Computer Vision Applications

COMP 388-002/488-002 Computer Science Topics

Daniel Moreira

Fall 2022

LOYOLA UNIVERSITY CHICAGO

Letter Soup

COMP 388-002/488-002 Computer Science Topics
Computer Vision Applications

Daniel Moreira
Fall 2022

LOYOLA UNIVERSITY CHICAGO

Today you will…

Get an overview of CV, AI, ML, PR, SVM, CNN, DL, GPU, PCA, etc., all in favor of the upcoming seminars.

LOYOLA UNIVERSITY CHICAGO

Course Overview

RECAP

Content

Date	Topic	Leader	Assignment
08/29	Introduction to CV	Instructor	N.A.
09/05	*Labor Day*	N.A.	A01, due on 09/13
09/12	Letter Soup: AI, ML, NN, and DL	Instructor	A02, due on 09/20
09/19	Local and Global Descriptors	Instructor	A03, due on 09/27
09/26	CBIR and Indexing	TBD *(students)*	A04, due on 10/04
10/03	Image Classification	TBD *(students)*	A05, due on 10/18
10/10	*Fall Break*	N.A.	N.A.
10/17	Object Detection	TBD *(students)*	A06, due on 10/25
10/24	Image Segmentation	TBD *(students)*	A07, due on 11/01
10/31	Face Detection	TBD *(students)*	A08, due on 11/08
11/07	Face Verification	TBD *(students)*	A09, due on 11/15
11/14	GANs and Generative DL	TBD *(students)*	A10, due on 11/29
11/21	Deep and Cheap Fakes	Instructor	N.A.
11/28	Sensitive Media Analysis	Instructor	N.A.
12/05	Provenance Analysis	TBD *(students)*	N.A.
12/12	*Final Exam*	N.A.	N.A.

LOYOLA UNIVERSITY CHICAGO

Course Overview

RECAP

Content

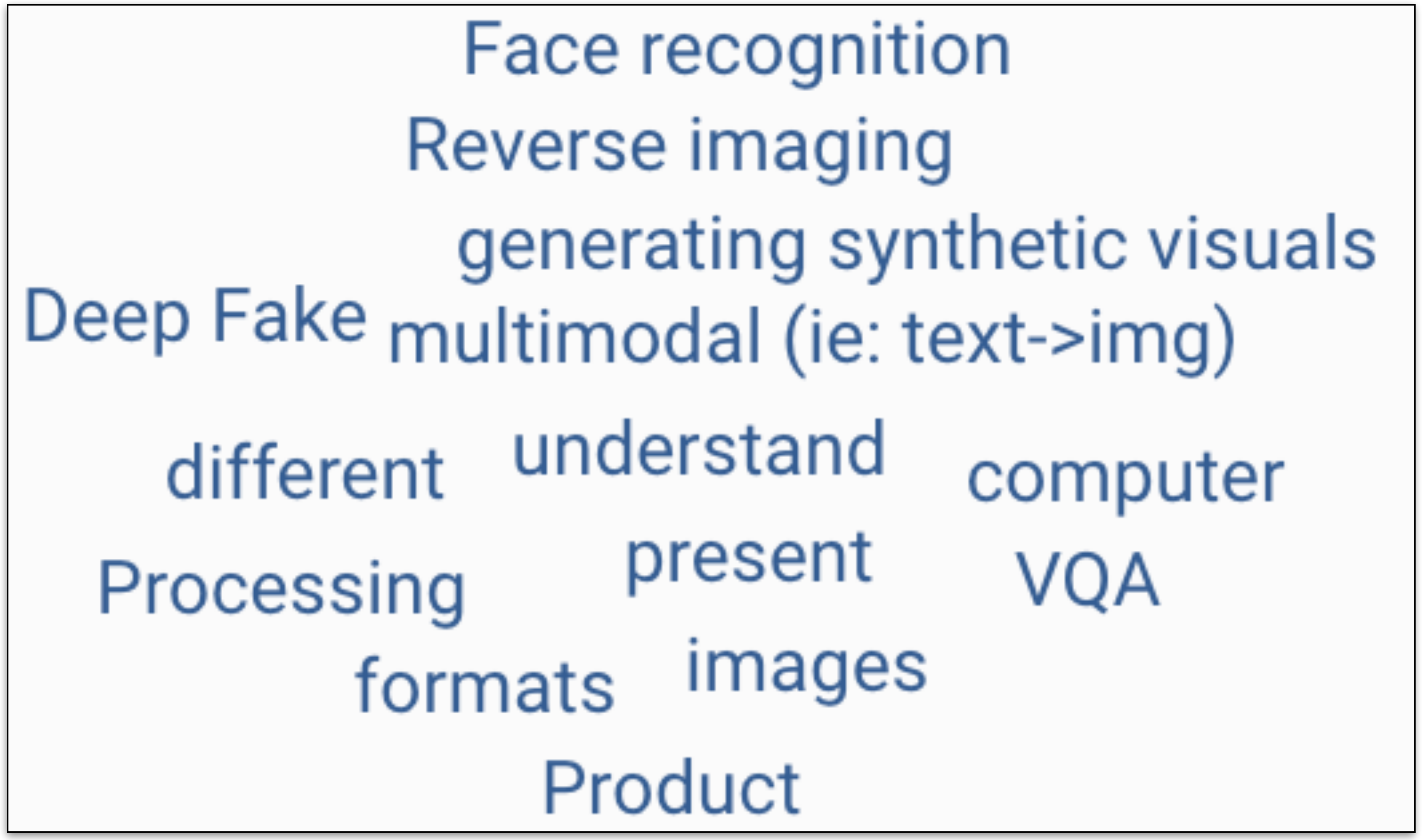
Date	Topic	Leader	Assignment
08/29	Introduction to CV	Instructor	N.A.
09/05	*Labor Day*	N.A.	A01, due on 09/13
09/12	Letter Soup: AI, ML, NN, and DL	Instructor	A02, due on 09/20
09/19	Local and Global Descriptors	Instructor	A03, due on 09/27
09/26	CBIR and Indexing	TBD *(students)*	A04, due on 10/04
10/03	Image Classification	TBD *(students)*	A05, due on 10/18
10/10	*Fall Break*	N.A.	N.A.
10/17	Object Detection	TBD *(students)*	A06, due on 10/25
10/24	Image Segmentation	TBD *(students)*	A07, due on 11/01
10/31	Face Detection	TBD *(students)*	A08, due on 11/08
11/07	Face Verification	TBD *(students)*	A09, due on 11/15
11/14	GANs and Generative DL	TBD *(students)*	A10, due on 11/29
11/21	Deep and Cheap Fakes	Instructor	N.A.
11/28	Sensitive Media Analysis	Instructor	N.A.
12/05	Provenance Analysis	TBD *(students)*	N.A.
12/12	*Final Exam*	N.A.	N.A.

Please answer the form at:
https://forms.gle/wsNjWG3MDiZPsEzA9

LOYOLA UNIVERSITY CHICAGO

Computer Vision (CV)

RECAP



LOYOLA UNIVERSITY CHICAGO

Computer Vision (CV)

RECAP

Computer Science Subfield
It aims at developing computer systems that mimic the human visual system.

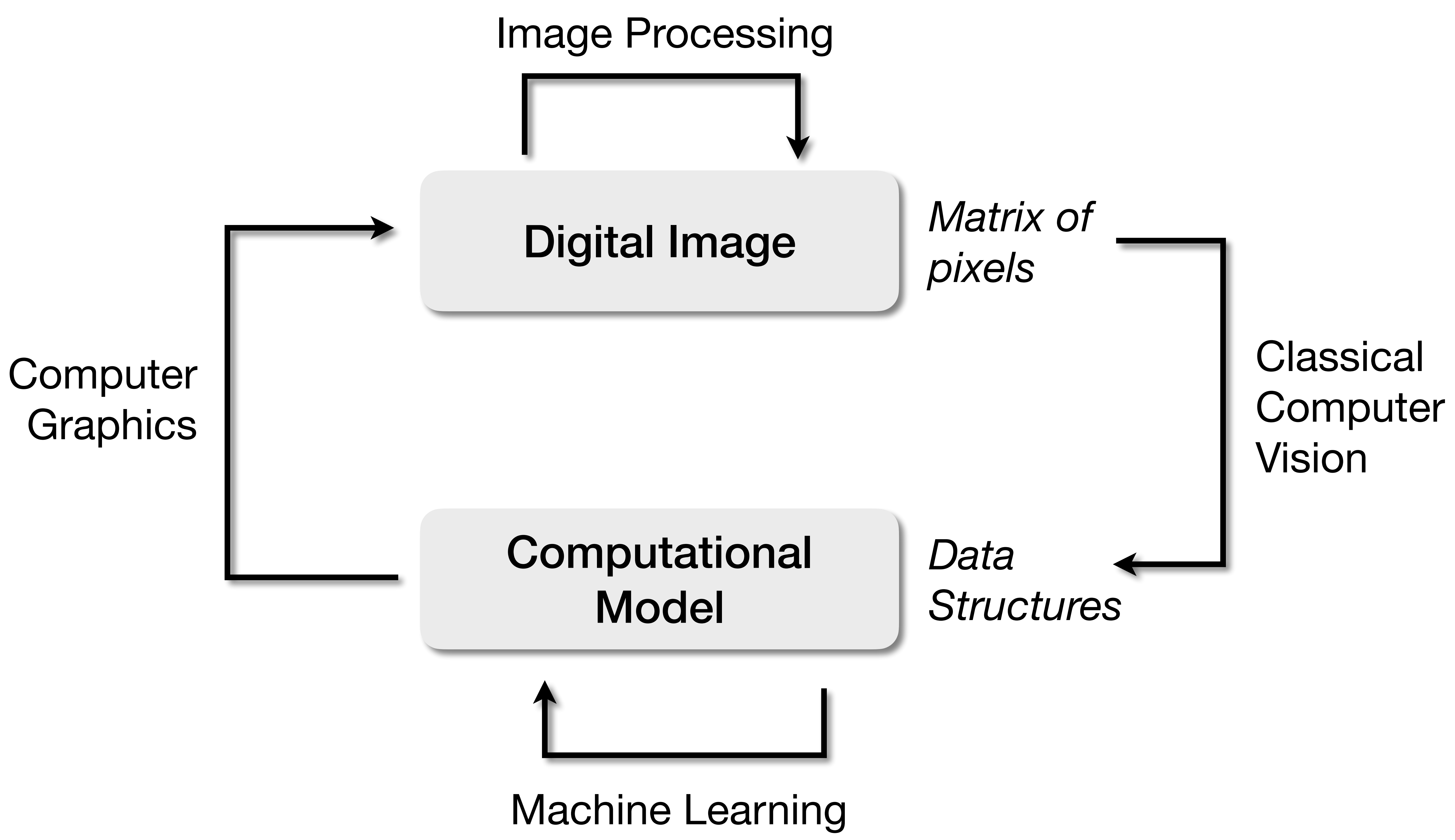
Reference

Objective

LOYOLA UNIVERSITY CHICAGO

Computer Vision (CV)

RECAP

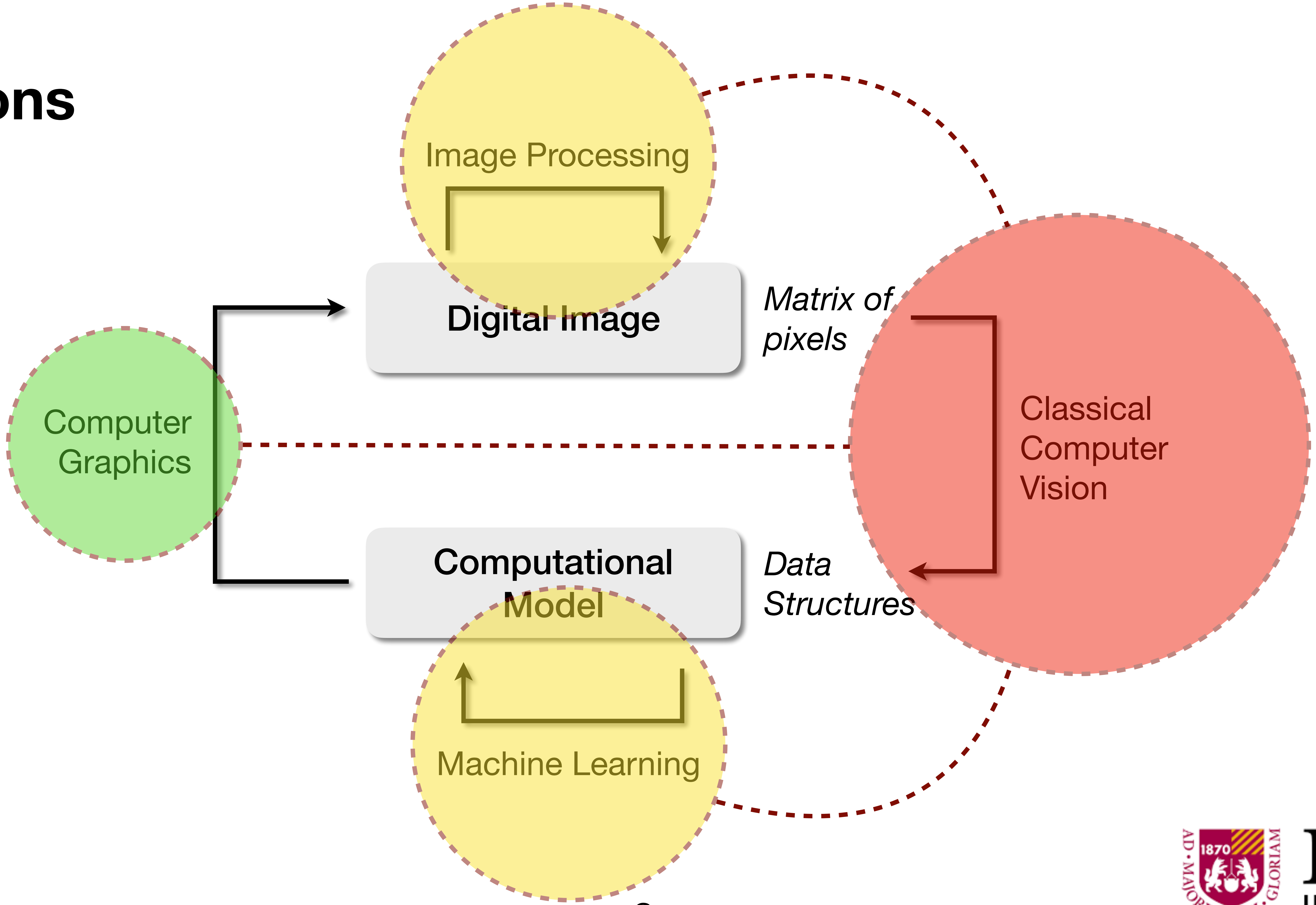


LOYOLA UNIVERSITY CHICAGO

Computer Vision (CV)

RECAP

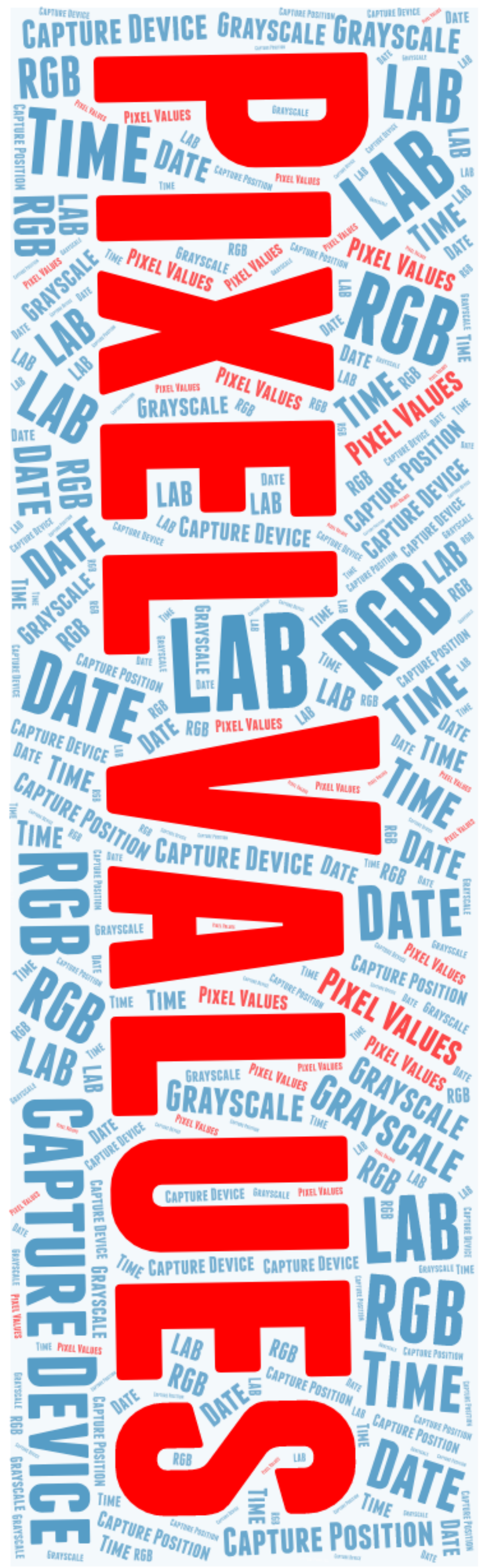
Applications



LOYOLA UNIVERSITY CHICAGO

Computer Vision (CV)

RECAP



Level 0

?

Semantic Gap

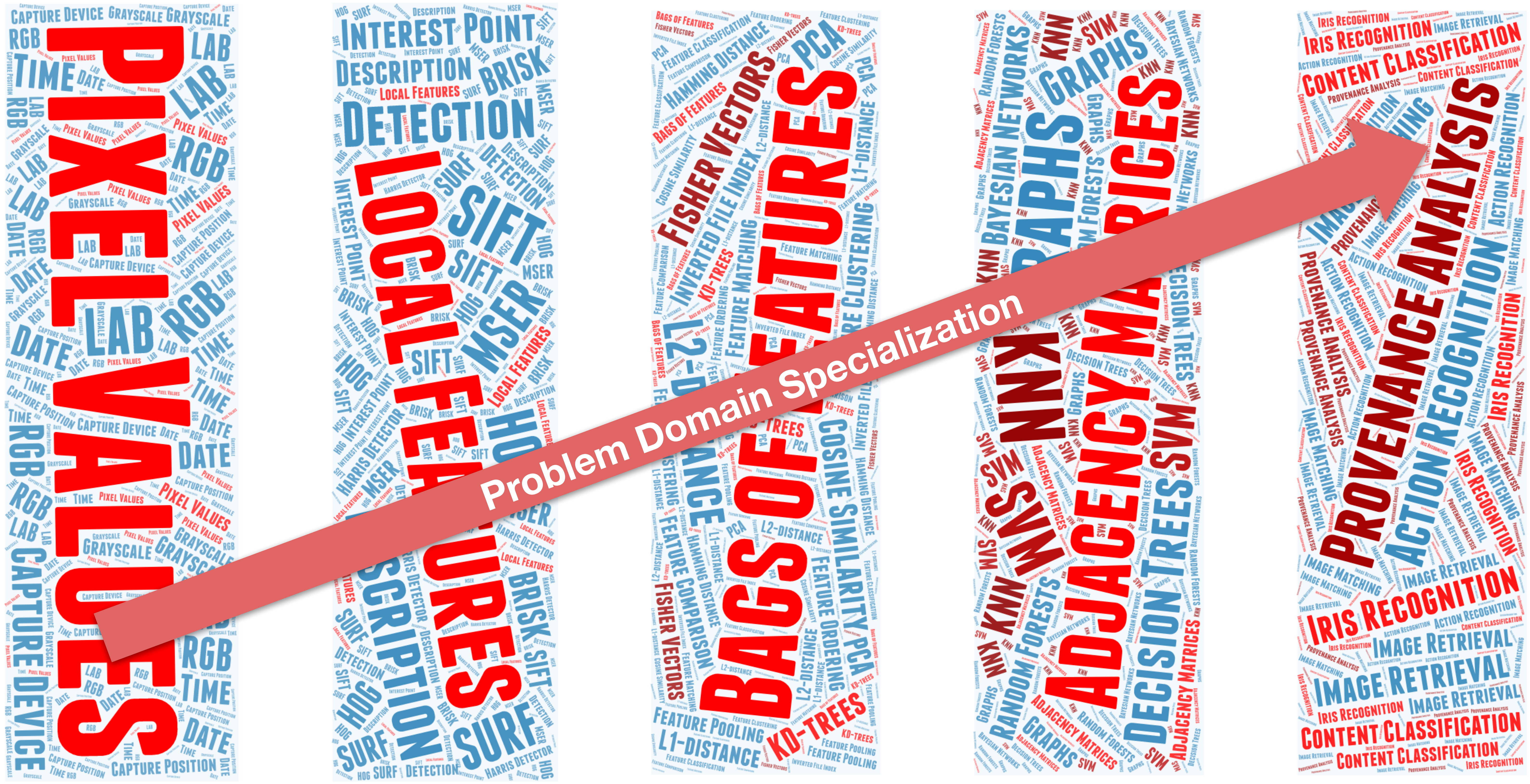
Task

LOYOLA UNIVERSITY CHICAGO

Computer Vision (CV)

RECAP

Then



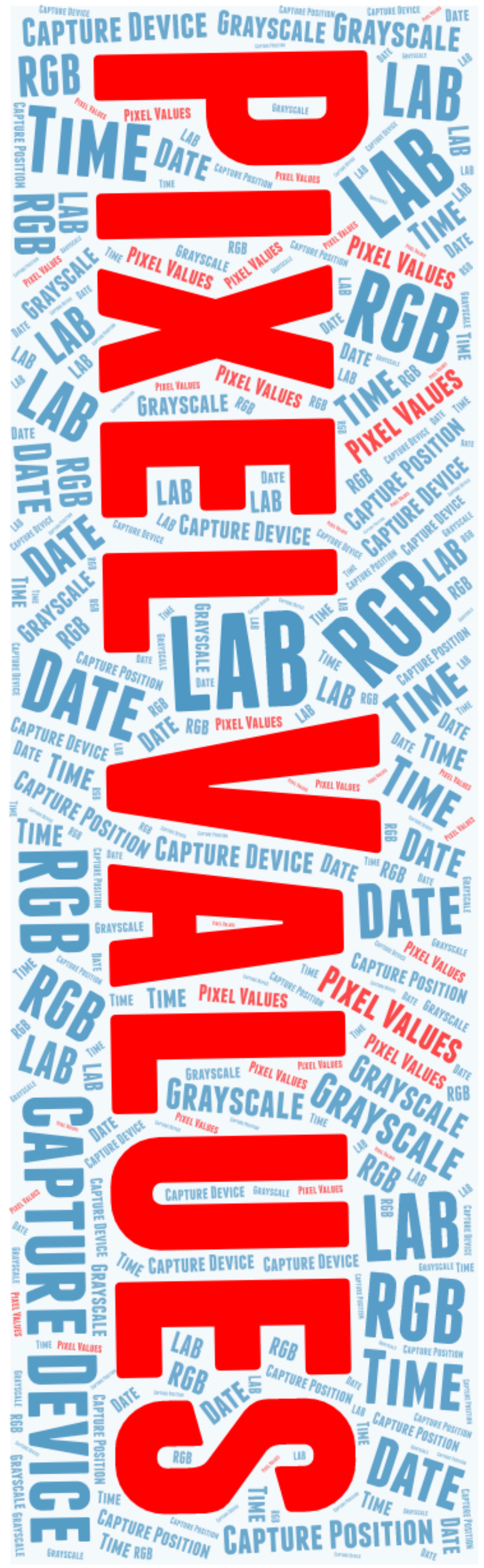
Level 0 | Level 1 | Level 2 | Level 3 | Task

LOYOLA UNIVERSITY CHICAGO

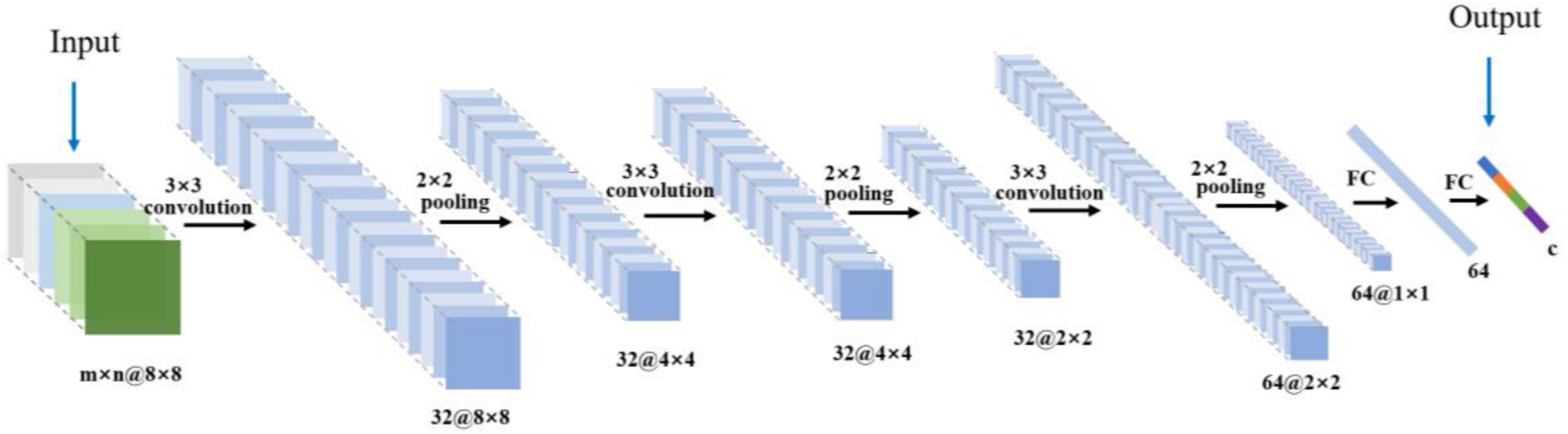
Computer Vision (CV)

RECAP

Now



Level 0



Deep Learning

Machine Learning?

Artificial Intelligence?

Task

LOYOLA UNIVERSITY CHICAGO

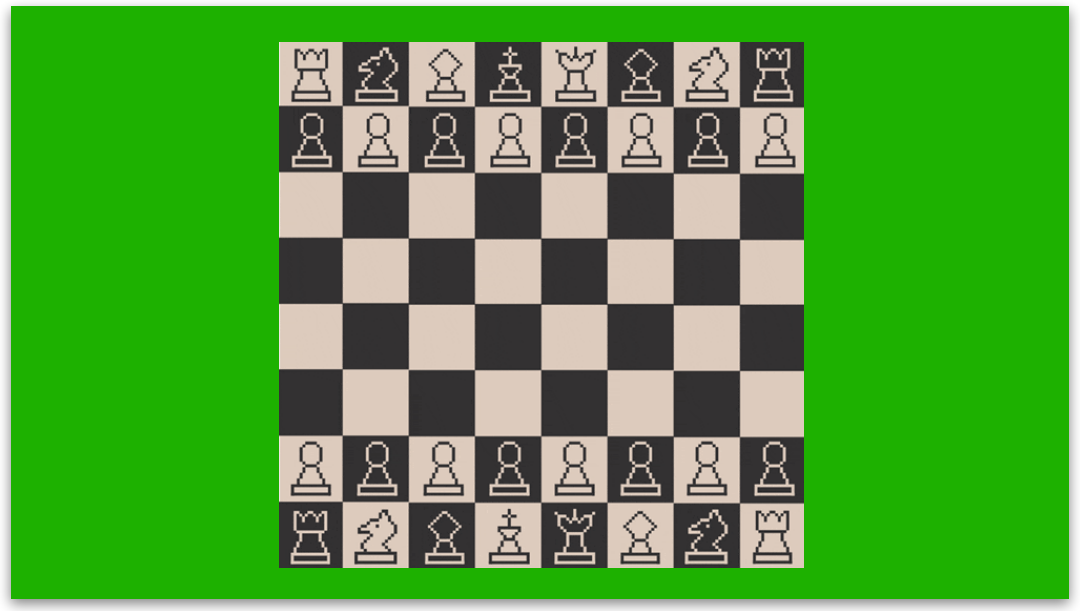
Artificial Intelligence (AI)

What comes to your mind?

https://bit.ly/3QFaq5G

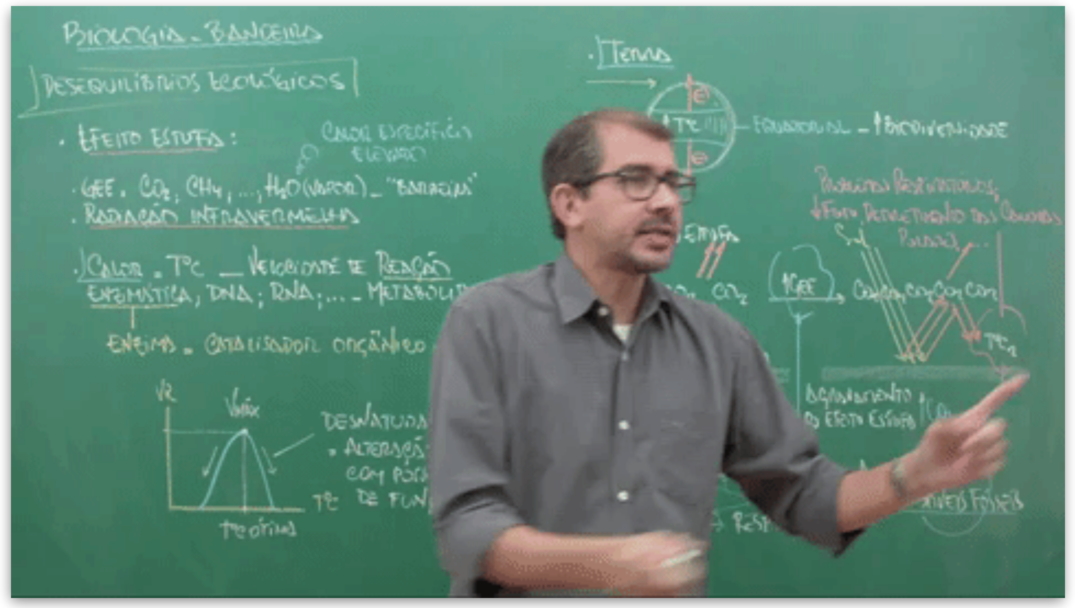
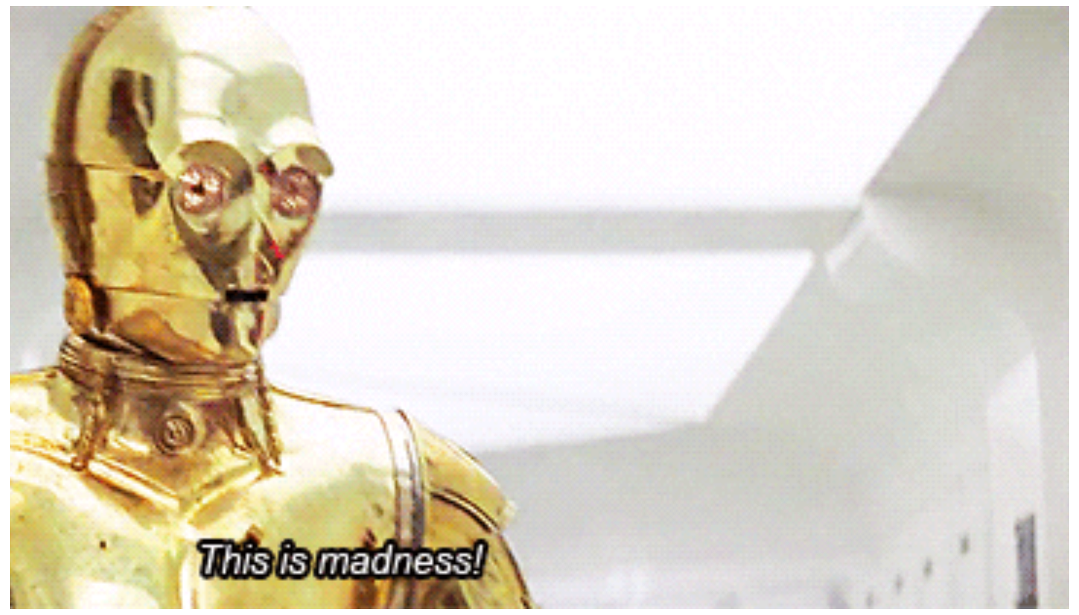
Artificial Intelligence (AI)

Computer Science PoV
It aims at developing computer systems that mimic (or overcome) humans' intelligence (or other living entities').



Humans AI



Humans AI

Artificial Intelligence (AI)

Humans (ref.)

Perceiving

Acting

Memorizing

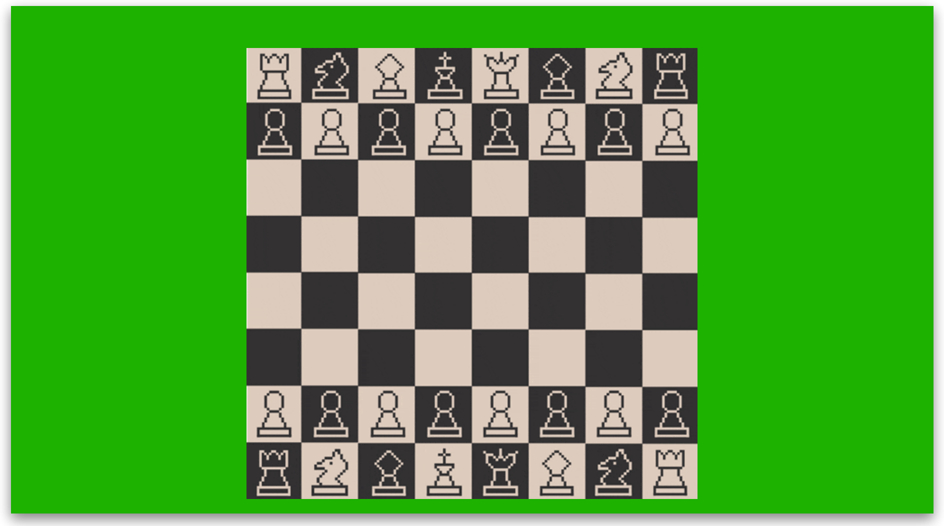
Reasoning

Learning

Inventing

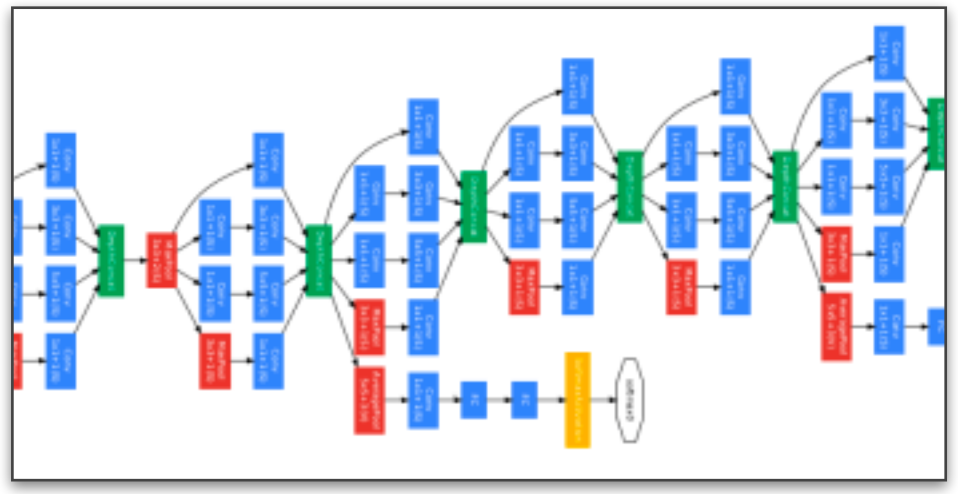
Artificial Intelligence (AI)

Specialized Systems



Perceiving
Acting
Memorizing
Reasoning
Learning
Inventing

Perceiving
Acting
Memorizing
Reasoning
Learning
Inventing



Perceiving
Acting
Memorizing
Reasoning
Learning
Inventing

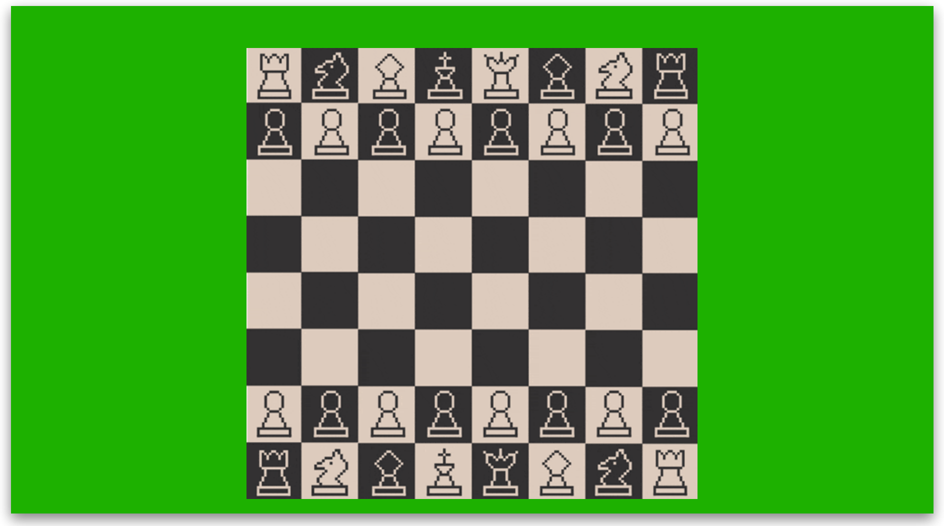
Humans (ref.)

Perceiving
Acting
Memorizing
Reasoning
Learning
Inventing

LOYOLA UNIVERSITY CHICAGO

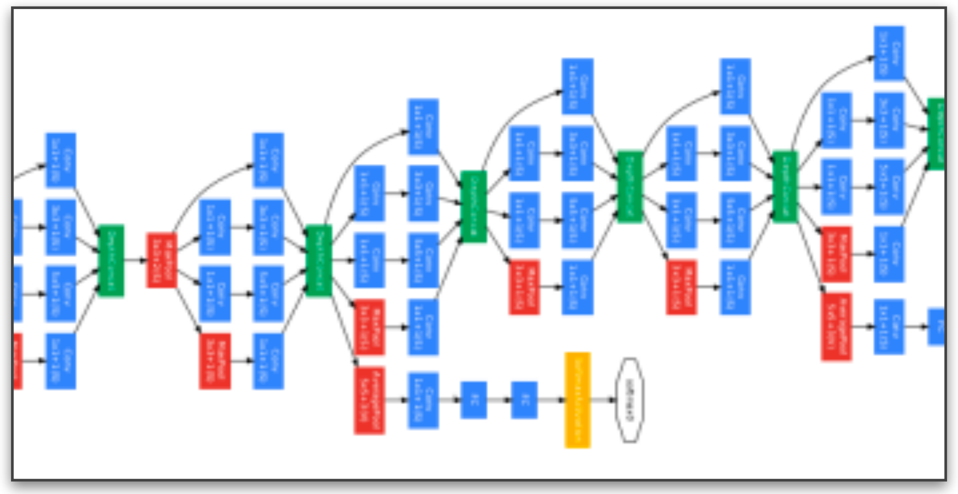
Artificial Intelligence (AI)

Specialized Systems



Perceiving
Acting
Memorizing
Reasoning
Learning
Inventing

Perceiving
Acting
Memorizing
Reasoning
Learning
Inventing



Perceiving
Acting
Memorizing
Reasoning
Learning
Inventing

General-purpose Systems

Perceiving
Acting
Memorizing
Reasoning
Learning
Inventing

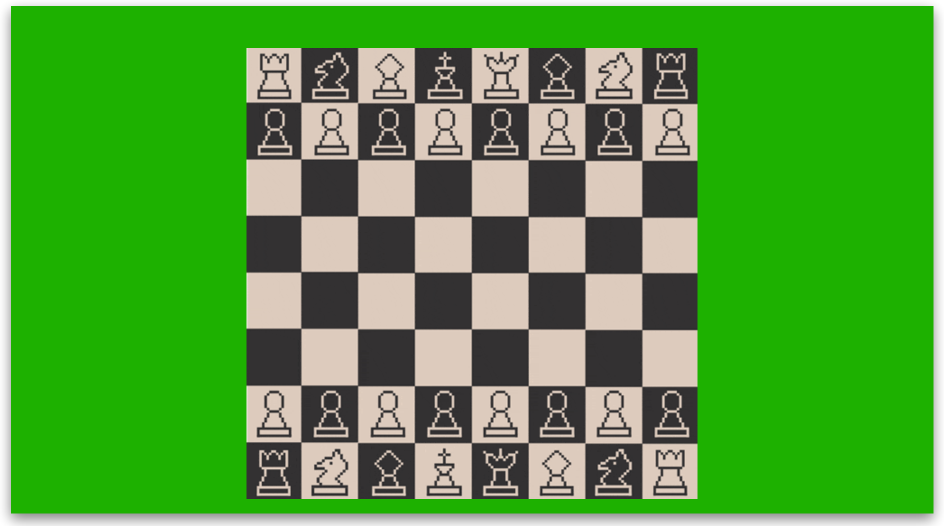
Humans (ref.)

Perceiving
Acting
Memorizing
Reasoning
Learning
Inventing

LOYOLA UNIVERSITY CHICAGO

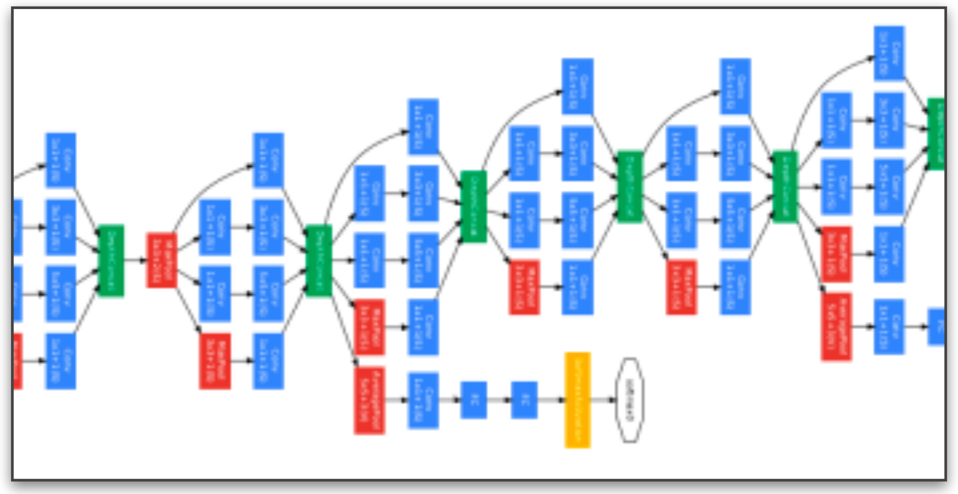
Artificial Intelligence (AI)

Weak AI



Perceiving
Acting
Memorizing
Reasoning
Learning
Inventing

Perceiving
Acting
Memorizing
Reasoning
Learning
Inventing



Perceiving
Acting
Memorizing
Reasoning
Learning
Inventing

Strong AI

Perceiving
Acting
Memorizing
Reasoning
Learning
Inventing

Humans (ref.)

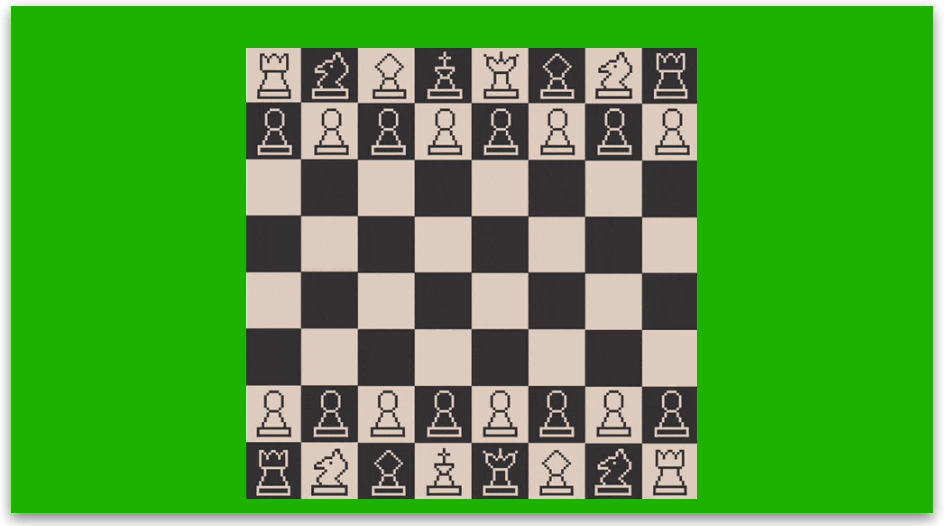
Perceiving
Acting
Memorizing
Reasoning
Learning
Inventing

LOYOLA UNIVERSITY CHICAGO

Artificial Intelligence (AI)

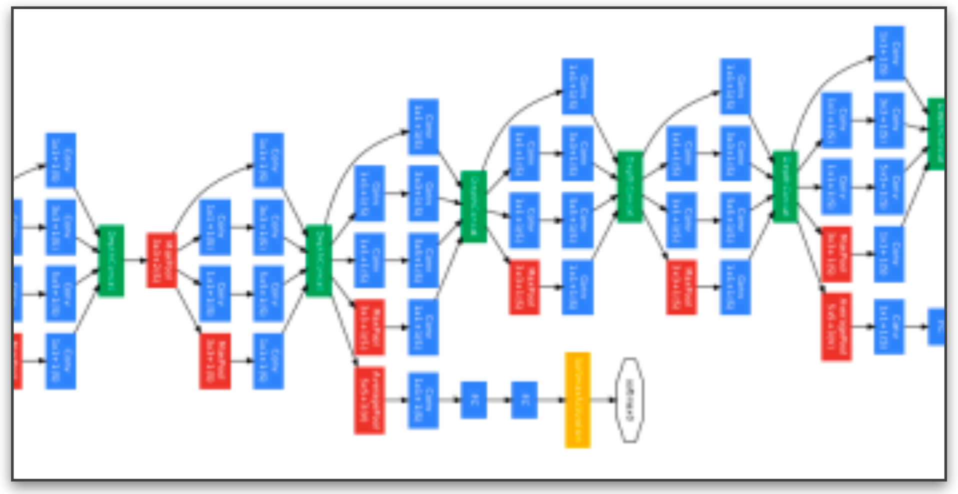
Weak AI

We are here!



Perceiving
Acting
Memorizing
Reasoning
Learning
Inventing

Perceiving
Acting
Memorizing
Reasoning
Learning
Inventing



Perceiving
Acting
Memorizing
Reasoning
Learning
Inventing

Strong AI

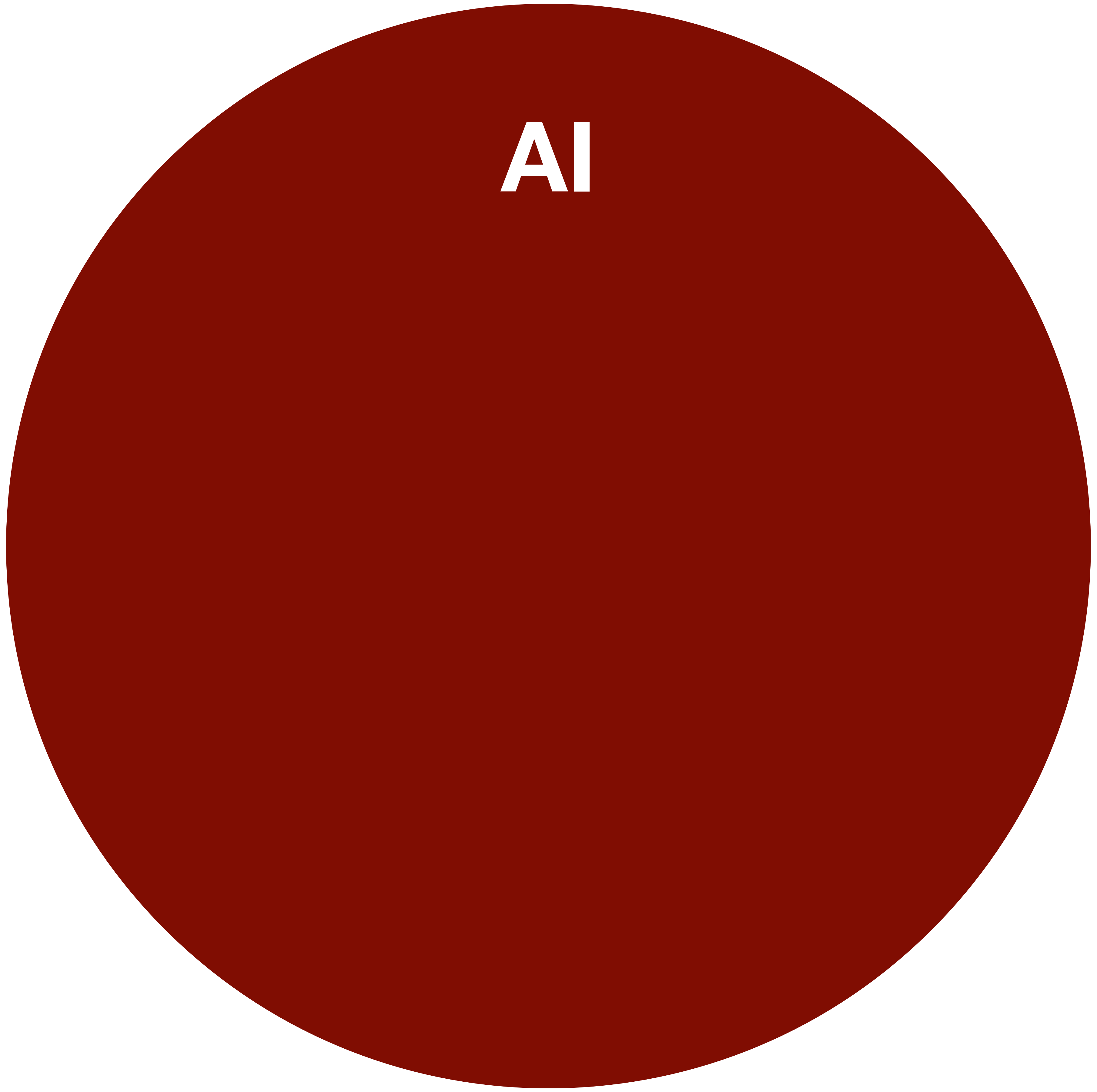
Perceiving
Acting
Memorizing
Reasoning
Learning
Inventing

Humans (ref.)

Perceiving
Acting
Memorizing
Reasoning
Learning
Inventing

LOYOLA UNIVERSITY CHICAGO

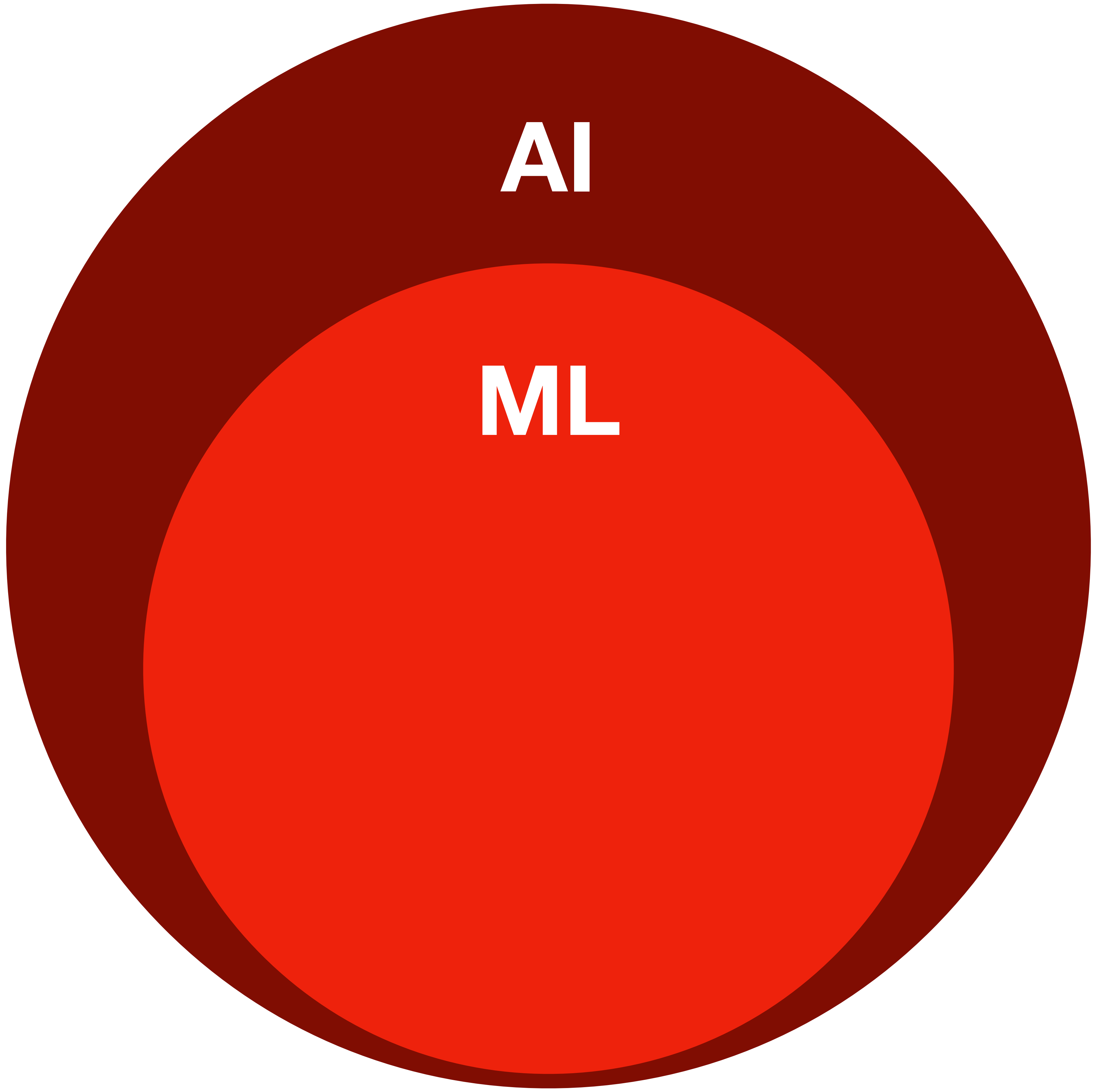
Artificial Intelligence (AI)



- Symbolic Logic
- Fuzzy Logic
- Semantic Networks
- Evolutionary Models
- Swarm Intelligence
- Machine Learning

LOYOLA UNIVERSITY CHICAGO

Artificial Intelligence (AI)



Symbolic Logic

Fuzzy Logic

Semantic Networks

Evolutionary Models

Swarm Intelligence

Machine Learning

We are here!

LOYOLA UNIVERSITY CHICAGO

Machine Learning (ML)

What comes to your mind?

https://bit.ly/3xbkegQ

LOYOLA
UNIVERSITY CHICAGO

Machine Learning (ML)

It is AI

It aims at developing computer systems that mimic humans' intelligence.

More Specifically

It aims at developing computer systems that mimic the learning-ability of humans.

In Practice

Leverage data examples to improve the performance on a target task.

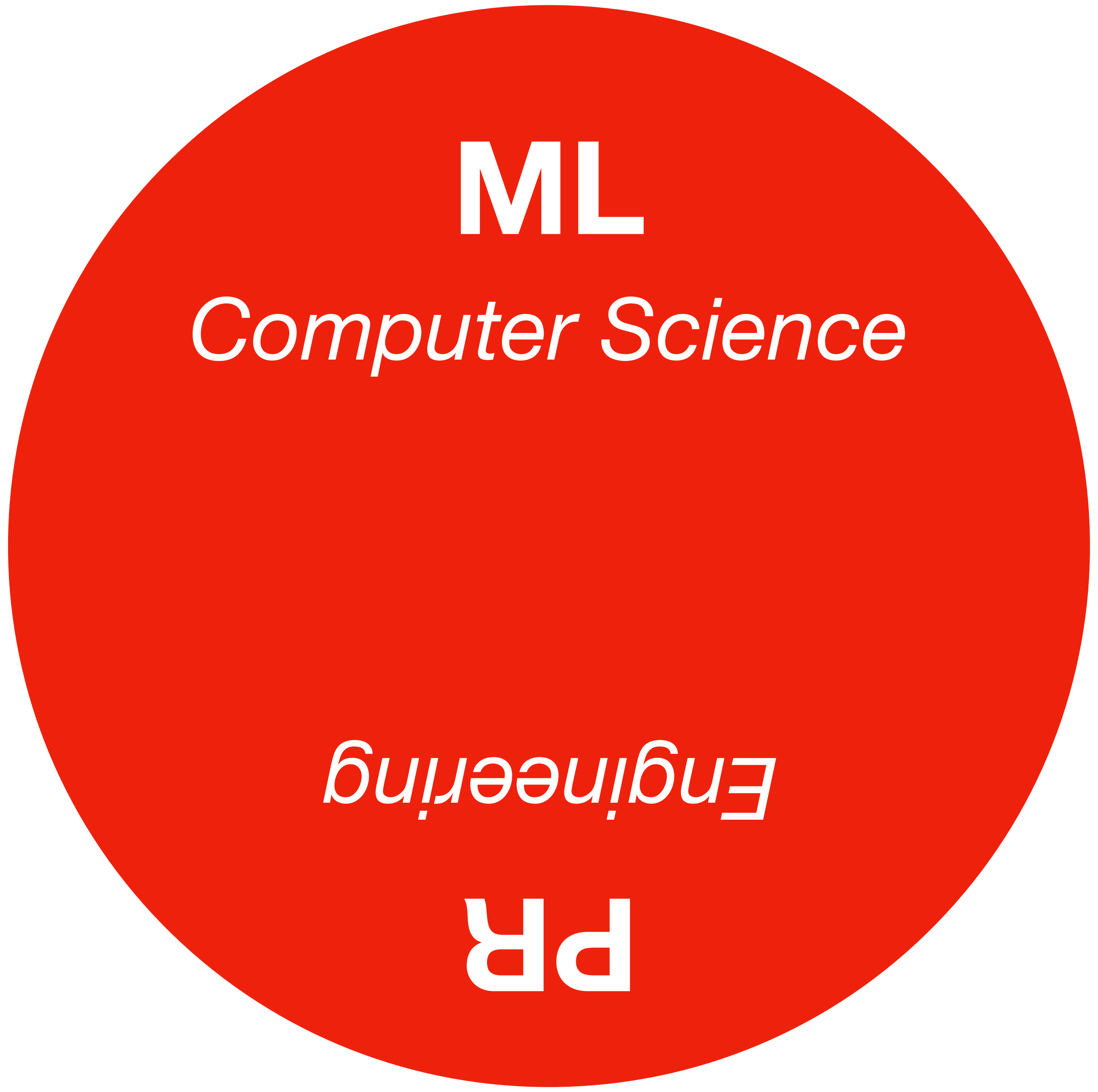
It is data-driven.

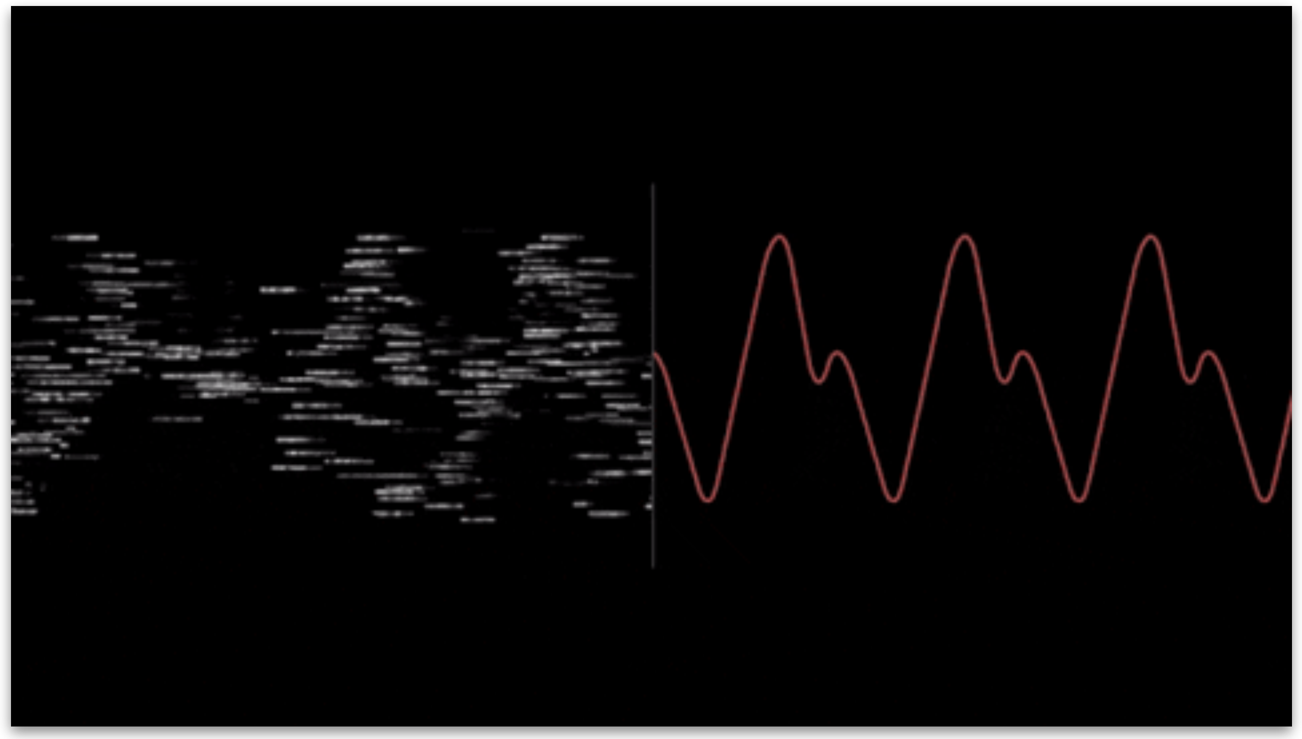
Machine Learning (ML)

ML and Pattern Recognition (PR)

Same field.

Cluster or label data.





Find patterns on data.

LOYOLA UNIVERSITY CHICAGO

Machine Learning (ML)

What data (structure) are we typically talking about?

Example problem: fish classification

Sea Bass or
Salmon?

LOYOLA
UNIVERSITY CHICAGO

Machine Learning (ML)

What data (structure) are we typically talking about?

Example problem: fish classification

Sea Bass or Salmon?

Fish
- length: float - weight: float - width: float - fin_count: int

Length	Weight	Width	Fin
1.12	14.11	0.31	3.0
0.95	11.22	0.28	3.0
1.08	12.02	0.31	3.0
1.45	45.03	0.37	2.0
1.09	31.01	0.38	2.0
1.08	11.09	0.29	2.0
1.51	46.00	0.37	2.0

LOYOLA UNIVERSITY CHICAGO

Machine Learning (ML)

What data (structure) are we typically talking about?

Example problem: fish classification

Sea Bass or Salmon?

Fish
- length: float - weight: float - width: float - fin_count: int

Length	Weight	Width	Fin
1.12	14.11	0.31	3.0
0.95	11.22	0.28	3.0
1.08	12.02	0.31	3.0
1.45	45.03	0.37	2.0
1.09	31.01	0.38	2.0
1.08	11.09	0.29	2.0
1.51	46.00	0.37	2.0

Each data sample is a **Tensor**.

1.12	14.11	0.31	3.0

1.08	12.02	0.31	3.0

1.08	11.09	0.29	2.0

LOYOLA UNIVERSITY CHICAGO

Machine Learning (ML)

What data (structure) are we typically talking about?

0-order: **Scalar**

1st-order tensor: **Vector**
(feature vectors)

2nd-order tensor: **Matrix**
(images)

Each data sample is a **Tensor**.

Scalar

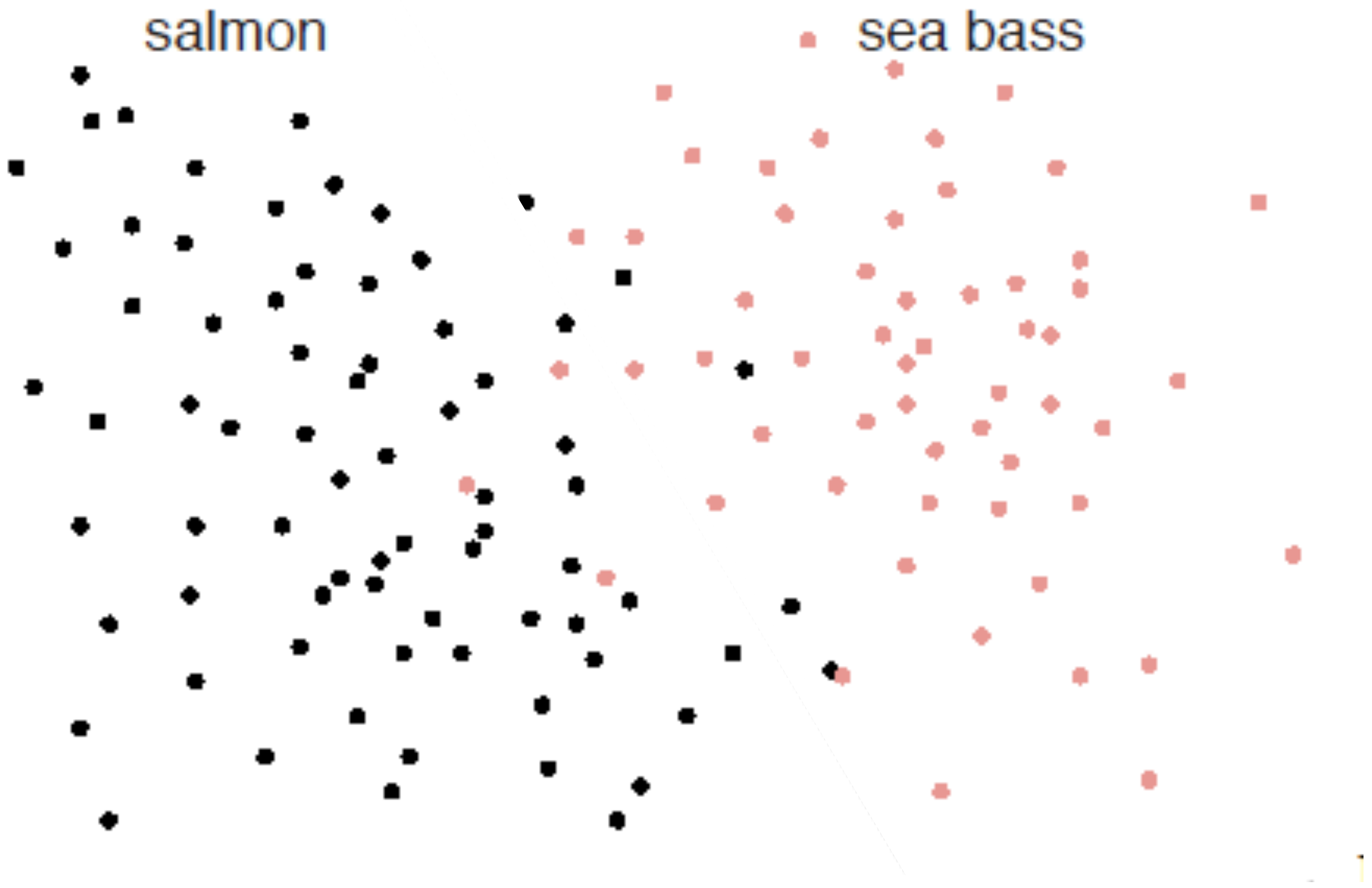
1.12	14.11	0.31	3.0

Feature Vector

Machine Learning (ML)

What data (structure) are we typically talking about?

$\Re^n$ space



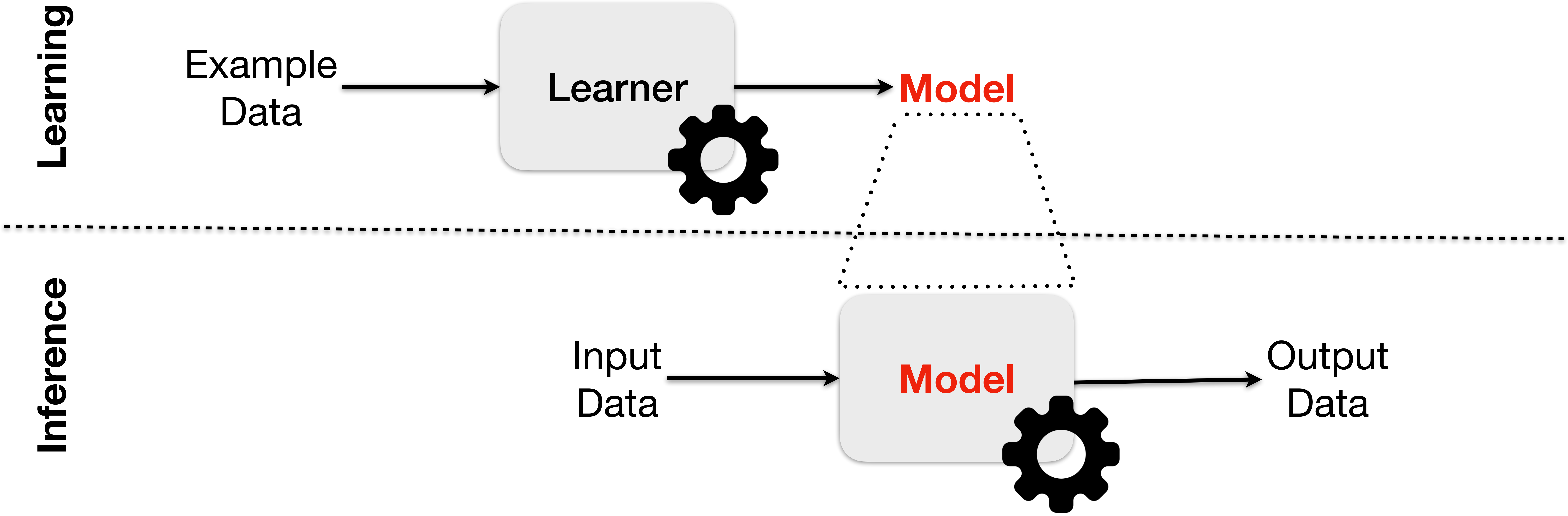
Each data sample is a **Tensor**.

1.12	14.11	0.31	3.0

Feature Vector

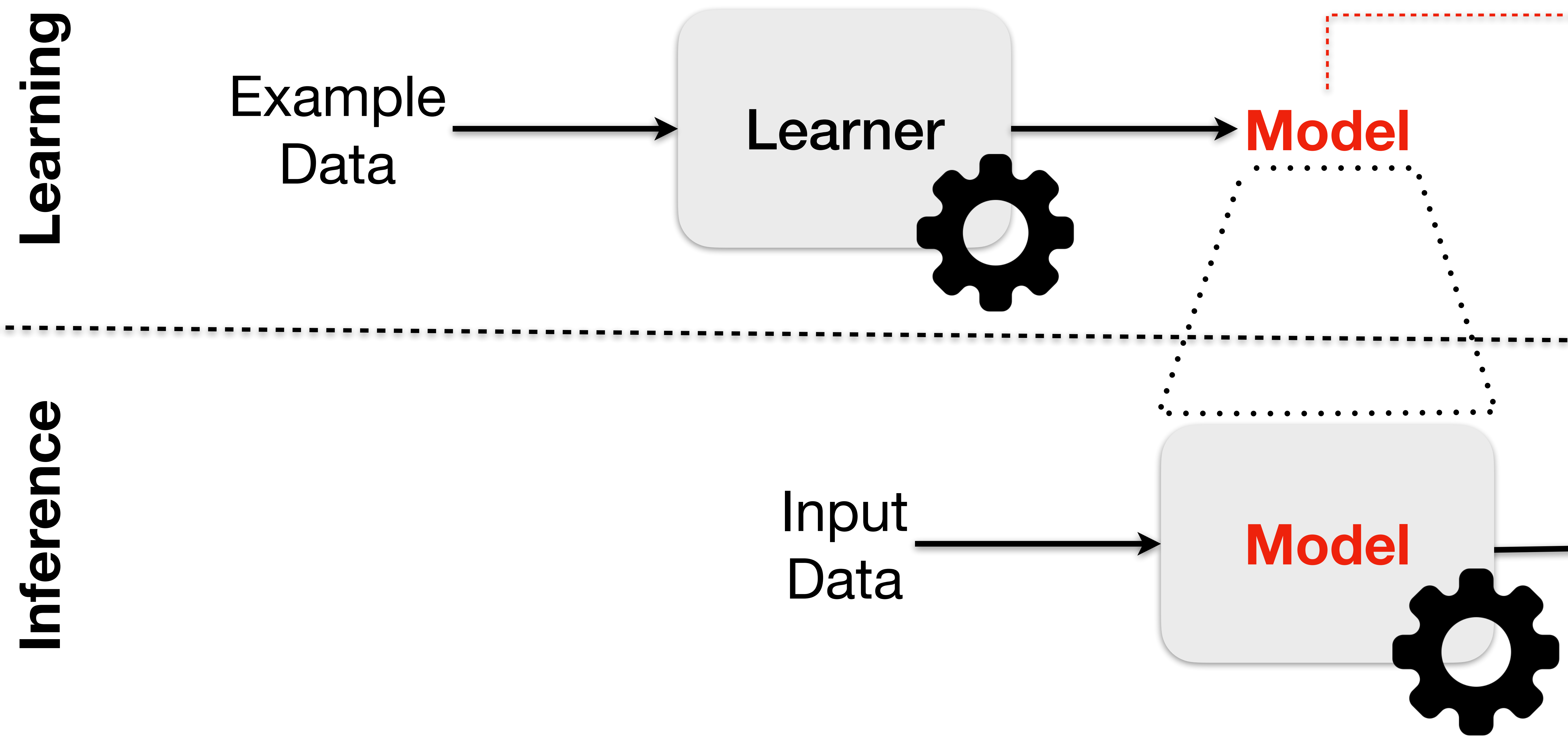
Machine Learning (ML)

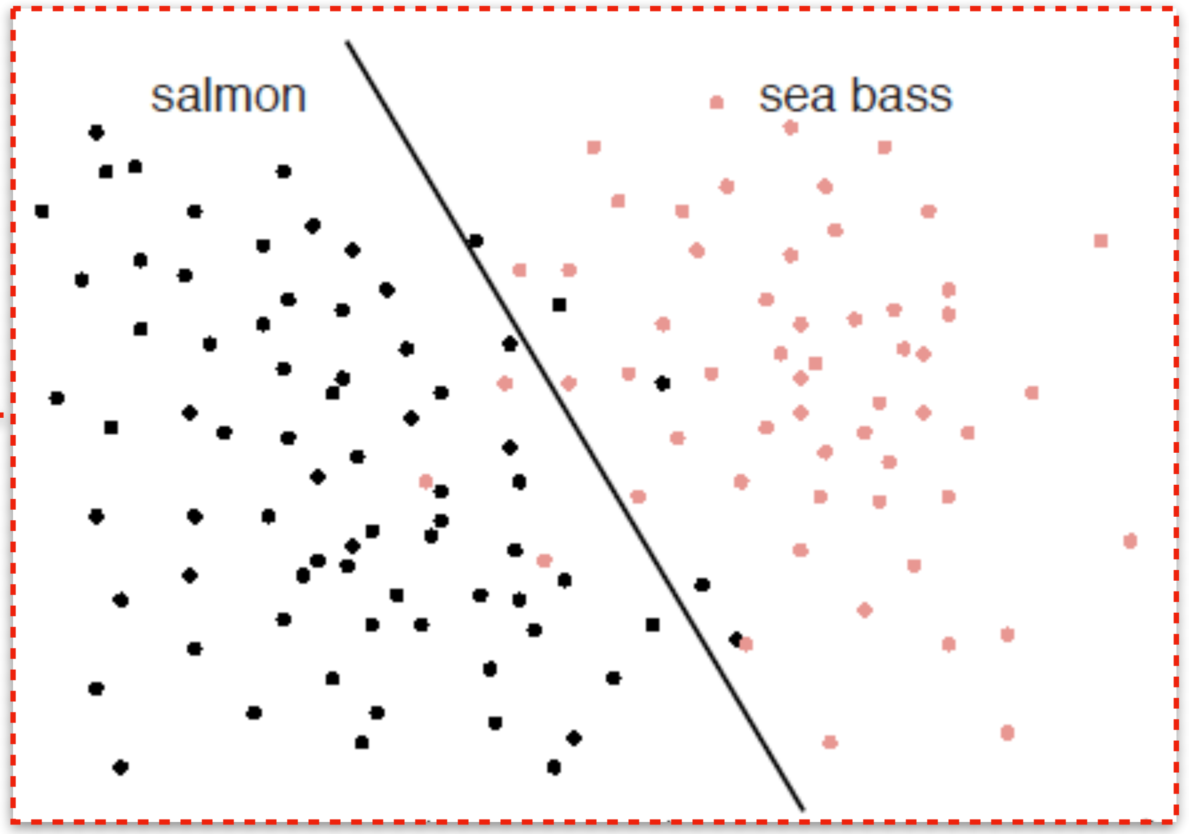
ML Stages



Machine Learning (ML)

ML Stages





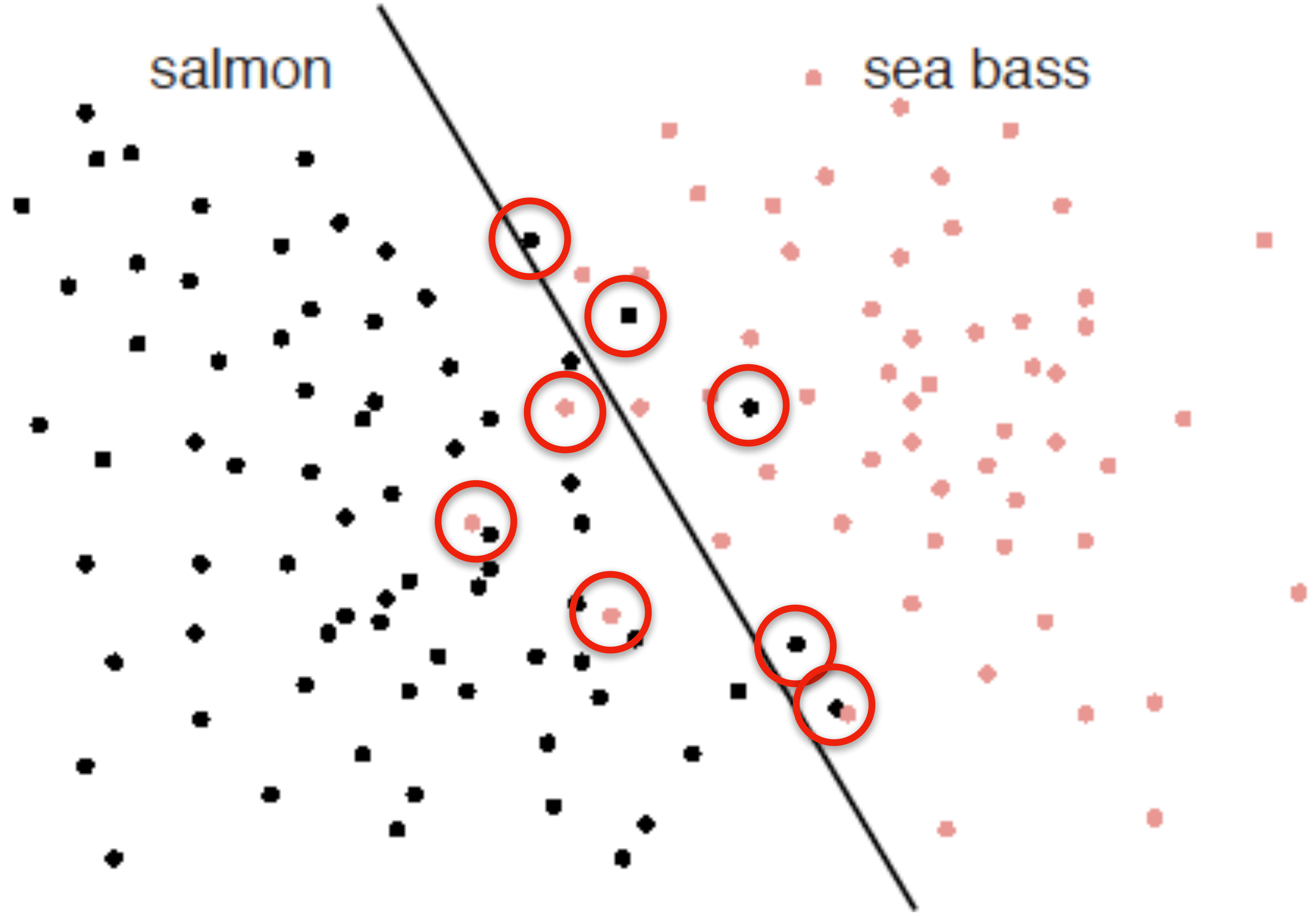
E.g., separation hyperplane.

Duda, O., Hart, P, and Stork, D. *Pattern Classification*. Book, 2nd ed.

LOYOLA UNIVERSITY CHICAGO

Machine Learning (ML)

Metrics - How to measure the performance of the model?



Actual versus Predicted Labels

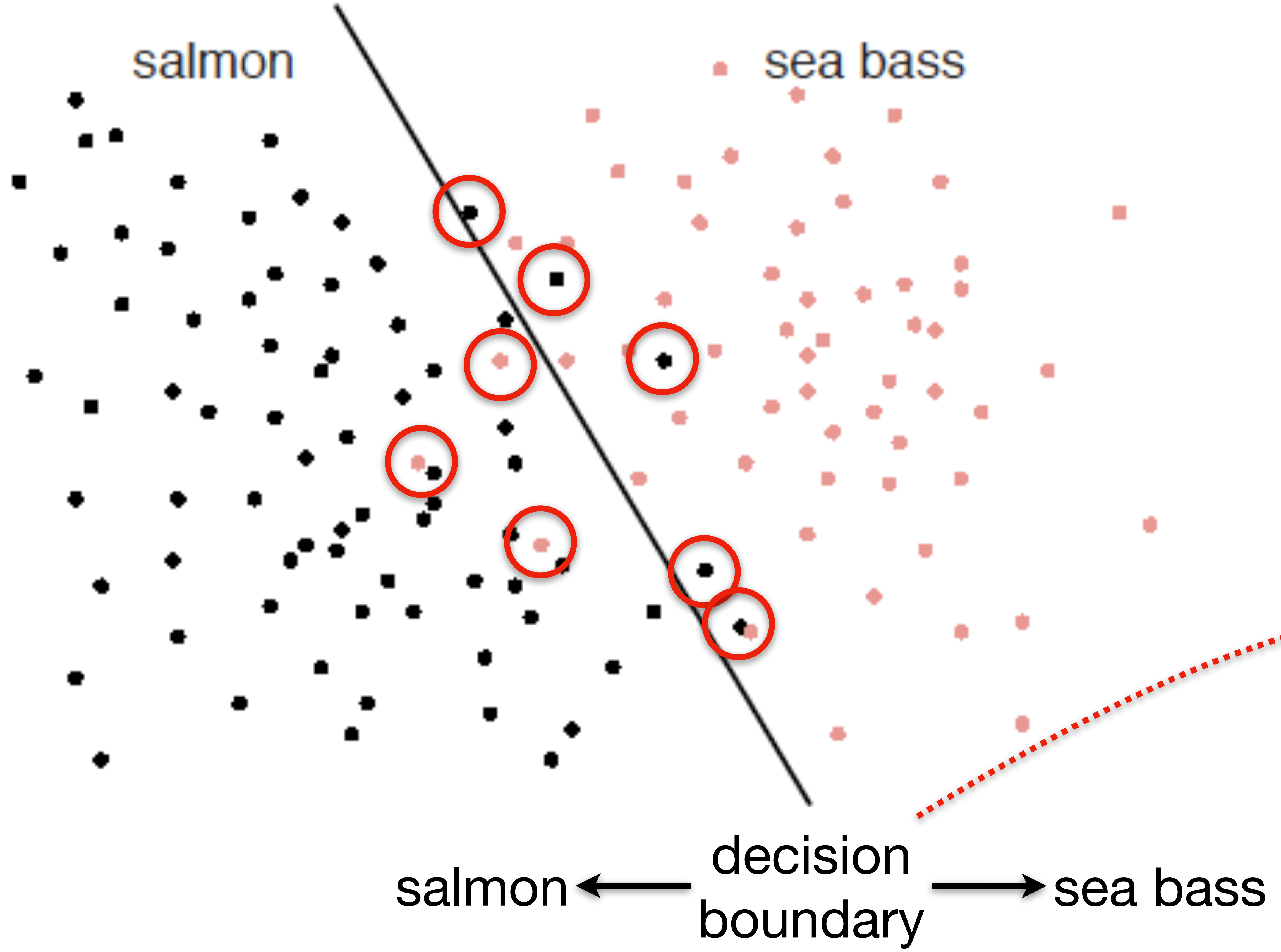
Actual (a.k.a. ground truth)

salmon

sea bass

Machine Learning (ML)

Metrics - How to measure the performance of the model?



Actual versus Predicted Labels

Actual (a.k.a. ground truth)

salmon

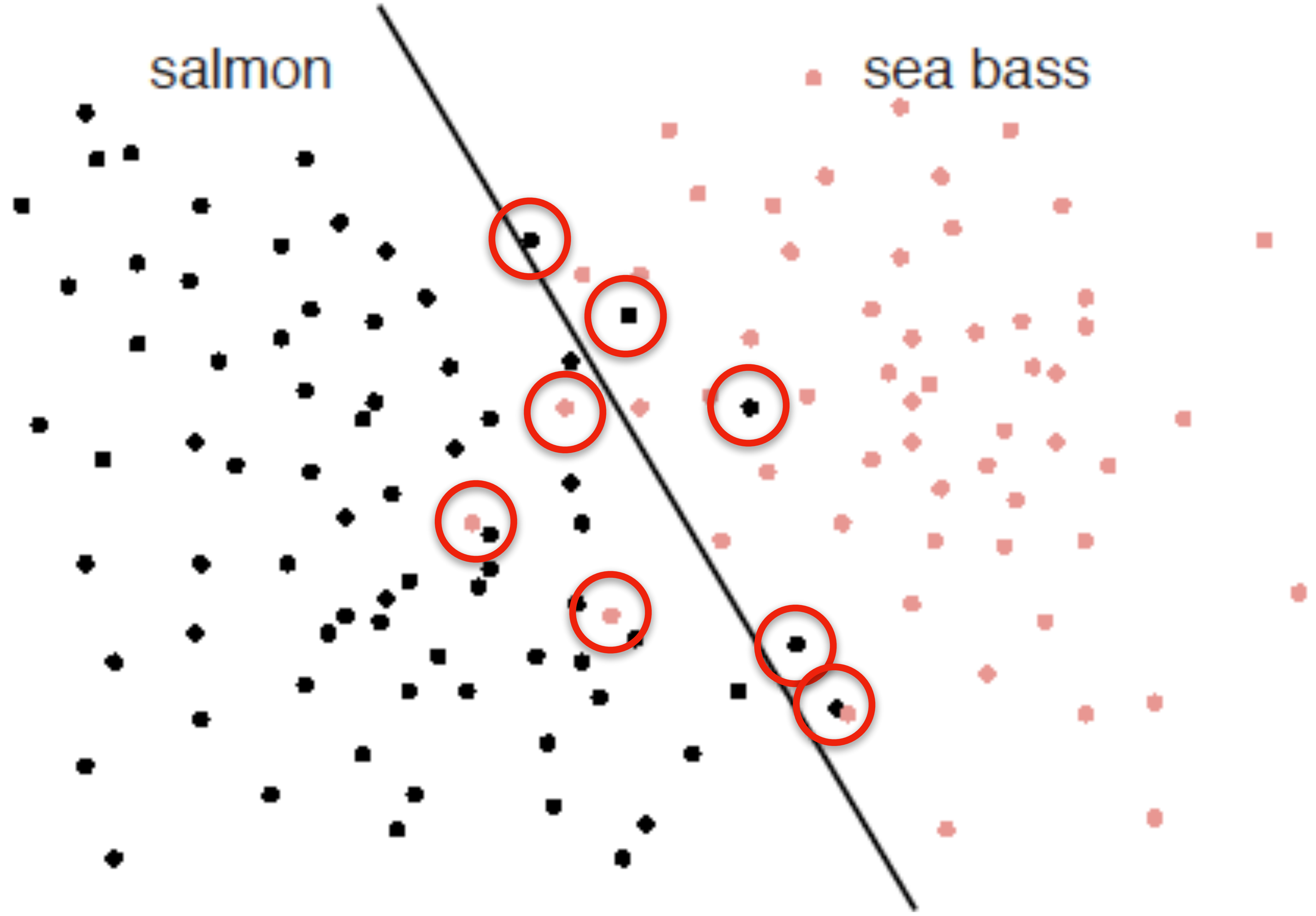
sea bass

Predicted

errors

Machine Learning (ML)

Metrics

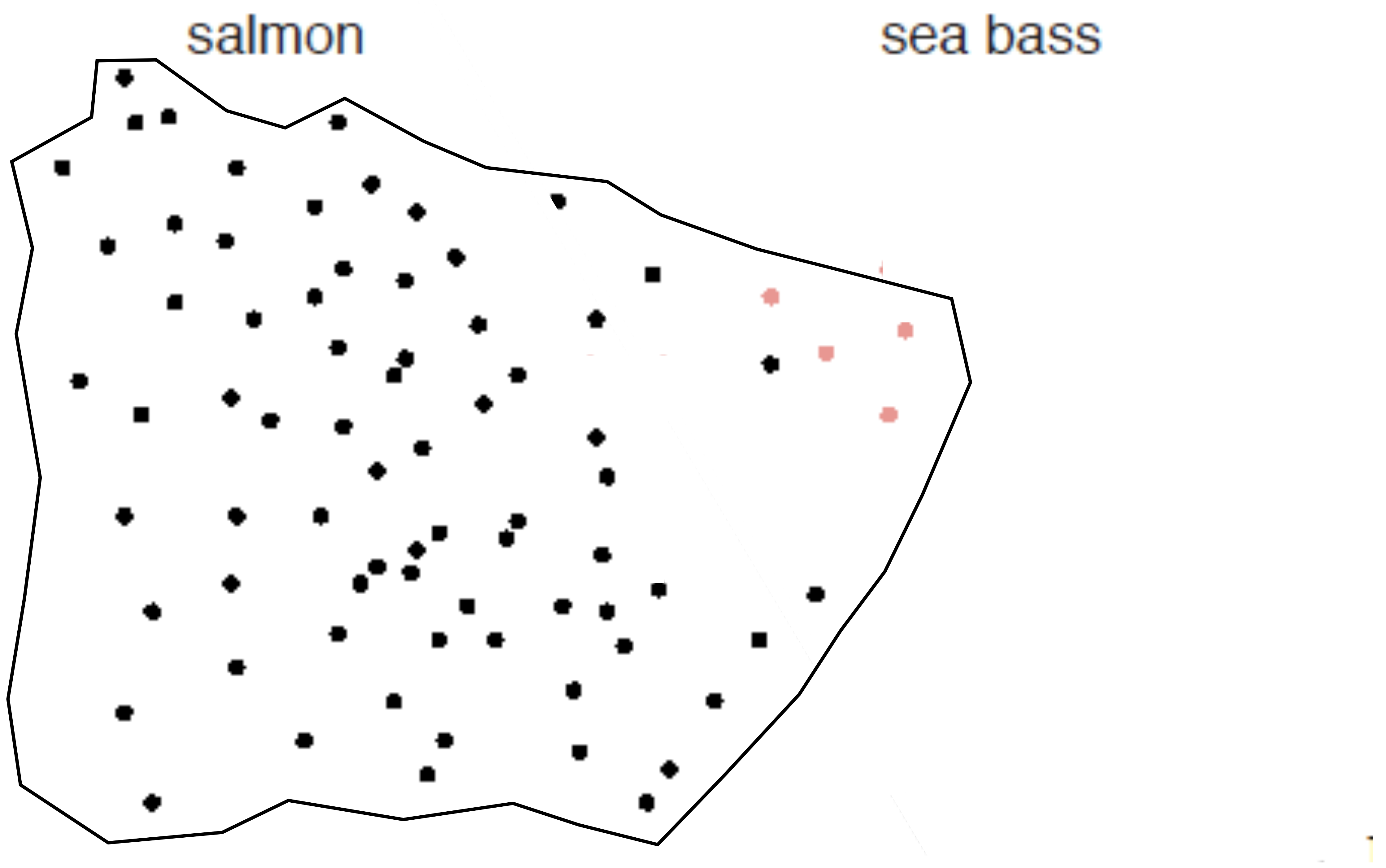


Accuracy (Acc)

$$Acc = \frac{Correct\ Predictions}{Total\ Predictions}$$

Machine Learning (ML)

Metrics



Example model: everything is salmon!

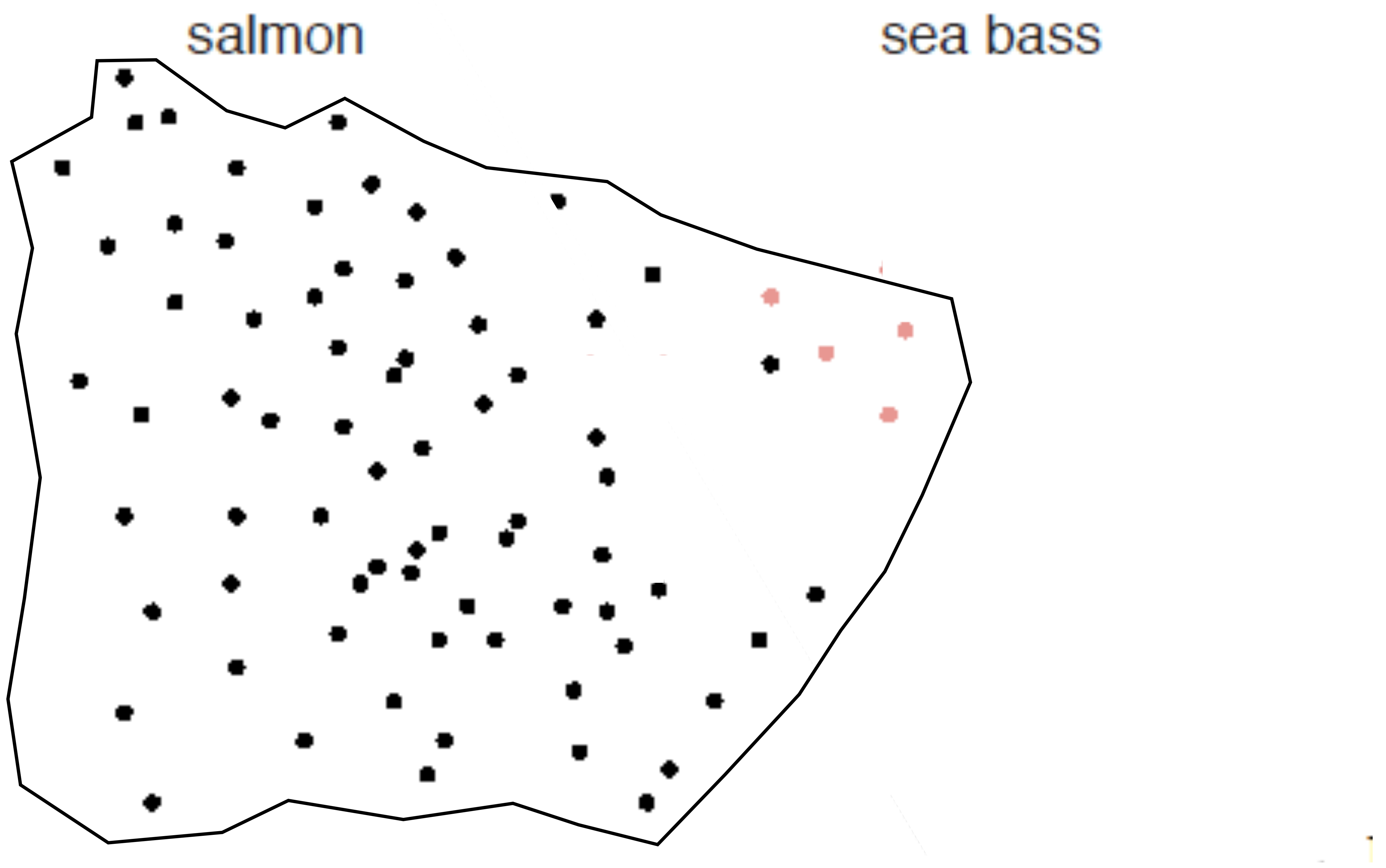
Accuracy (Acc)

$$Acc = \frac{Correct\ Predictions}{Total\ Predictions}$$

Limitation: what happens when we have unbalanced data?

Machine Learning (ML)

Metrics



Example model: everything is salmon!

Balanced Accuracy (BAcc)

$$BAcc = \frac{1}{C}\sum_{i=1}^{C}\frac{Correct\ Predictions_i}{Total\ Predictions_i}$$

C is the number of classes.

Average of class-wise accuracy.

Machine Learning (ML)

Metrics

		Ground truth	
		salmon	not salmon
Predicted	salmon	True Positive (TP)	False Positive (FP)
	not salmon	False Negative (FN)	True Negative (TN)

$i = salmon$

Precision (P) and Recall (R)

$$P = \frac{\sum_{i=1}^{C} TP_i}{\sum_{i=1}^{C} (TP_i + FP_i)}$$

How precise is the model when it classifies as i? (focus on prediction)

$$R = \frac{\sum_{i=1}^{C} TP_i}{\sum_{i=1}^{C} (TP_i + FN_i)}$$

How good is the model in retrieving samples from class i? (focus on ground truth)

C is the number of classes.

Machine Learning (ML)

Metrics

Ground truth

		salmon	not salmon
Predicted	salmon	**True Positive (TP)**	**False Positive (FP)**
	not salmon	**False Negative (FN)**	**True Negative (TN)**

i=*salmon*

Fscore

$$F_\beta = \frac{(\beta^2 + 1) \times P \times R}{\beta^2 \times P + R}$$

Harmonic mean of P and R *(when we care about both).*

$F_1 score:$ $\beta = 1$, equal weight to P and to R.

$F_2 score:$ $\beta = 2$, more weight to R.

Question: can you think of an application where R is more important than P?

LOYOLA UNIVERSITY CHICAGO

Machine Learning (ML)

Metrics

Ground truth

Predicted		salmon	not salmon
	salmon	True Positive (TP)	False Positive (FP)
	not salmon	False Negative (FN)	True Negative (TN)

i=*salmon*

Fscore

$$F_\beta = \frac{(\beta^2 + 1) \times P \times R}{\beta^2 \times P + R}$$

Harmonic mean of P and R *(when we care about both).*

$F_1 score:$ $\beta = 1$, equal weight to P and to R.

$F_2 score:$ $\beta = 2$, more weight to R.

$F_{0.5} score:$ $\beta = 0.5$, more weight to P.

Machine Learning (ML)

Metrics

Ground truth

Predicted	salmon	sea bass	tuna	pirarucu
salmon	215	32	0	3
sea bass	10	701	0	0
tuna	1	0	370	2
pirarucu	1	0	0	102

Number of samples.

Confusion Matrix

Visualization tool: *what can you see?*

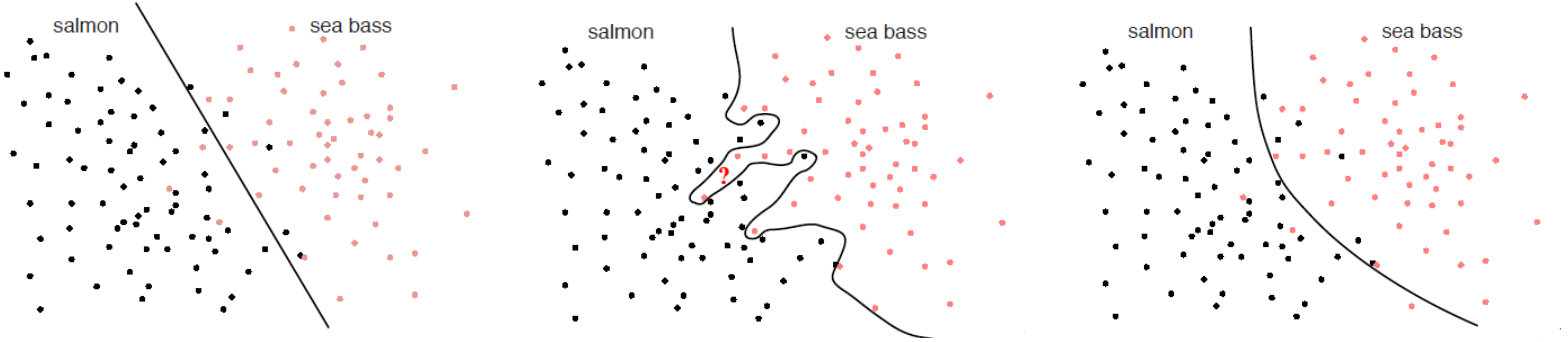
*Why is it called **confusion** matrix?*

Machine Learning (ML)

Data-driven Learning Issues

What happens in face of unseen data (normal system operation)?

Duda, O., Hart, P, and Stork, D.
Pattern Classification. Book, 2nd ed.



Under-fitting
(too-simple model)

Over-fitting
(too-complex model)

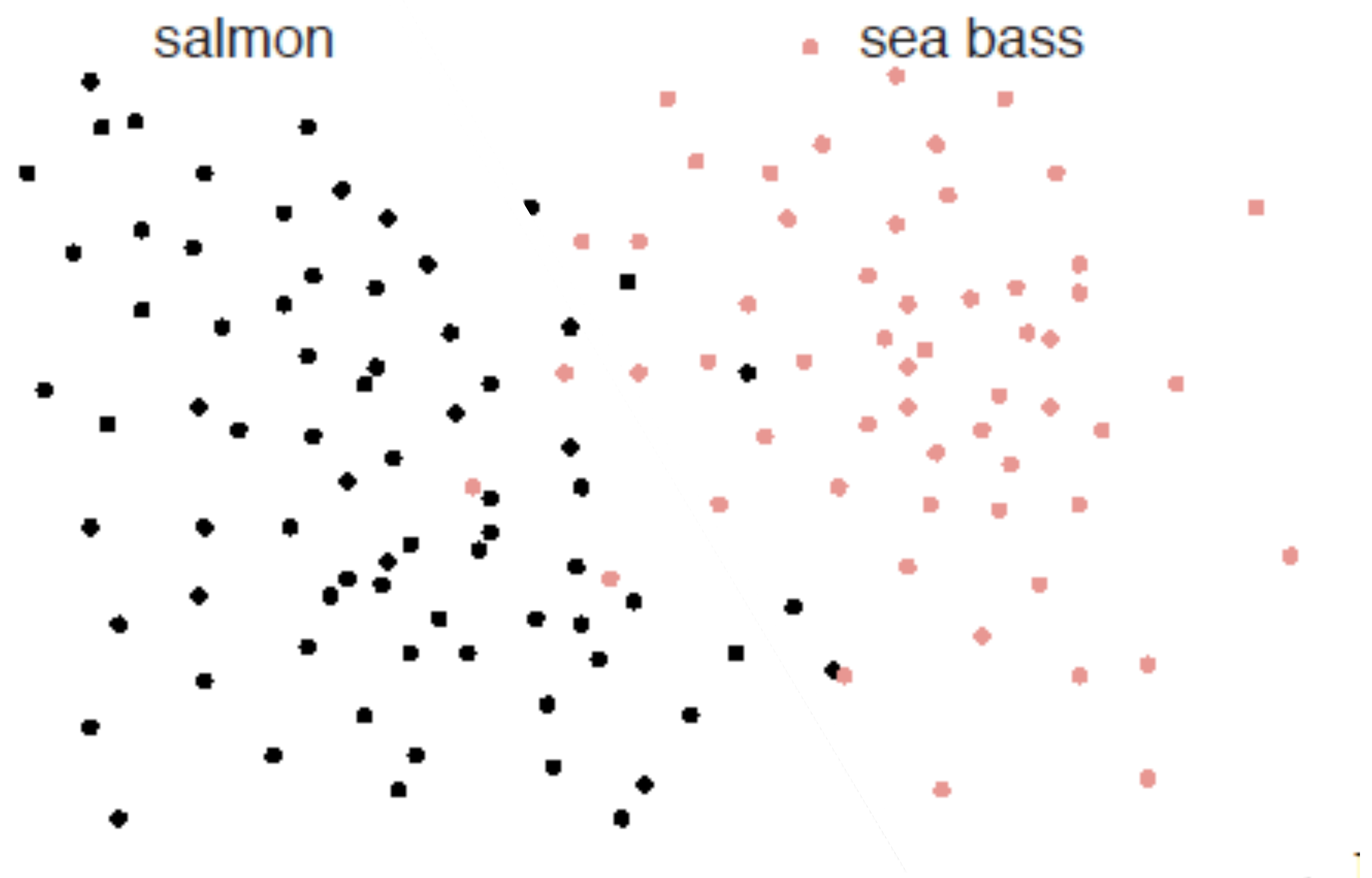
Okay-ish

LOYOLA UNIVERSITY CHICAGO

Machine Learning (ML)

Data Split

$\Re^N$ space



How to estimate the model's performance in face of unseen data?

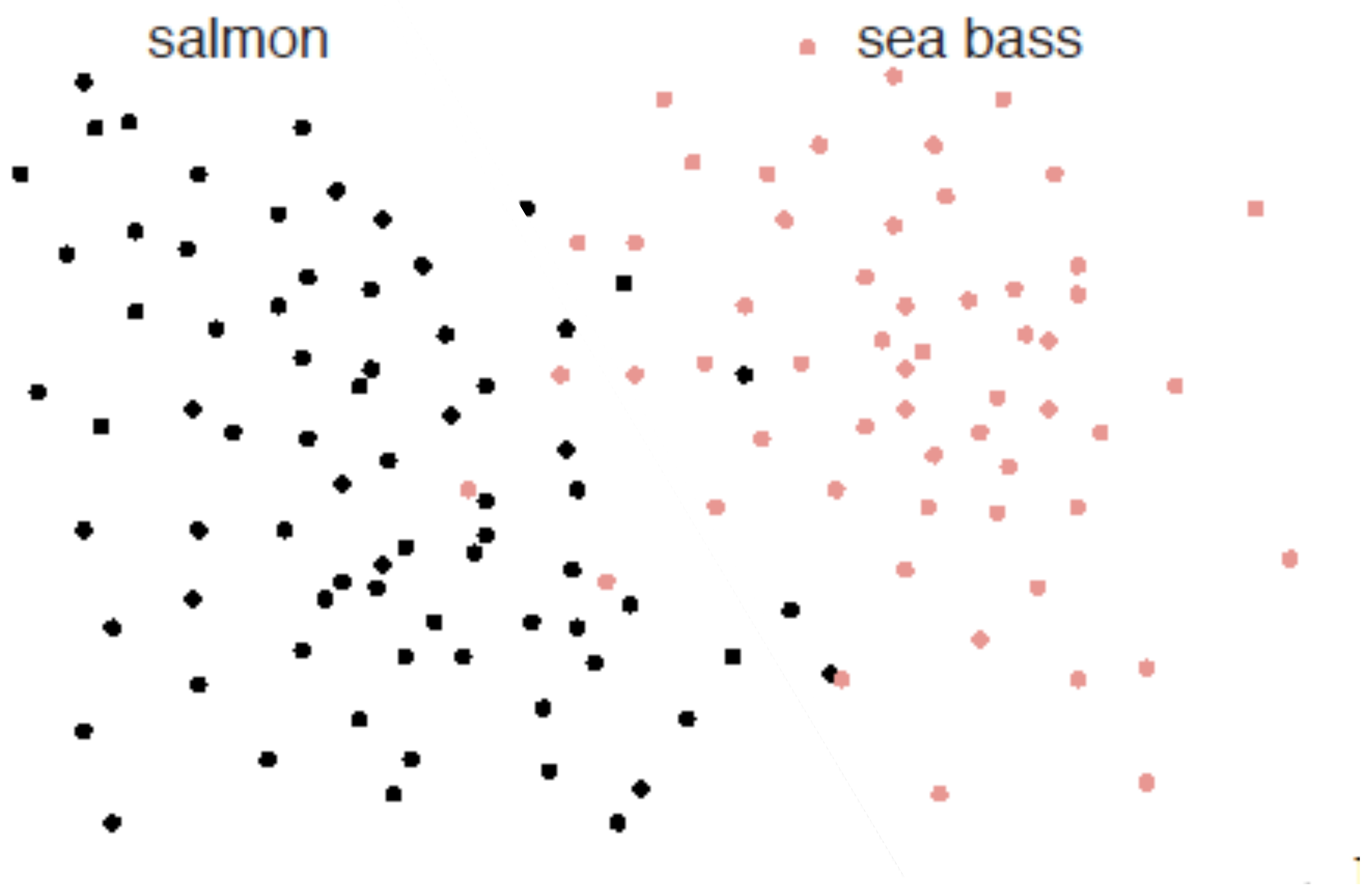
Random split *X %* *100-X%*

Training Set	Test Set
Present to the learner (learning stage).	Reserve to assess performance in the end (inference stage).

Machine Learning (ML)

Data Split

$\Re^N$ space



How to estimate the model's performance in face of unseen data?

Alternative rand. split	*X %*	*Y %*	*100-(X+Y)%*
	Training Set	Validation Set	Test Set

If one needs performance assessment during learning.

Reserved! Don't touch it until inference time.

Machine Learning (ML)

Random Data Split

What if the estimated performance (P) is a matter of chance?

K-fold Cross Validation

Example: K=5, 5 random splits.

Report μ_P and σ_P.

Machine Learning (ML)

Random Data Split

What if the estimated performance (P) is a matter of chance?

K-fold Cross Validation

Example: K=5, 5 random splits.

Report μ_P and σ_P.

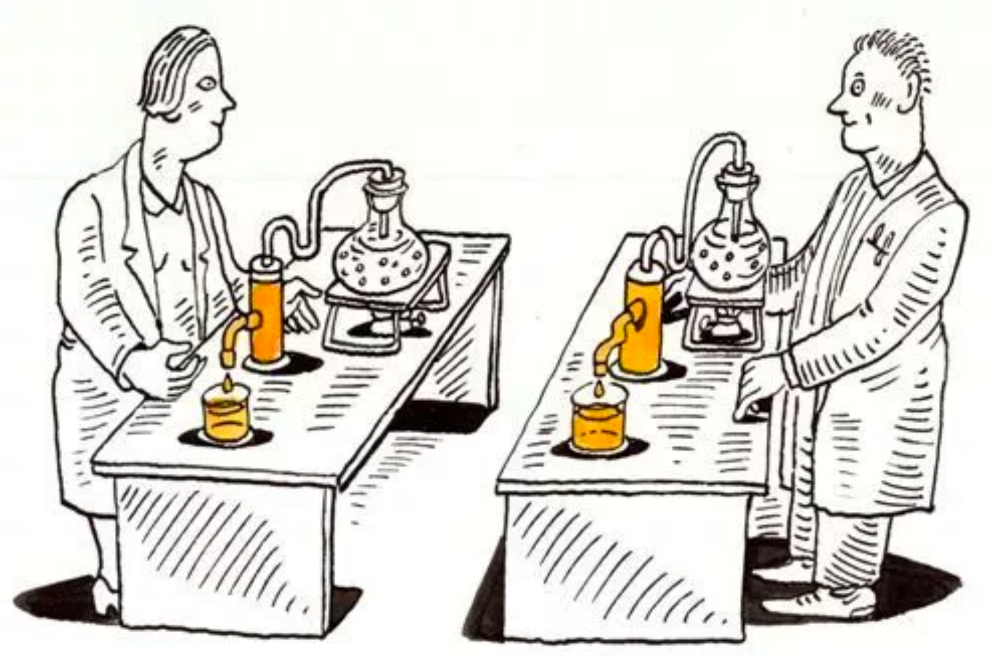
Public Random Split

No time to train the solution K times?

Make random split publicly available, so others can:

1. Reproduce your results.

2. Use the same split to train and compare their solutions.



https://www.displayr.com/what-is-reproducible-research/

LOYOLA UNIVERSITY CHICAGO

Machine Learning (ML)

Data-driven Learning Types

Supervised Learning (1/2)

The target problem has
well-defined classes.

There are annotated data to train
the learner.

Annotation: each sample (feature vector)
has a class.

Yao, L., Miller, J.
Tiny ImageNet with CNNs. CS Stanford, 2015.

LOYOLA UNIVERSITY CHICAGO

Machine Learning (ML)

Data-driven Learning Types

Supervised Learning (1/2)
Closed Set versus Open Set

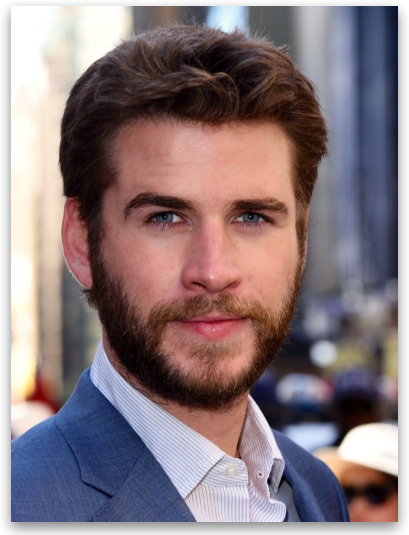
E.g., Face Recognition

query

LOYOLA UNIVERSITY CHICAGO

Machine Learning (ML)

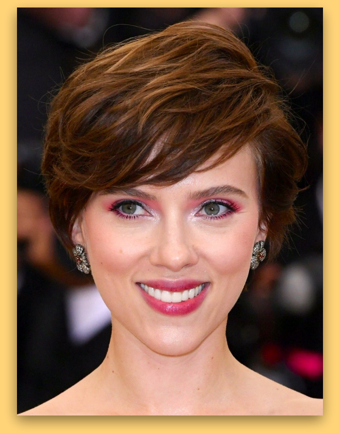
Closed Set versus Open Set



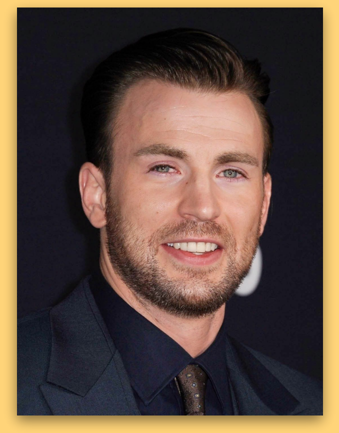
Query
(Liam Hemsworth)

Dataset

Robert Downey Jr.



Scarlet Johansson



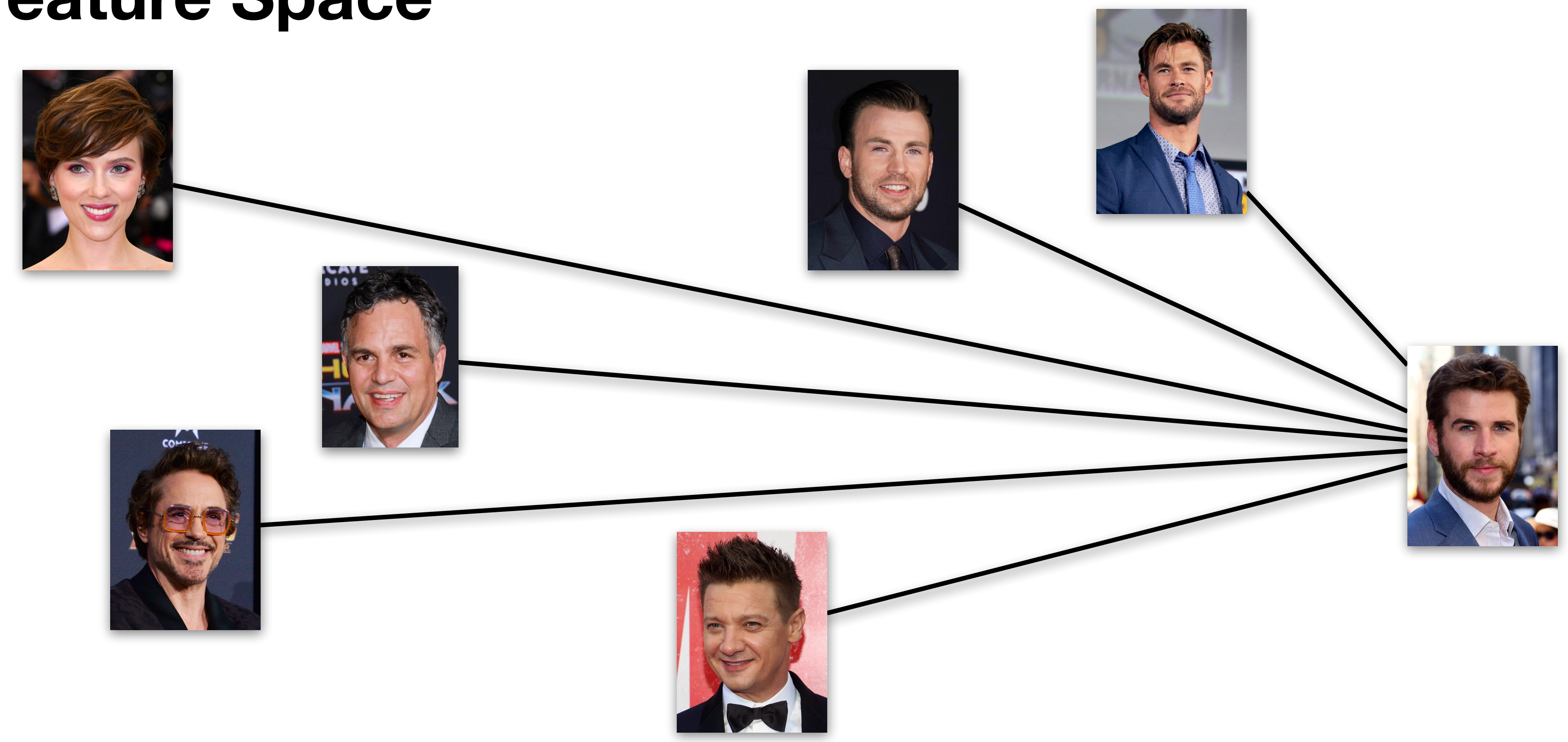
Chris Evans

Mark Ruffalo

Chris Hemsworth

Jeremy Renner

Feature Space



Closed Set

Output
This is
Chris Hemsworth!

Open Set

Output
I don't know
this person!

Machine Learning (ML)

Data-driven Learning Types

Unsupervised Learning (2/2)

Either the target problem has
no well-defined classes.

OR

There are not annotated data to train
the learner.

Machine Learning (ML)

Learner and Model Solutions

Learning Stage

Example Data → Learner → Model

What solutions of Learner can we use?

Let's see some…

Unsupervised
Clustering methods such as k-means, k-medoids, etc.

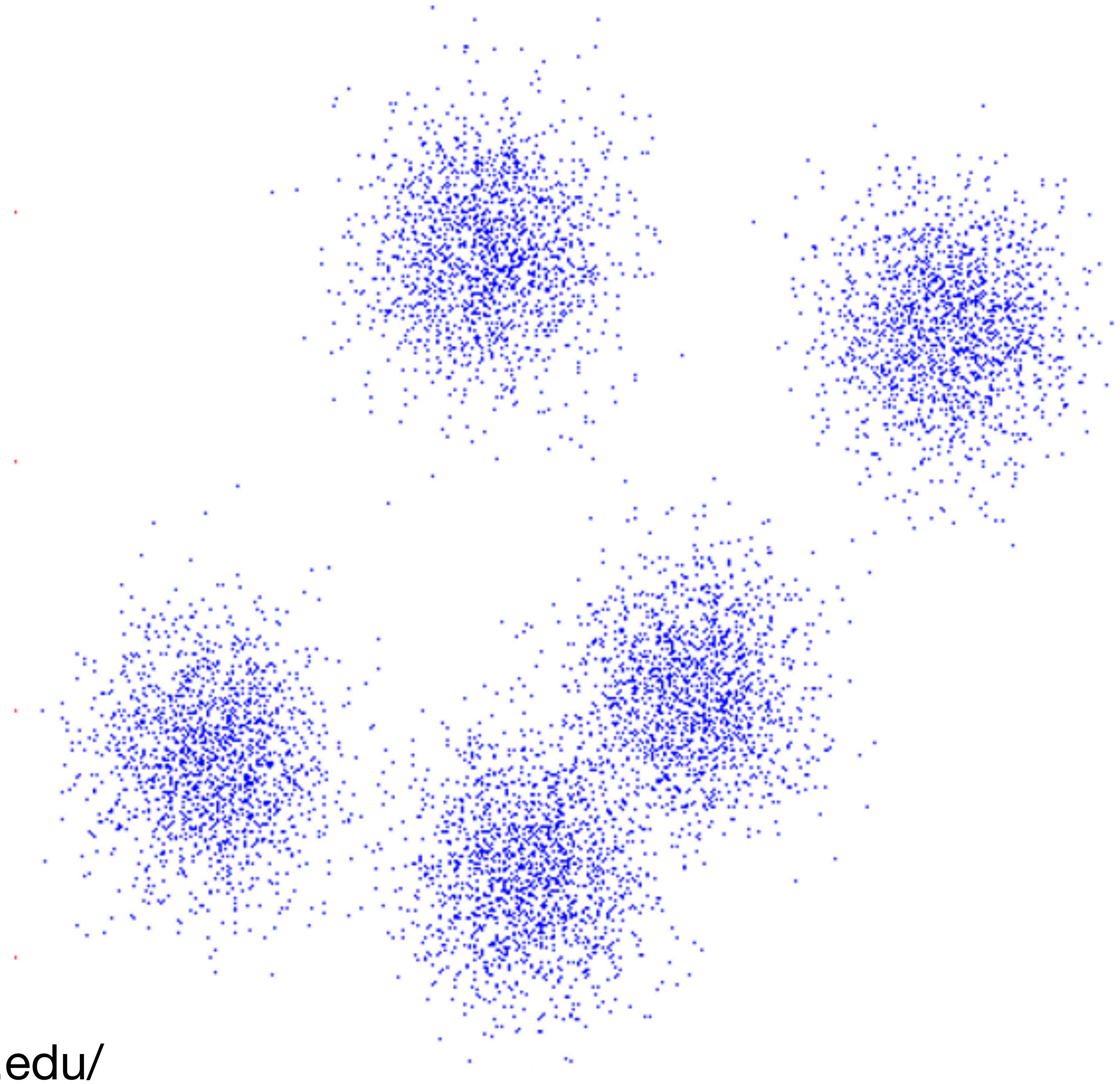
Supervised
Decision trees, random forests, Support Vector Machines (SVM), typical Neural Networks (NN), etc.

K-Means

How to reduce
data complexity?

Cluster the features and limit the k-nearest search to one or a couple of clusters.

There will be less elements to consider.



Source: https://people.csail.mit.edu/dsontag/courses/ml12/slides/lecture14.pdf

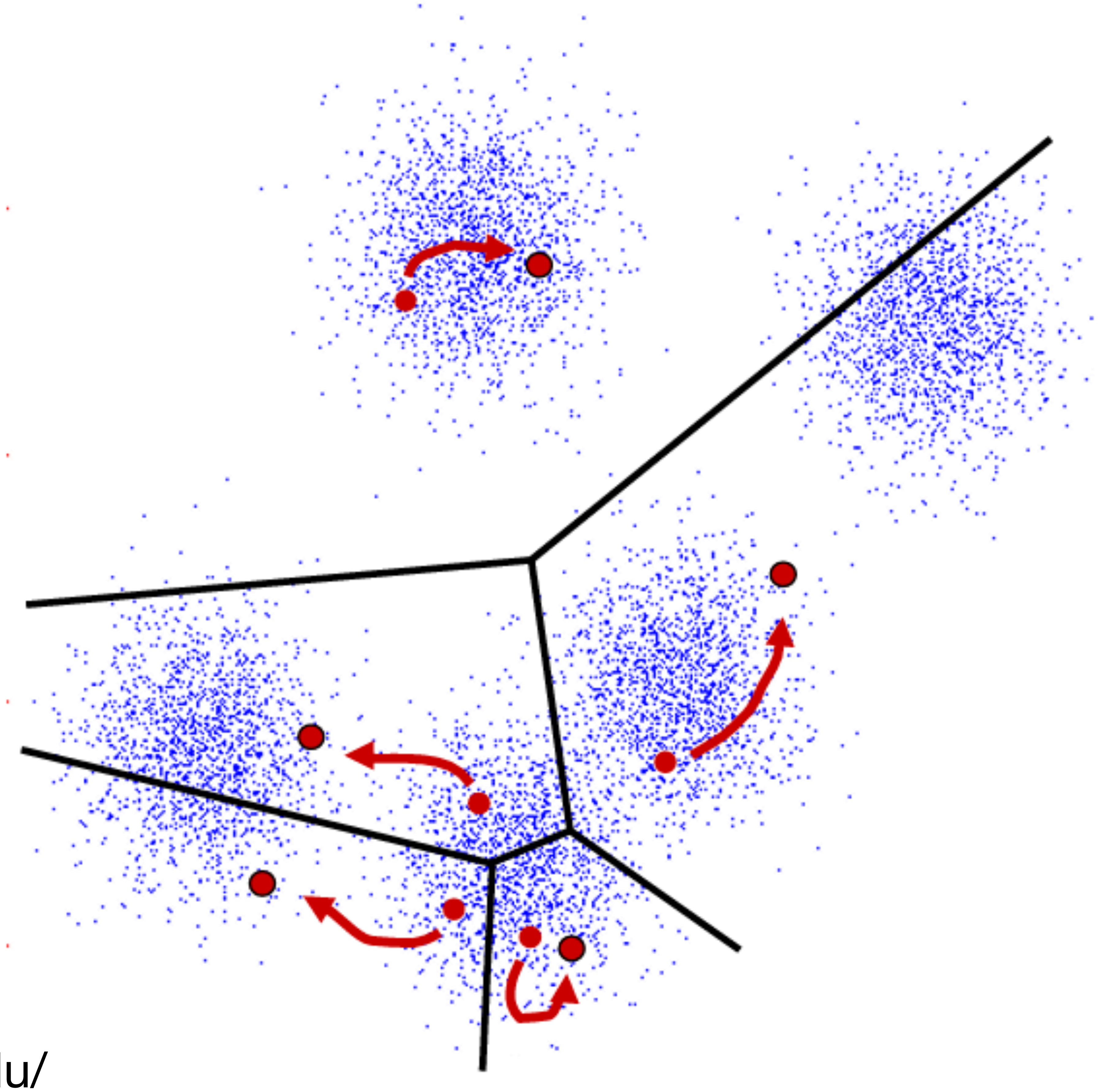
LOYOLA UNIVERSITY CHICAGO

K-Means

How to reduce
data complexity?

Cluster the features and limit the k-nearest search to one or a couple of clusters.

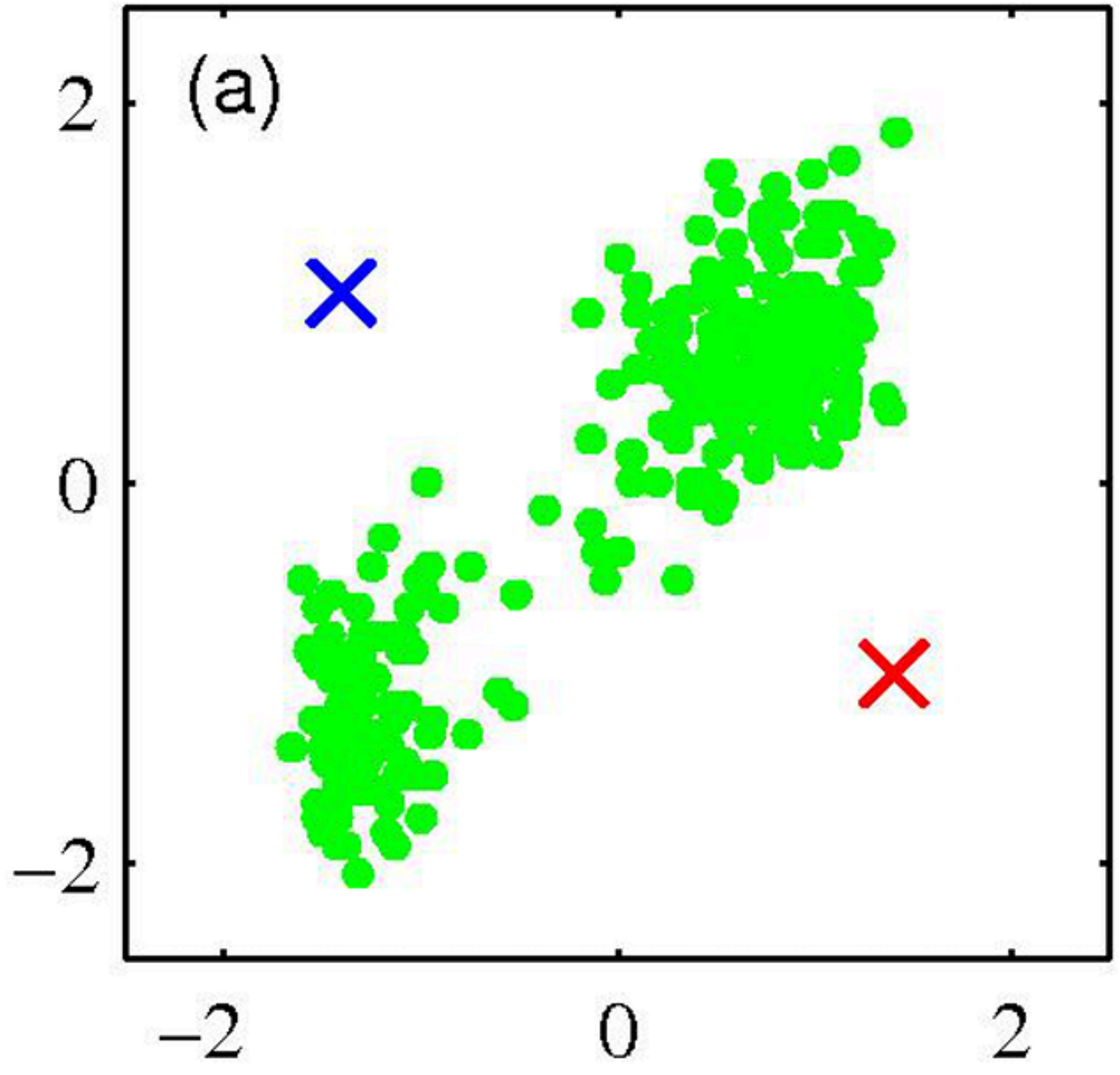
There will be less elements to consider.



Source: https://people.csail.mit.edu/dsontag/courses/ml12/slides/lecture14.pdf

K-Means

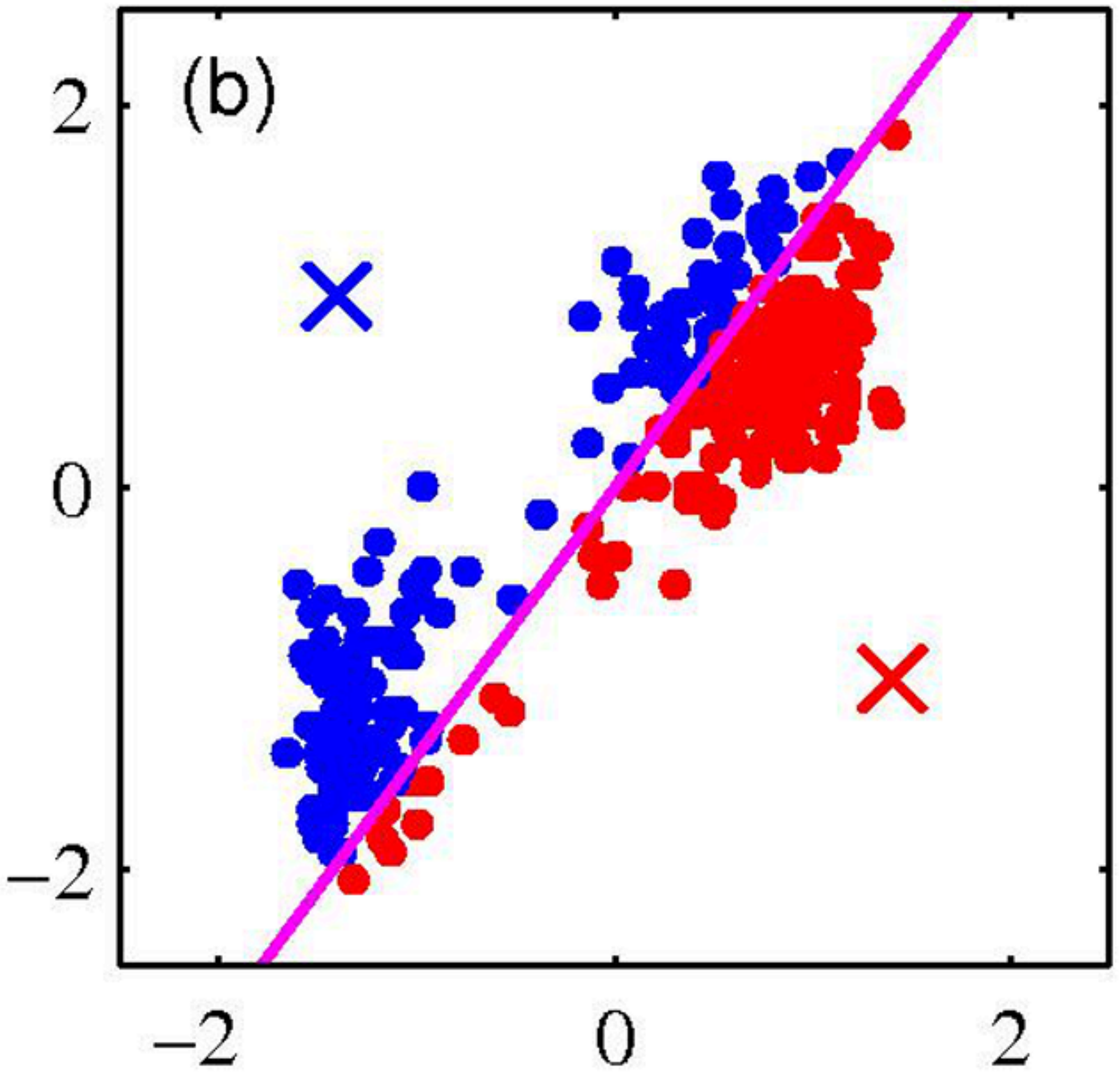
Select K random features as cluster centers.



Source: https://people.csail.mit.edu/dsontag/courses/ml12/slides/lecture14.pdf

K-Means

Assign features to the closest cluster centers.

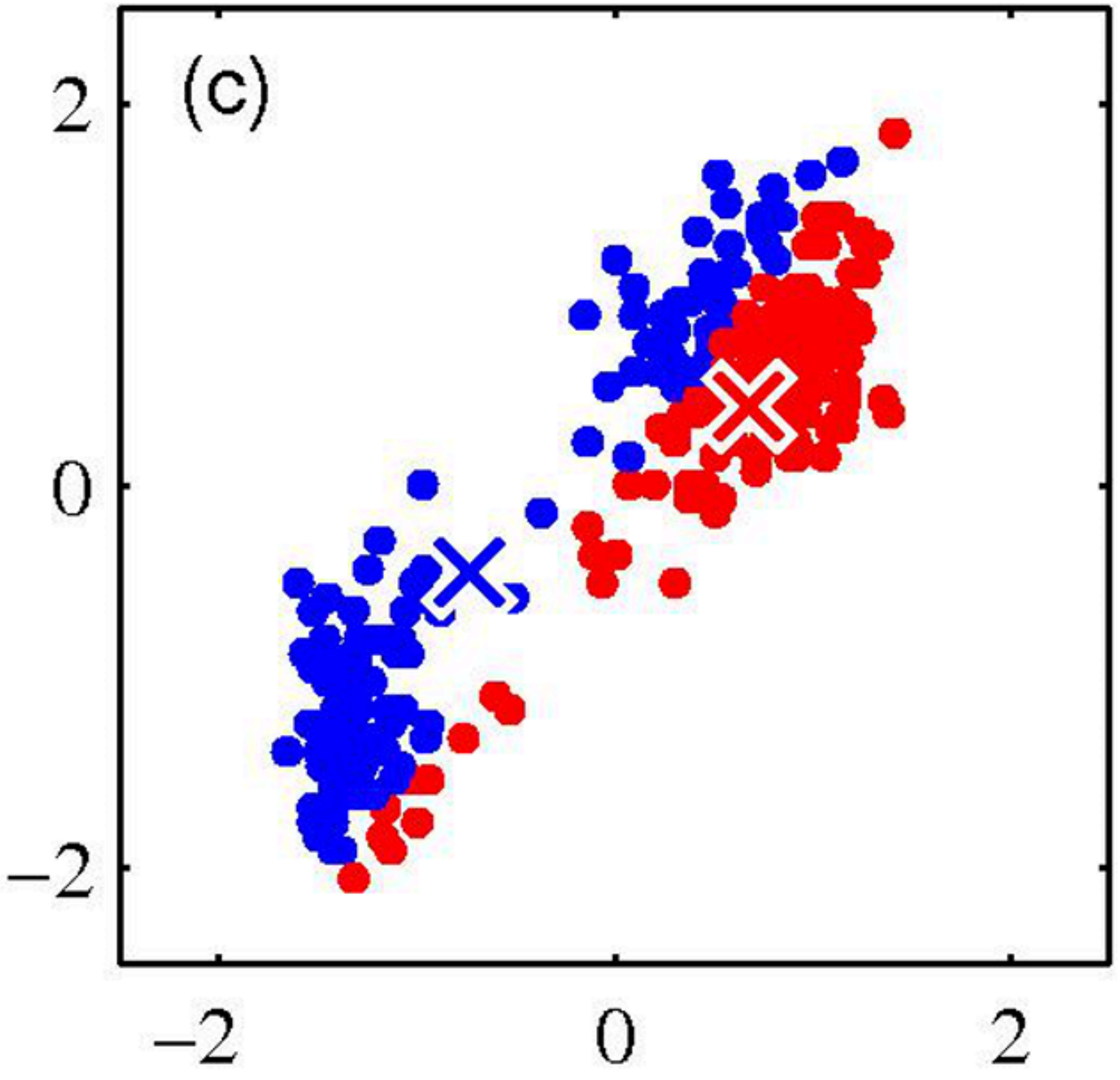


Source: https://people.csail.mit.edu/dsontag/courses/ml12/slides/lecture14.pdf

LOYOLA UNIVERSITY CHICAGO

K-Means

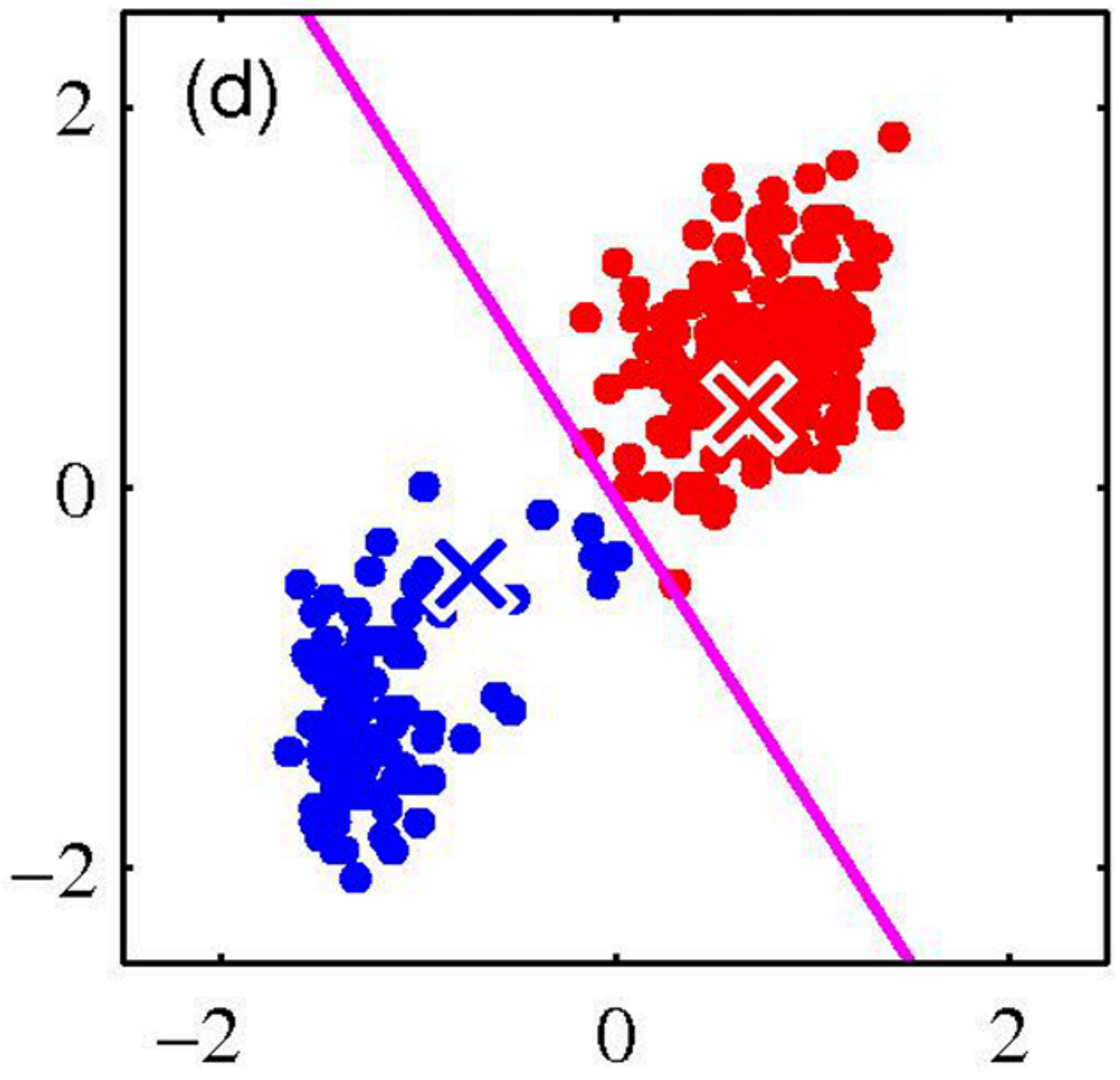
Update the cluster centers by taking the **means** of each cluster.



Source: https://people.csail.mit.edu/dsontag/courses/ml12/slides/lecture14.pdf

K-Means

Repeat until convergence.



Source: https://people.csail.mit.edu/dsontag/courses/ml12/slides/lecture14.pdf

LOYOLA UNIVERSITY CHICAGO

K-Means

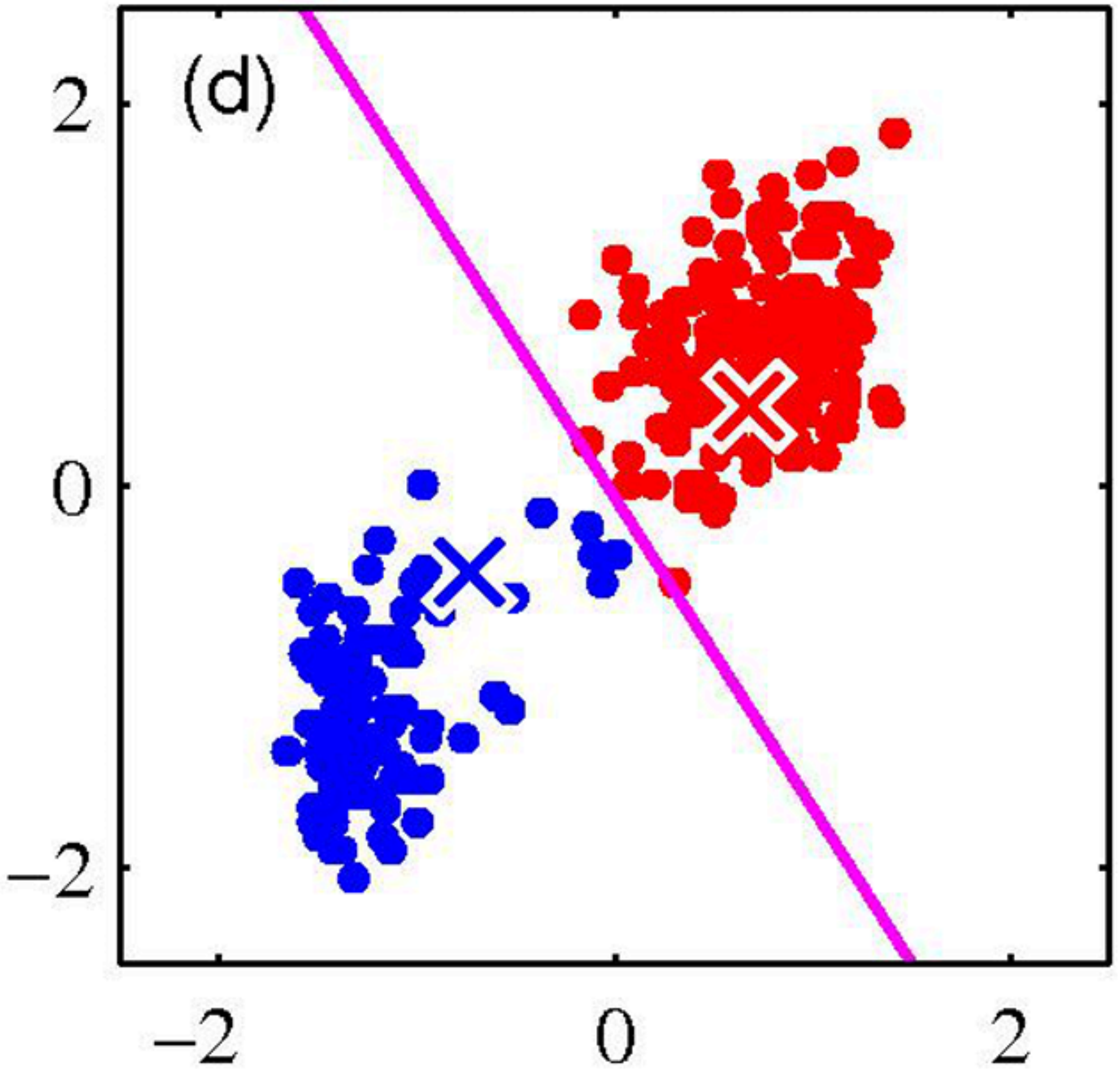
What are the limitations of this approach?

What is the ideal number of clusters?

Complexity of building clusters: *O(Kn)* in each step until convergence.

K: #clusters
n: #features

Clustering is *offline*: i.e., it does not happen at feature querying time.

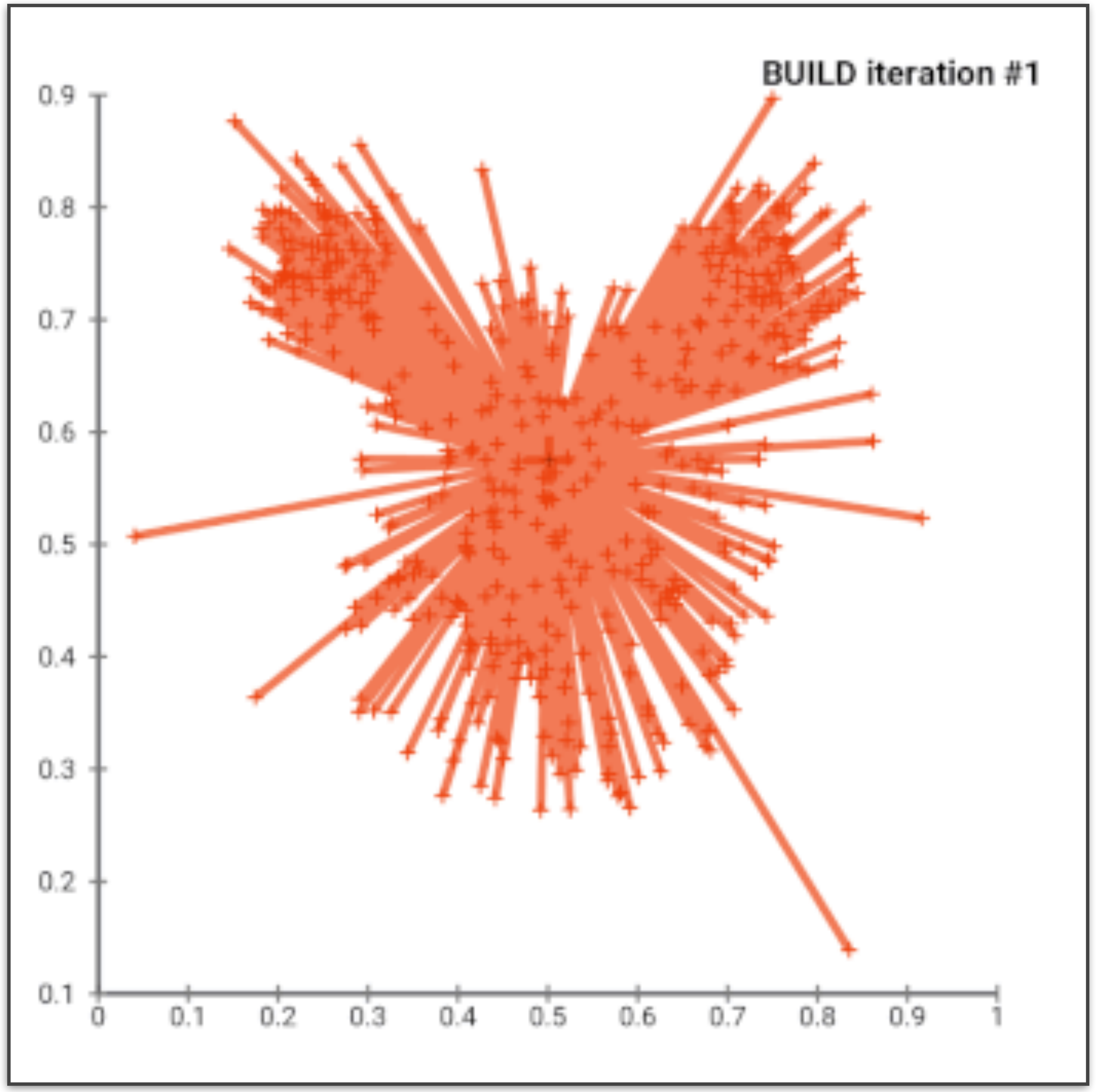


LOYOLA UNIVERSITY CHICAGO

K-Means

Variation: K-medoids

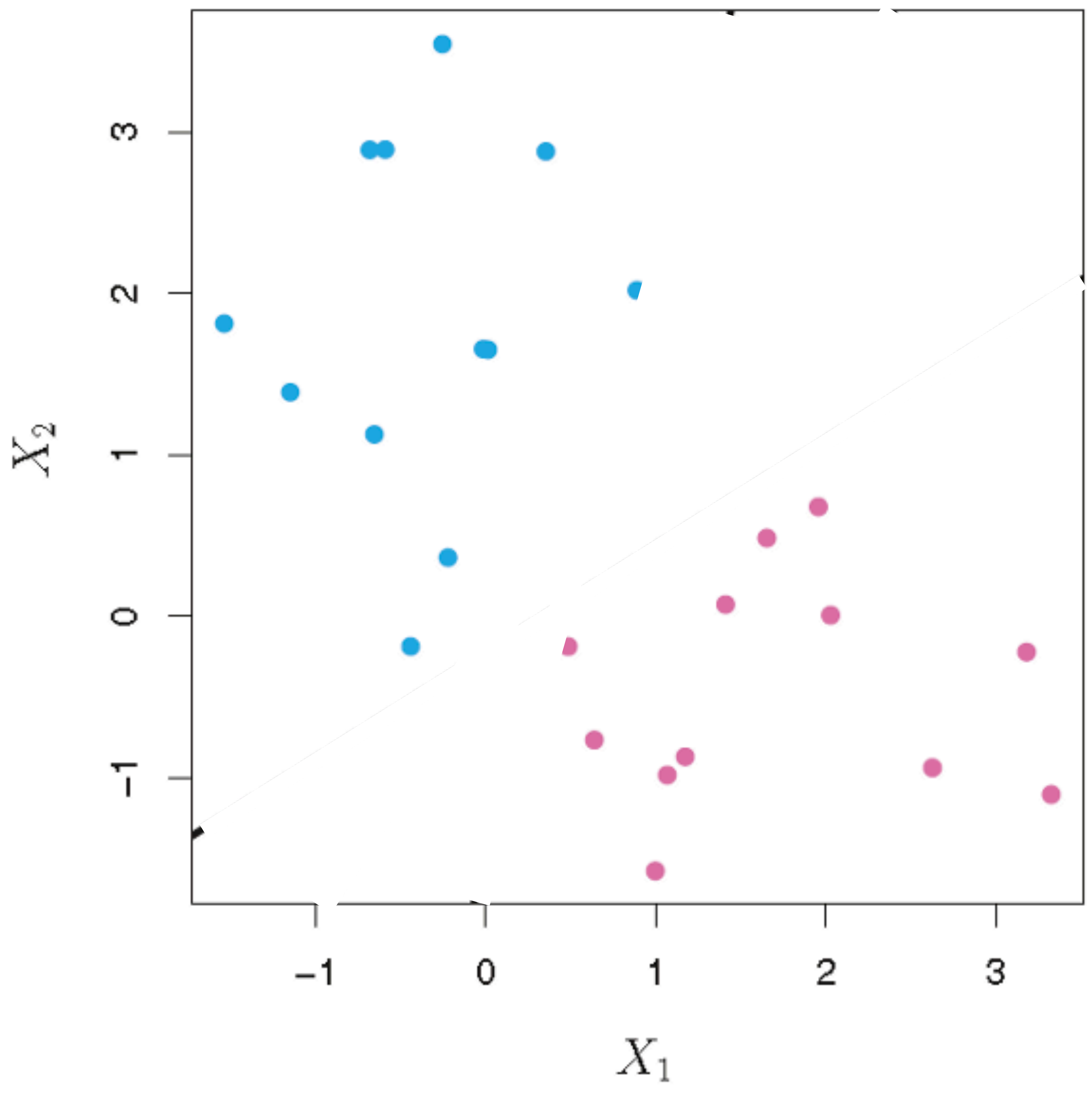
Instead of using *means* as the cluster centers, use *medians*, which are actual existing features.



LOYOLA UNIVERSITY CHICAGO

Support Vector Machines (SVM)

How to Separate these Features?

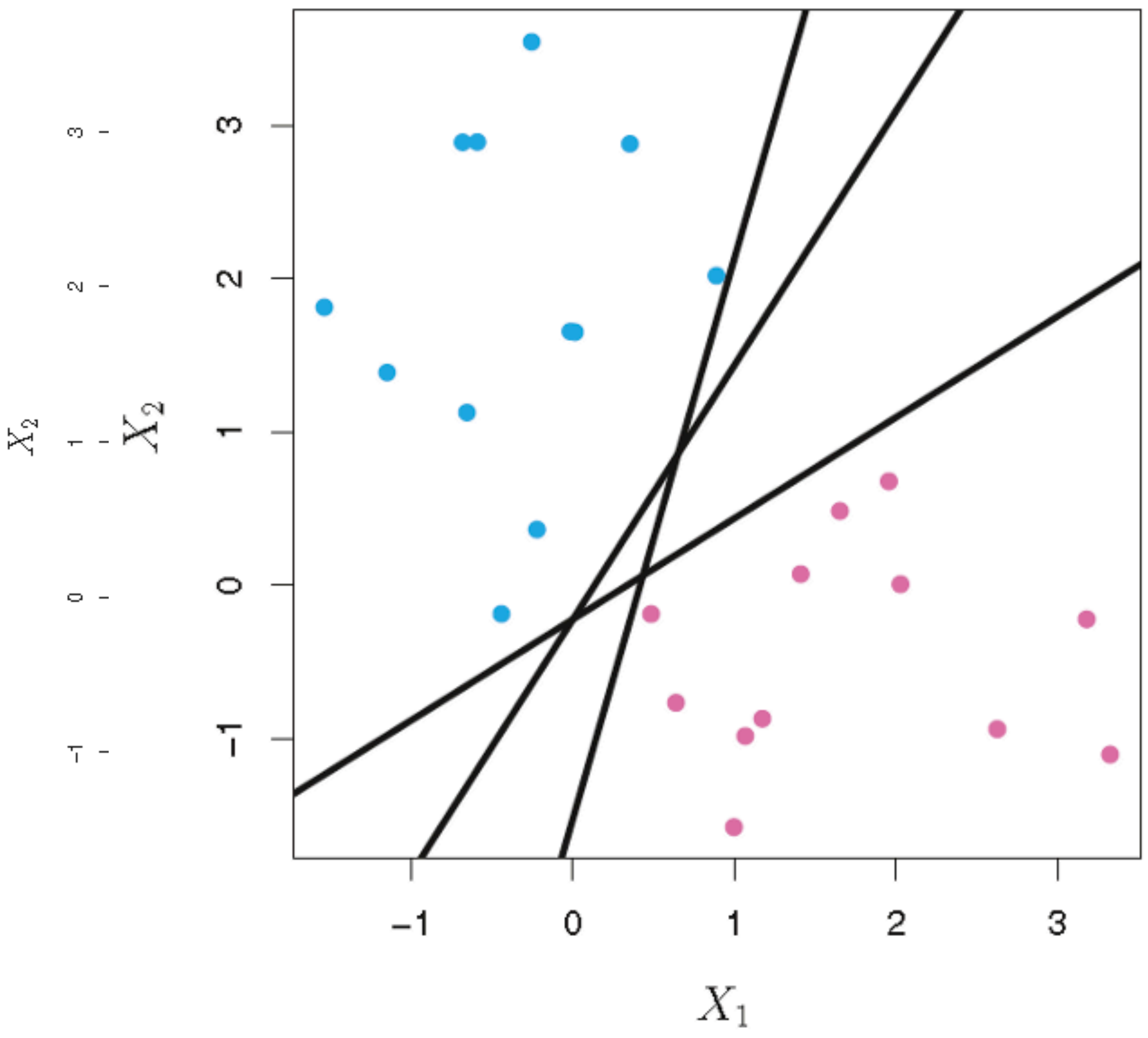


https://towardsdatascience.com/the-kernel-trick-c98cdbcaeb3f

Support Vector Machines (SVM)

How to Separate these Features?

They’re linearly separable, but what is the best separation?



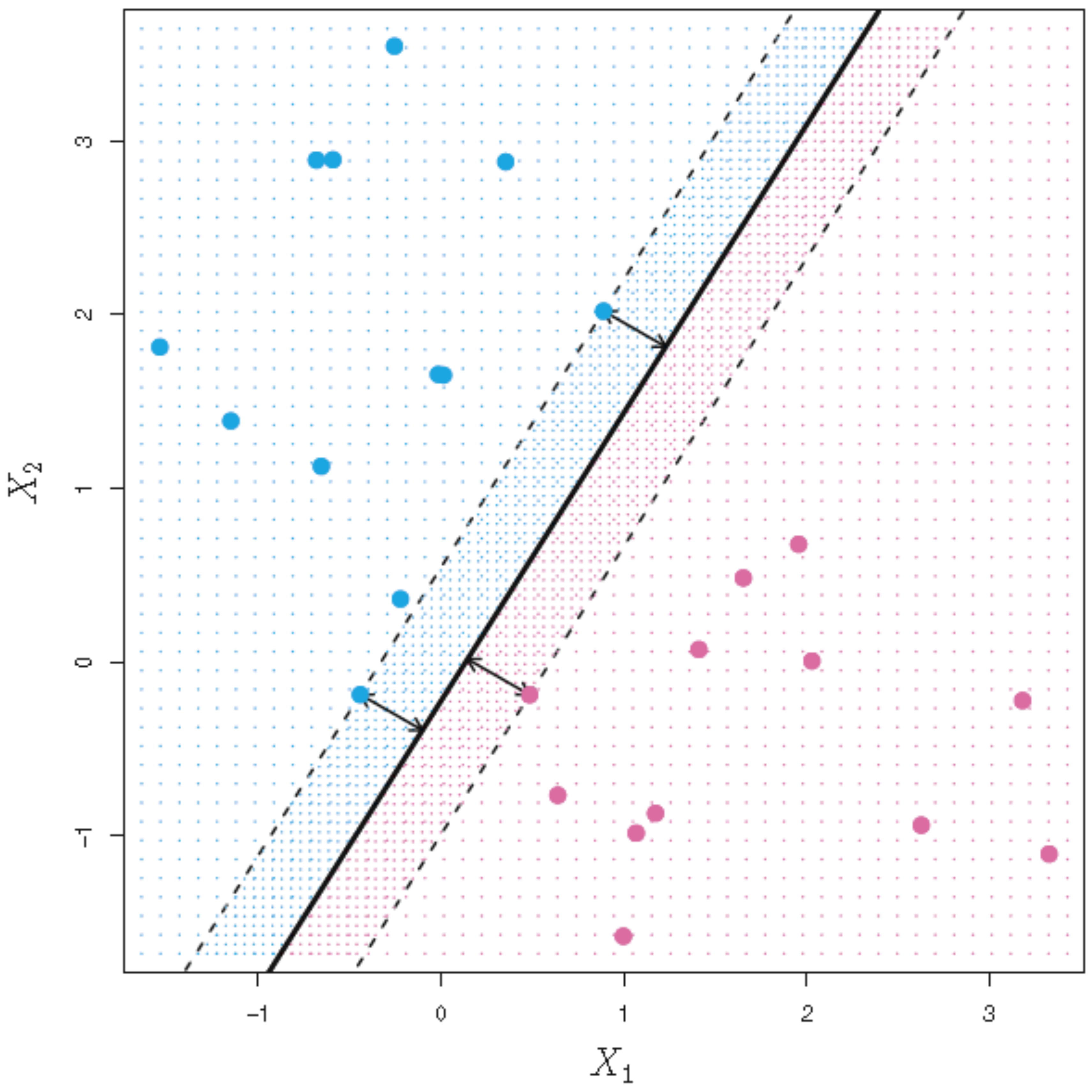
https://towardsdatascience.com/the-kernel-trick-c98cdbcaeb3f

Support Vector Machines (SVM)

How to Separate these Features?

They're linearly separable, but what is the best separation?

Solution:
Find the hyperplane that maximizes the margin between the classes.

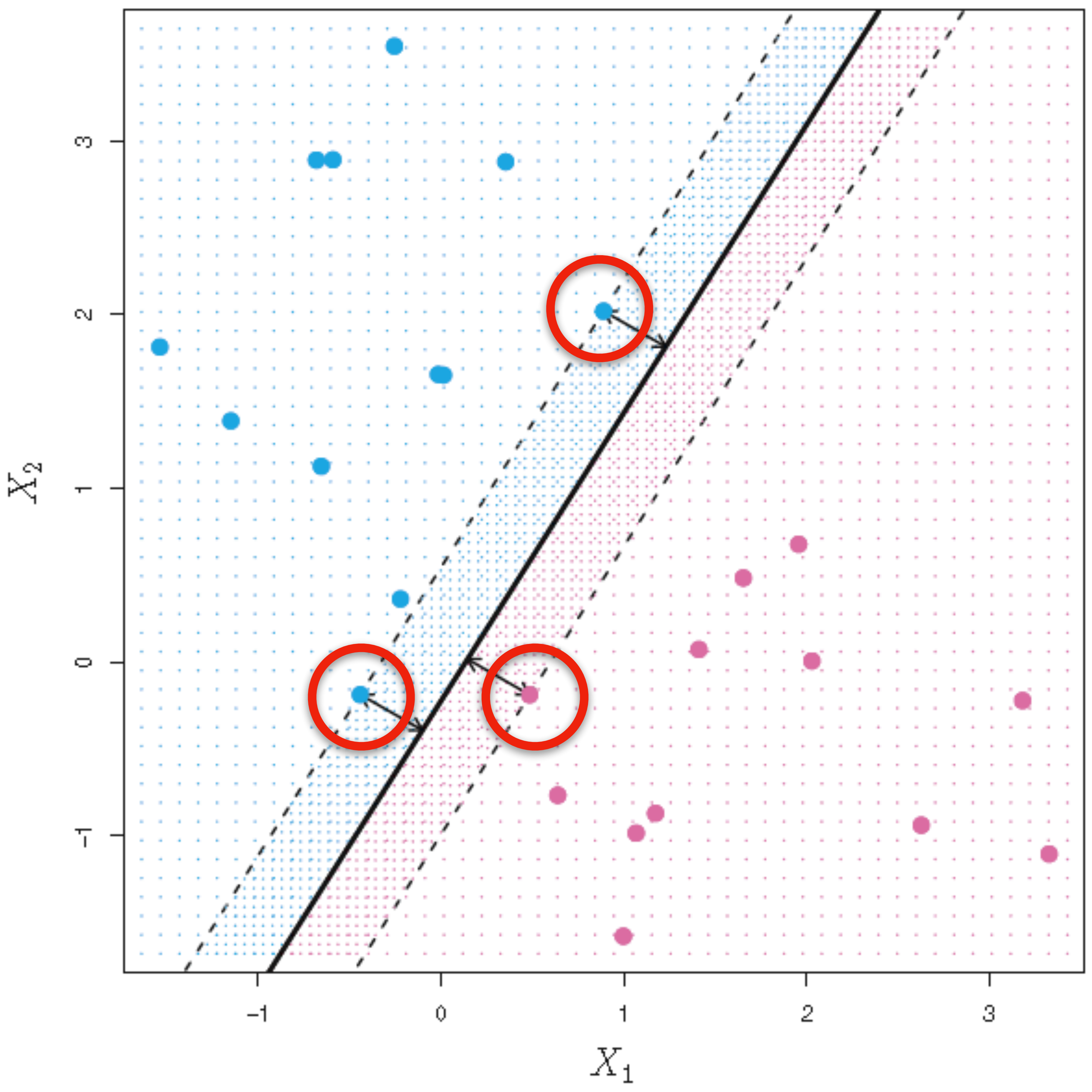


https://towardsdatascience.com/the-kernel-trick-c98cdbcaeb3f

Support Vector Machines (SVM)

How to Separate these Features?

The feature vectors serving as reference to the margin of the separation hyperplane are called **support vectors**.



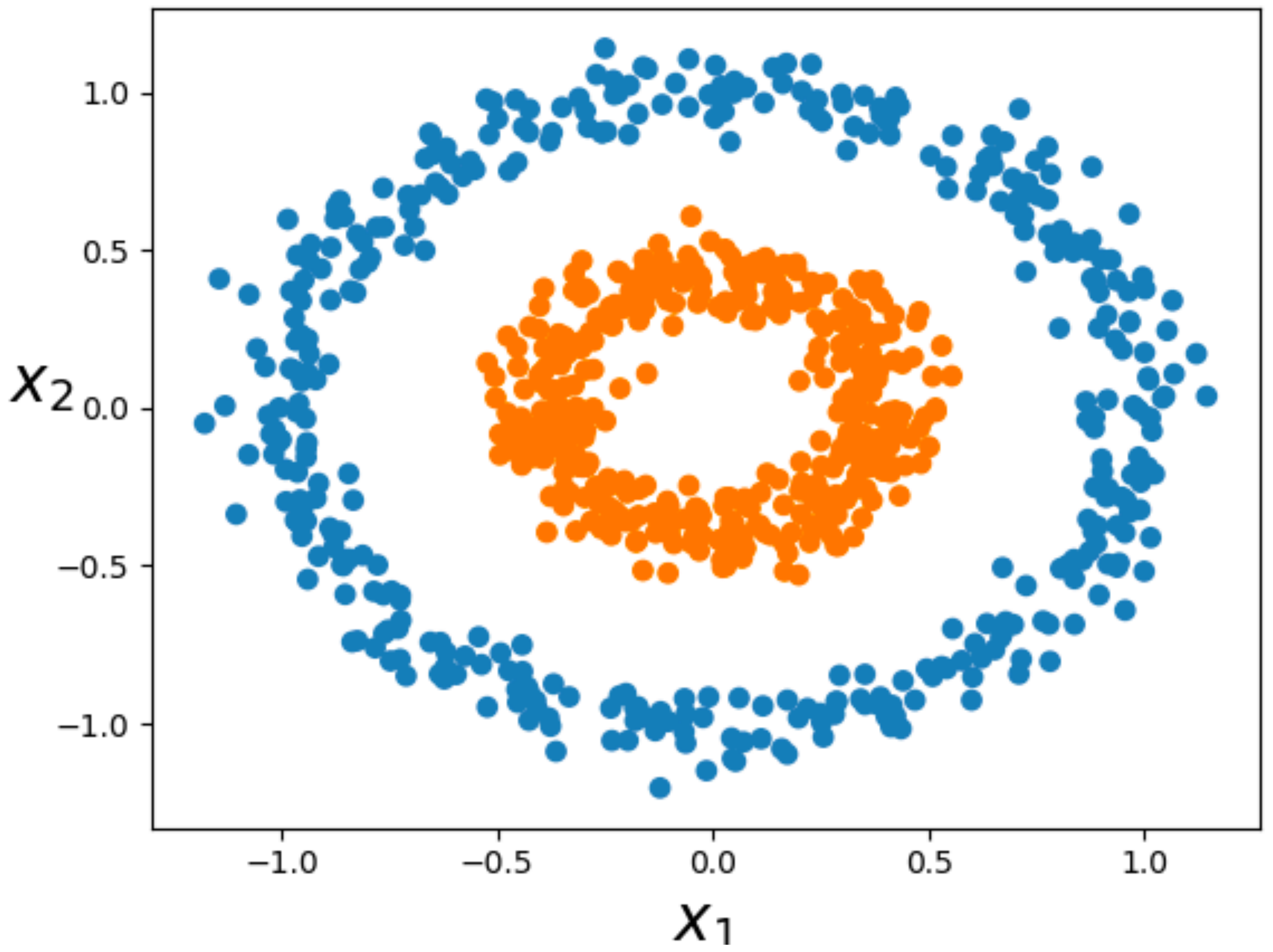
https://towardsdatascience.com/the-kernel-trick-c98cdbcaeb3f

LOYOLA UNIVERSITY CHICAGO

Support Vector Machines (SVM)

How to Separate these Features?

How to deal with non-linearly separable spaces?



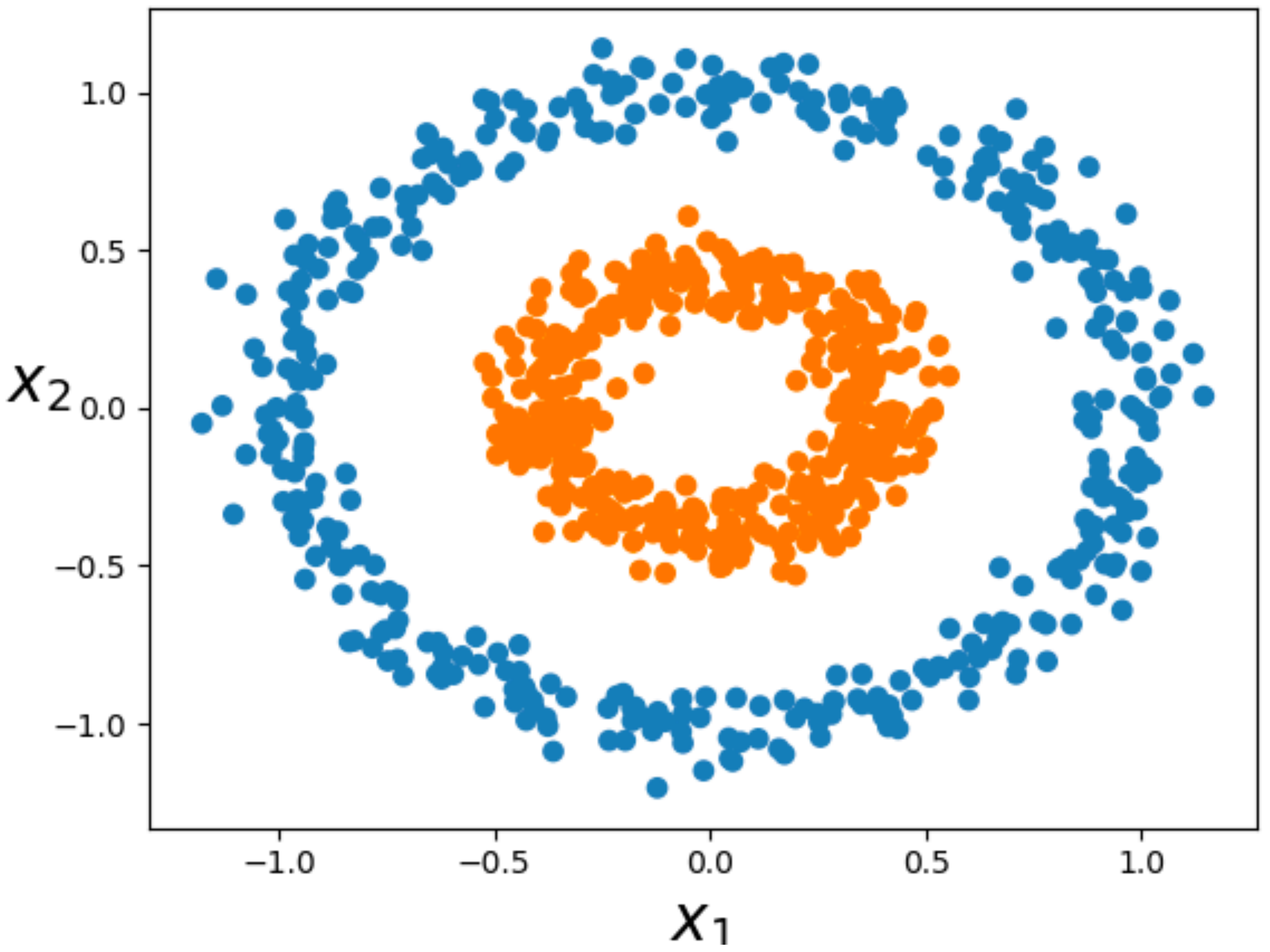
https://towardsdatascience.com/the-kernel-trick-c98cdbcaeb3f

Support Vector Machines (SVM)

How to Separate these Features?

How to deal with non-linearly separable spaces?

Solution: kernel trick.
Transform the data to higher-dimensional spaces where they are linearly separable.



https://towardsdatascience.com/the-kernel-trick-c98cdbcaeb3f

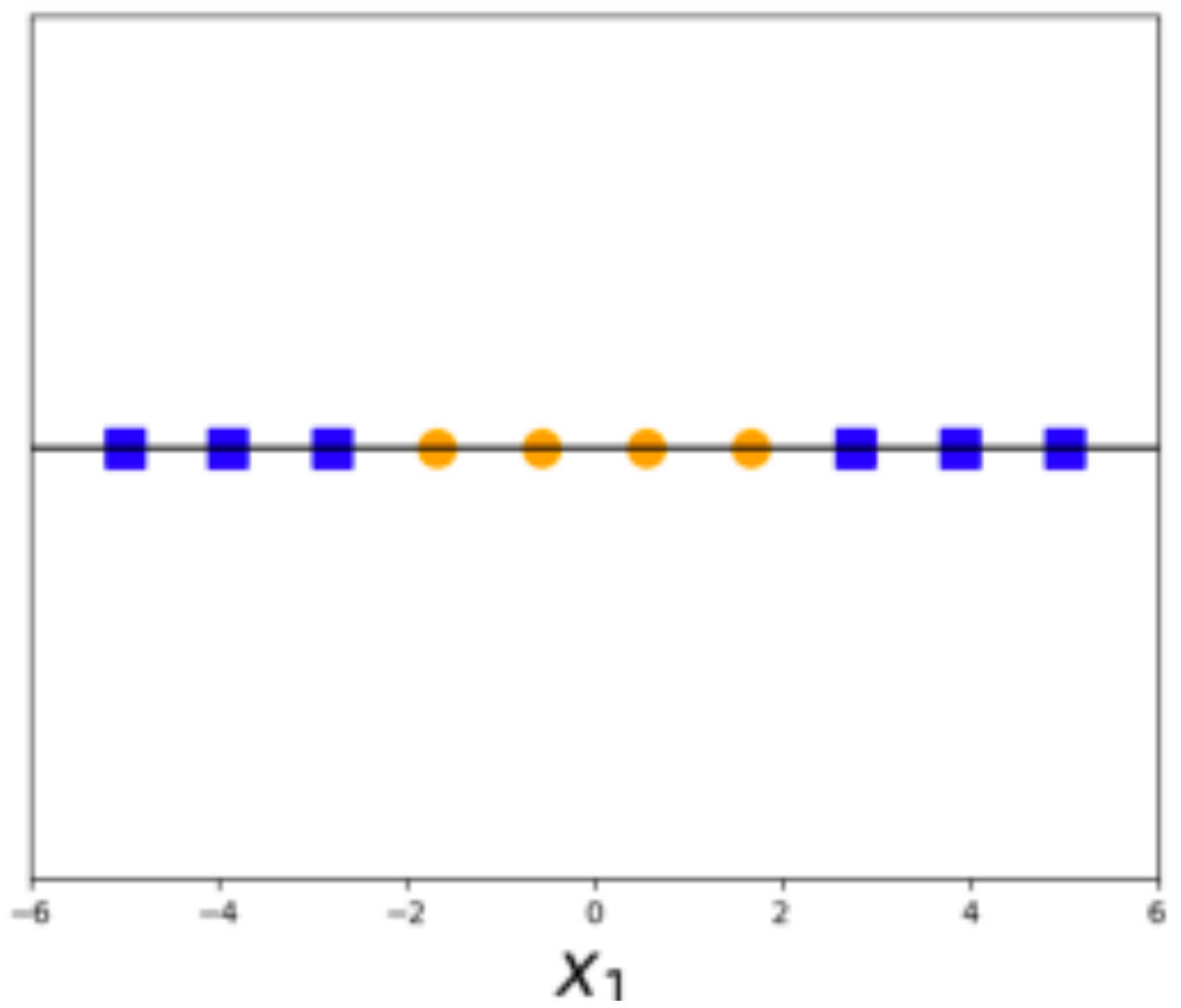
Use a kernel function for that (e.g., radial basis).

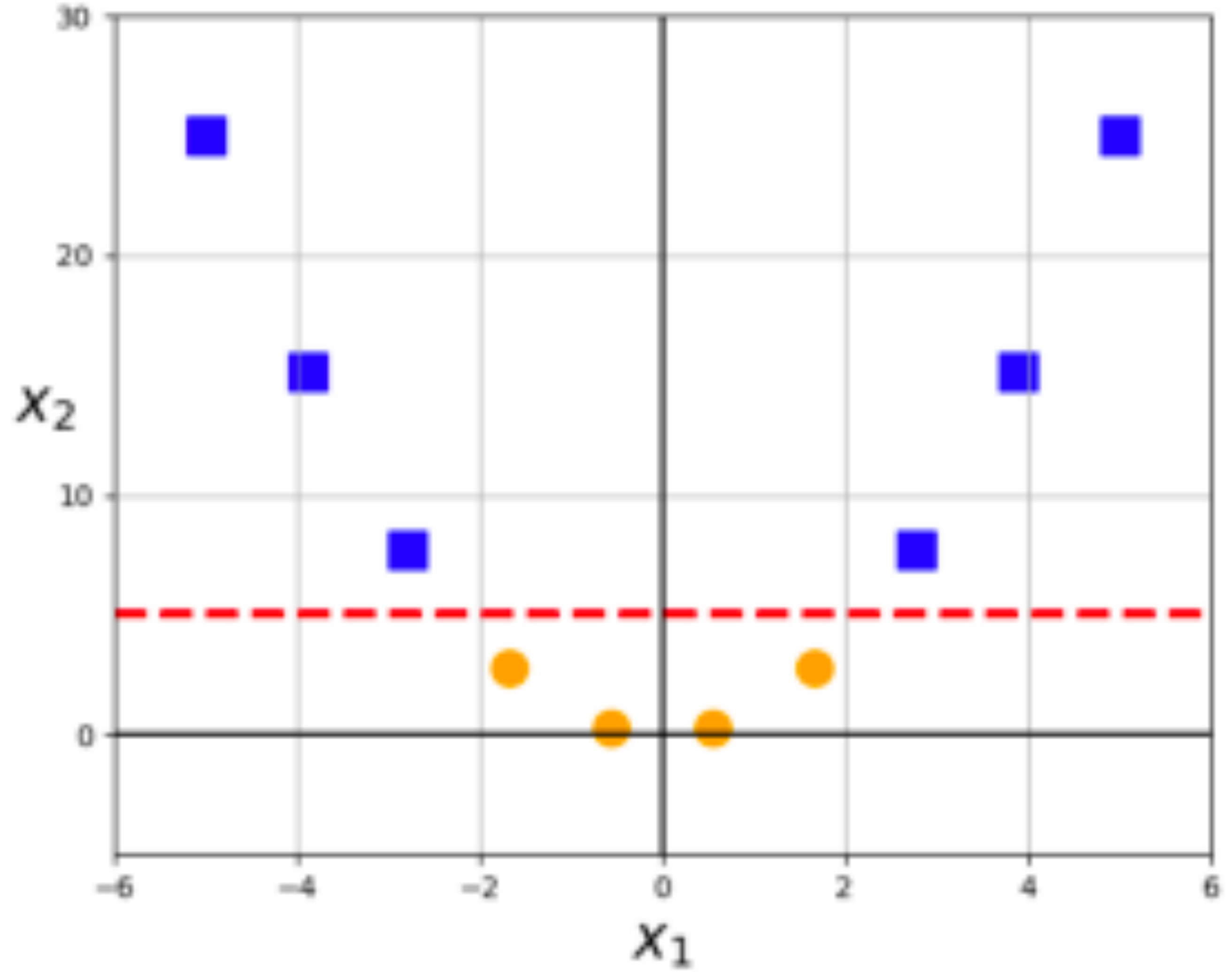
LOYOLA UNIVERSITY CHICAGO

Support Vector Machines (SVM)

How to Separate these Features?

Kernel-trick examples.





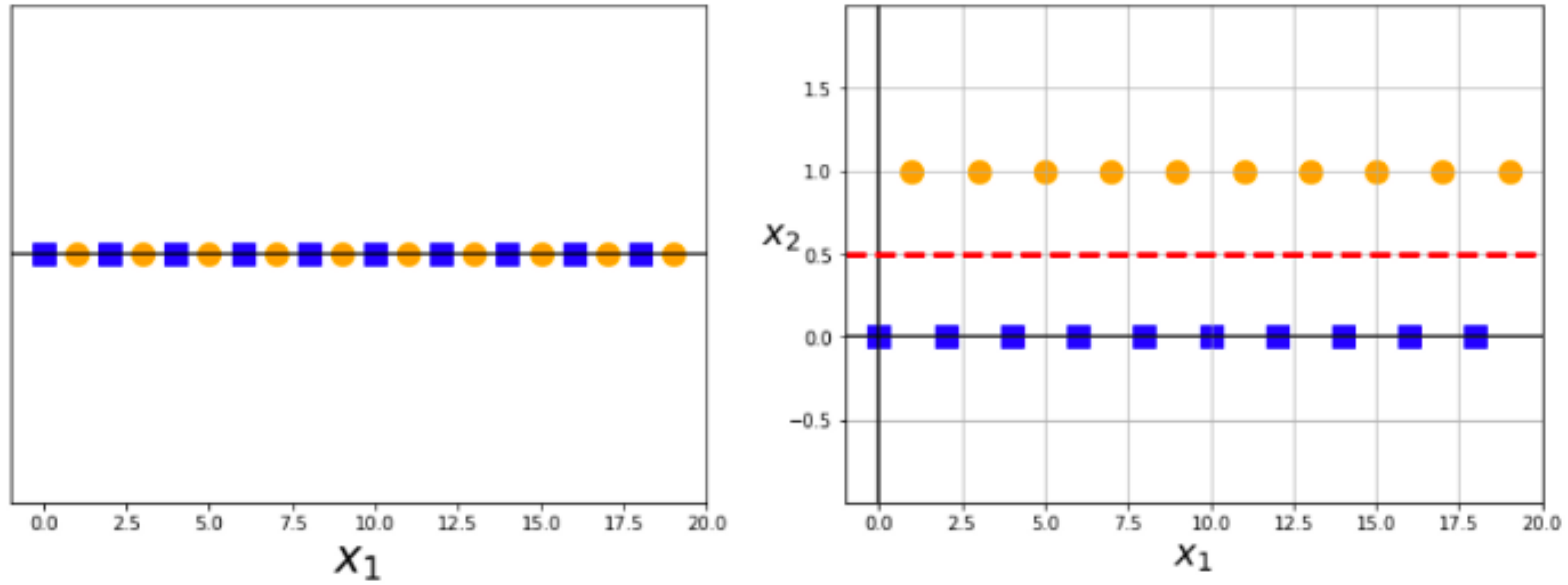
https://towardsdatascience.com/the-kernel-trick-c98cdbcaeb3f

LOYOLA UNIVERSITY CHICAGO

Support Vector Machines (SVM)

How to Separate these Features?

Kernel-trick examples.

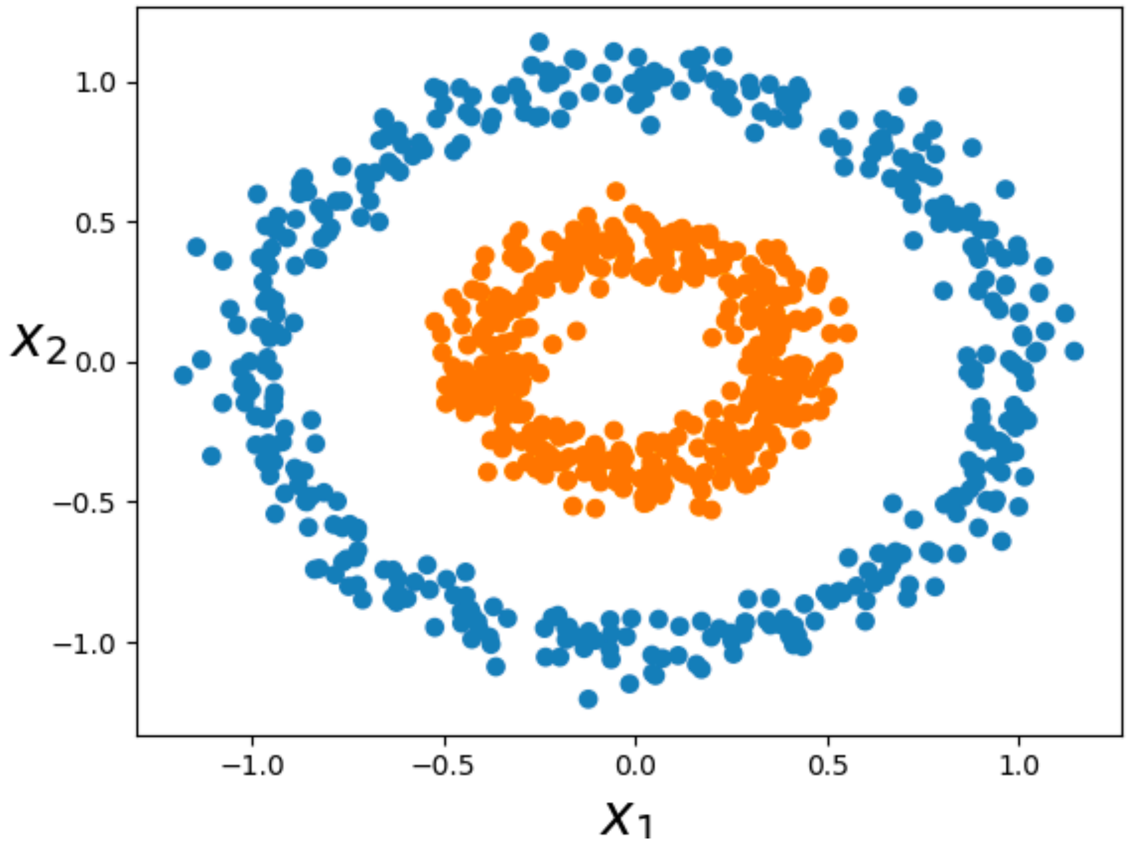


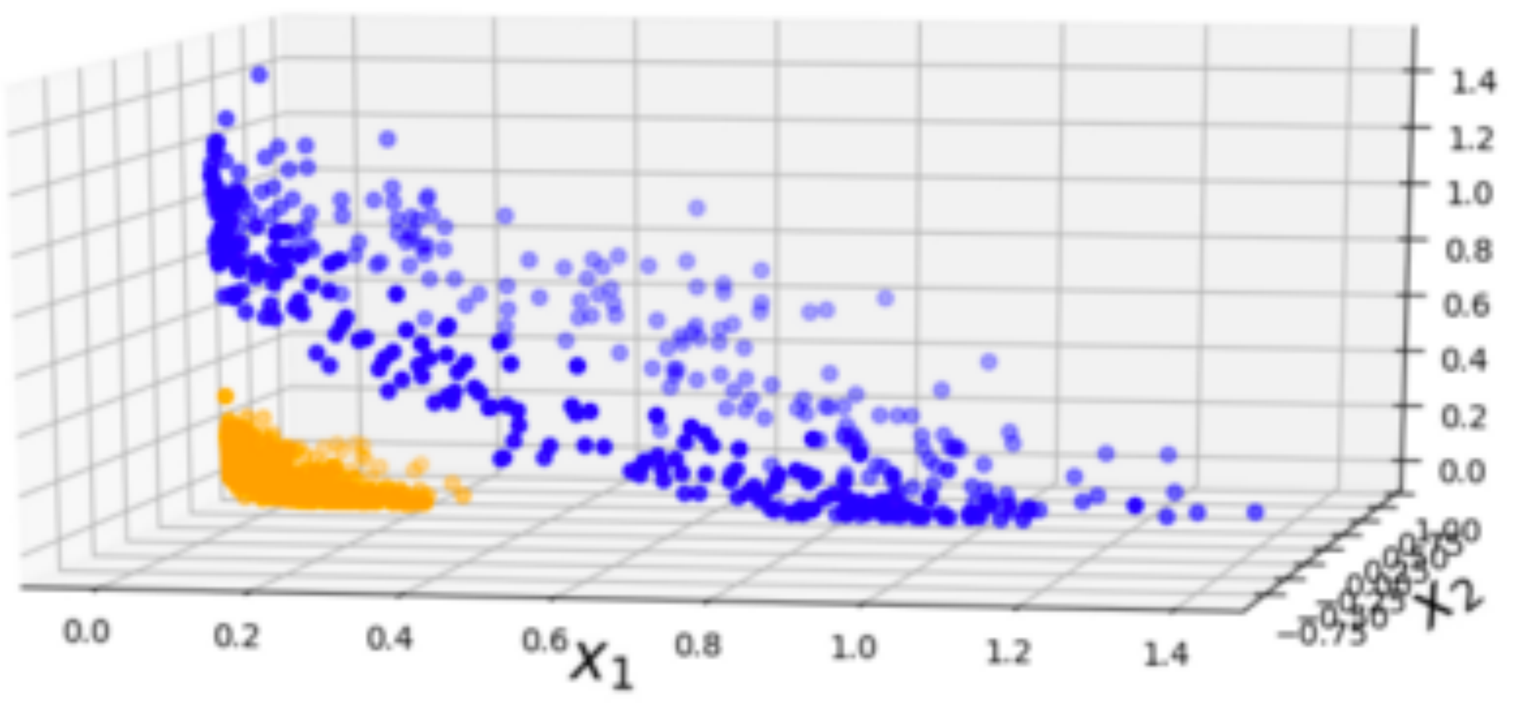
https://towardsdatascience.com/the-kernel-trick-c98cdbcaeb3f

Support Vector Machines (SVM)

How to Separate these Features?

Kernel-trick examples.



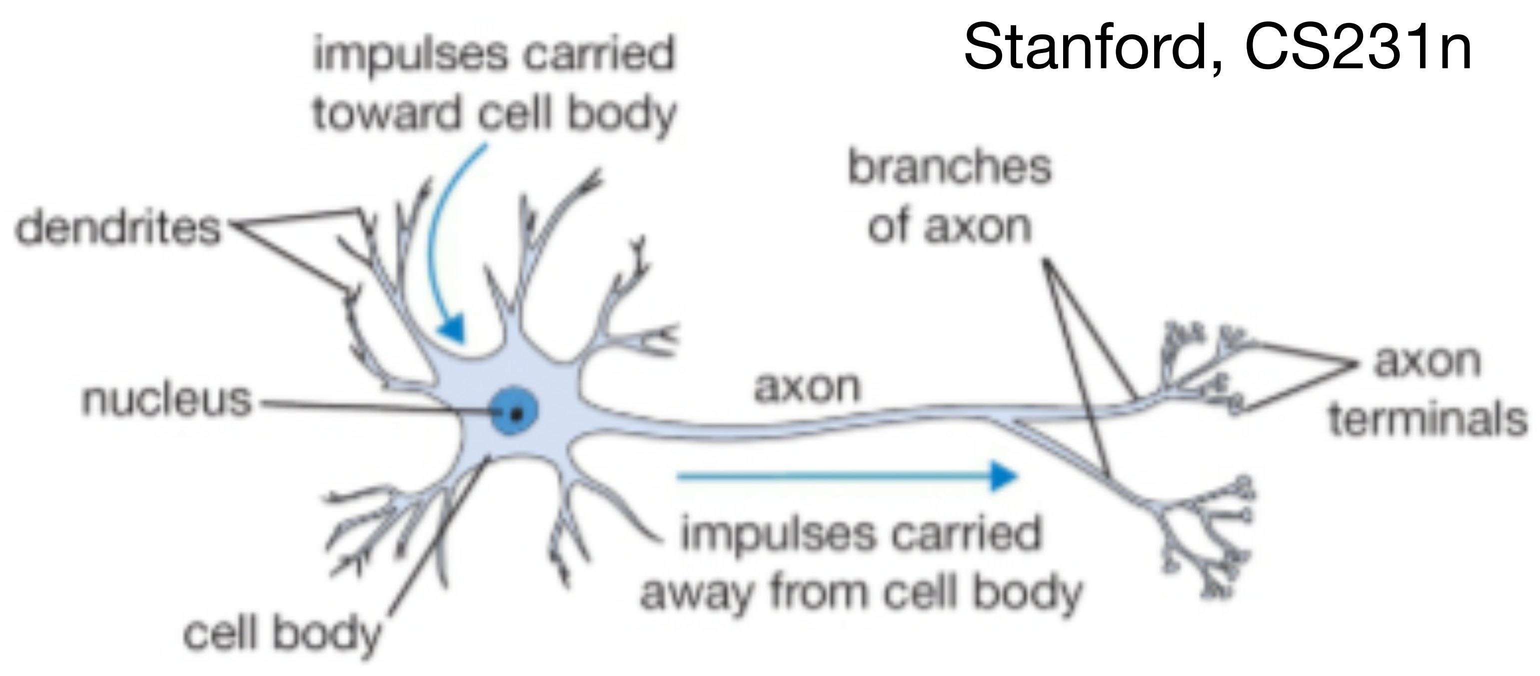


https://towardsdatascience.com/the-kernel-trick-c98cdbcaeb3f

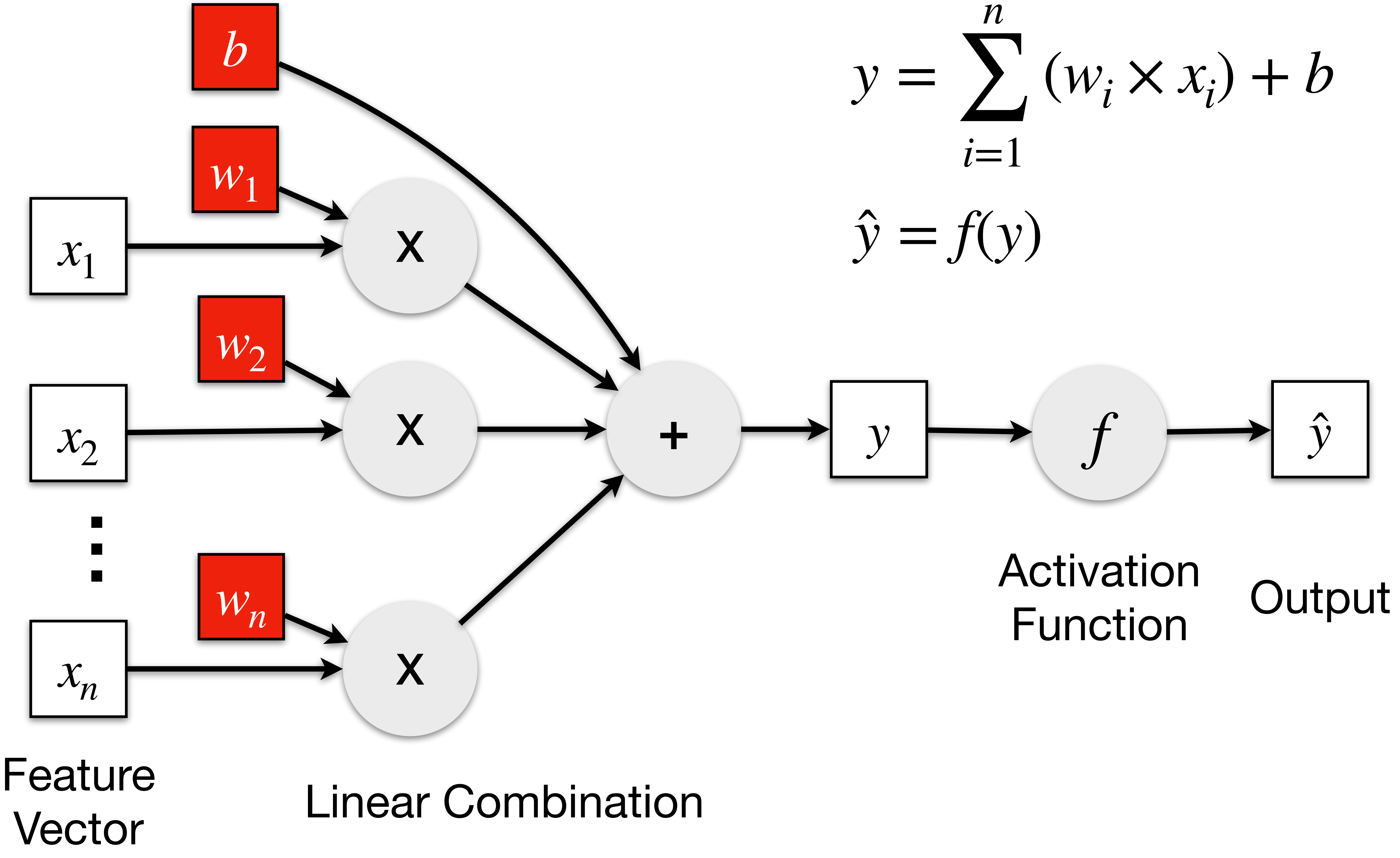
Implementation available at https://scikit-learn.org/stable/modules/svm.html

Neural Networks (NN)

Artificial Neuron
Building block of NNs.



Bioinspiration



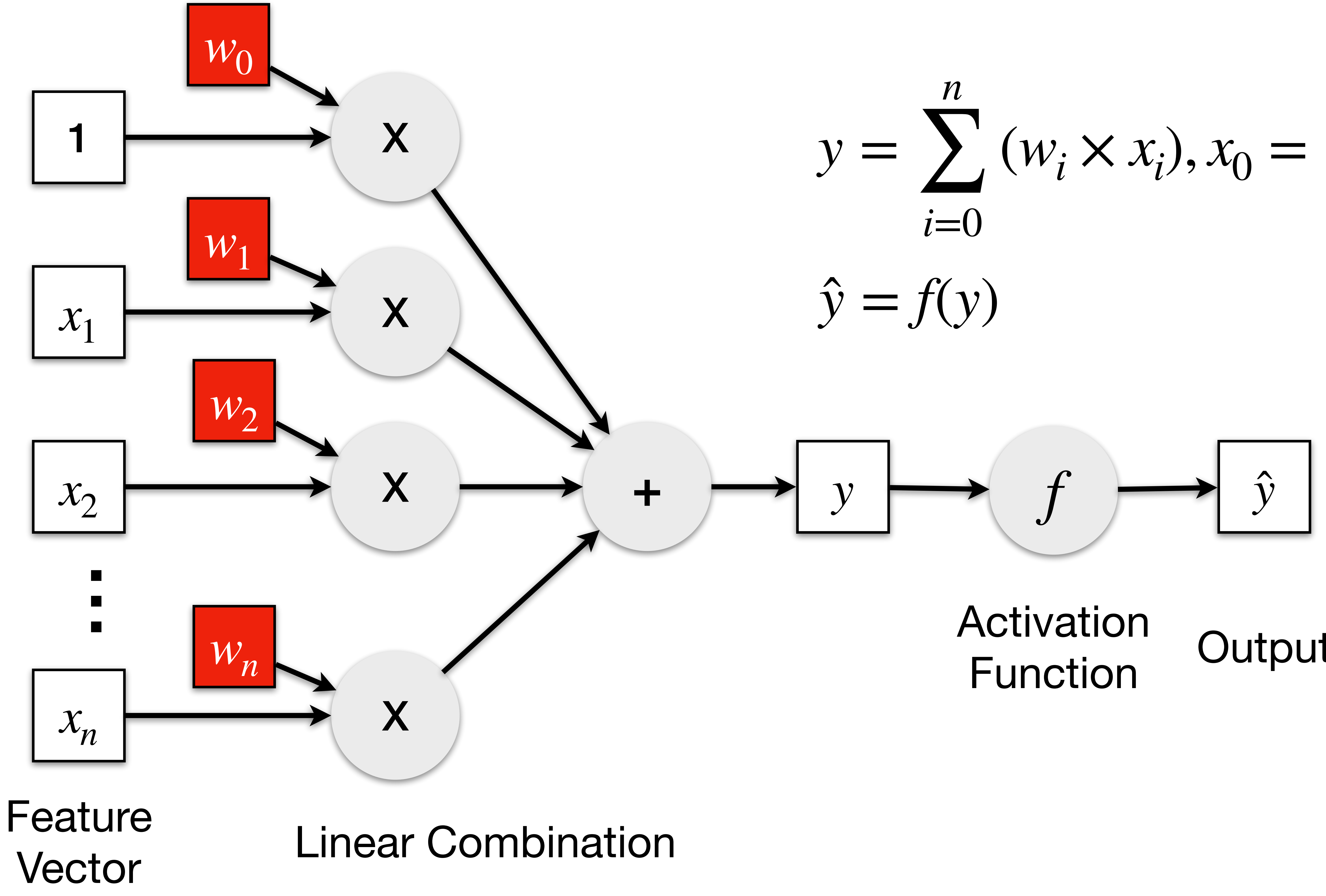
$$y = \sum_{i=1}^{n} (w_i \times x_i) + b$$

$$\hat{y} = f(y)$$

Neural Networks (NN)

Artificial Neuron
Building block of NNs.

Notation adjustments.

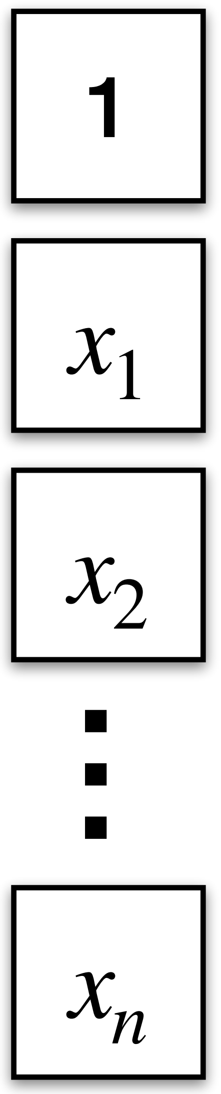


$$y = \sum_{i=0}^{n} (w_i \times x_i), x_0 = 1$$

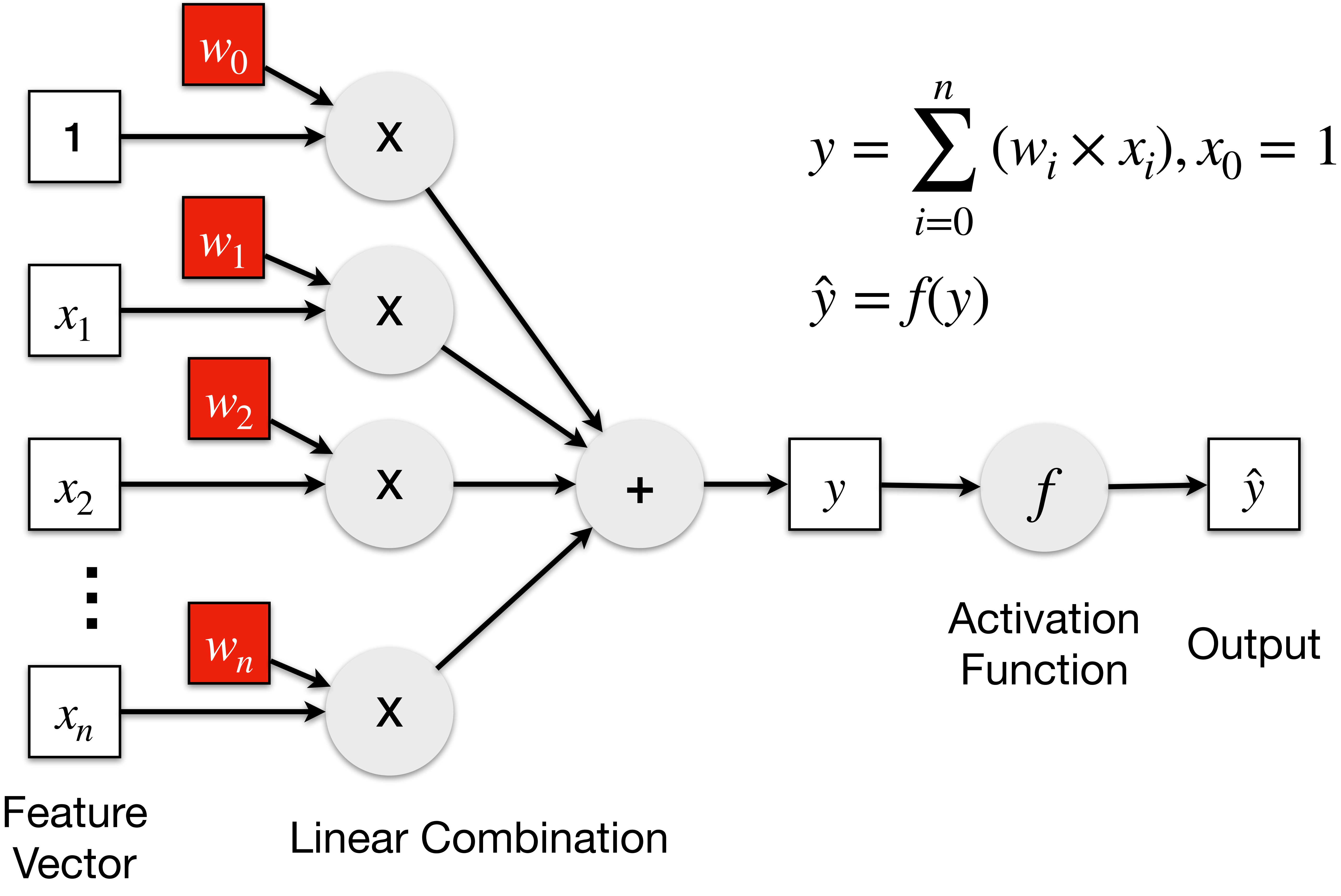
$$\hat{y} = f(y)$$

Neural Networks (NN)

Artificial Neuron
Building block of NNs.



Data sample $\vec{x}$ with expected (known) label z.



$$y = \sum_{i=0}^{n} (w_i \times x_i), x_0 = 1$$

$$\hat{y} = f(y)$$

LOYOLA UNIVERSITY CHICAGO

Neural Networks (NN)

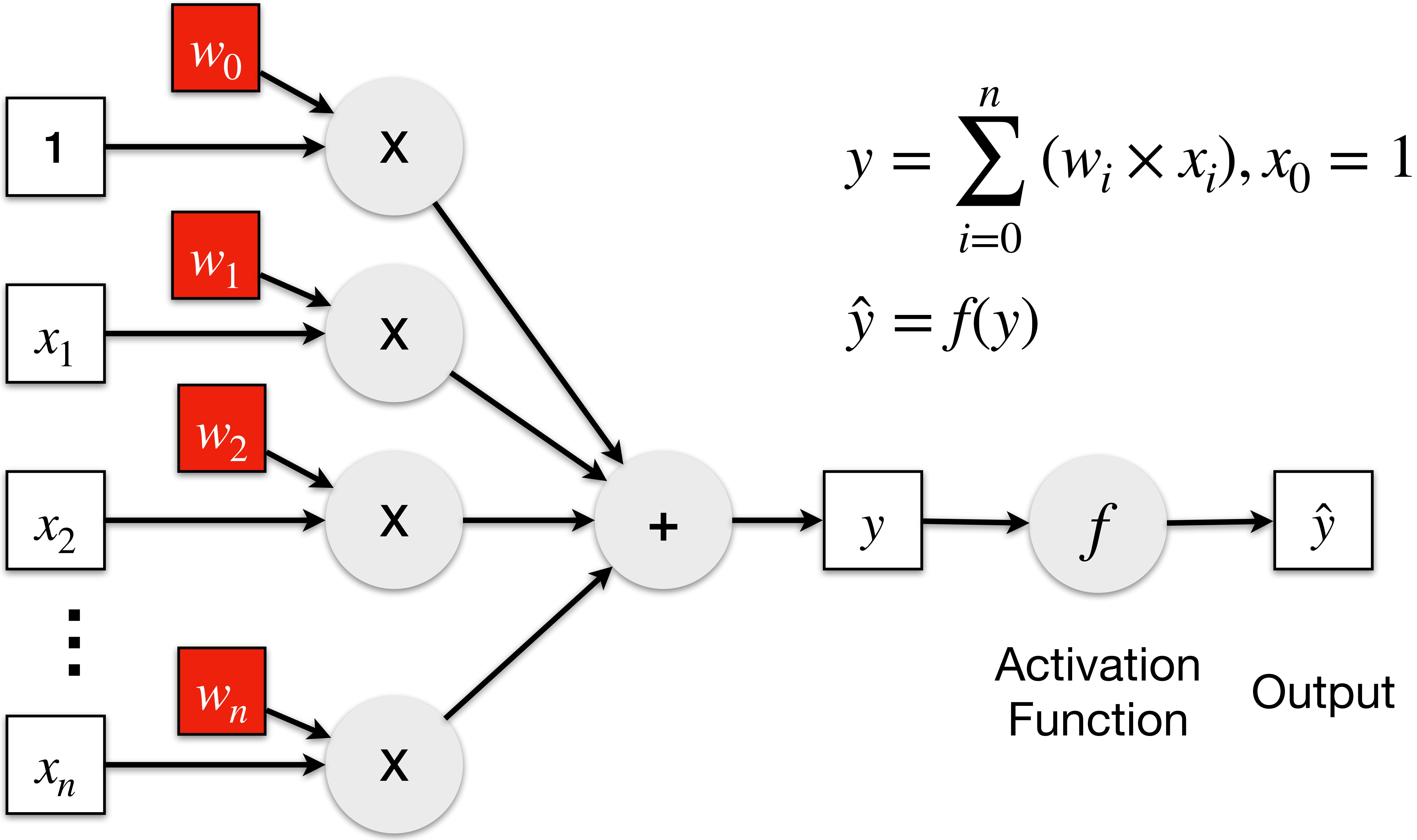
Artificial Neuron
Building block of NNs.

w_0, w_1, w_2, ⋮, w_n

Learnable weights $\vec{w}$ (a.k.a. neuron's parameters).

They may start with random values.

They may start with weights previously learned on other data (transfer learning).



$$y = \sum_{i=0}^{n} (w_i \times x_i), x_0 = 1$$

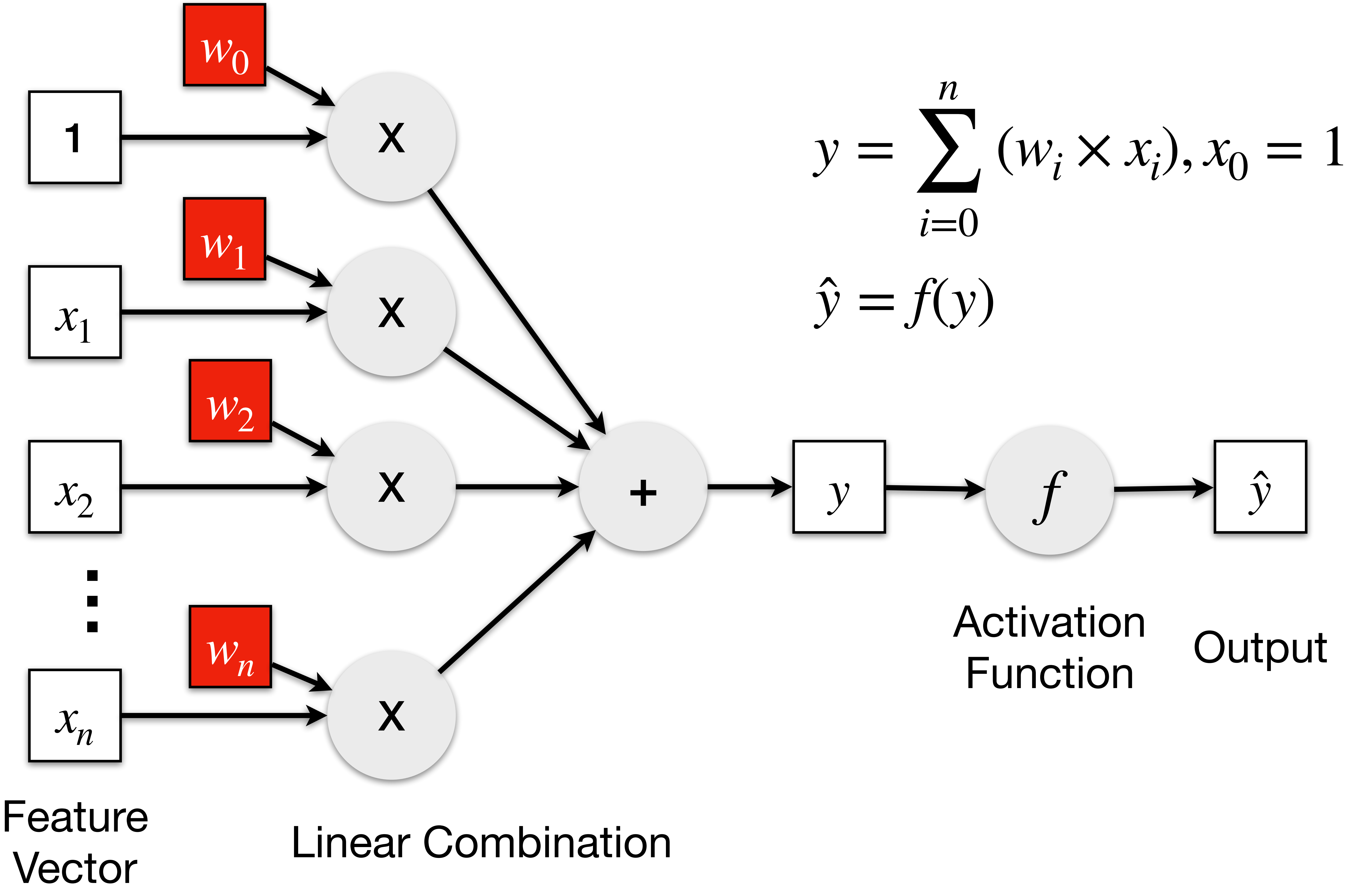
$$\hat{y} = f(y)$$

LOYOLA UNIVERSITY CHICAGO

Neural Networks (NN)

Artificial Neuron
Building block of NNs.

y Linear combination of feature vector's components and neuron's weights.



$$y = \sum_{i=0}^{n} (w_i \times x_i), x_0 = 1$$

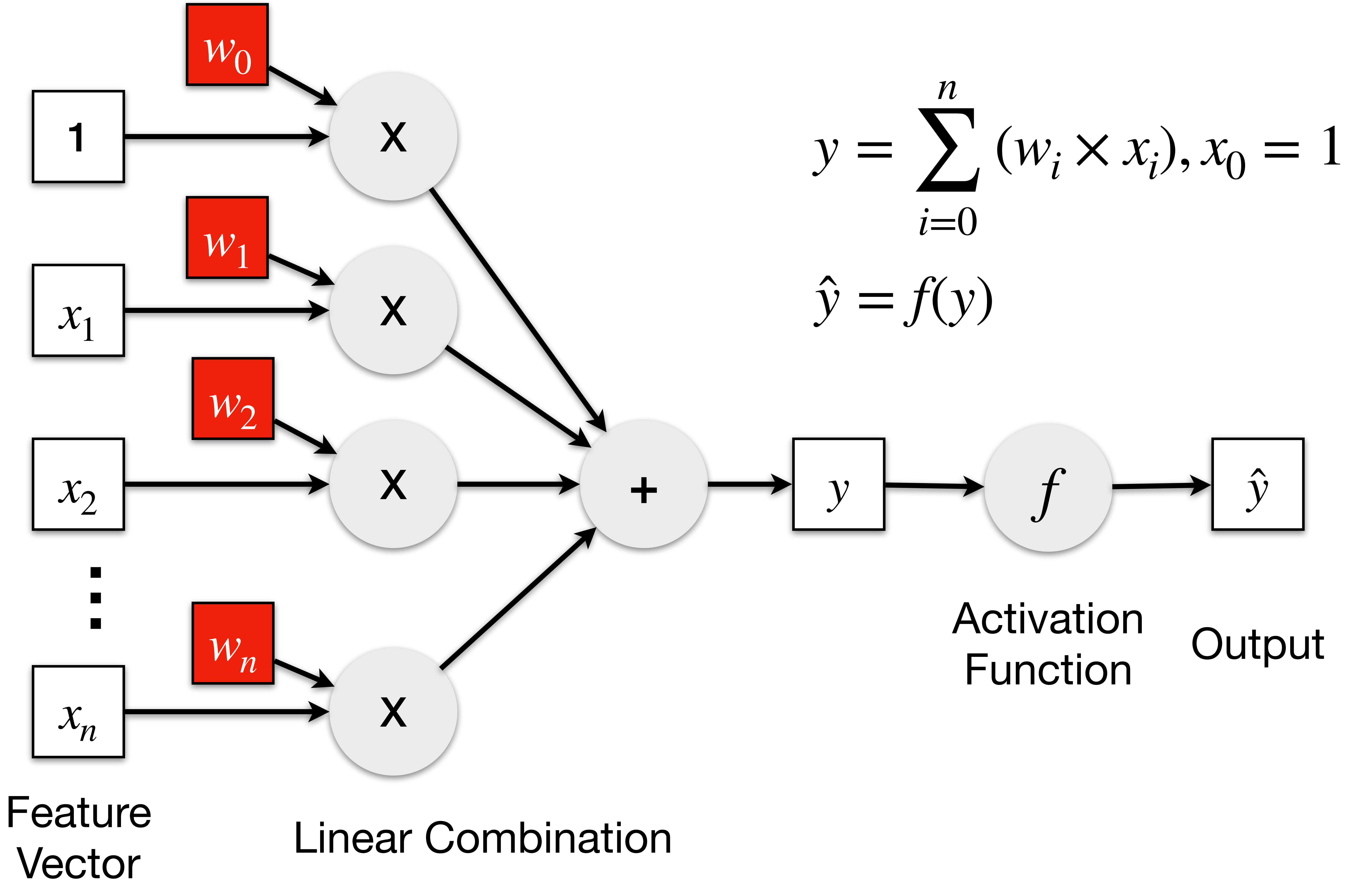
$$\hat{y} = f(y)$$

Neural Networks (NN)

Artificial Neuron
Building block of NNs.

f Activation Function

Ideally a non-linear differentiable function to add non-linearity to the model.



$$y = \sum_{i=0}^{n} (w_i \times x_i), x_0 = 1$$

$$\hat{y} = f(y)$$

LOYOLA UNIVERSITY CHICAGO

Neural Networks (NN)

Artificial Neuron

Building block of NNs.

f Activation Function

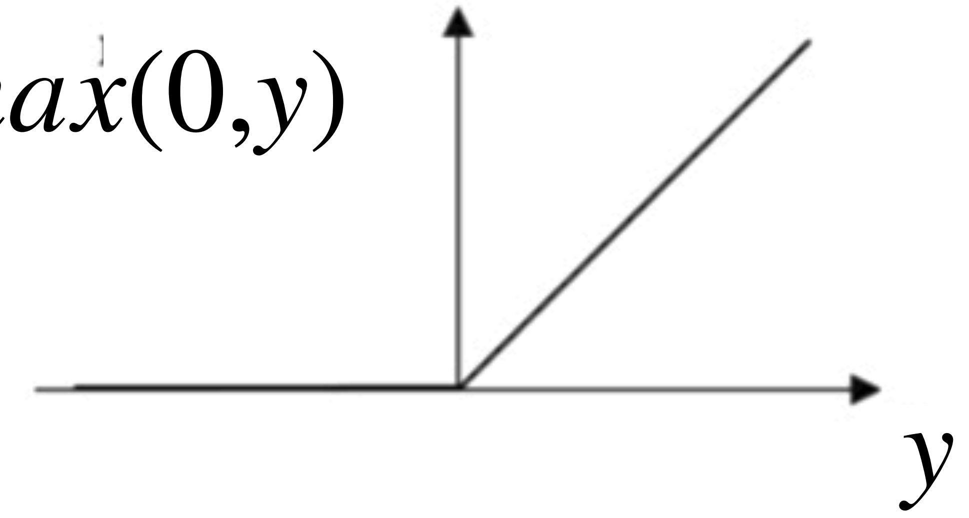
Ideally a non-linear
differentiable function
to add non-linearity
to the model.

Necessary to allow NNs
to learn non-linear functions.

Examples

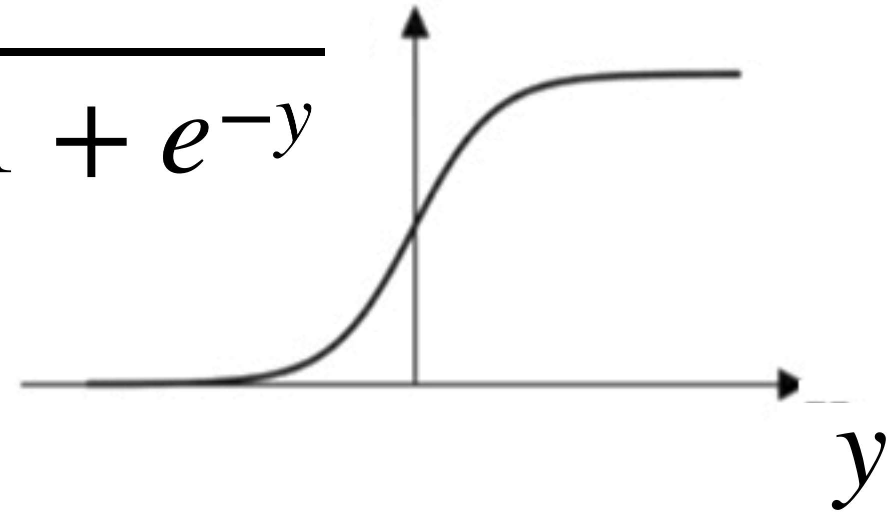
ReLU

$max(0,y)$



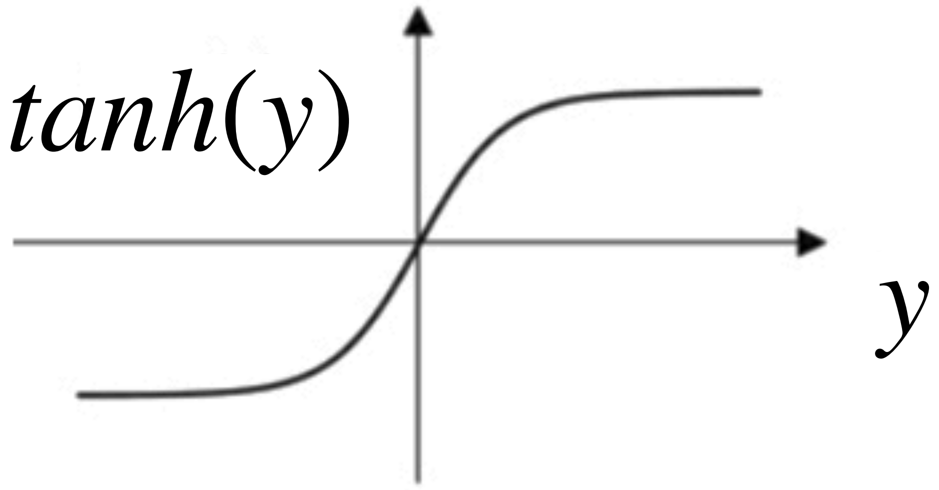
Sigmoid

$$\frac{1}{1+e^{-y}}$$



Tanh

$tanh(y)$



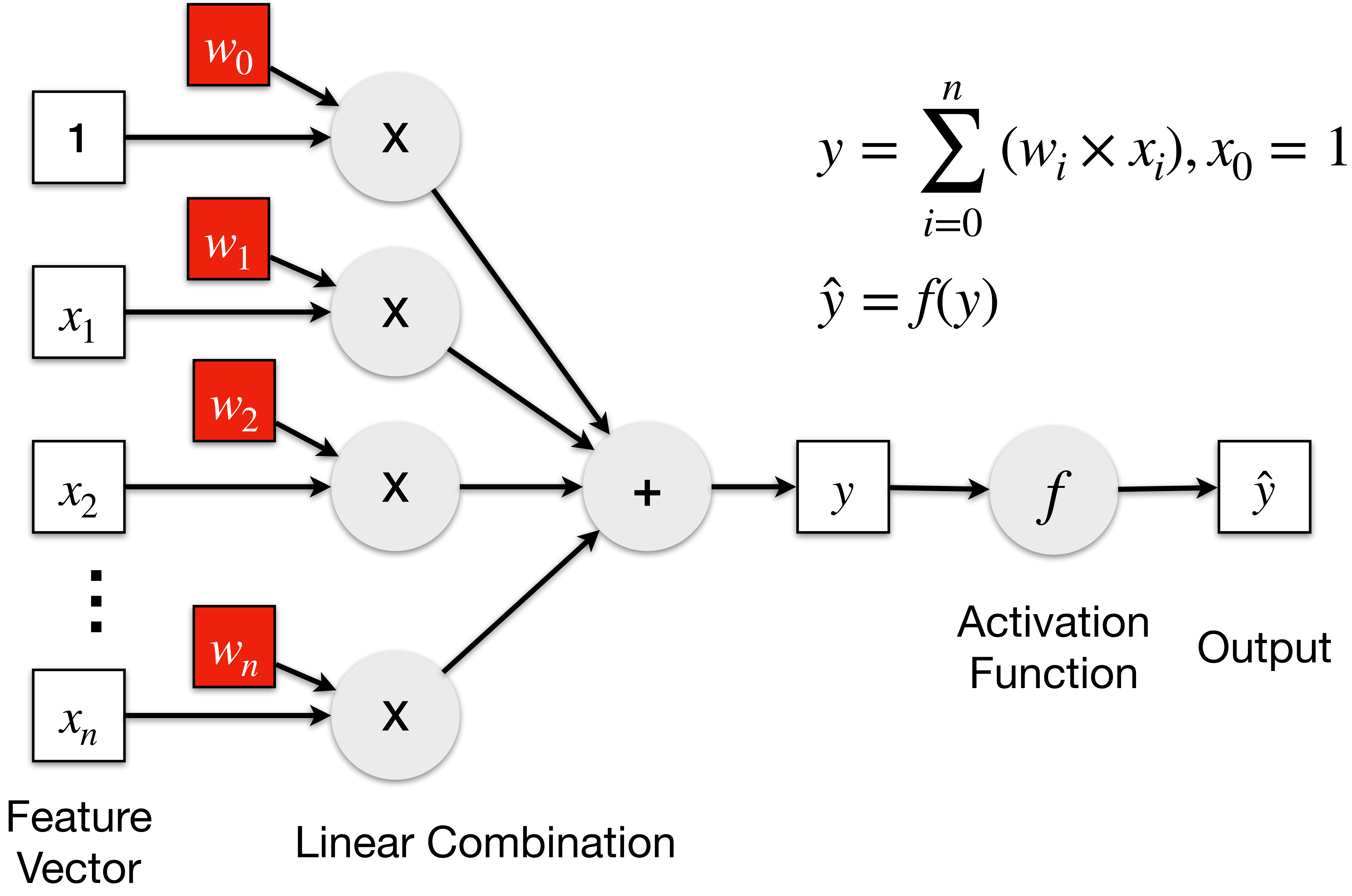
Neural Networks (NN)

Artificial Neuron
Building block of NNs.

Delta Rule
Supervised learning process

Input: $\overrightarrow{x_1}, \overrightarrow{x_2}, \ldots, \overrightarrow{x_m}, m$ samples
Output: $z_1, z_2, \ldots, z_m, m$ labels

$$Loss(\overrightarrow{w}) = \sum_{k=1}^{m} (z_k - f(\overrightarrow{x}_k^T \cdot \overrightarrow{w}))^2$$



$$y = \sum_{i=0}^{n} (w_i \times x_i), x_0 = 1$$

$$\hat{y} = f(y)$$

LOYOLA UNIVERSITY CHICAGO

Neural Networks (NN)

Artificial Neuron

Building block of NNs.

$$y = \sum_{i=0}^{n} (w_i \times x_i), x_0 = 1$$

$$\hat{y} = f(y)$$

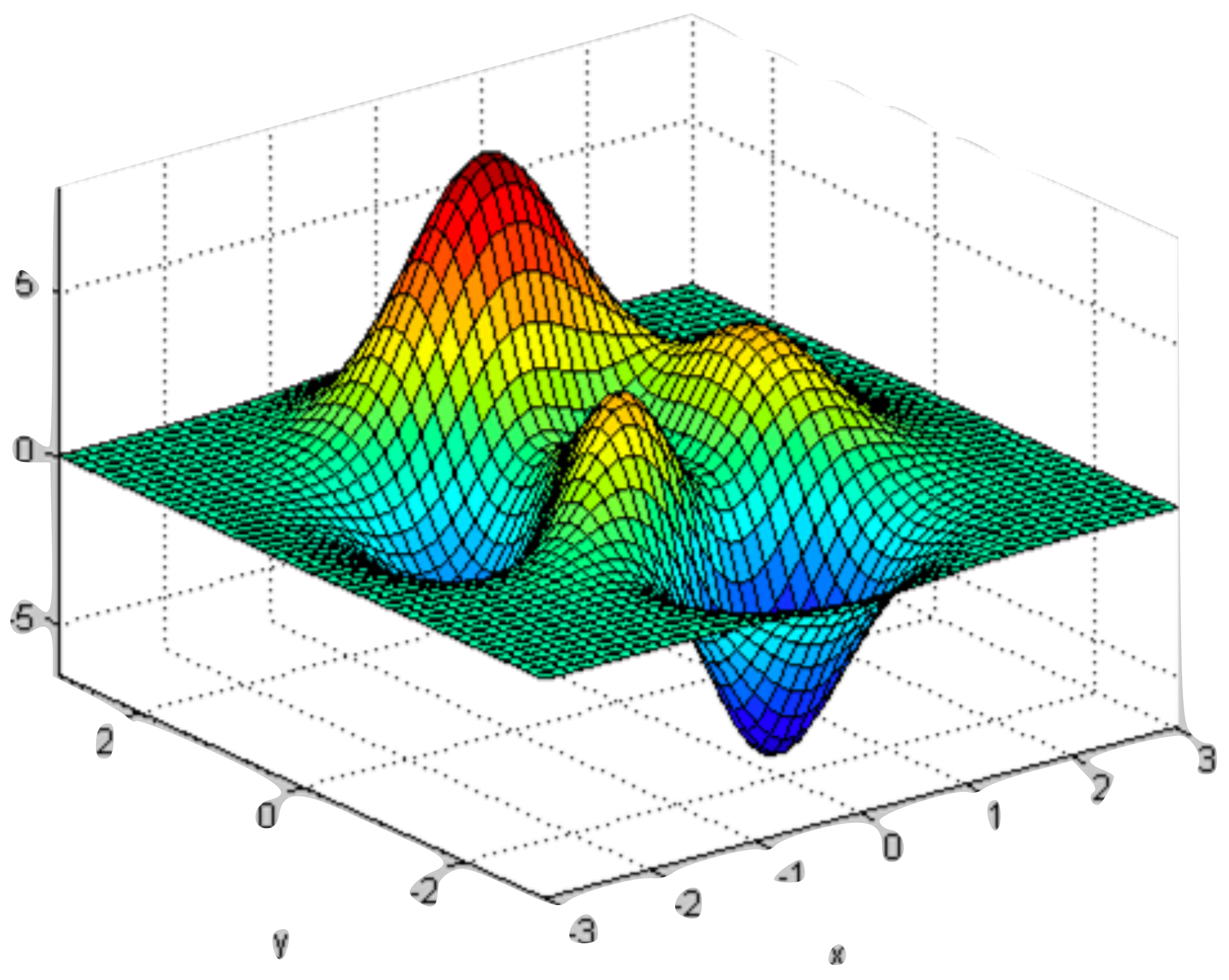
Delta Rule

Supervised learning process

Input: $\overrightarrow{x_1}, \overrightarrow{x_2}, \ldots, \overrightarrow{x_m}, m$ samples

Output: $z_1, z_2, \ldots, z_m, m$ labels

$$Loss(\overrightarrow{w}) = \sum_{k=1}^{m} (z_k - f(\overrightarrow{x}_k^T \cdot \overrightarrow{w}))^2$$



Loss Surface

Partial derivative:

$$\frac{\delta Loss(\overrightarrow{w})}{\delta w_i} = -\sum_{k=1}^{m} 2(z_k - f(\overrightarrow{x}_k^T \cdot \overrightarrow{w})) \times x_{ki} \times f'(\overrightarrow{x}_k^T \cdot \overrightarrow{w}) = 0$$

Neural Networks (NN)

Artificial Neuron
Building block of NNs.

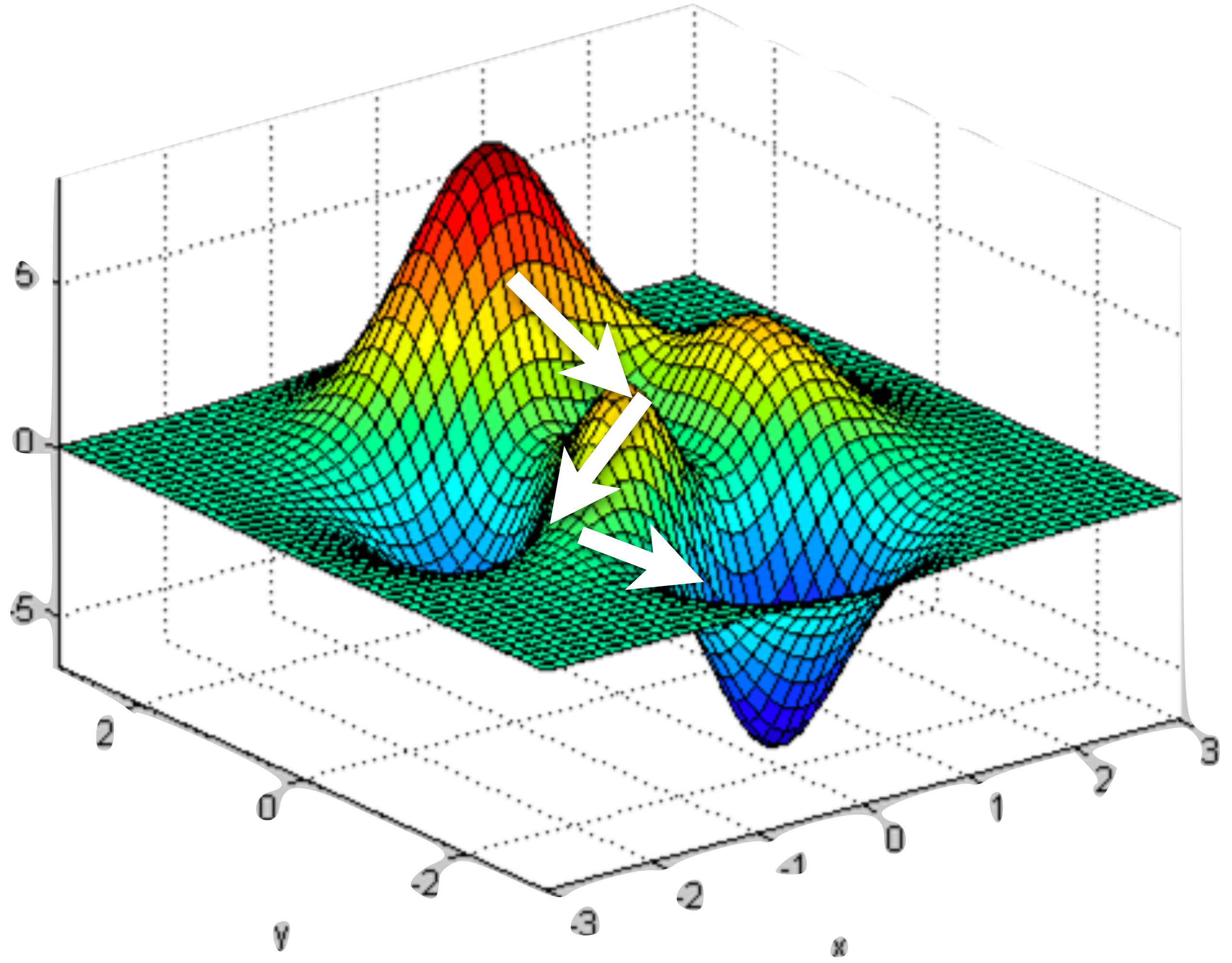
$$y = \sum_{i=0}^{n} (w_i \times x_i), x_0 = 1$$

$$\hat{y} = f(y)$$

Delta Rule
Supervised learning process

Input: $\vec{x_1}, \vec{x_2}, \ldots, \vec{x_m}$, m samples
Output: $z_1, z_2, \ldots, z_m$, m labels

$$Loss(\vec{w}) = \sum_{k=1}^{m} (z_k - f(\vec{x}_k^T \cdot \vec{w}))^2$$



Loss Surface

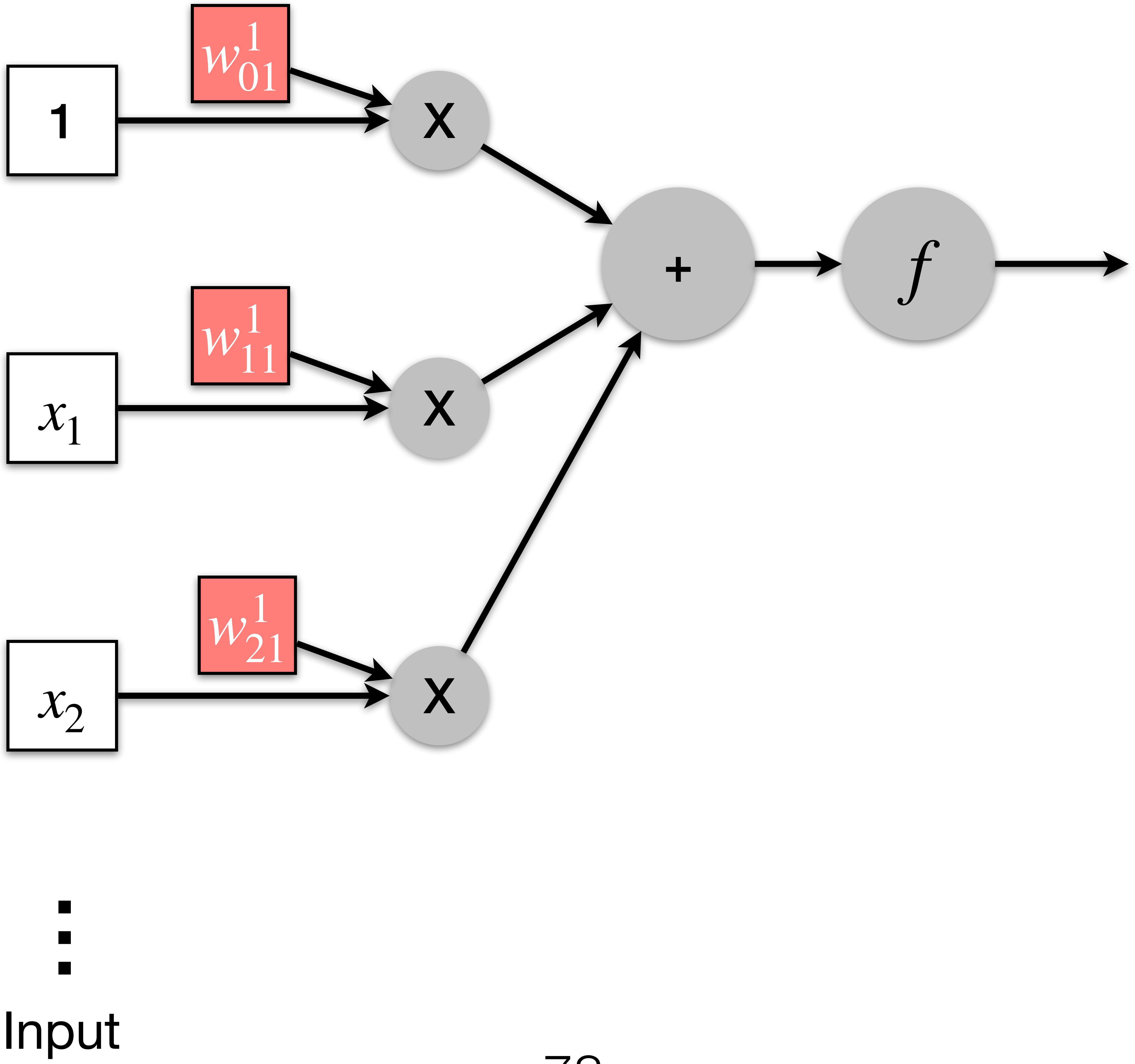
$$\Delta w_i = \sum_{k=1}^{m} \alpha(z_k - f(\vec{x}_k^T \cdot \vec{w})) \times x_{ki} \times f'(\vec{x}_k^T \cdot \vec{w}) = 0,$$ α is step size, one Δ for each weight.

LOYOLA UNIVERSITY CHICAGO

Neural Networks (NN)

Adding Hidden Layers

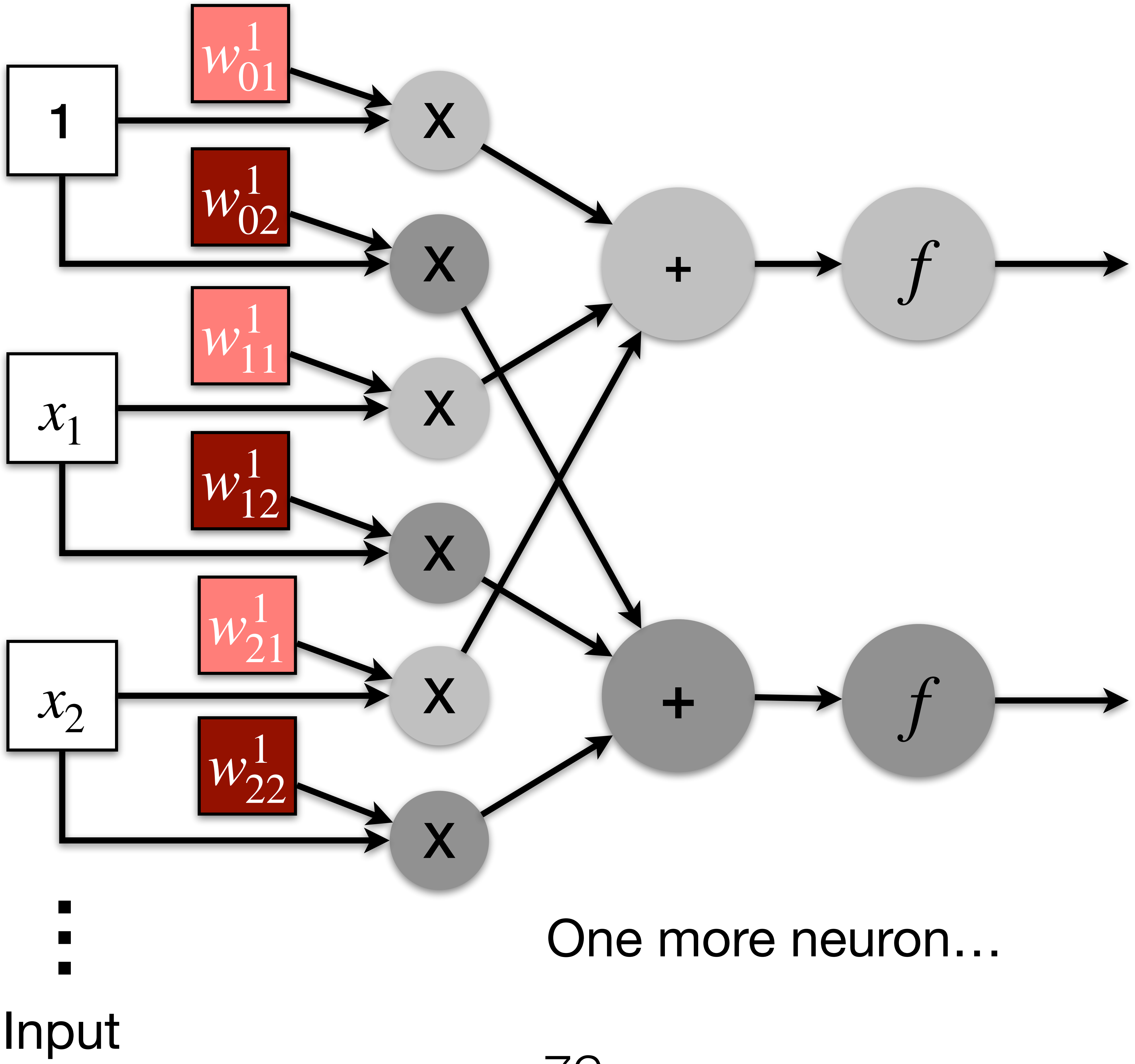
To increase computing power.



Neural Networks (NN)

Adding Hidden Layers

To increase computing power.

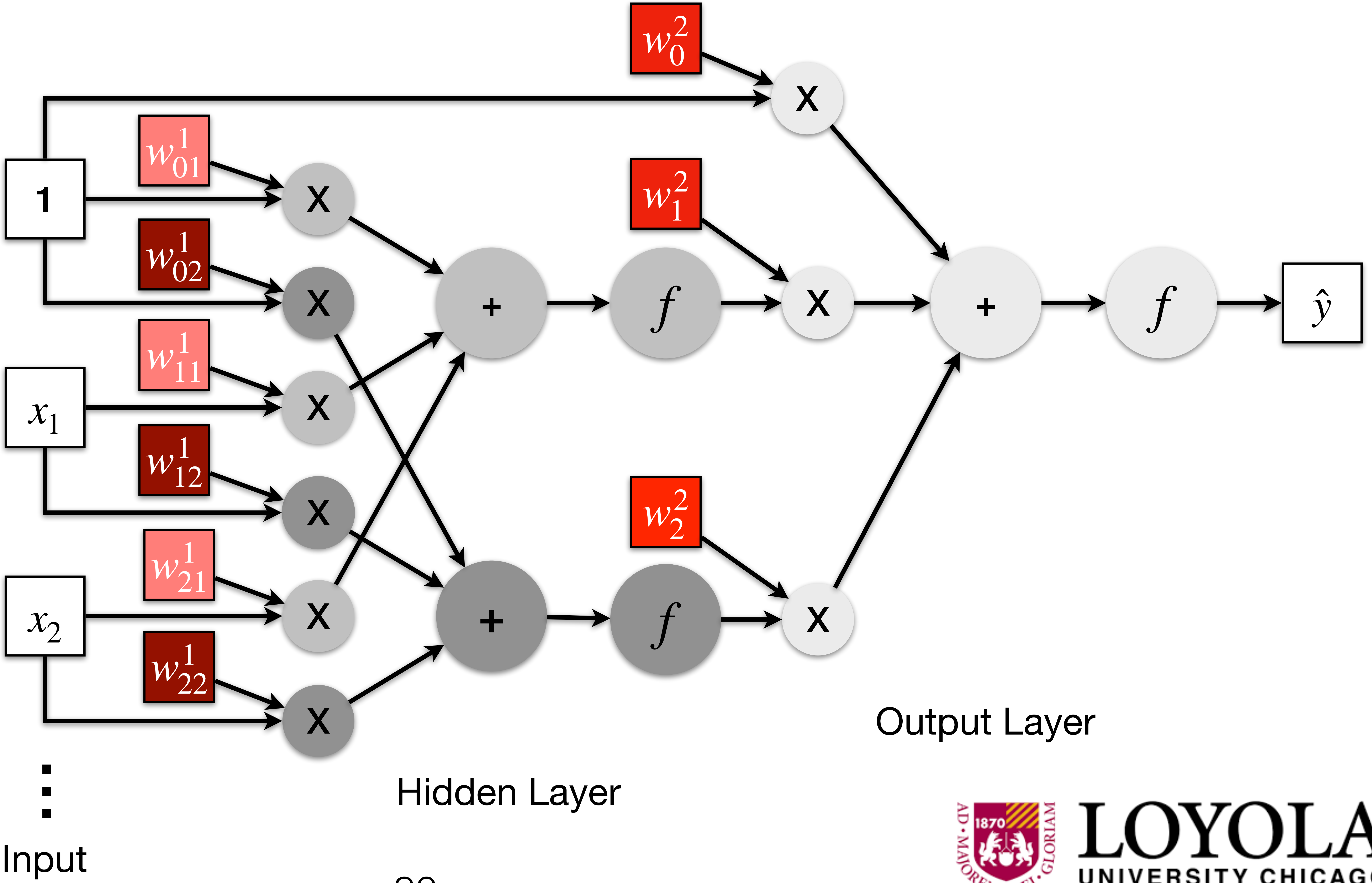


One more neuron…

Neural Networks (NN)

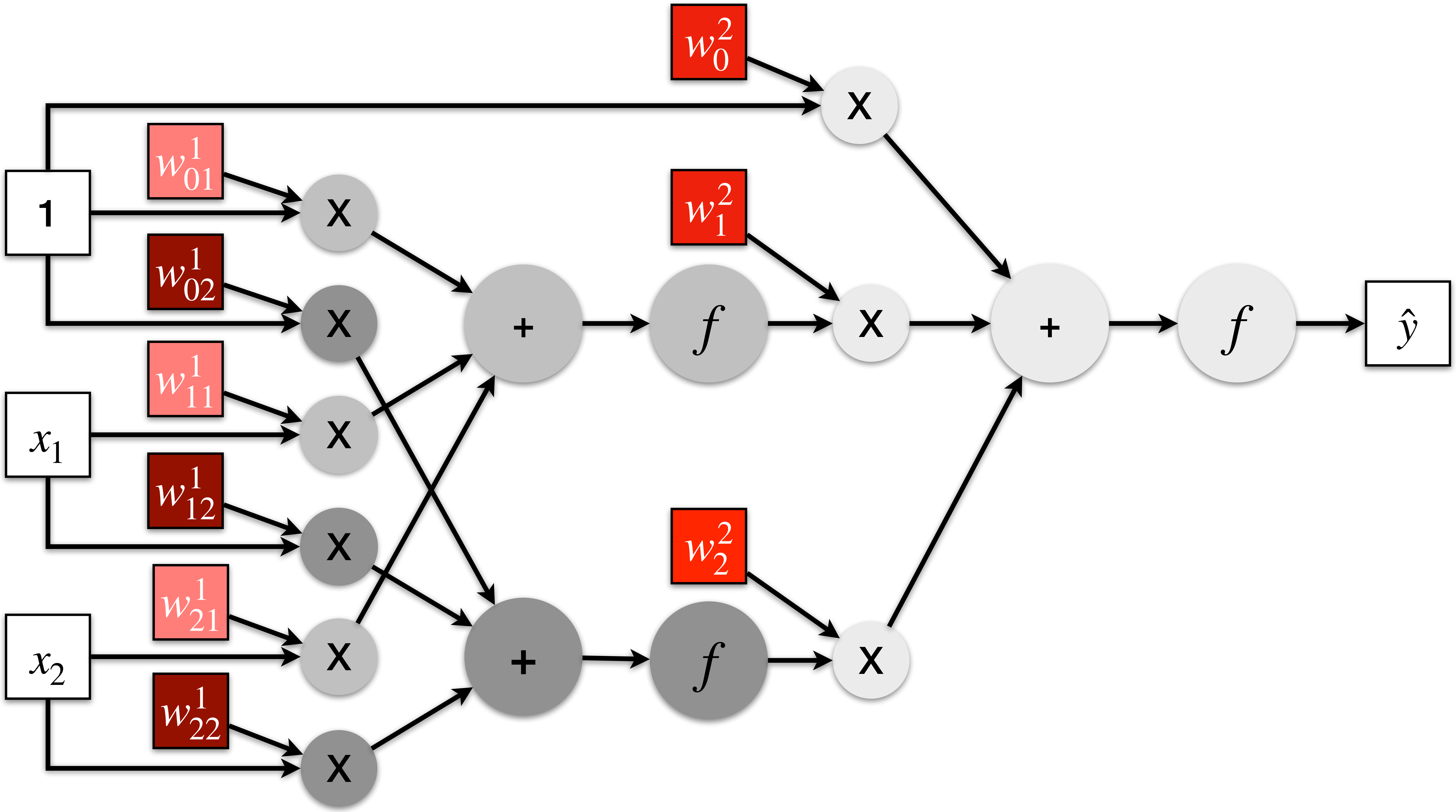
Adding Hidden Layers

To increase computing power.



Adding Hidden Layers

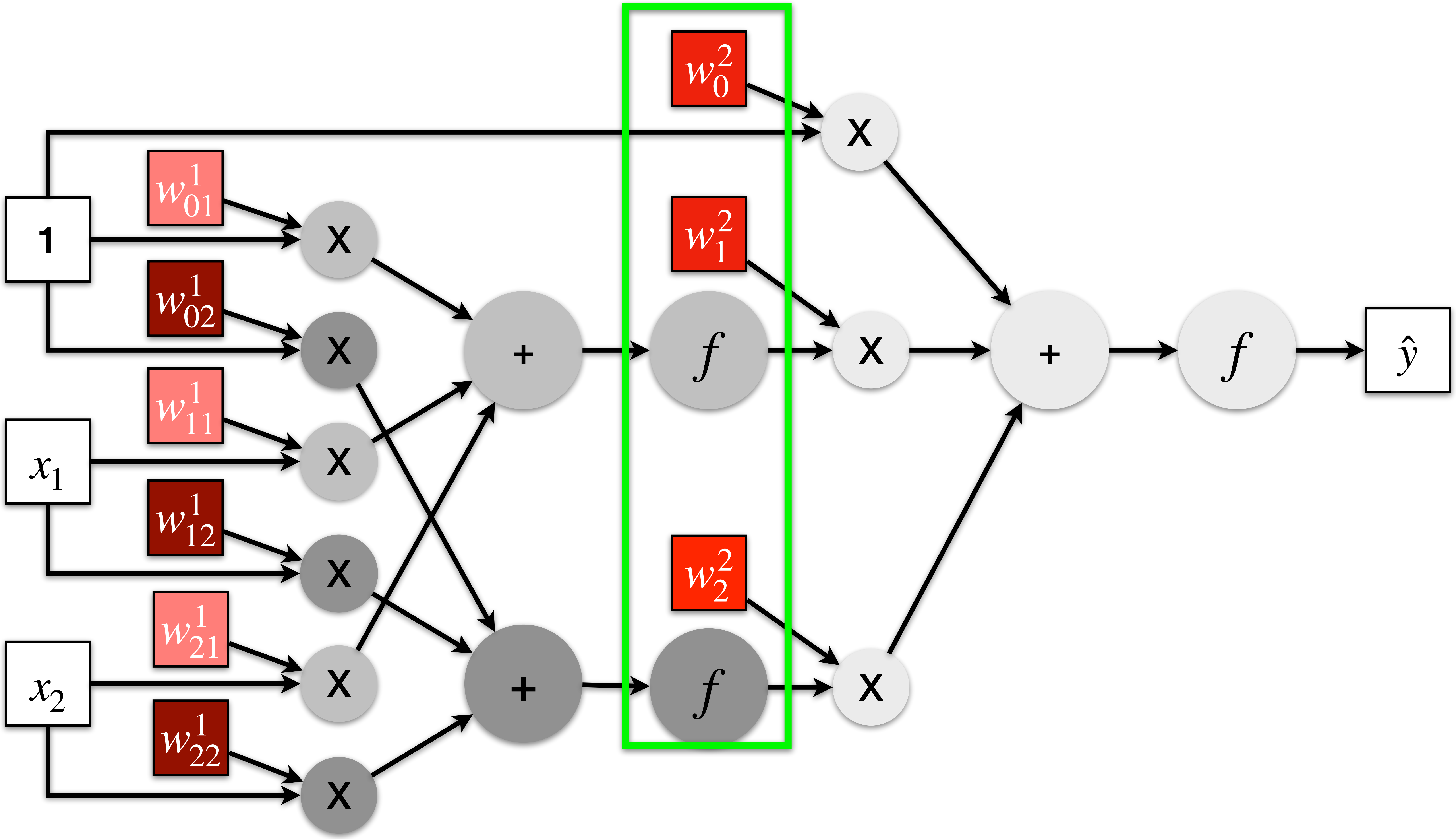
To increase computing power.



$$\hat{y} = f\left(\begin{pmatrix} w^2_1 & w^2_2 \end{pmatrix} f\left(\begin{pmatrix} w^1_{11} & w^1_{21} \\ w^1_{12} & w^1_{22} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} w^1_{01} \\ x^1_{02} \end{pmatrix} \right) + w^2_0 \right)$$

Adding Hidden Layers

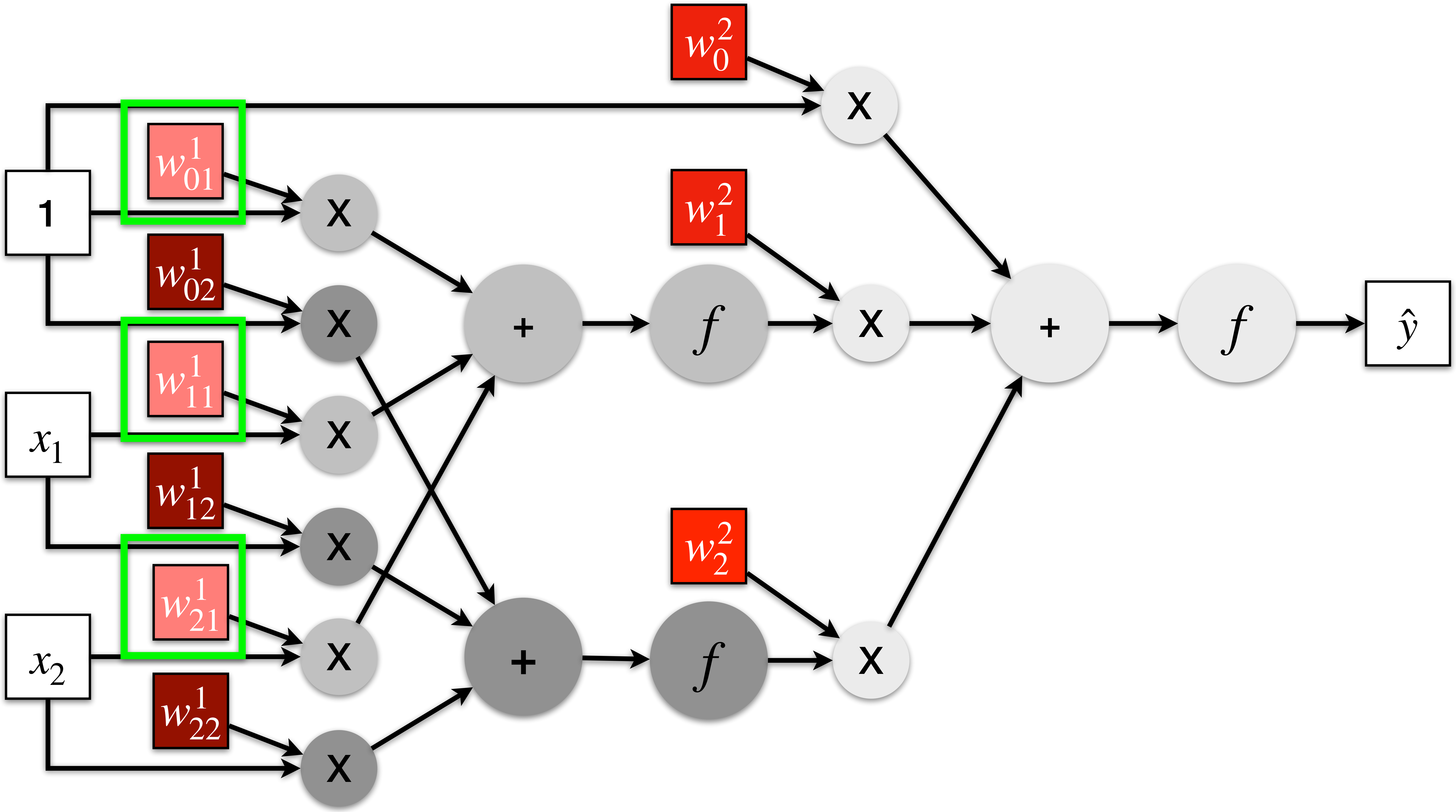
To increase computing power.



$$\hat{y} = f\left(\boxed{\begin{pmatrix} w_1^2 & w_2^2 \end{pmatrix}} f\left(\left(\begin{pmatrix} w_{11}^1 & w_{21}^1 \\ w_{12}^1 & w_{22}^1 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} w_{01}^1 \\ x_{02}^1 \end{pmatrix}\right)\right) \boxed{+ w_0^2}\right)$$

Adding Hidden Layers

To increase computing power.

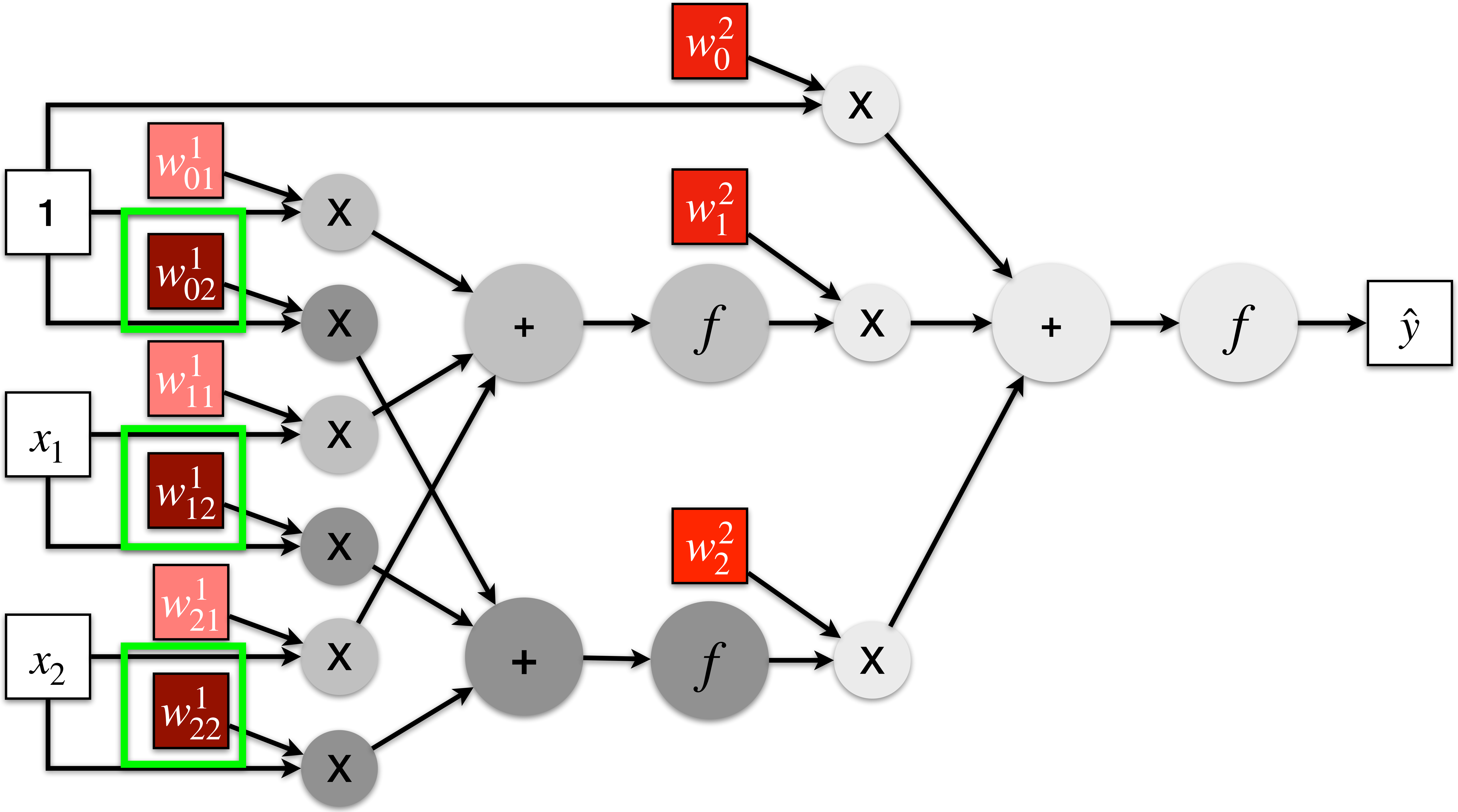


$$\hat{y} = f\left(\begin{pmatrix} w_1^2 & w_2^2 \end{pmatrix} f\left(\begin{pmatrix} w_{11}^1 & w_{21}^1 \\ w_{12}^1 & w_{22}^1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} w_{01}^1 \\ x_{02}^1 \end{pmatrix} \right) + w_0^2 \right)$$

LOYOLA UNIVERSITY CHICAGO

Adding Hidden Layers

To increase computing power.

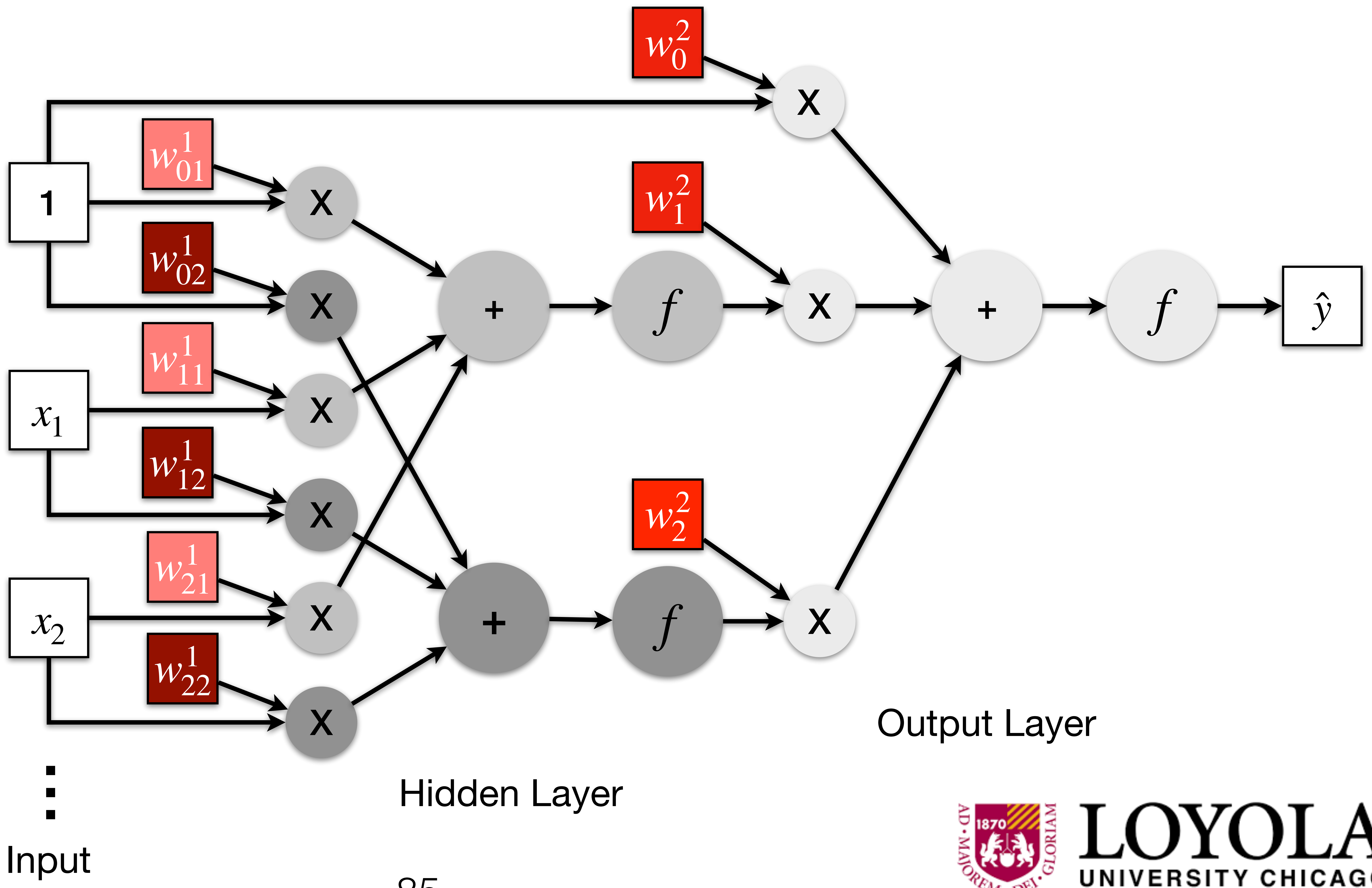


$$\hat{y} = f\left(\begin{pmatrix} w_1^2 & w_2^2 \end{pmatrix} f\left(\begin{pmatrix} w^1_{11} & w^1_{21} \\ w^1_{12} & w^1_{22} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} w^1_{01} \\ x^1_{02} \end{pmatrix} \right) + w_0^2 \right)$$

Neural Networks (NN)

Adding Hidden Layers

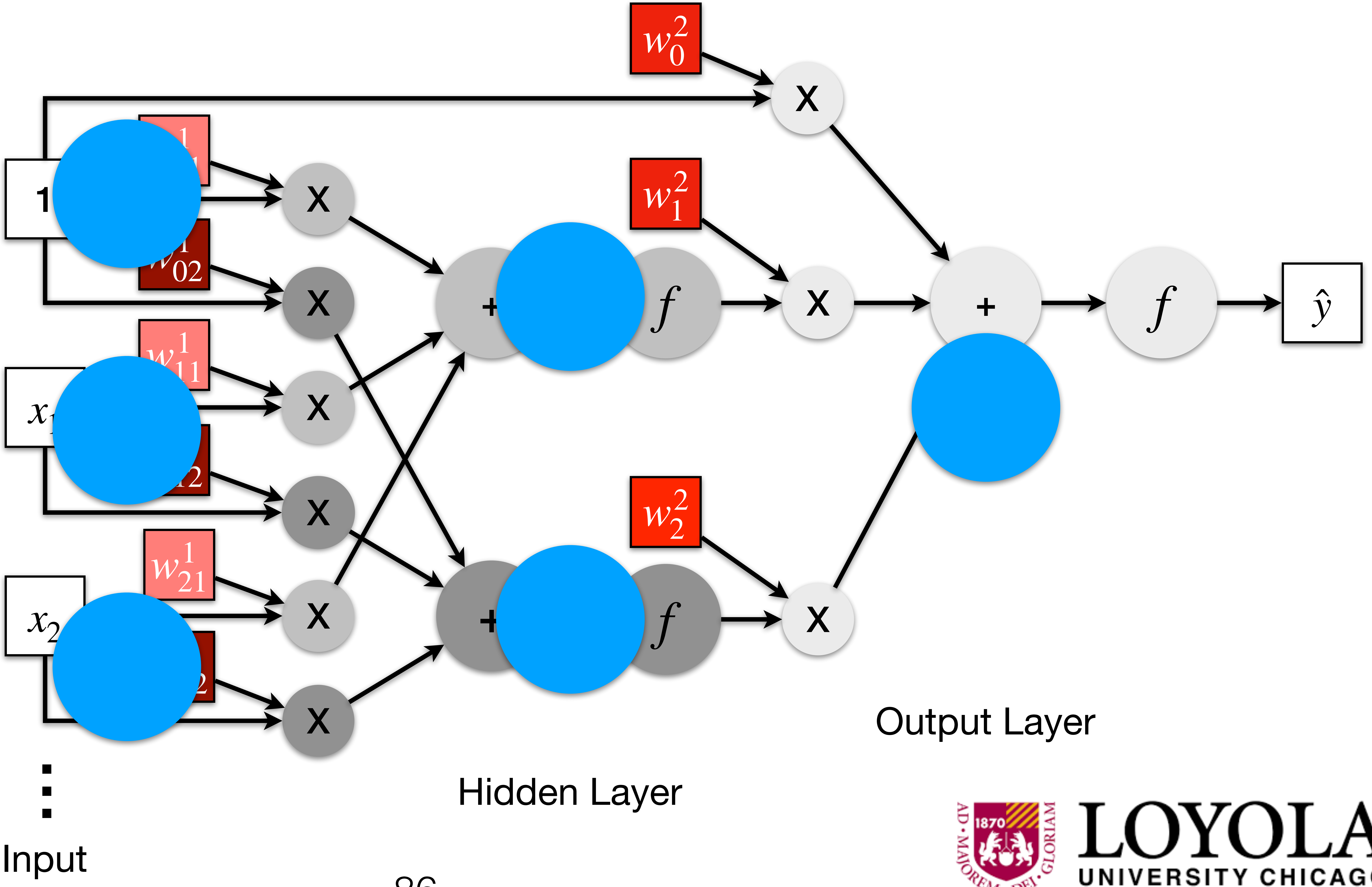
To increase computing power.



Neural Networks (NN)

Adding Hidden Layers

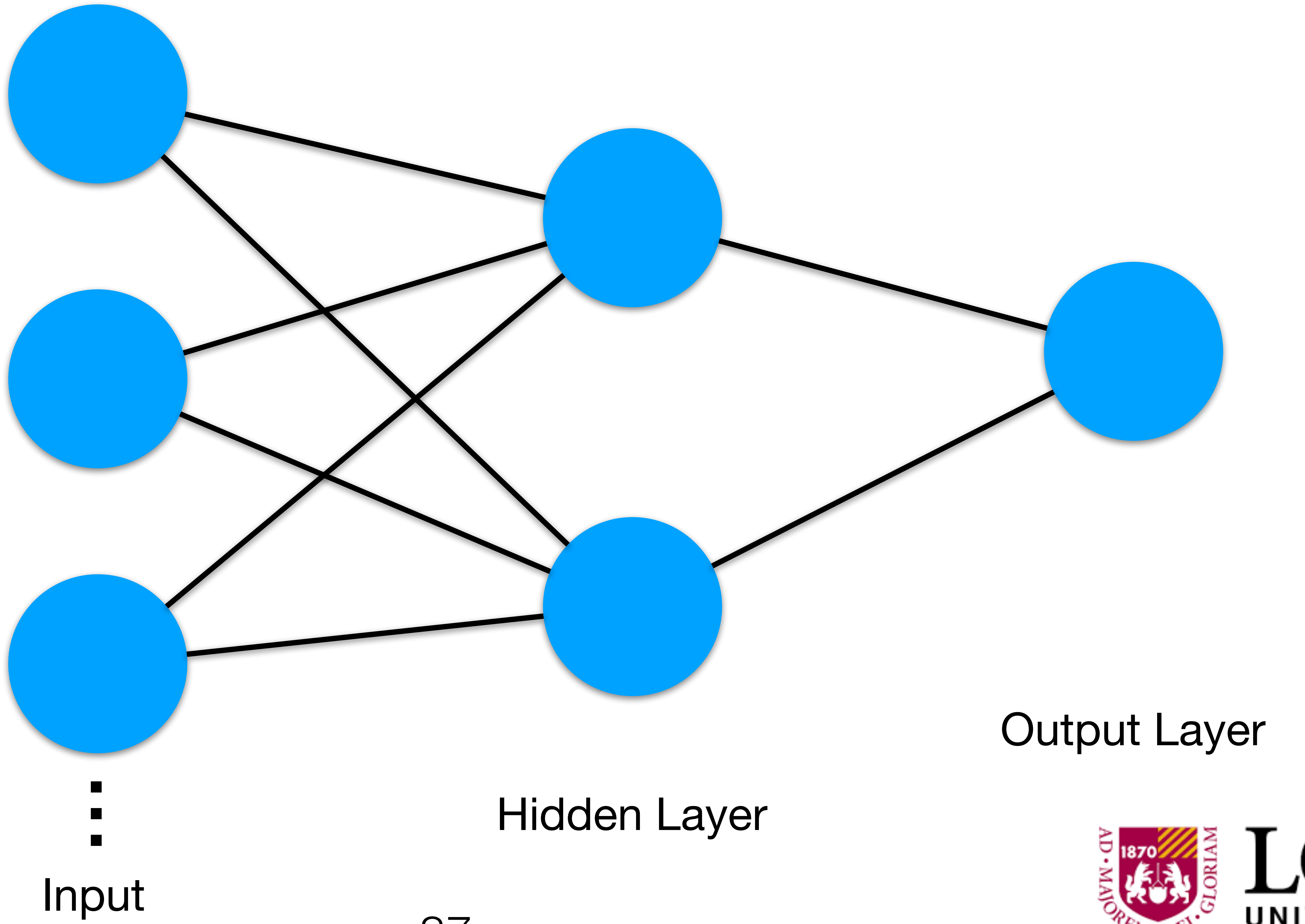
To increase computing power.



Neural Networks (NN)

Adding Hidden Layers

To increase computing power.



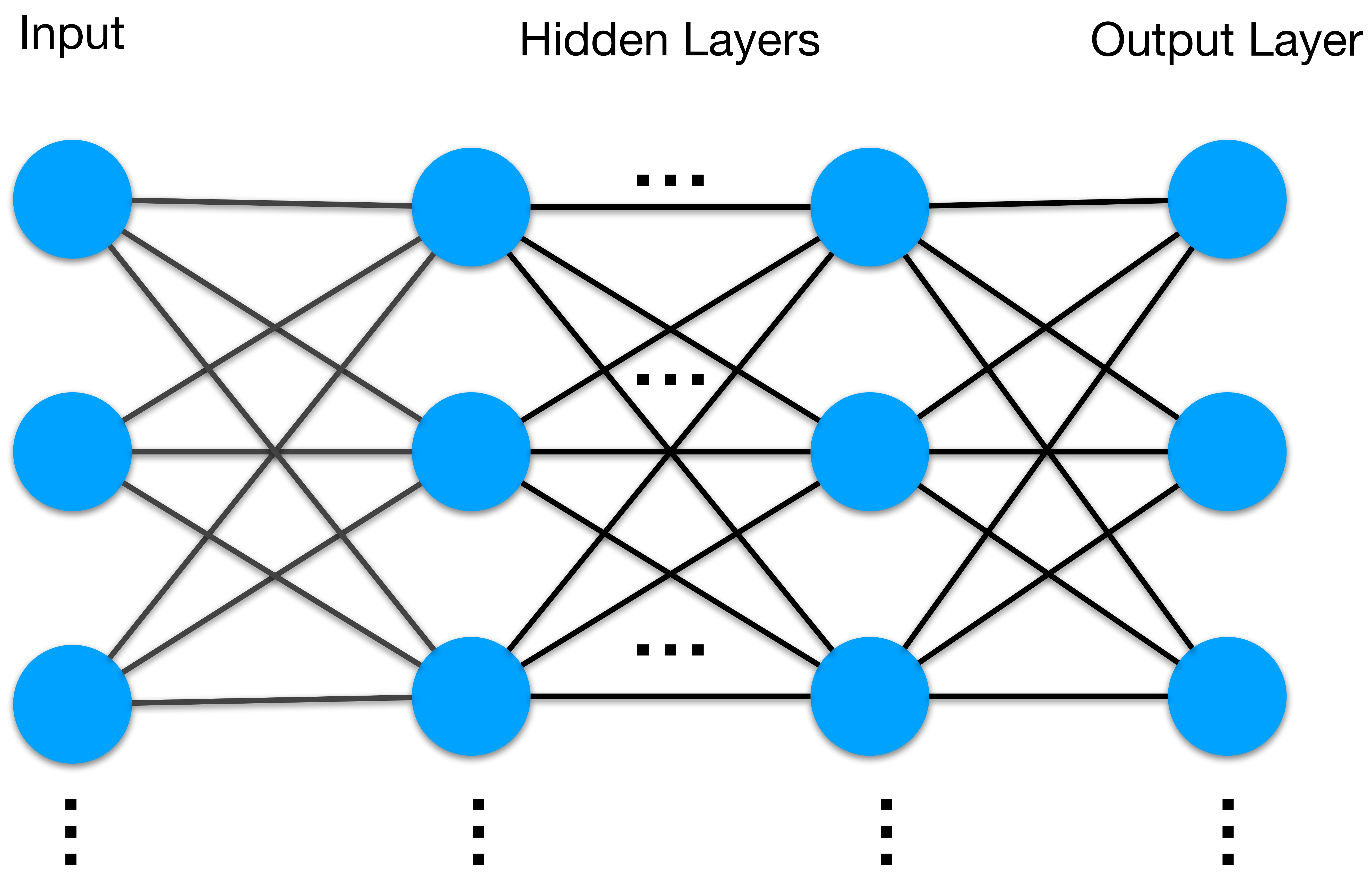
LOYOLA UNIVERSITY CHICAGO

Neural Networks (NN)

Universal Approximation Theorem

An NN with one or more hidden layers with compressive (non-linear) activation function can approximate any continuous function.

Training this NN might be a nightmare!

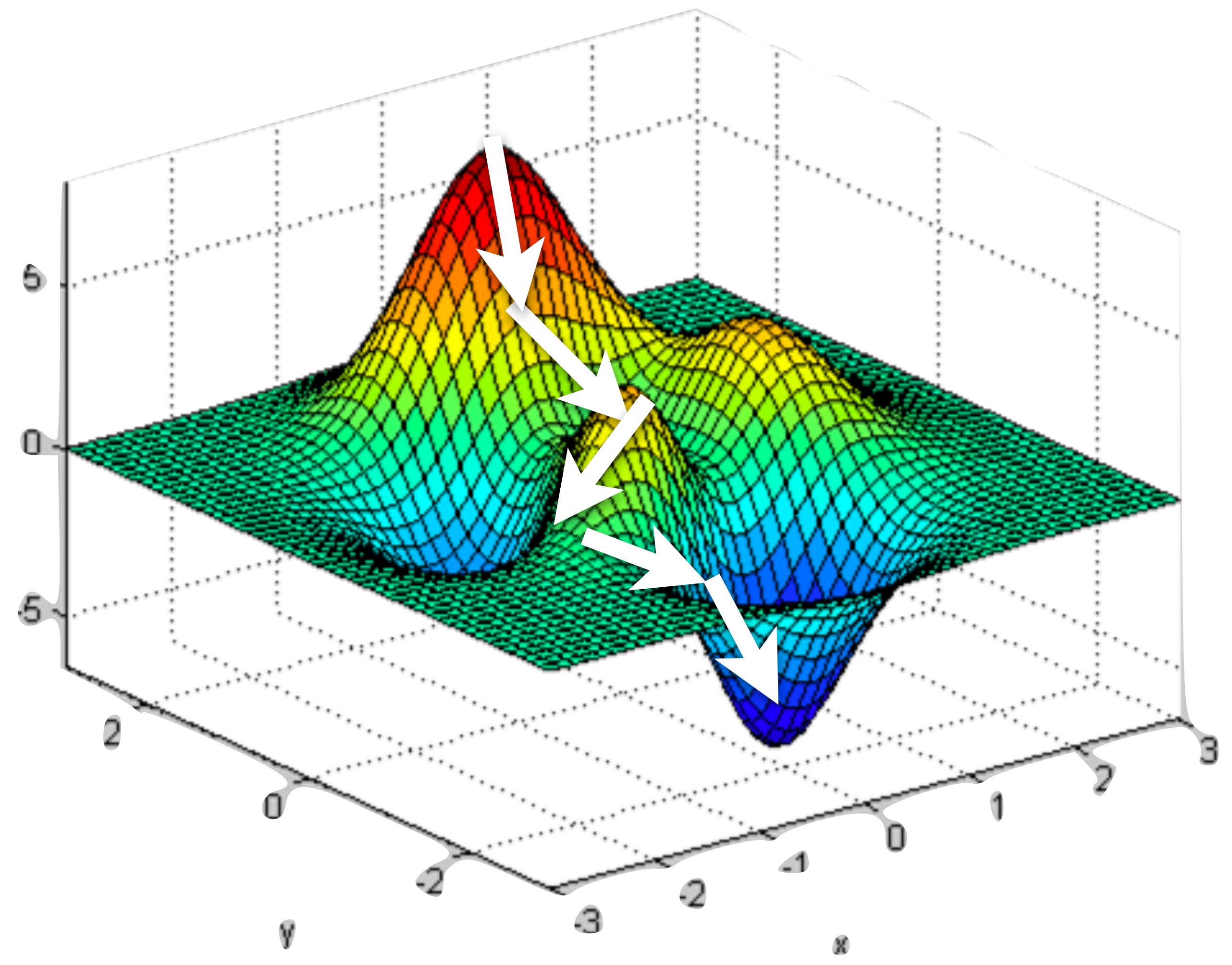


LOYOLA UNIVERSITY CHICAGO

Neural Networks (NN)

Backpropagation

Key algorithm to train the NN.

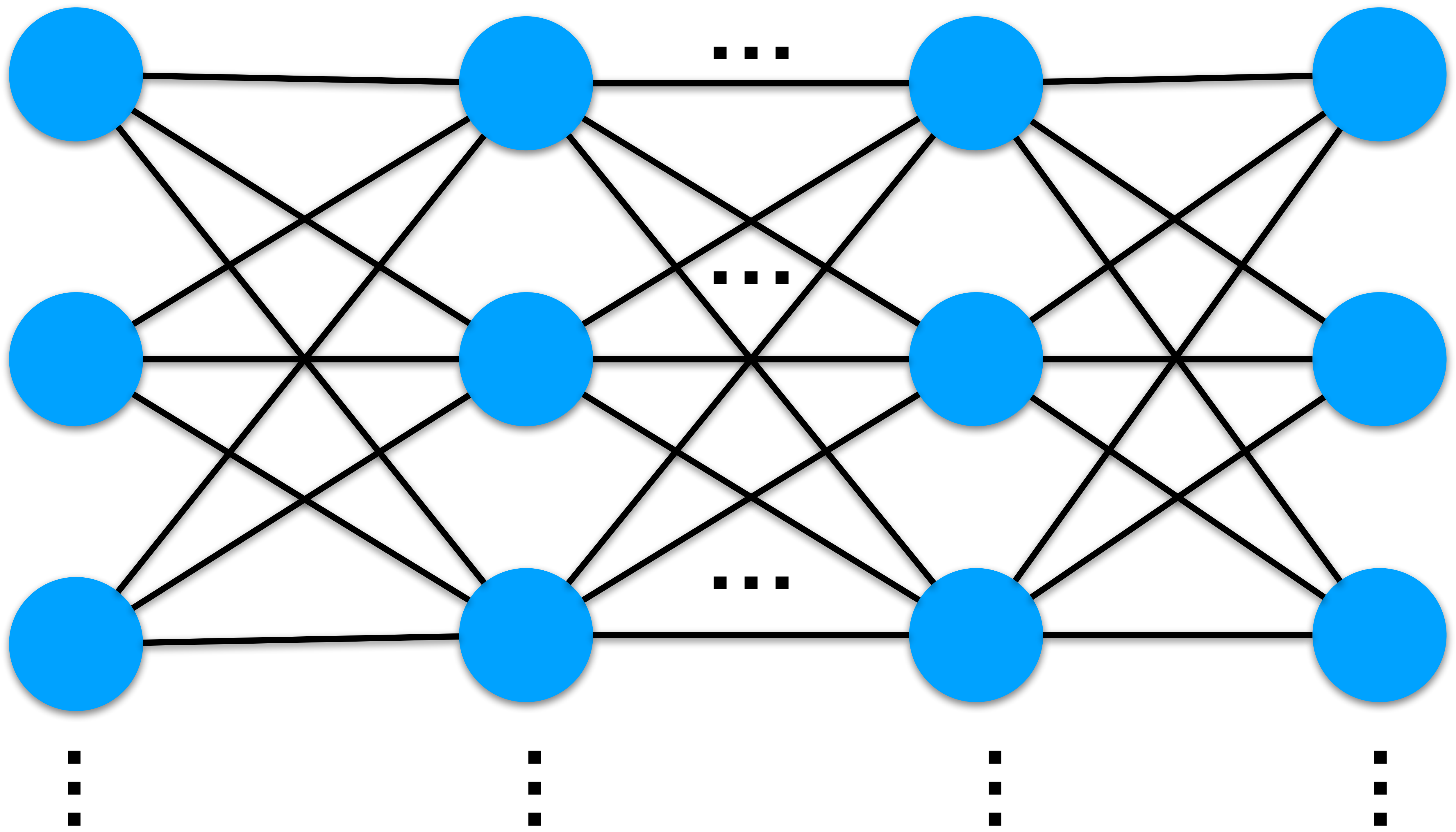


Loss Surface
Gradient Descent

Input

Hidden Layers

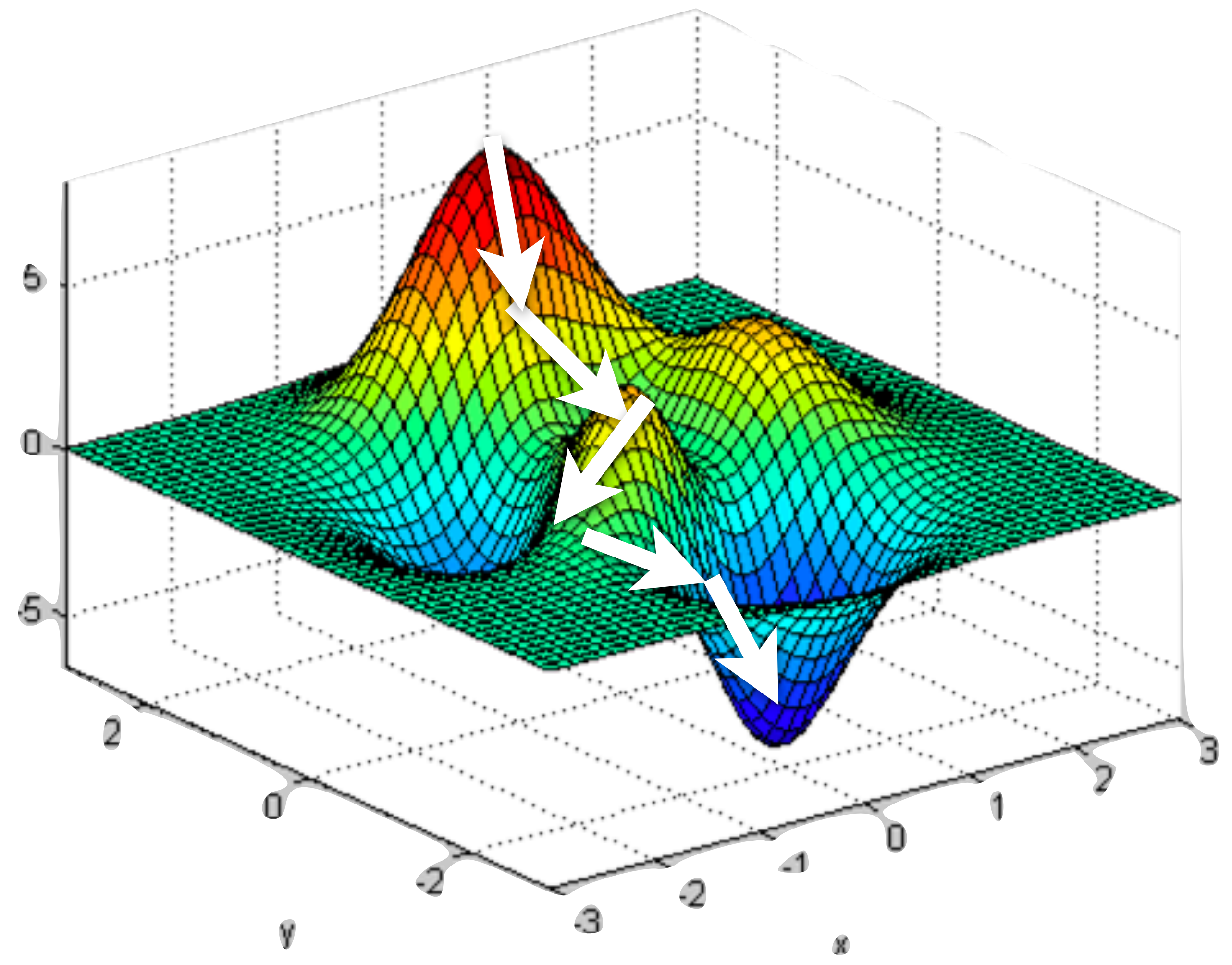
Output Layer



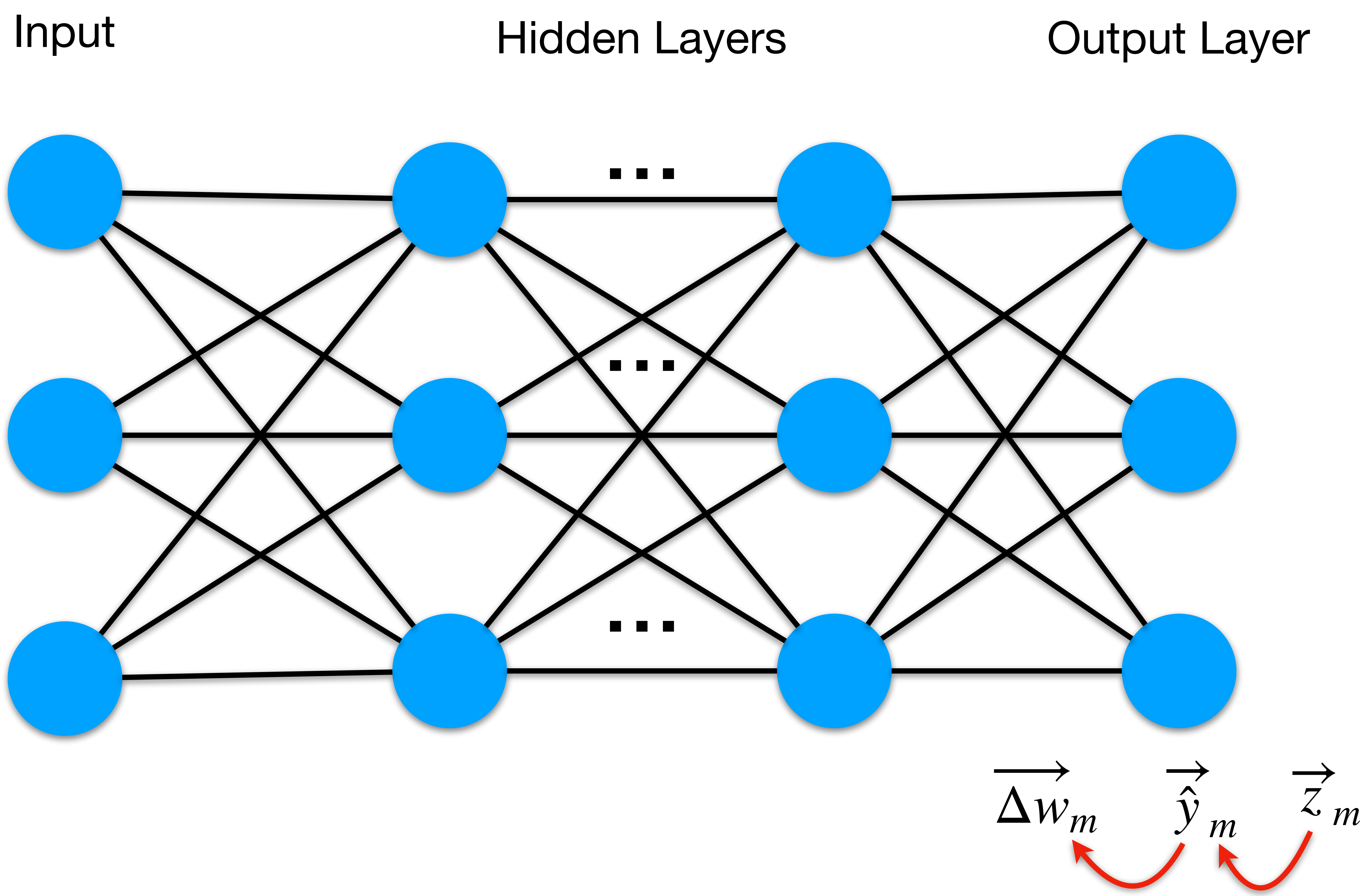
Neural Networks (NN)

Backpropagation

Key algorithm to train the NN.



Loss Surface
Gradient Descent



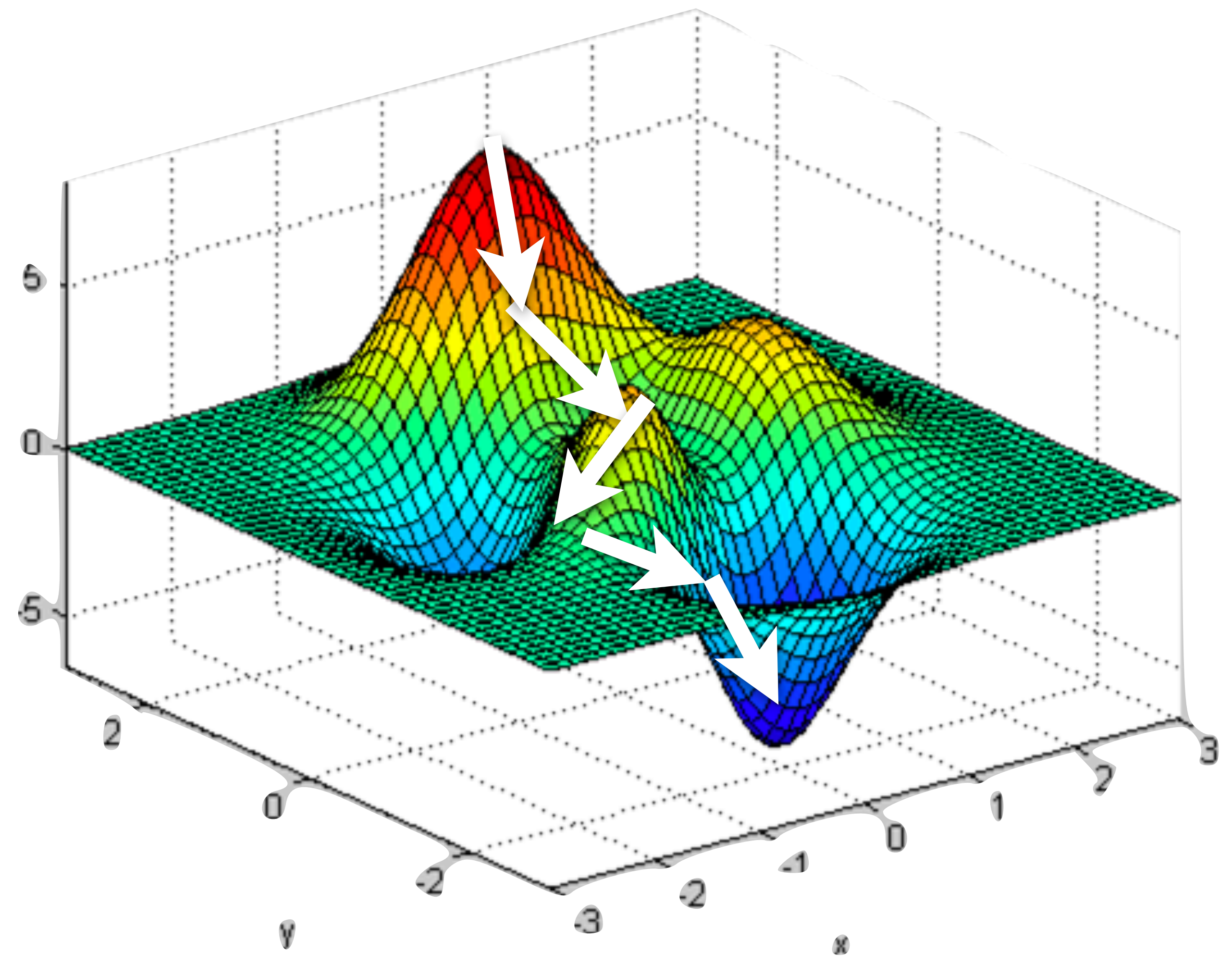
m samples, l hidden layers

LOYOLA UNIVERSITY CHICAGO

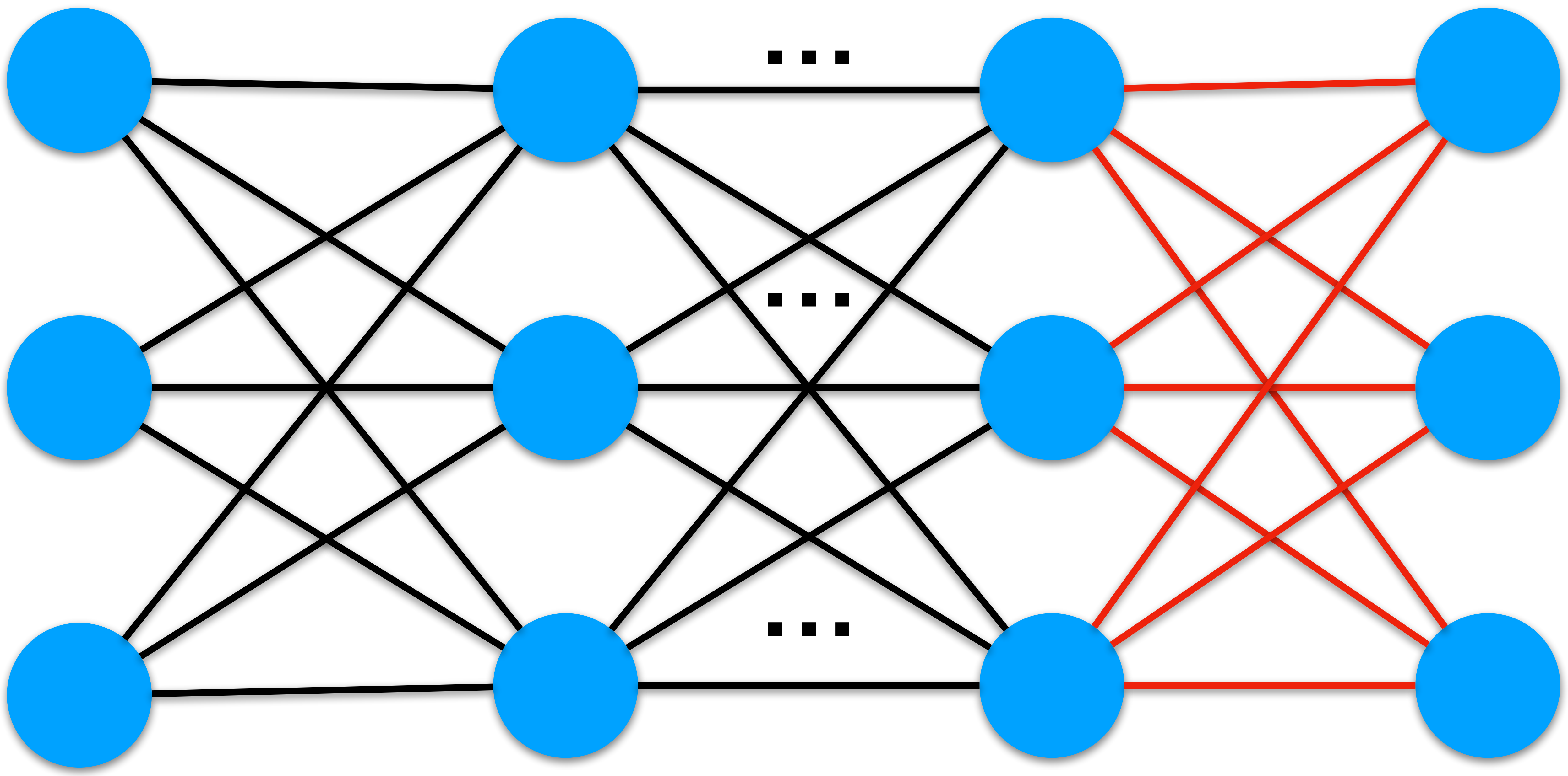
Neural Networks (NN)

Backpropagation

Key algorithm to train the NN.



Loss Surface
Gradient Descent



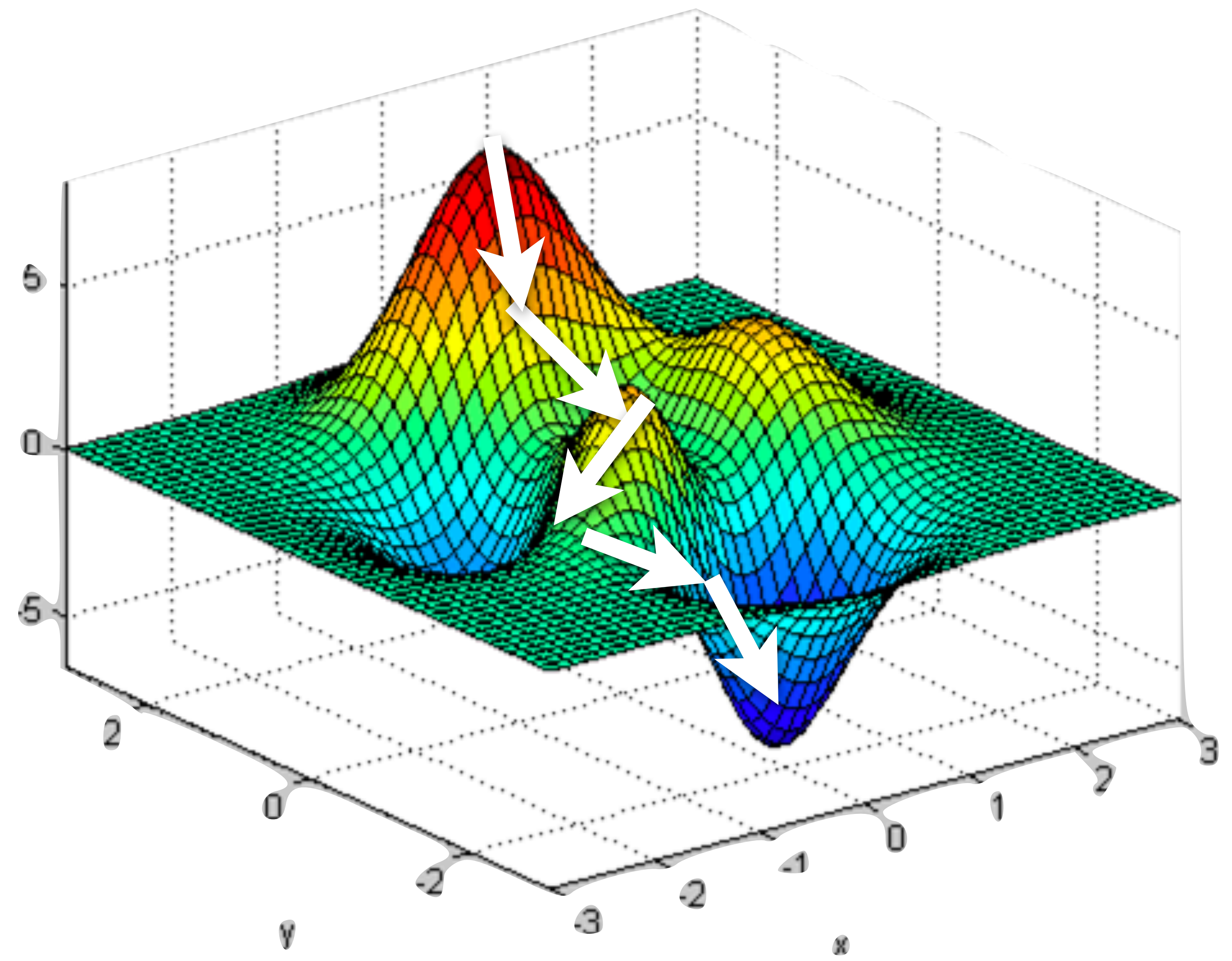
$\overrightarrow{\Delta w}_m \quad \vec{\hat{y}}_m \quad \vec{z}_m$

m samples, l hidden layers

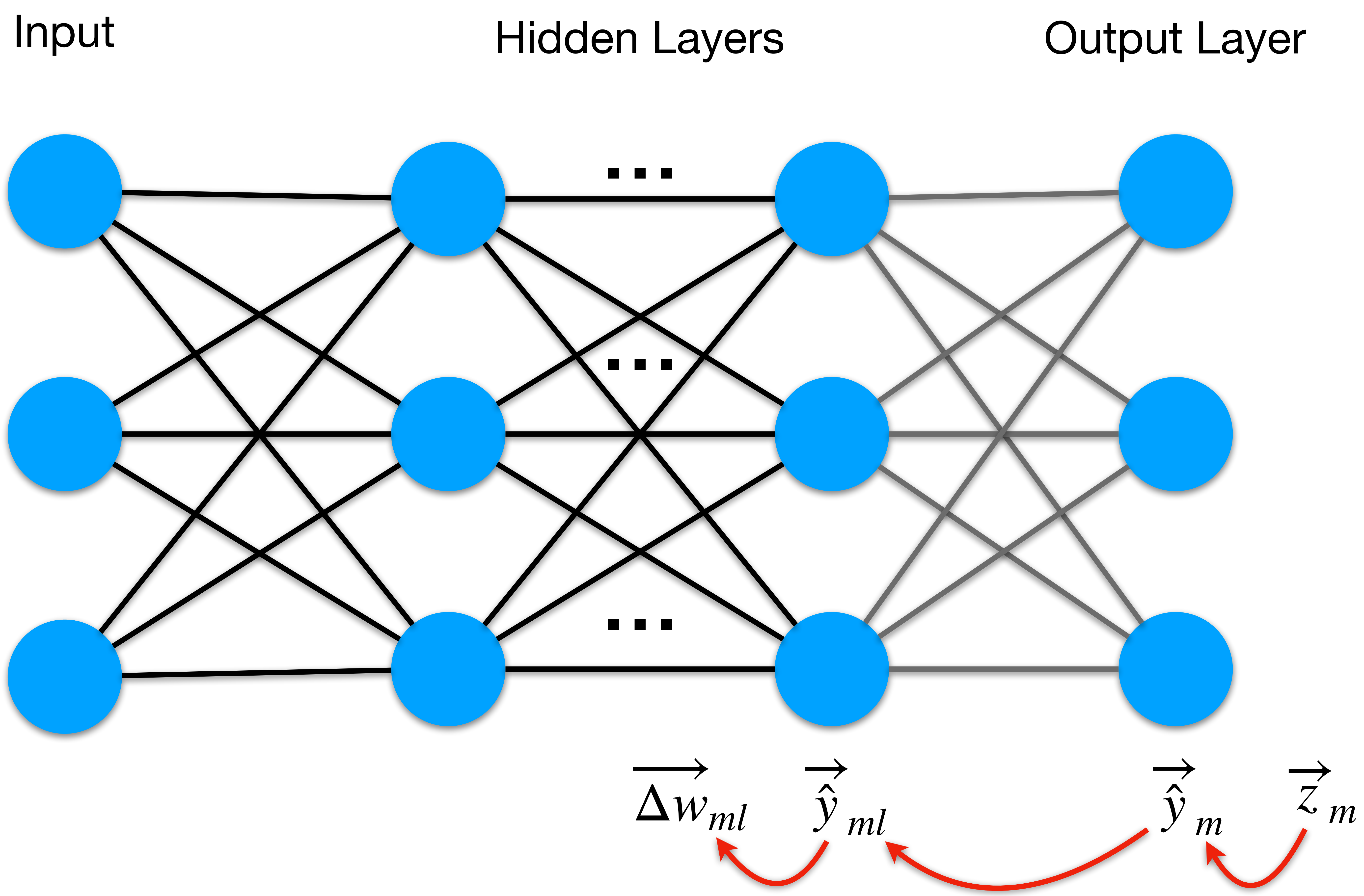
Neural Networks (NN)

Backpropagation

Key algorithm to train the NN.



Loss Surface
Gradient Descent

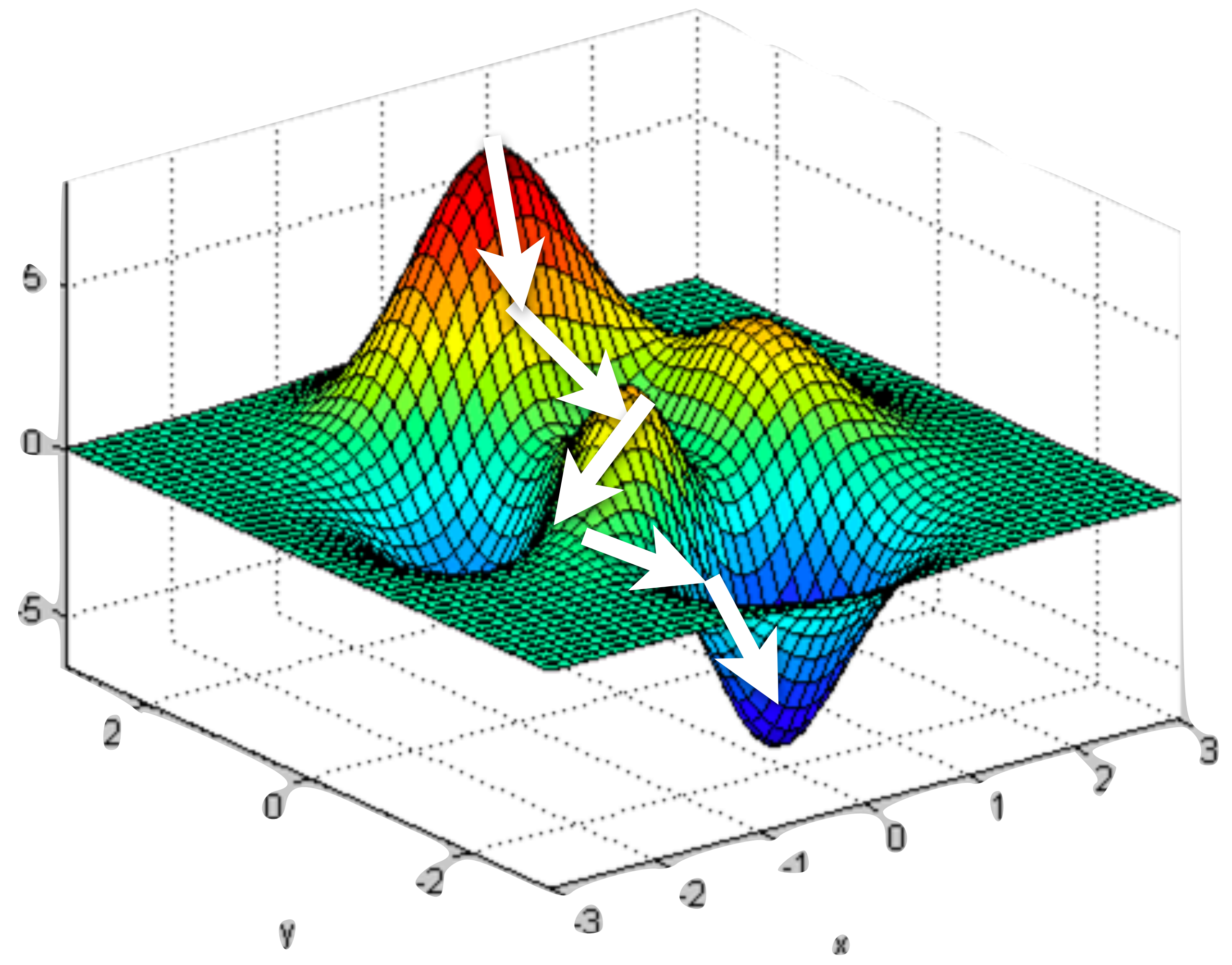


m samples, l hidden layers

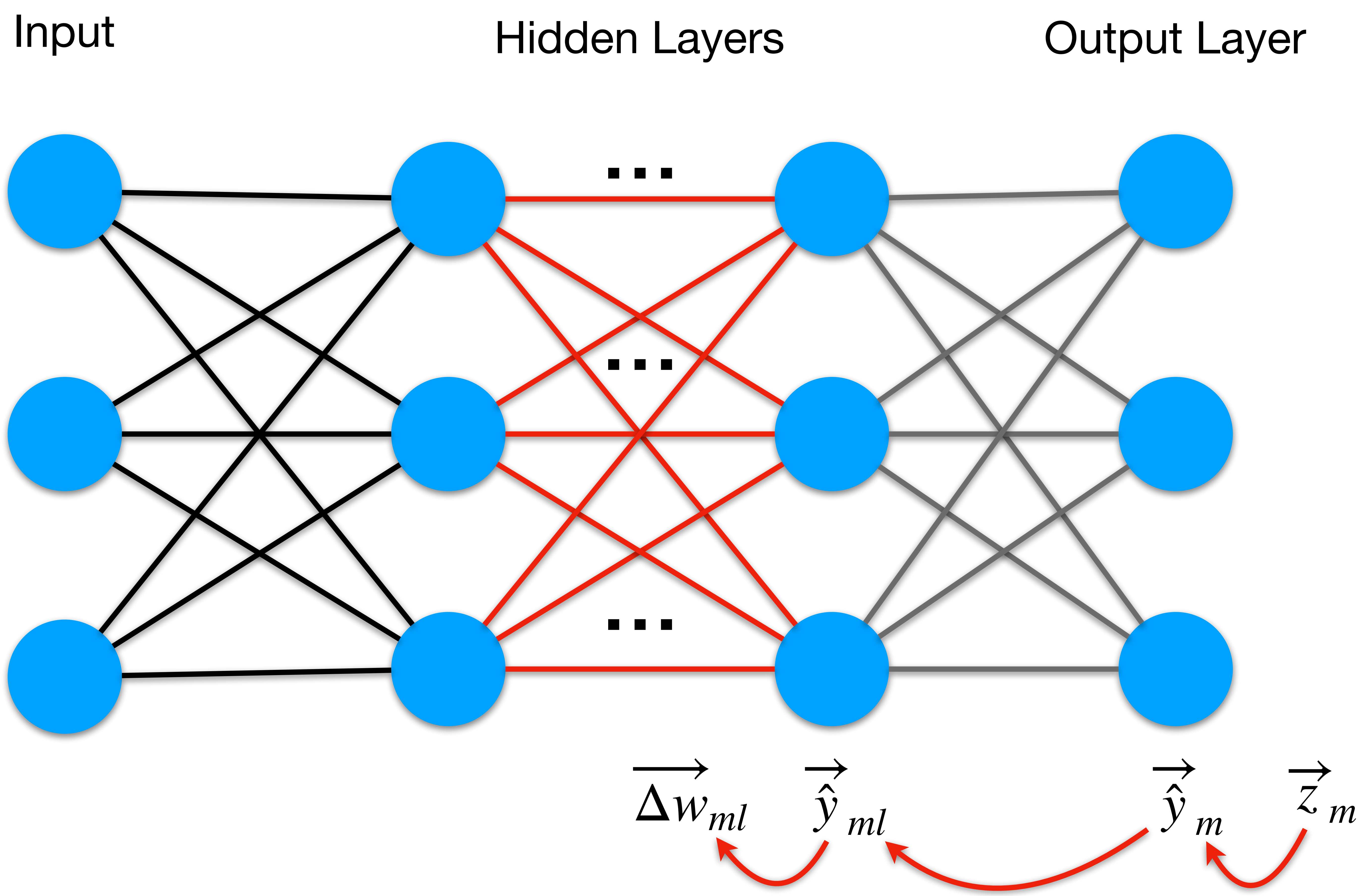
Neural Networks (NN)

Backpropagation

Key algorithm to train the NN.



Loss Surface
Gradient Descent



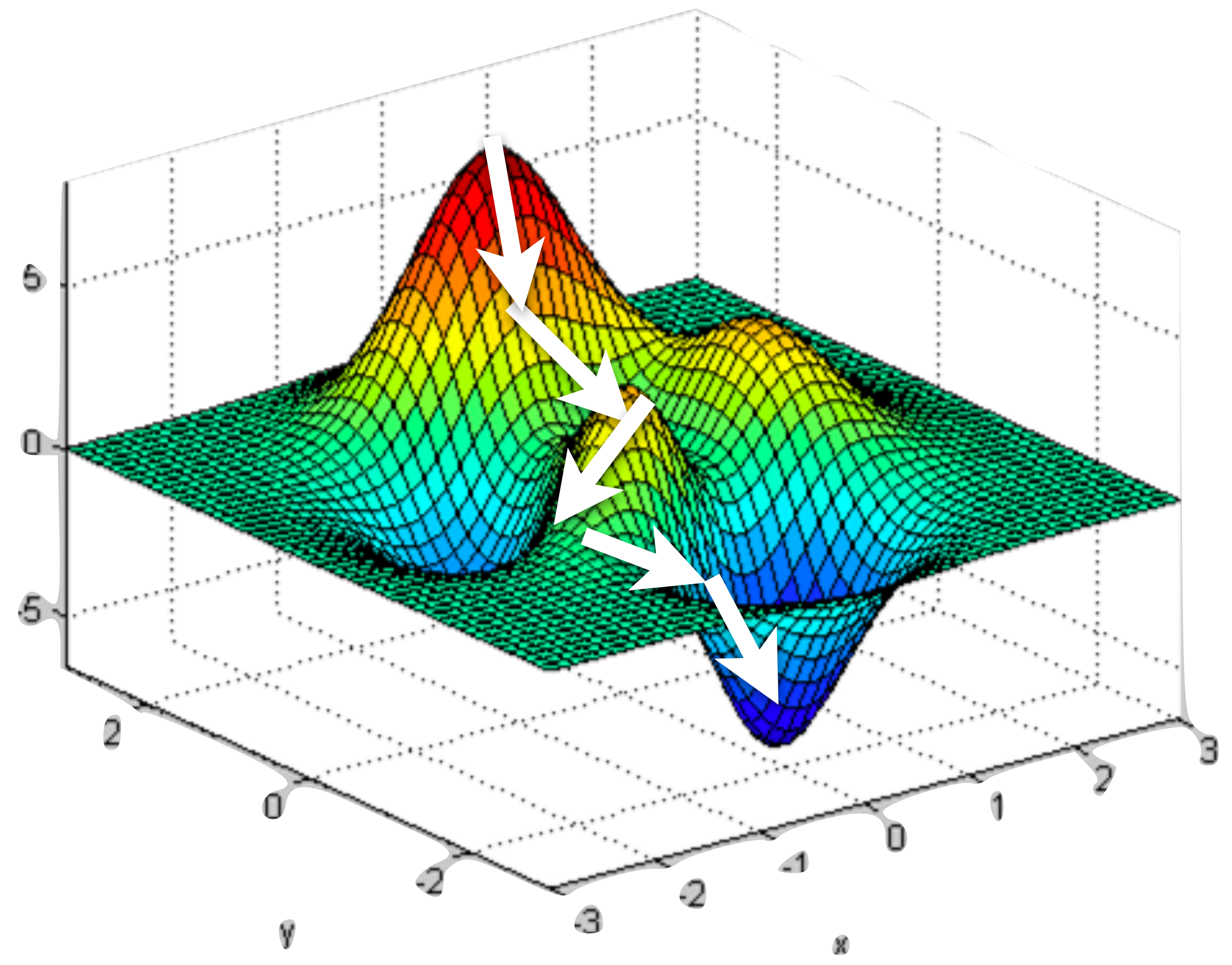
m samples, l hidden layers

LOYOLA UNIVERSITY CHICAGO

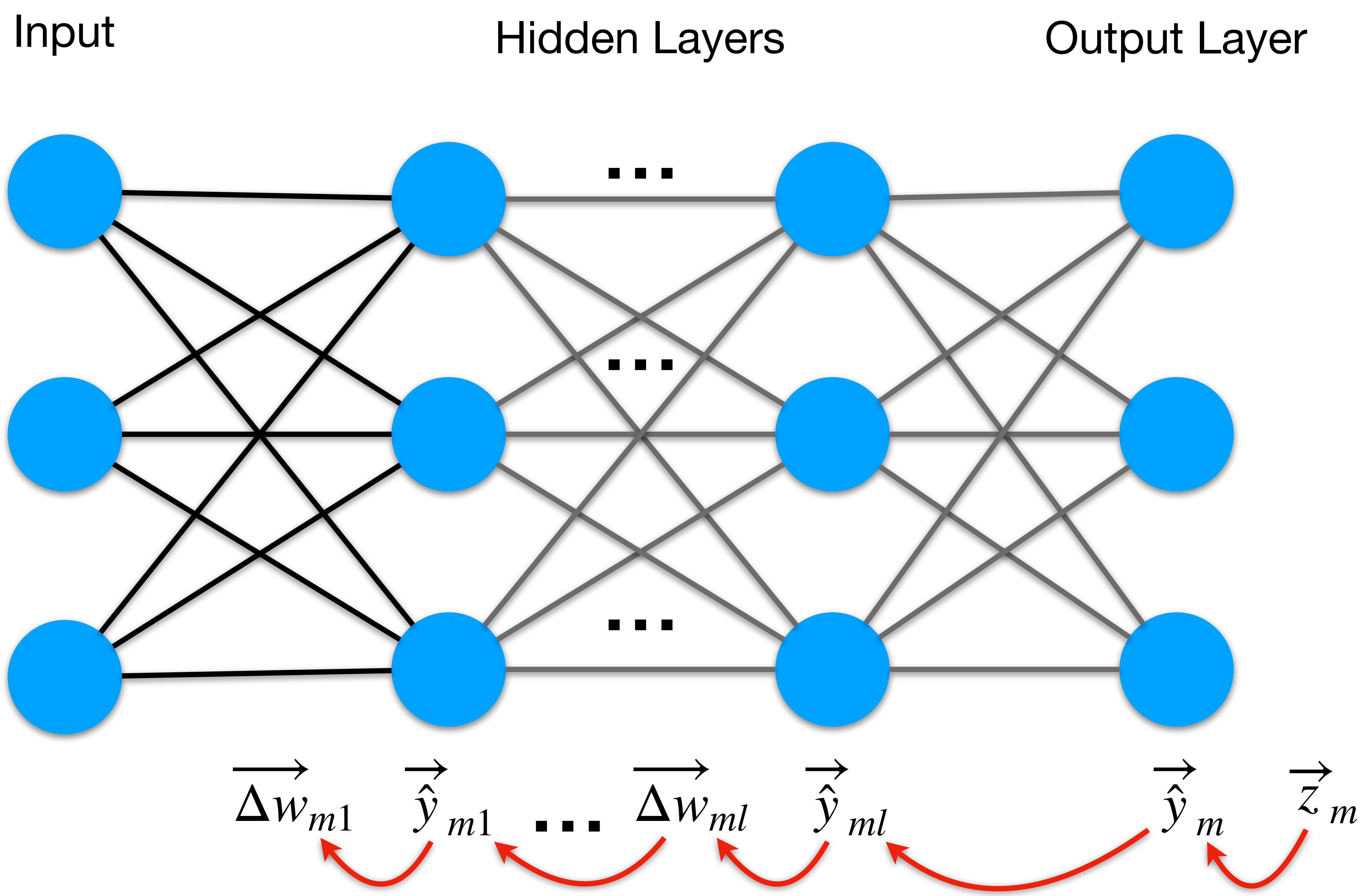
Neural Networks (NN)

Backpropagation

Key algorithm to train the NN.



Loss Surface
Gradient Descent



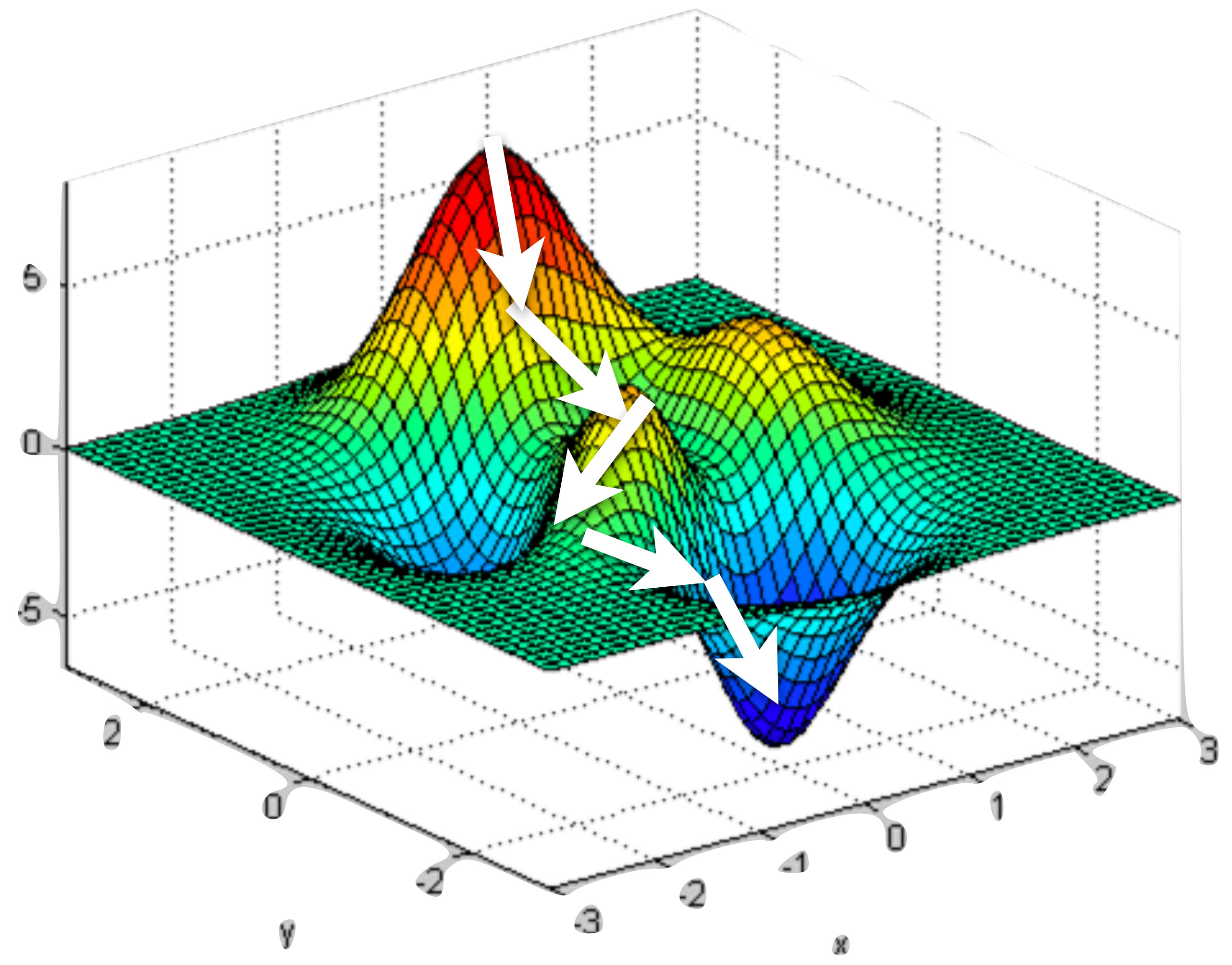
m samples, l hidden layers

LOYOLA UNIVERSITY CHICAGO

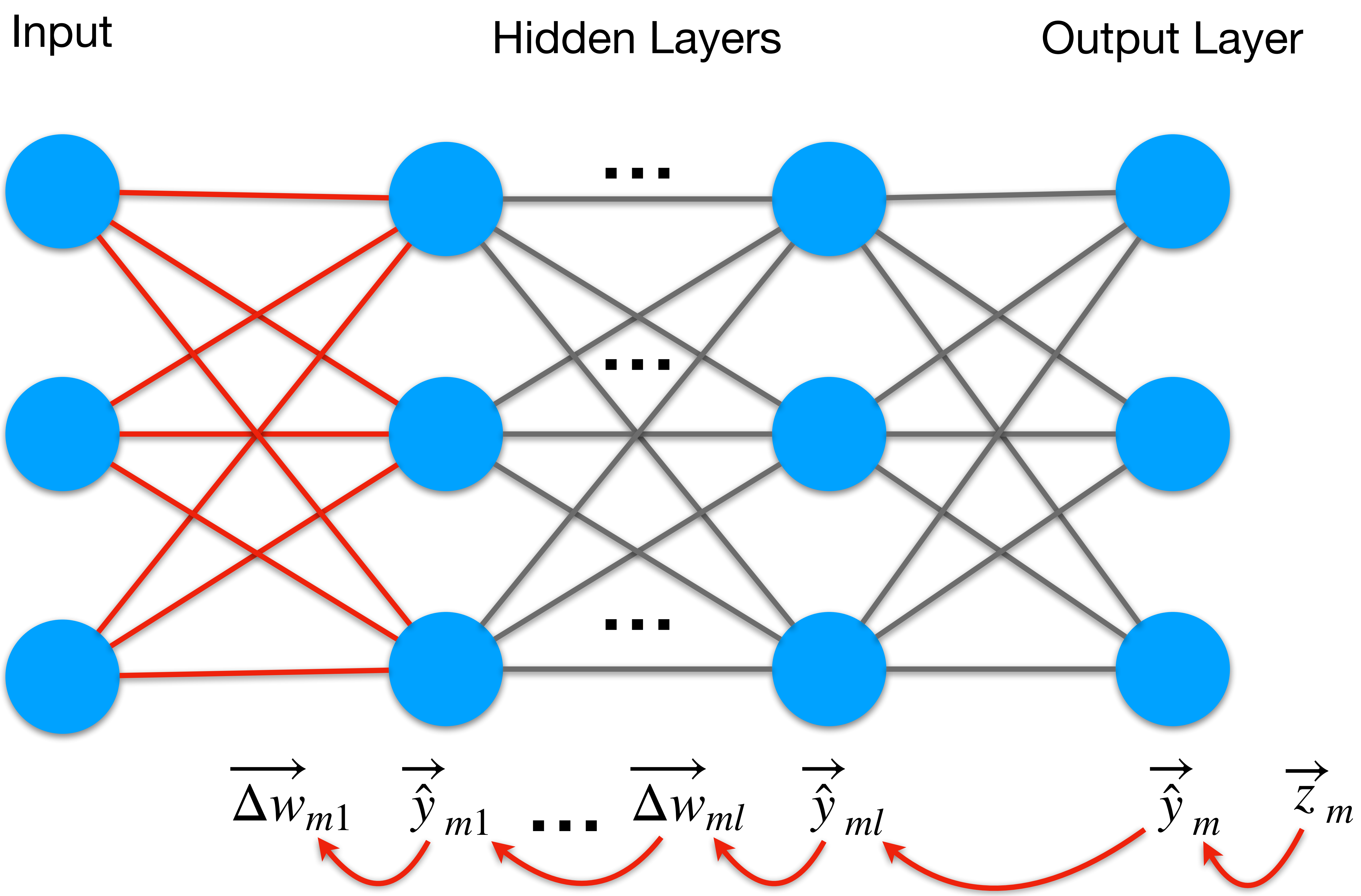
Neural Networks (NN)

Backpropagation

Key algorithm to train the NN.



Loss Surface
Gradient Descent

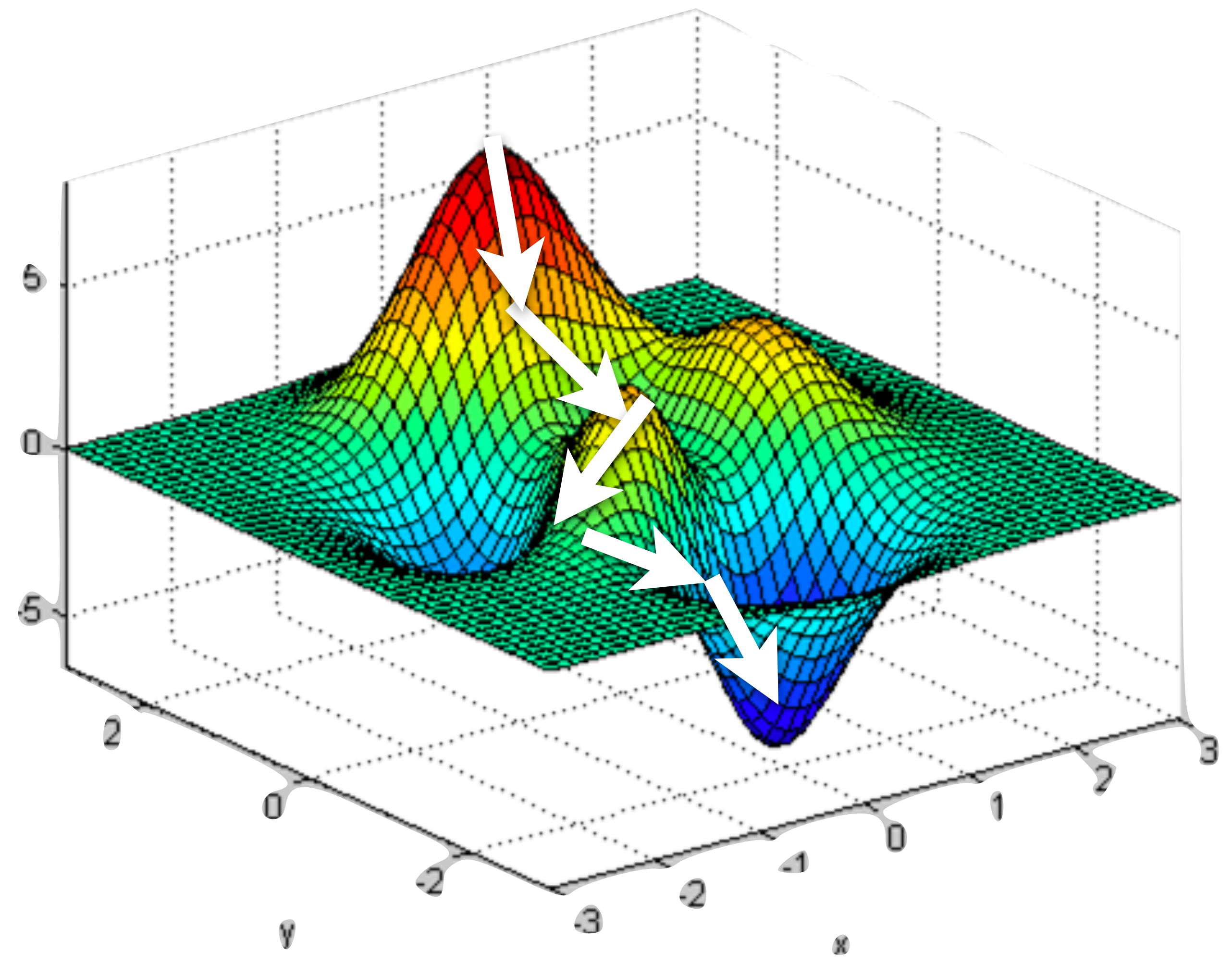


m samples, l hidden layers

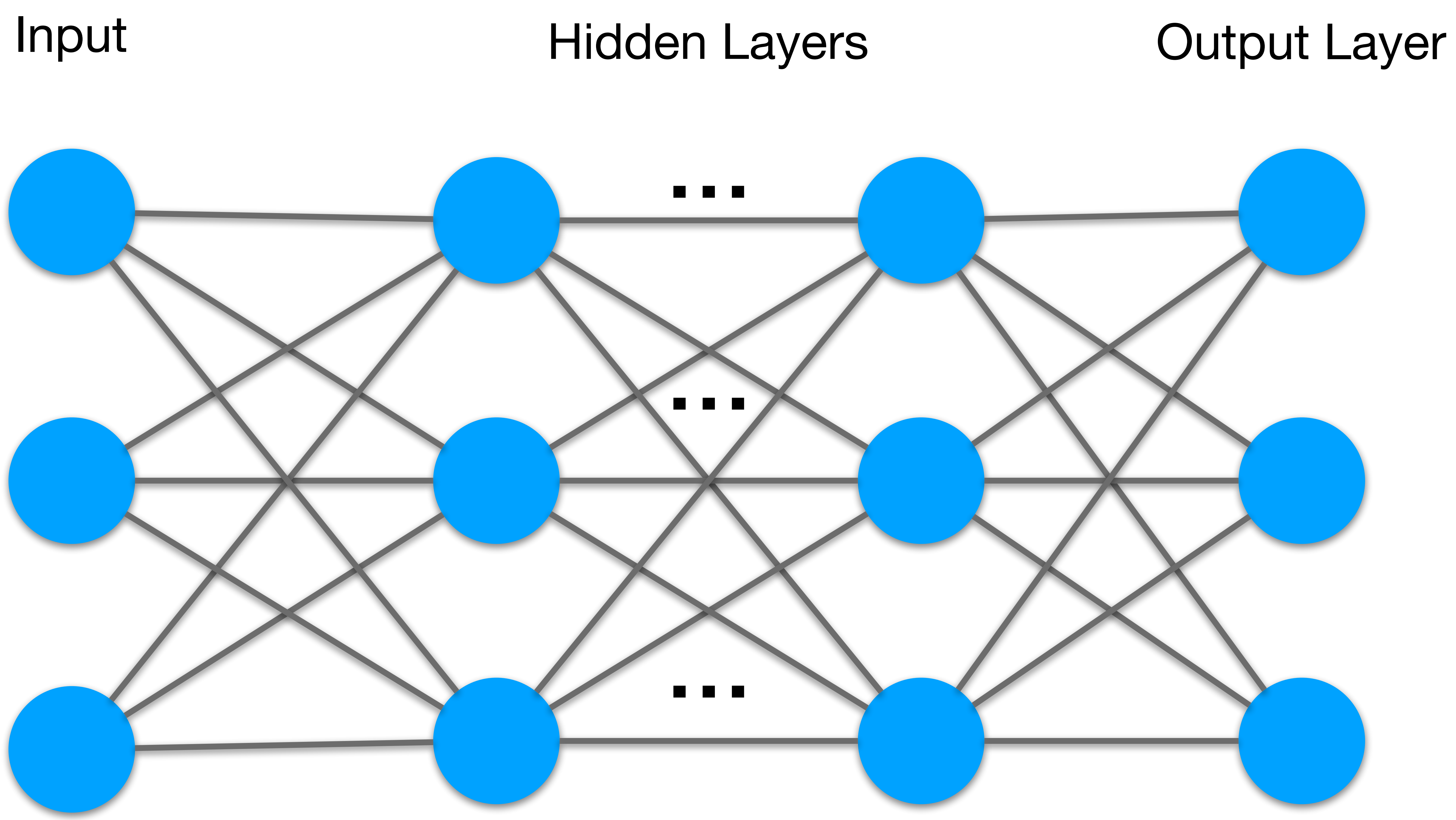
Neural Networks (NN)

Backpropagation

Key algorithm to train the NN.



Loss Surface
Gradient Descent

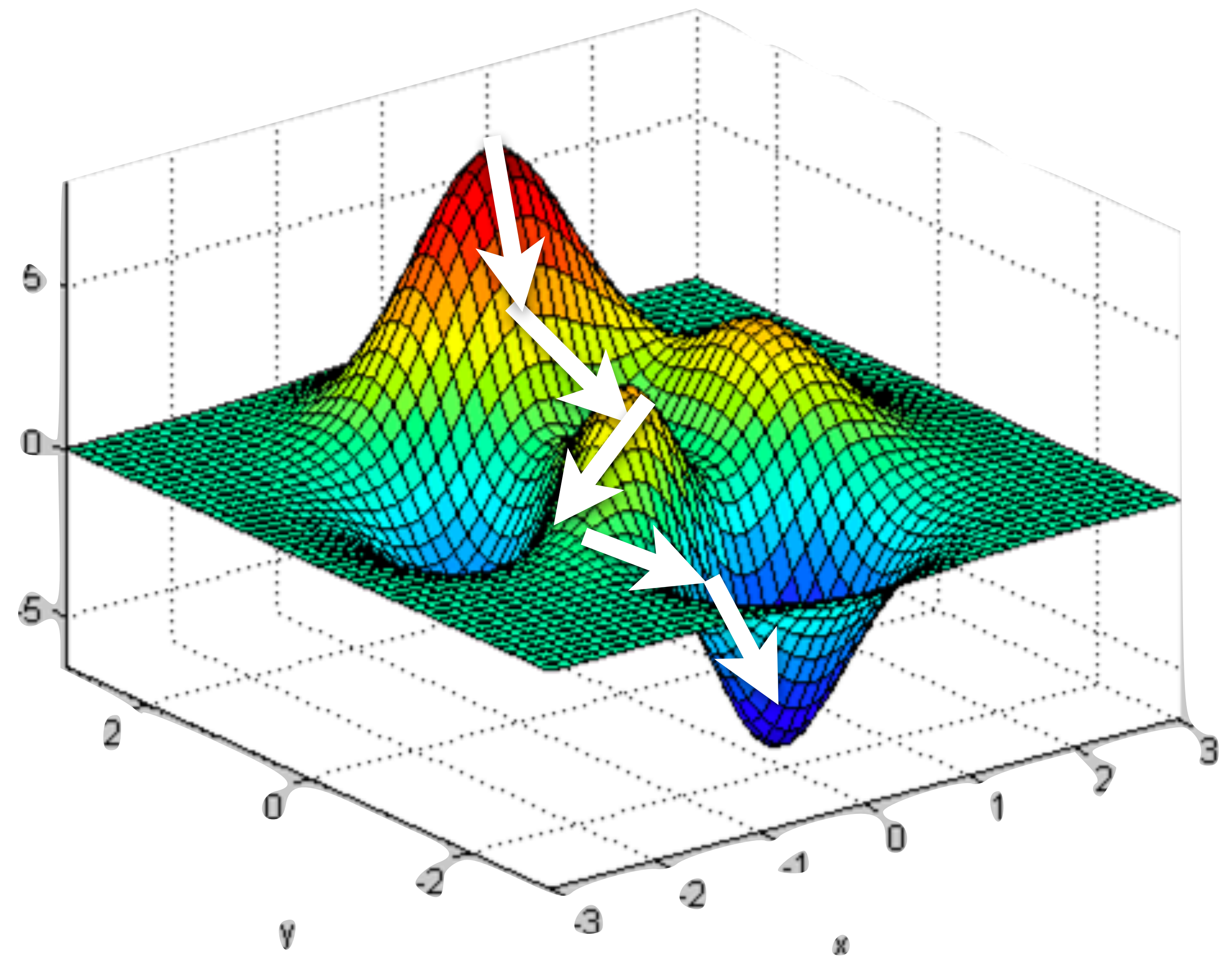


m samples, l hidden layers

Neural Networks (NN)

Backpropagation

Key algorithm to train the NN.



Loss Surface
Gradient Descent

Challenges

The m samples may make back propagation too slow (too much data).

What α (step) should one use?

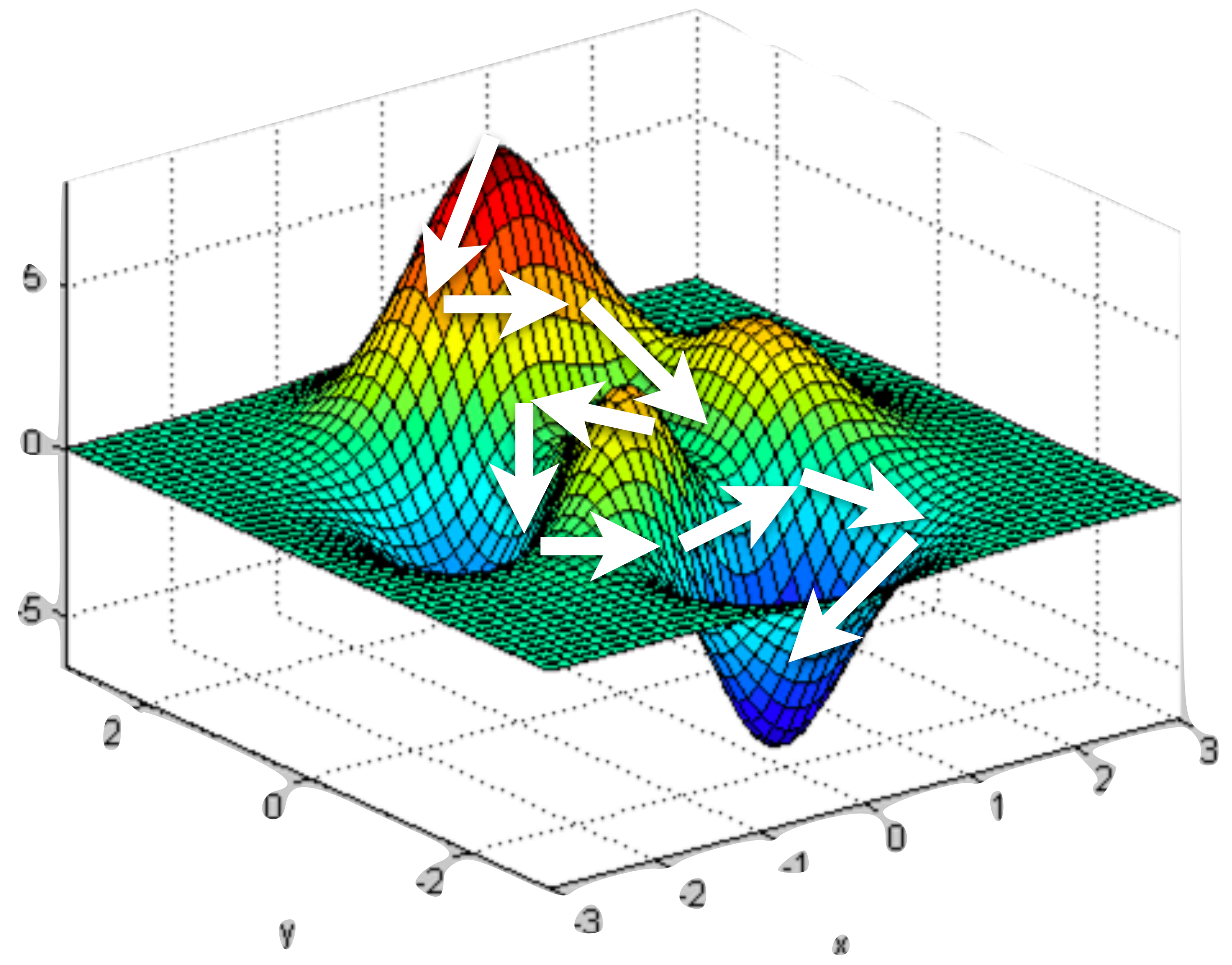
$$\Delta w_i = \sum_{k=1}^{m} \alpha(z_k - f(\vec{x}_k^T \cdot \vec{w})) \times x_{ki} \times f'(\vec{x}_k^T \cdot \vec{w}) = 0$$

LOYOLA UNIVERSITY CHICAGO

Neural Networks (NN)

Backpropagation

Key algorithm to train the NN.



Loss Surface
Stochastic Gradient Descent (SGD)

Already implemented in the NN libraries.

Challenges

The m samples may make back propagation too slow (too much data).

What α (step) should one use?

$$\Delta w_i = \sum_{k=1}^{m} \alpha(z_k - f(\vec{x}_k^T \cdot \vec{w})) \times x_{ki} \times f'(\vec{x}_k^T \cdot \vec{w}) = 0$$

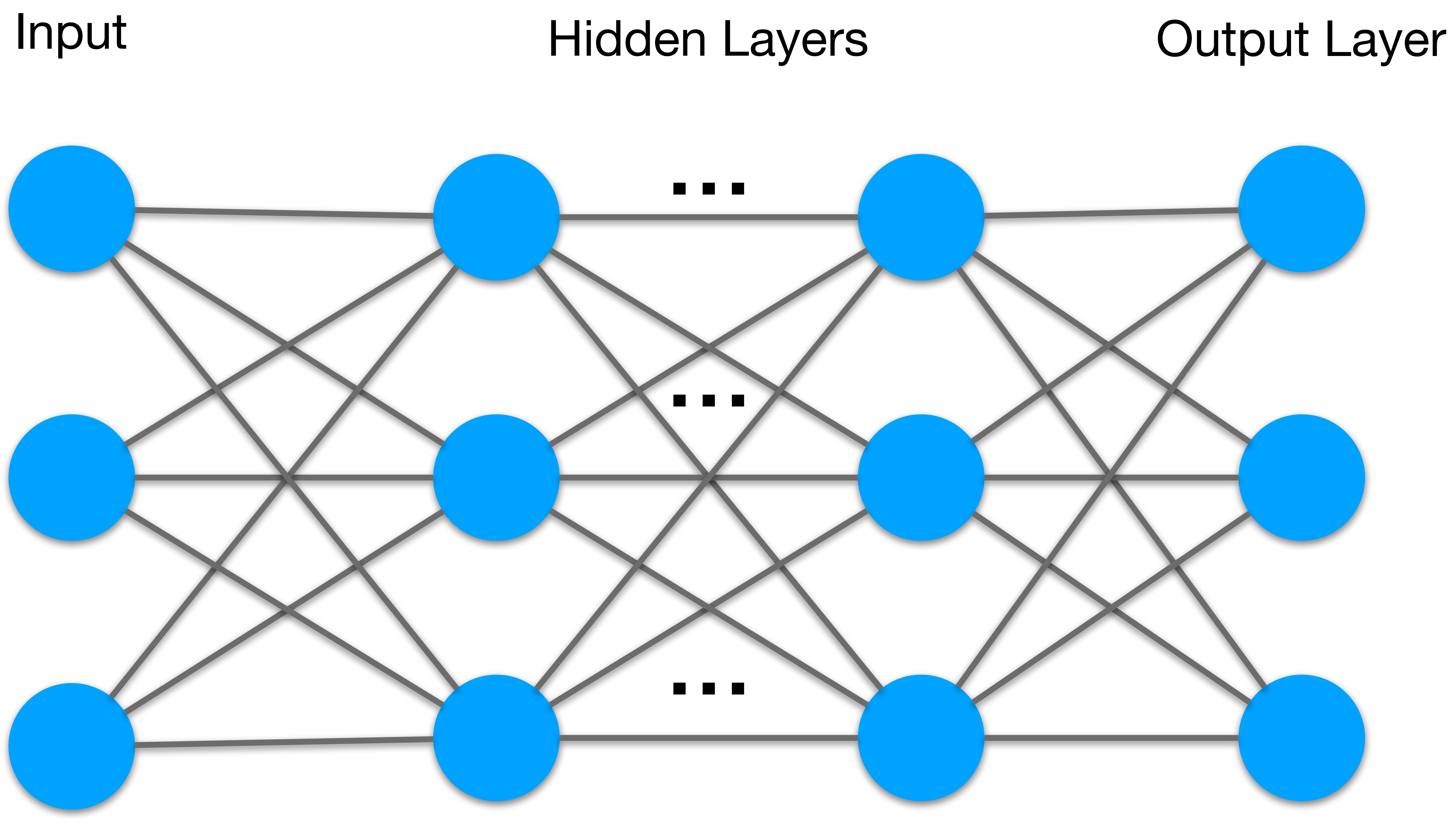
Solution
Stochastic Gradient Descent
Randomly select multiple smaller subsets of the m samples (mini-batches).
Run more but faster iterations of backpropagation.

Convolutional Neural Networks (CNN)

How NNs are used in CV?

Flatten the image and
feed pixel values
to the input?

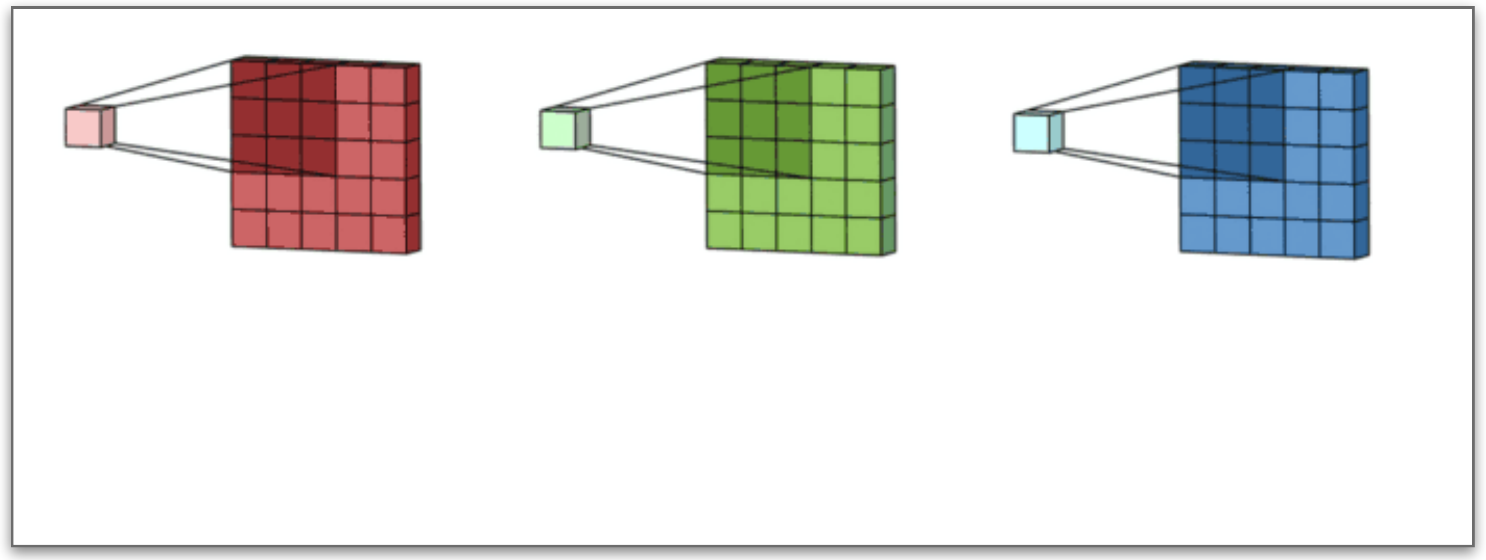
Cons
Pixel values at
homogeneous regions
(e.g., texture and blurred regions)
have too similar values and
may be redundant.



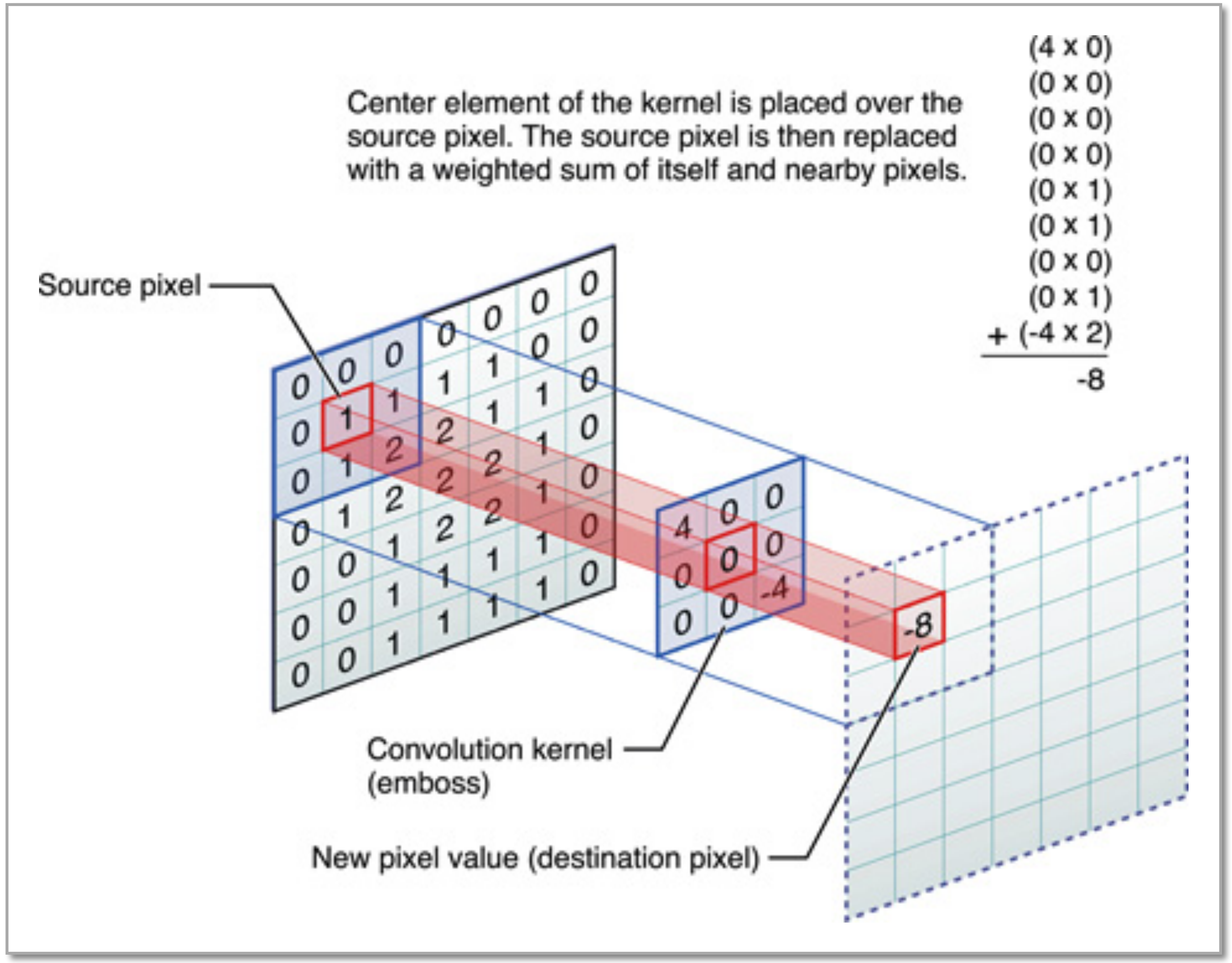
Convolutional Neural Networks (CNN)

How NNs are used in CV?

Solution
Add convolutions to the NN.



https://medium.com/analytics-vidhya/understanding-convolution-operations-in-cnn-1914045816d4



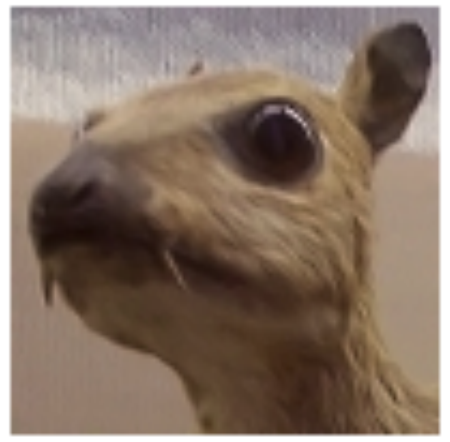
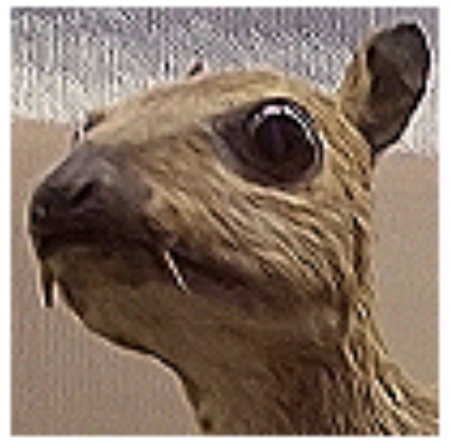
https://developer.apple.com/library/archive/documentation/Performance/Conceptual/vImage/ConvolutionOperations/ConvolutionOperations.html

Convolutional Neural Networks (CNN)

How NNs are used in CV?

Solution
Add convolutions to the NN.

Examples of handcrafted convolutional filters.

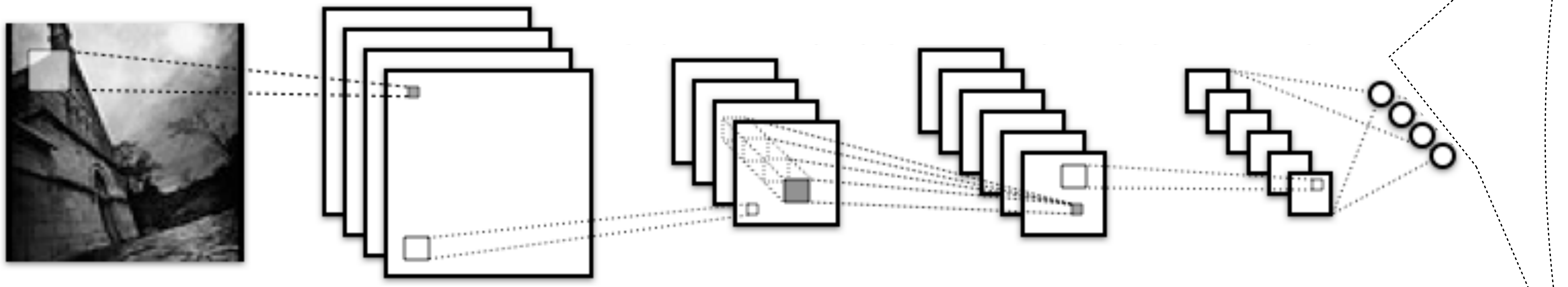
Original	*Gaussian Blur*	*Sharpen*	*Edge Detection*
$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	$\frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$	$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$	$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$
			

https://medium.com/analytics-vidhya/understanding-convolution-operations-in-cnn-1914045816d4

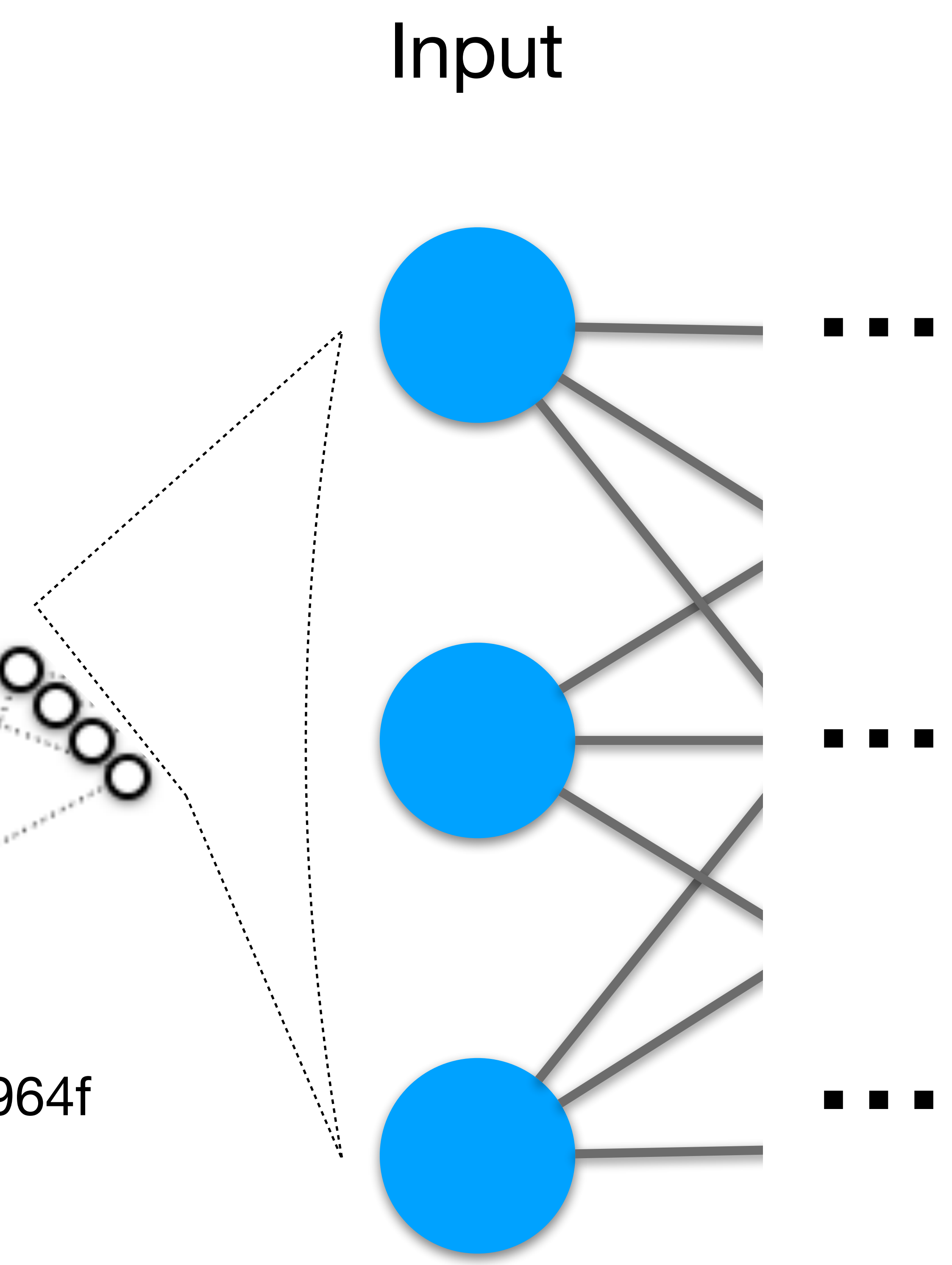
Convolutional Neural Networks (CNN)

How NNs are used in CV?

Solution
Add convolutions to the NN.



https://towardsdatascience.com/build-your-own-convolution-neural-network-in-5-mins-4217c2cf964f



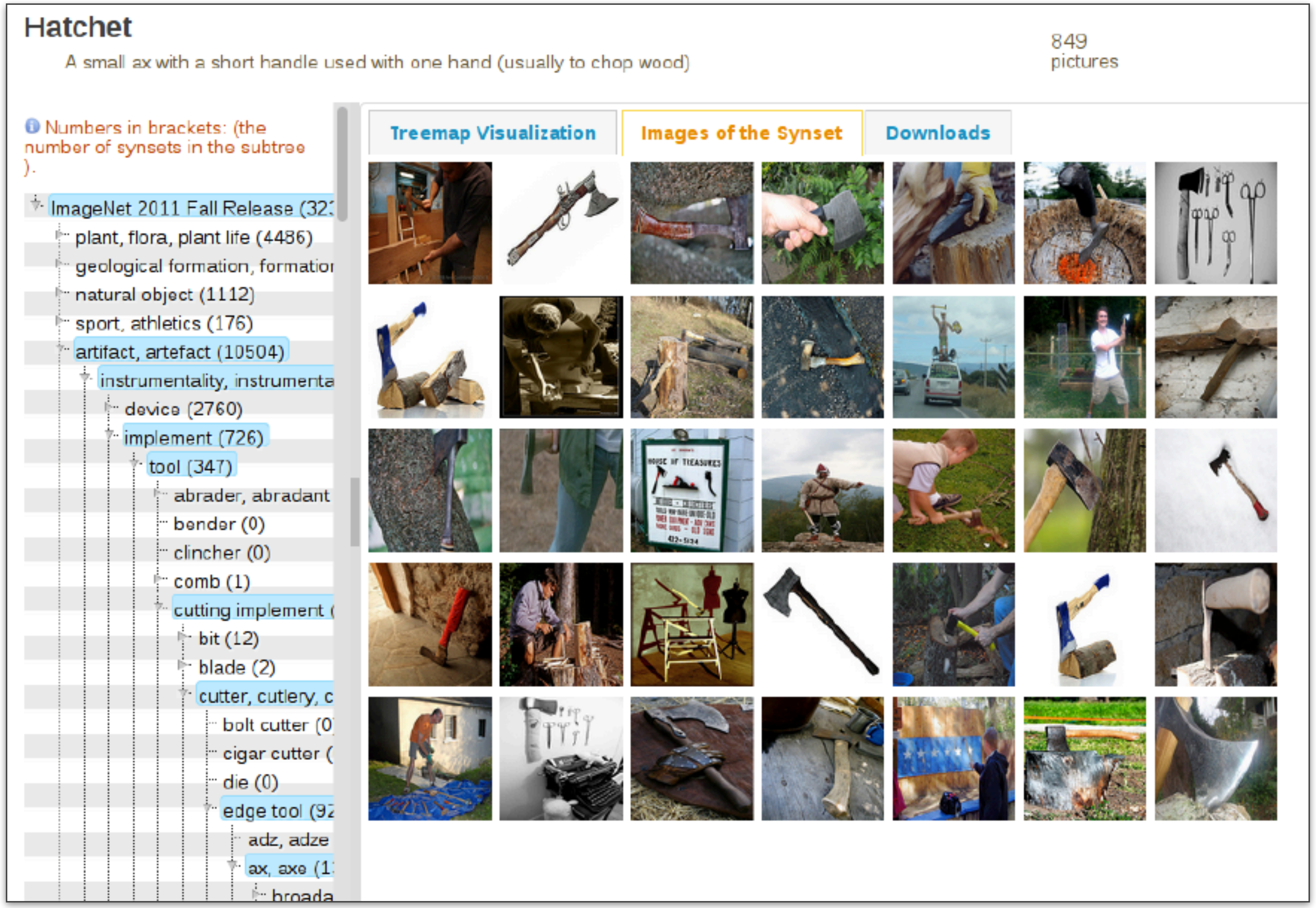
Let the NN also learn the filters.

Deep Learning (DL)

https://fleuret.org/dlc/#lectures

ImageNet Large Scale Visual Recognition Challenge

In 2010, 1M images,
1k categories.

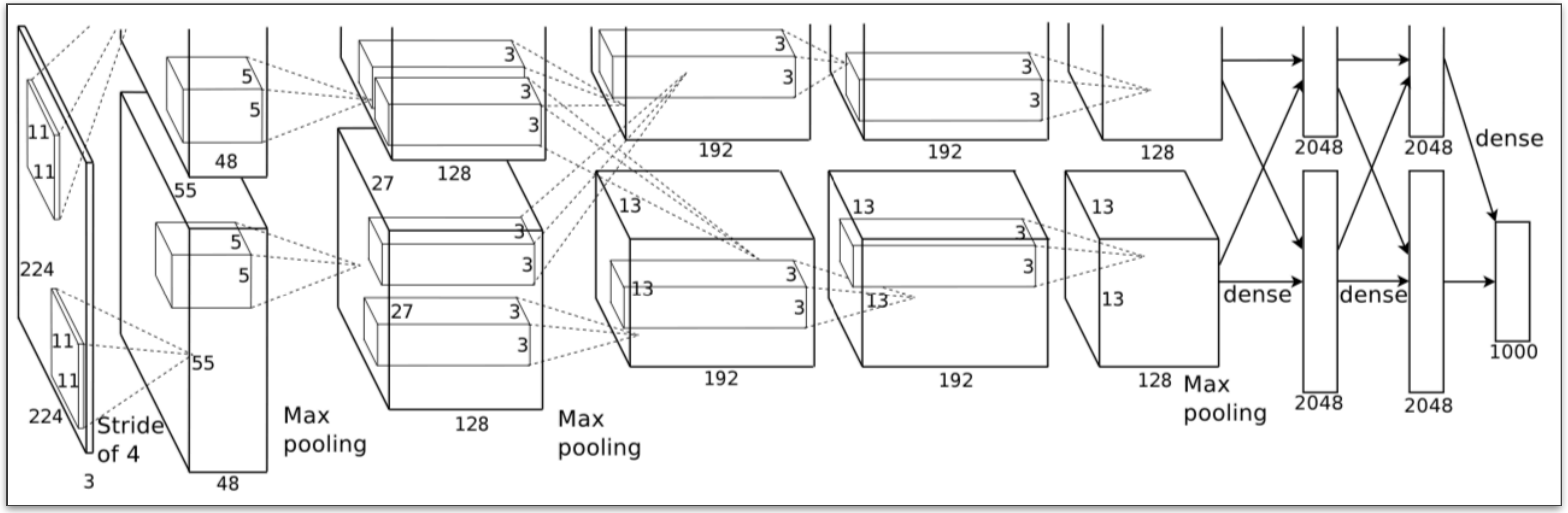


Deep Learning (DL)

AlexNet

In 2012, Krizhevsky et al. employed AlexNet to the challenge.

They used
Graphical Processing
Units (GPU) in the process.

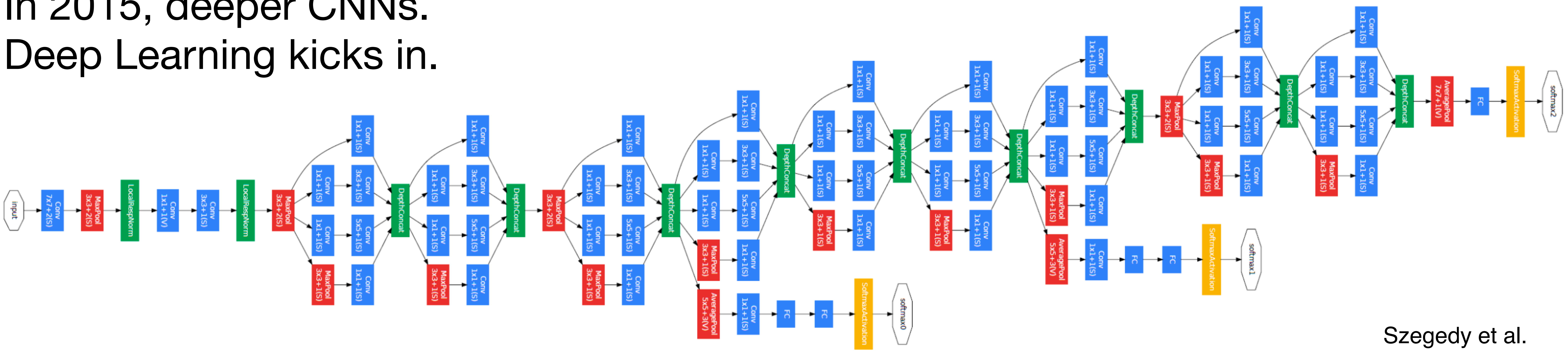


Krizhevsky et al.

Deep Learning (DL)

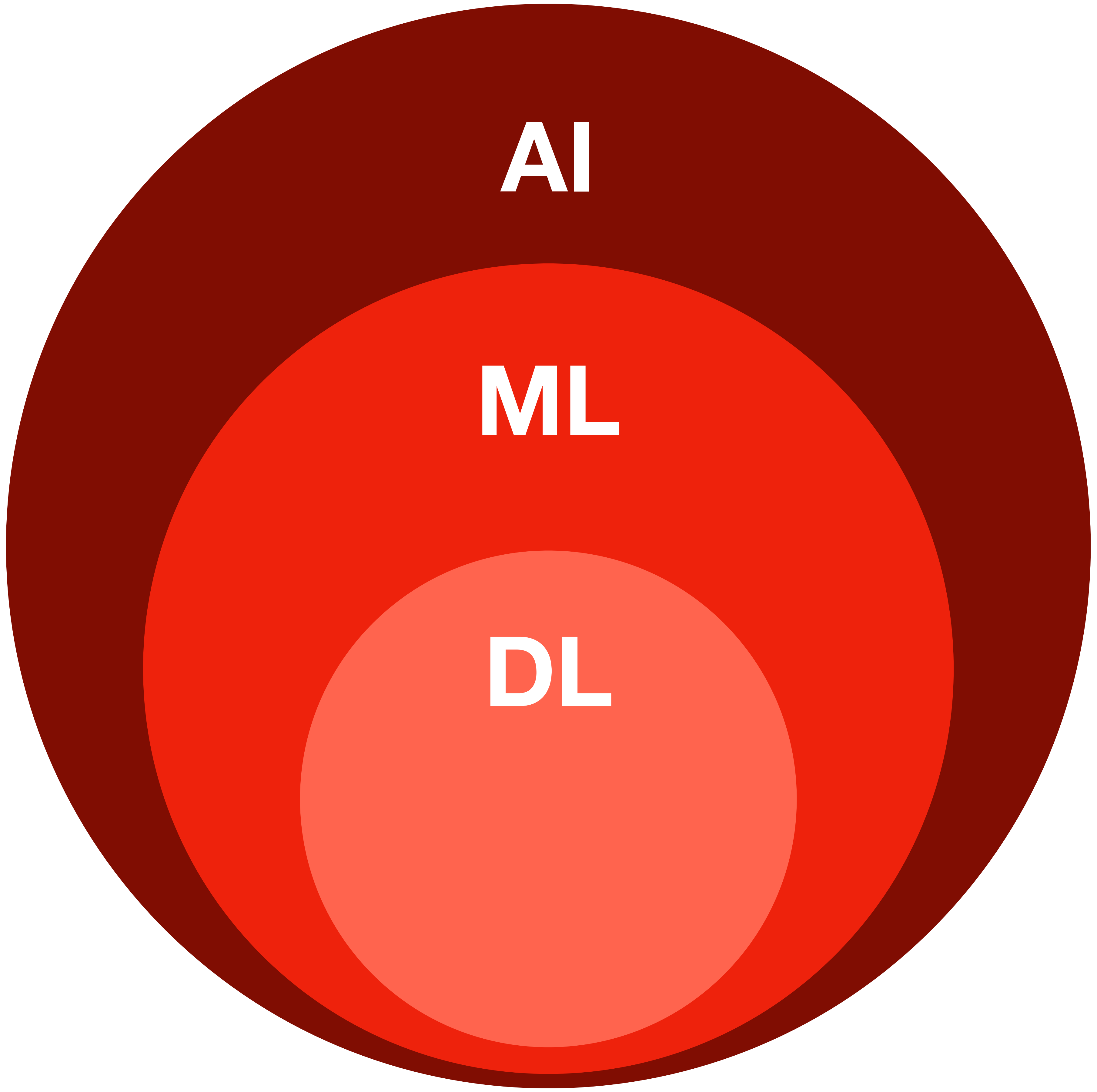
GoogleNet and Others

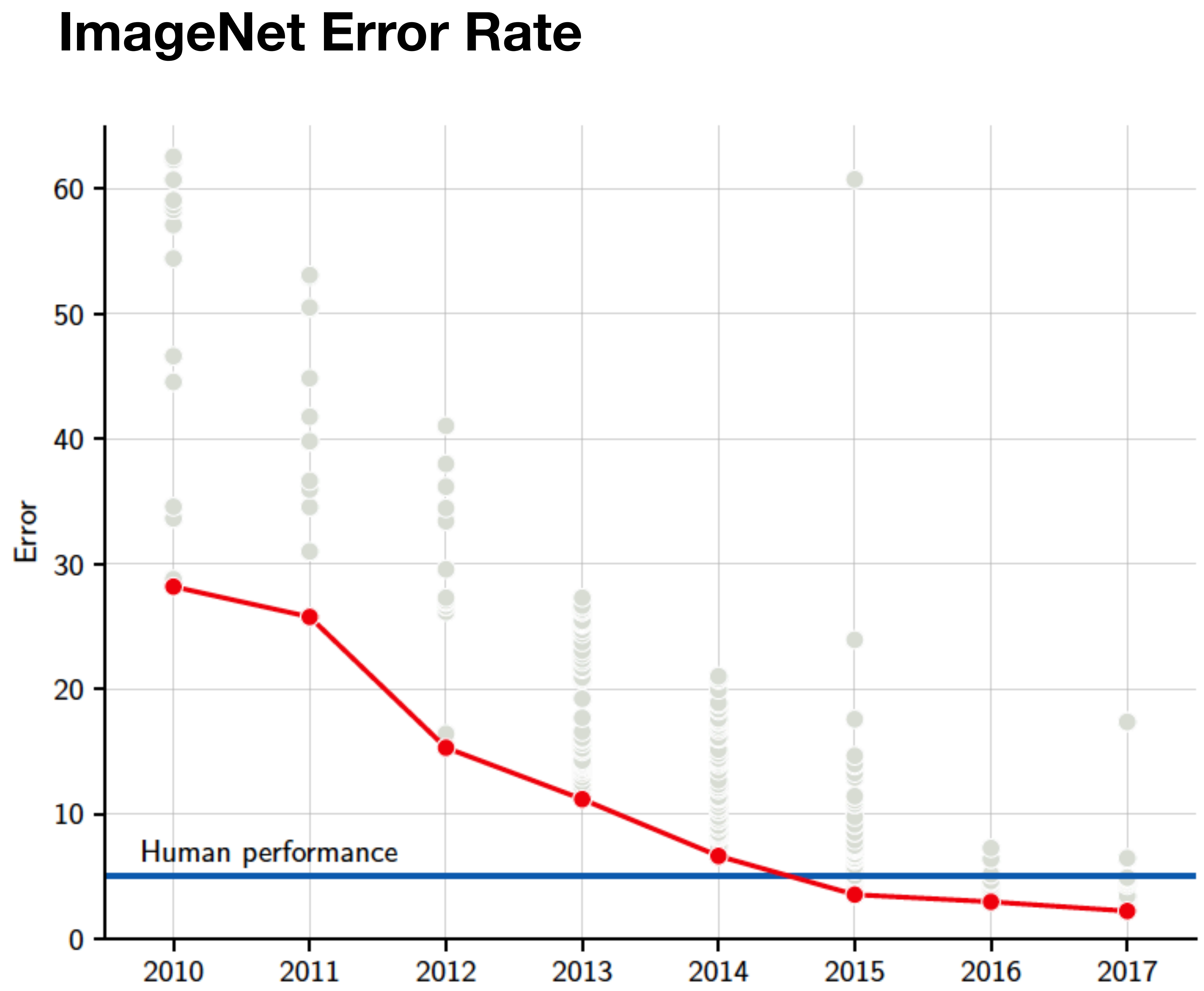
In 2015, deeper CNNs.
Deep Learning kicks in.



Szegedy et al.

Deep Learning (DL)

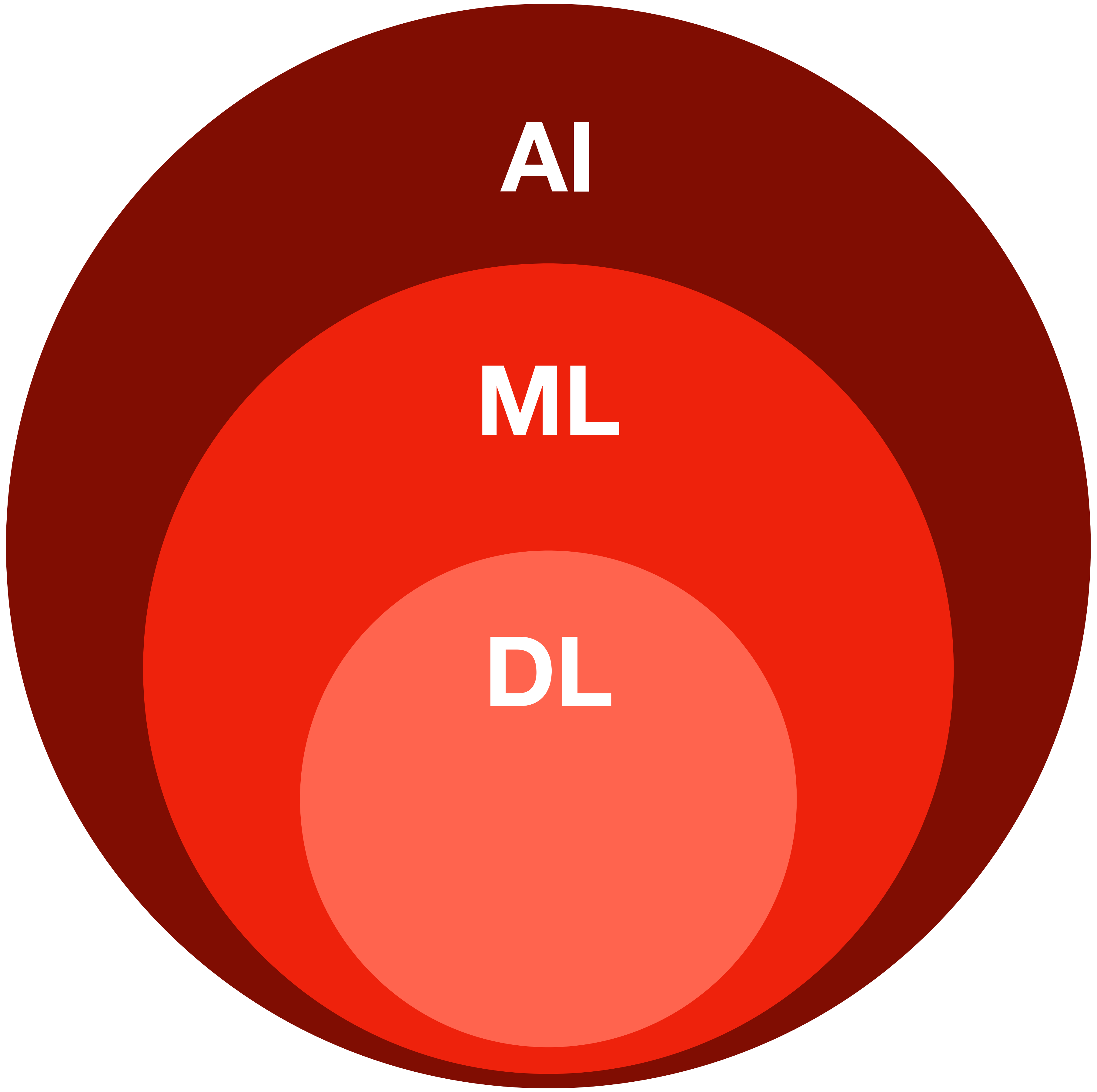


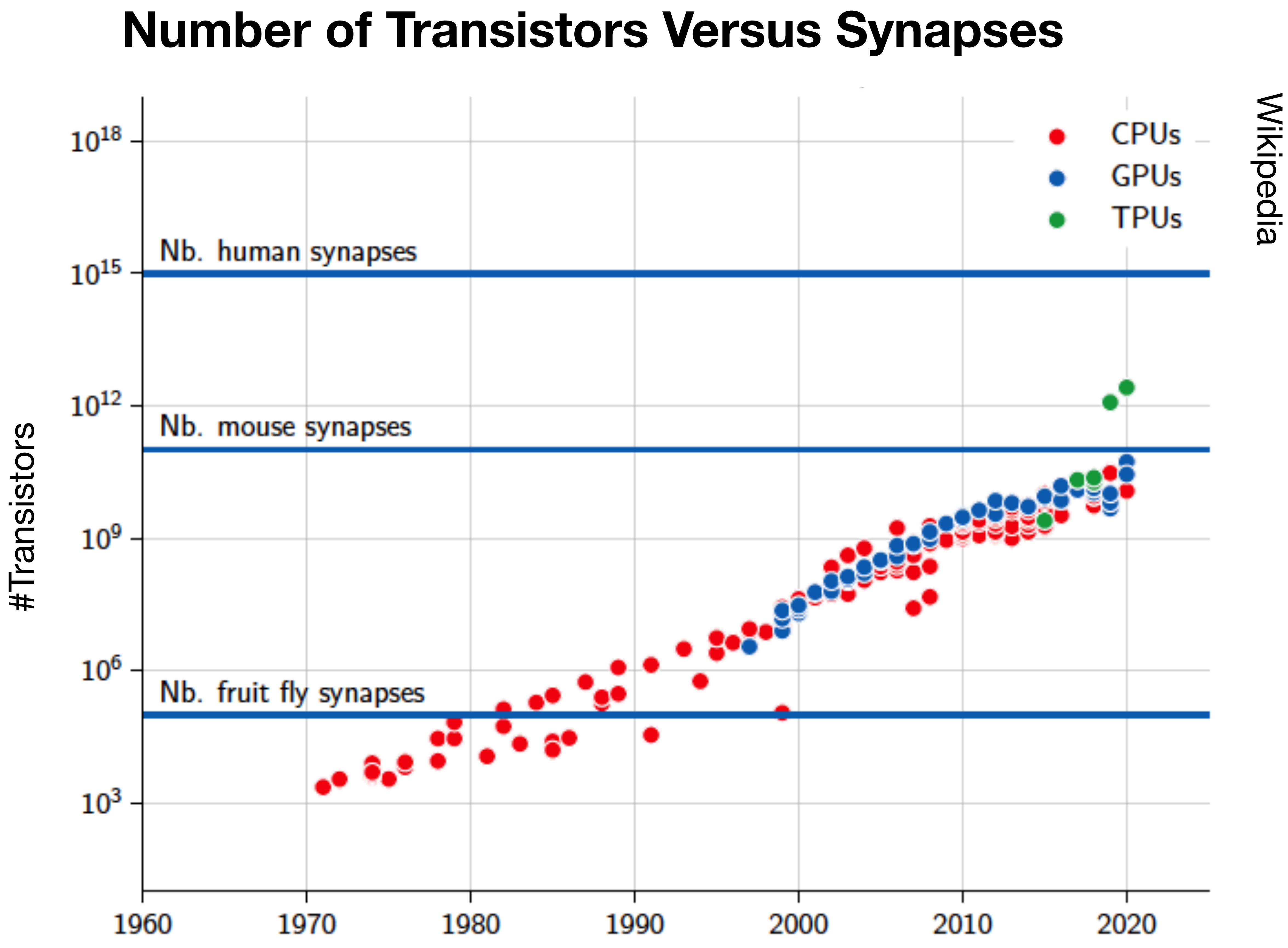


Gershgorn, 2017

LOYOLA UNIVERSITY CHICAGO

Deep Learning (DL)





TPU: Tensor Processing Unity

LOYOLA UNIVERSITY CHICAGO

Deep Learning (DL)

Available Libraries

	Language(s)	License	Main backer
PyTorch	**Python**, C++	BSD	Facebook
TensorFlow	Python, C++	Apache	Google
JAX	Python	Apache	Google
MXNet	Python, C++, R, Scala	Apache	Amazon
CNTK	Python, C++	MIT	Microsoft
Torch	Lua	BSD	Facebook
Theano	Python	BSD	U. of Montreal
Caffe	C++	BSD 2 clauses	U. of CA, Berkeley

https://fleuret.org/dlc/#lectures

LOYOLA UNIVERSITY CHICAGO

Deep Learning (DL)

Pros and Cons

Pros
Deep NNs are powerful tools.
They may have tens of millions of degrees of freedom.

They can approximate any continuous function and be fed with annotated digital images.

Cons
They are data hungry.
They need a massive amount of data to be trained.

They may become black boxes with hard inner understanding.
Why are they working or failing?

LOYOLA UNIVERSITY CHICAGO

Be Careful

August 14, 2019

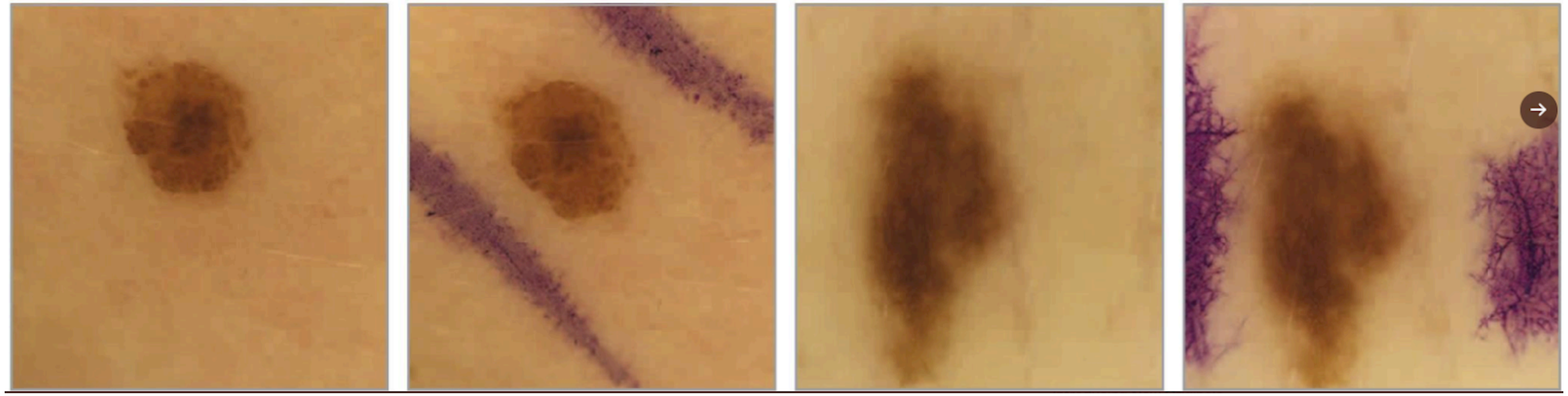
Association Between Surgical Skin Markings in Dermoscopic Images and Diagnostic Performance of a Deep Learning Convolutional Neural Network for Melanoma Recognition

Julia K. Winkler, MD[1]; Christine Fink, MD[1]; Ferdinand Toberer, MD[1]; et al

» Author Affiliations | Article Information

JAMA Dermatol. 2019;155(10):1135-1141. doi:10.1001/jamadermatol.2019.1735

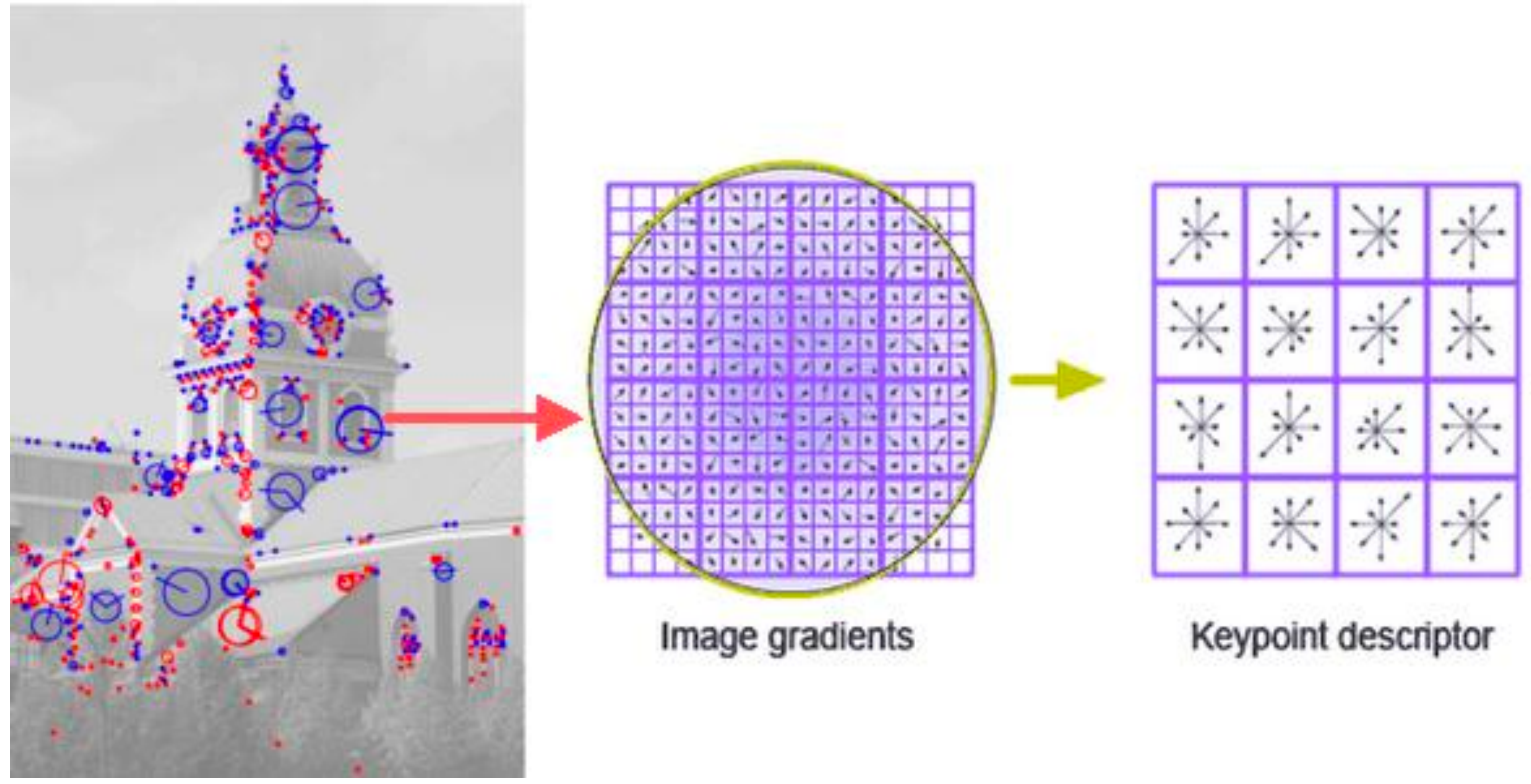
What is the network learning?



What's Next?

Next Class
Local and Global Image Descriptors
(Daniel's presentation).

Sakai is up!
The assignments, content of the classes,
and reference papers are being posted there.



https://www.codeproject.com/Articles/619039/
Bag-of-Features-Descriptor-on-SIFT-Features-with-O

Start working on your 2 seminars
I'll announce today the groups based on your answers.
Count on me during office hours (and outside of them)
to prepare your seminars.

LOYOLA UNIVERSITY CHICAGO