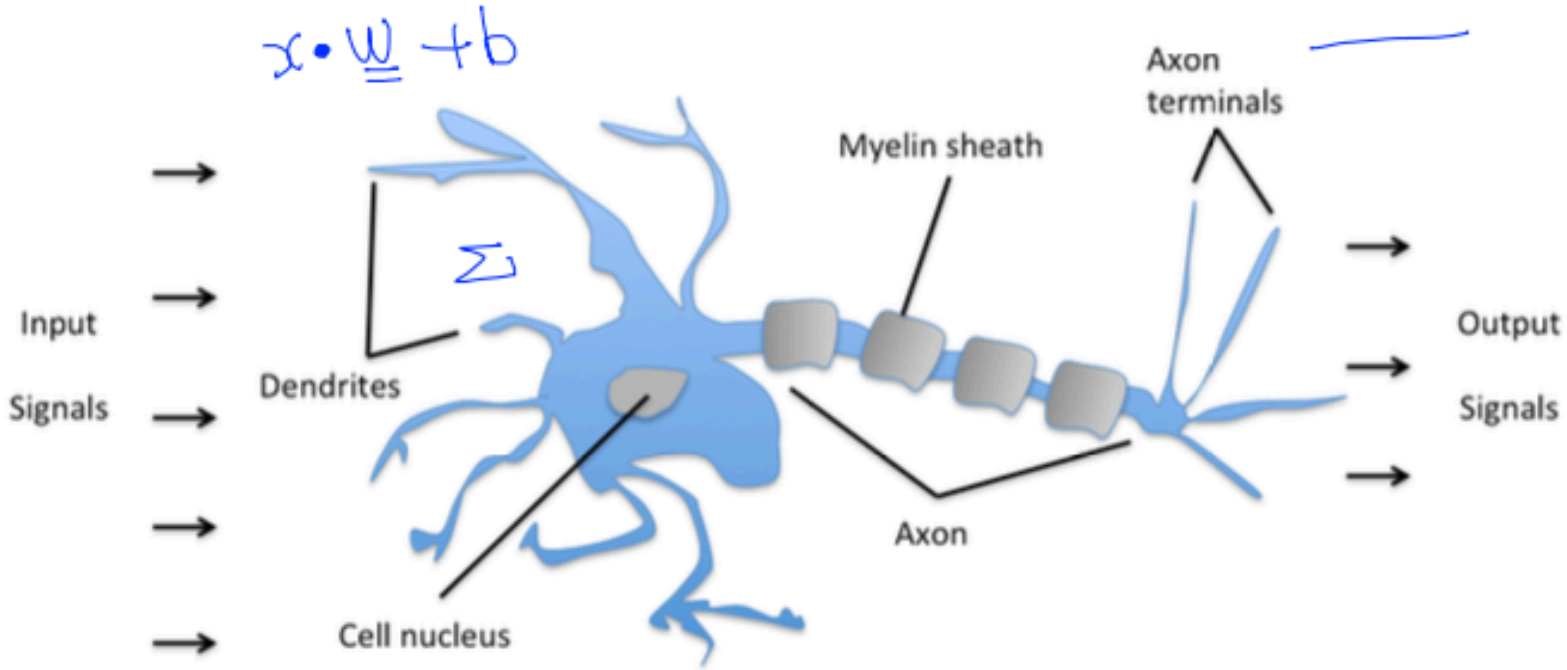
- Note that larger weight means more fluctuations!

Original cost function for linear regression

$$J(\theta) = \frac{1}{2m}\left[\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2 + \lambda \sum_{j=1}^{n}\theta_j^2\right]$$

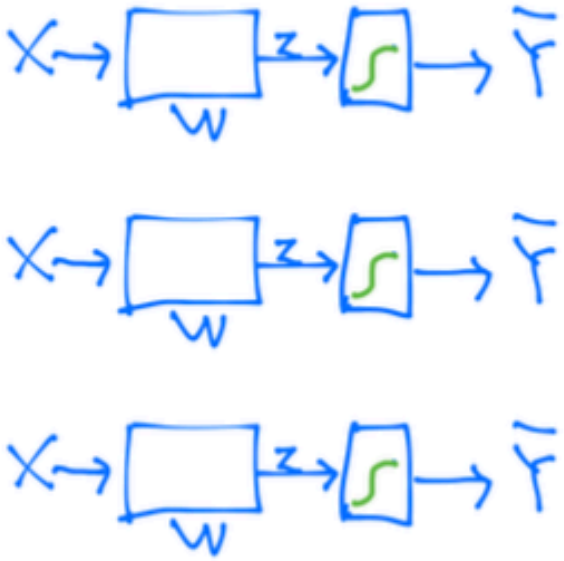
Regularization term

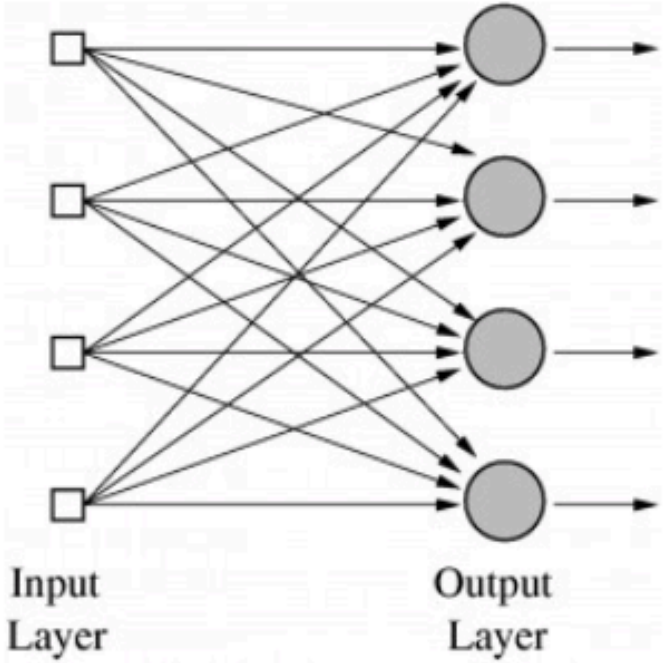
History of artificial intelligence



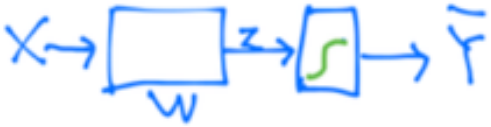
Schematic of a biological neuron.

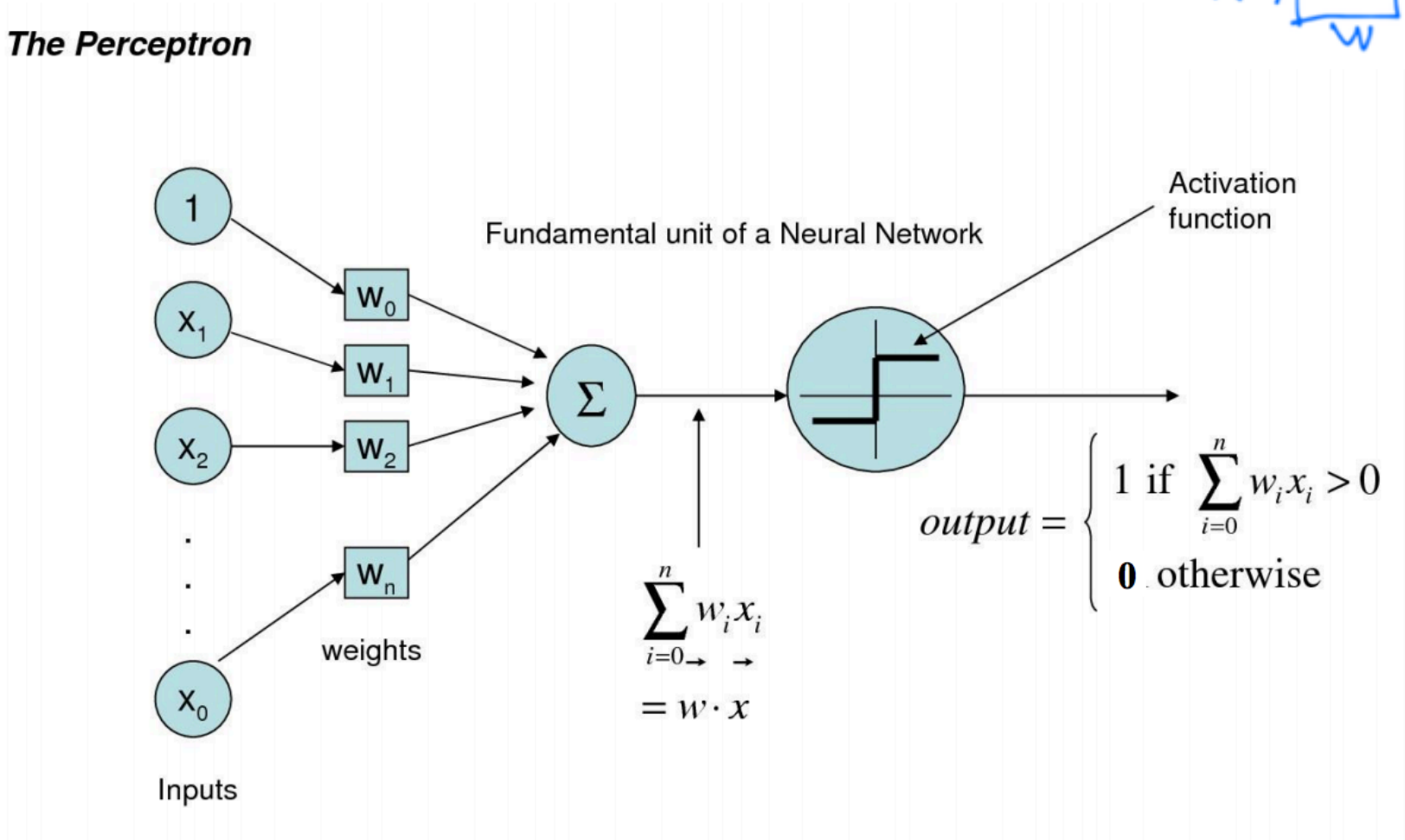
Review:
logistic regression units



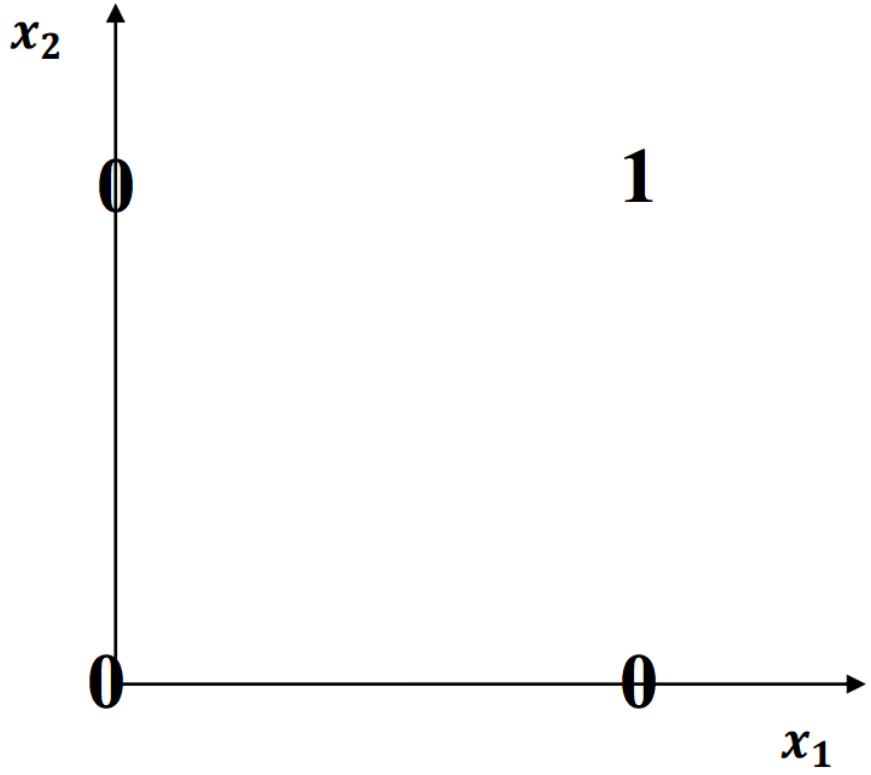


The perceptron



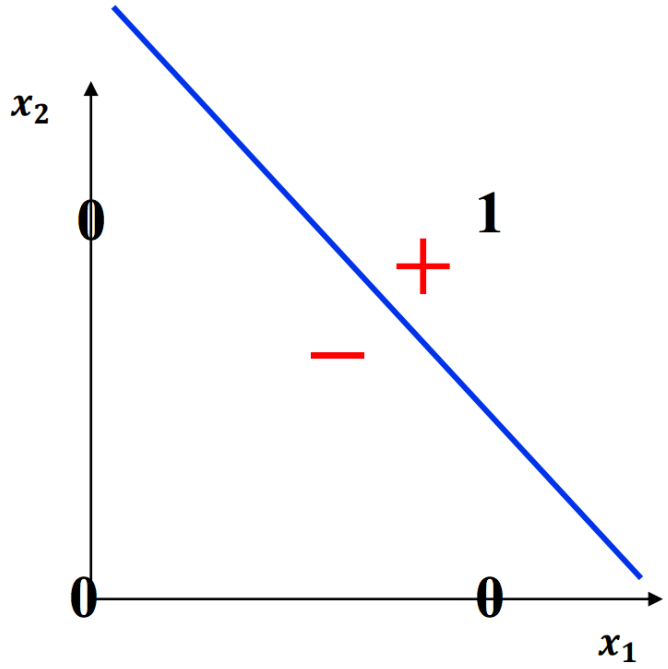


Perceptron: AND problem



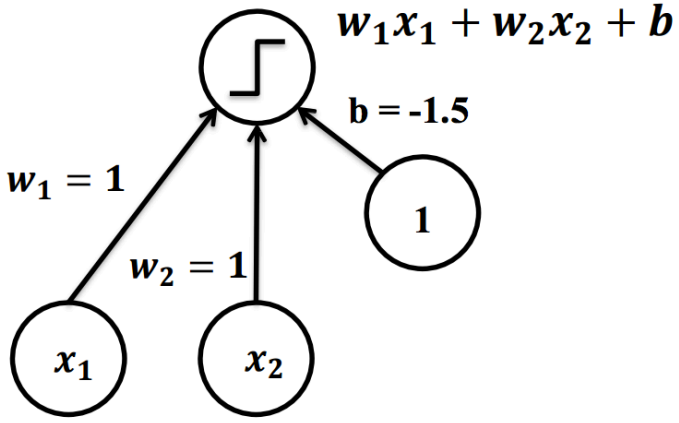
Input		Output
x_1	x_2	y
0	0	**0**
0	1	**0**
1	0	**0**
1	1	**1**

Perceptron: AND problem

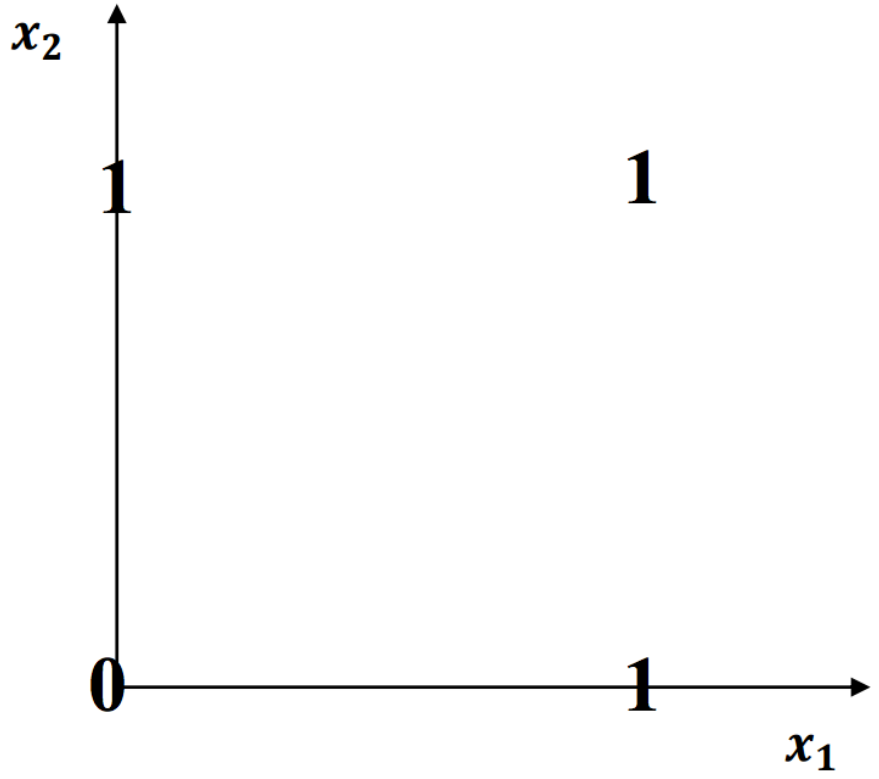


Input		Output
x_1	x_2	y
0	0	**0**
0	1	**0**
1	0	**0**
1	1	**1**

$$f(x_1, x_2) = x_1 + x_2 - 1.5$$

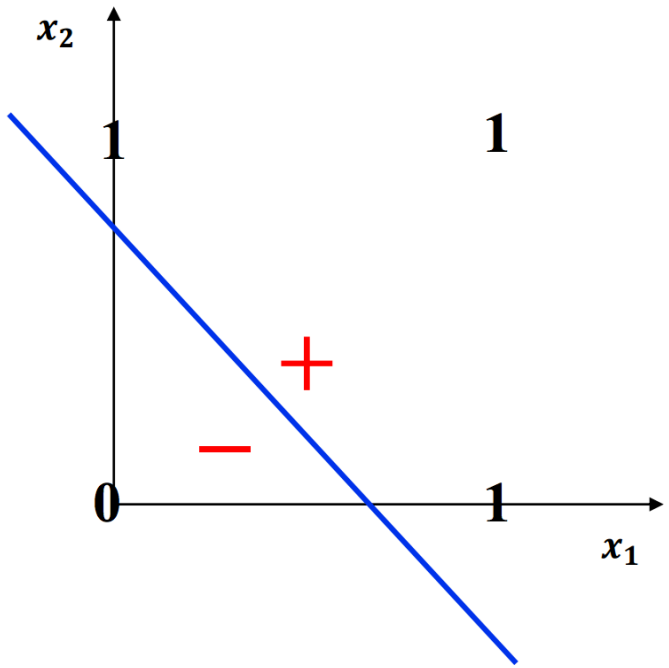


Perceptron: OR problem



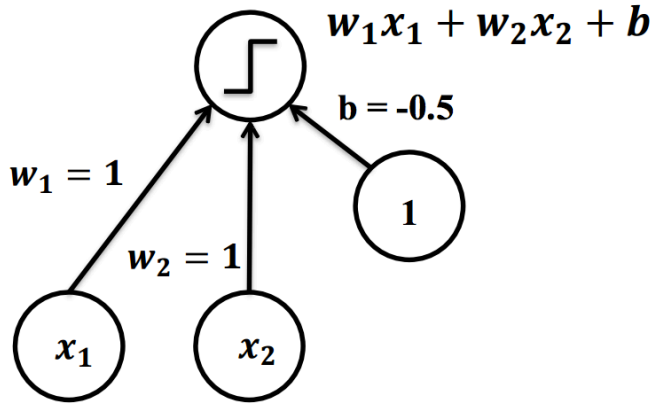
Input		Output
x_1	x_2	y
0	0	**0**
0	1	**1**
1	0	**1**
1	1	**1**

Perceptron: OR problem

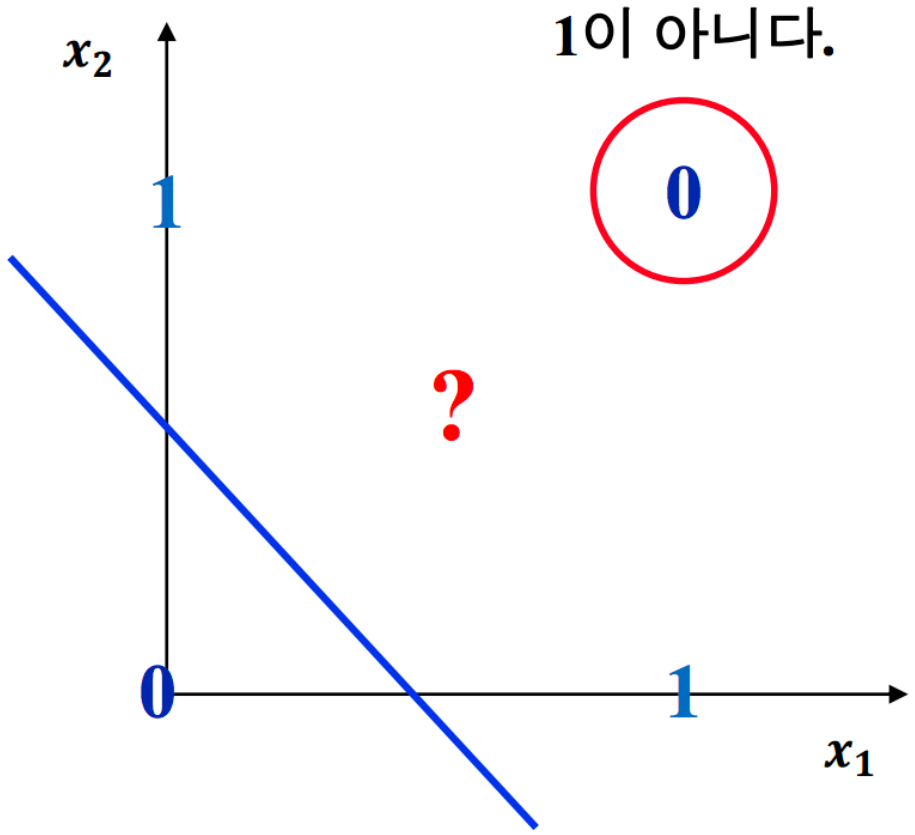


$$f(x_1, x_2) = x_1 + x_2 - 0.5$$

Input		Output
x_1	x_2	y
0	0	**0**
0	1	**1**
1	0	**1**
1	1	**1**



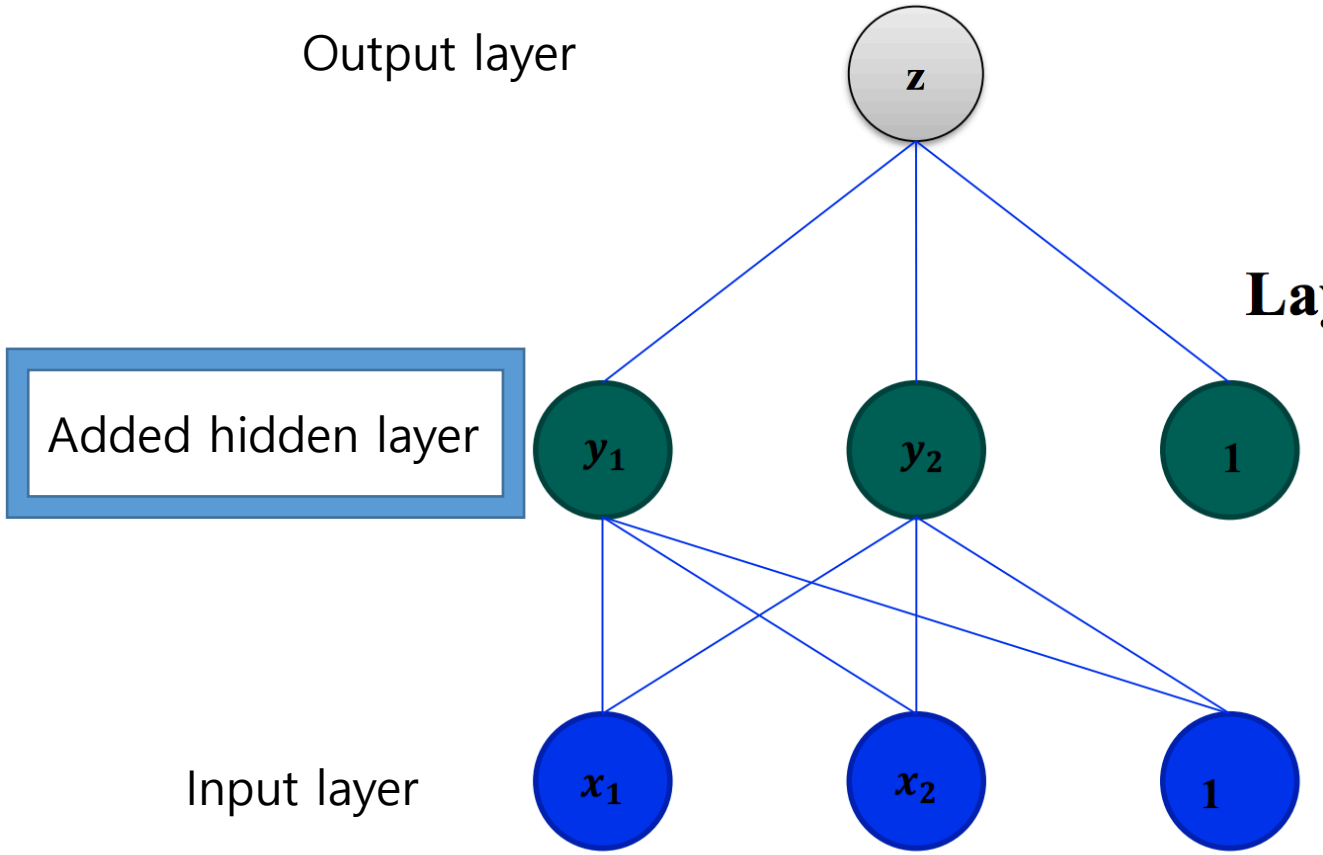
Perceptron: XOR problem



Input		Output
x_1	x_2	y
0	0	**0**
0	1	**1**
1	0	**1**
1	1	**0**

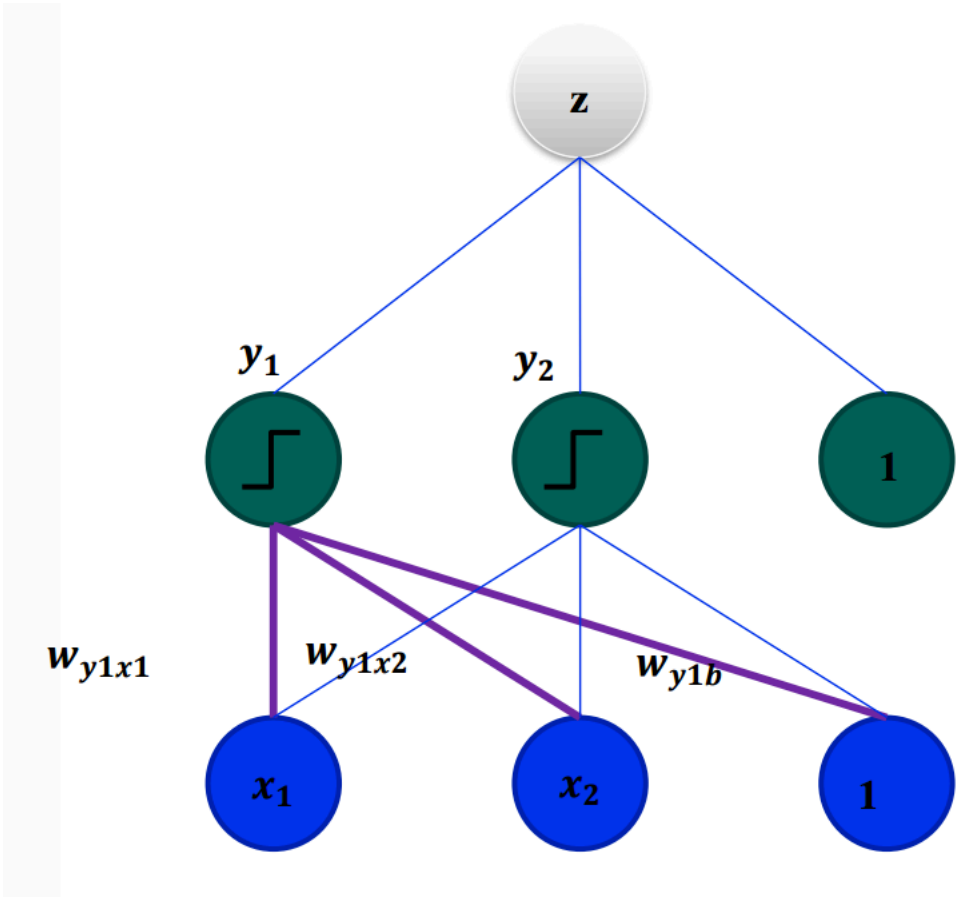
Multiple lines can separate the data correctly!

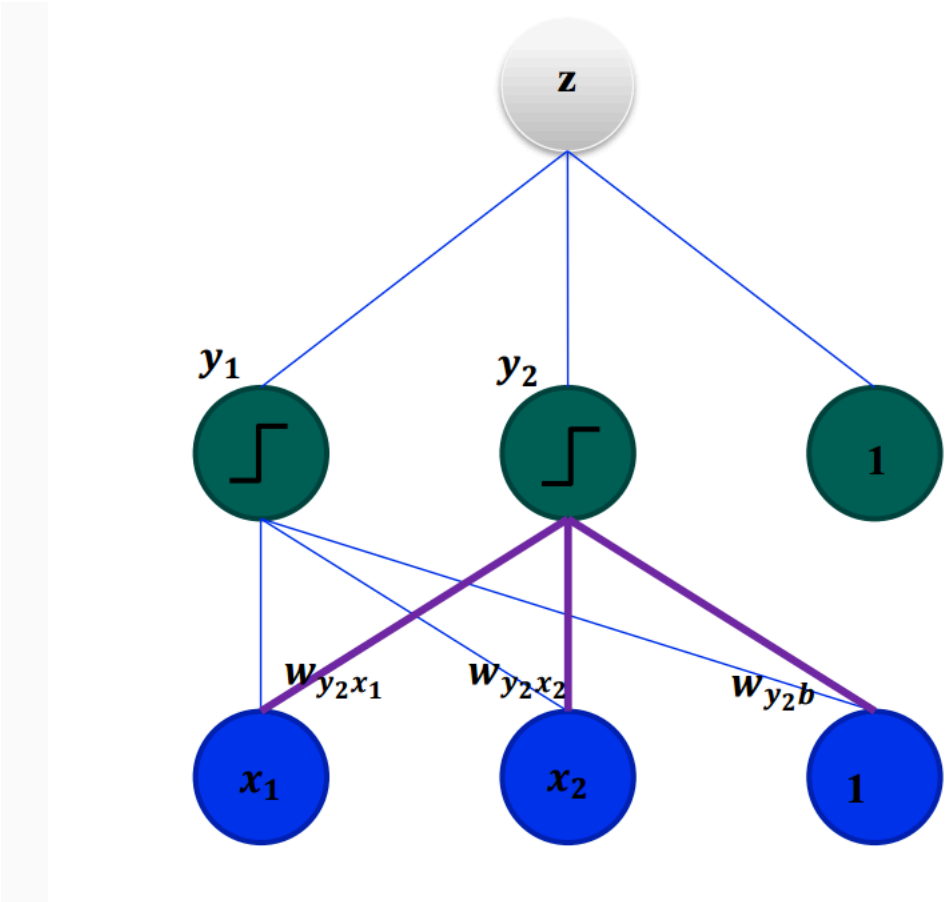
Perceptron: XOR problem



Layer를 추가하면 해결 가능

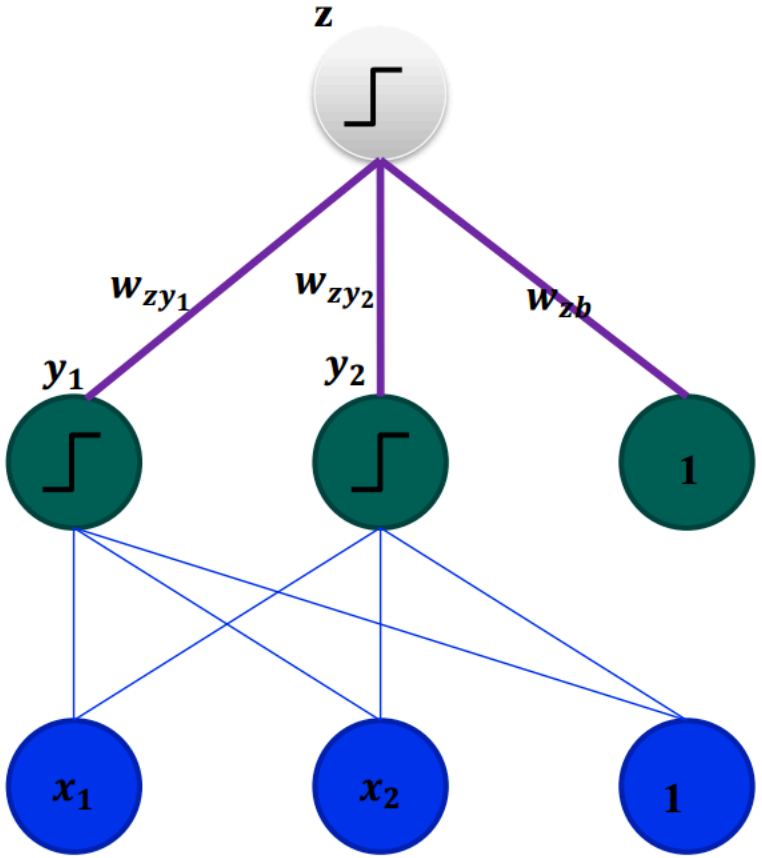
What does this all mean?





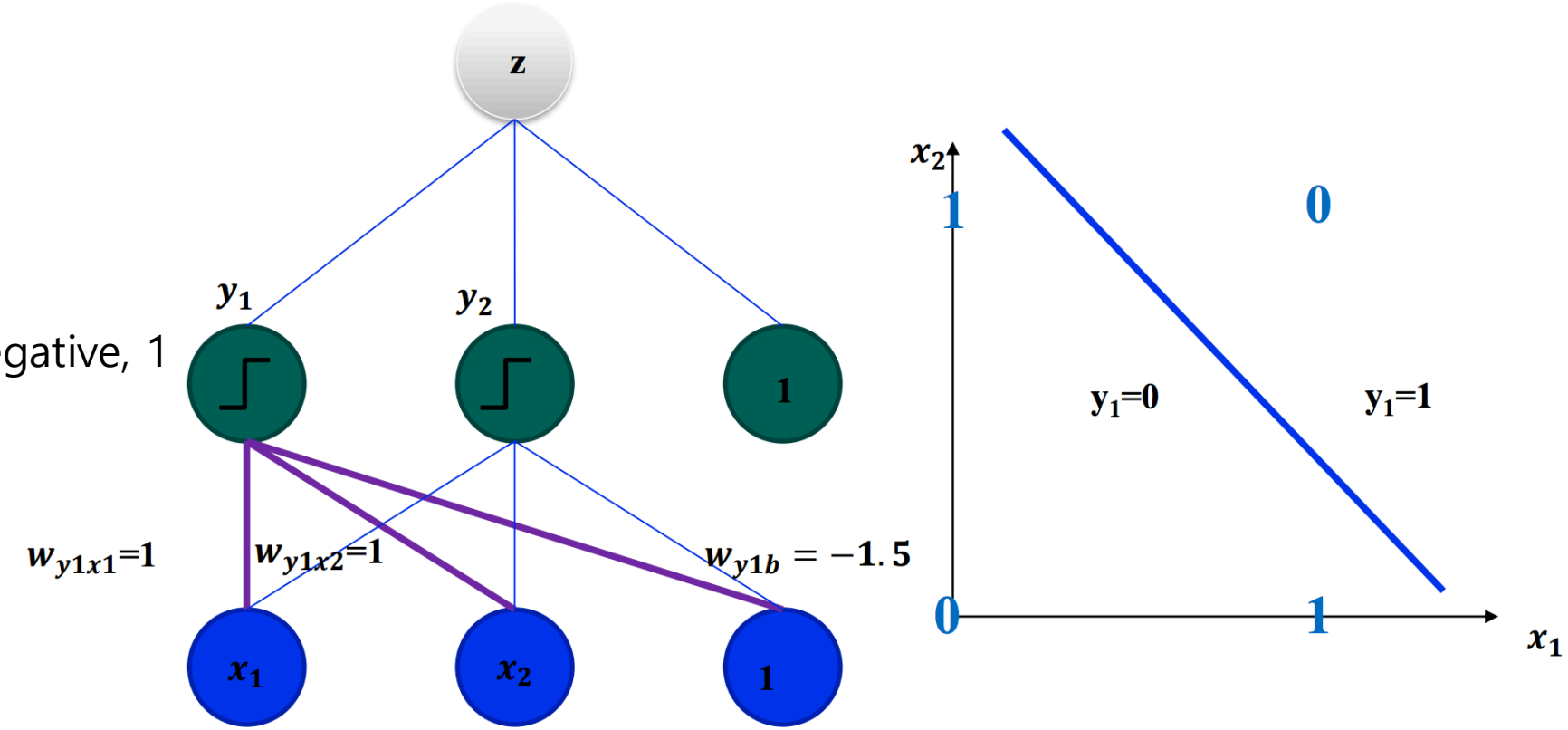
What does this all mean?

1. Use the first two lines to convert the data
2. Classify on the modified data set

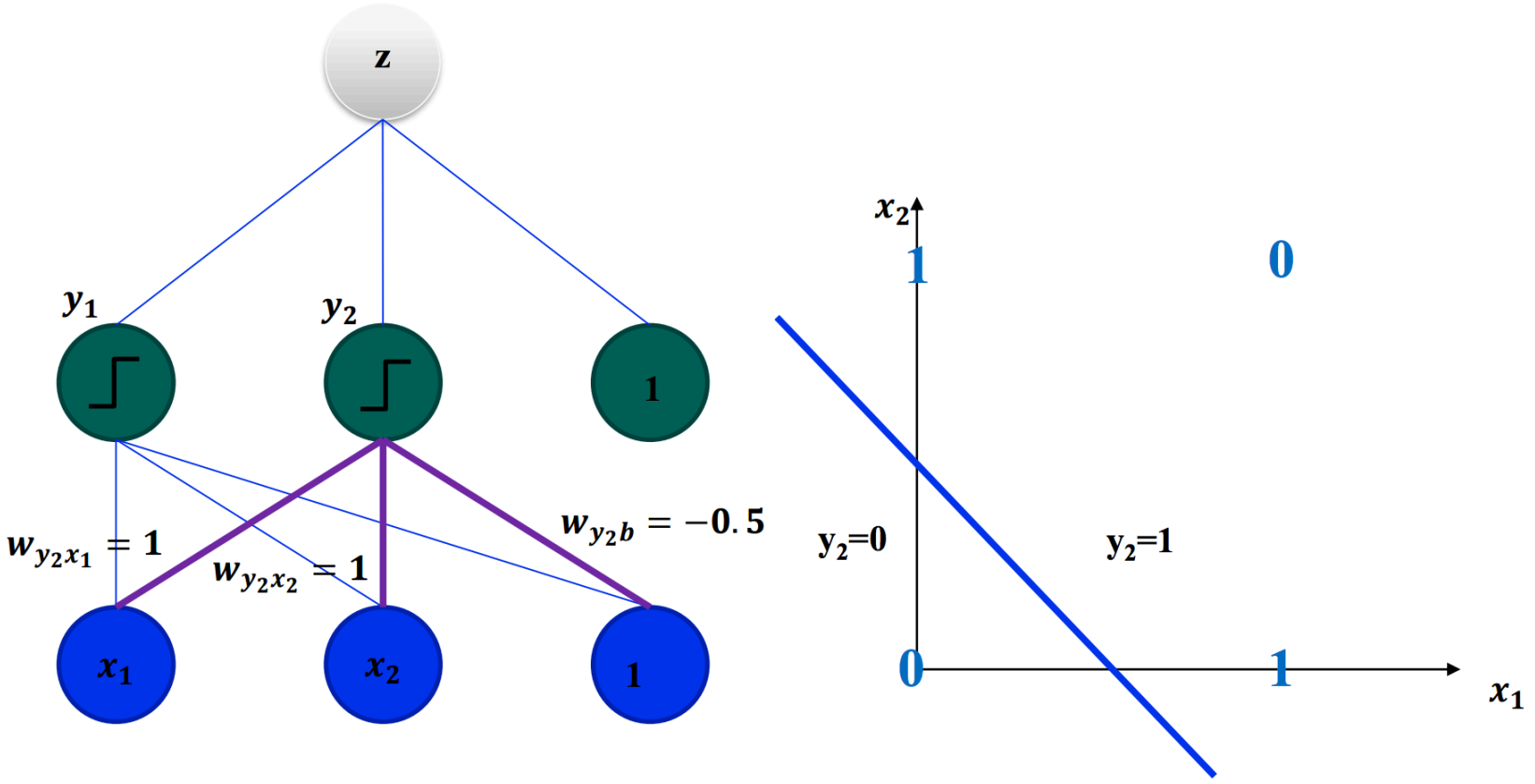


Converting the data

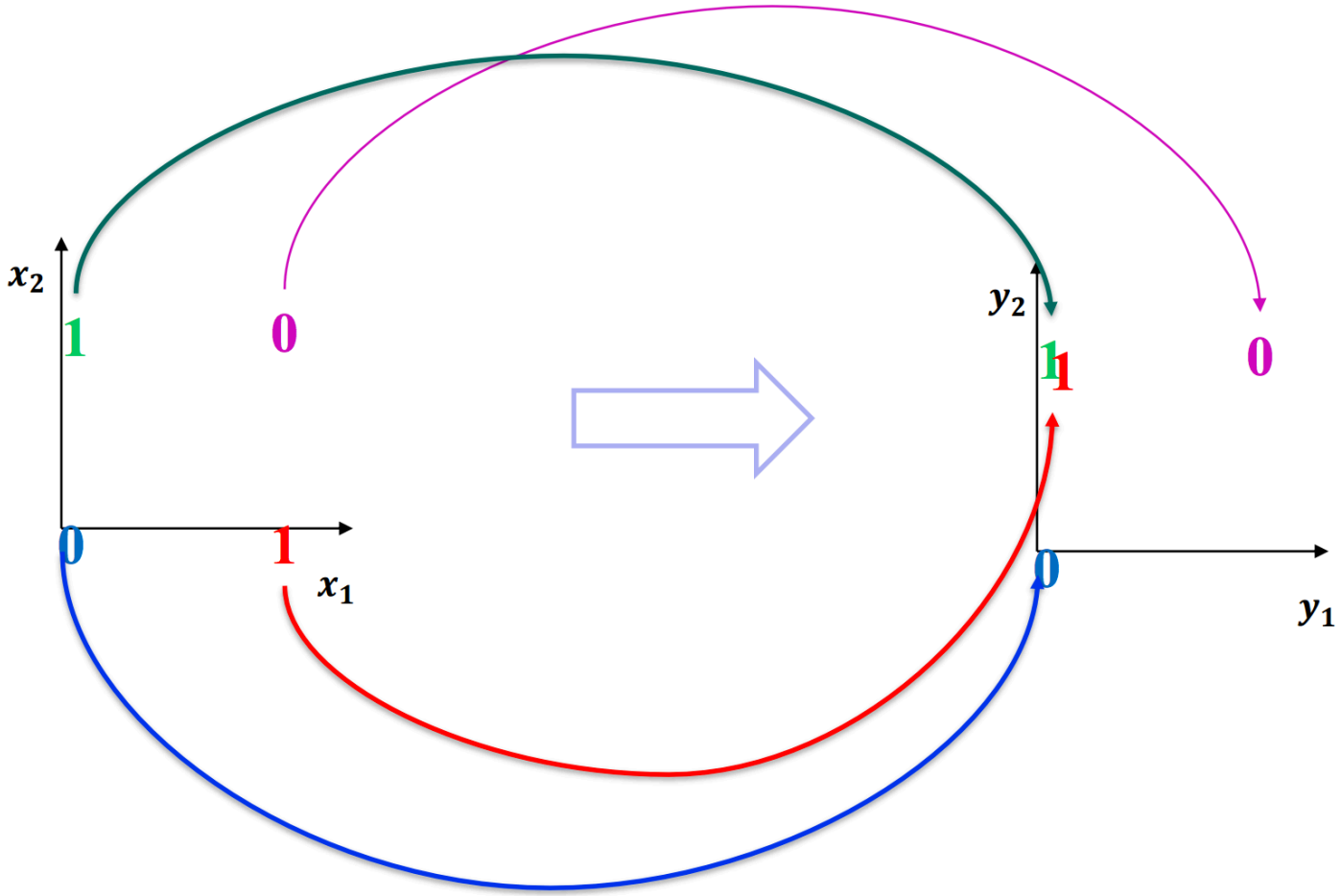
Outputs 0 if negative, 1 if positive



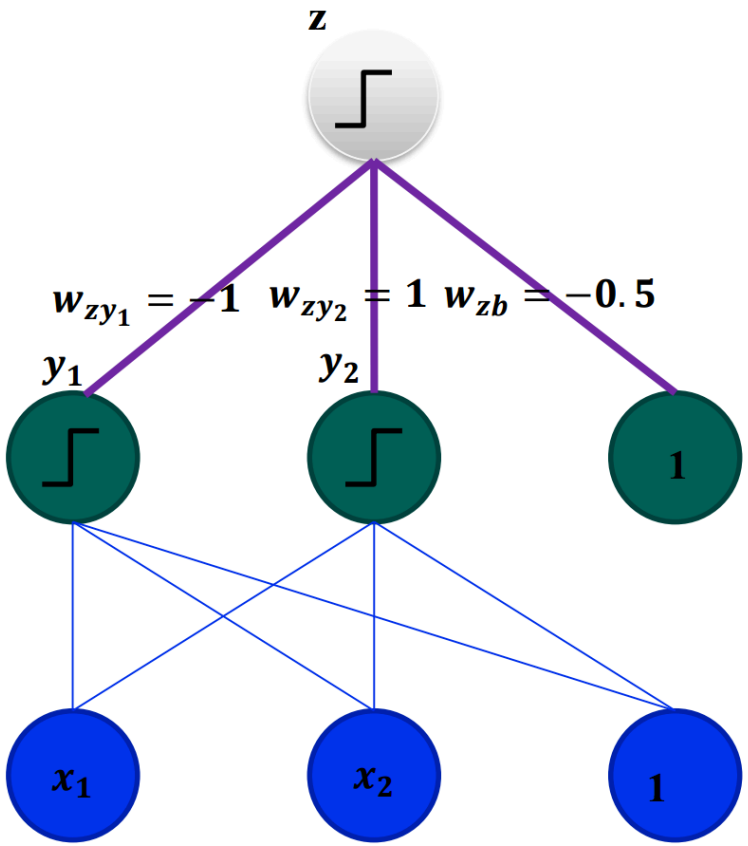
Converting the data

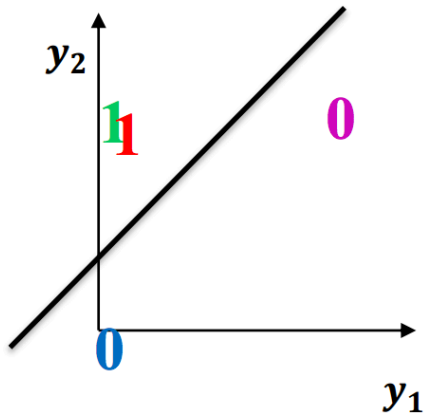


After the conversion



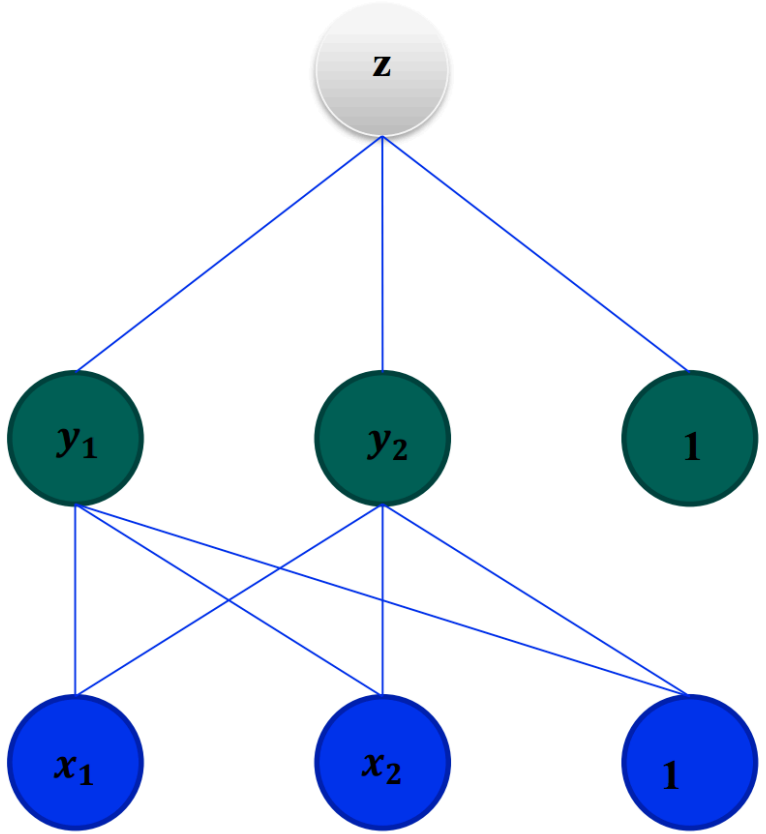
XOR classified!





Multi-layer perceptron

- Add more layers to solve more complex questions!



False promises

- “The Navy revealed the embryo of an electronic computer today that ***it expects will be able to walk, talk, see, write, reproduce itself*** an be **conscious of its existence** … Dr. Frank Rosenblatt, a research psychologist at the Cornell Aeronautical Laboratory, Buffalo, said Perceptrons might be fired to the planets as mechanical space explorers”, The New York Times, July 08, 1958

The first A.I. winter

- Massive investment in machine translation during the cold war

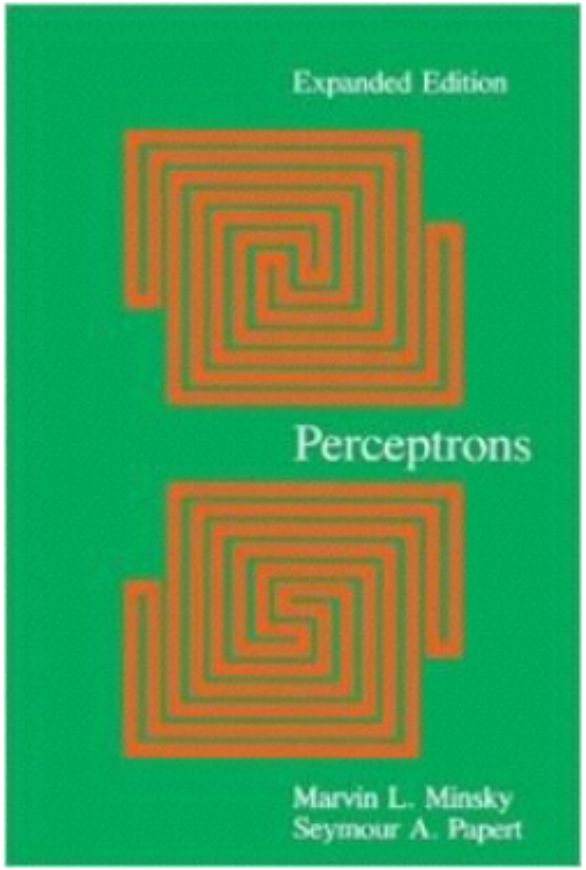
- “the spirit is willing but the flesh is weak” -> “the vodka is good but the meat is rotten”
- “out of sight, out of mind” -> “blind idiot”

- *Careers destroyed, research ended*

The first A.I. winter

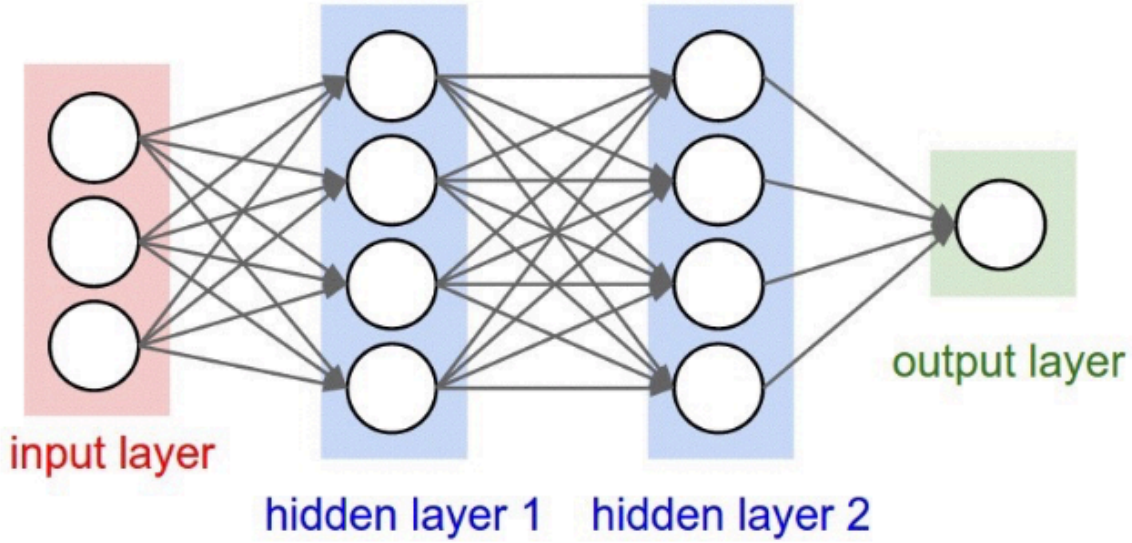
- Perceptrons (1969) by Marvin Minsky, founder of the MIT AI Lab

- We need to use MLP
- "**No one on earth had found a viable way to train MLPs** good enough to learn such simple functions"



The first A.I. winter

- "**No one on earth had found a viable way to train MLPs**"



***Marvin Minsky, 1969**

http://cs231n.github.io/convolutional-networks/

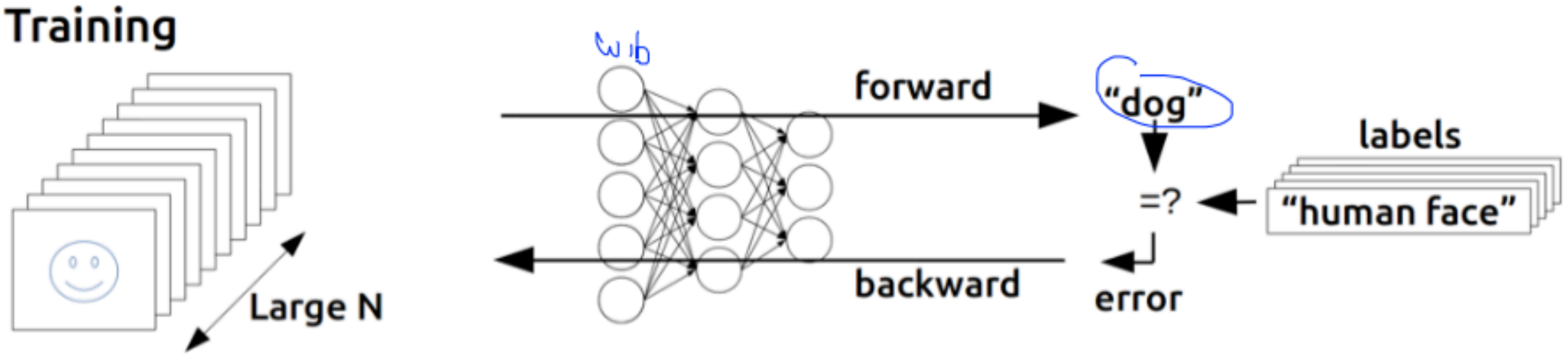
What do you mean? What about all the learning algorithms?

1. A number of perceptrons equal to the number of the function's outputs would be started off with small initial weights
2. For the inputs of an example in the training set, compute the perceptions' output
3. For each Perceptron, if the output does not match the example's output, adjust the weights accordingly
4. Go to the next example in the training set and repeat steps 2-4 until the perceptrons no longer make mistakes

- This only specifies the correct output for the final output layer, not the layers in before that!

The thaw of the A.I. winter: backpropagation

First proposed in 1974, 1982 by Paul Werbos, popularized in 1986 by Hinton



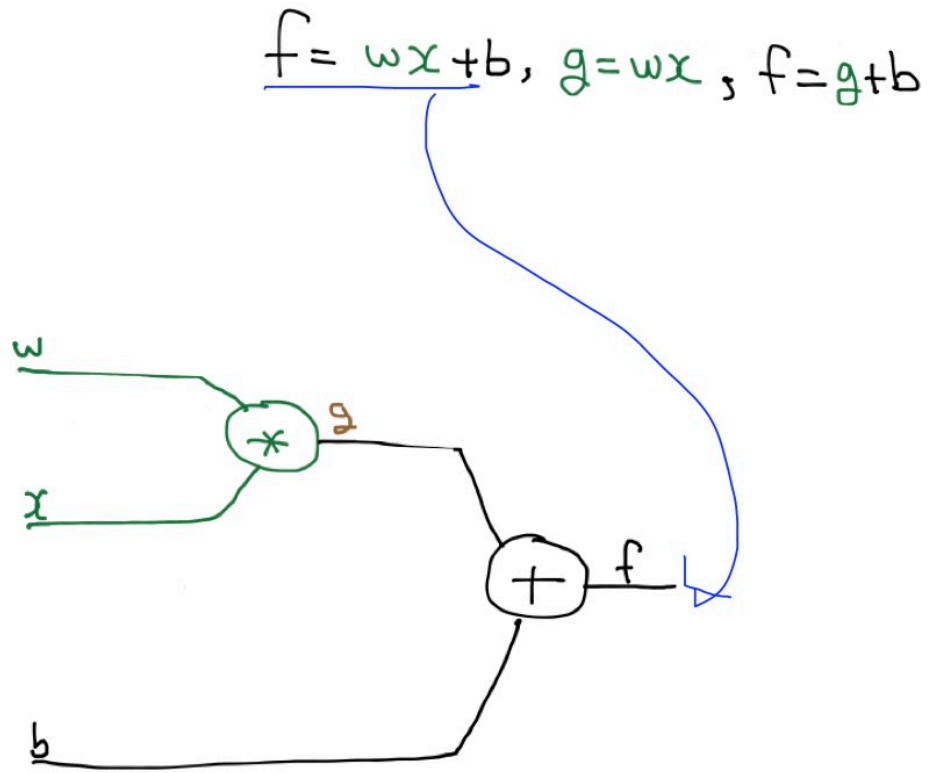
$$W := W - \alpha \frac{\partial}{\partial W} cost(W)$$

Back propagation (chain rule)

$f = wx + b$, $g = wx$, $f = g + b$

$$W := W - \alpha \frac{\partial}{\partial W} cost(W)$$

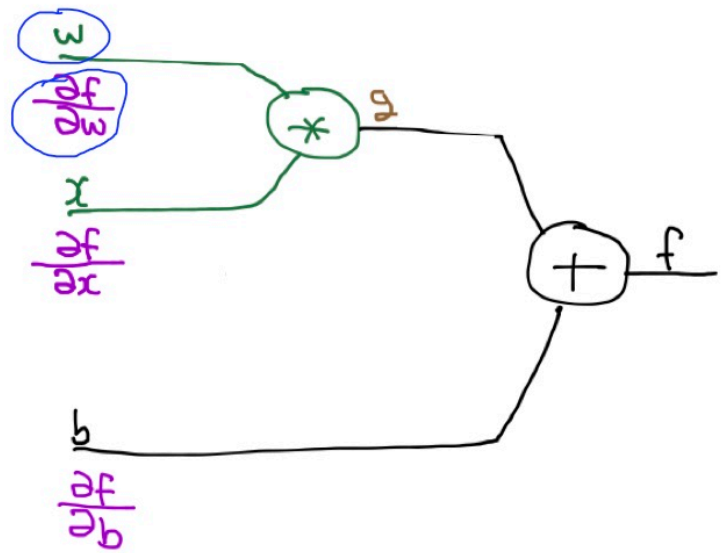
Back propagation (chain rule)



$$W := W - \alpha \frac{\partial}{\partial W} cost(W)$$

Back propagation (chain rule)

$f = wx + b$, $g = wx$, $f = g + b$



chain rule

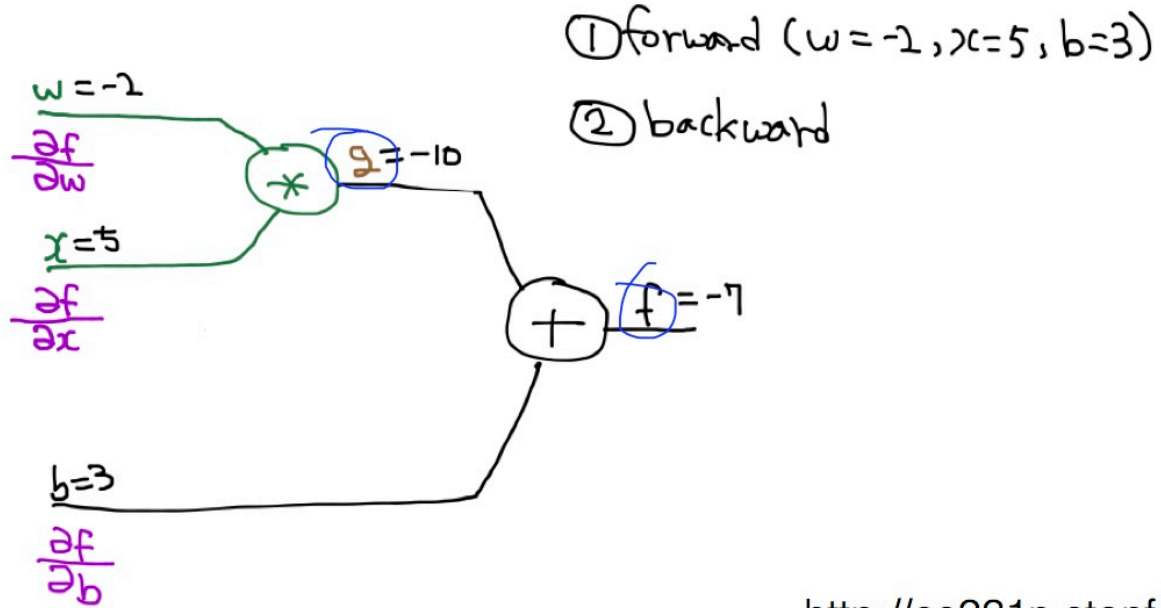
$f(g(x))$

$$\frac{\partial f}{\partial x} = \frac{\partial f}{\partial g} \cdot \frac{\partial g}{\partial x}$$

$$W := W - \alpha \frac{\partial}{\partial W} cost(W)$$

Back propagation (chain rule)

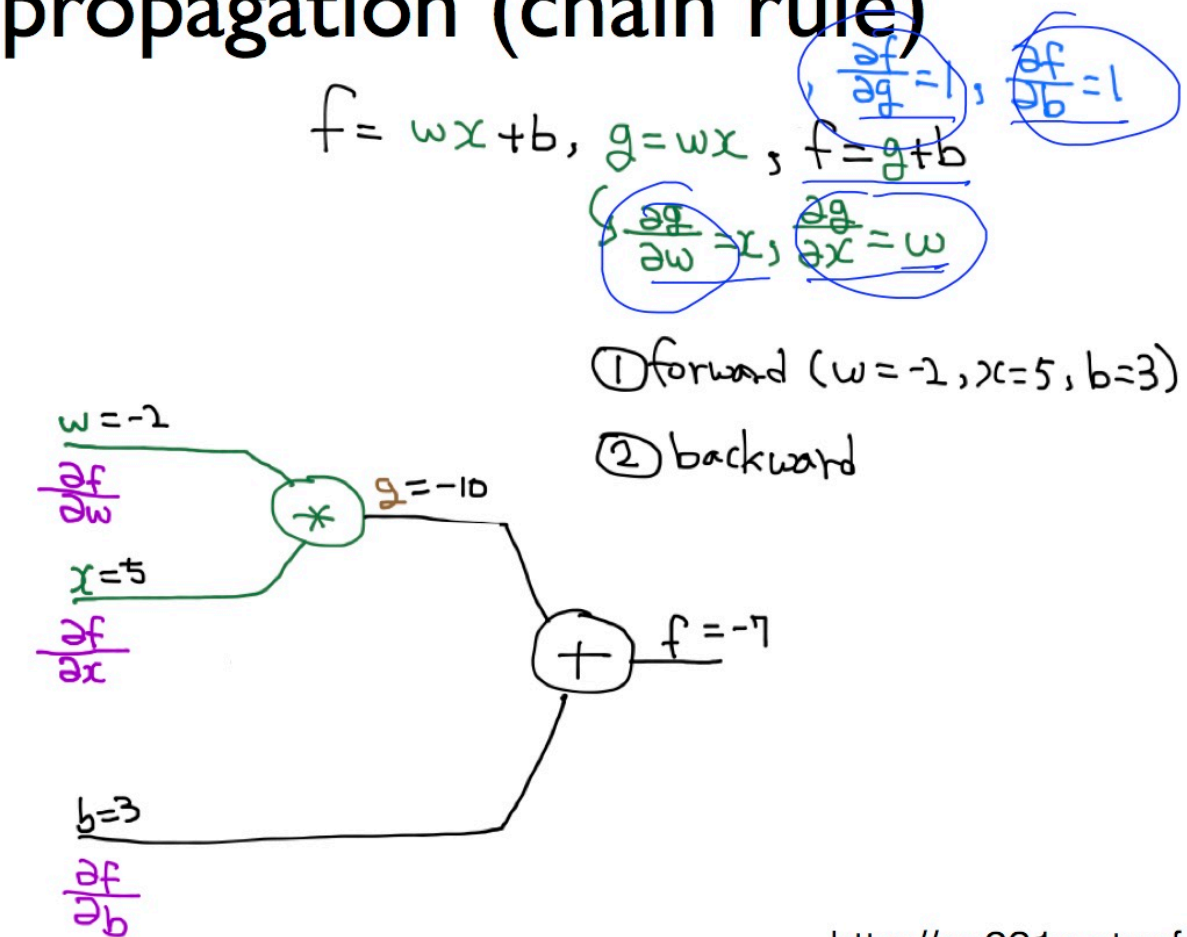
$f = wx + b$, $g = wx$, $f = g + b$



http://cs231n.stanford.edu/

$$W := W - \alpha \frac{\partial}{\partial W} cost(W)$$

Back propagation (chain rule)



http://cs231n.stanford.edu/

$$W := W - \alpha \frac{\partial}{\partial W} cost(W)$$

Back propagation (chain rule)

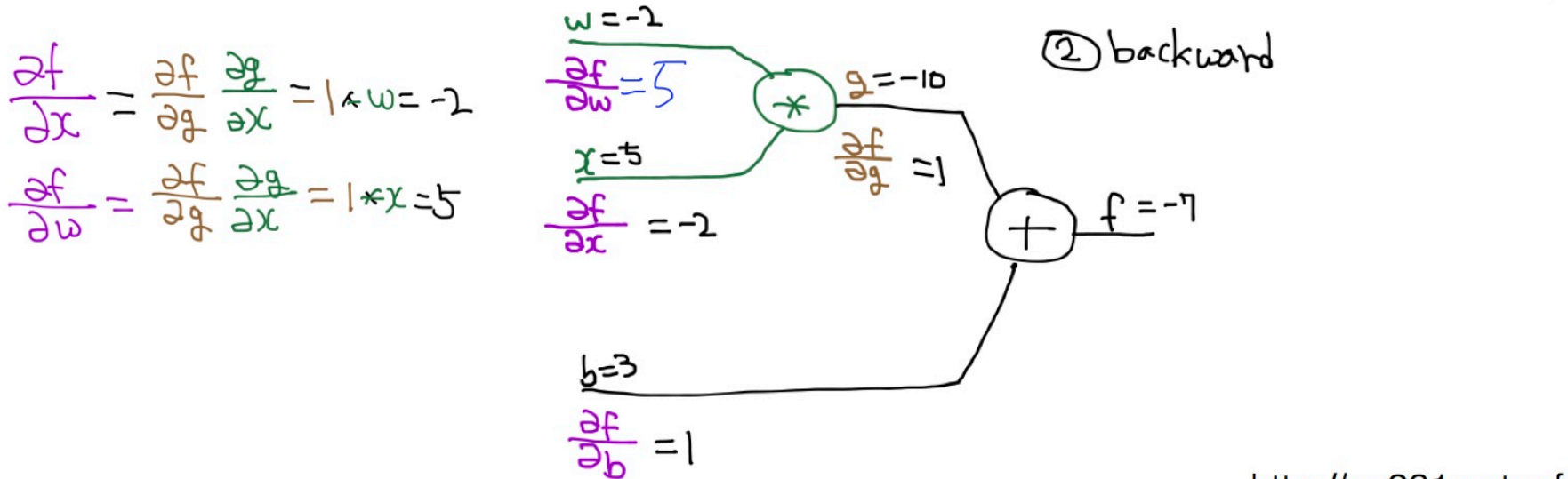
$f = wx + b$, $g = wx$, $f = g + b$

$\frac{\partial f}{\partial g} = 1$, $\frac{\partial f}{\partial b} = 1$

$\frac{\partial g}{\partial w} = x$, $\frac{\partial g}{\partial x} = w$

① forward ($w = -2, x = 5, b = 3$)

② backward

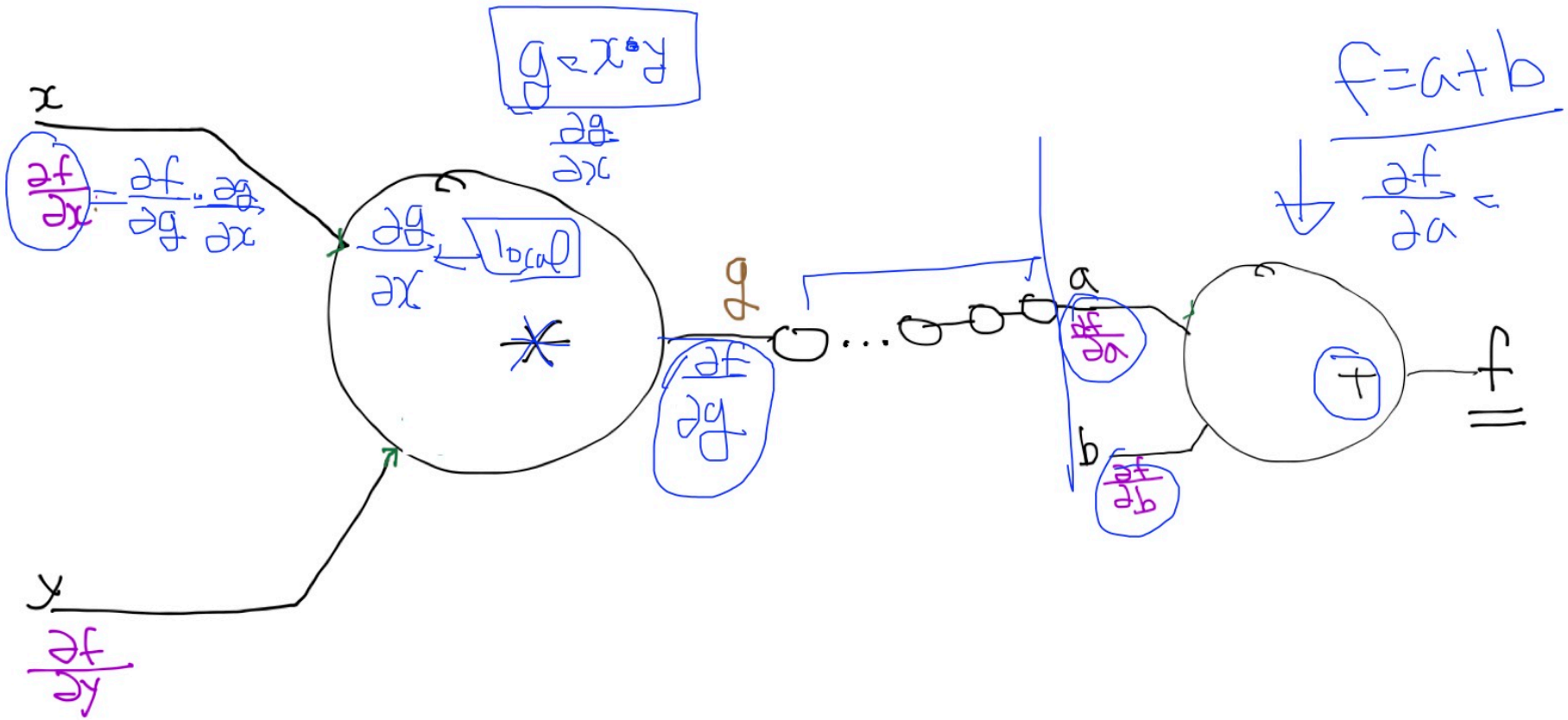


$\frac{\partial f}{\partial x} = \frac{\partial g}{\partial x} \frac{\partial f}{\partial g} = w \times 1 = -2$

$\frac{\partial f}{\partial w} = \frac{\partial g}{\partial w} \frac{\partial f}{\partial g} = x \times 1 = 5$

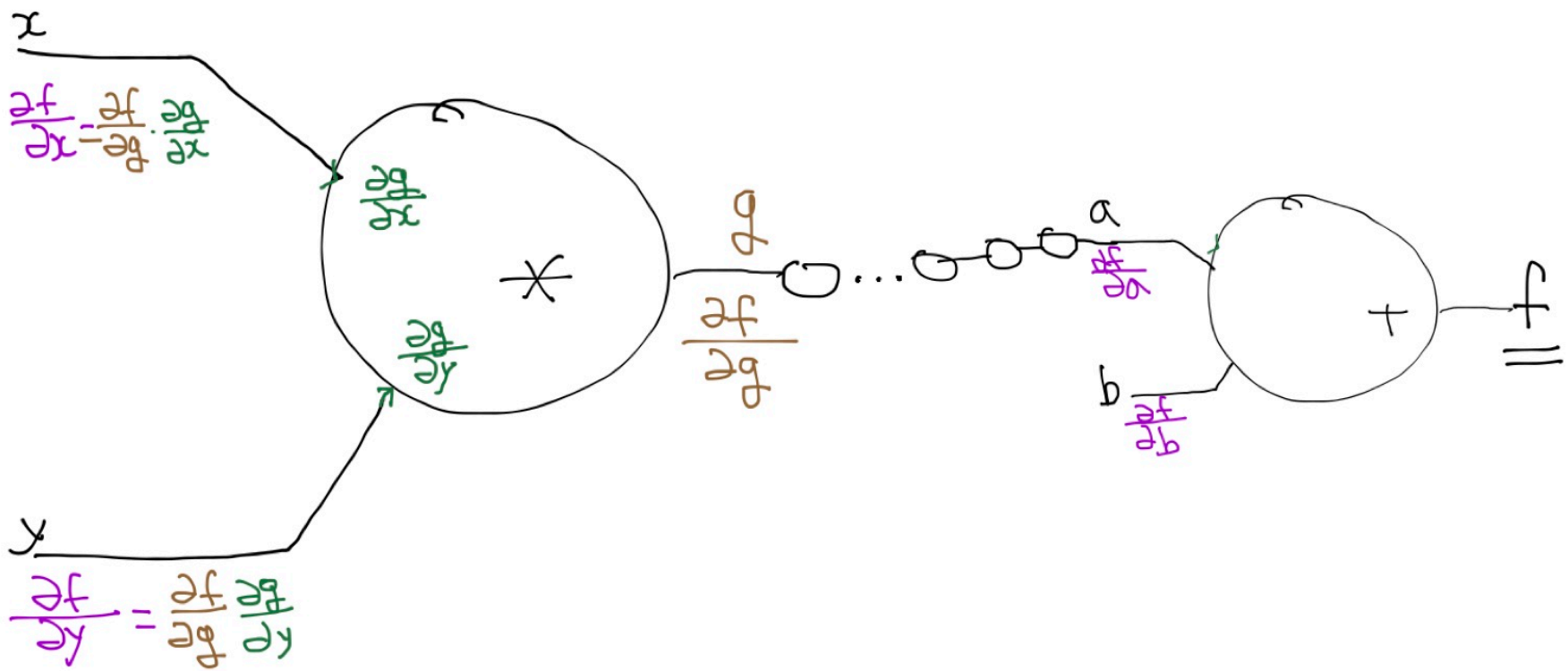
http://cs231n.stanford.edu/

Back propagation (chain rule)



http://cs231n.stanford.edu/

Back propagation (chain rule)

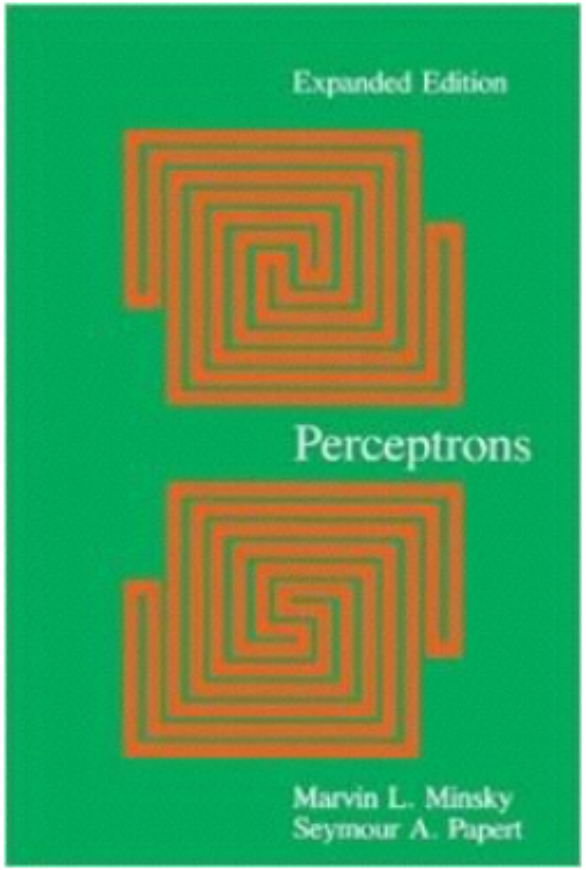


http://cs231n.stanford.edu/

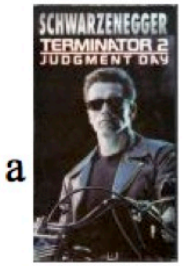
The first A.I. winter

- Perceptrons (1969) by Marvin Minsky, founder of the MIT AI Lab

- We need to use MLP
- "**No one on earth had found a viable way to train MLPs** good enough to learn such simple functions"



Terminator 2 (1991)



JOHN: Can you learn? So you can be... you know. More human. Not such a dork all the time.

TERMINATOR: My CPU is a neural-net processor... a learning computer. But **Skynet** presets the switch to "read-only" when we are sent out alone.

...

We'll learn how to **set** the neural net

TERMINATOR Basically. (starting the engine, backing out) The **Skynet** funding bill is passed. The system goes on-line August 4th, 1997. Human decisions are removed from strategic defense. **Skynet** begins to learn, at a geometric rate. It becomes **self-aware** at 2:14 a.m. eastern time, August 29. In a panic, they try to pull the plug.

SARAH: And **Skynet** fights back.

TERMINATOR: Yes. It launches its ICBMs against their targets in Russia.

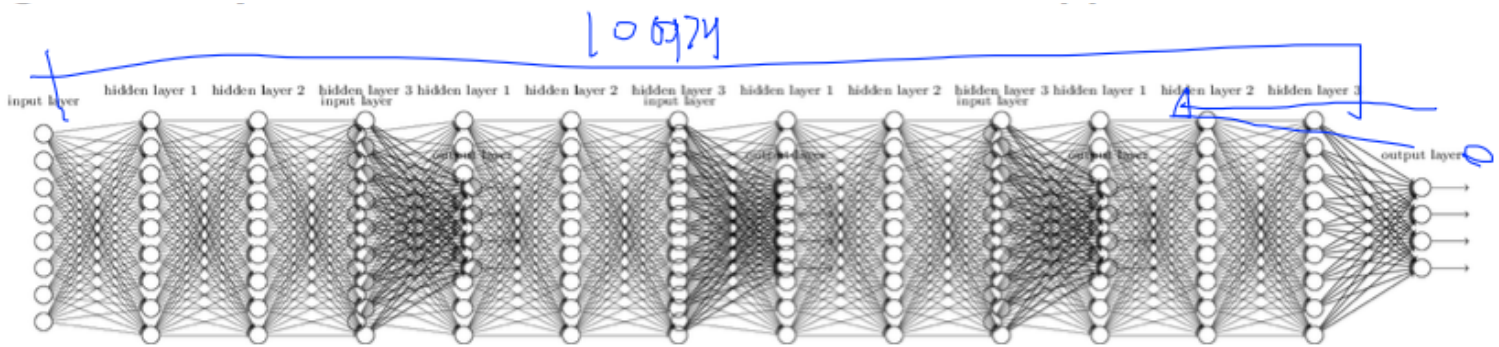
SARAH: Why attack Russia?

TERMINATOR: Because **Skynet** knows the Russian counter-strike will remove its enemies here.

http://pages.cs.wisc.edu/~jerryzhu/cs540/handouts/neural.pdf

Dawn of the second winter(1986-2006)

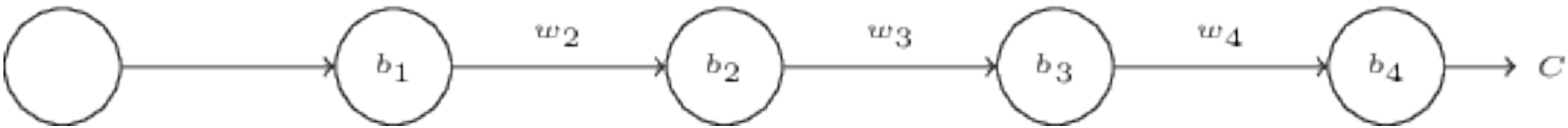
- Backpropagation just did not work well for normal neural nets with many layers
- Other rising machine learning algorithms: SVM, RandomForest, etc.
- 1995 “Comparison of Learning Algorithms For Handwritten Digit Recognition” by LeCun et al. found that this new approach worked better

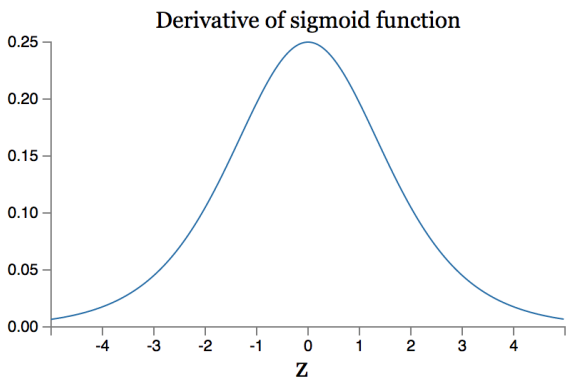


Brief sketch of vanishing gradient

- A toy network

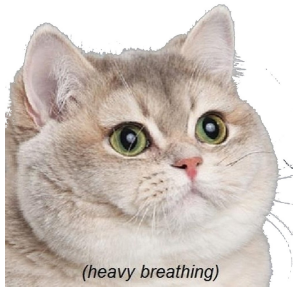
$$\frac{\partial C}{\partial b_1} = \sigma'(z_1) \times w_2 \times \sigma'(z_2) \times w_3 \times \sigma'(z_3) \times w_4 \times \sigma'(z_4) \times \frac{\partial C}{\partial a_4}$$





The thaw of the second A.I. winter: breakthroughs, Geoffrey Hinton

Heavy breathing



WOW

WOW

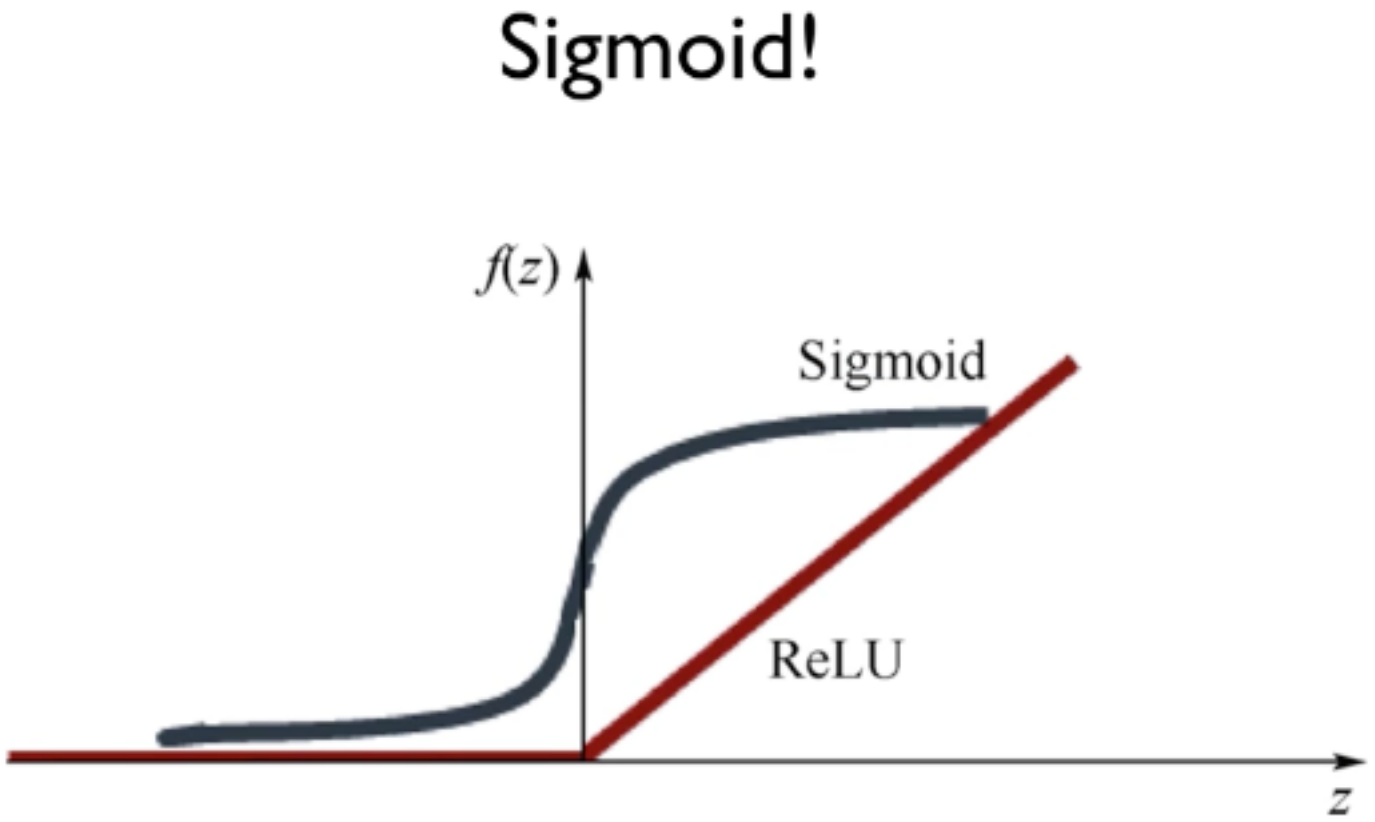
Geoffrey Hinton's summary of findings up to today (2006)

- Our labeled datasets were thousands of times too small.
- Our computers were millions of times too slow.
- We initialized the weights in a stupid way.
- **We used the wrong type of non-linearity.**

Non-linearity = sigmoid function

4. We used the wrong type of non-linearity: Rectified Linear Unit (ReLU)

But we are still using sigmoid for the output layer! Why?



Insight into the workings of ReLU

- The rectifier activation function allows a network to easily obtain sparse representations.

- For example, after uniform initialization of the weights, around 50% of hidden units continuous output values are real zeros.

- This fraction can easily increase with sparsity-inducing regularization [Xavier Glorot *et al.,* 2011]

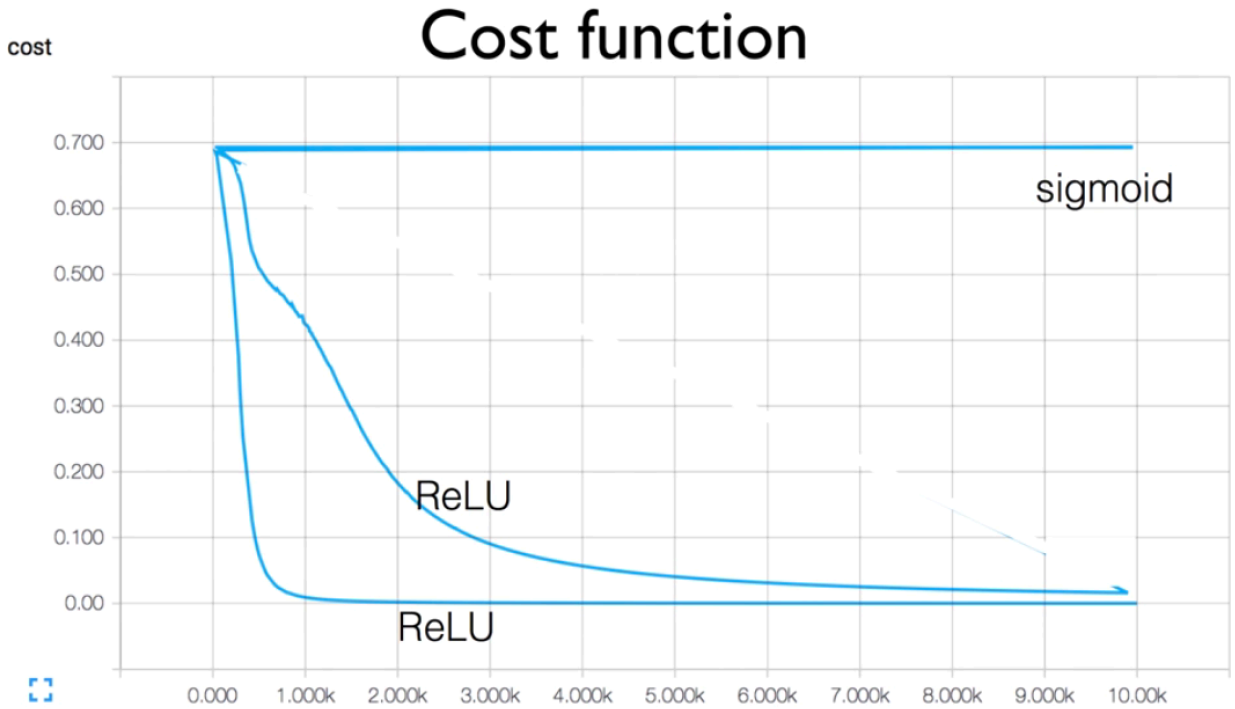
Insight into the workings of ReLU (cont.)

- This introduces sparsity effect on the network.
- Implications in computing:
  - There is a lot of multiplications by zero; lessons the amount of computations
- Implications in results:
  - Information disentangling – think of it as a feature selection
  - Linear separability

3. We initialized the weights in a stupid way

- Neural networks with many layers really could be trained well, ***if the weights are initialized in a clever way*** rather than randomly.

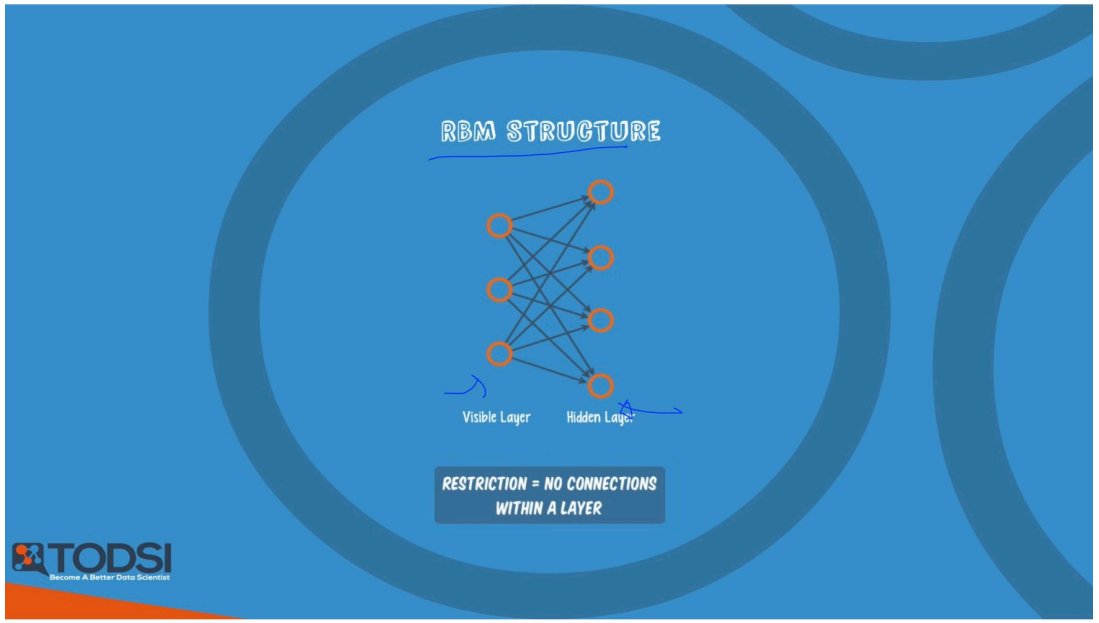
- How?

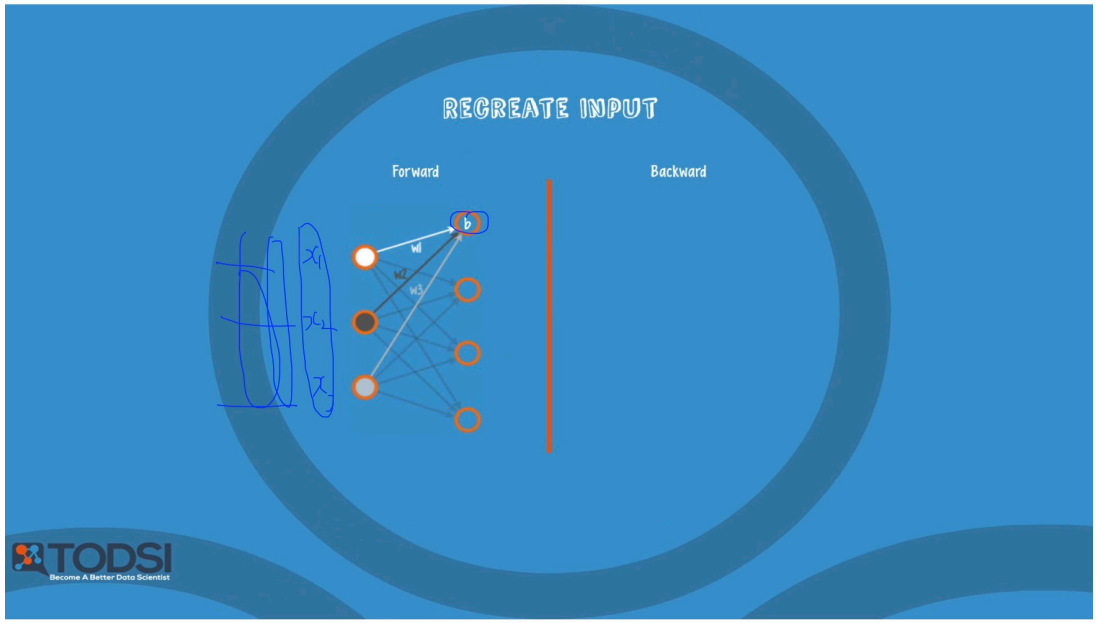


Need to set the initial weight values wisely

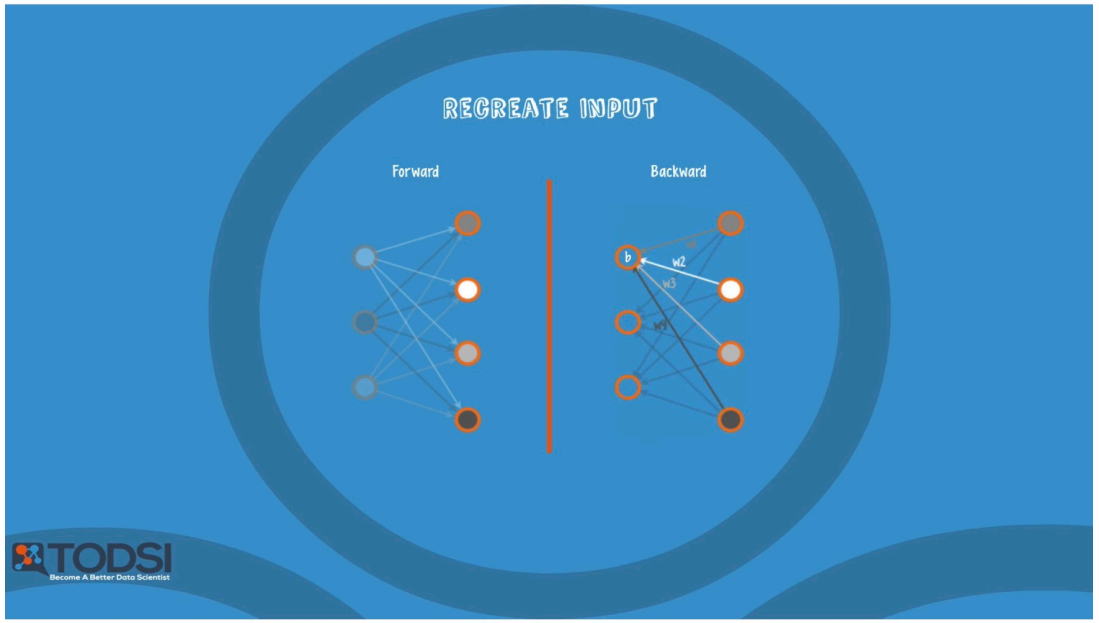
- Not all 0's
- Challenging issue
- Hinton *et al.*, 2006 "A Fast Learning Algorithm for Deep Belief Nets": **Restricted Boltzmann Machine (RBM)**

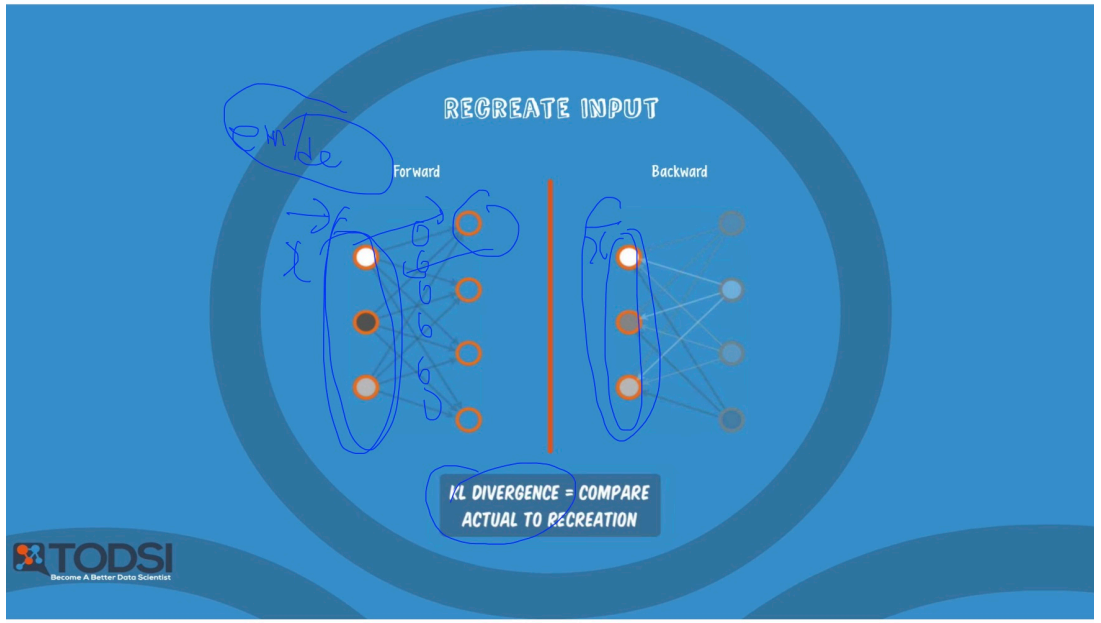
Sketch of Restricted Boltzmann Machine





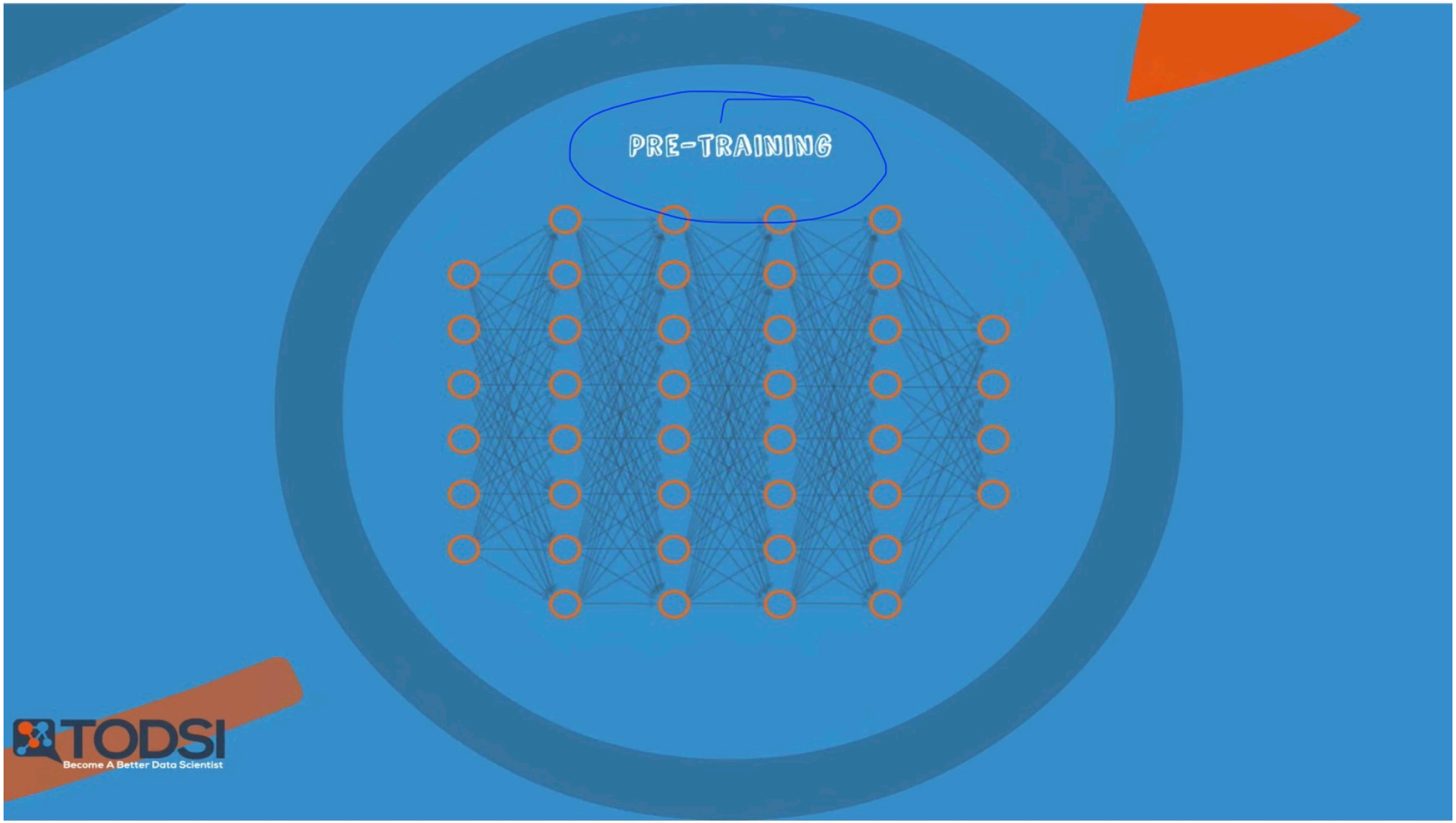
Sketch of Restricted Boltzmann Machine

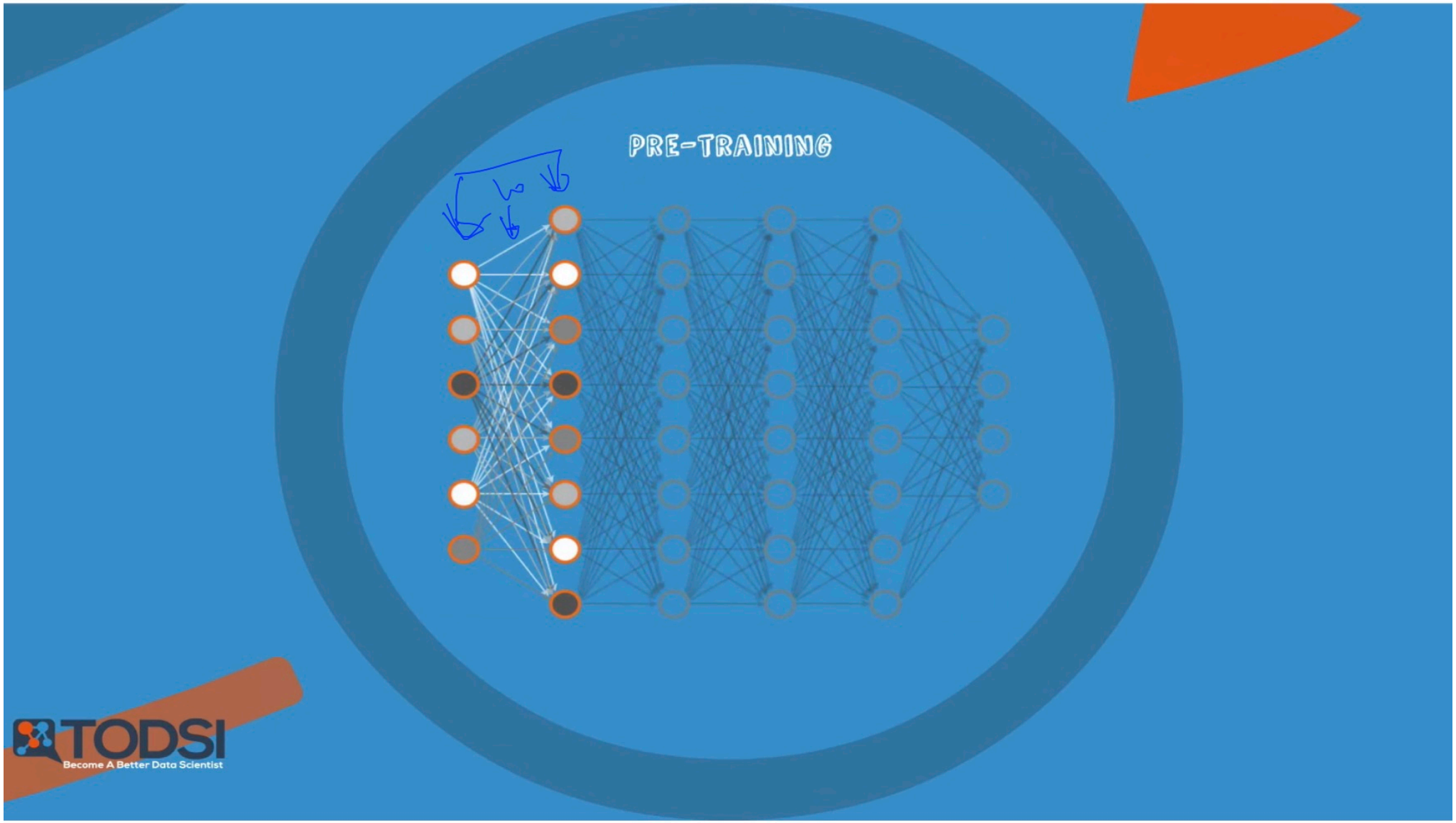


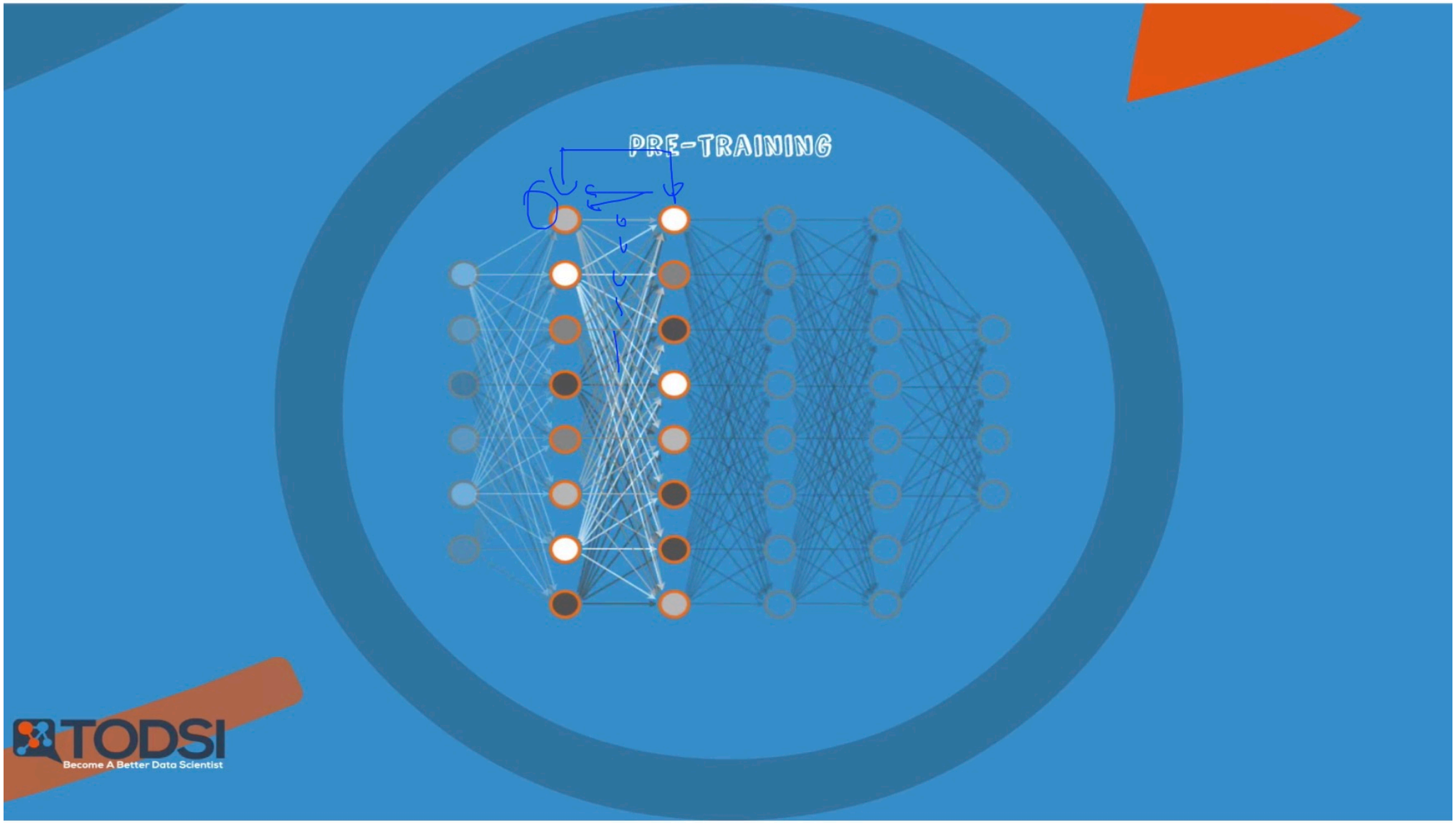


How can we use RBM to initialize weights?

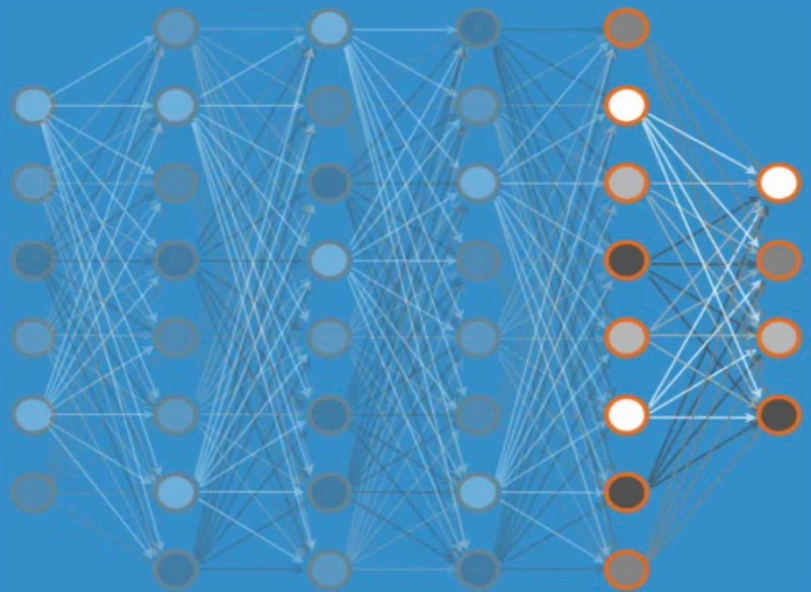
- Apply the RBM idea on adjacent two layers as a pre-training step
- Continue the first process to all layers
- This will set weights
- Example: Deep Belief Network
  - Weight initialized by RBM


PRE-TRAINING
TODSI
Become A Better Data Scientist

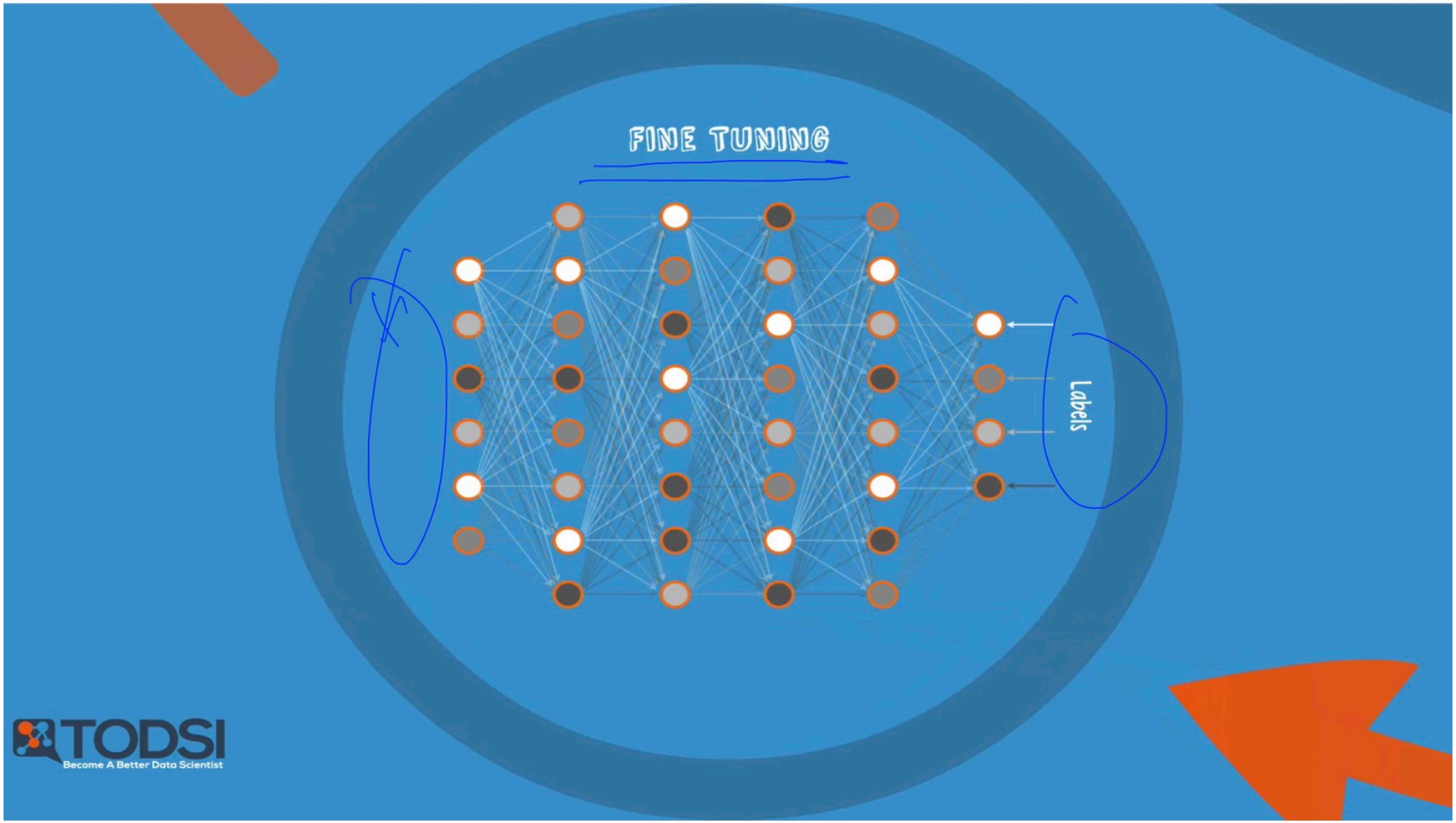

PRE-TRAINING
TODSI
Become A Better Data Scientist


PRE-TRAINING
TODSI
Become A Better Data Scientist

PRE-TRAINING



TODSI
Become A Better Data Scientist


FINE TUNING
Labels
TODSI
Become A Better Data Scientist

Good news

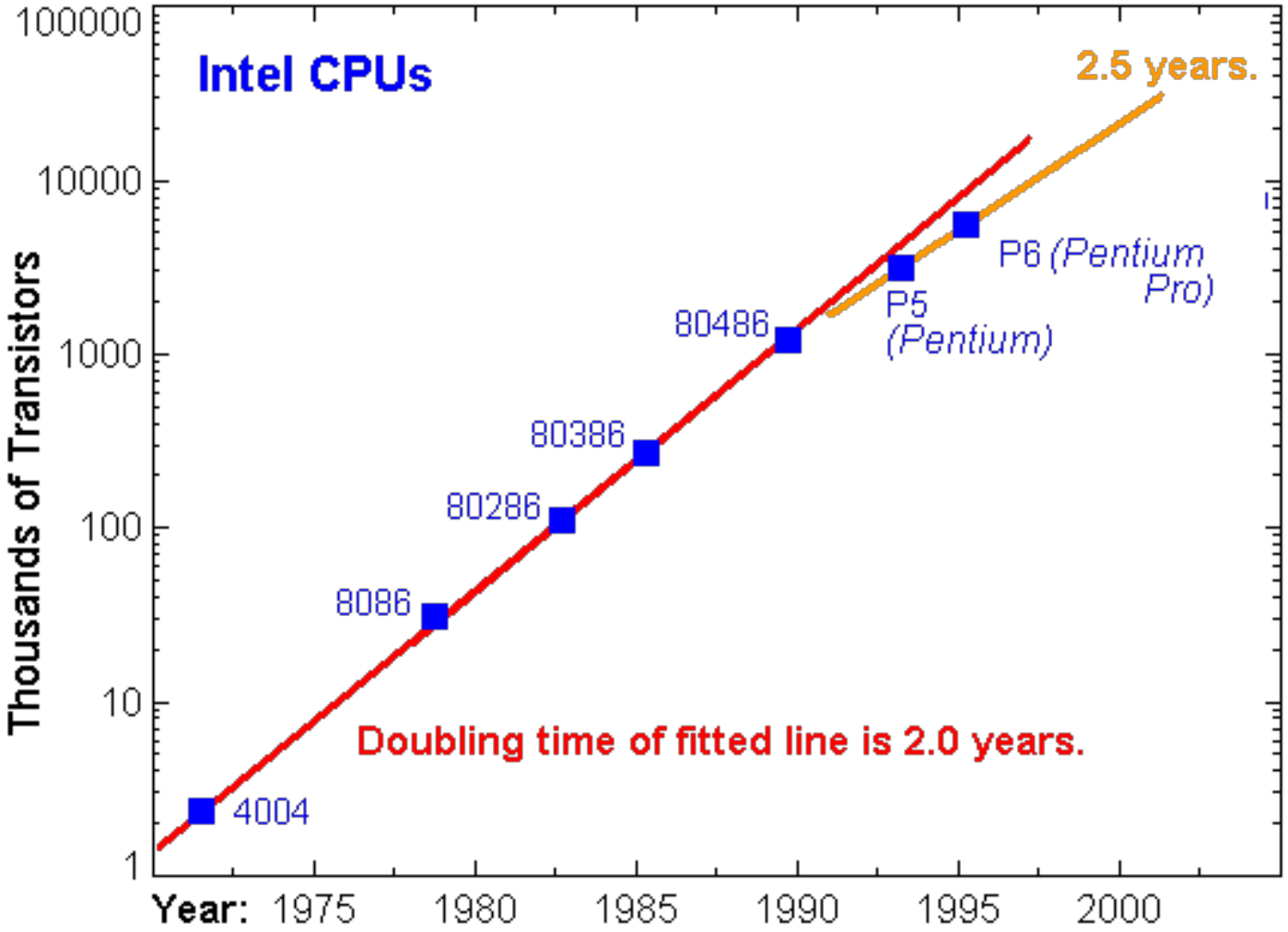
- No need to use complicated RBM for weight initializations
- Simple methods are OK
  - Xavier initialization: X. Glorot and Y.Bengio, “Understanding the difficulty of training deep feedforward neural networks,” in international conference on artificial intelligence and statistics, 2010
  - He’s initialization: K. He, X. Zhang, S. Ren, and J. Sun, “Delving Deep into Rectifiers: Surpassing Human-Level Performanec on ImageNet Classification,” 2015
- Make sure the weights are “just right,” not too small, not too big

Still an active area of research

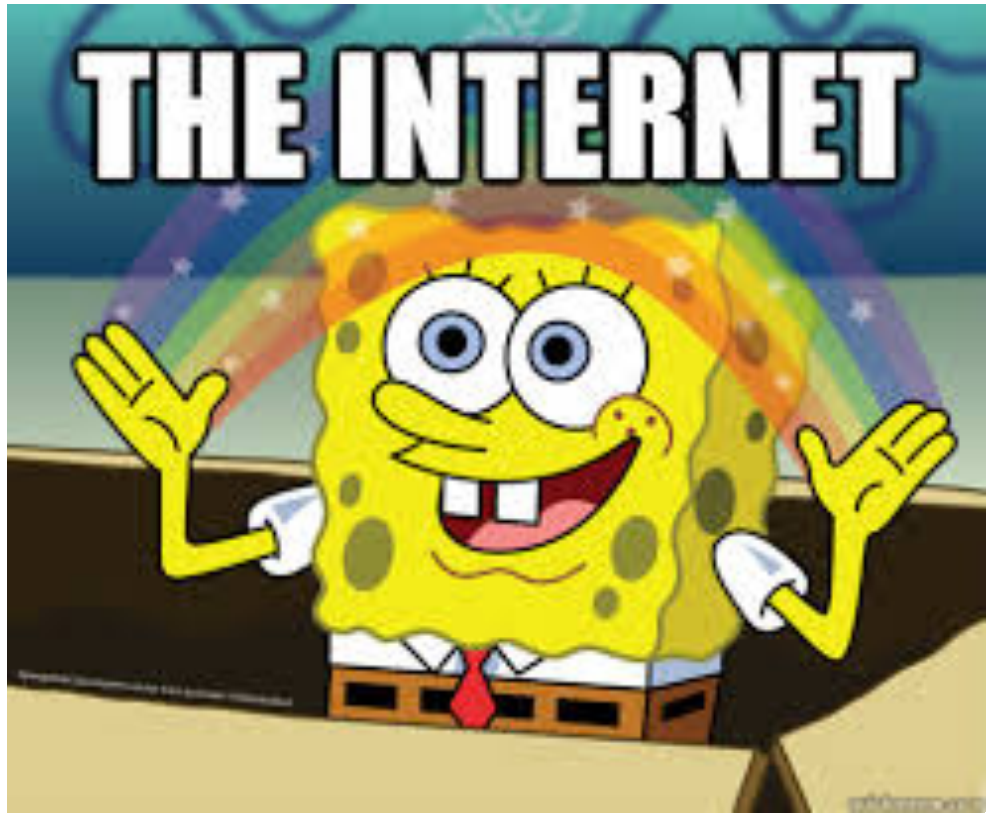
- We don’t know how to initialize perfect weight values, yet
- Many new algorithms
  - Batch normalization
  - Layer sequential uniform variance
  - Etc.

2. Our computers were million of times too slow

Moore's law refers to an observation made by Intel co-founder
Gordon **Moore** in 1965.
He noticed that the number of transistors per square inch on integrated circuits had doubled every year since their invention.



1. Our labeled datasets were thousands of times too slow

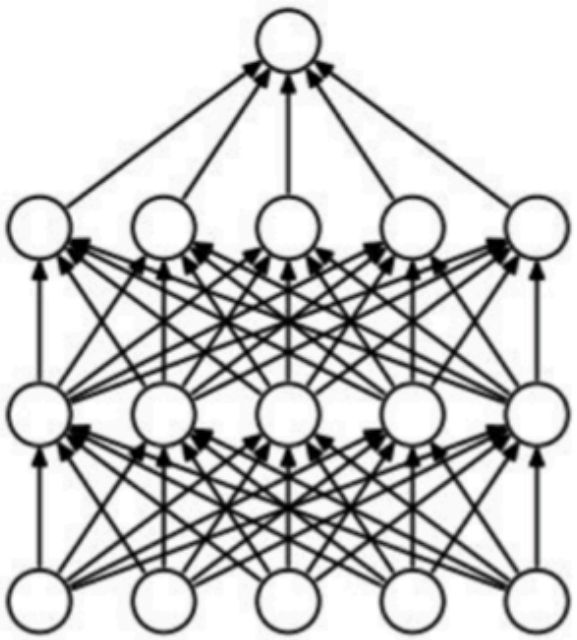


Winter is (not) coming

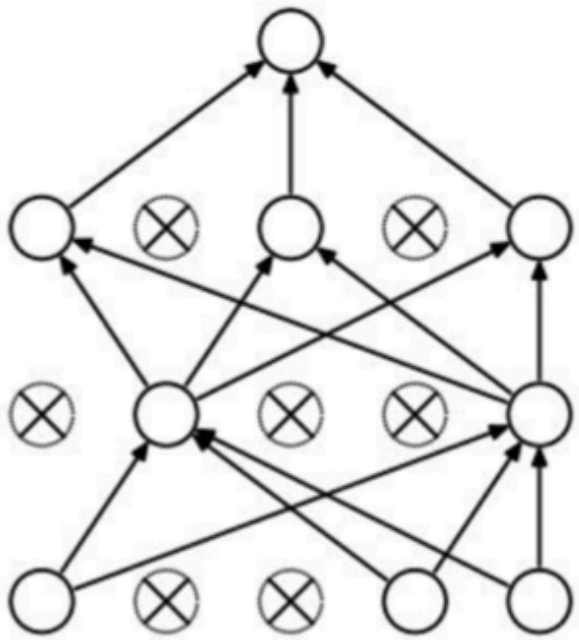
- Breakthroughs in 2006, 2007 by Hinton and Bengio
- Rebranding to ***Deep Nets, Deep Learning***

Recent breakthroughs: dropout

"randomly set some neurons to zero in the forward pass"



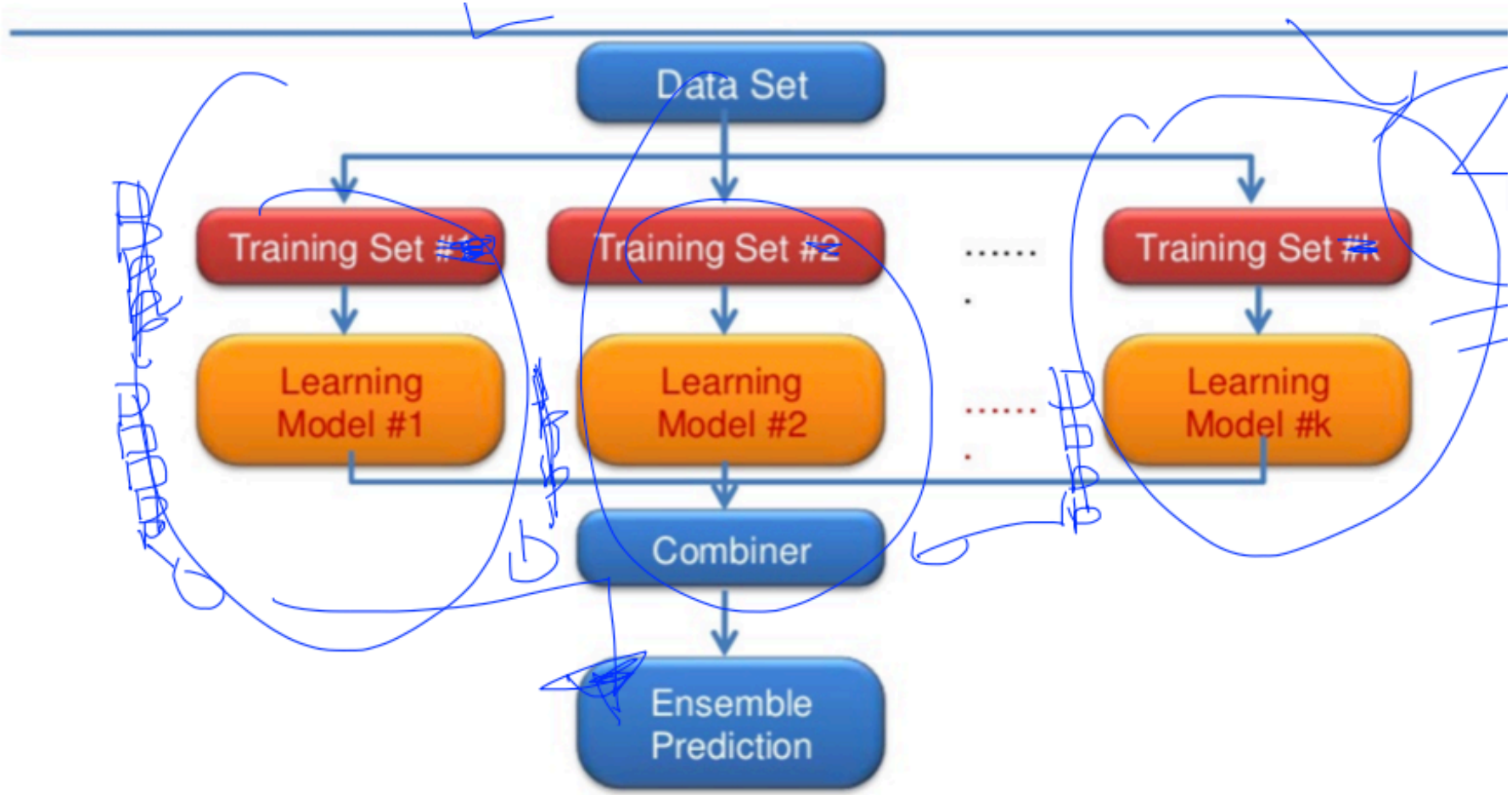
(a) Standard Neural Net



(b) After applying dropout.

[Srivastava et al., 2014]

Recent breakthroughs: ensemble learning



IMAGENET Large Scale Visual Recognition Challenge

Steel drum

The Image Classification Challenge:

1,000 object classes

1,431,167 images

Output:
Scale
T-shirt
Steel drum
Drumstick
Mud turtle

✔

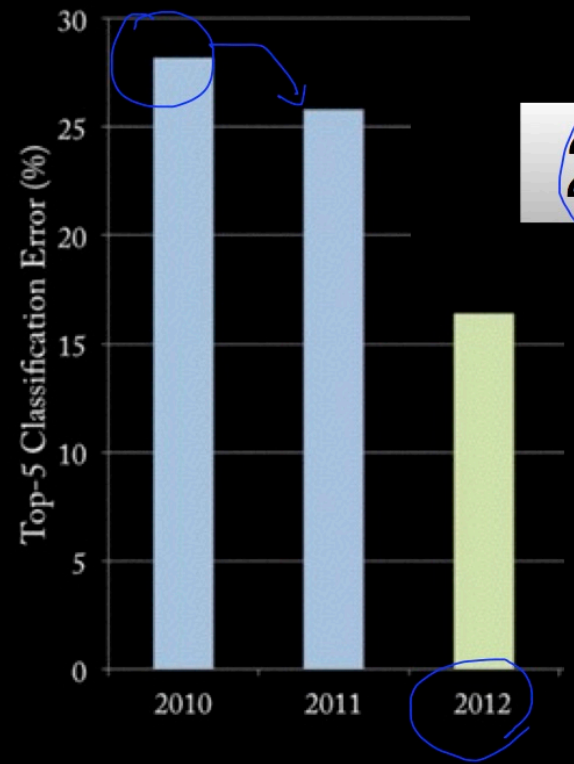
Output:
Scale
T-shirt
Giant panda
Drumstick
Mud turtle

✗

Russakovsky et al. arXiv, 2014

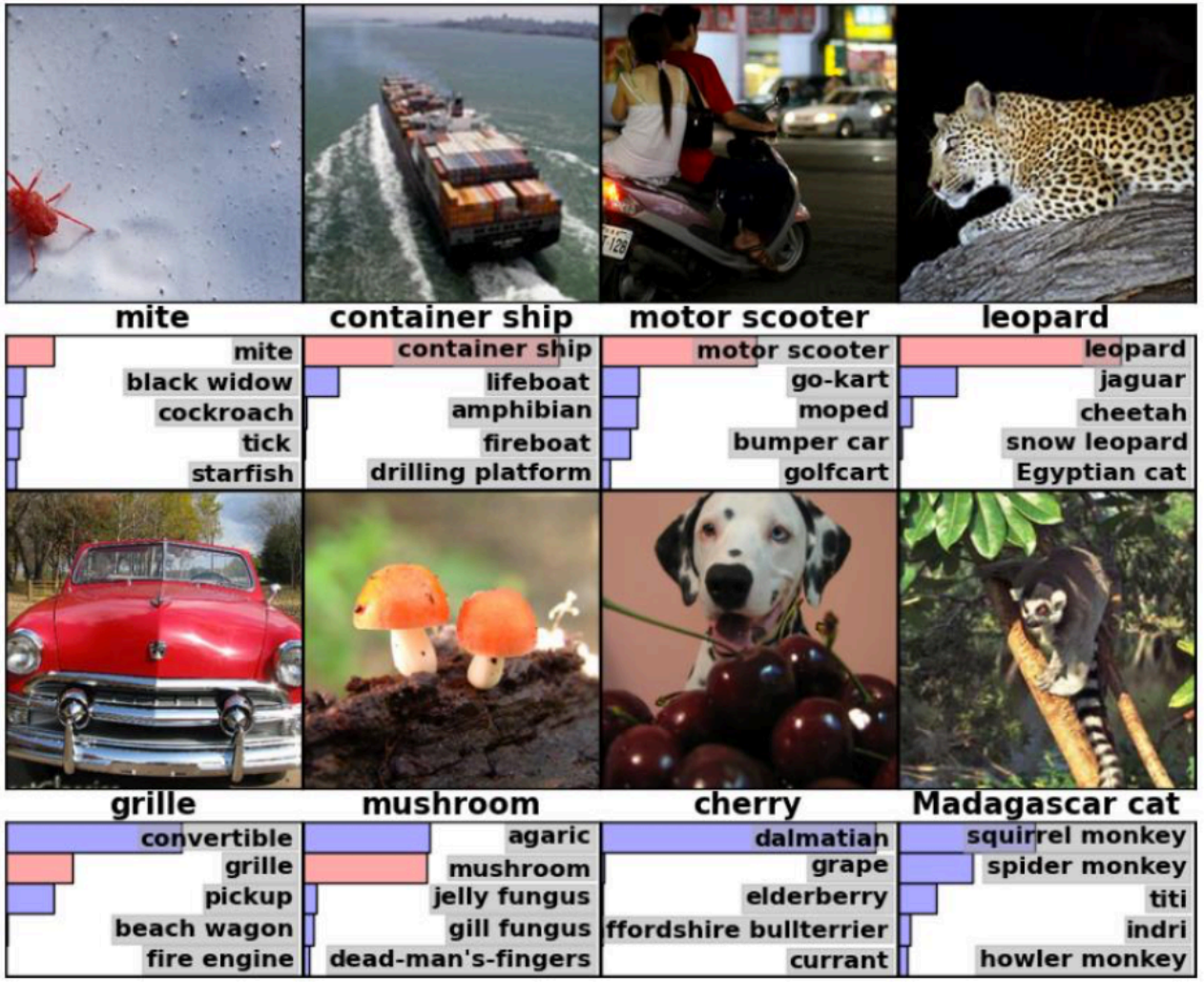
GPU TECHNOLOGY CONFERENCE

ImageNet Classification (2010 –



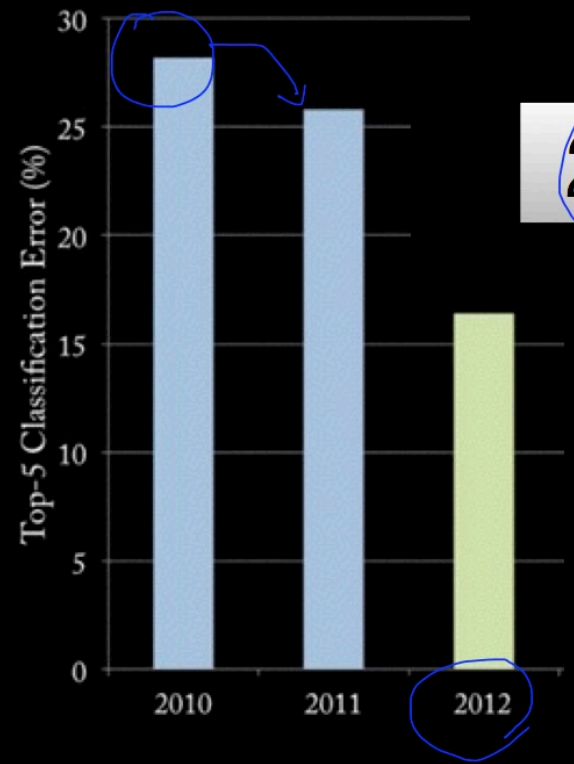
26.2% to 15.3%

Results of AlexNet



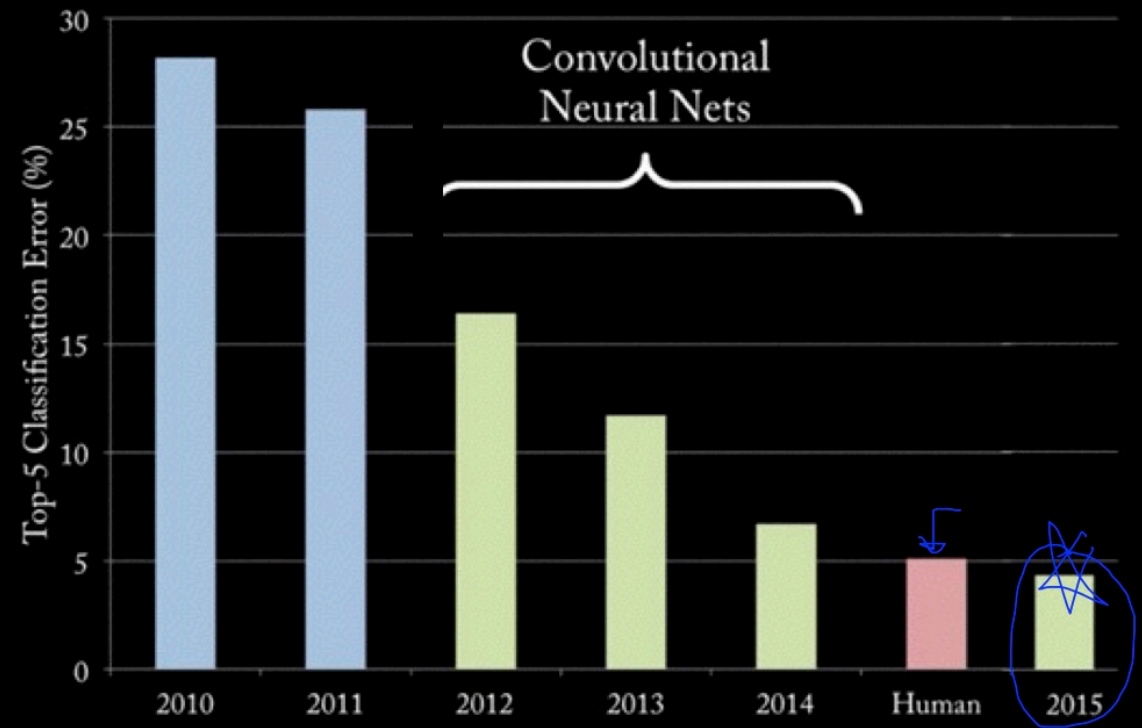
GPU TECHNOLOGY CONFERENCE

ImageNet Classification (2010 –

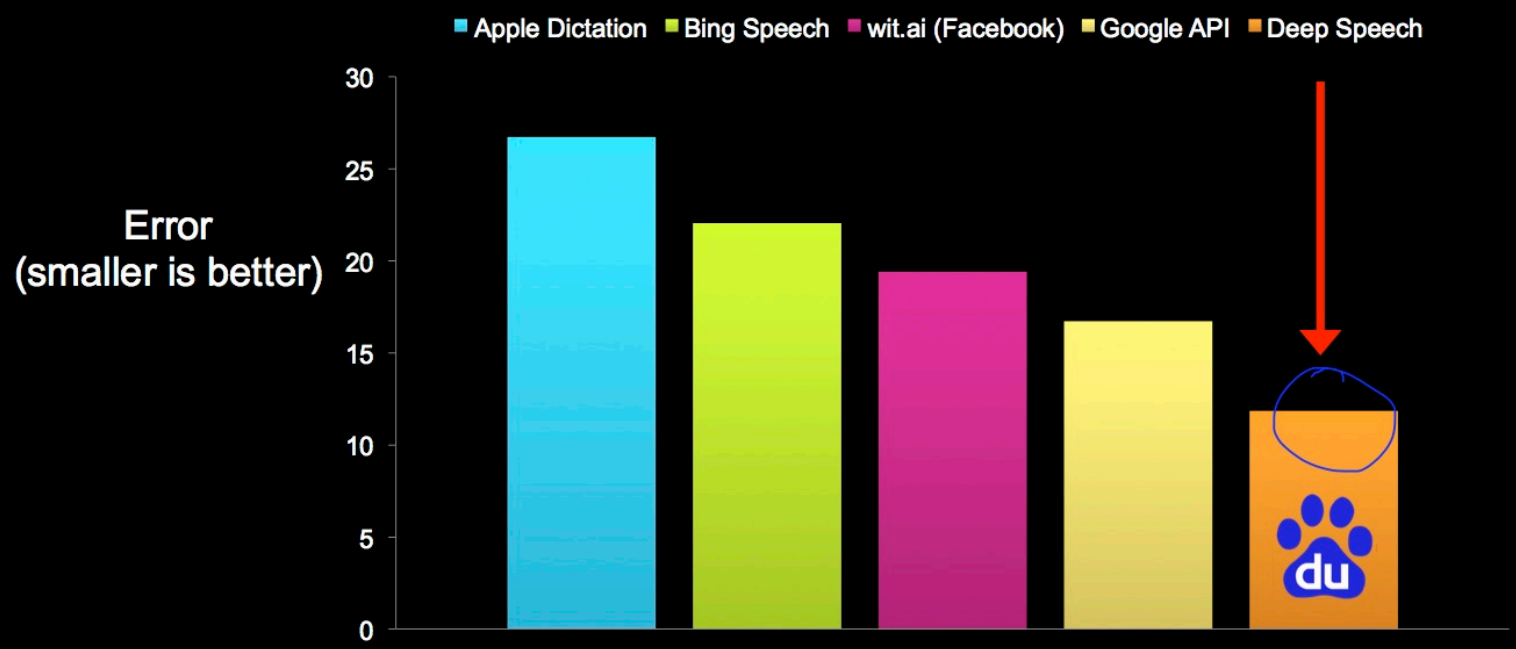


26.2% to 15.3%

ImageNet Classification (2010 – 2015)

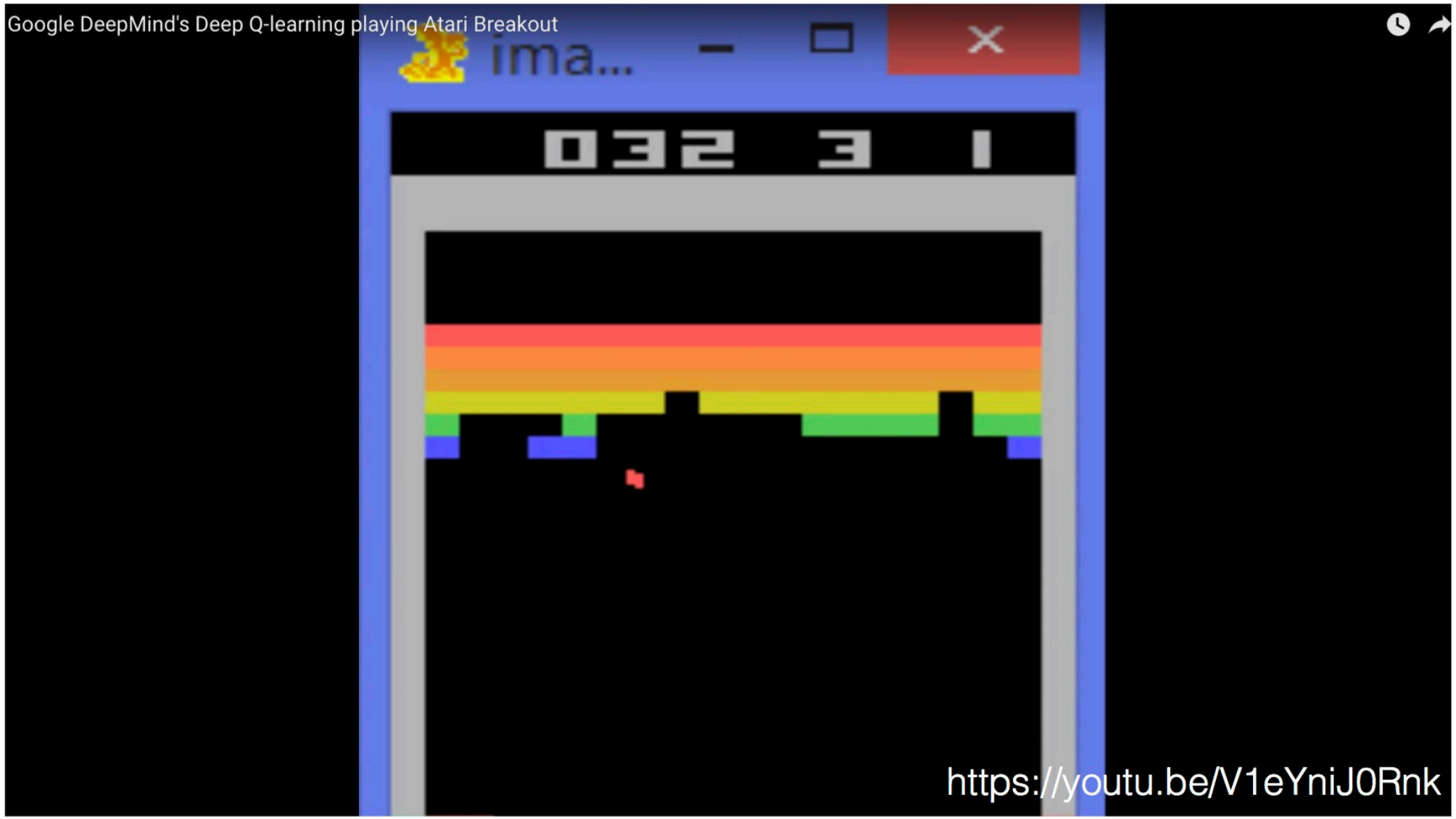


Speech recognition errors



Baidu Research

Andrew Ng


Google DeepMind's Deep Q-learning playing Atari Breakout
ima...
032 3
https://youtu.be/V1eYniJ0Rnk


ALPHAGO
01:58:30
LEE SEDOL
AlphaGo
Lee Sedol

Are we there yet?

- Alpha Go
  - Board size, 19 * 19
  - 3 states, black, white, empty
  - 3^(19*19) = 3^361
  - Computer beats us in a world with less than 3^361 states!

Are we there yet?

- Alpha Go
  - Board size, 19 * 19
  - 3 states, black, white, empty
  - 3^(19*19) = 3^361
  - Computer beats us in a world with less than 3^361 states!

- Speech Recognition

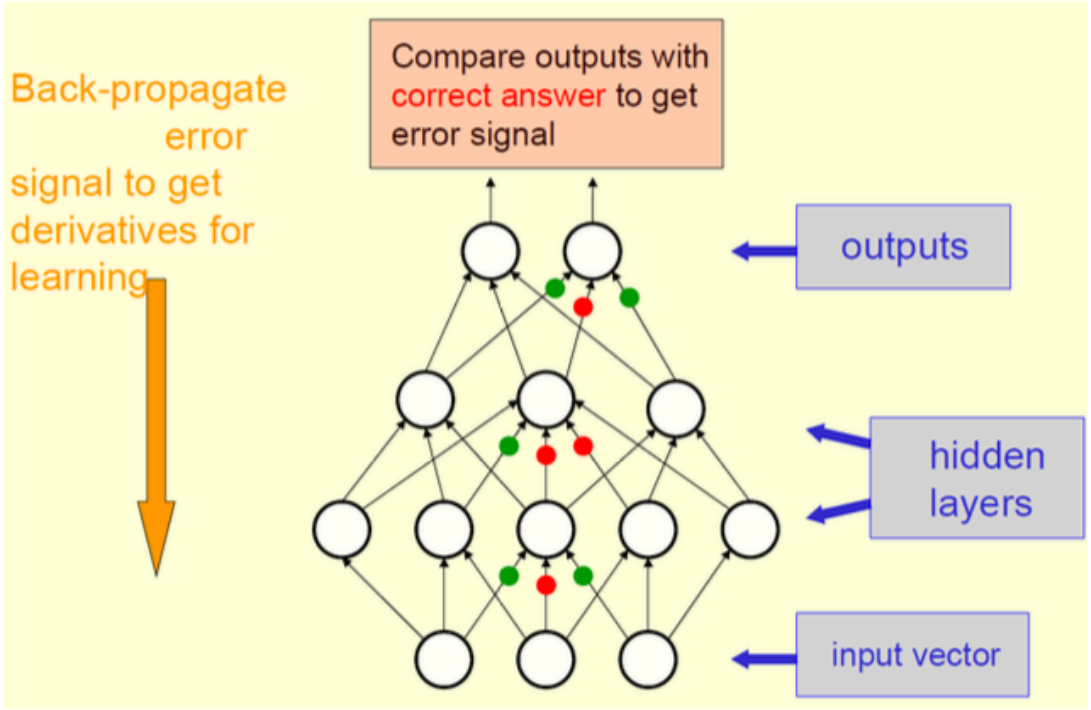
사람의 가청 주파수 대역:
20Hz ~ 20kHz

■ **음성인터페이스의 가능한 입력 개수 분석(계속)**

- 정량적 분석
  - 가정 : 입력 길이 1초, sampling rate 44.1k, sample당 2B사용
  - 저장에 1 * 44100 * 2 = 88,200B 필요
  - 가능한 입력의 개수: $2^{88200*8}$
  - 입력 길이의 제한이 없으므로 가능한 입력의 개수: 무한대

Machine learning vs Deep learning

- **Hidden layer의 개수가 2 이상인 모델을 DNN이라 함**
  - If # of hidden layers <=1 → Shallow neural network
  - If # of hidden layers >=2 → Deep Neural Network (DNN)

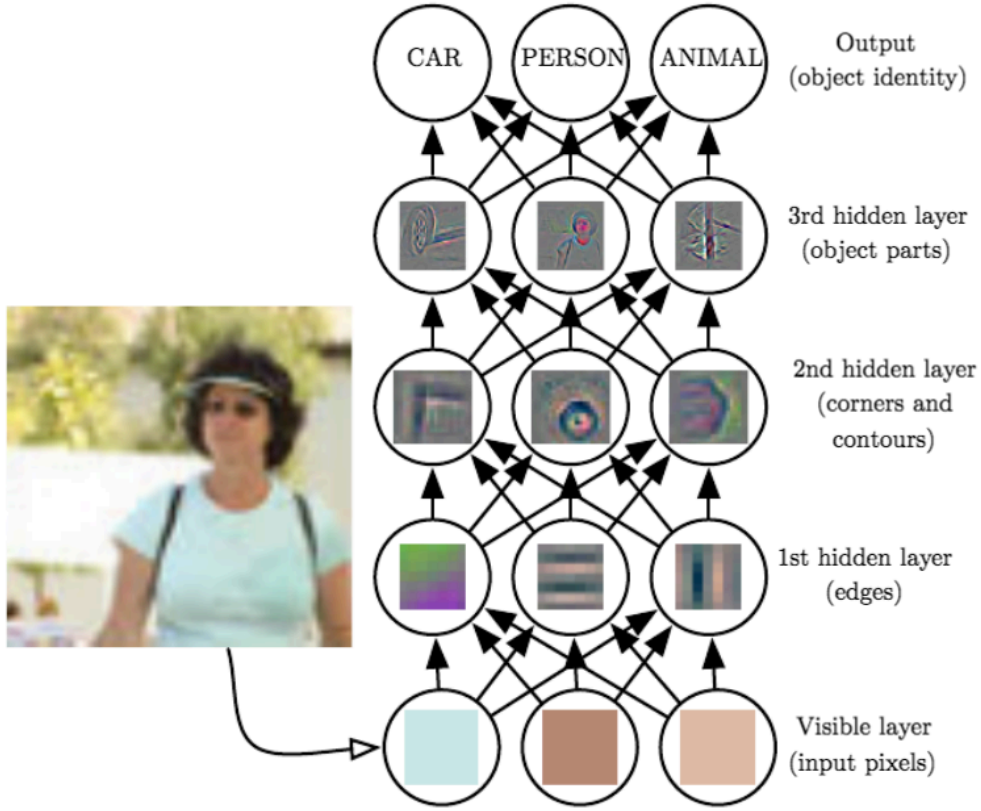


[Hinton, 2013]

A little intuition: adding layers

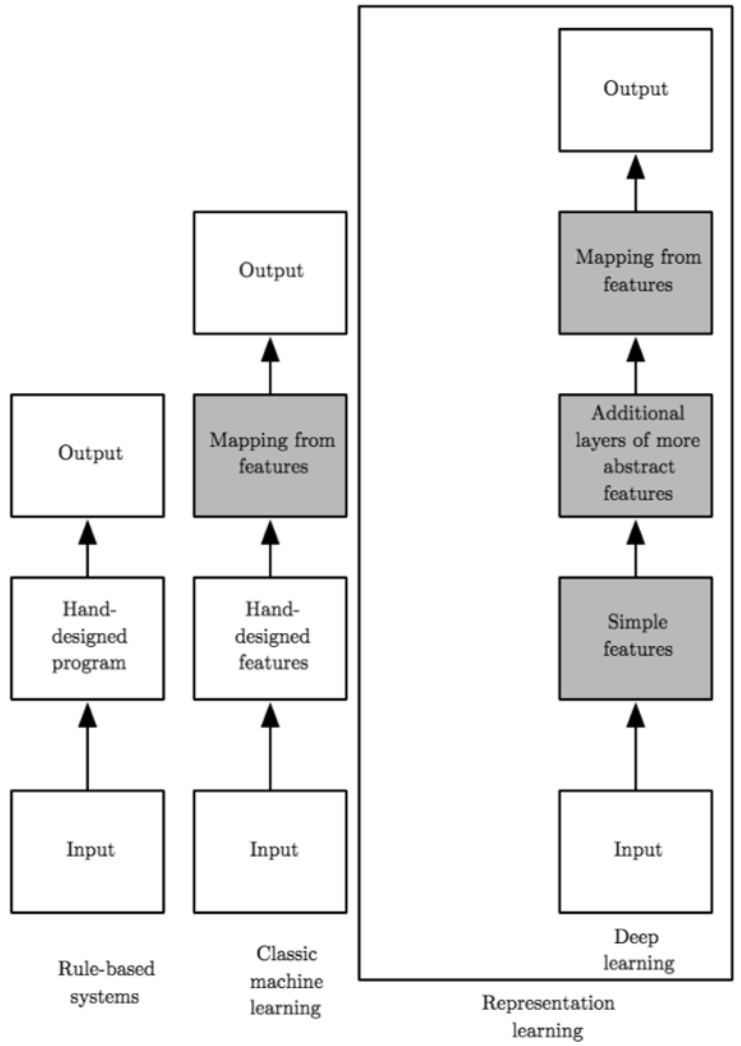
- Builds complex concepts(features) from simpler concepts(features)

- Almost impossible to detect objects in a picture with just raw pixel values



Rule based vs Machine learning vs Deep learning

- The model determines which concepts are useful for explaining the relationships in the observed data



To wrap up

- What is explicit programming? What are the issues?
- What is machine learning? What are the issues?
- What is A.I.? What is A.I. winter?
- What are perceptron, neuron, neural-network, and deep learning?
- How is deep learning different from machine learning and rule-based program?

THANK YOU

SO MUCH

memegenerator.net