Making Deep Learning Understandable for Analyzing Sequential Data about Gene Regulation

Dr. Yanjun Qi

Department of Computer Science

University of Virginia

Tutorial @ ACM BCB-2018

Today

- Machine Learning: a quick review
- Deep Learning: a quick review
- Background Biology: a quick review
- Deep Learning for analyzing Sequential Data about Regulation:
  - DeepChrome
  - AttentiveChrome
  - DeepMotif

https://qdata.github.io/deep2Read/

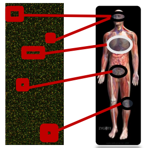
https://www.deepchrome.org

OUR DATA-RICH WORLD

- Biomedicine
  - Patient records, brain imaging, MRI & CT scans, …
  - Genomic sequences, bio-structure, drug effect info, …
- Science
  - Historical documents, scanned books, databases from astronomy, environmental data, climate records, …
- Social media
  - Social interactions data, twitter, facebook records, online reviews, …
- Business
  - Stock market transactions, corporate sales, airline traffic, …

Challenge of Data Explosion in Biomedicine

Molecular signatures of
tumor / blood s

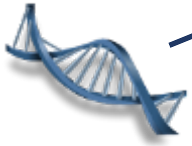


Signs &
Symptoms

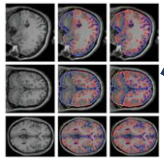
Patient Medical
History &
Demographics

Public Health
Data

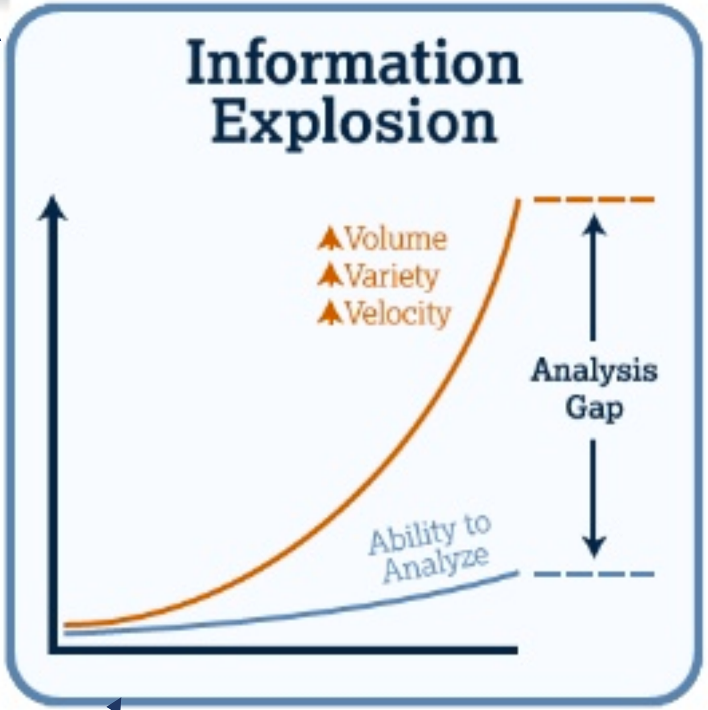
Genetic Data



Mobile medical
sensor data



Medical Images



Traditional
Approaches

**Machine
Learning**

BASICS OF MACHINE LEARNING

- "The goal of machine learning is to build computer systems that can learn and adapt from their experience." – Tom Dietterich

- "Experience" in the form of available data examples (also called as instances, samples)

- Available examples are described with properties (data points in feature space X)

e.g. SUPERVISED LEARNING

- Find function to map input space X to output space Y

$$f : X \longrightarrow Y$$

- So that the difference between *y* and *f(x)* of each example ***x*** is small.

e.g.

x	
	I believe that this book is not at all helpful since it does not explain thoroughly the material . it just provides the reader with tables and calculations that sometimes are not easily understood …

Input X : e.g. a piece of English text

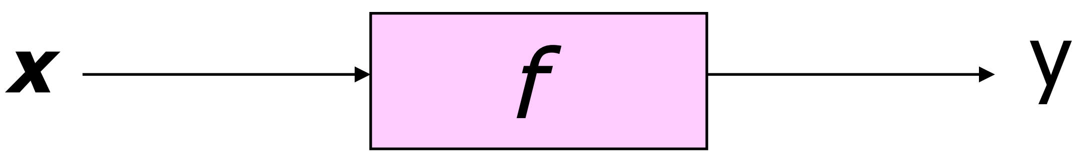
→

y	-1

Output Y: {1 / Yes , -1 / No }
e.g. Is this a positive product review ?

SUPERVISED Linear Binary Classifier

- Now let us check out a VERY SIMPLE case of

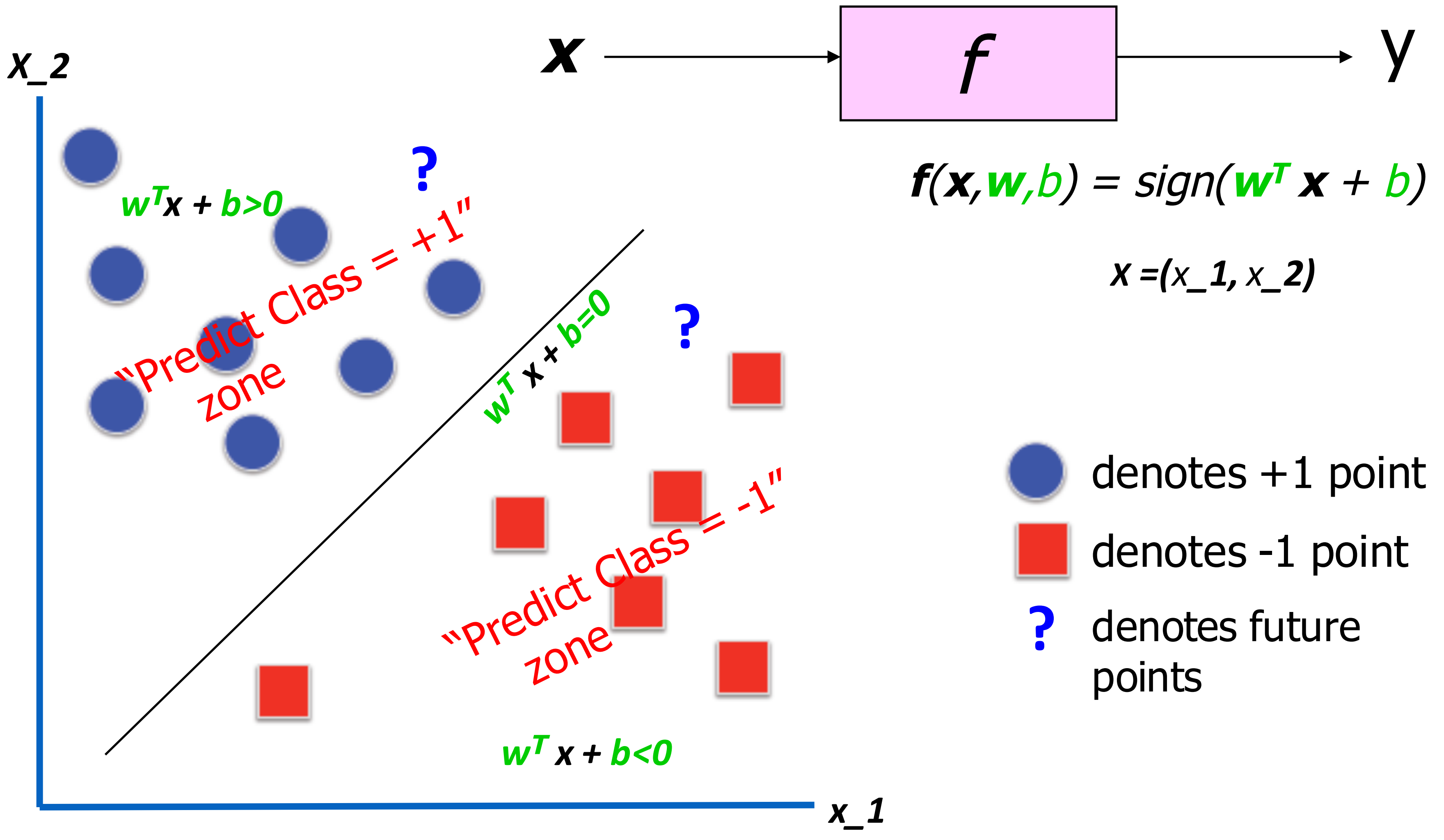


e.g.: Binary y / Linear f / X as R^2

$$f(\mathbf{x},\mathbf{w},b) = sign(\mathbf{w}^T \mathbf{x} + b)$$

$$x = (x\_1, x\_2)$$

SUPERVISED Linear Binary Classifier



$f(\mathbf{x},\mathbf{w},b) = sign(\mathbf{w}^T \mathbf{x} + b)$

$\mathbf{x} = (x\_1, x\_2)$

Courtesy slide from Prof. Andrew Moore's tutorial

Basic Concepts

- Training (i.e. learning parameters **w**,*b*)
  - Training set includes
    - available examples' feature represenation: $\boldsymbol{x}_1,\ldots,\boldsymbol{x}_L$
    - available corresponding labels $y_1,\ldots,y_L$

- Find (**w**,*b*) by minimizing loss (i.e. difference between y and $f(\boldsymbol{x})$ *on available examples in training set*)

$$(\mathbf{W}, b) = \underset{\mathbf{w},\, b}{\operatorname{argmin}} \sum_{i=1}^{L} \ell(f(x_i), y_i)$$

Basic Concepts

- Testing (i.e. evaluating performance on "future" points)
  - Difference between true $y_?$ and the predicted $f(\mathbf{x}_?)$ on a set of testing examples (i.e. *testing set*)
  - Key: example $\mathbf{x}_?$ not in the training set

- Generalisation: learn function / hypothesis from past data in order to "explain", "predict", "model" or "control" new data examples

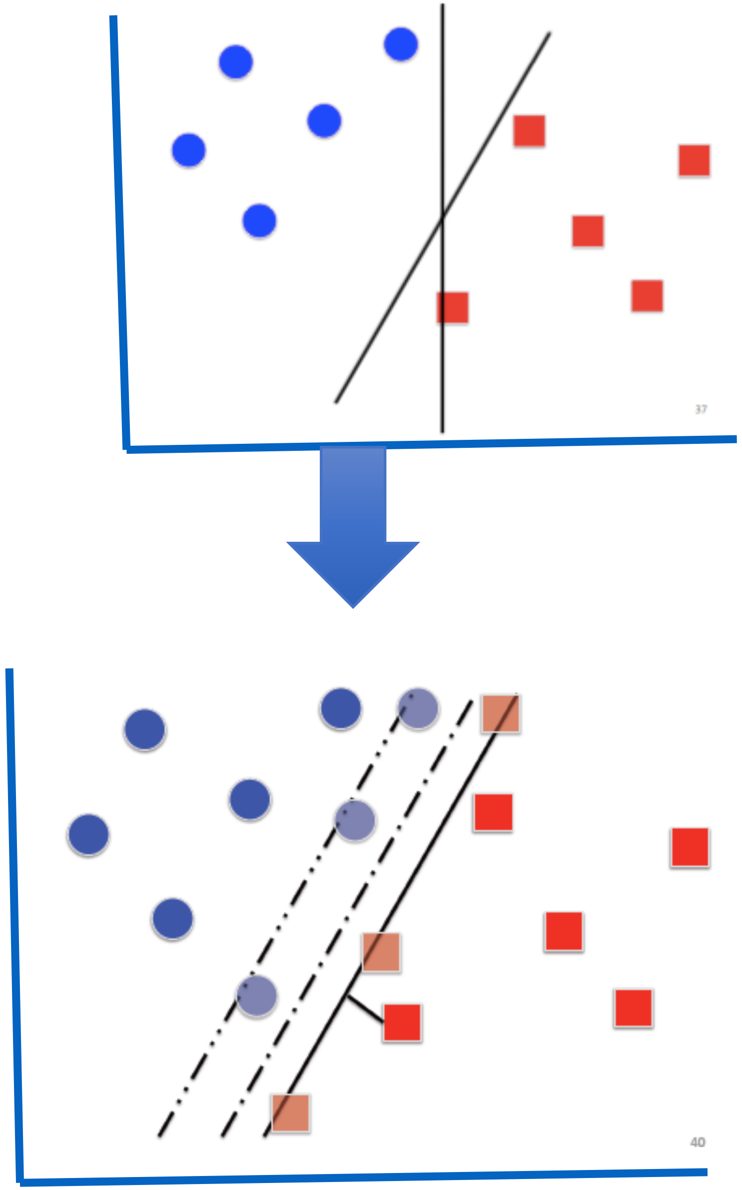
Basic Concepts

- Loss function
  - e.g. hinge loss for binary classification task

$$\sum_{i=1}^{L} \ell(f(x_i), y_i) = \sum_{i=1}^{L} \max(0, 1 - y_i f(x_i)).$$

- Regularization
  - E.g. additional information added on loss function to control f

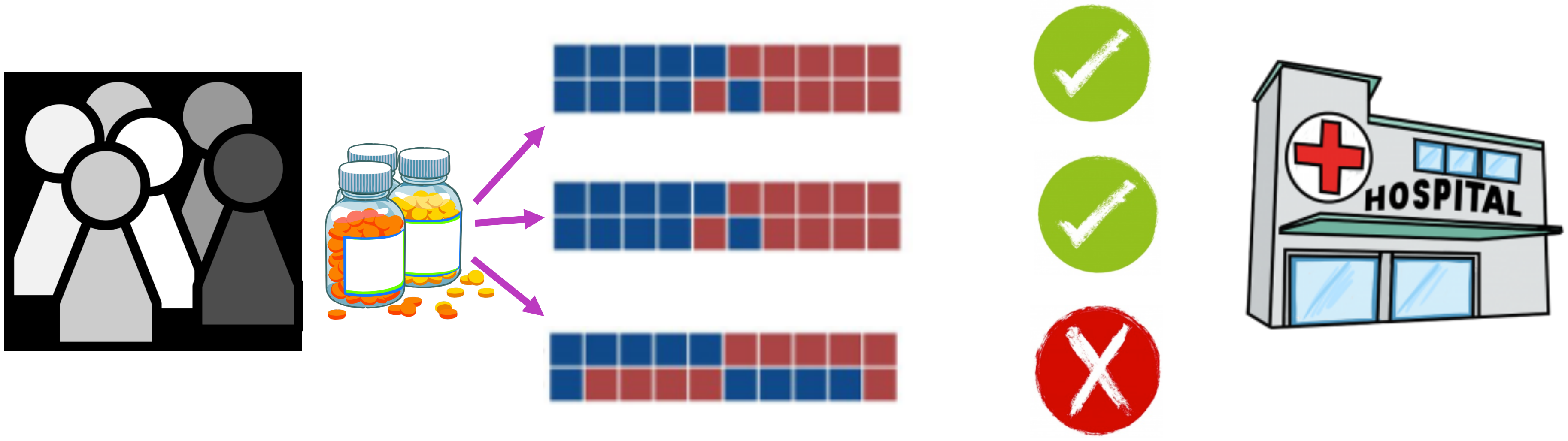
Maximize Separation Margin => Minimize $\|w\|^2$

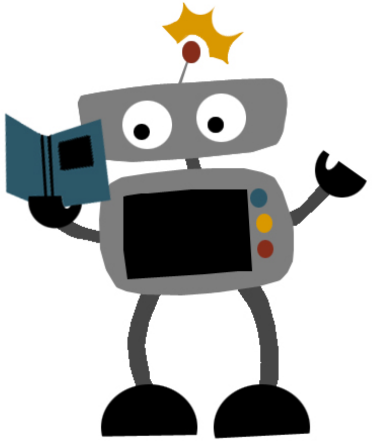


Basics of Machine Learning

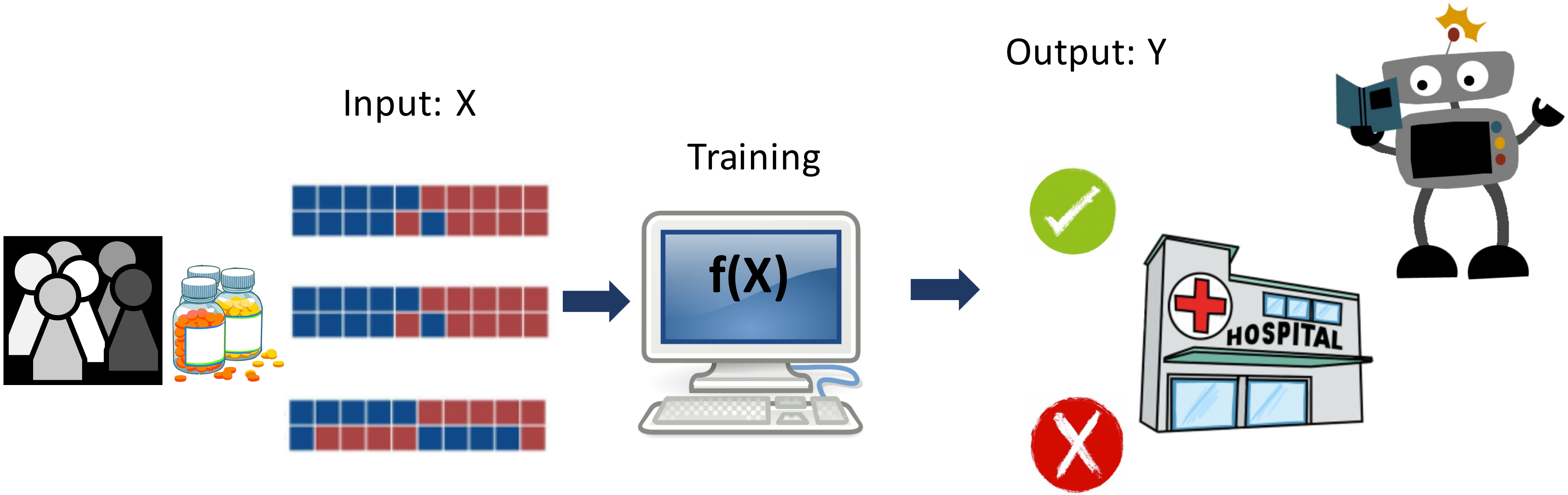
Input: X

Output: Y



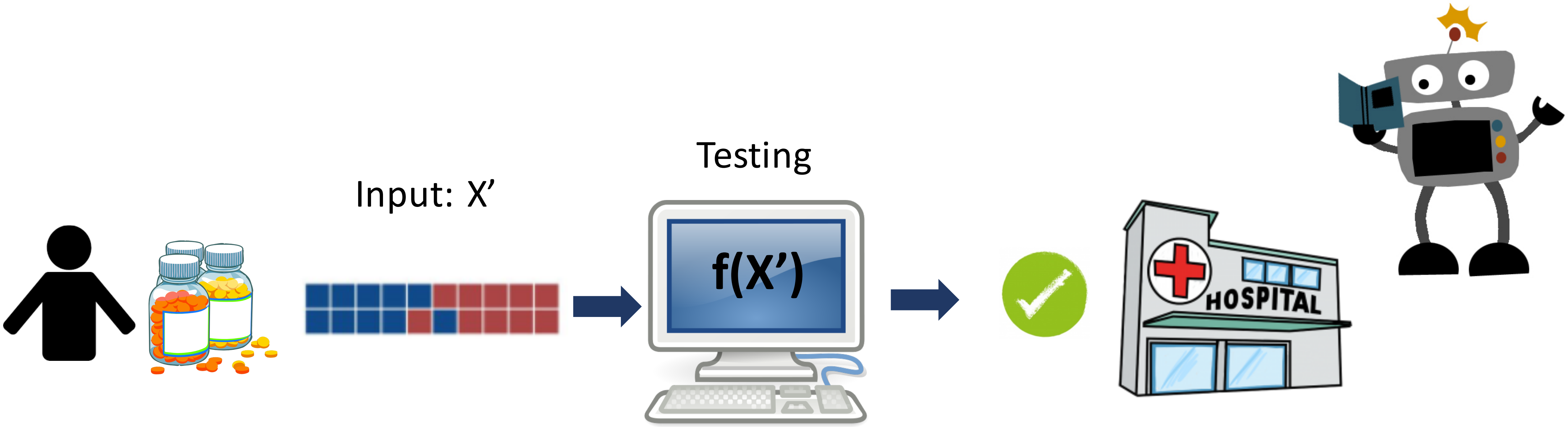


Basics of Machine Learning



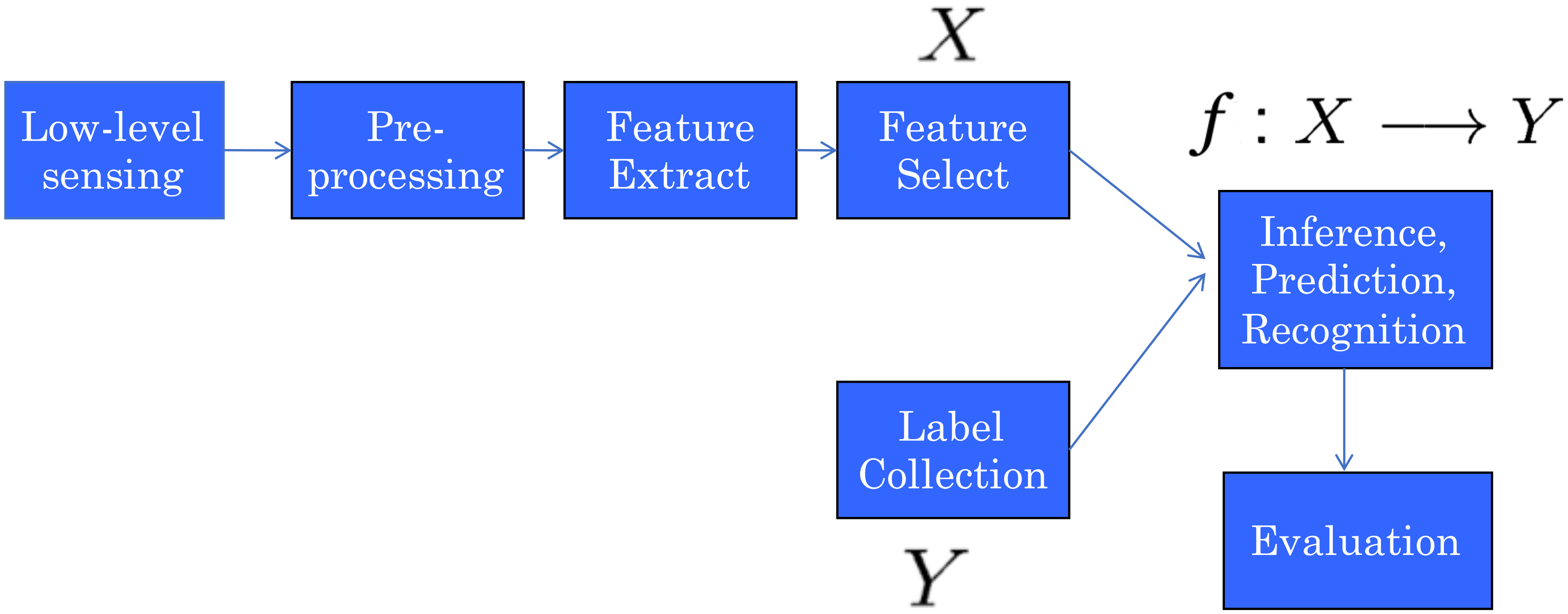
f(X)=Y

Basics of Machine Learning

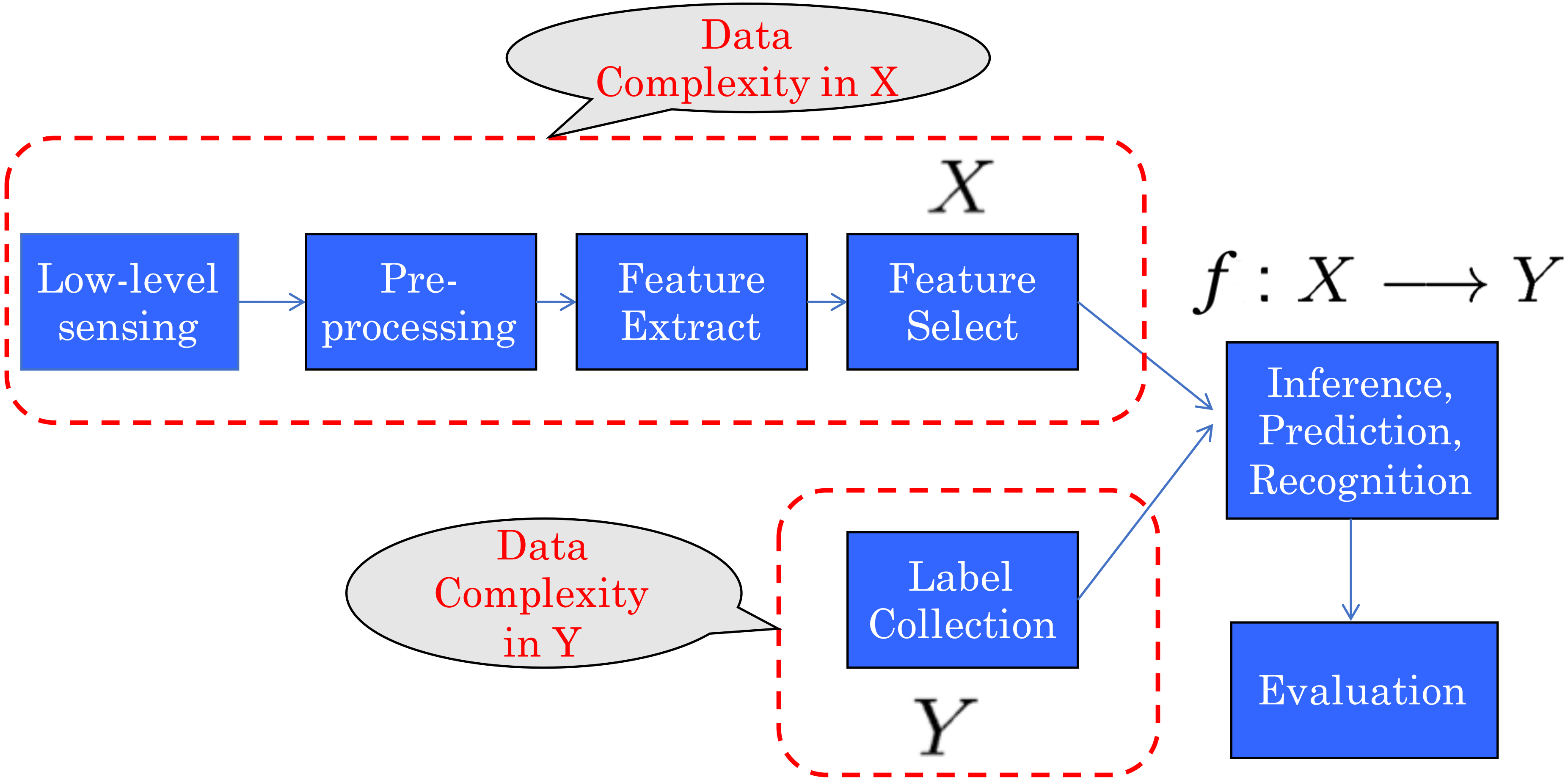


Supervised Classification

TYPICAL MACHINE LEARNING SYSTEM

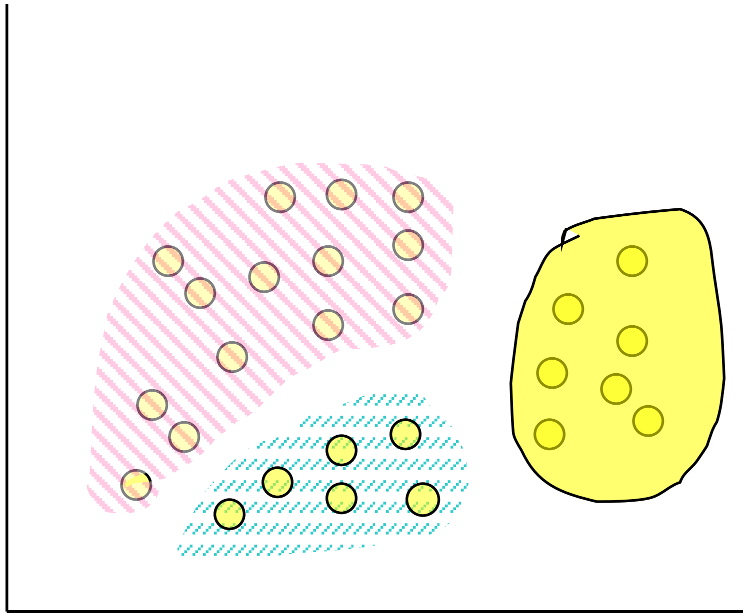


TYPICAL MACHINE LEARNING SYSTEM



UNSUPERVISED LEARNING : [COMPLEXITY OF Y]

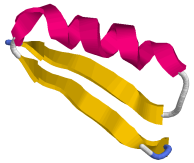
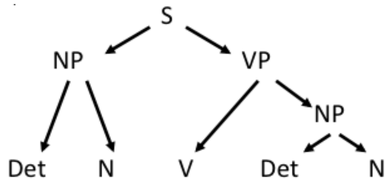
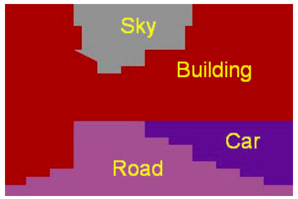
- No labels are provided (e.g. No Y provided)
- Find patterns from unlabeled data, e.g. clustering



e.g. clustering => to find "natural" grouping of instances given un-labeled data

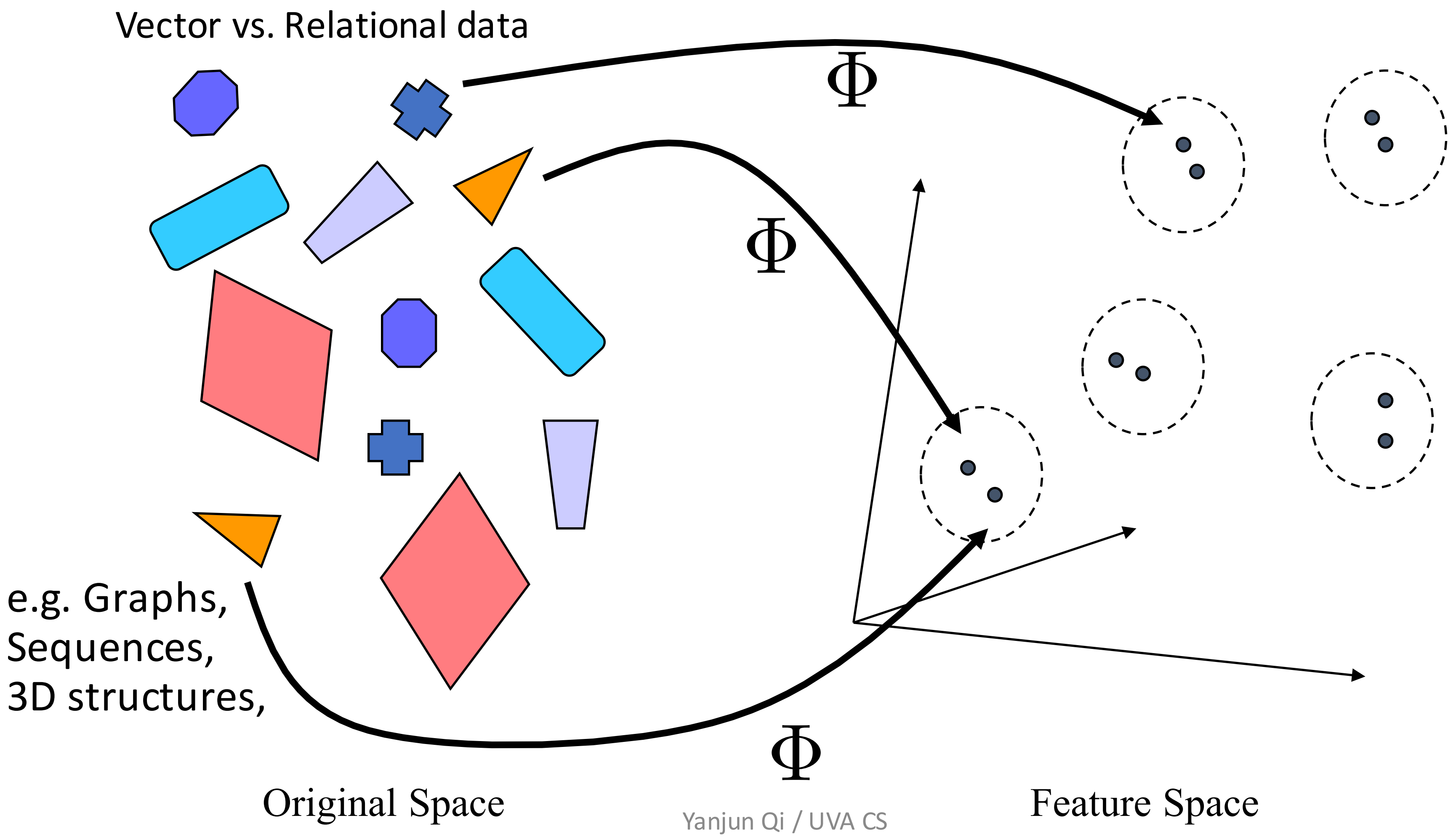
Structured Output Prediction:
[COMPLEXITY in Y]

- Many prediction tasks involve output labels having structured correlations or constraints among instances

Structured Dependency between Examples' Y	Sequence	Tree	Grid
Input X	APAFSVSPASGACGPECA…	The dog chased the cat	
Output Y	 CCEEEEECCCCCHHHCCC…		

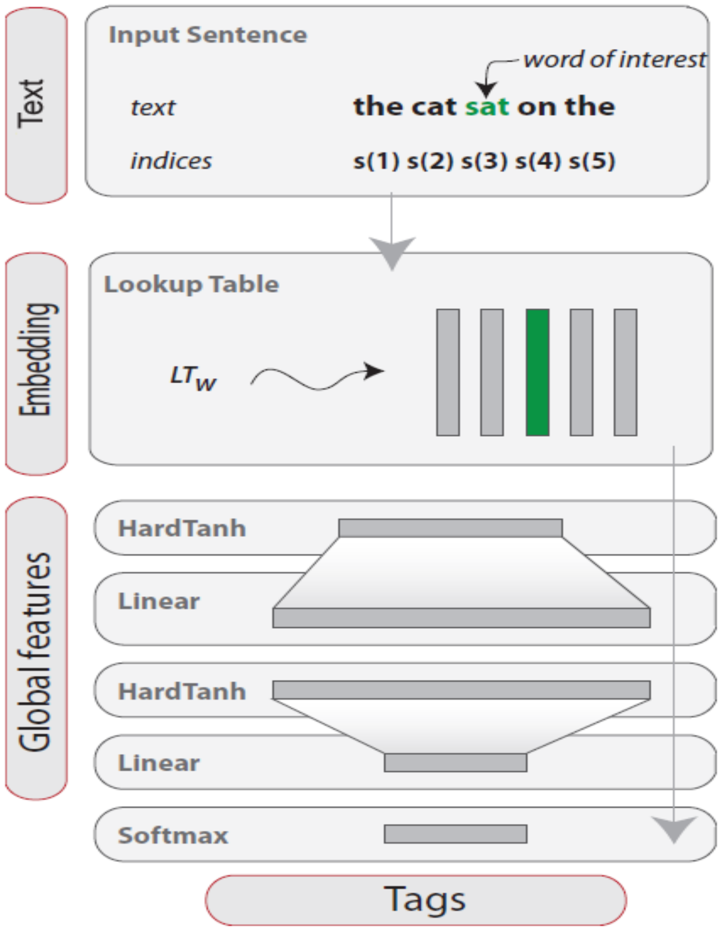
Many more possible structures between y_i , e.g. spatial , temporal, relational …

Structured Input: Kernel Methods
[COMPLEXITY OF X]

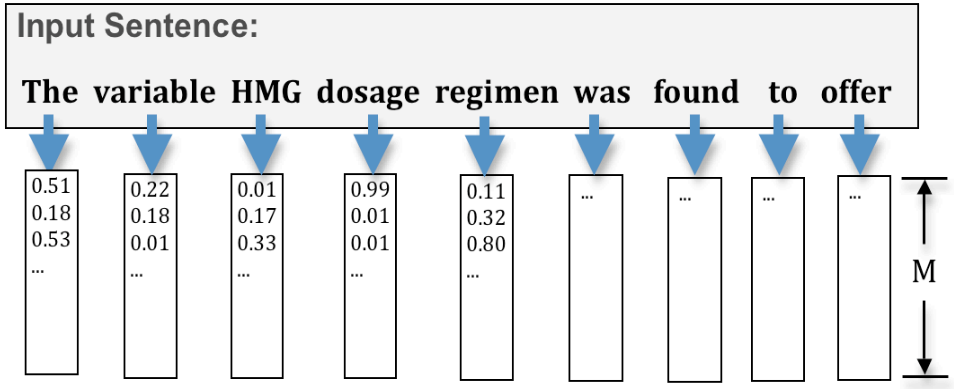


More Recent: Representation Learning
[COMPLEXITY OF X]

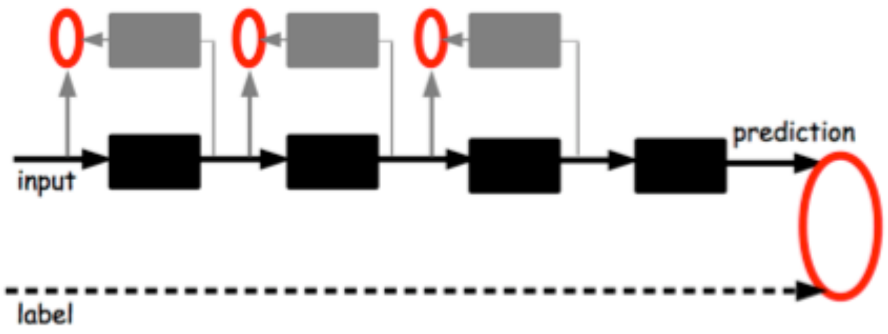
Deep Learning

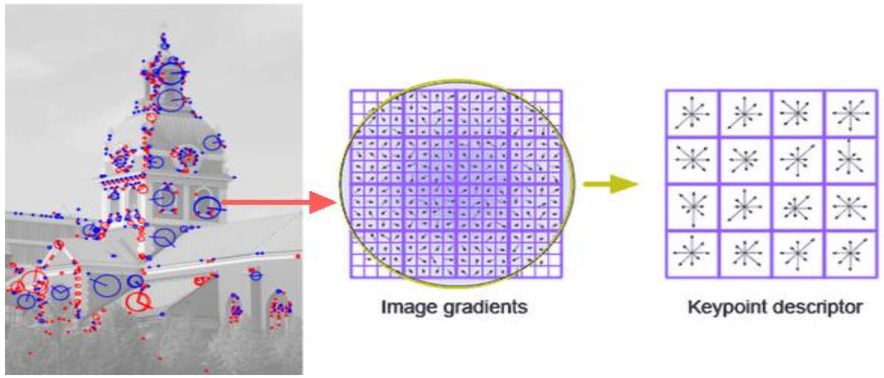


Supervised Embedding

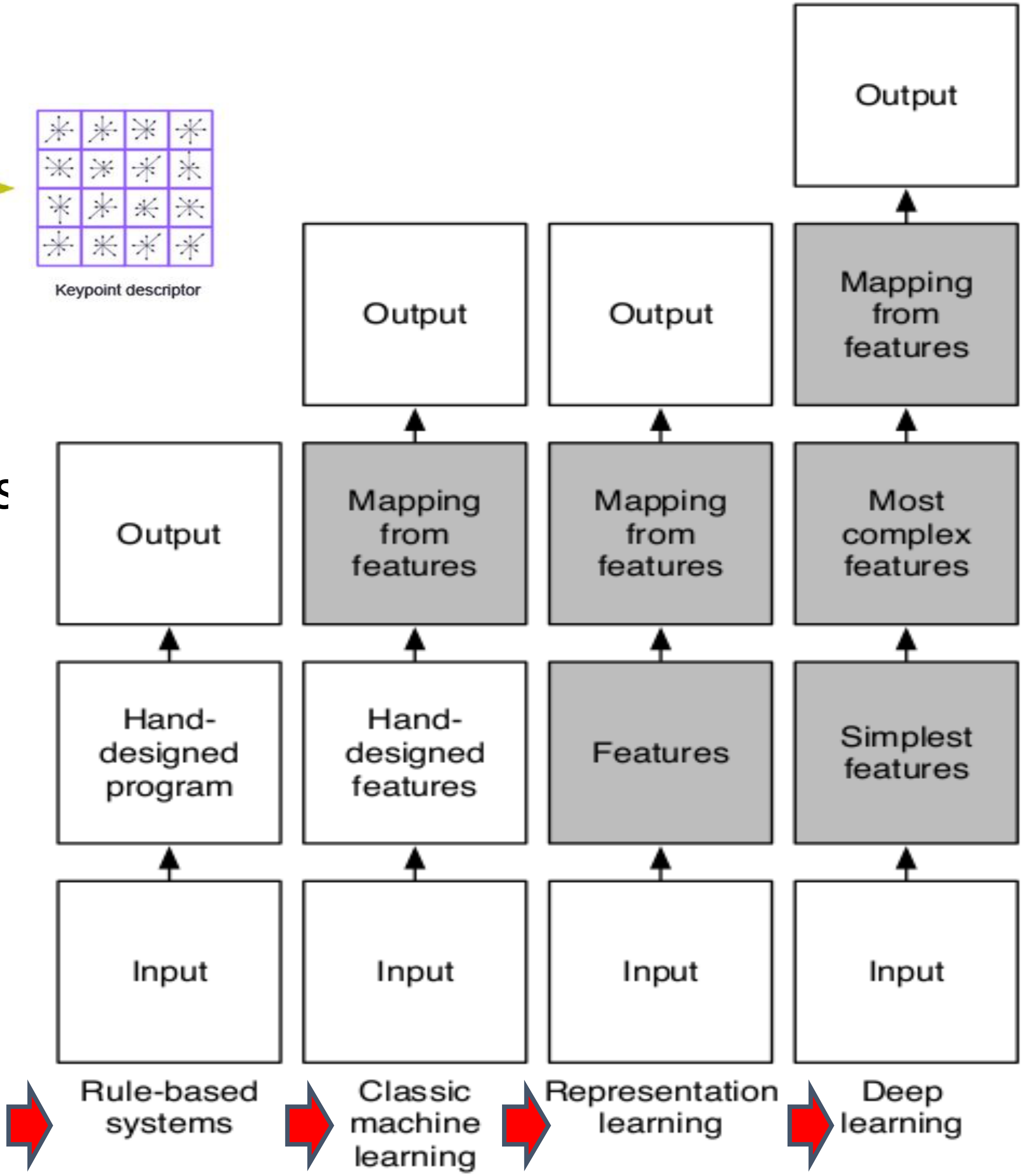


Layer-wise Pretraining





Why learn features



When to use Machine Learning ?

- 1. Extract knowledge from data
  - Relationships and correlations can be hidden within large amounts of data
  - The amount of knowledge available about certain tasks is simply too large for explicit encoding (e.g. rules) by humans

- 2. Learn tasks that are difficult to formalise
  - Hard to define well, except by examples (e.g. face recognition)
- 3. Create software that improves over time
  - New knowledge is constantly being discovered.
  - Rule or human encoding-based system is difficult to continuously re-design "by hand".

Recap

$$f : X \longrightarrow Y$$

- Goal of Machine Learning: Generalisation
- Training
- Testing
- Loss

Today

- Machine Learning: a quick review
- Deep Learning: a quick review
- Background Biology: a quick review
- Deep Learning for analyzing Sequential Data about Regulation:
  - DeepChrome
  - AttentiveChrome
  - DeepMotif

https://qdata.github.io/deep2Read/

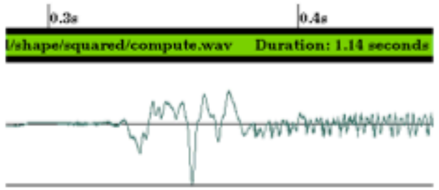
https://www.deepchrome.org

- Deep Learning
  - Why is this a breakthrough ?
  - Basics
  - History
  - A Few Recent trends

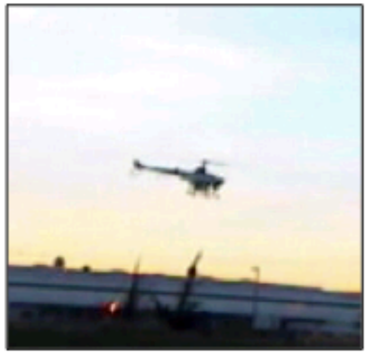
https://qdata.github.io/deep2Read/

Deep Learning is Changing the World

How may I help you, human?



Speech Recognition



Control learning

Object recognition

Text analysis

Peter H. van Oppen , Chairman of the Board & Chief Executive Officer
Mr. van Oppen has served as chairman of the board and chief executive officer of ADIC since its acquisition by Interpoint in 1994 and a director of ADIC since 1986. Until its acquisition by Crane Co. in October 1996, Mr. van Oppen served as chairman of the board of directors, president and chief executive officer of Interpoint . Prior to 1985, Mr. van Oppen worked as a consulting manager at Price Waterhouse LLP and at Bain & Company in Boston and London. He has additional experience in medical electronics and venture capital. Mr. van Oppen also serves as a director of Seattle FilmWorks Inc. and Spacelabs Medical, Inc.. He holds a B.A. from Whitman College and an M.B.A. from Harvard Business School, where he was a Baker Scholar.

Many more !

MIT Technology Review

10 Breakthrough Technologies 2013

Think of the most frustrating, intractable, or simply annoying problems you can imagine. Now think about what technology is doing to fix them. That's what we did in coming up with our annual list of 10 Breakthrough Technologies. We're looking for technologies that we believe will expand the scope of human possibilities.

Deep Learning

10 Breakthrough Technologies 2017

These technologies all have staying power. They will affect the economy and our politics, improve medicine, or influence our culture. Some are unfolding now; others will take a decade or more to develop. But you should know about all of them right now.

Deep Reinforcement Learning

10 BREAKTHROUGH TECHNOLOGIES 2018

Generative Adversarial Network (GAN)

Why breakthrough ?

Breakthrough from 2012 Large-Scale Visual Recognition Challenge (ImageNet)

10% improve with deepCNN

72%, 2010

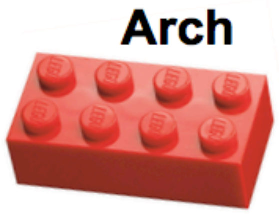
74%, 2011

85%, 2012

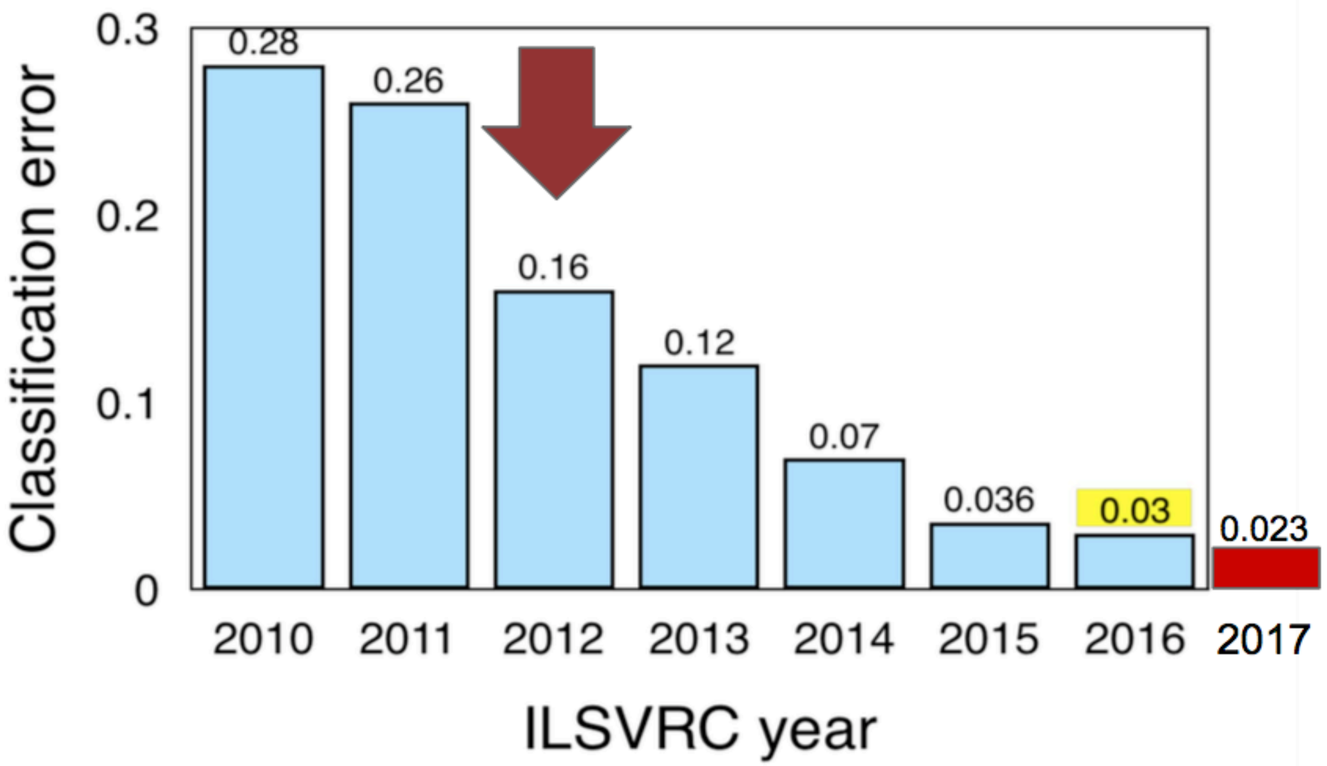
In one "very large-scale" benchmark competition

(1.2 million images [X] vs.1000 different word labels [Y])

ImageNet Challenge



- 2010-11: hand-crafted computer vision pipelines
- 2012-2016: ConvNets
  - 2012: AlexNet
    - major deep learning success
  - 2013: ZFNet
    - improvements over AlexNet
  - 2014
    - VGGNet: deeper, simpler
    - InceptionNet: deeper, faster
  - 2015
    - ResNet: even deeper
  - 2016
    - ensembled networks
  - 2017
    - Squeeze and Excitation Network

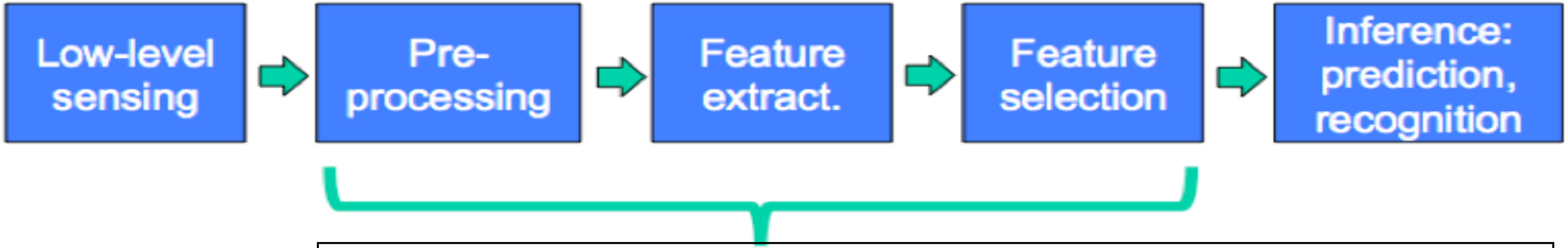


Adapt from From NIPS 2017 DL Trend Tutorial

DNNs help us build more intelligent computers

- Perceive the world,
  - e.g., objective recognition, speech recognition, …
- Understand the world,
  - e.g., machine translation, text semantic understanding
- Interact with the world,
  - e.g., AlphaGo, AlphaZero, self-driving cars, …
- Being able to think / reason,
  - e.g., learn to code programs, learn to search deepNN, …
- Being able to imagine / to make analogy,
  - e.g., learn to draw with styles, ……

Deep Learning Way: Learning Representation from data



Feature Engineering

- ✓ Most critical for accuracy
- ✓ Account for most of the computation
- ✓ Most time-consuming in development cycle
- ✓ Often hand-craft and task dependent in practice

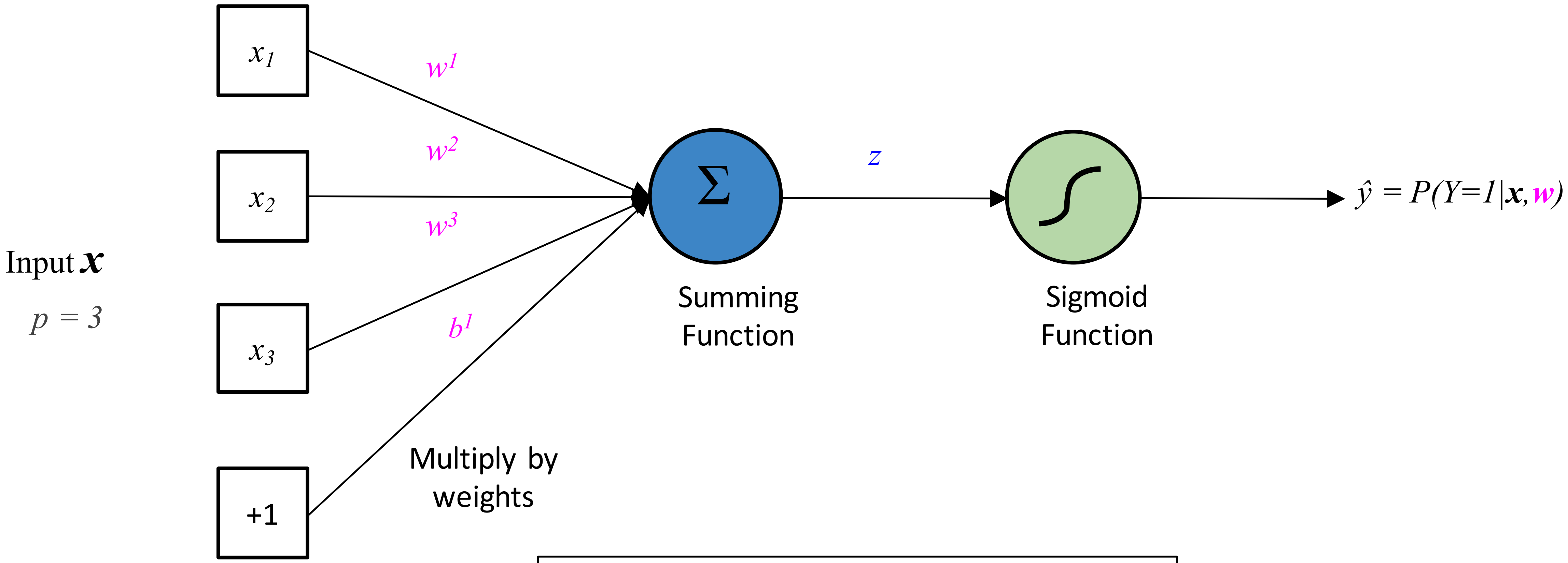
Feature Learning

- ✓ Easily adaptable to new similar tasks
- ✓ Learn layerwise representation from data

Basics

- Basic Neural Network (NN)
  - single neuron, e.g. logistic regression unit
  - multilayer perceptron (MLP)
  - various loss function
    - E.g., when for multi-class classification, softmax layer
  - training NN with backprop algorithm

One "Neuron": Expanded Logistic Regression

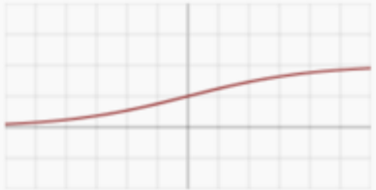
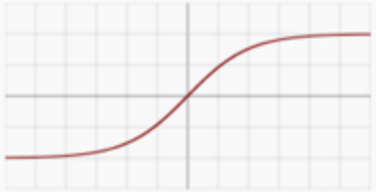
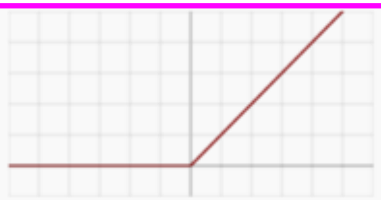


$$z = \boldsymbol{w}^T \cdot \boldsymbol{x} + \boldsymbol{b}$$

$$y = sigmoid(z) = \frac{e^z}{1 + e^z}$$

E.g., Many Possible Nonlinearity Functions

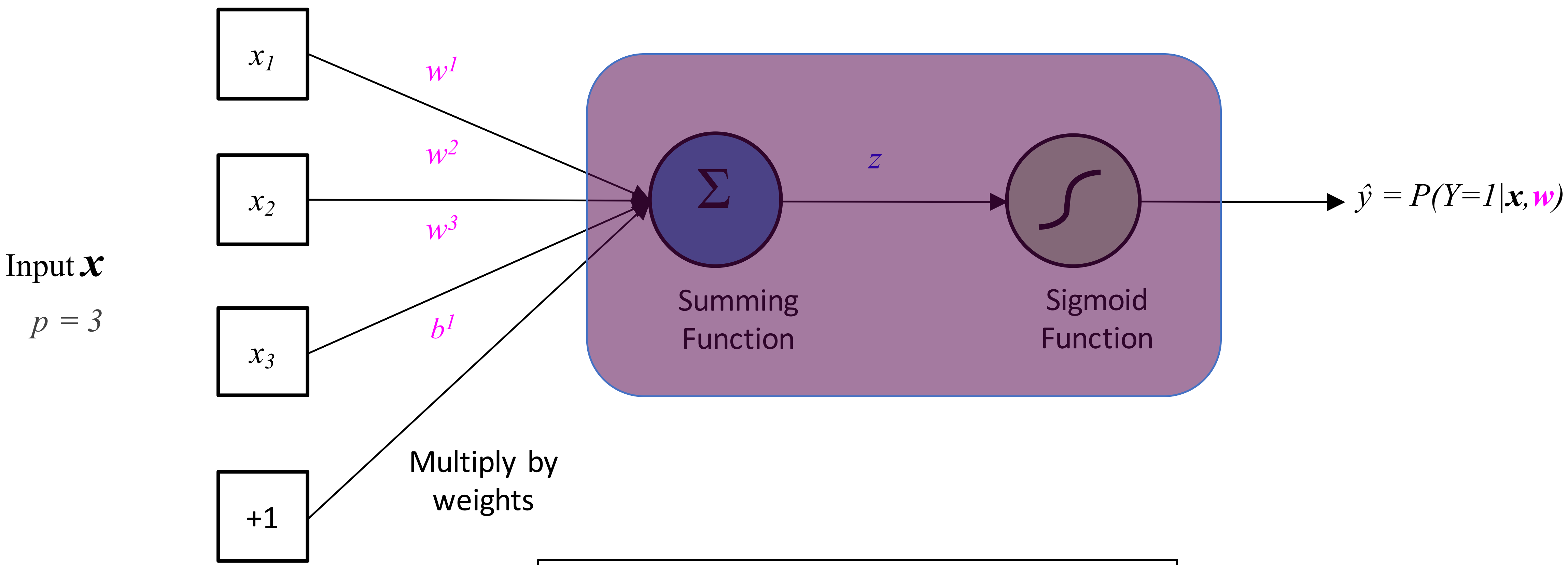
(aka transfer or activation functions)

Name	Plot	Equation	Derivative (w.r.t x)
Binary step		$f(x) = \begin{cases} 0 & \text{for} \quad x < 0 \\ 1 & \text{for} \quad x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for} \quad x \neq 0 \\ ? & \text{for} \quad x = 0 \end{cases}$
Logistic (a.k.a Soft step)		$f(x) = \frac{1}{1 + e^{-x}}$	$f'(x) = f(x)(1 - f(x))$
TanH		$f(x) = \tanh(x) = \frac{2}{1 + e^{-2x}} - 1$	$f'(x) = 1 - f(x)^2$
Rectifier (ReLU)[9]		$f(x) = \begin{cases} 0 & \text{for} \quad x < 0 \\ x & \text{for} \quad x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for} \quad x < 0 \\ 1 & \text{for} \quad x \geq 0 \end{cases}$

usually works best in practice

https://en.wikipedia.org/wiki/Activation_function#Comparison_of_activation_functions

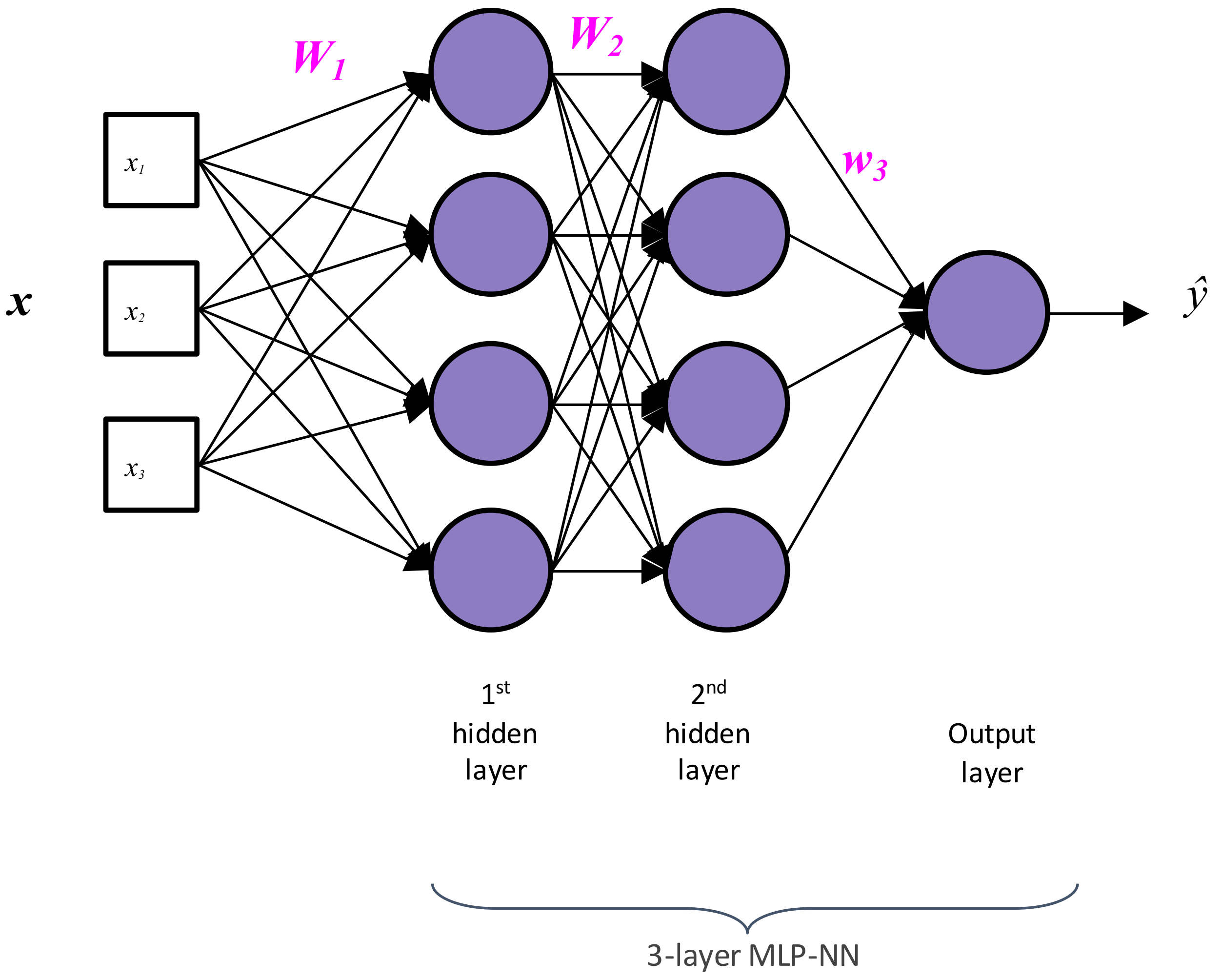
One "Neuron": Expanded Logistic Regression => "Neuron View"



$$z = \boldsymbol{w}^T \cdot \boldsymbol{x} + \boldsymbol{b}$$

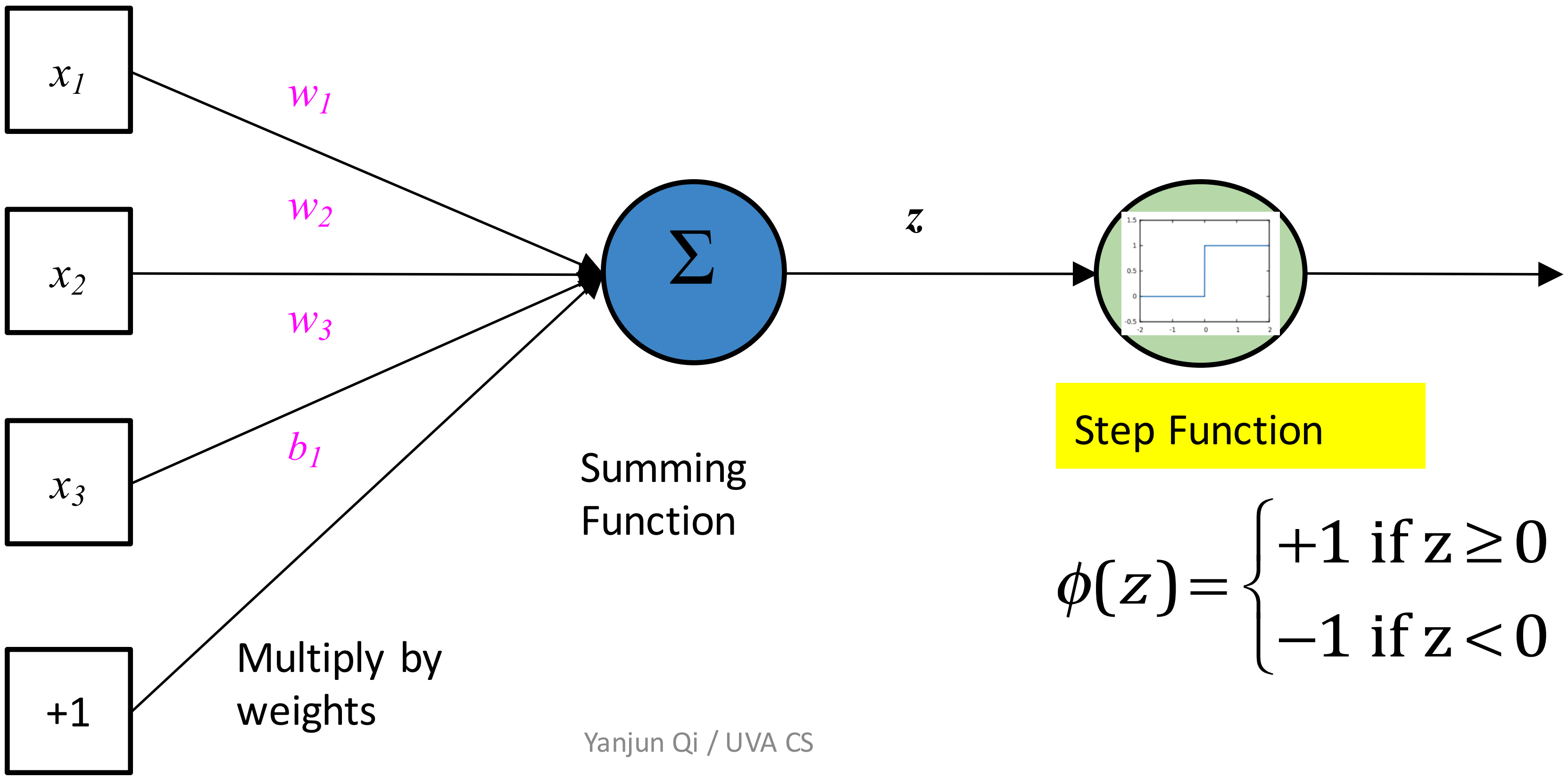
$$y = sigmoid(z) = \frac{e^z}{1 + e^z}$$

Multi-Layer Perceptron (MLP)- (Feed-Forward NN)



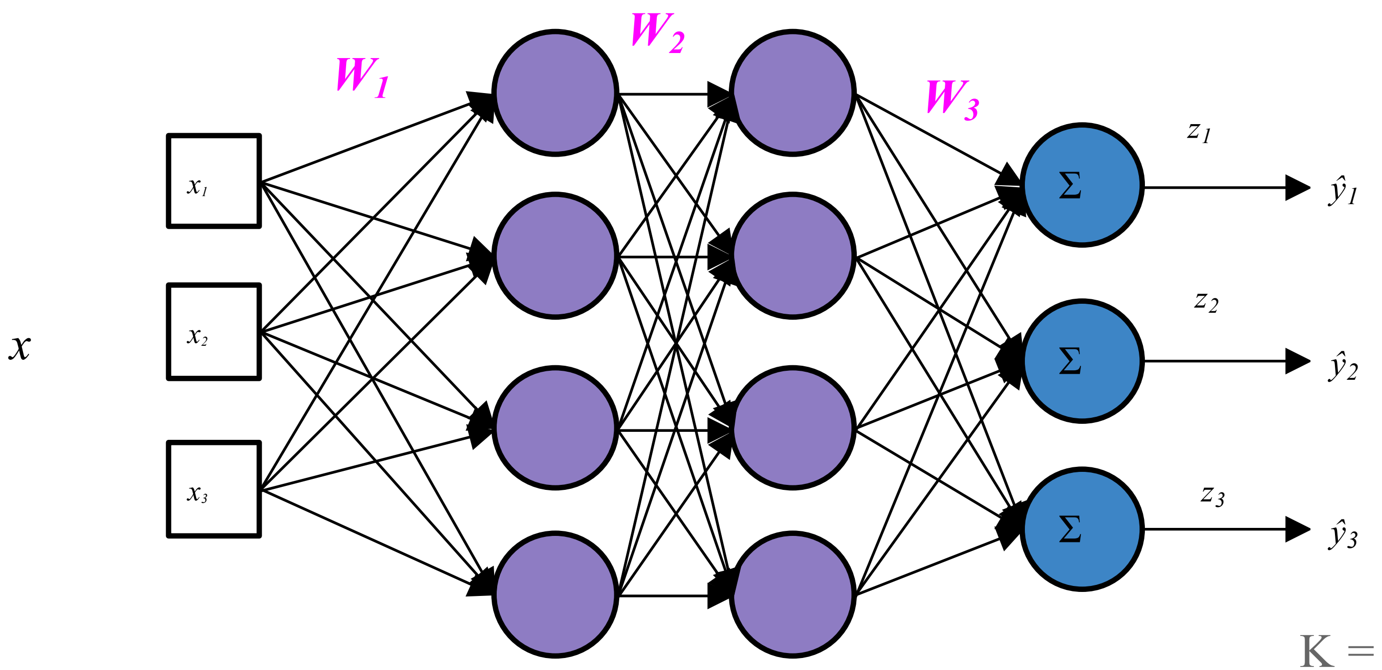
History → Perceptron: 1-Neuron Unit with Step

- First proposed by Rosenblatt (1958)
- A simple neuron that is used to classify its input into one of two categories.
- A perceptron uses a **step function**



$$\phi(z) = \begin{cases} +1 \text{ if } z \geq 0 \\ -1 \text{ if } z < 0 \end{cases}$$

E.g., Cross-Entropy Loss for Multi-Class Classification

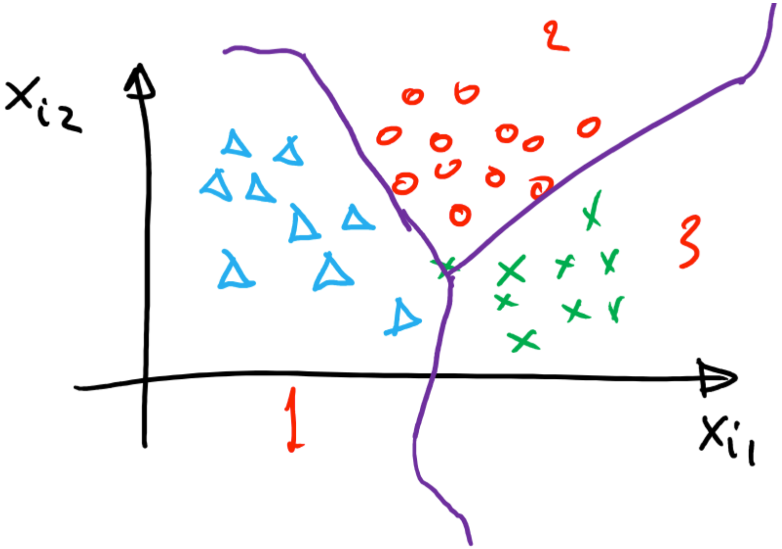


$K = 3$

$$\hat{y}_i = \frac{e^{z_i}}{\sum_j e^{z_j}} = P(\hat{y}_i = 1 \mid \boldsymbol{x})$$

"Softmax" function. Normalizing function which converts each class output to a probability.

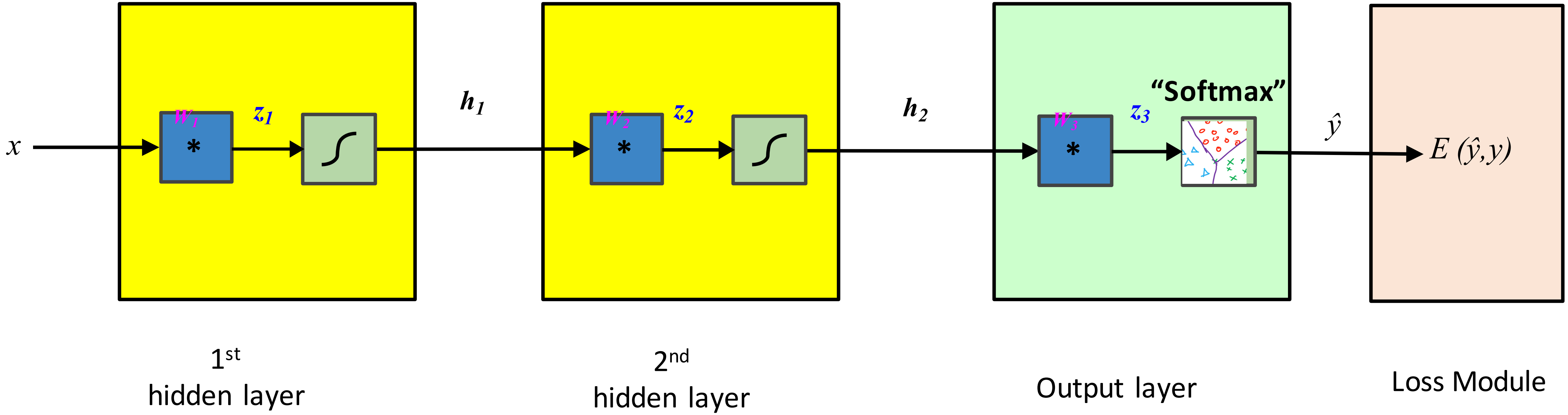
Y_{i1}	Y_{i2}	Y_{i3}	
0	1	0	Class 2
1	0	0	Class 1
0	0	1	Class 3



$$E_W(\hat{y}, y) = \text{loss} = \sum_{j=1...K} y_j \ln \hat{y}_j$$

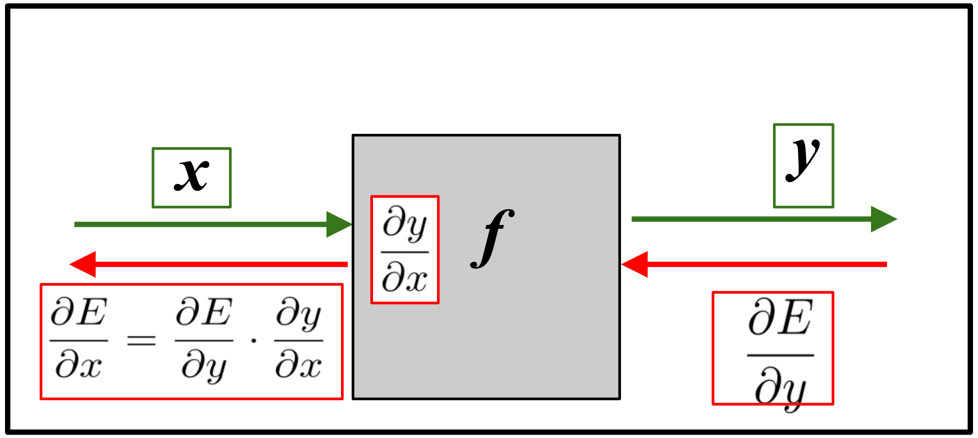
Cross-entropy loss

“Block View”



x
W_1
*
z_1
h_1
W_2
*
z_2
h_2
W_3
*
z_3
“Softmax”
$\hat{y}$
$E(\hat{y},y)$
1st
hidden layer
2nd
hidden layer
Output layer
Loss Module

Building Deep Neural Nets



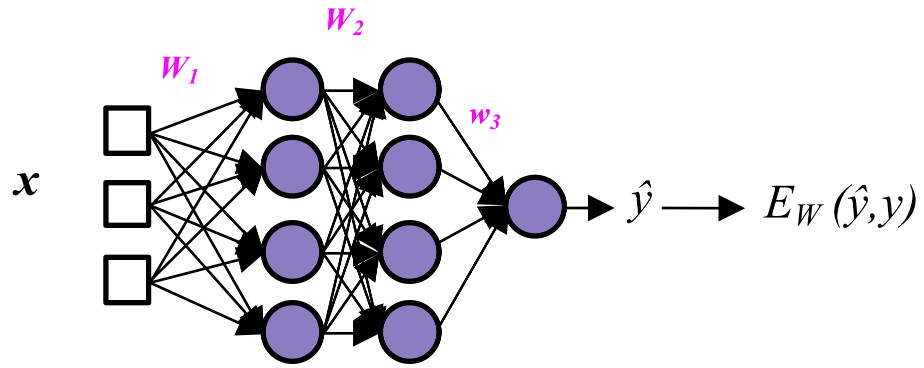
http://cs231n.stanford.edu/slides/winter1516_lecture5.pdf

Training Neural Networks

How do we learn the optimal weights W_L for our task??

- **Stochastic Gradient descent:**

$$W_L^{t} = W_L^{t-1} - \eta \frac{\partial E}{\partial W_L}$$



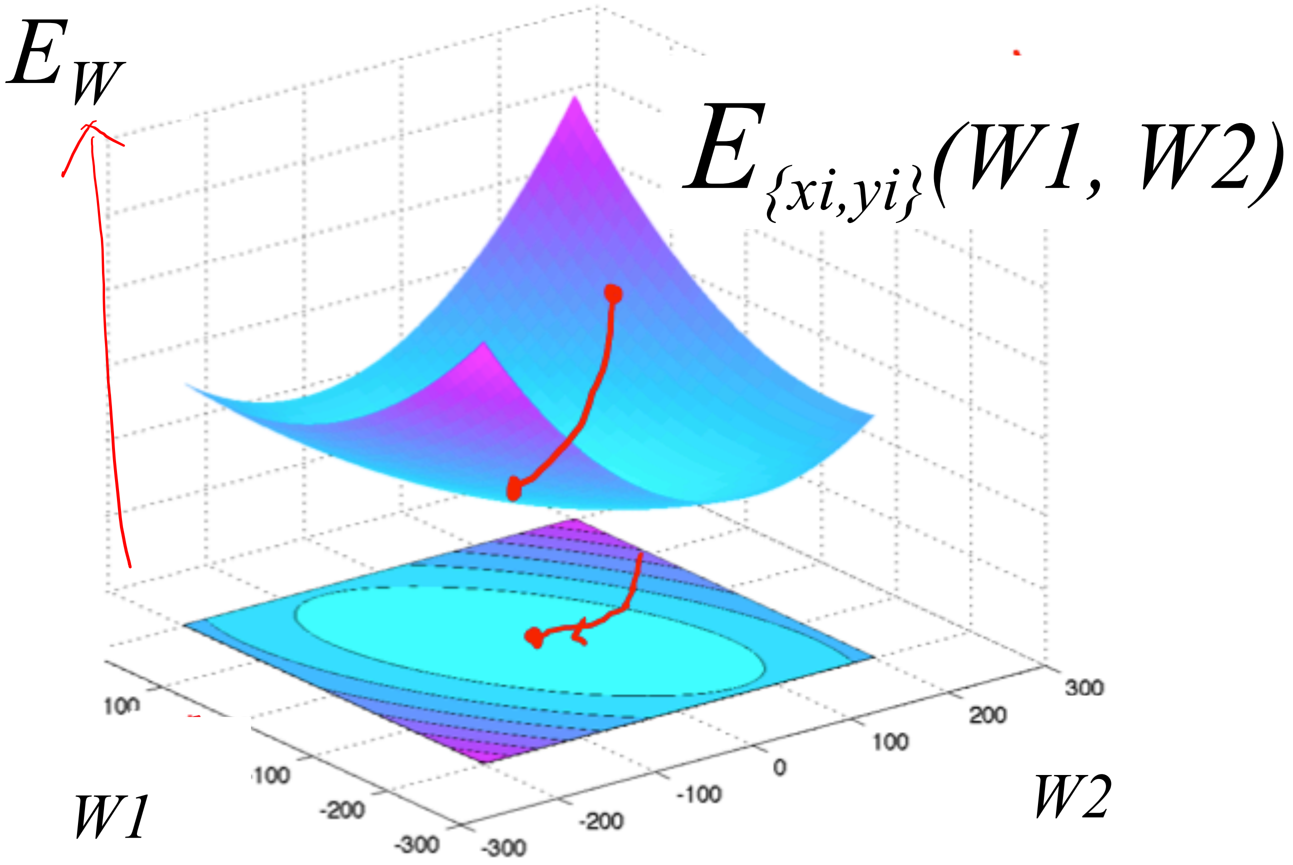
But how do we get gradients of lower layers?

- **Backpropagation!**
  - Repeated application of chain rule of calculus
  - Locally minimize the objective
  - Requires all "blocks" of the network to be differentiable

– Main Idea: error in hidden layers

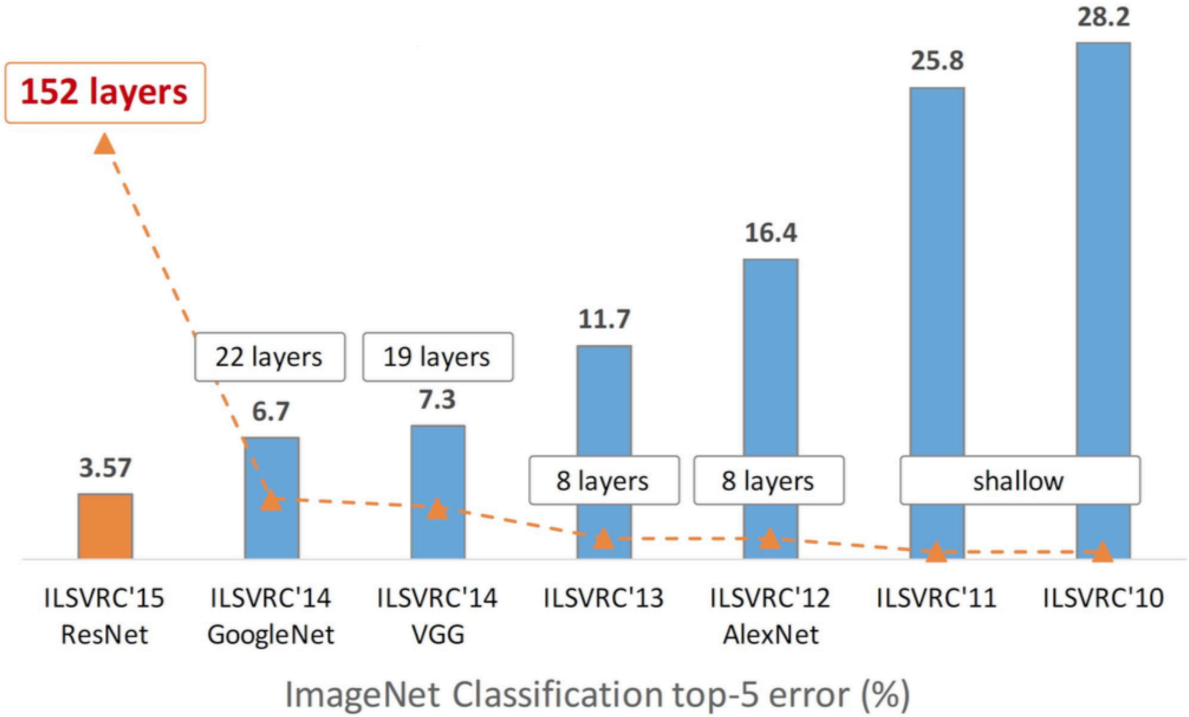
LeCun et. al. *Efficient Backpropagation*. 1998

Illustrating Objective Loss Function (extremely simplified) and Gradient Descent (2D case)



The gradient points in the direction (in the variable space) of the greatest rate of increase of the function and its magnitude is the slope of the surface graph in that direction

Revolution of Depth



Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Deep Residual Learning for Image Recognition". CVPR 2016.

Adapt from From NIPS 2017 DL Trend Tutorial

Important Block: Convolutional Neural Networks (CNN)

- Prof. Yann LeCun invented CNN in 1998
- First NN successfully trained with many layers



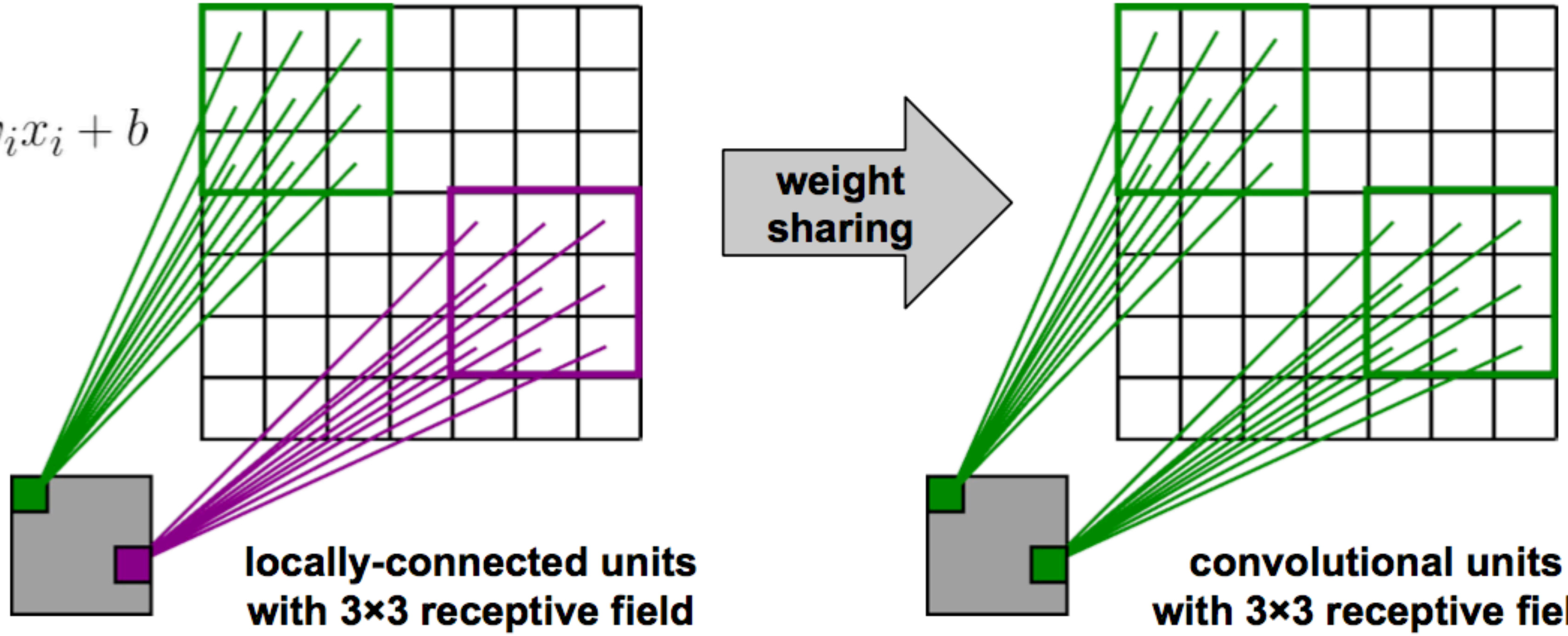
The bird occupies a local area and looks the same in different parts of an image.
We should construct neural nets which exploit these properties!

Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, Gradient-based learning applied to document recognition, Proceedings of the IEEE 86(11): 2278–2324, 1998.

CNN models Locality and Translation Invariance

Make fully-connected layer locally-connected and sharing weight

$$y = \sum_{i \in receptive\ field} w_i x_i + b$$

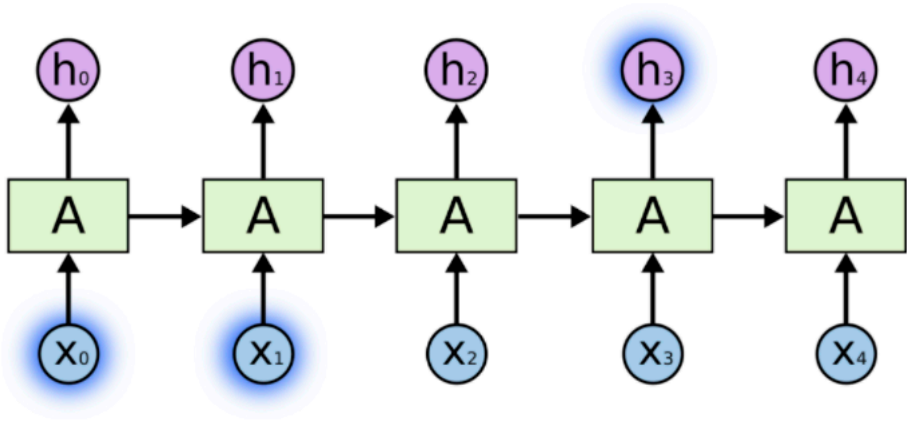


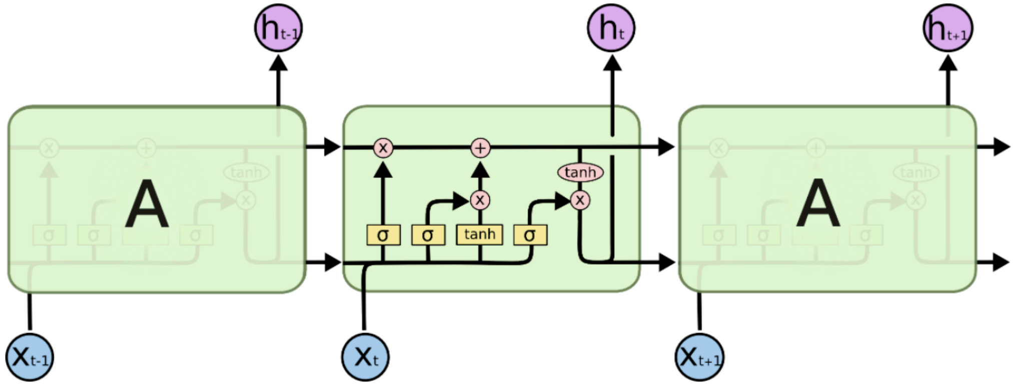
$$y = w * x + b$$

Adapt from From NIPS 2017 DL Trend Tutorial

Important Block: Recurrent Neural Networks (RNN)

- Prof. Schmidhuber invented "Long short-term memory" – Recurrent NN (LSTM-RNN) model in 1997





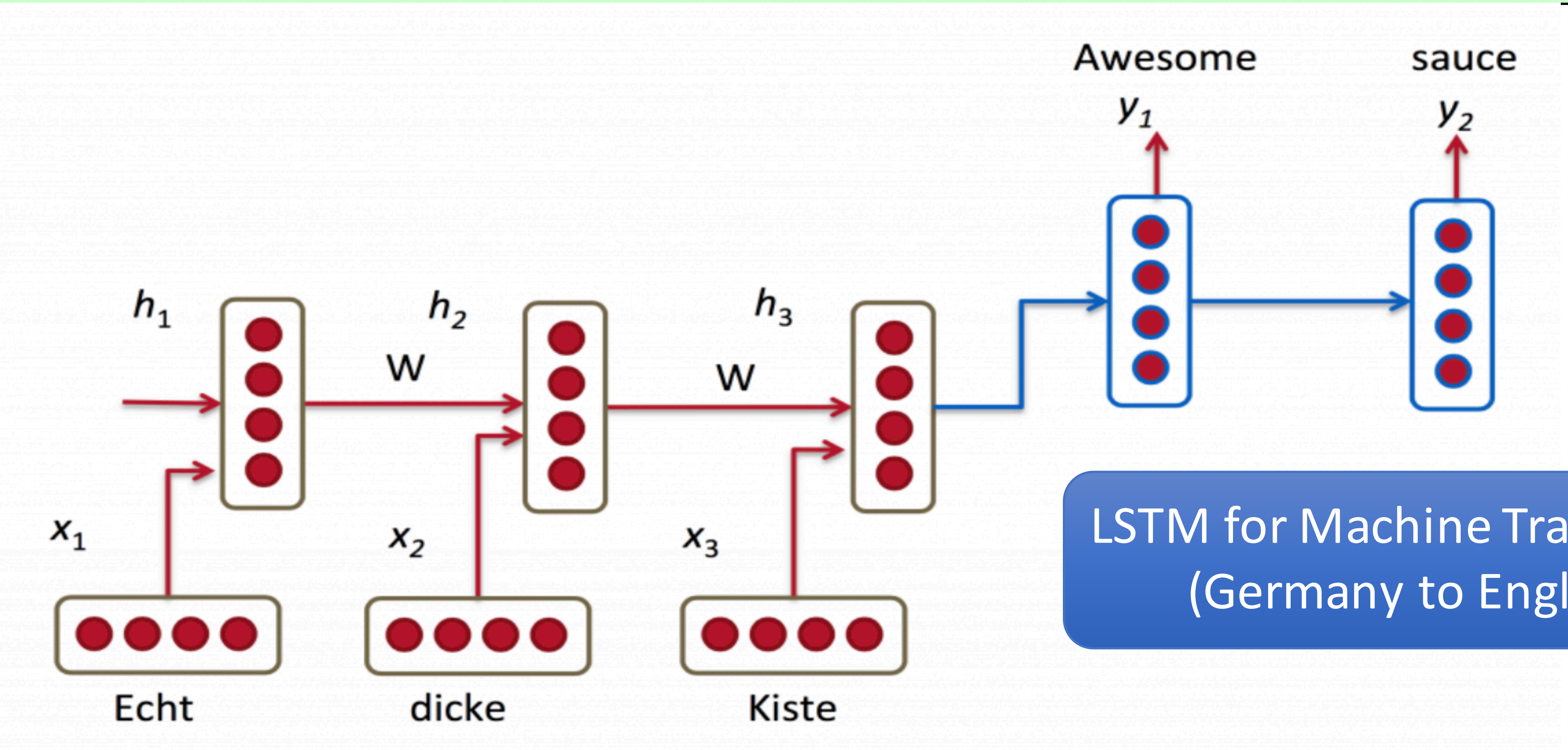
The repeating module in an LSTM contains four interacting layers.

Sepp Hochreiter; Jürgen Schmidhuber (1997). "Long short-term memory". Neural Computation. 9 (8): 1735–1780.

Image Credits from Christopher Olah

RNN models dynamic temporal dependency

- Make fully-connected layer model each unit recurrently
- Units form a directed chain graph along a sequence
- Each unit uses recent history and current input in modeling



LSTM for Machine Translation (Germany to English)

Image credit : wildML

- Deep Learning
  - Why is this a breakthrough ?
  - Basics
  - History
  - A Few Recent trends

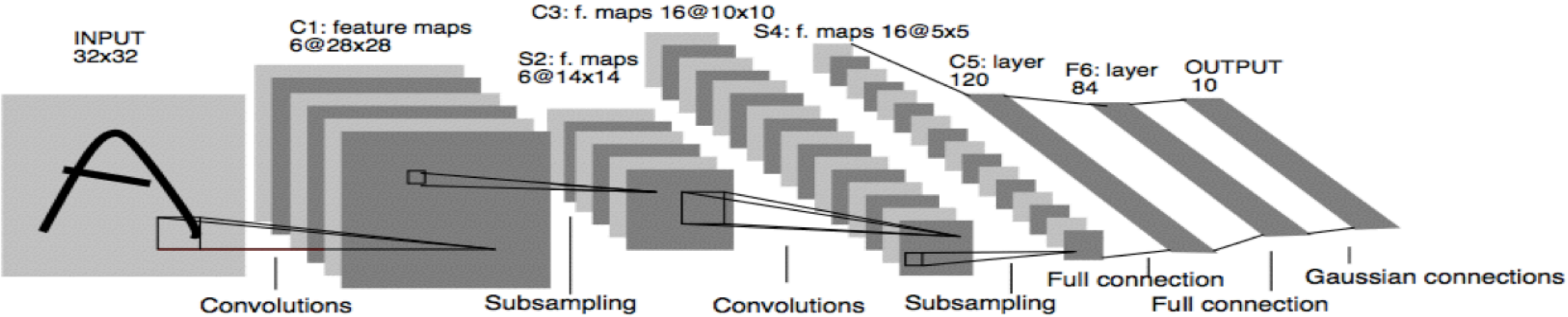
https://qdata.github.io/deep2Read/

Many classification models invented since late 80's

- Neural networks
- Boosting
- Support Vector Machine
- Maximum Entropy
- Random Forest
- ……

Deep Learning (CNN) in the 90's

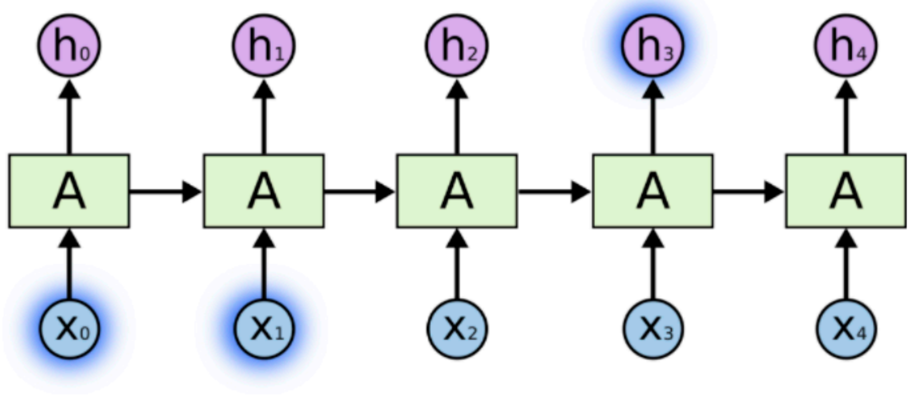
- Prof. Yann LeCun invented Convolutional Neural Networks (CNN) in 1998
- First NN successfully trained with many layers

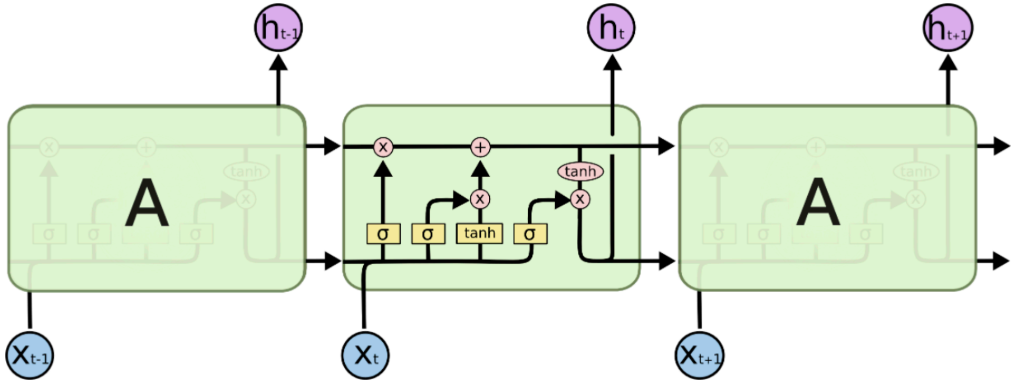


Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, Gradient-based learning applied to document recognition, Proceedings of the IEEE 86(11): 2278–2324, 1998.

Deep Learning (RNN) in the 90's

- Prof. Schmidhuber invented "Long short-term memory" – Recurrent NN (LSTM-RNN) model in 1997





The repeating module in an LSTM contains four interacting layers.

Sepp Hochreiter; Jürgen Schmidhuber (1997). "Long short-term memory". Neural Computation. 9 (8): 1735–1780.

Image Credits from Christopher Olah

Between ~2000 to ~2011
Machine Learning Field Interest

- Learning with Structures ! + Convex Formulation!
  - Kernel learning
  - Manifold Learning
  - Sparse Learning
  - Structured input-output learning …
  - Graphical model
  - Transfer Learning
  - Semi-supervised
  - Matrix factorization
  - ……

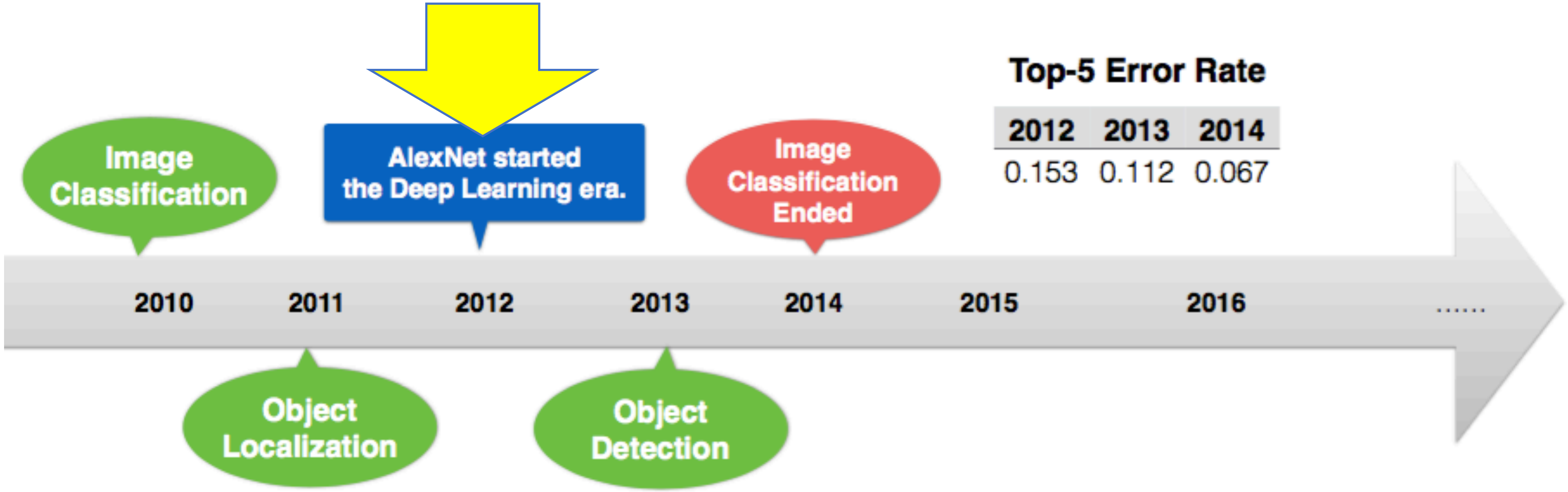
“Winter of Neural Networks” Since 90’s to ~2011

- Non-convex

- Need a lot of tricks to play with
  - How many layers ?
  - How many hidden units per layer ?
  - What topology among layers ? …….

- Hard to perform theoretical analysis

Breakthrough in 2012 Large-Scale Visual Recognition Challenge (ImageNet) : Milestones in Recent Vision/AI Fields



Top-5 Error Rate

2012	2013	2014
0.153	0.112	0.067

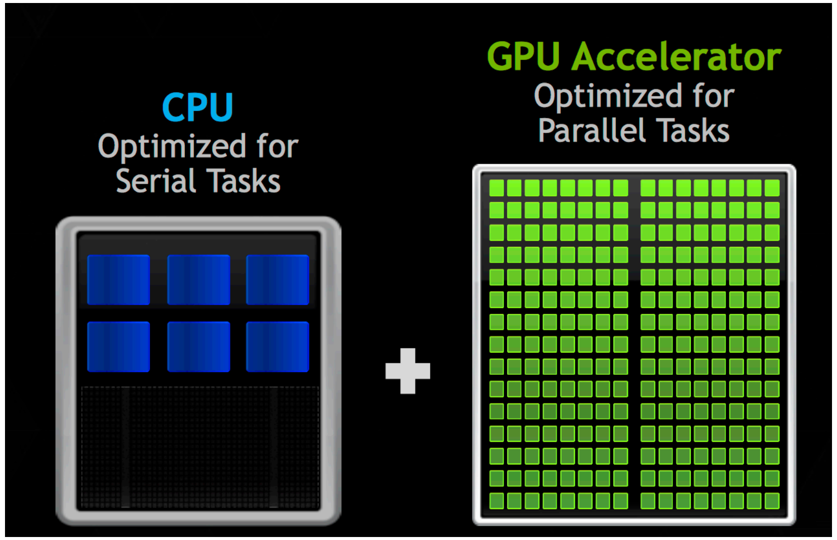
- 2013, Google Acquired Deep Neural Networks Company headed by Utoronto "Deep Learning" Professor Hinton
- 2013, Facebook Built New Artificial Intelligence Lab headed by NYU "Deep Learning" Professor LeCun
- 2016, Google's DeepMind defeats legendary Go player Lee Se-dol in historic victory / 2017 Alpha Zero

Reason: Plenty of (Labeled) Data

- Text: trillions of words of English + other languages
- Visual: billions of images and videos
- Audio: thousands of hours of speech per day
- User activity: queries, user page clicks, map requests, etc,
- Knowledge graph: billions of labeled relational triplets
- ....

Dr. Jeff Dean's talk

Reason: Advanced Computer Architecture that fits DNNs



http://www.nvidia.com/content/events/geoInt2015/LBrown_DL.pdf

	Neural Networks	GPUs
Inherently Parallel	✓	✓
Matrix Operations	✓	✓
FLOPS	✓	✓

GPUs deliver --

- *same or better prediction accuracy*
- *faster results*
- *smaller footprint*
- *lower power*

Some Recent Trends

https://qdata.github.io/deep2Read/

- 1. Autoencoder / layer-wise training
- 2. CNN / Residual / Dynamic parameter
- 3. RNN / Attention / Seq2Seq, …
- 4. Neural Architecture with explicit Memory
- 5. NTM 4program induction / sequential decisions
- 6. Learning to optimize / Learning DNN architectures
- 7. Learning to learn / meta-learning/ few-shots
- 8. DNN on graphs / trees / sets
- 9. Deep Generative models, e.g., autoregressive
- 10. Generative Adversarial Networks (GAN)
- 11. Deep reinforcement learning
- 12. Validate / Evade / Test / Understand / Verify DNNs

Recap

https://qdata.github.io/deep2Read/

Inputs and Outputs

Losses

Architectures:

Learned Models

Adapt from From NIPS 2017 DL Trend Tutorial

Making Deep Learning Understandable for Analyzing Sequential Data about Gene Regulation

Dr. Yanjun Qi
Department of Computer Science
University of Virginia

Tutorial @ ACM BCB-2018

BREAK 5mins ->Second Half