FACULTAD DE MEDICINA
UNIVERSIDAD DE CHILE

MAURICIO CERDA + ASHISH MAHABAL

DEEP LEARNING

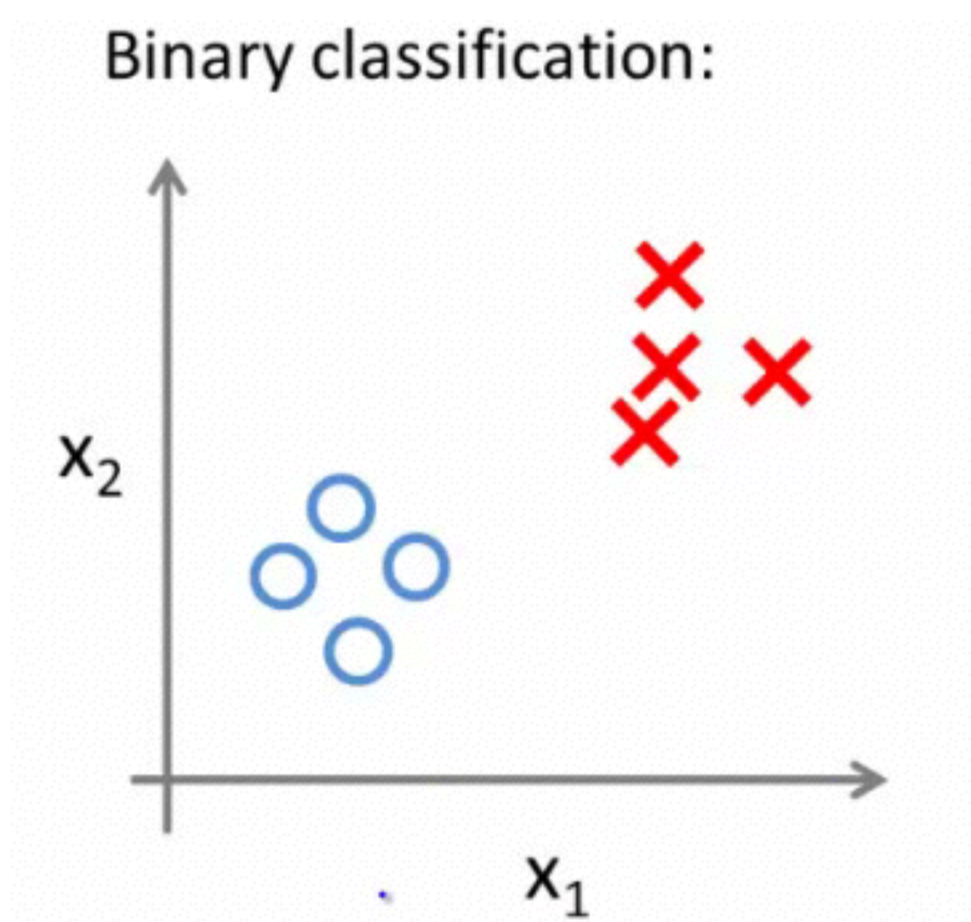
- La Serena, 8/28/2017 -

Outline

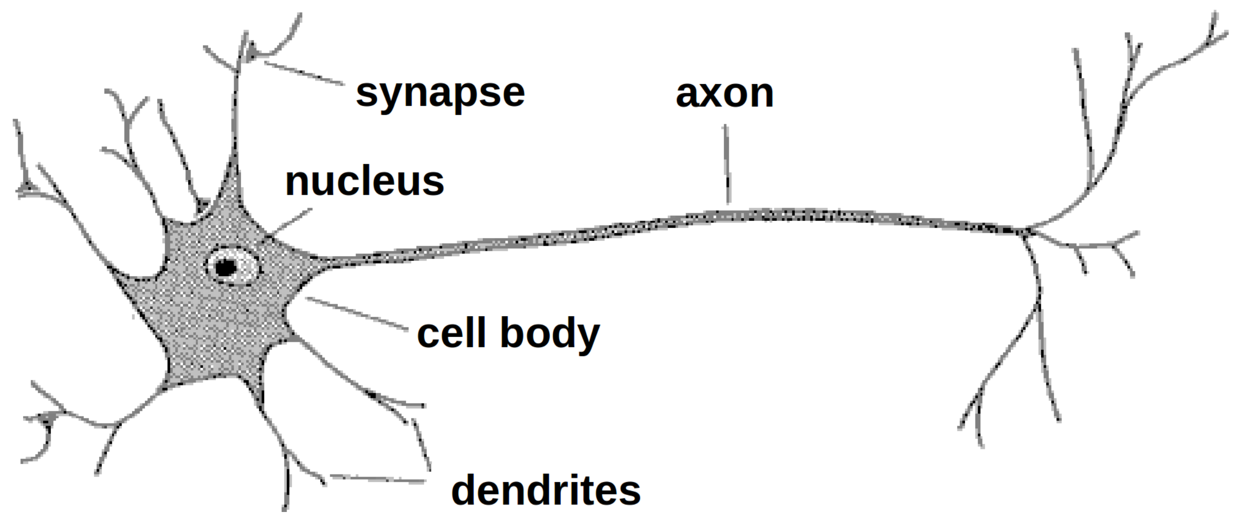
- Perceptron & Multilayer Perceptron
- Deep Learning
- Demo

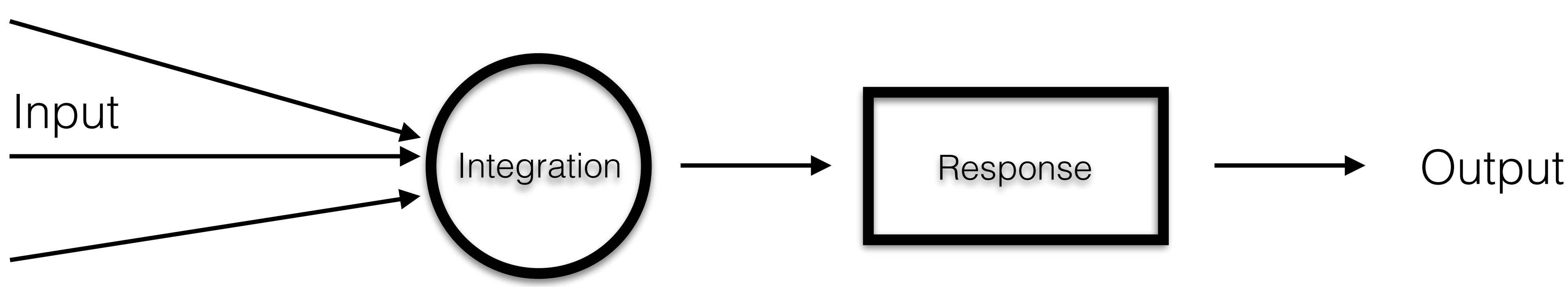
Supervised learning

- Objective: learn input/output association.



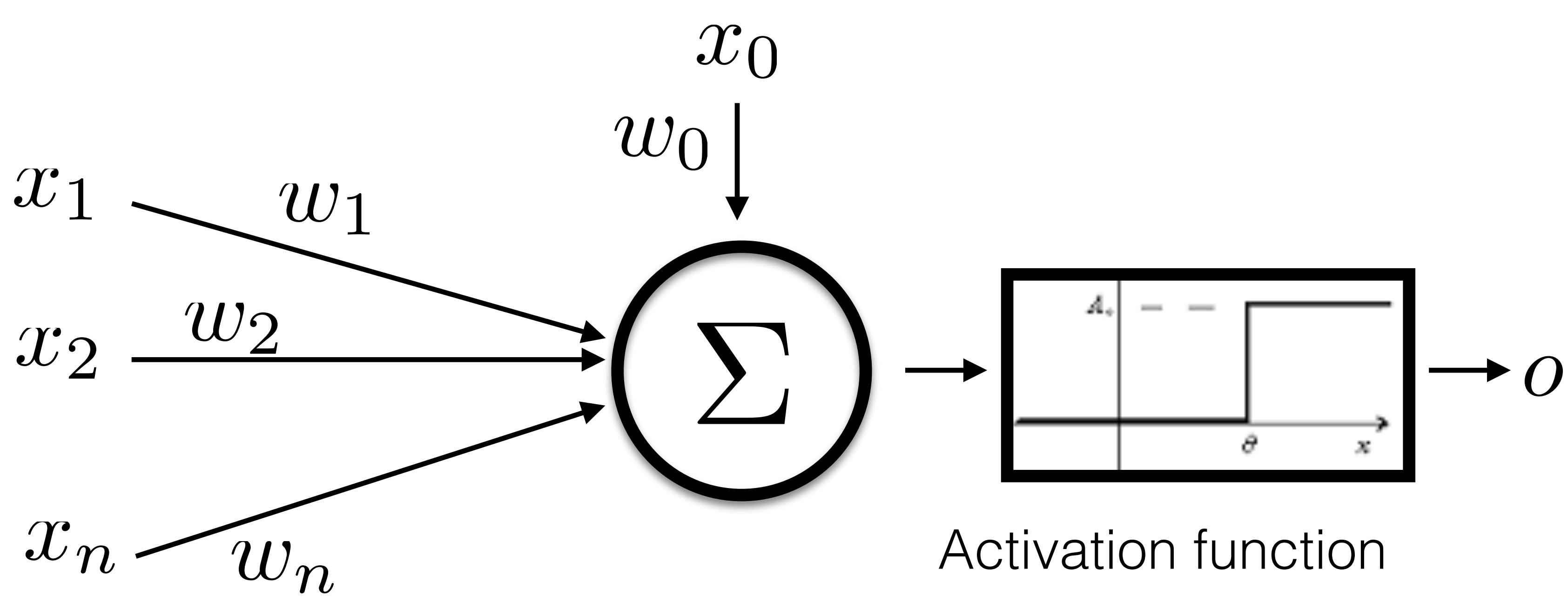
The human brain

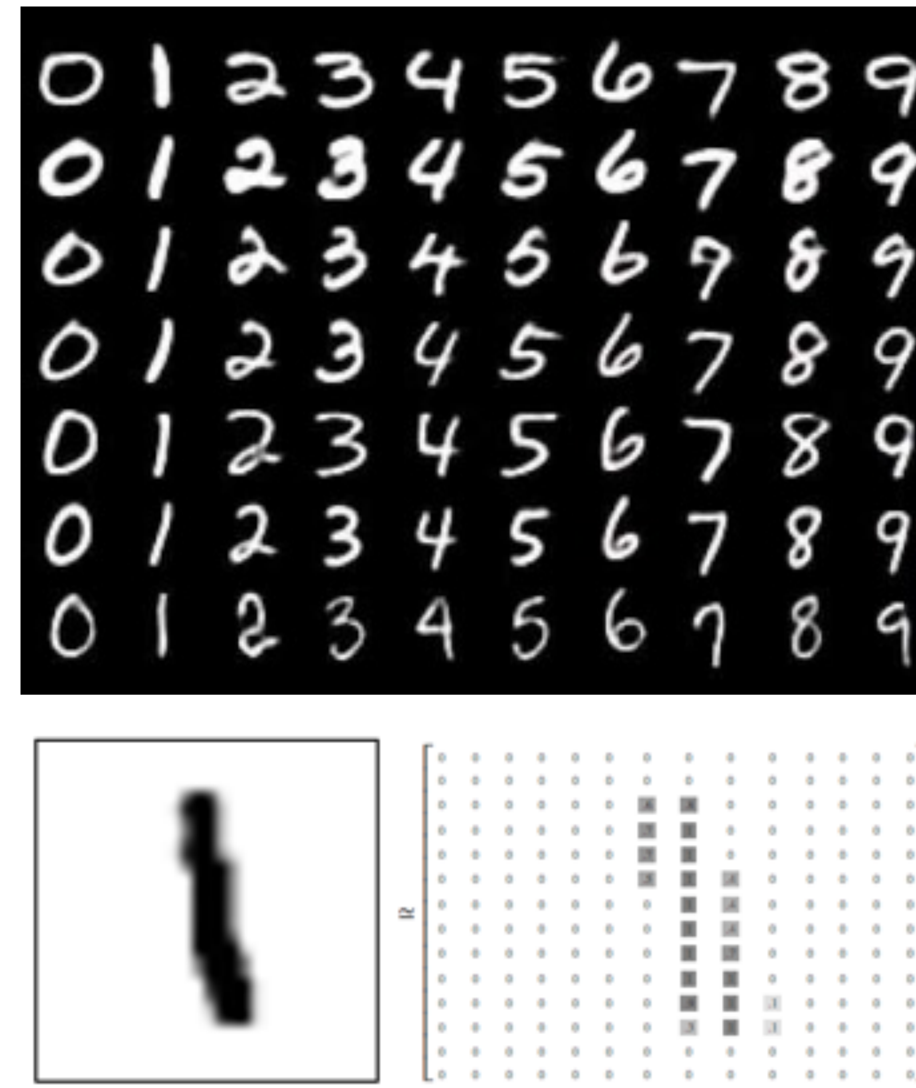




Perceptron

- A (functional) model of how neurons work.





MNIST digits

0100000000

Real and artificial neurons

Learning rule

- How to learn with a perceptron?

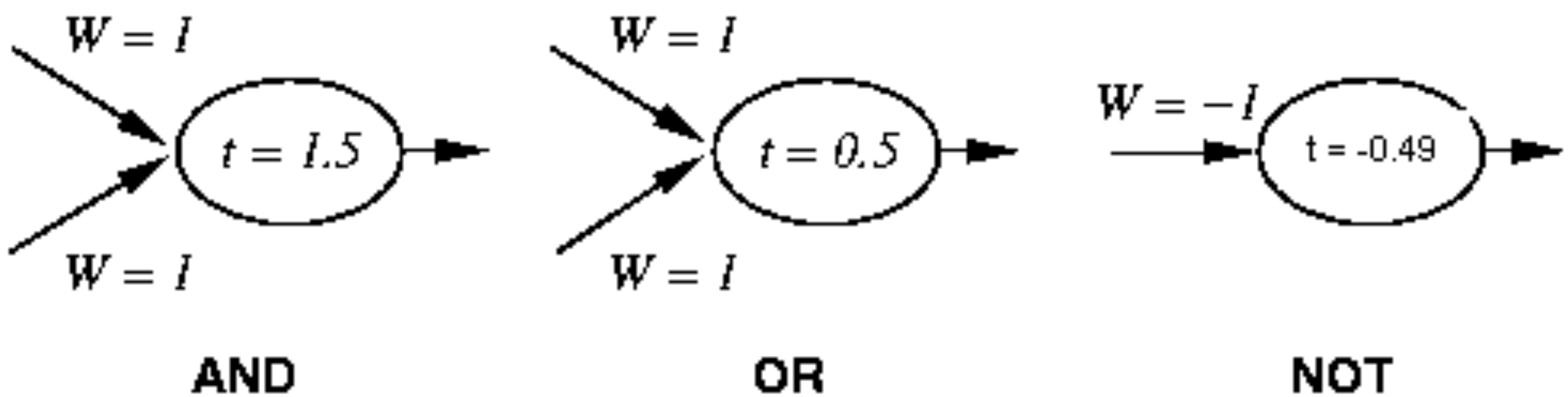
$$w_i = w_i + \Delta w_i$$
$$\Delta w_i = \rho(t - o)x_i$$

where t is the objective, ρ is the learning rate, o is perceptron output.

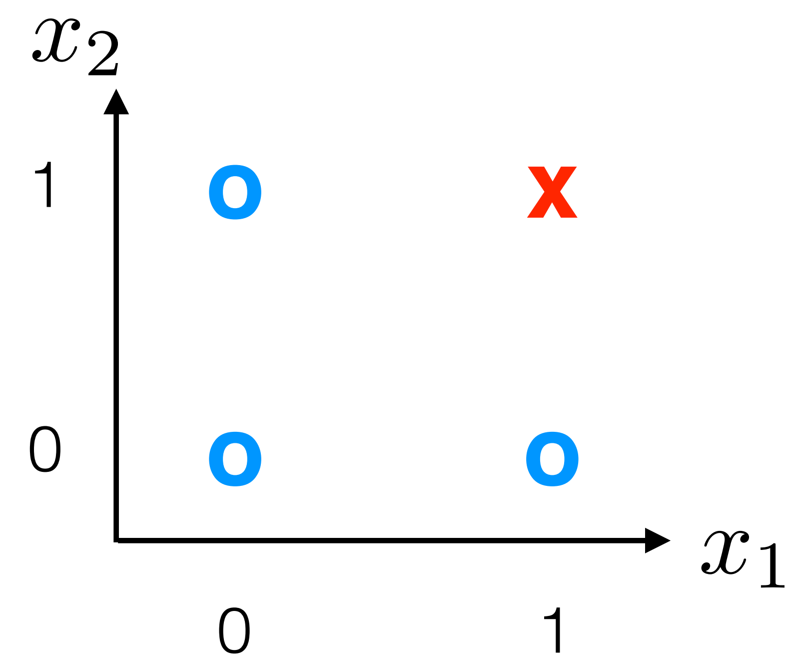
- How does it work?
- If the output is correct (t= o), w does not change.
- If the output is incorrect (t != o), w will change to make the output as similar as possible to the objective.
- The algorithm will converge if:
  - Data is linearly separable.
  - ρ is small enough

Example perceptron

- A few examples:



Perceptron



- Training the AND operation:

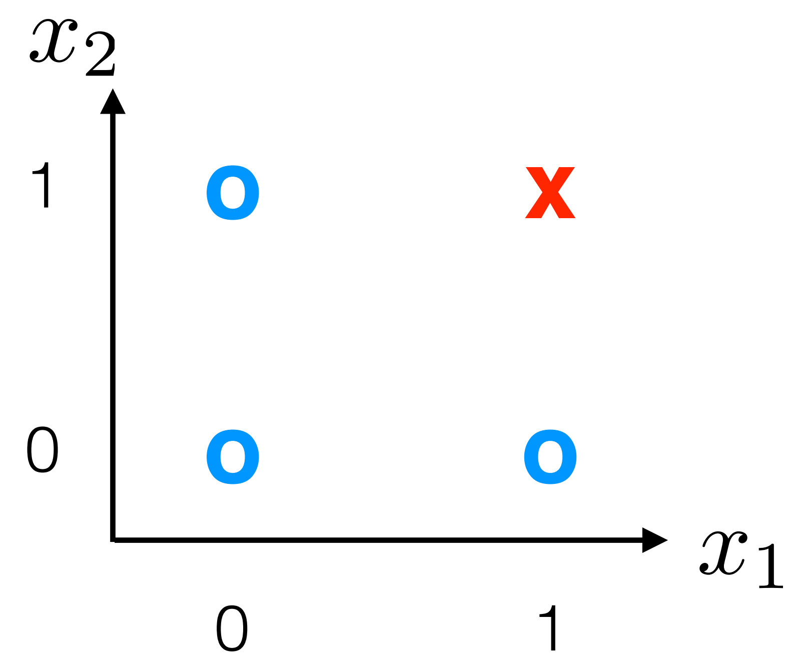
Iteration 1, f(x)=x>0.5, w=(0.1, 0.2, 0.3) $\rho = 0.1$

x1	x2	x3	<w,x>	o	t
-1	0	0		0	0
-1	0	1		0	0
-1	1	0		0	0
-1	1	1		**0**	**1**

Iteration 2, f(x)=x>0.5, w=(, ,) $\rho = 0.1$

x1	x2	x3	<w,I>	o	t
-1	0	0		0	0
-1	0	1		0	0
-1	1	0		0	0
-1	1	1		1	1

Perceptron



- Training the AND operation:

Iteration 1, f(x)=x>0.5, w=(0.1, 0.2, 0.3) $\rho = 0.1$

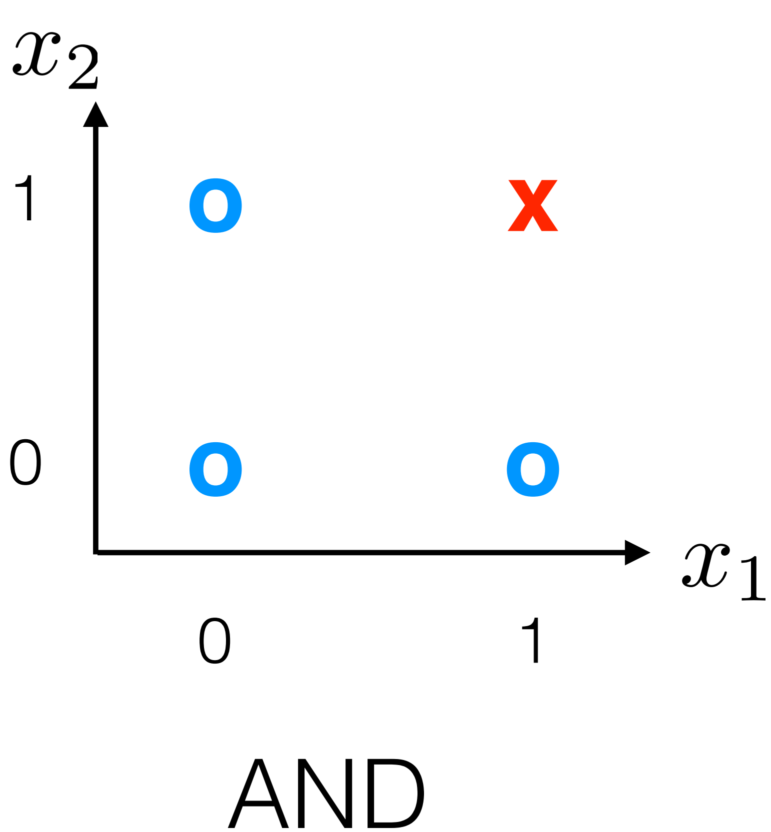
x1	x2	x3	<w,x>	o	t
-1	0	0	0.1*-1+0.2*0+0.3*0	0	0
-1	0	1	0.1*-1+0.2*0+0.3*1	0	0
-1	1	0	0.1*-1+0.2*1+0.3*0	0	0
-1	1	1	0.1*-1+0.2*1+0.3*1	**0**	**1**

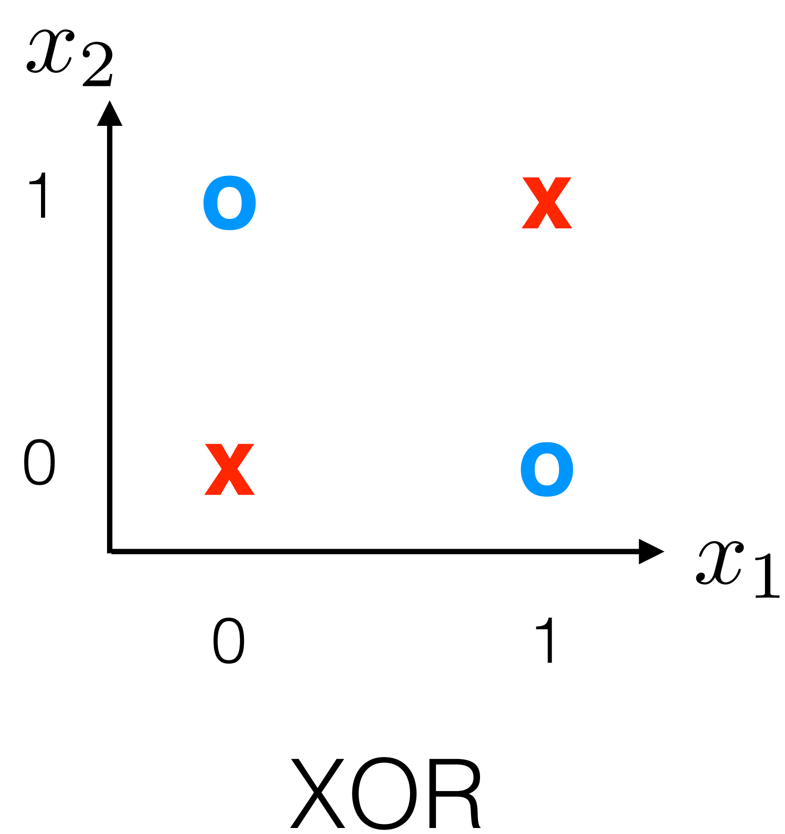
Iteration 2, f(x)=x>0.5, w=(0, 0.3, 0.4) $\rho = 0.1$

x1	x2	x3	<w,l>	o	t
-1	0	0	-1*0+0.3*0+0.4*0	0	0
-1	0	1	-1*0+0.3*0+0.4*1	0	0
-1	1	0	-1*0+0.3*1+0.4*0	0	0
-1	1	1	-1*0+0.3*1+0.4*1	1	1

Perceptron

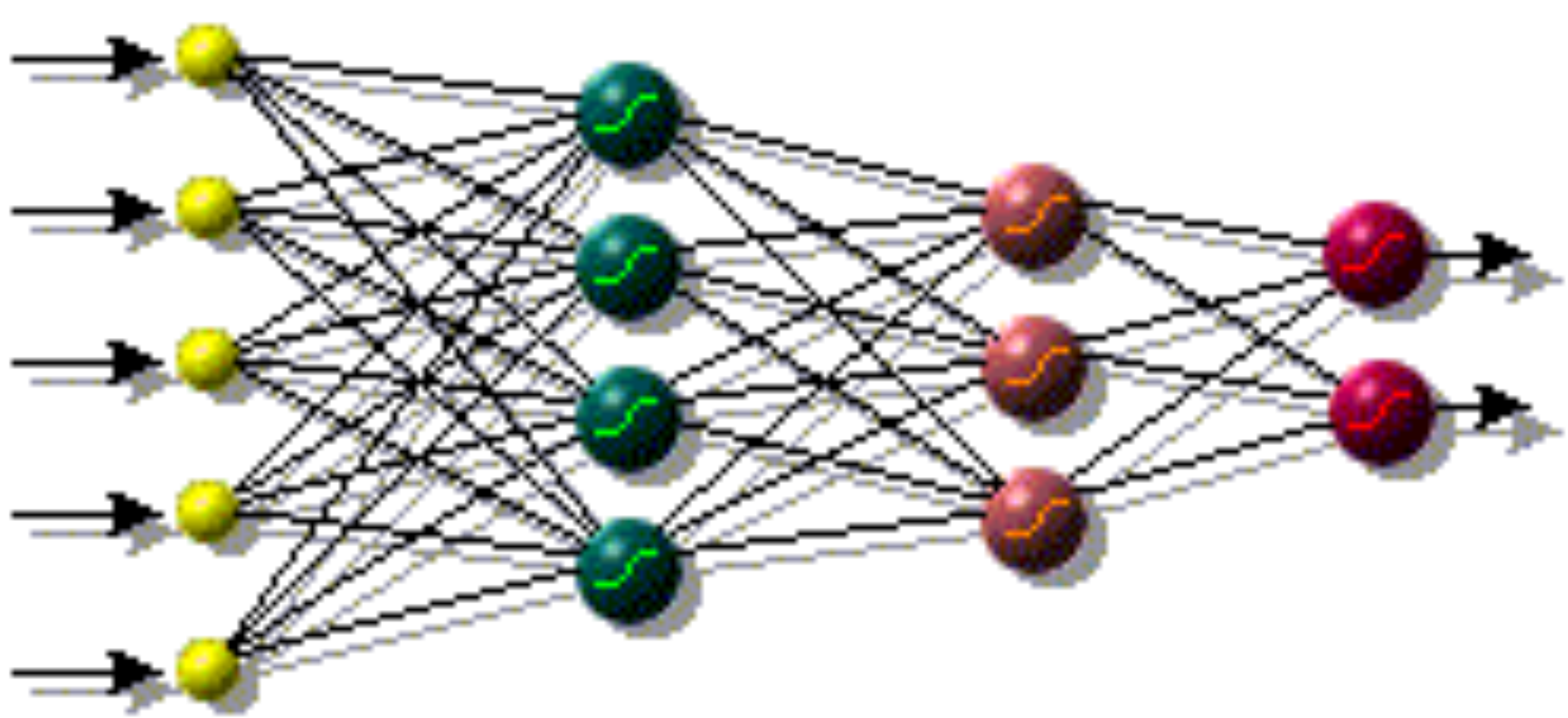
- But perceptron can do only linear separations.
- In the 70-80 researchers hit this problem.



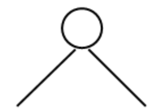
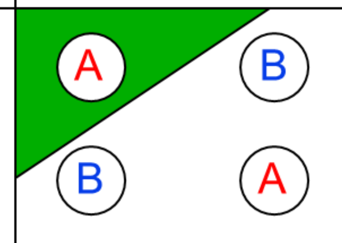
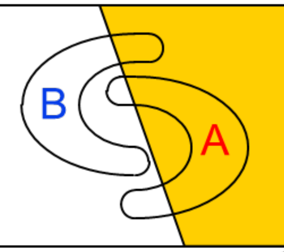
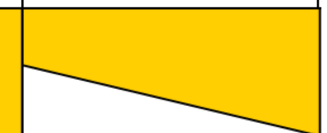
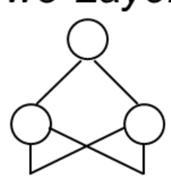
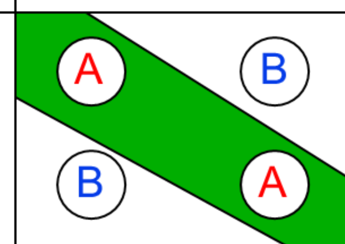
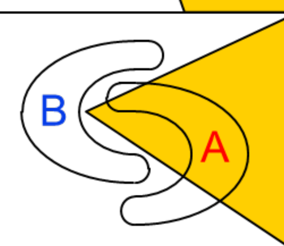
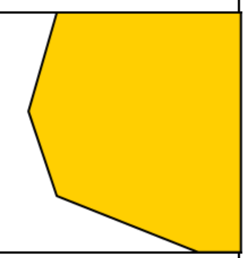
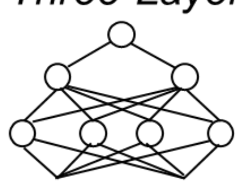
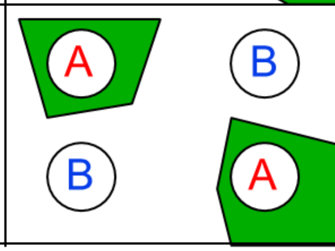
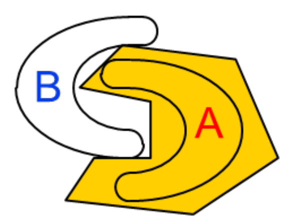
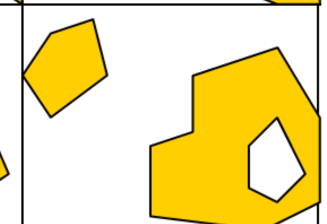


MultiLayer Perceptron

- What about more layers? (*MultiLayer Perceptron o MLP*)
  - For more complex problems
  - Solve classification problems that are not linearly separable
  - Learning must be propagated between layers



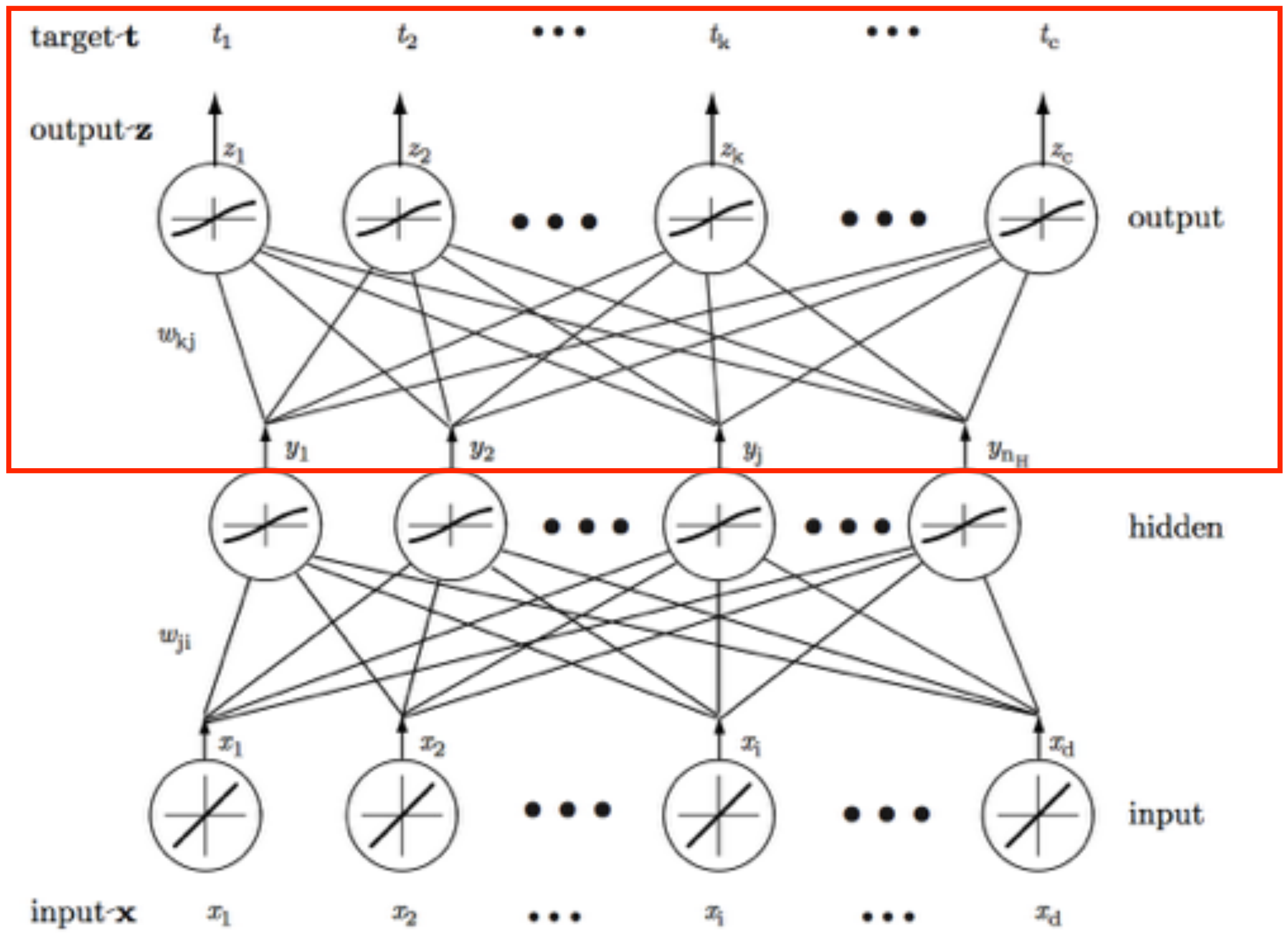
MLP

Structure	Types of Decision Regions	Exclusive-OR Problem	Classes with Meshed regions	Most General Region Shapes
Single-Layer 	*Half Plane Bounded By Hyperplane*			
Two-Layer 	*Convex Open Or Closed Regions*			
Three-Layer 	Arbitrary (Complexity Limited by No. of Nodes)			

- Three layers are enough in theory, but more may be useful in practice.

MLP: Backpropagation

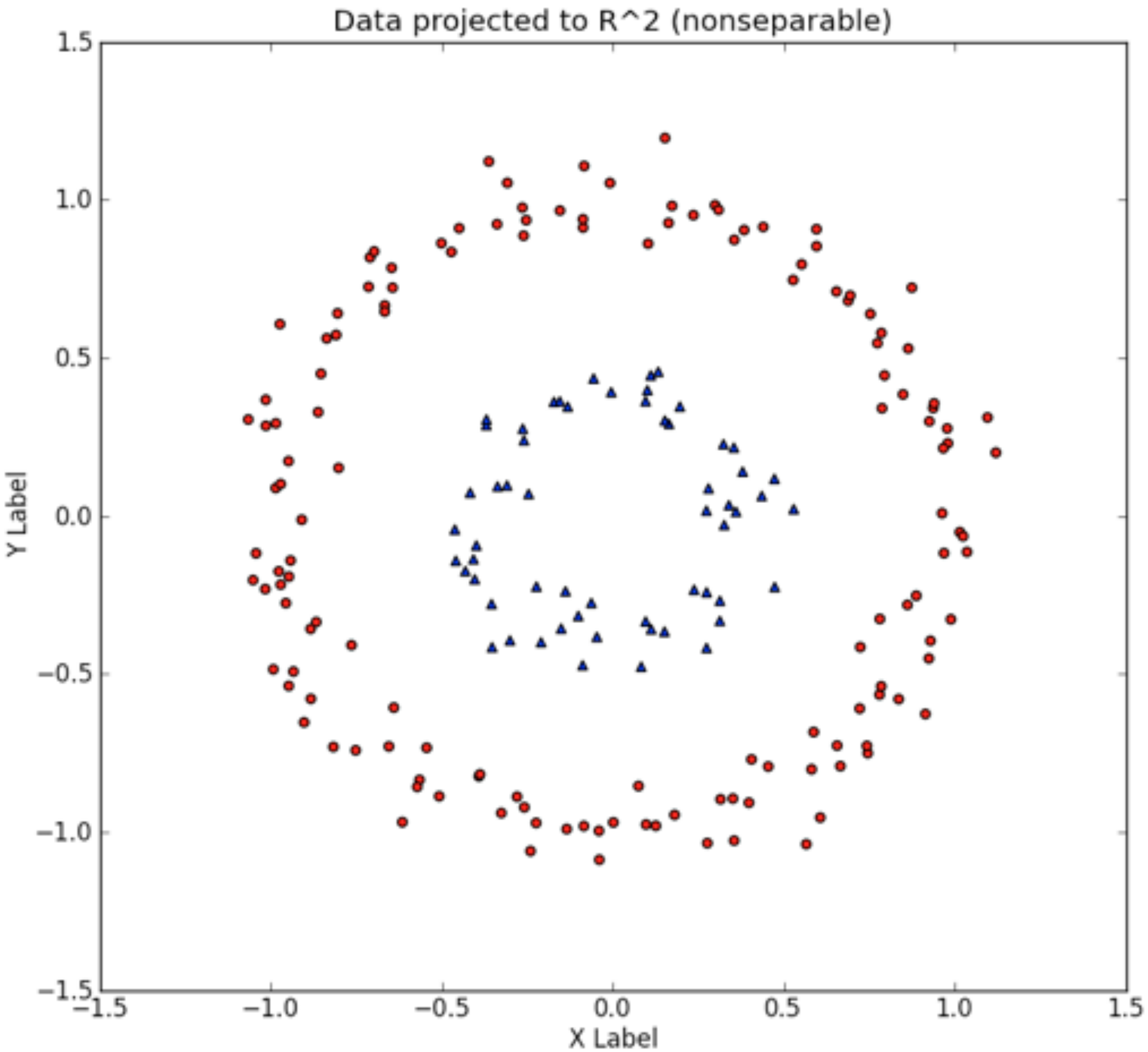
- In a multilayer network:

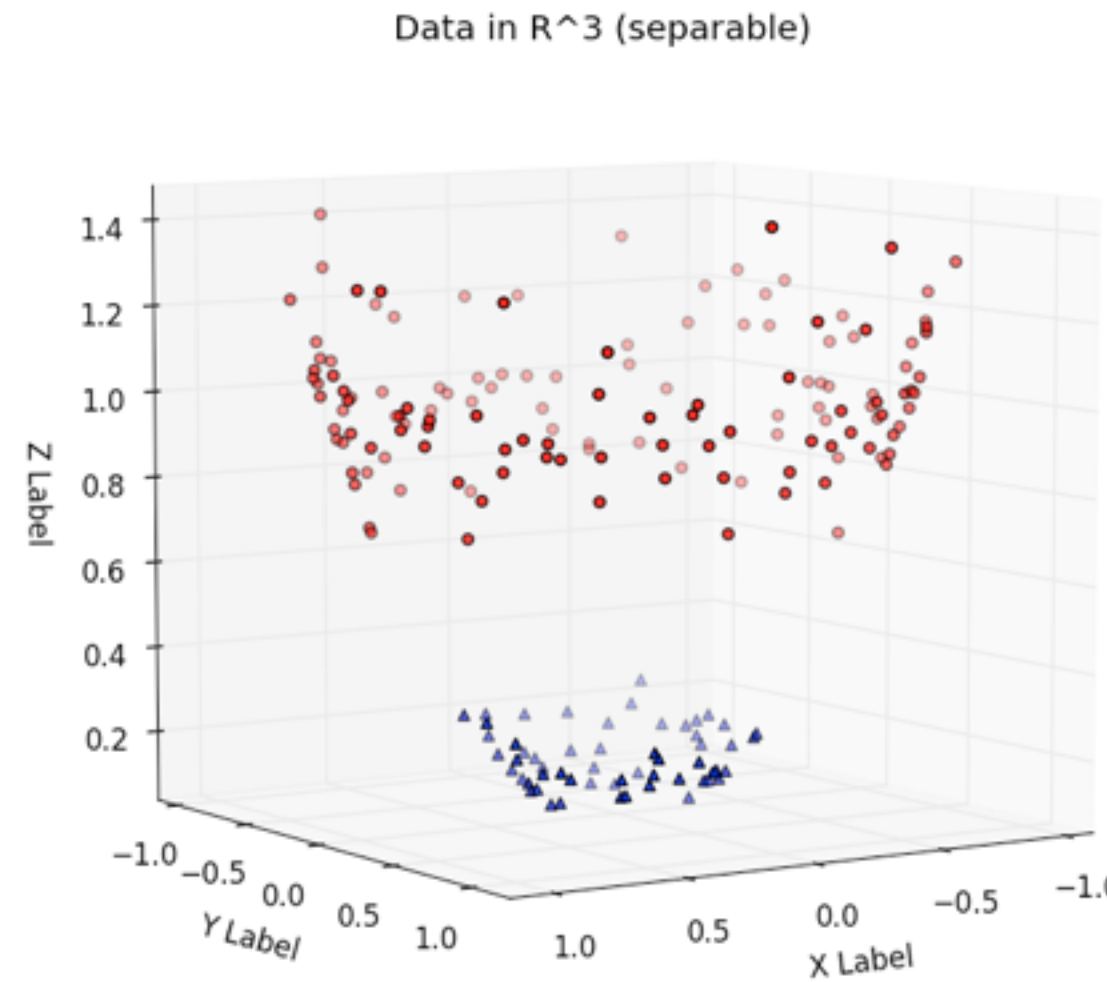


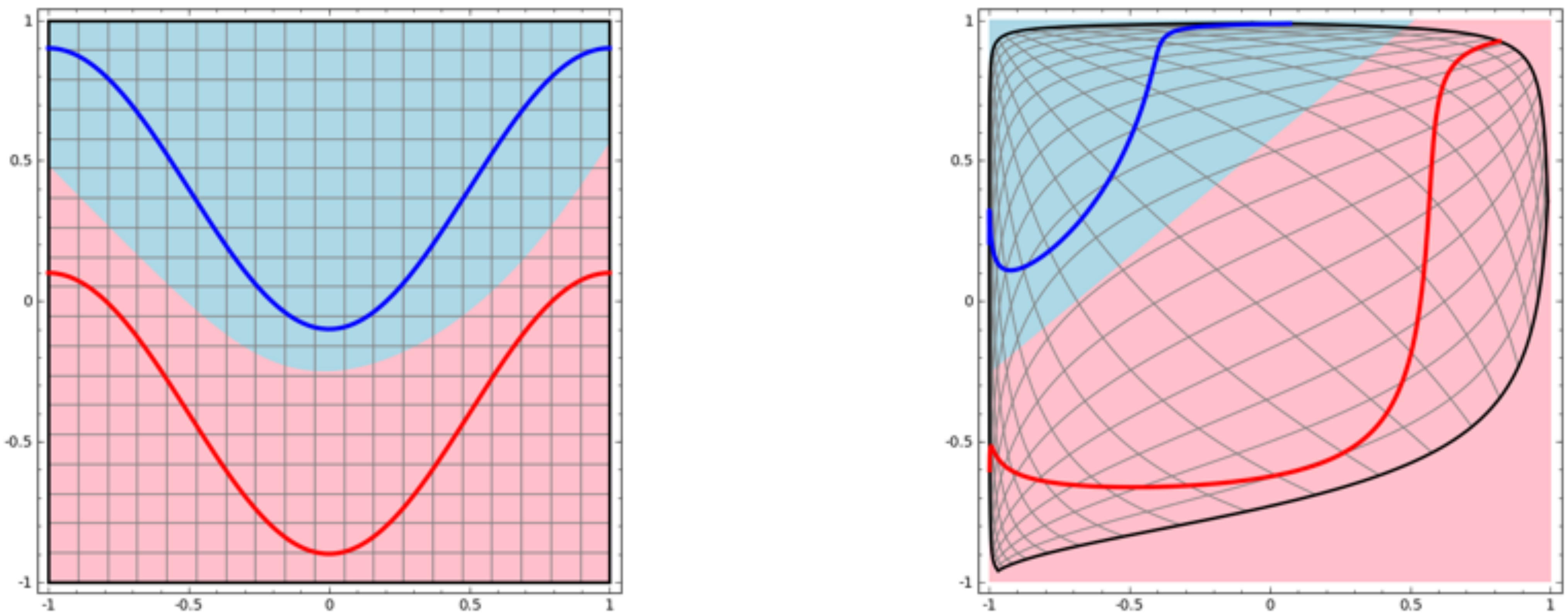
- Learning rule is:

$$\Delta w_{kj} = -\frac{\partial J}{\partial net_k}\frac{\partial net_k}{\partial w_{kj}} = \rho(t_k - z_k)y_j f'(net_k)$$

Non-linear SVM



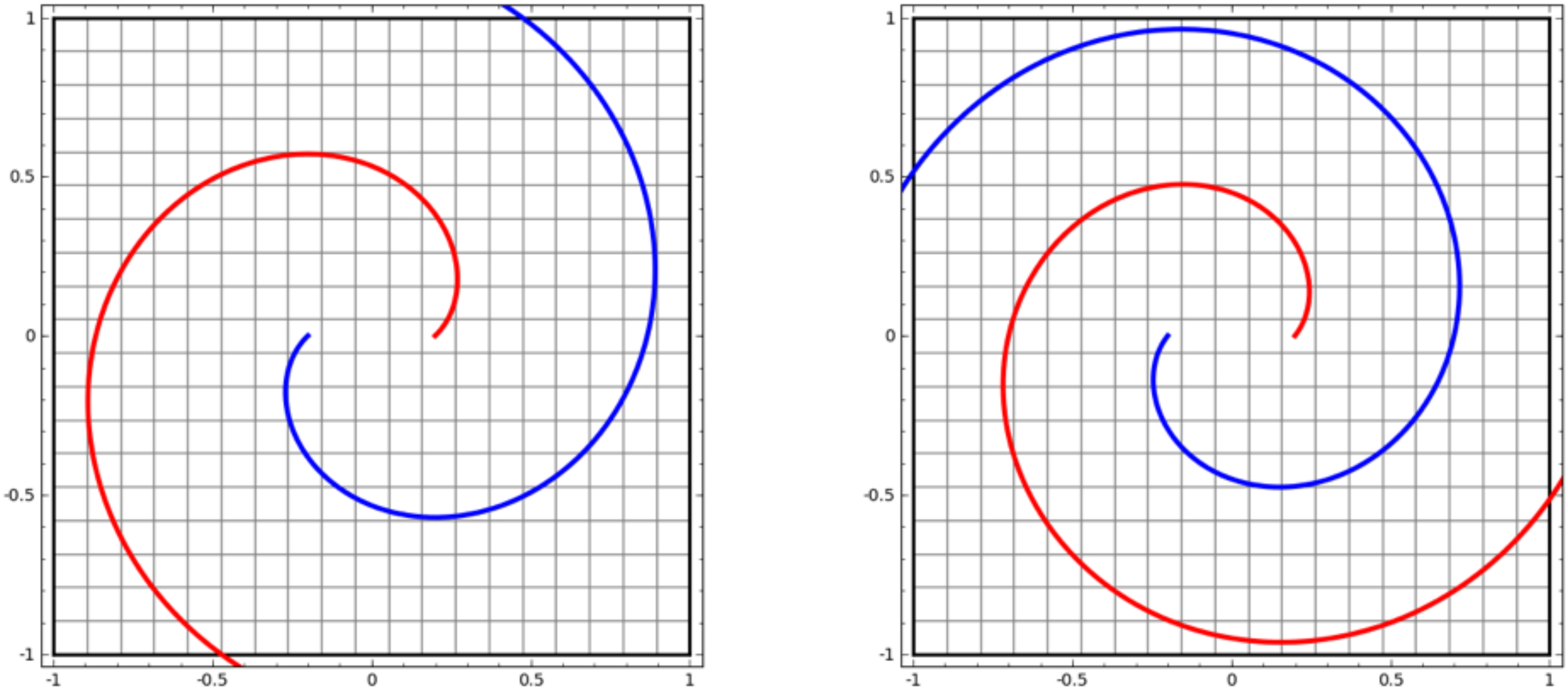




Mapping in order to linearly separate clusters

http://colah.github.io/posts/2014-03-NN-Manifolds-Topology/

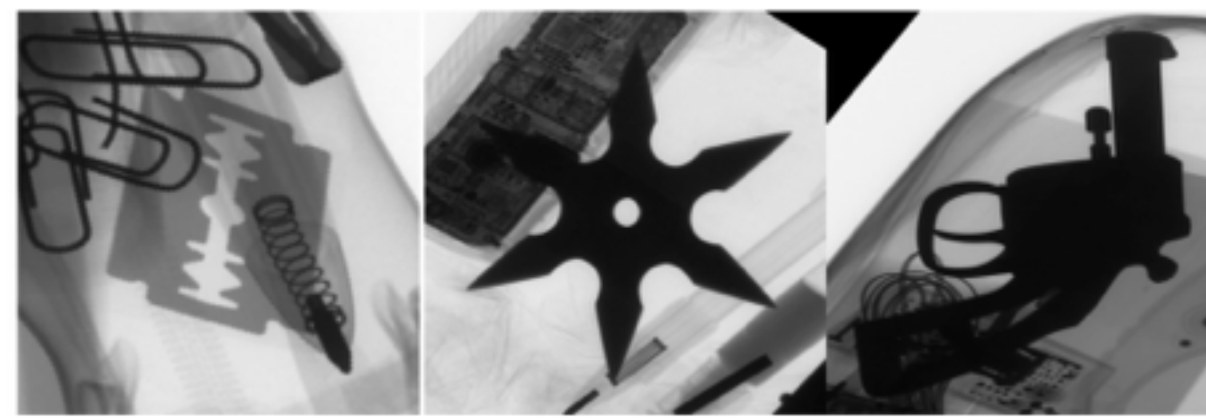
Disentangling with multiple layers



http://colah.github.io/posts/2014-03-NN-Manifolds-Topology/

Easy

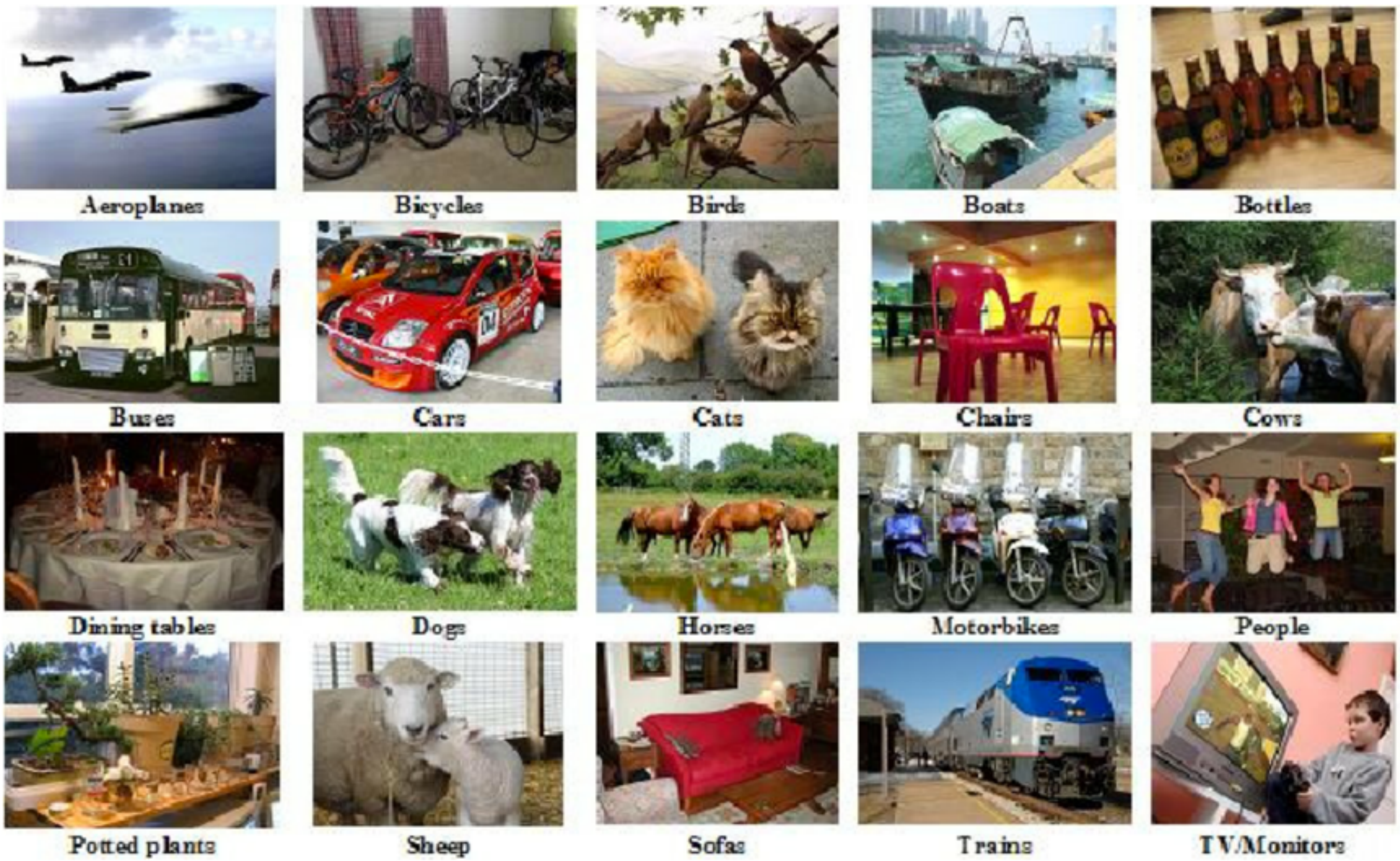
Difficult



Easy

Difficult

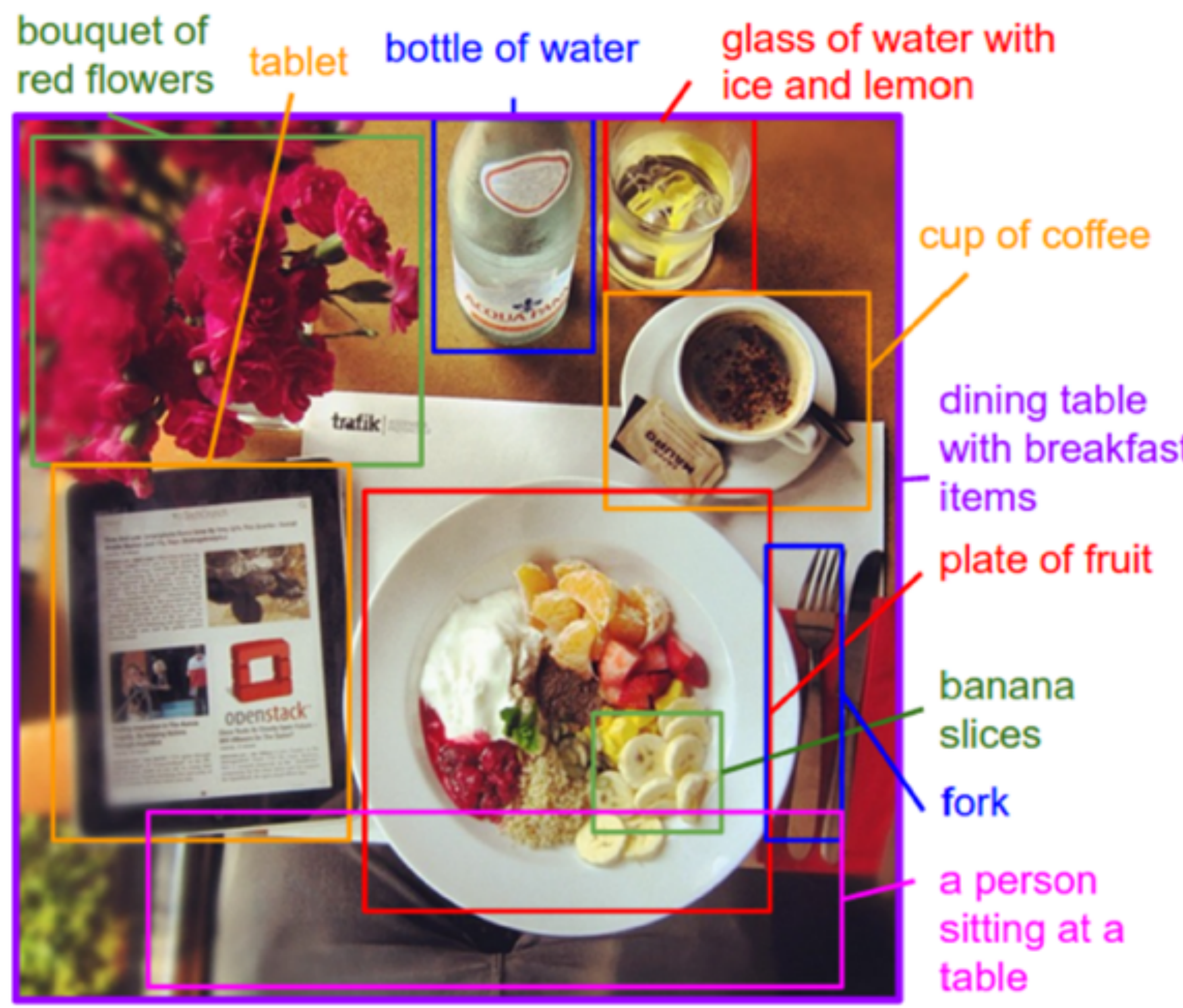
Difficult


Aeroplanes
Bicycles
Birds
Boats
Bottles
Buses
Cars
Cats
Chairs
Cows
Dining tables
Dogs
Horses
Motorbikes
People
Potted plants
Sheep
Sofas
Trains
TV/Monitors

More Difficult

Laundry list for image archives

- Large sets
- Labelled data
- Metadata (**CDEs!**)
- Peripheral data
- Balanced datasets



Example output of the model

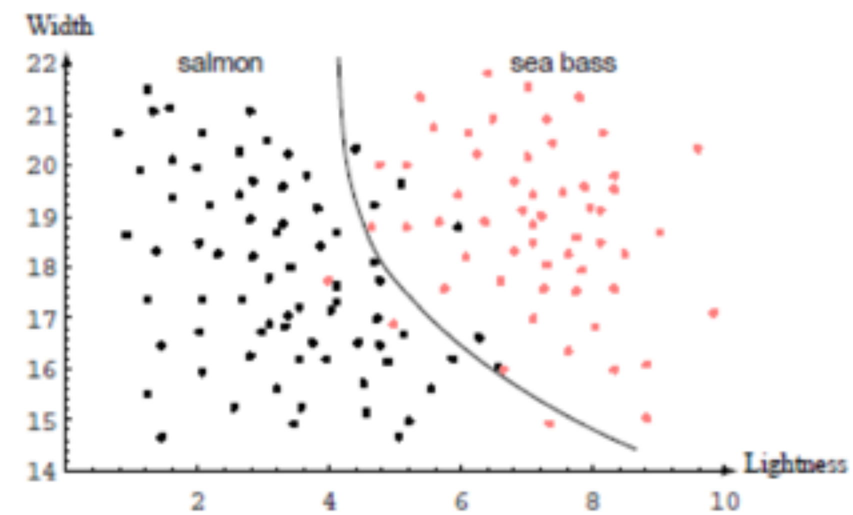
https://arxiv.org/pdf/1412.2306v2.pdf

Standard workflow

Datos

$$[x_1, x_2, ...x_d]^T$$

Características

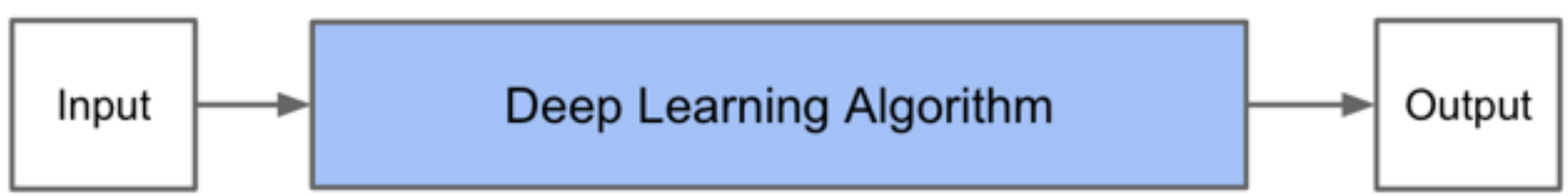


Clasificación

Cats Dogs

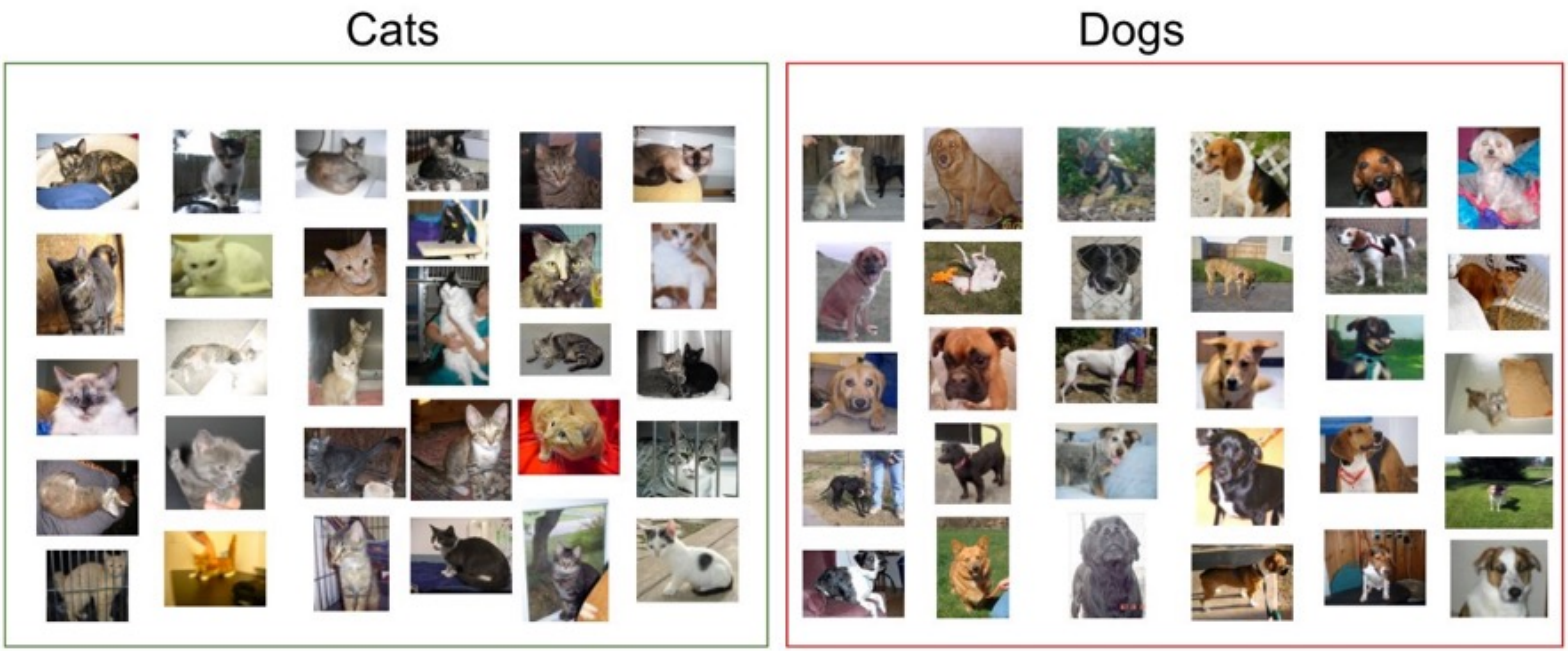
Sample of cats & dogs images from Kaggle Dataset

Traditional Machine Learning Flow



Deep Learning Flow

Adil Moujahid

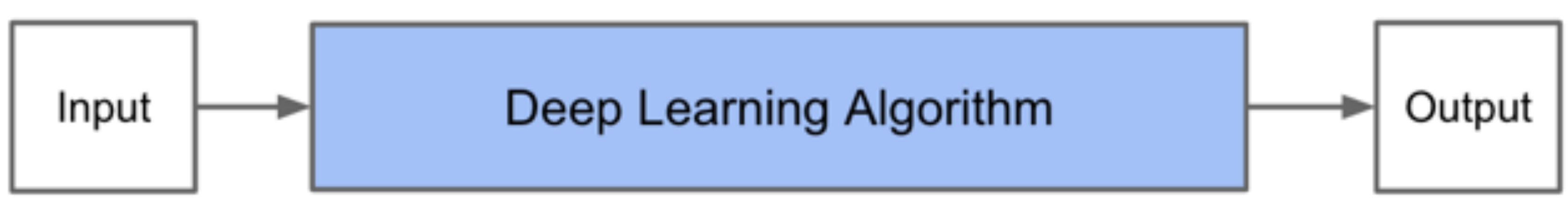


Sample of cats & dogs images from Kaggle Dataset

Promise:
Works better

Traditional Machine Learning Flow

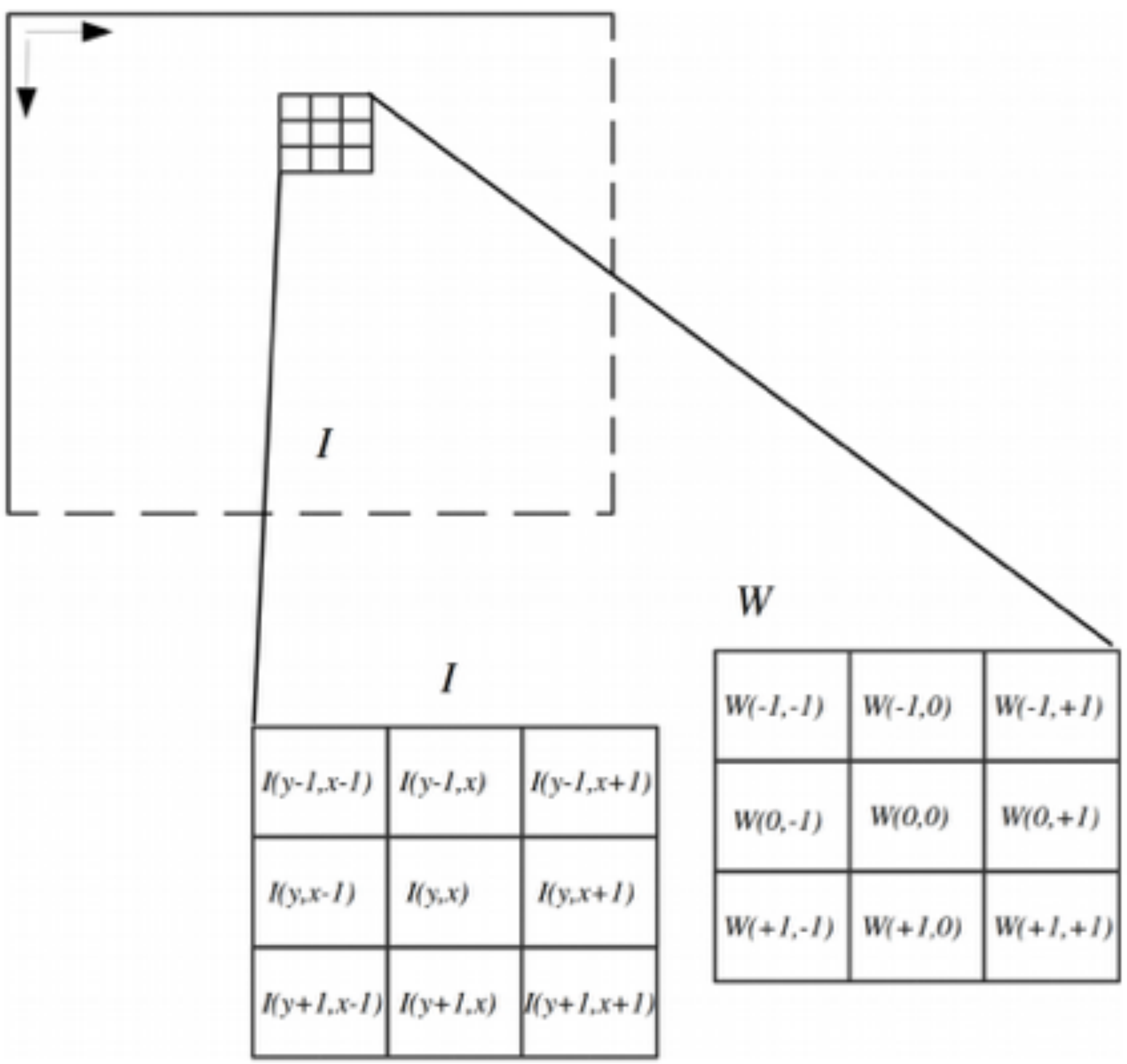
Pitfall:
Blacker box

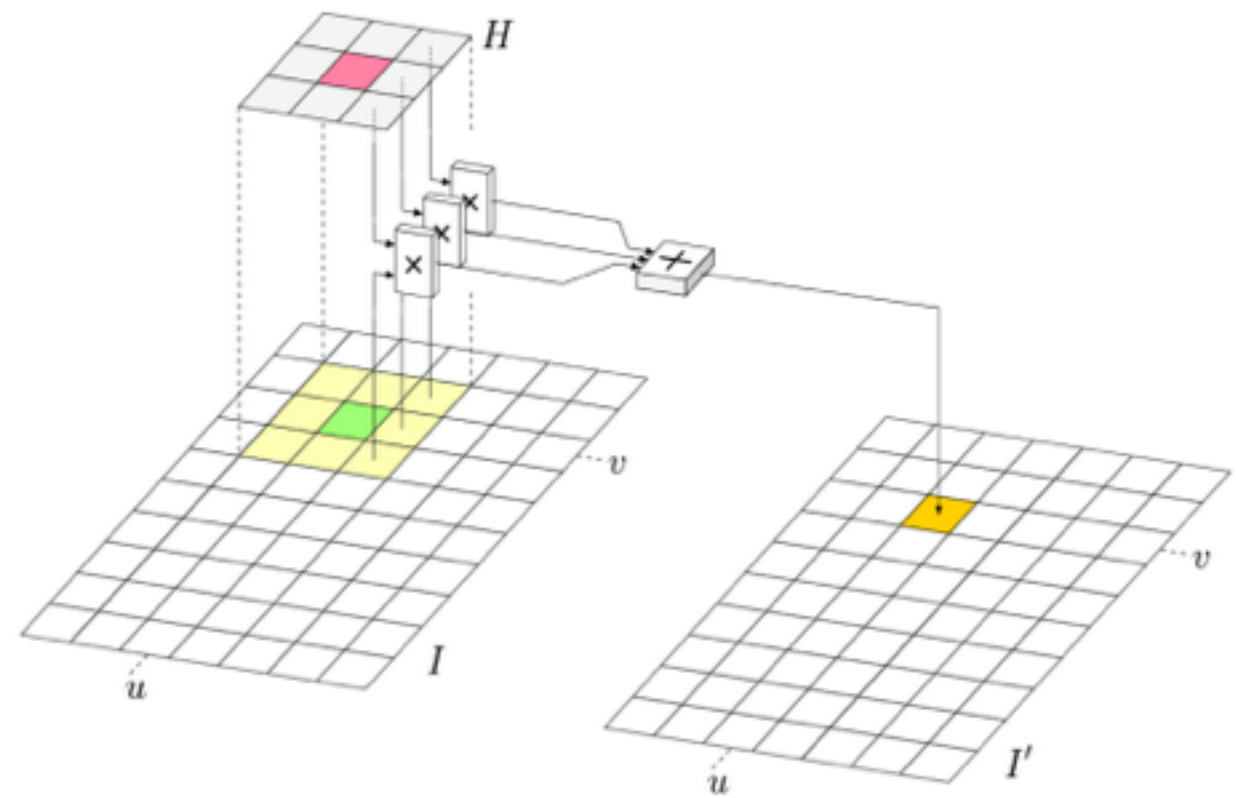


Deep Learning Flow

Adil Moujahid

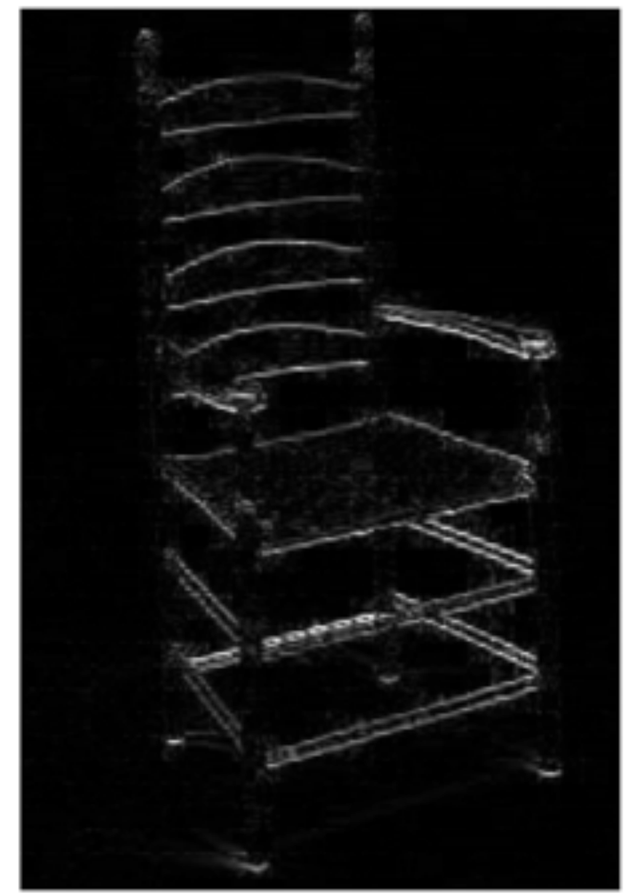
Convolution





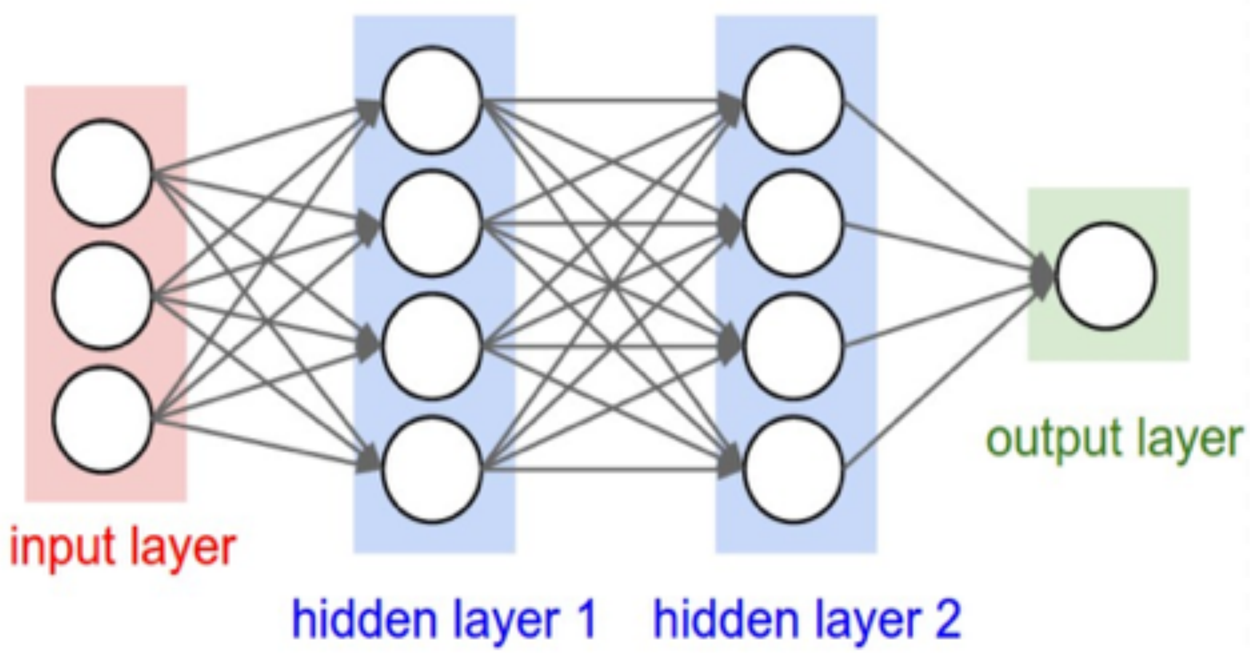
Convolution filter

-1
0
1

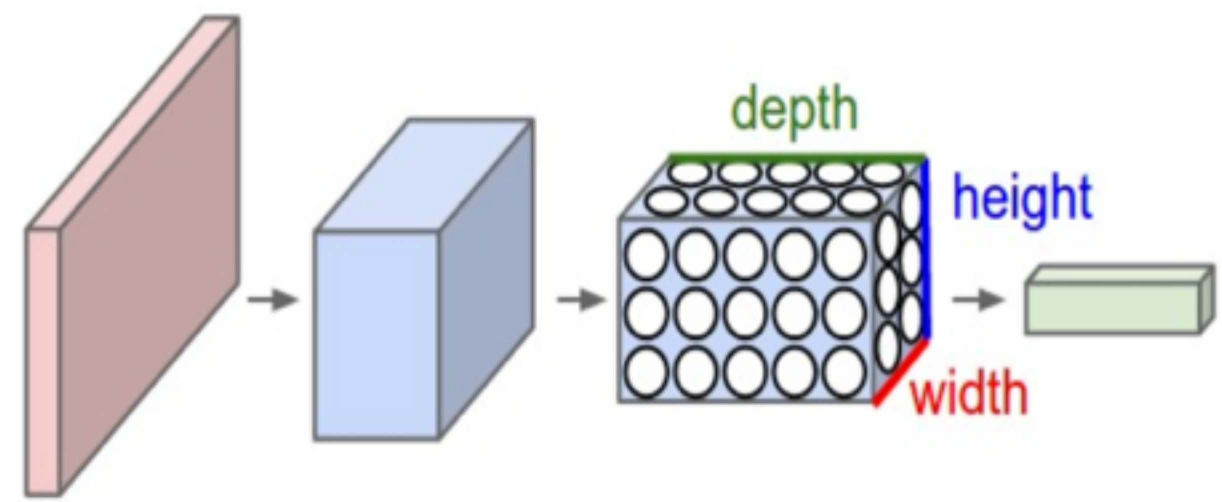


CONVOLUTION AS FEATURE EXTRACTOR

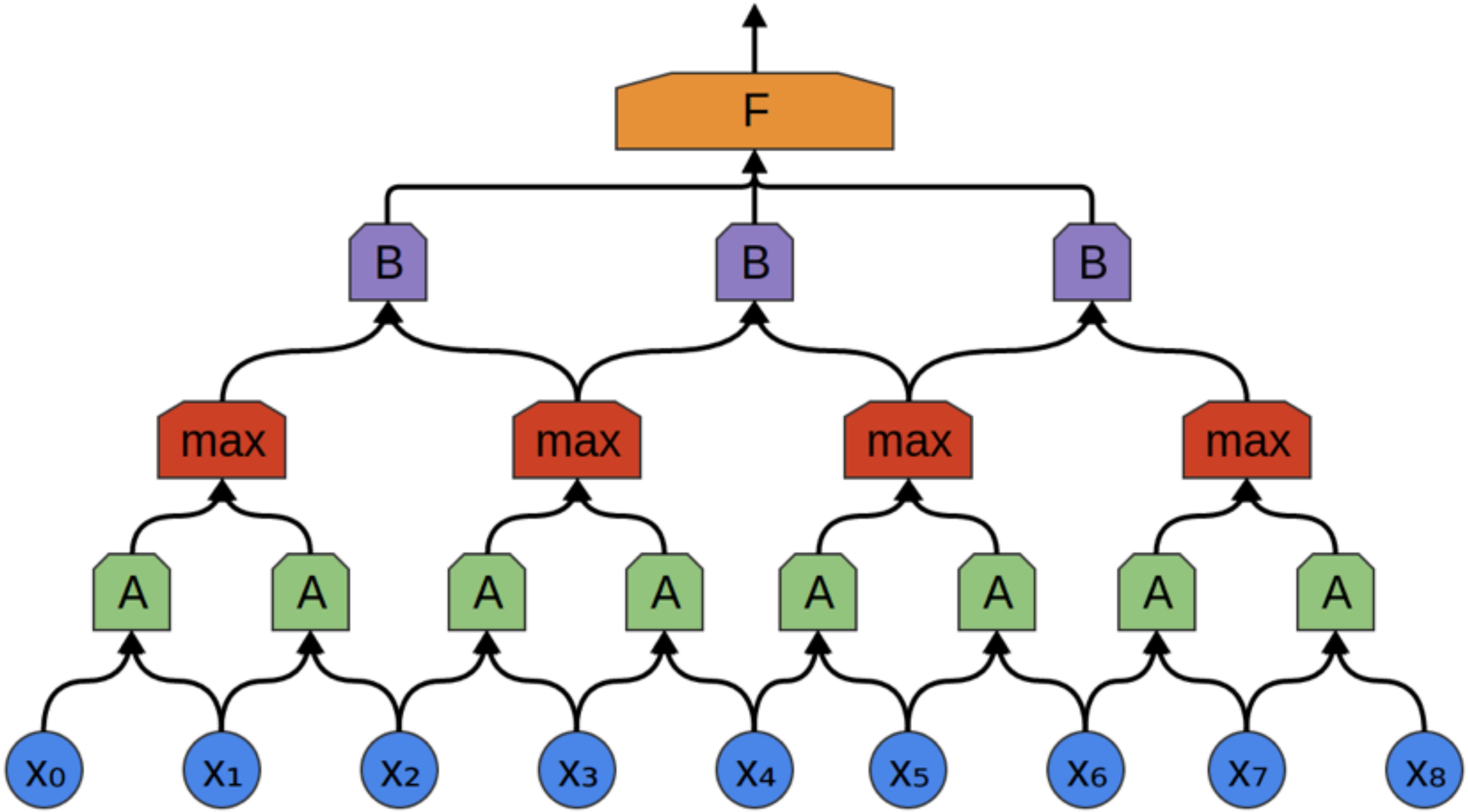
MLP & CNN



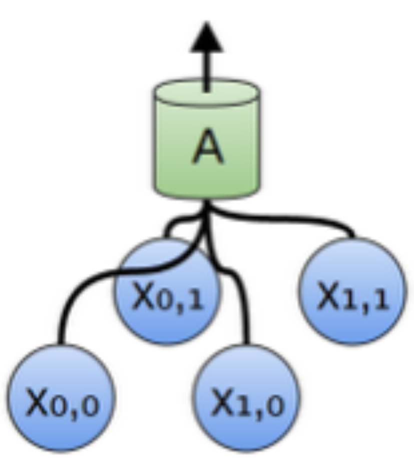
Multi-Layer Perceptron



Convolutional Neural Net (CNN)

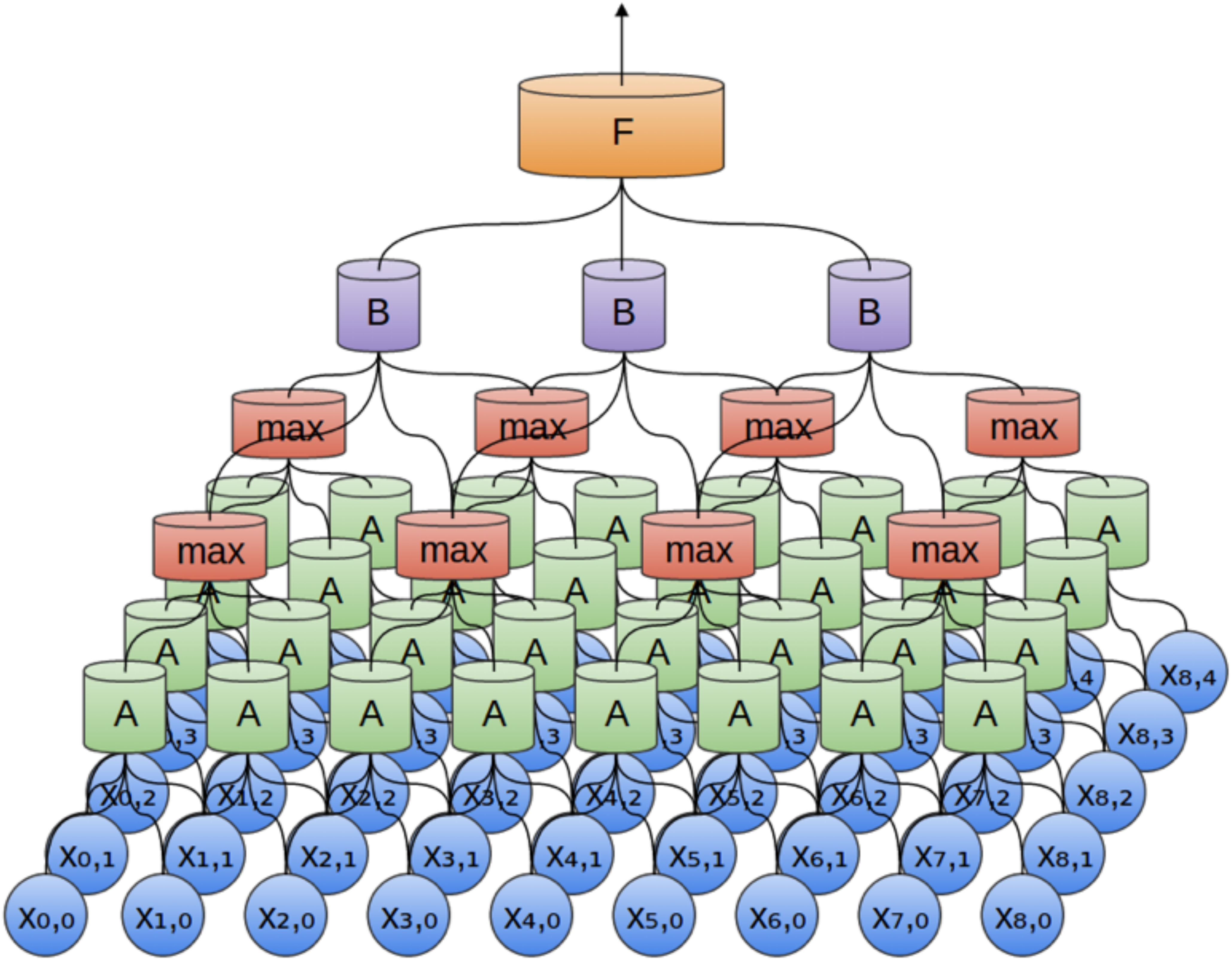


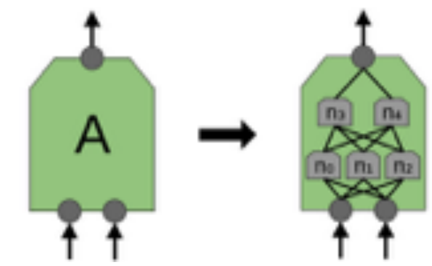
conv + pool + conv + connected



http://colah.github.io/posts/2014-07-Conv-Nets-Modular/

2D version of convolution

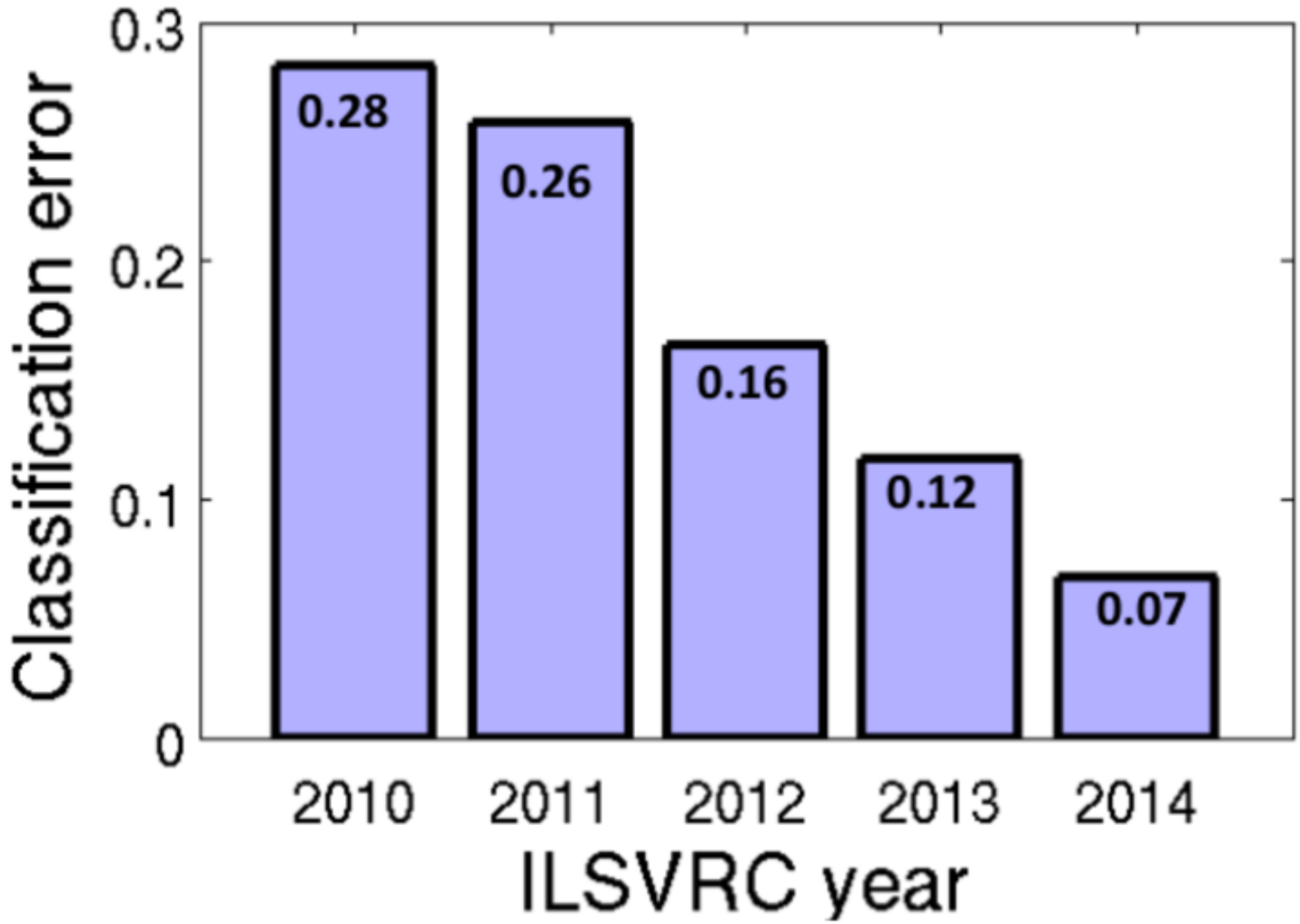




http://colah.github.io/posts/2014-07-Conv-Nets-Modular/

Final layers are fully connected

Evolution of CNNs



ImageNet Competition

2015 ILSVRC leaderboard

Team name	Entry description	**Number of object categories won**	mean AP
MSRA	An ensemble for detection.	194	0.620741
Qualcomm Research	NeoNet ensemble with bounding box regression. Validation mAP is 54.6	4	0.535745
CUImage	Combined multiple models with the region proposals of cascaded RPN, 57.3% mAP on Val2.	2	0.527113
The University of Adelaide	9 models	0	0.514434
MCG-ICT-	2 models on 2 proposals without category information: {[SS+EB]+	[illegible]	[illegible]

Classification error:
0.03567

Yellow: Winner in category
Yellow/White: Reveal code
Gray: Won't reveal code

2016 ILSVRC leaderboard

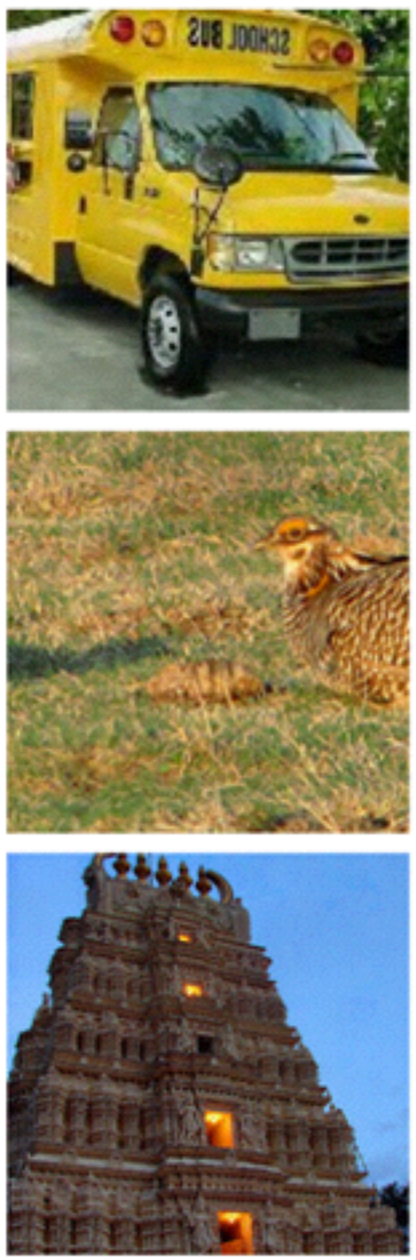
Team name	Entry description	Number of object categories won	mean AP
CUImage	Ensemble of 6 models using provided data	109	0.662751
Hikvision	Ensemble A of 3 RPN and 6 FRCN models, mAP is 67 on val2	30	0.652704
Hikvision	Ensemble B of 3 RPN and 5 FRCN models, mean AP is 66.9, median AP is 69.3 on val2	18	0.652003
NUIST	submission_1	15	0.608752
NUIST	submission_2	9	0.607124
Trimps-Soushen	Ensemble 2	8	0.61816
360+MCG-ICT-CAS_DET	9 models ensemble with validation and 2 iterations	4	0.615561
360+MCG-ICT-CAS_DET	Baseline: Faster R-CNN with Res200	4	0.590596
Hikvision	Best single model, mAP is 65.1 on val2	2	0.634003
CIL	Ensemble of 2 Models	1	0.553542
360+MCG-ICT-CAS_DET	9 models ensemble	0	0.613045

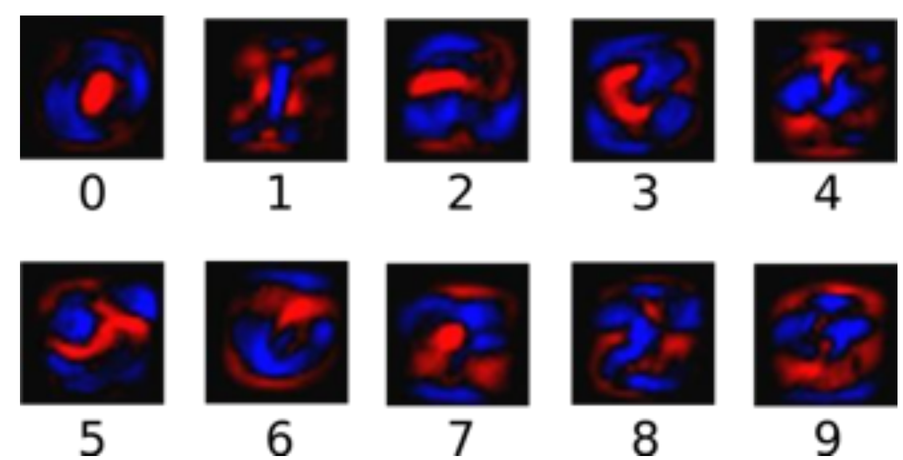
Classification error:
0.02991

What bird is that?

or: what features is my deep network using?

Including adversarial examples during training

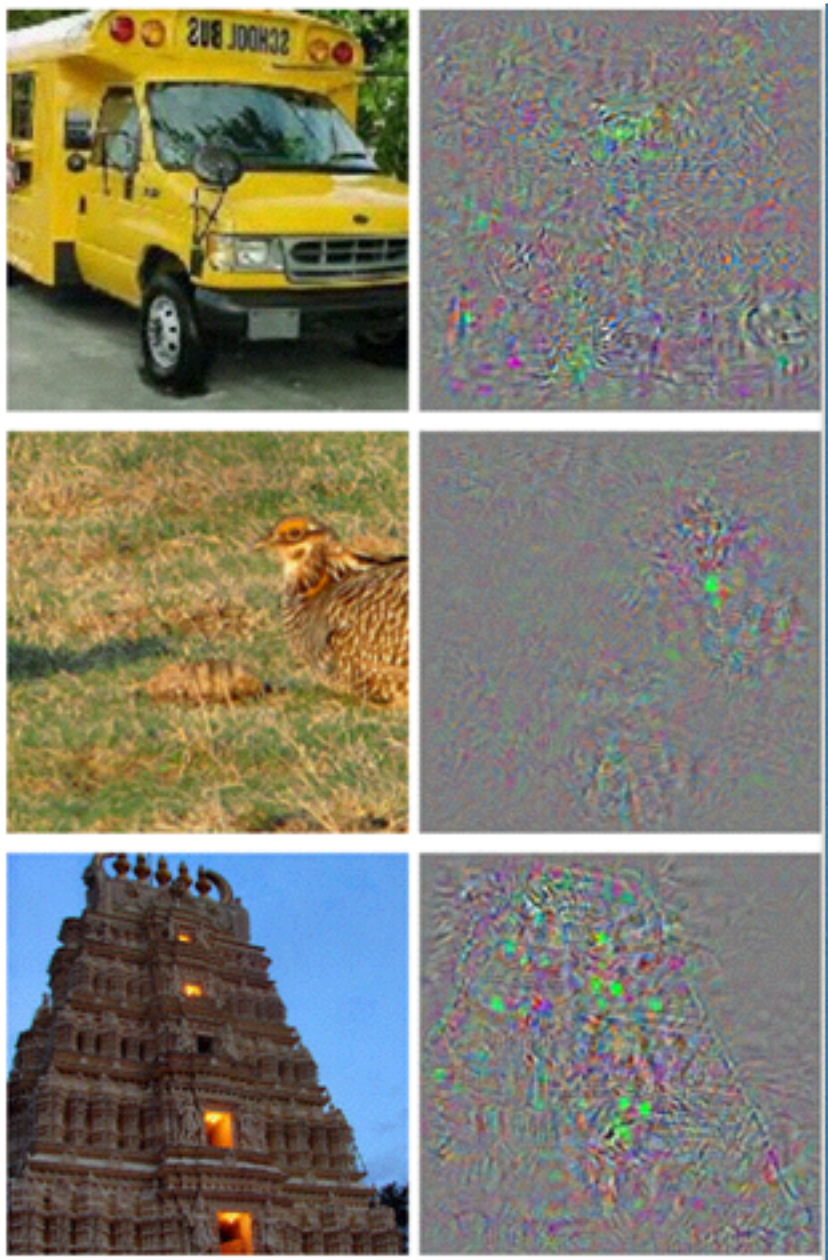


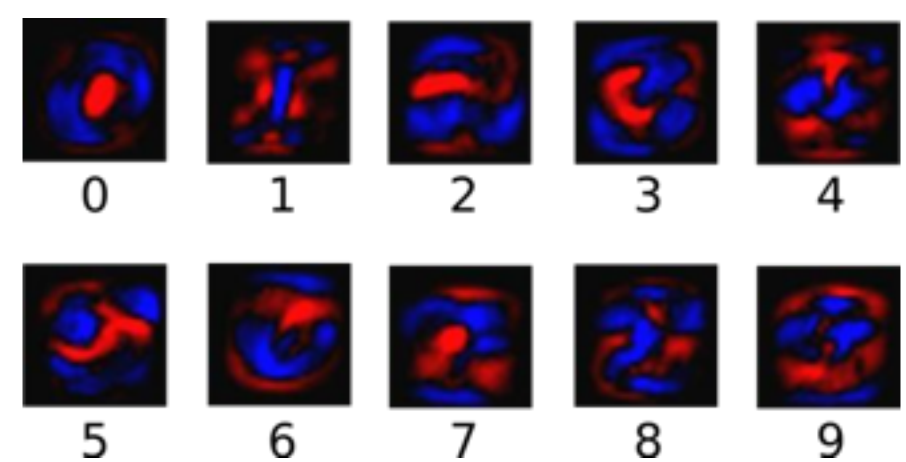


MNIST digits significance map

https://arxiv.org/pdf/1312.6199v4.pdf

Including adversarial examples during training

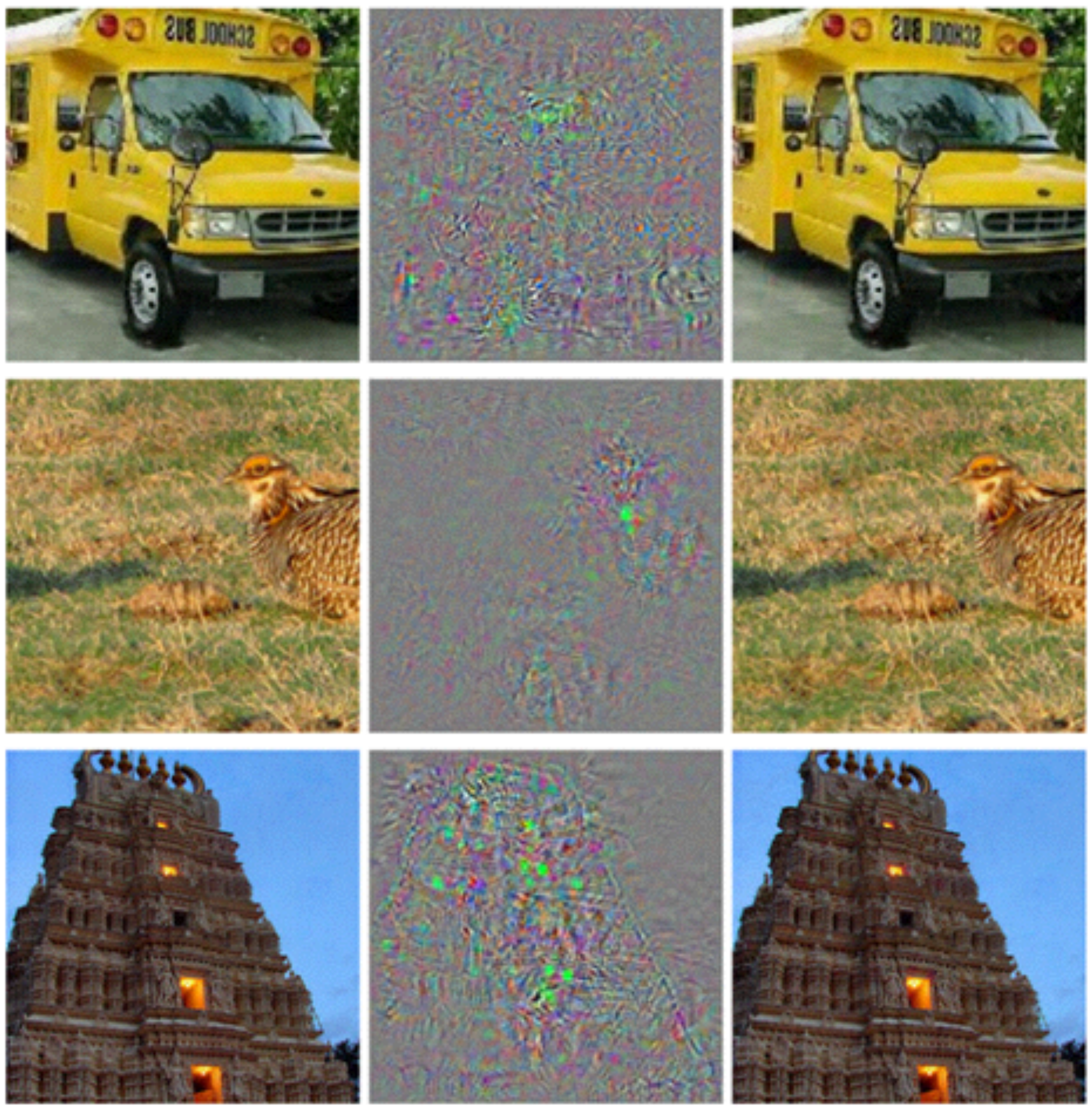




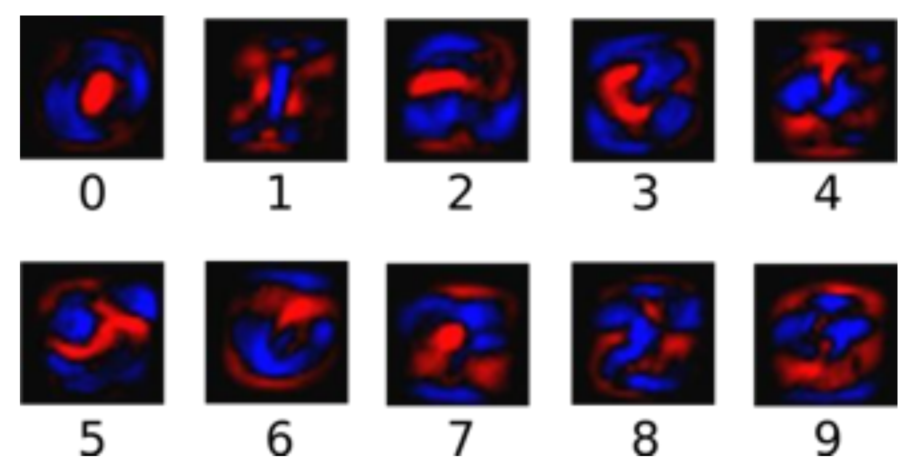
MNIST digits significance map

https://arxiv.org/pdf/1312.6199v4.pdf

Including adversarial examples during training



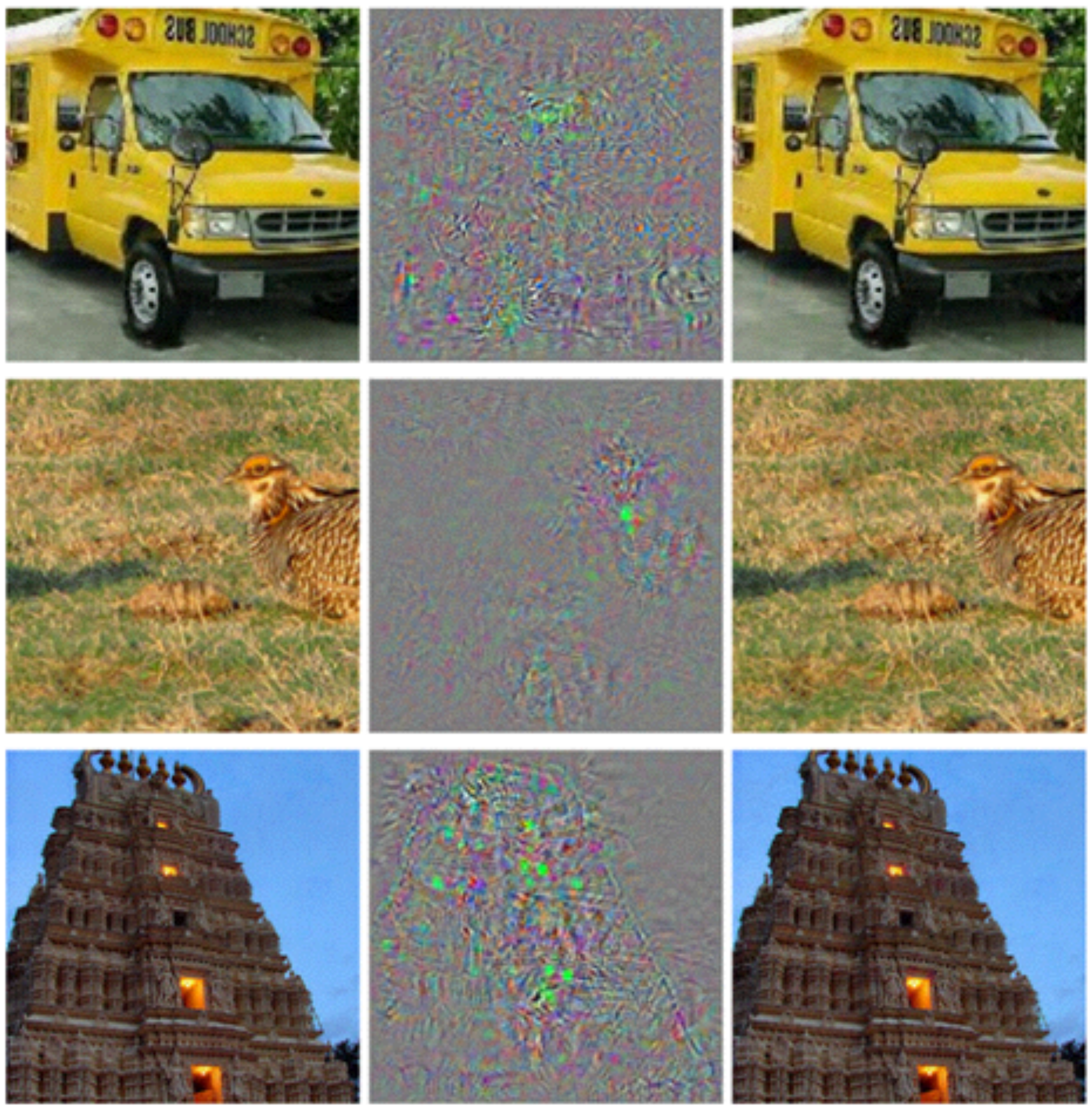
The images in the left most column are correctly classified examples. The middle column represents the distortion between the left and right images. The images in the right most column are predicted to be of the class ostrich! Even though the difference between the images on the left and right is imperceptible to humans, the ConvNet makes drastic errors in classification.



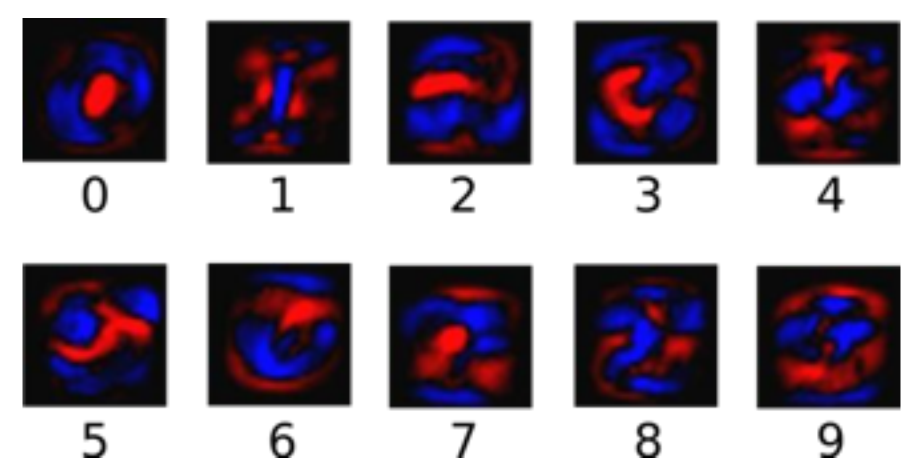
MNIST digits significance map

https://arxiv.org/pdf/1312.6199v4.pdf

Including adversarial examples during training



The images in the left most column are correctly classified examples. The middle column represents the distortion between the left and right images. The images in the right most column are predicted to be of the class ostrich! Even though the difference between the images on the left and right is imperceptible to humans, the ConvNet makes drastic errors in classification.

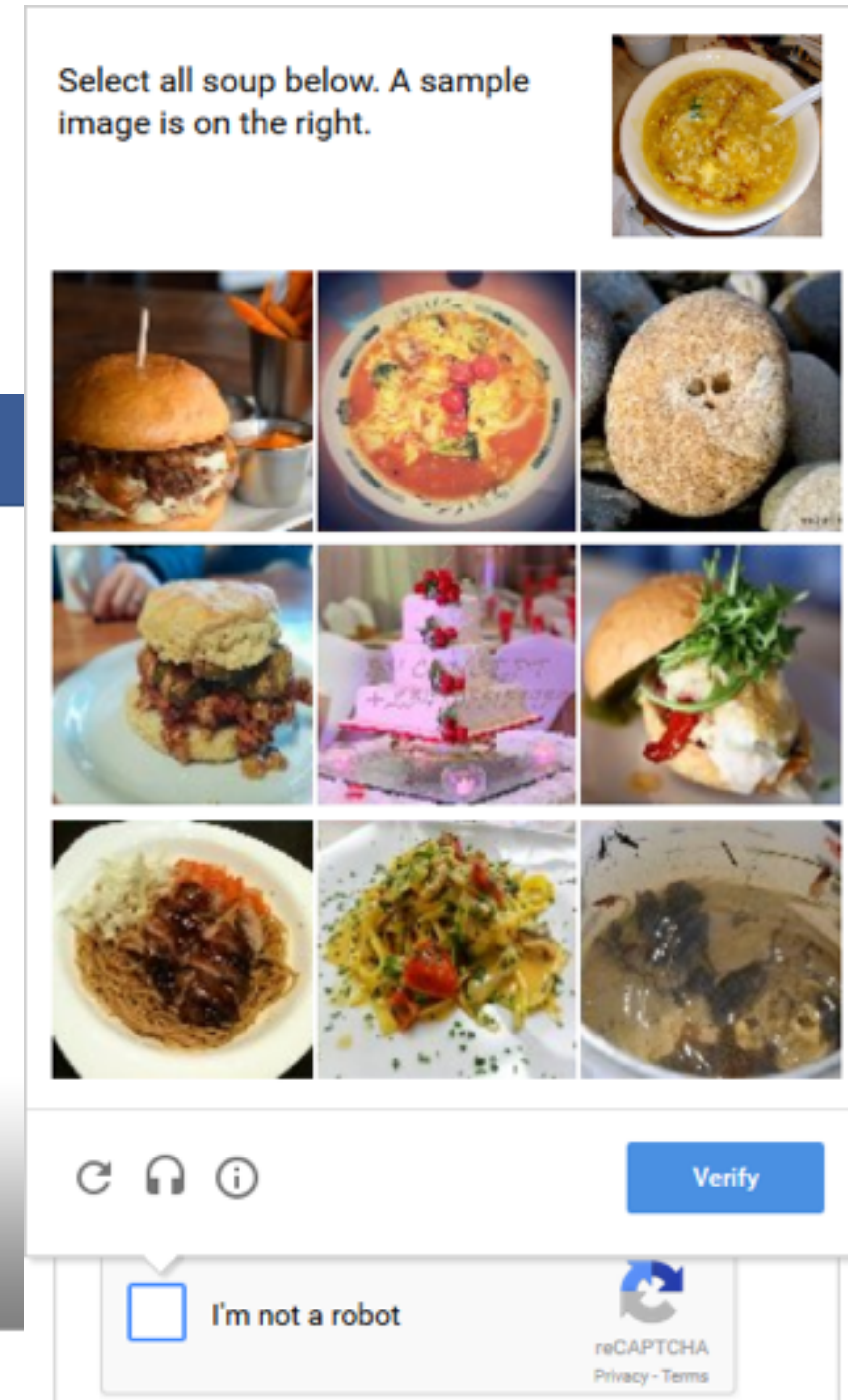


MNIST digits significance map

https://arxiv.org/pdf/1312.6199v4.pdf

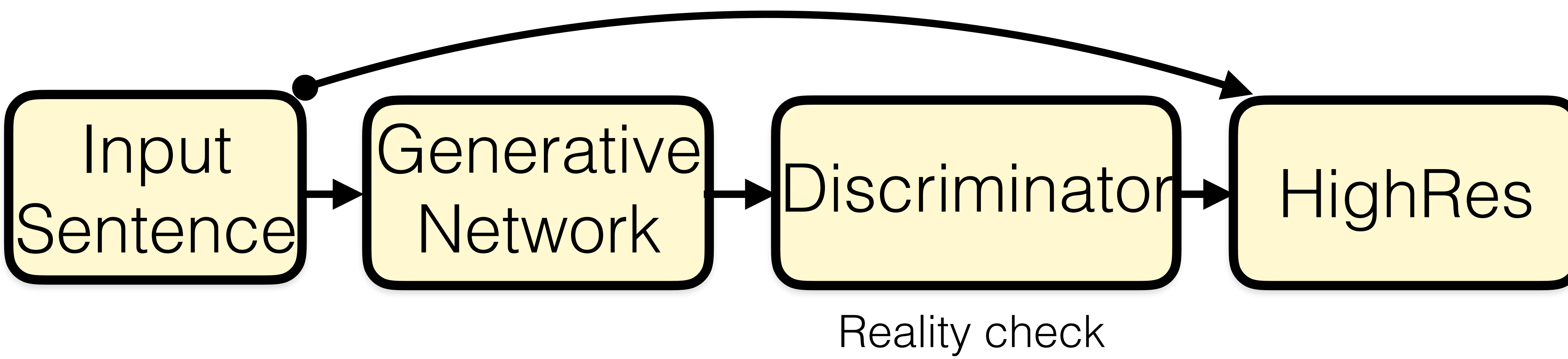
Pitfall: Overlearning

Labels are everywhere

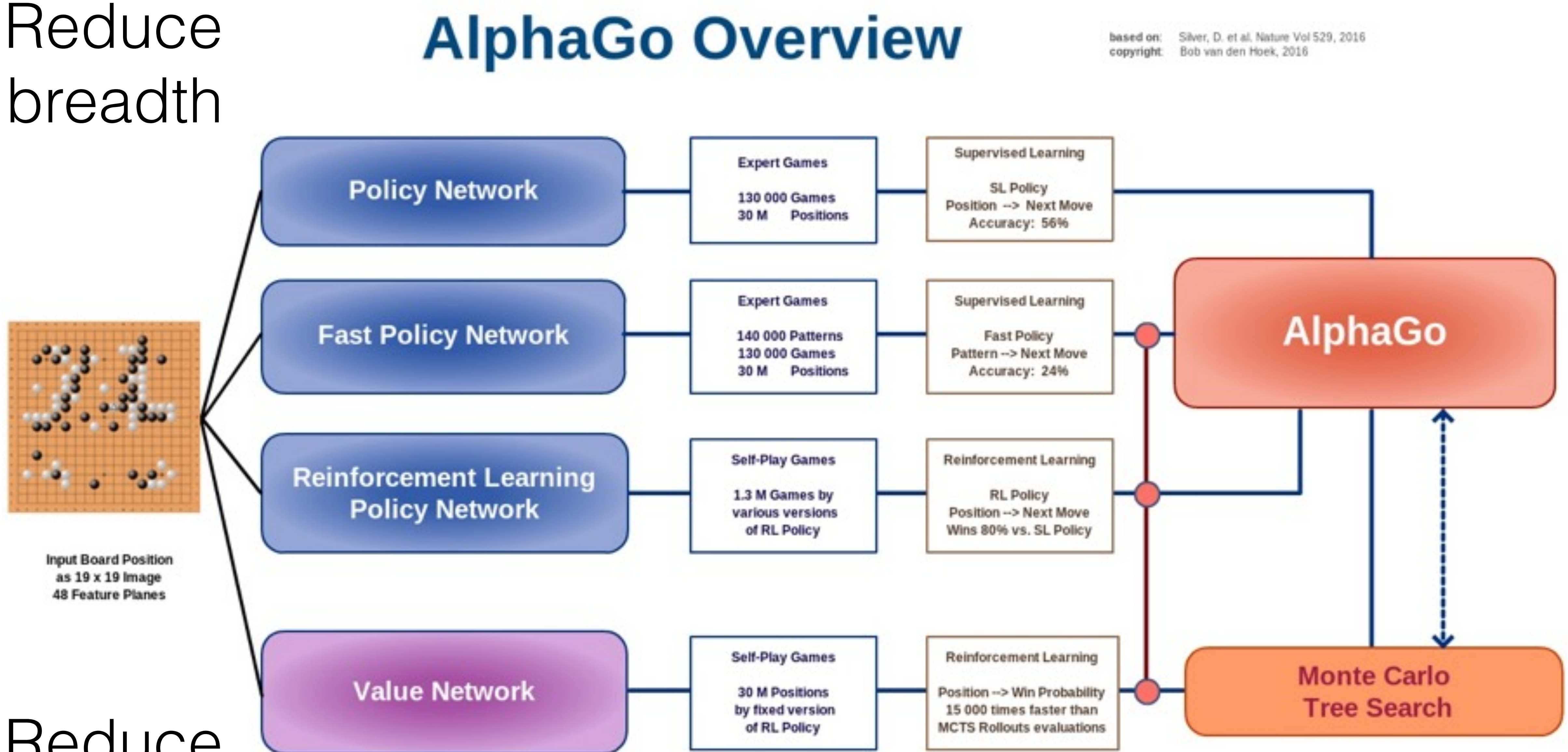


Generative Adversarial Networks

Zhang et al. 2016

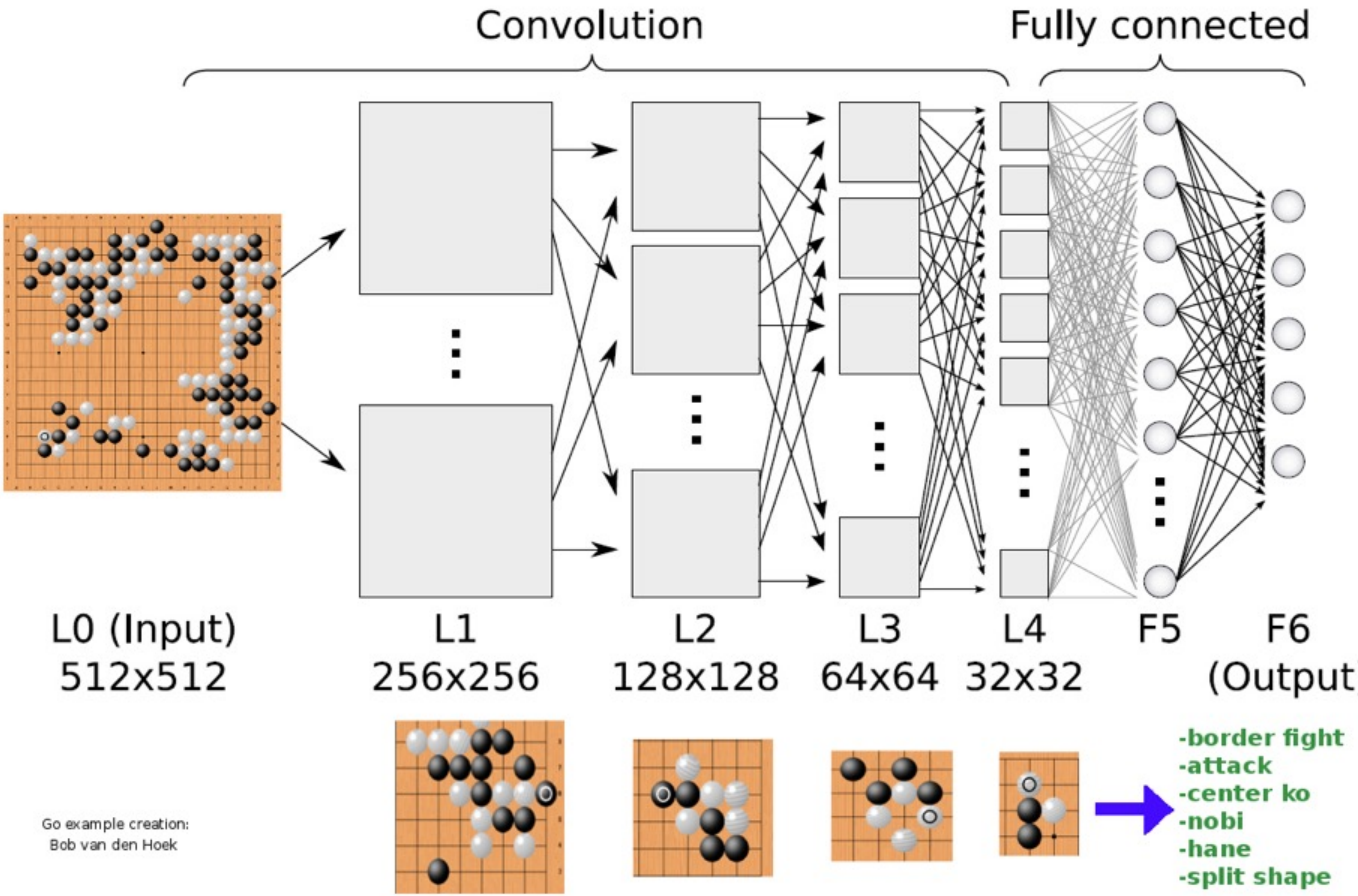


Reduce breadth



Reduce depth

http://deeplearningskysthelimit.blogspot.com/2016/04/part-2-alphago-under-magnifying-glass.html



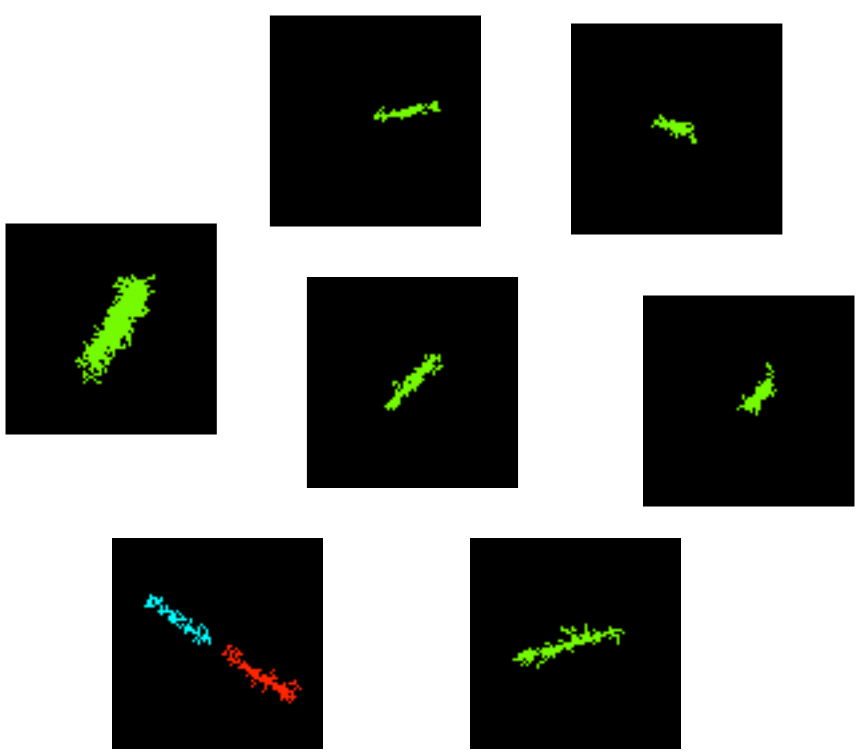
http://gobase.org/online/intergo/?query=%22hane%20nobi%22

http://deeplearningskysthelimit.blogspot.com/2016/04/part-2-alphago-under-magnifying-glass.html

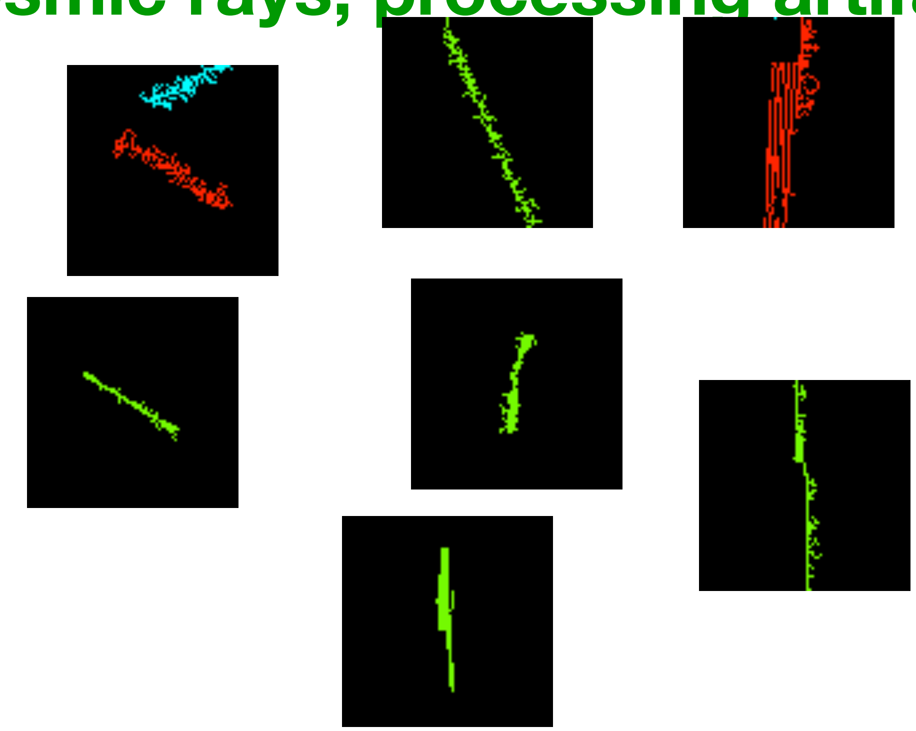
Asteroid Detection from Sky Survey Imagery

- **Goal:** automatically distinguish real vs. bogus asteroids from Palomar Transient Factory (PTF) imagery
- **Current dataset:** 240 confirmed asteroids, 1441 synthetically-generated asteroids, 20072 bogus

Confirmed Asteroids



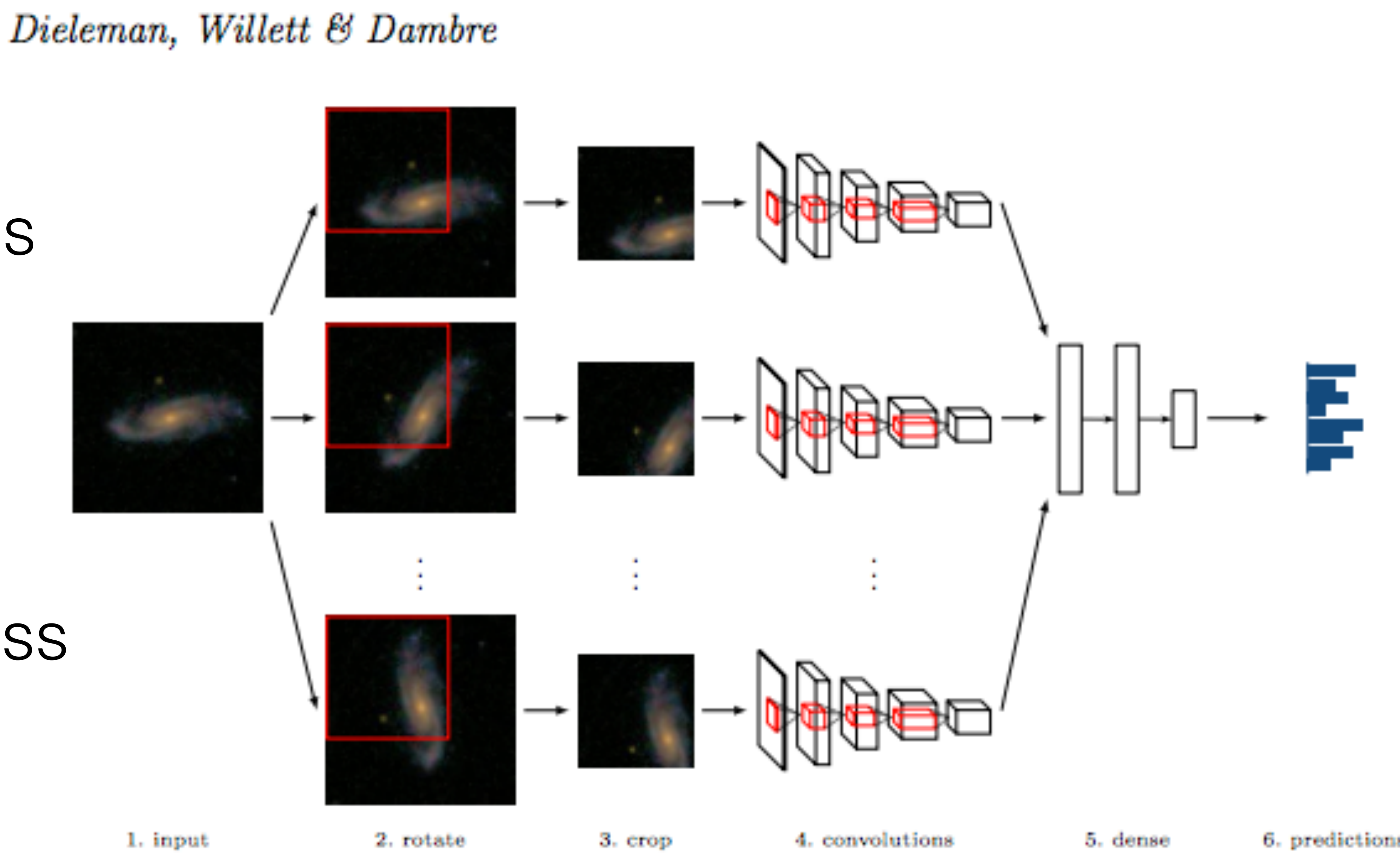
Bogus Detections
(cosmic rays, processing artifacts)



B Bue

Applications

smoothness
edge-on
bar
spiral
bulge
roundedness

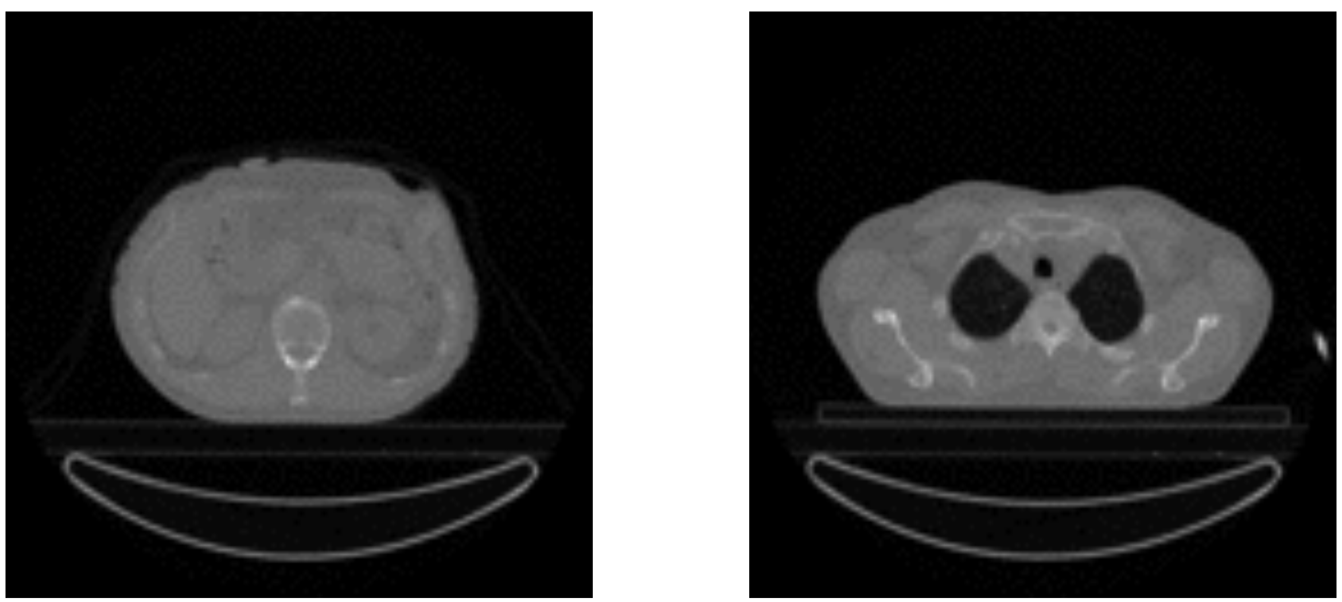


2015

Galaxy images (Huertas-Company et al., 2015)

Plans with Cancer datasets

Lung dataset:
https://wiki.cancerimagingarchive.net/display/Public/NSCLC-Radiomics
DataType: **non-small cell lung cancer (NSCLC)**
modalities: CT, RSTRUCT
number of patients: **422**
number of images: **51K**
pixel dimensions: **512x512**



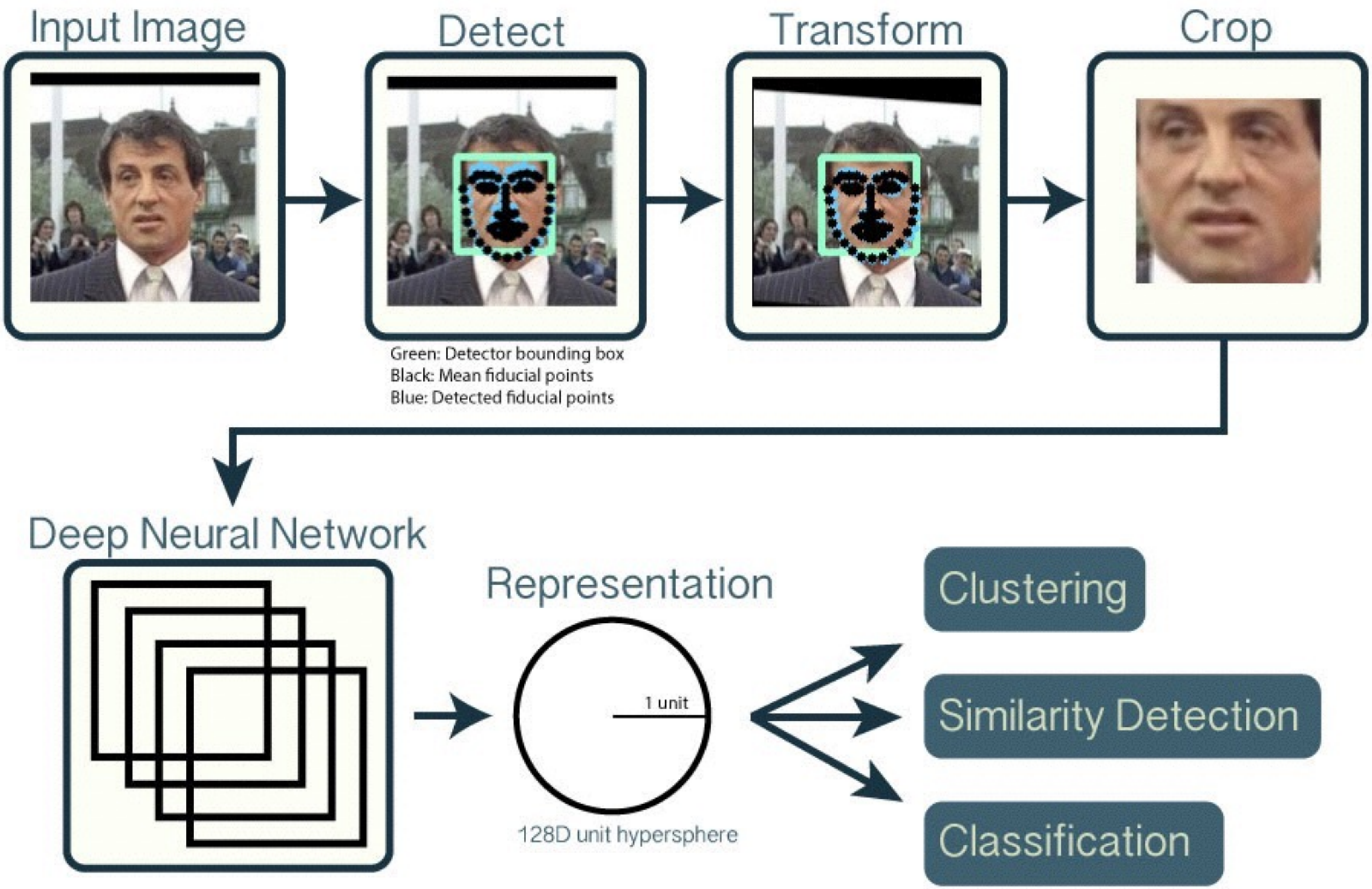
With and without cavity

	A	B	C	D	E	F	G	H	I	J	K
1	PatientID	age	clinical.T.Stage	Clinical.N.St	Clinical.M.Stage	Overall.Stage	Histology	gender	Survival.ti	deadstatus.event	
2	LUNG1-00	78.7515	2	3	0	IIIb	large cell	male	2165	1	
3	LUNG1-00	83.8001	2	0	0	I	squamous	male	155	1	
4	LUNG1-00	68.1807	2	3	0	IIIb	large cell	male	256	1	
5	LUNG1-00	70.8802	2	1	0	II	squamous	male	141	1	
6	LUNG1-00	80.4819	4	2	0	IIIb	squamous	male	353	1	
7	LUNG1-00	73.8864	3	1	0	IIIa	squamous	male	173	1	
8	LUNG1-00	81.5288	2	2	0	IIIa	squamous	male	137	1	
9	LUNG1-00	71.666	2	2	0	IIIa	adenocarc	male	77	1	
10	LUNG1-00	56.1342	2	2	0	IIIa	squamous	male	131	1	
11	LUNG1-01	71.0554	4	3	0	IIIb	squamous	female	2119	0	
12	LUNG1-01	64.3313	4	0	0	IIIb	squamous	male	515	1	
13	LUNG1-01	71.2553	3	2	0	IIIa	squamous	male	85	1	

David Liu

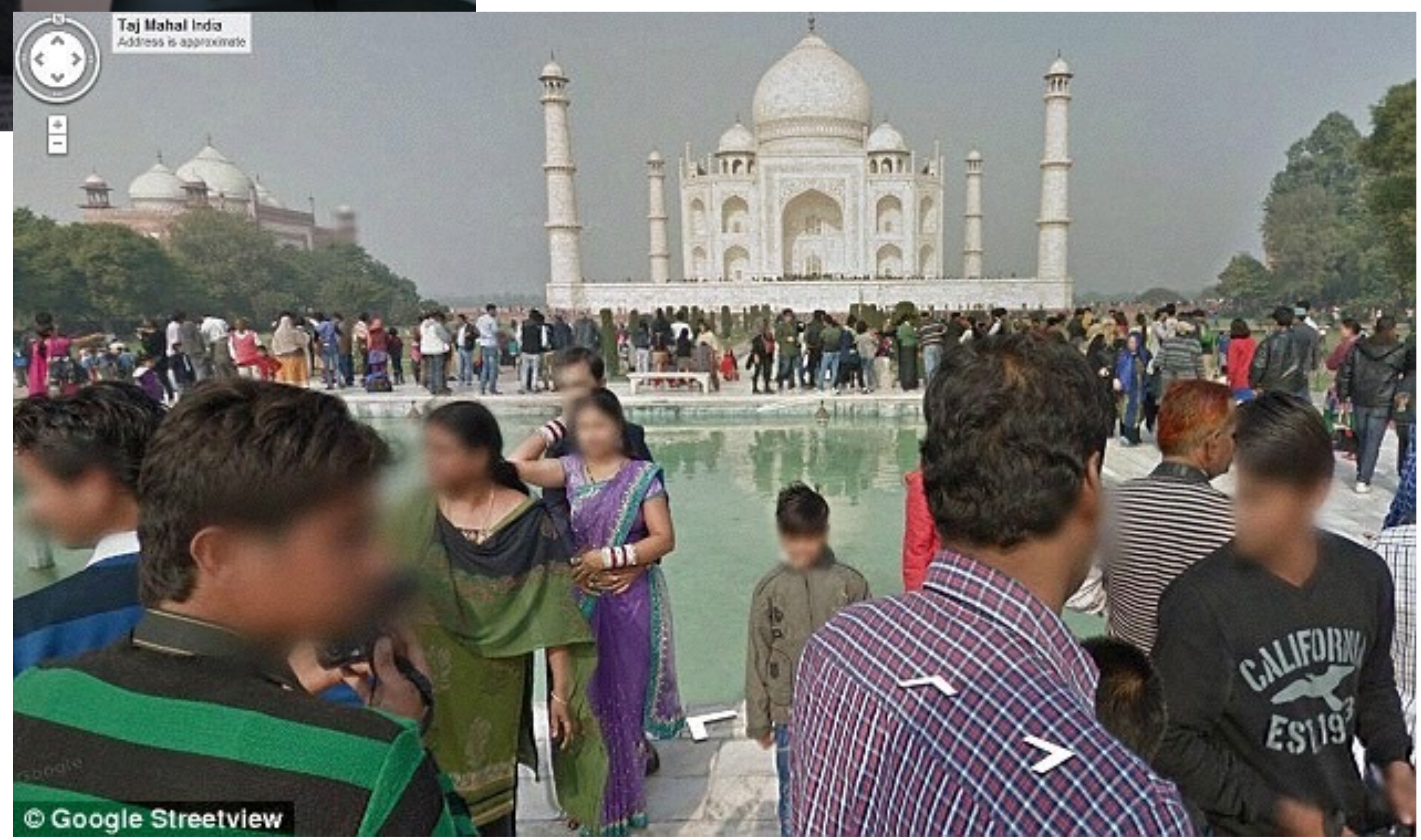
Will use NLST

OpenFace



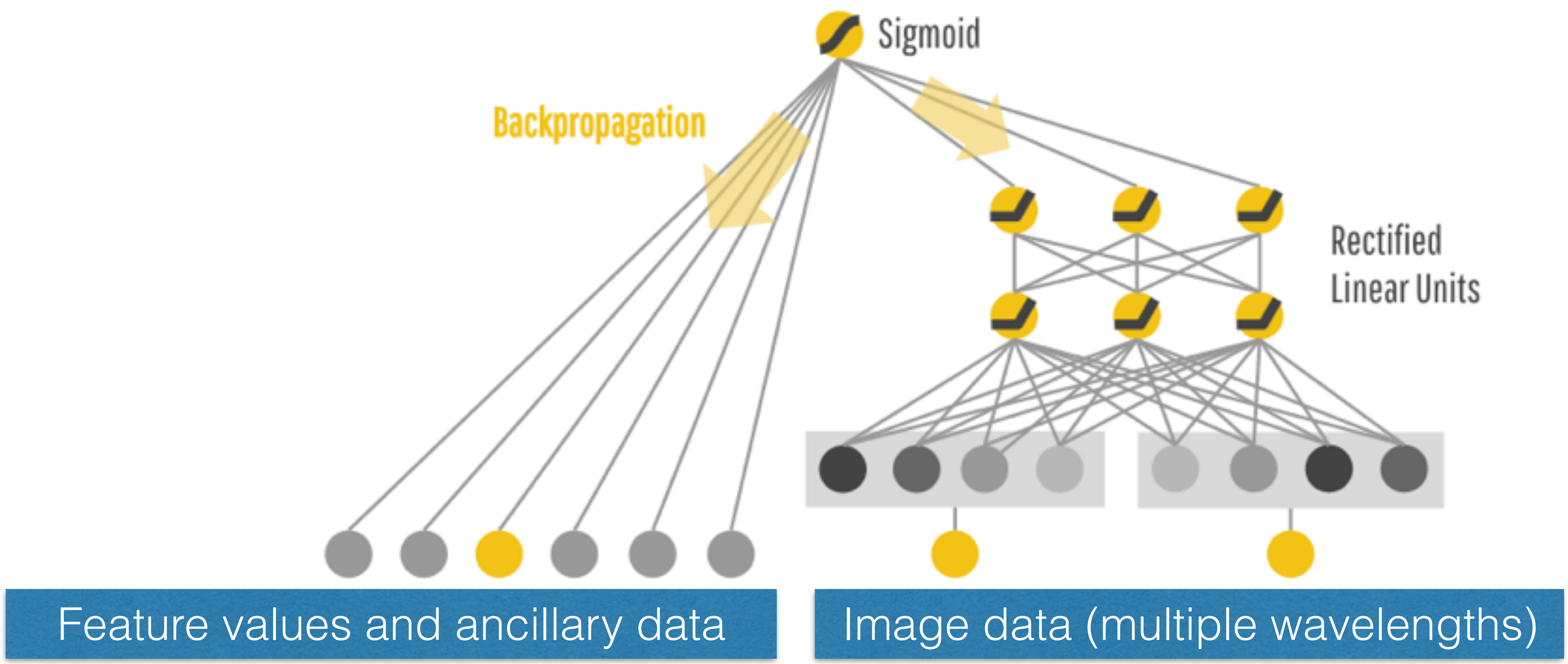
https://cmusatyalab.github.io/openface/

Another use for the face API



Faces masked in Streetview

Combining with unstructured data



The “comments” or metadata become additional features (GoogLeNet)

https://research.googleblog.com/2016/06/wide-deep-learning-better-together-with.html

In <your area of interest> what can you apply deep learning to?

- One speculative example
- One more directly related to your work

Demo ConvNet

Sample of cats & dogs images from Kaggle Dataset

Useful links:

https://prateekvjoshi.com/2016/01/05/how-to-install-caffe-on-ubuntu/

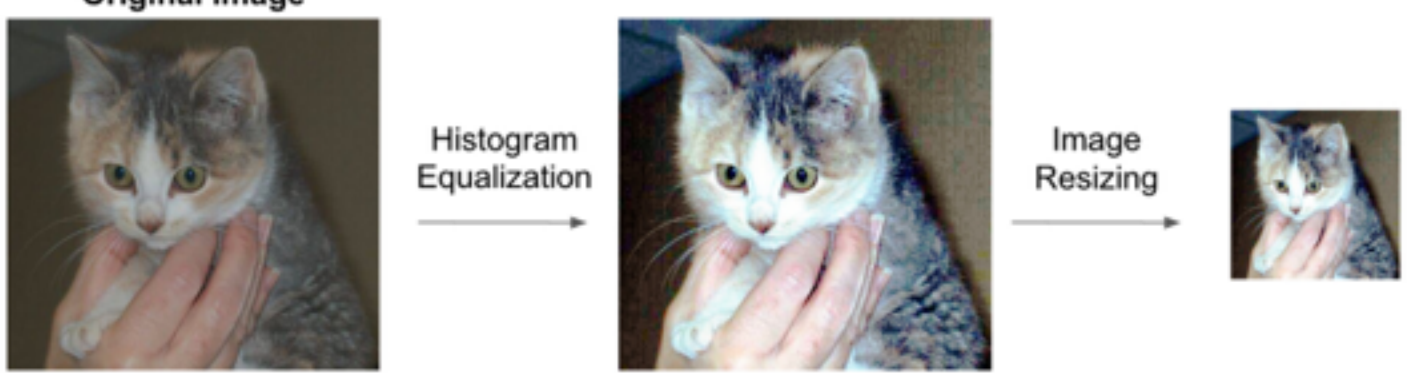
https://prateekvjoshi.com/2016/02/02/deep-learning-with-caffe-in-python-part-i-defining-a-layer/

http://adilmoujahid.com/posts/2016/06/introduction-deep-learning-python-caffe/

Demo ConvNet (kaggle)

25.000 examples

- Designing layers [demo]
- Pre-processing :

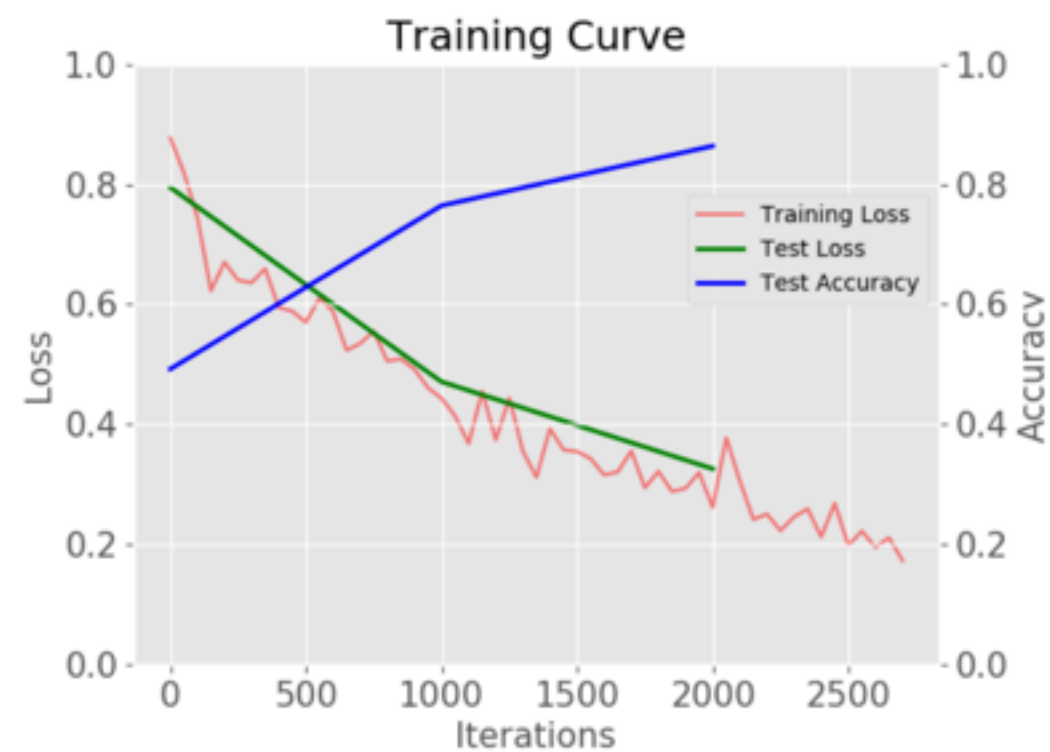


Example of image transformations applied to one training image

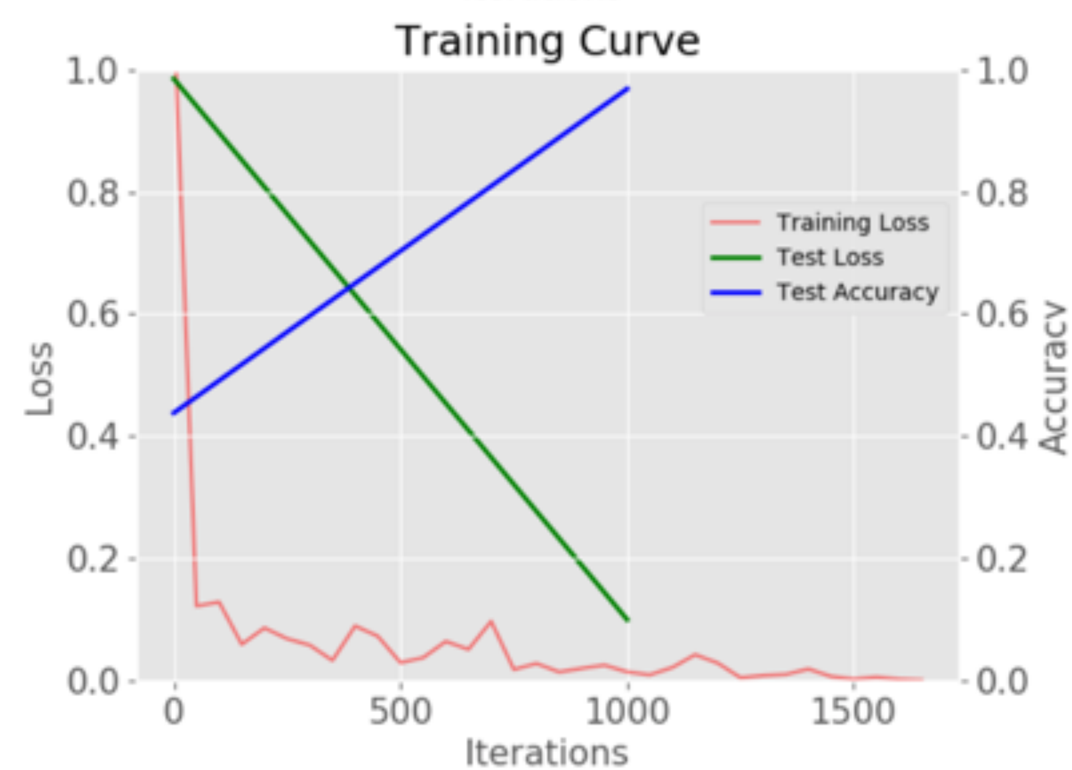
- Convolutional network architecture (AlexNet):

Demo ConvNet (kaggle)

- Training and test results (traditional, 2 days):



- Training and test results (transfer learning, 2 days):



Model zoo:
https://github.com/BVLC/caffe/wiki/Model-Zoo

Demo ConvNet

Online deep learning! [demo]

http://demo.caffe.berkeleyvision.org/

Summary

- CNNs are taking over, especially the image domain
- Can come up with features not thought of before
- Abstracted libraries and visualizations available
- Over-learning can be a problem:
  - augmentation
  - adversarial examples/generative networks
- Should ensure they do not become convoluted
- Deep and wide networks may prove to be a boon