Introduction to Deep Learning

Lecture 1
Introduction

Maria Vakalopoulou & Stergios Christodoulidis

MICS Laboratory

CentraleSupélec
Université Paris-Saclay

Friday, November 10, 2023

About me

- Undergrad at National Technical University of Athens, Greece
- Ph.D in CS and CV at National Technical University of Athens, Greece
- Postdoc researcher at CentraleSupelec
- Assistant Professor at CentraleSupelec (since January 2019)

Research interests: ML, DL, Computer Vision, Medical Imaging and Remote Sensing

The Team

Maria Vakalopoulou

Office: Bâtiment Bouygues,
MICS Lab, Room SB.132
Email:maria.vakalopoulou@centralesupelec.fr

Stergios Christodoulidis

Office: Bâtiment Bouygues,
MICS Lab, Room SB.132
Email:stergios.christodoulidis@centralesupelec.fr

Teaching Assistant

Leo Fillioux
Office: Bâtiment Bouygues, MICS Lab
Email: leo.fillioux@centralesupelec.com

Loic Le Bescond
Office: Bâtiment Bouygues, CVN Lab
Email:loic.le-bescond@centralesupelec.fr

Office hours: we will be available right after the lecture
Or, send us an email and we will find a good time to meet

Prerequisites

- Machine Learning
- Calculus, Linear Algebra
- Statistics
- Python programming
- Time & patience!

Learning Goals

The course aims to introduce students to the design of deep learning methodologies both in theory and practice. We expect that by the end of the course, the students will:

- Have knowledge of state-of-the-art deep learning techniques
- Have a deeper understanding of deep learning methods
- Have practical experience with deep learning frameworks

Outline of the course

- Wednesdays and Fridays
- 1h30 Lecture and 1h30 Practical Session *[in PyTorch and Colab]*
- Interactions via edunao.
  - Foundation of Deep Learning – DSBA (2023-2024)

Content:

- General Deep Learning;
- Image & Text Understanding;
- Different architectures and learning schemes

Outline of the course

- The evaluation of the course will be based on:
  - One assignment: the assignments will include theoretical questions as well as hands-on practical questions. The assignment will be individual! [50%]
  - Kaggle Competition: The students are expected to form groups of 2 people and participate in a kaggle competition. The final grade will be assigned by the performance and the methodology, described by a report and a well documented code [50%]

Schedule of the course

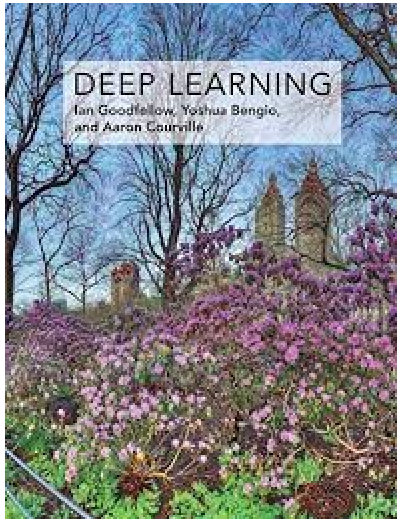
- Introduction. *[Maria Vakalopoulou]*
- Backpropagation and Modularity. 1H30 Theory, 1H30 Lab *[Stergios Christodoulidis]*
- Optimization. 1H30 Theory, 1H30 Lab *[Maria Vakalopoulou]*
- Convolutional Neural Networks. 1H30 Theory, 1H30 Lab *[Maria Vakalopoulou]*
- Modern CNNs. 1H30 Theory, 1H30 Lab *[Maria Vakalopoulou]*
- TBA. 1H30 Theory, 1H30 Lab
- Generative Adversarial Networks. 1H30 Theory, 1H30 Lab *[Maria Vakalopoulou]*
- RNNs & LSTMs 1H30 Theory, 1H30 Lab *[Stergios Christodoulidis]*

Overview

- Bibliography
  - Deep Learning by I. Goodfellow, Y. Bengio, A. Courville (available online)
  - Neural Networks and Deep Learning by M. Nielsen (available online)
- ++ A lot of papers...
  - CVPR, ICCV, ECCV, ACCV, BMVC, NIPS, ICLR, ICML, AAAI, …
  - PAMI, Journal of Machine Learning Research, Artificial Intelligence, …

A continuously involving field, every topic covered in the class, you can find material in textbooks or even in the web.

Very practical course. Apply what you've learn in theory.



Acknowledgments

- The lecture is partially based on material by:
  - Efstratios Gavves (University of Amsterdam)
  - Francois Fleuret (EPFL)
  - Vincent Lepetit (University of Bordeaux)
  - Chloé-Agathe Azencott (Mines ParisTech)

Thank you!

Roadmap

- Motivation and Applications of Deep Learning
- A brief history of Neural Networks and Deep Learning
- Deep Neural Networks and Learning Schemes

Motivation

Computer vision

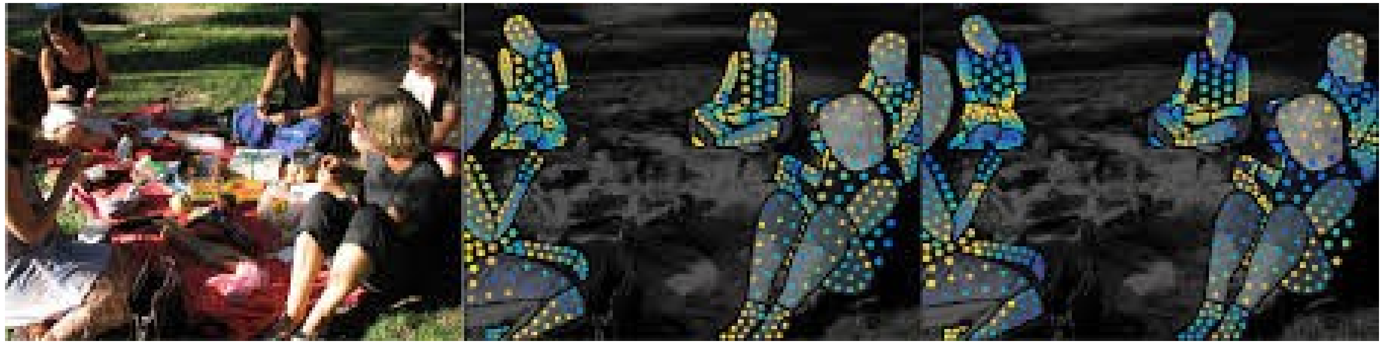
- Object Detection and [segmentation]

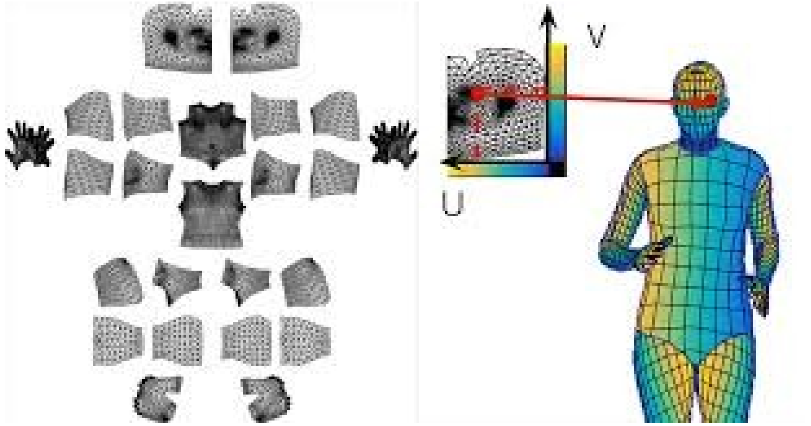
Pinheiro et al., 2016

Motivation

Computer vision

- Pose Estimation





Riza Alp Guler et al., 2018

Motivation

Computer vision

- Auto-captioning

Wei et al., 2016

Motivation

Computer vision

- Image generation

Brock et al., 2018

Motivation

Computer vision

- [Style transfer]

Gatys et al., 2017

Motivation

Natural Language Processing

- Translation

“The reason Boeing are doing this is to cram more seats in to make their plane more competitive with our products,” said Kevin Keniston, head of passenger comfort at Europe’s Airbus.

→ “La raison pour laquelle Boeing fait cela est de créer plus de sièges pour rendre son avion plus compétitif avec nos produits”, a déclaré Kevin Keniston, chef du confort des passagers chez Airbus.

When asked about this, an official of the American administration replied: “The United States is not conducting electronic surveillance aimed at offices of the World Bank and IMF in Washington.”

→ Interrogé à ce sujet, un fonctionnaire de l’administration américaine a répondu: “Les États-Unis n’effectuent pas de surveillance électronique à l’intention des bureaux de la Banque mondiale et du FMI à Washington”

Wu et al., 2016

Motivation

Natural Language Processing

- Question answering

```
I: Jane went to the hallway.
I: Mary walked to the bathroom.
I: Sandra went to the garden.
I: Daniel went back to the garden.
I: Sandra took the milk there.
Q: Where is the milk?
A: garden

I: It started boring, but then it got interesting.
Q: What's the sentiment?
A: positive
```

Kumar et al., 2015

Motivation

Natural Language Processing

- Text generation

System Prompt (human-written)

```
In a shocking finding, scientist discovered a herd of unicorns living
in a remote, previously unexplored valley, in the Andes
Mountains. Even more surprising to the researchers was the fact that
the unicorns spoke perfect English.
```

Model Completion (machine-written, 10 tries)

```
The scientist named the population, after their distinctive horn,
Ovid's Unicorn. These four-horned, silver-white unicorns were
previously unknown to science.

Now, after almost two centuries, the mystery of what sparked this odd
phenomenon is finally solved.

Dr. Jorge Pérez, an evolutionary biologist from the University of La
Paz, and several companions, were exploring the Andes Mountains when
they found a small valley, with no other animals or humans. Pérez
noticed that the valley had what appeared to be a natural fountain,
surrounded by two peaks of rock and silver snow.
```

Motivation

Text-to-Image Synthesis

Romback et al., 2022

Motivation

Text-to-Video, Text-to-3D

“A dog wearing a superhero outfit with red cape flying through the

“a raccoon astronaut holding his helmet“

Singer et al. “Make-a-Video: Text-to-Video Generation without Text-Video Data”. In: arXiv Preprint 2022. https://make-a-video.github.io/
Poole et al. “DreamFusion: Text-to-3D using 2D Diffusion”. In arXiv Preprint 2022. https://dreamfusion3d.github.io/

Motivation

Reinforcement Learning

- Self-trained, plays 49 games at human level.

Mnih et al., 2016

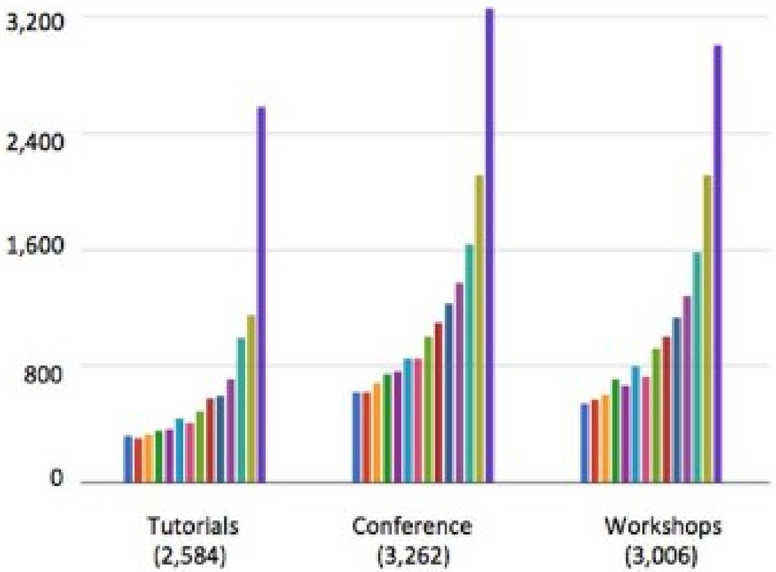
Motivation

Robotics

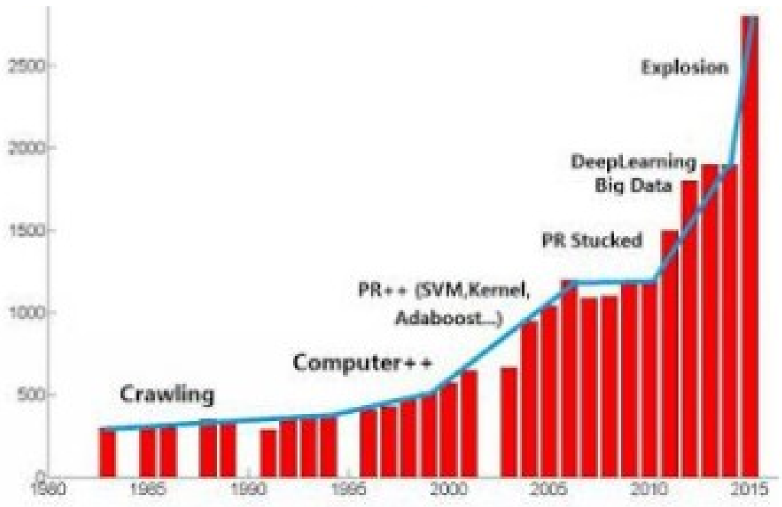
Motivation

Very very popular!

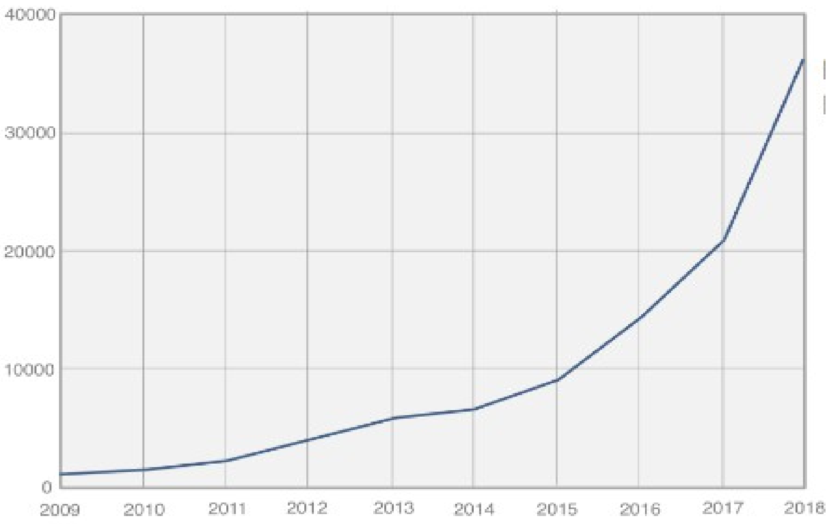
NIPS



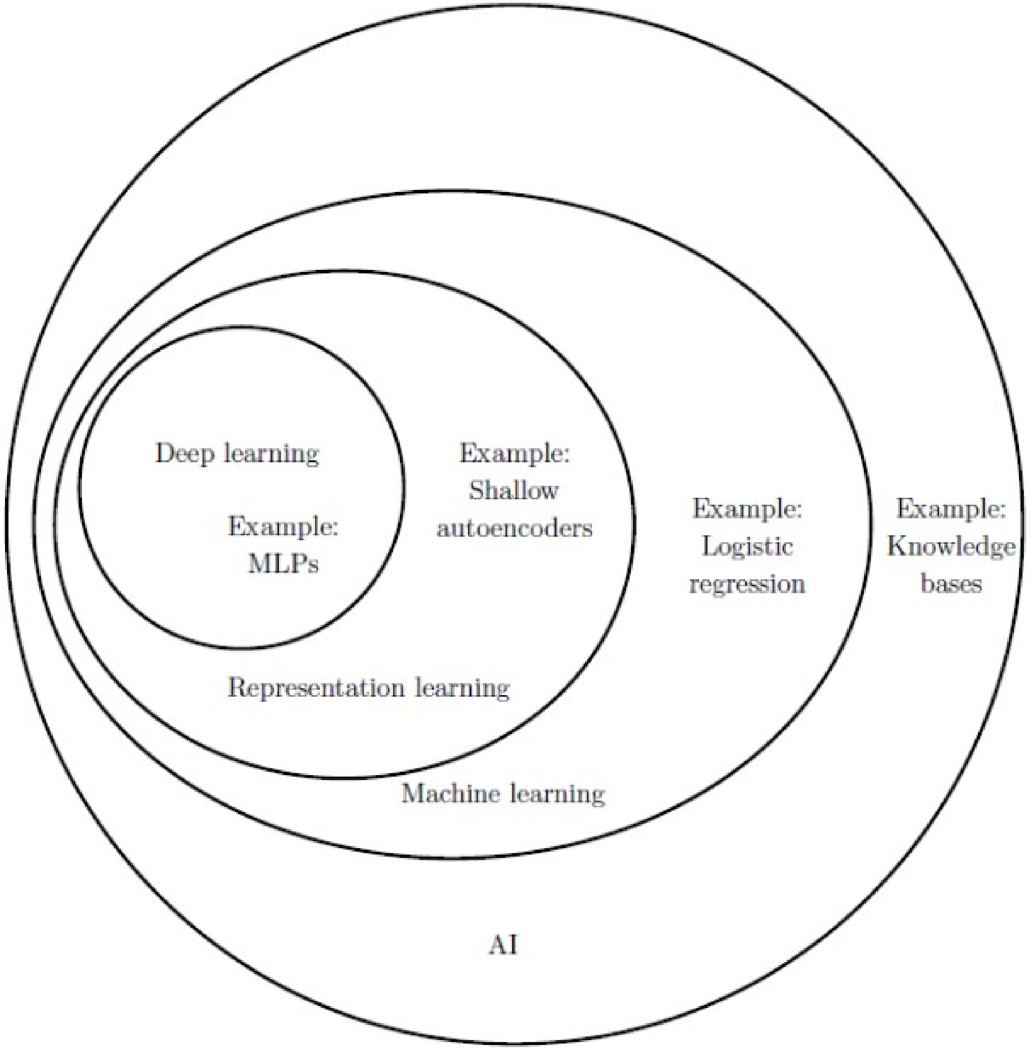
CVPR



arXiv



Deep Learning in the context of AI



Goodfellow et al., 2016

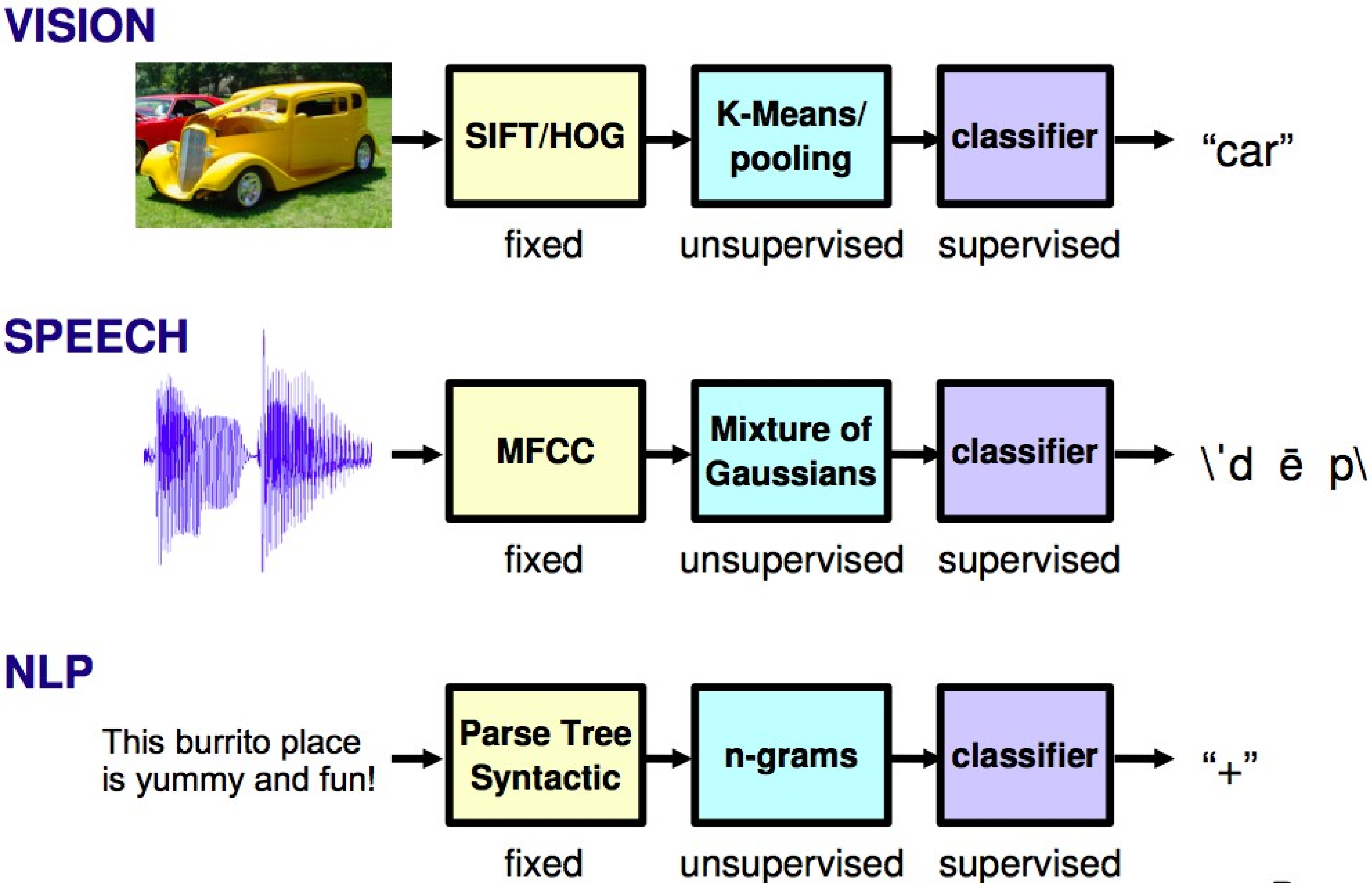
Why should we be impressed?

- Vision is ultra challenging!
  - For 256x256 resolution -> $25^{24,288}$ of possible images
  - Large visual object variations (viewpoints, scales, deformations, occlusions)
  - Large semantic object variations
- Robotics is typically considered in controlled environment
- Game AI involves extreme number of possible games states ($10^{10^{(48)}}$) possible GO games)
- NLP is extremely high dimensional and vague (just for English: 150K words)

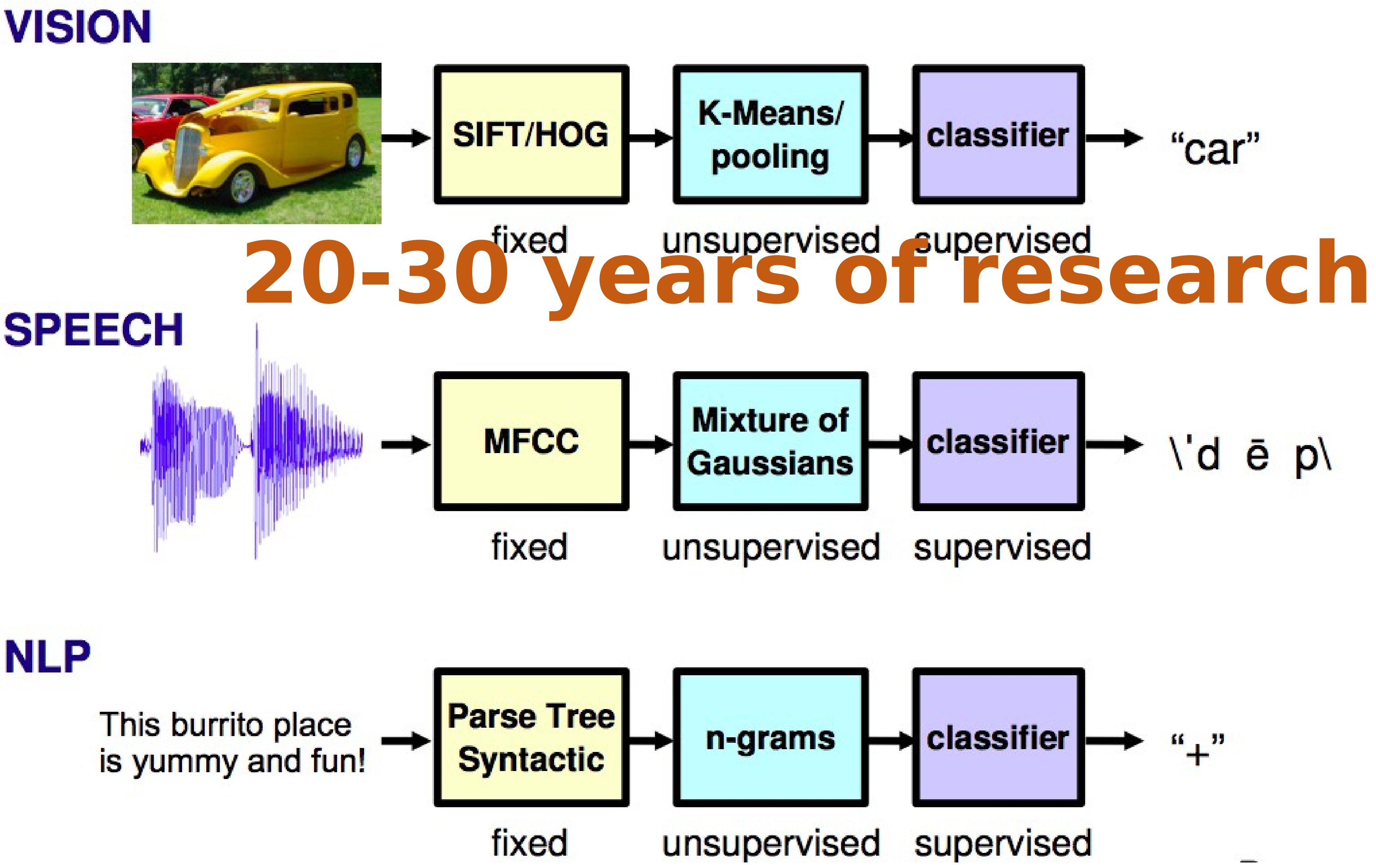




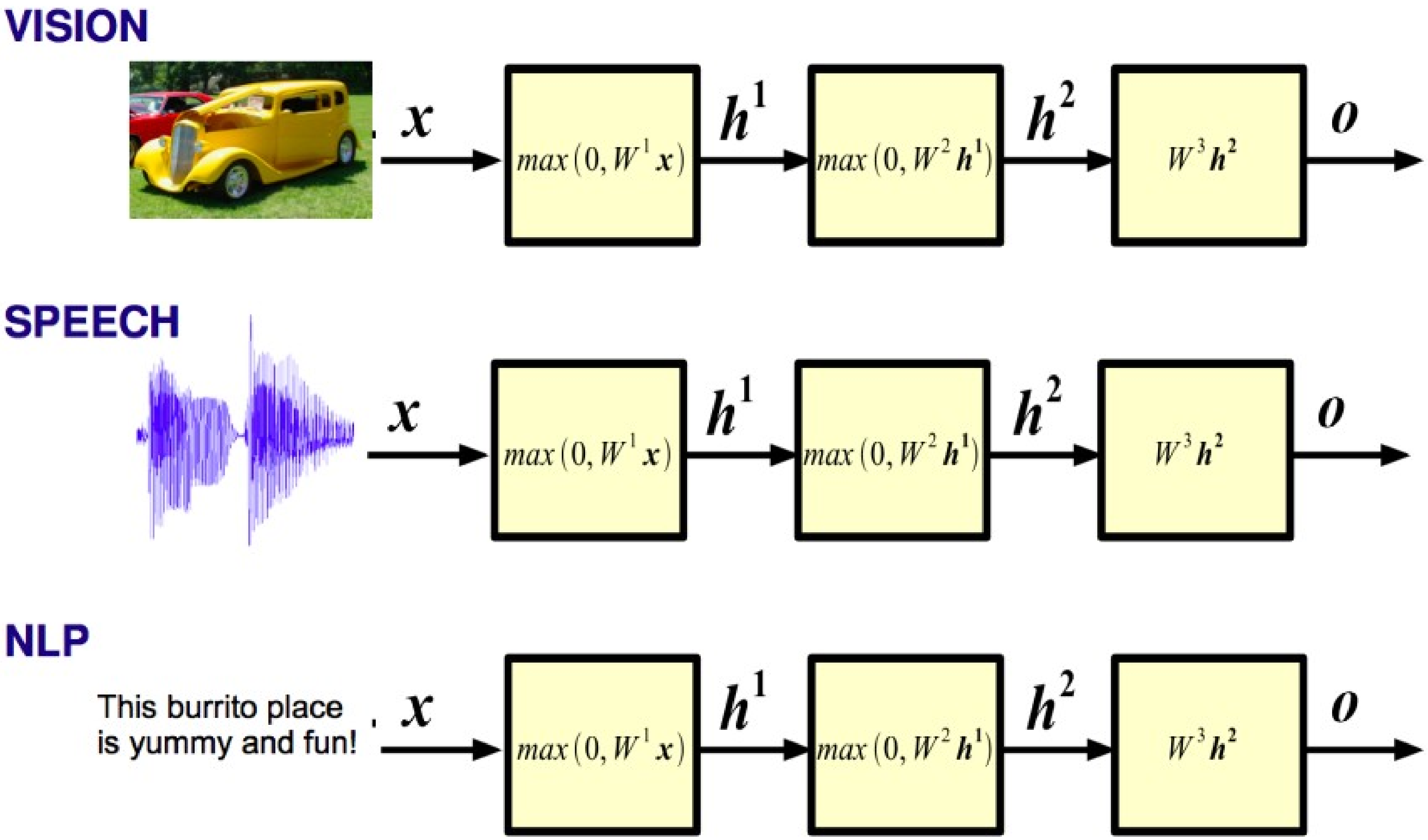
Machine Learning for X: features for X + ML



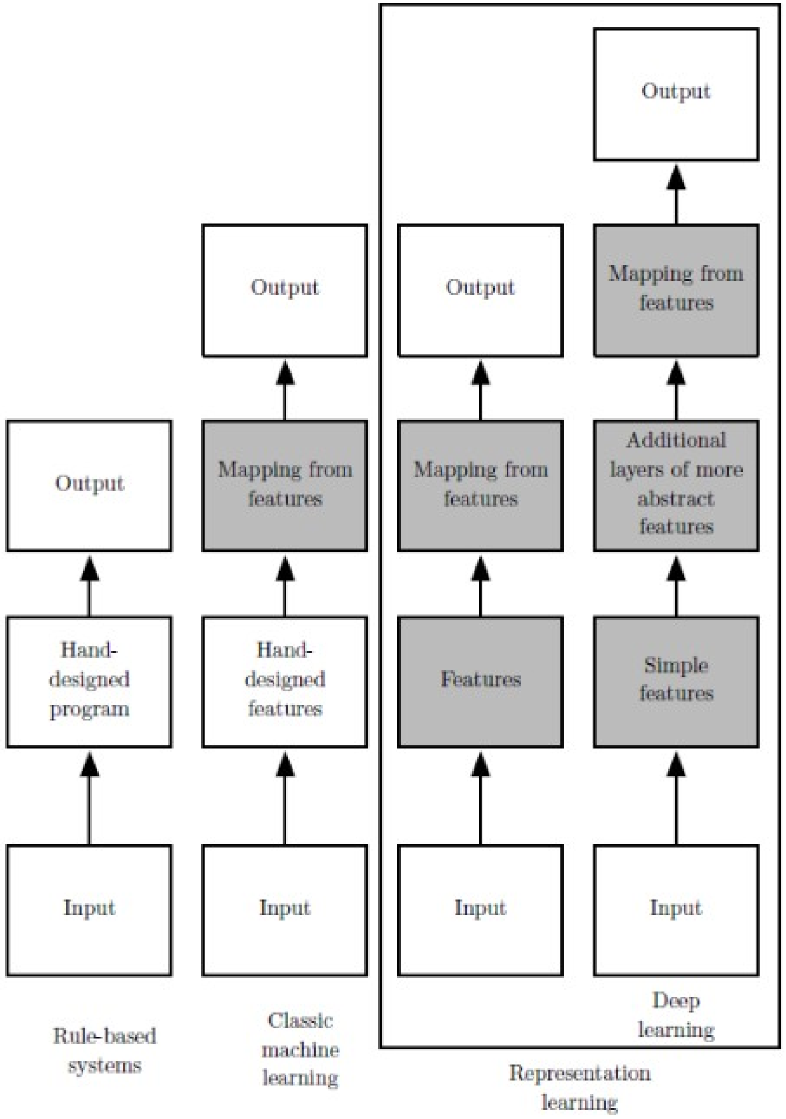
Machine Learning for X: features for X + ML



Deep Learning breakthrough for all of AI



Hand-designed vs Learnable Schemes



Goodfellow et al., 2016

Why learn the features and not just design them?

- Designing features manually is too time consuming and requires expert knowledge
- Learned features give us a better understanding of the data
- Learned features are more compact and specific for the task at hand
- Learned features are easy to adapt

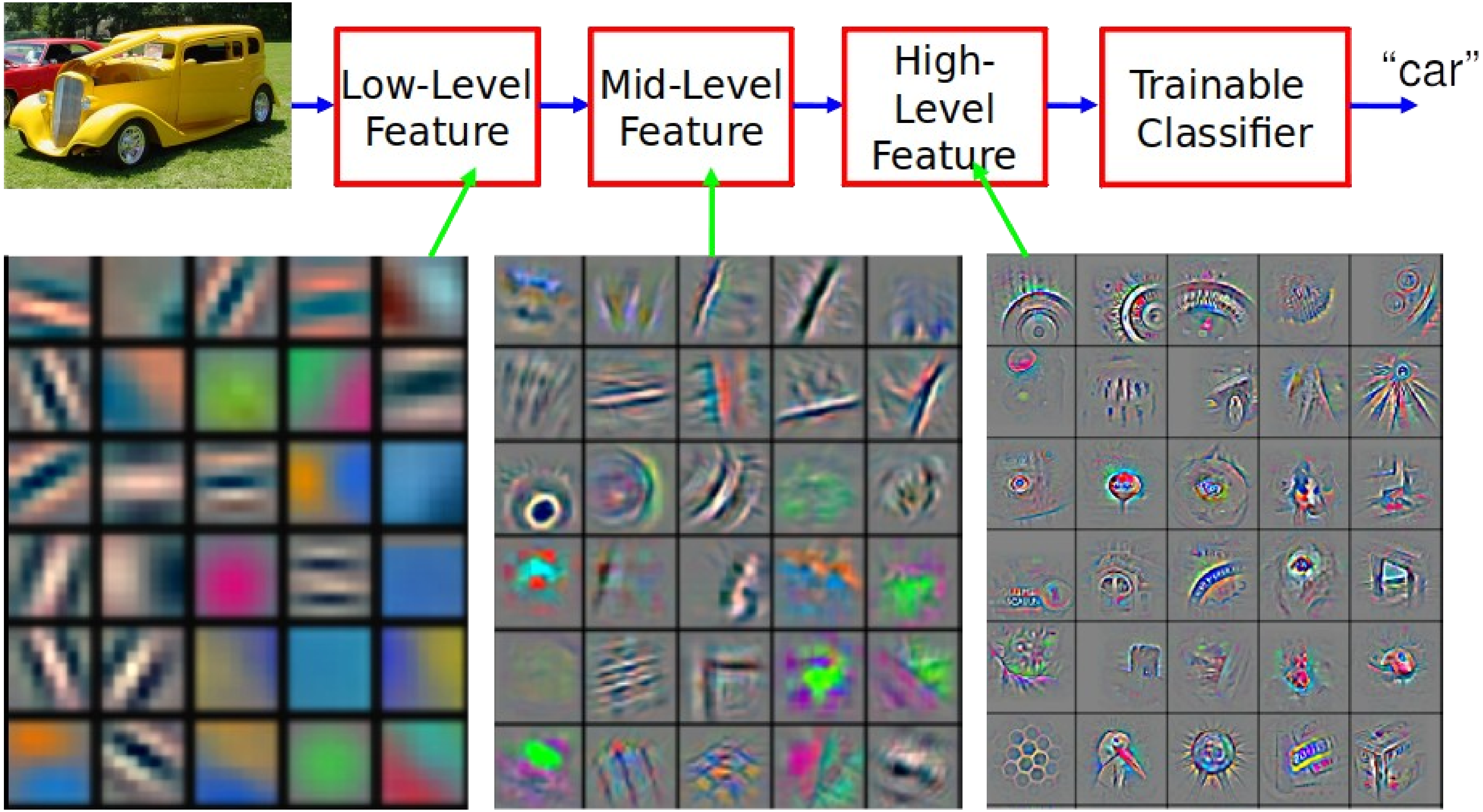
Why learn the features?

- Manually designed features
  - Expensive to research & validate

- Learned features
  - If data is enough, easy to learn, compact and specific

- Time spent for designing features now spent for designing architectures

Deep Learning = Hierarchical compositionality



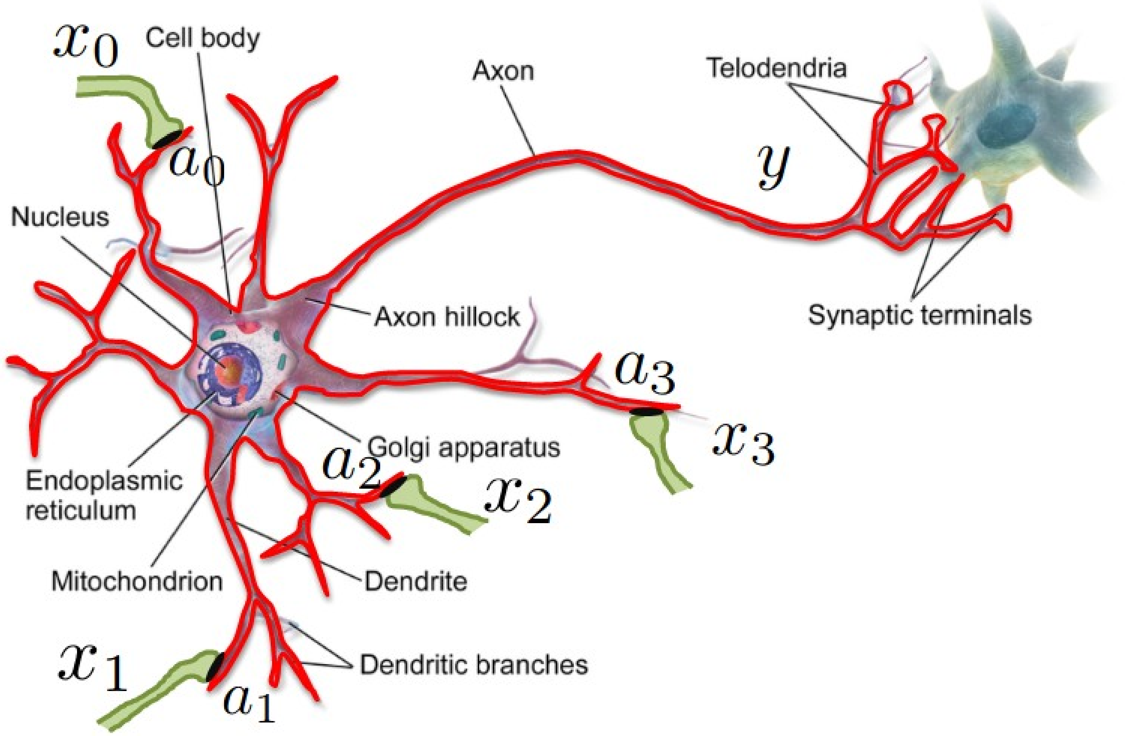
Feature visualization of convolutional net trained on ImageNet from [Zeiler & Fergus 2013]

History of Neural Networks & Deep Learning

First appearance (roughly)

Perceptron – The Idea

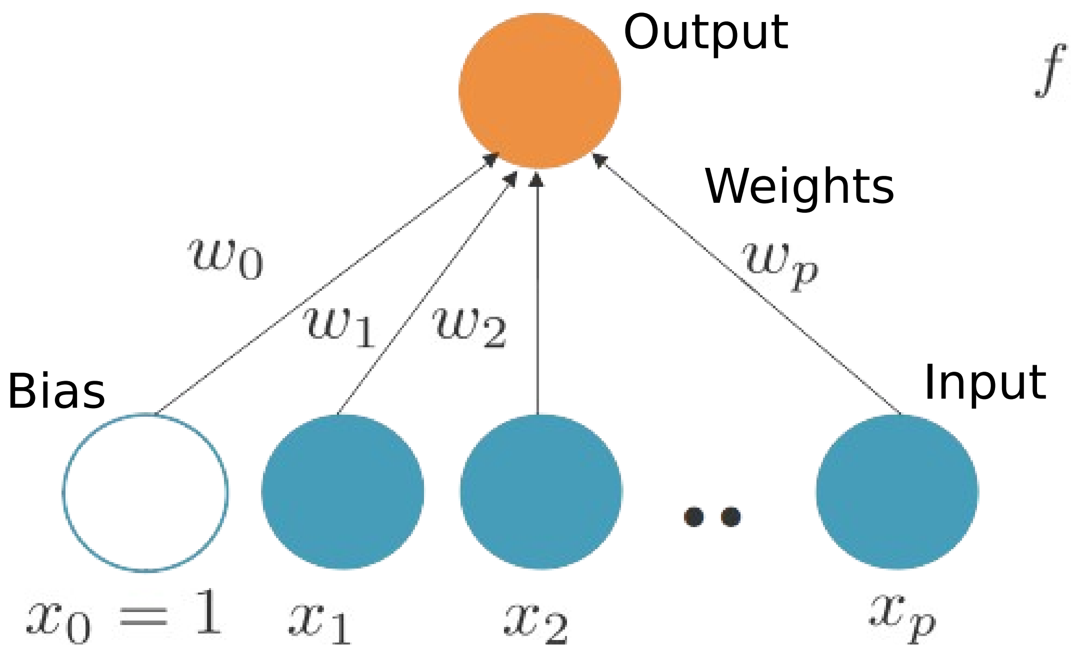
- Biology inspired



$$
\begin{aligned}
y &= \mathrm{sign}(a_0 x_0 + a_1 x_1 + a_2 x_2 + a_3 x_3) \\
&= \mathrm{sign}(\mathbf{a}^\top \mathbf{x})
\end{aligned}
$$

Perceptron

- Perceptron [Rosenblatt 1962]: A linear discriminant model for binary classification

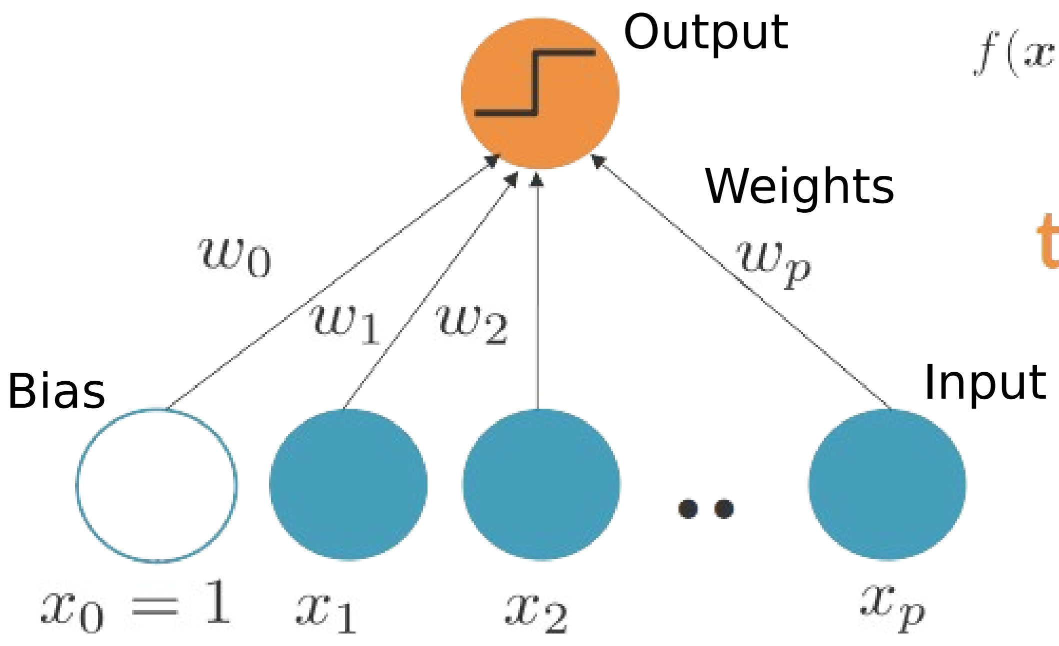


$$f(\boldsymbol{x}) = \sum_{j=1}^{p} w_j x_j + w_0 = \boldsymbol{w}^\top \boldsymbol{x}$$

How can we do classification?

Perceptron

- Perceptron [Rosenblatt 1962]: A linear discriminant model for binary classification



$$f(\boldsymbol{x}) = s\left(w_0 + \sum_{j=1}^{p} w_j x_j\right) = s(\boldsymbol{w}^\top \boldsymbol{x})$$

threshold function

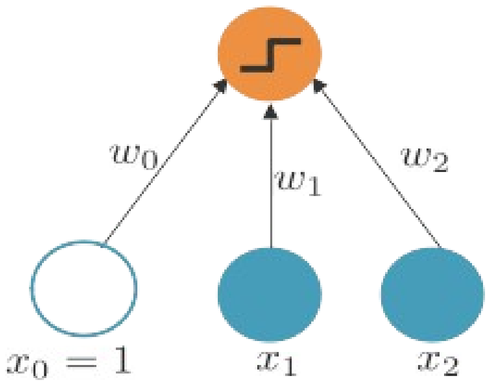
$$s(o(\boldsymbol{x})) = \begin{cases} 1 & \text{if } o(\boldsymbol{x}) > 0 \\ 0 & \text{otherwise.} \end{cases}$$

The decision boundary is a hyperplane

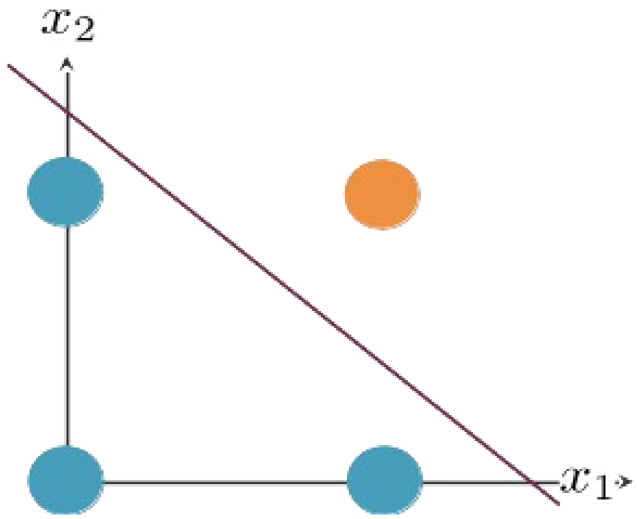
Perceptron - Example

- Example of Perceptron: Learn the operation AND

x1	x2	y
0	0	0
0	1	0
1	0	0
1	1	1



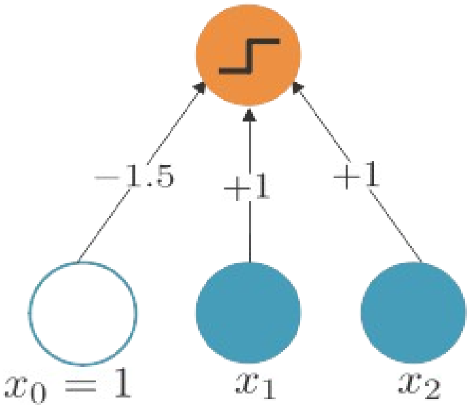
$$f(x) = s(w_0 + w_1 x_1 + w_2 x_2)$$



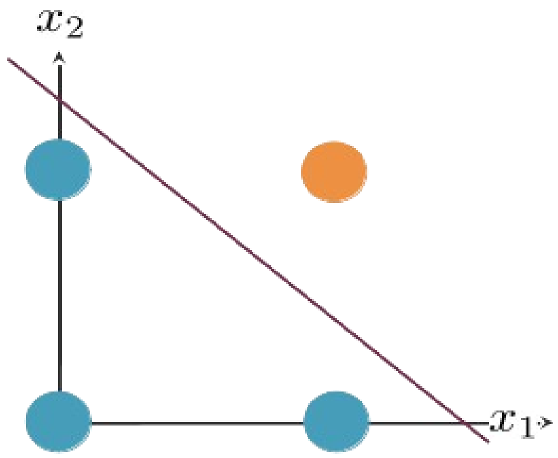
Perceptron - Example

- Example of Perceptron: Learn the operation AND

x1	x2	y
0	0	0
0	1	0
1	0	0
1	1	1



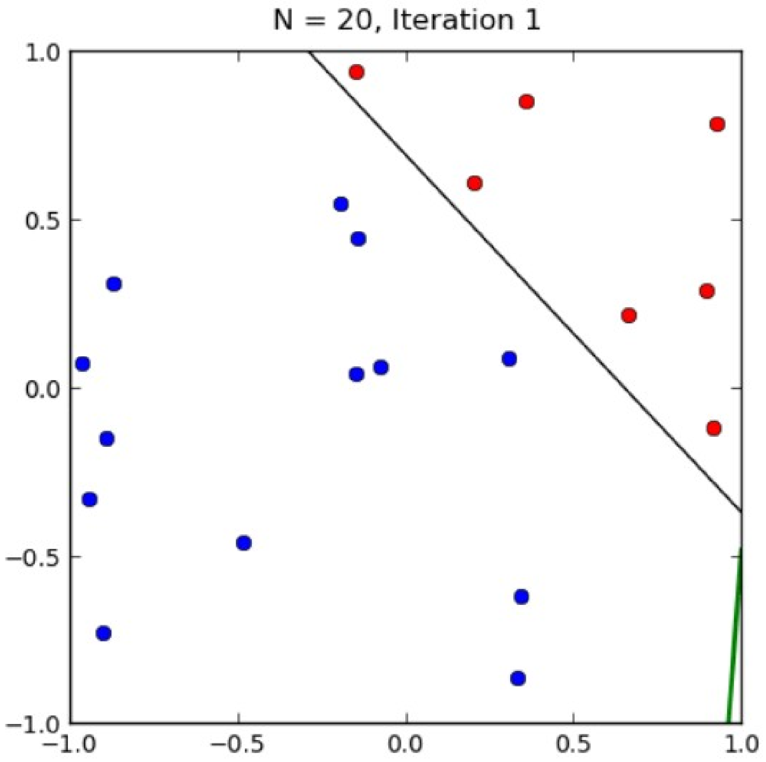
$$f(x) = s(w_0 + w_1 x_1 + w_2 x_2)$$



x1	x2	f(x)
0	0	s(-1.5 + 0 + 0) = s(-1.5) = 0
0	1	s(-1.5 + 0 + 1) = s(-0.5) = 0
1	0	s(-1.5 + 1 + 0) = s(-0.5) = 0
1	1	s(-1.5 + 1 + 1) = s(0.5) = 1

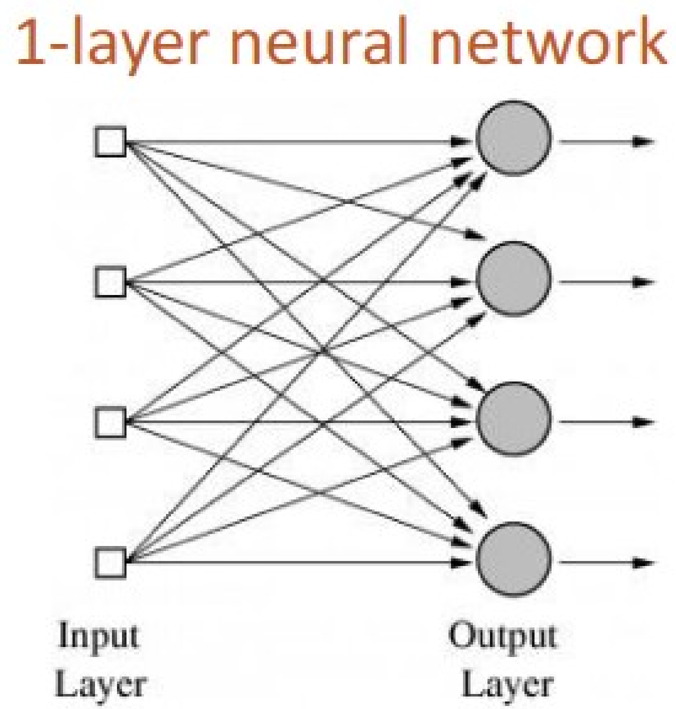
Training a Perceptron

- Rosenblatt's innovation was mainly the learning algorithm for perceptrons
- Learning algorithm
  - Initialize weights randomly
  - Take one sample x_i and predict y_i
  - For erroneous predictions update weights
    - If prediction y' = 0 and ground truth y_i = 1, increase weights
    - If prediction y' = 1 and ground truth y_i = 0, decrease weights
  - Repeat until no errors are made



From a single layer to multiple layers

- 1 perceptron == 1 decision
- What about multiple decisions?
  - E.g. digit classification
- Stack as many outputs as the possible outcomes into a layer
  - Neural Network



Q: What is a potential problem with perceptrons?

- They can only return one output, so only work for binary problems
- They are linear machines, so can only solve linear problems
- They can only work for vector inputs
- They are too complex to train, so they can work with big computers only

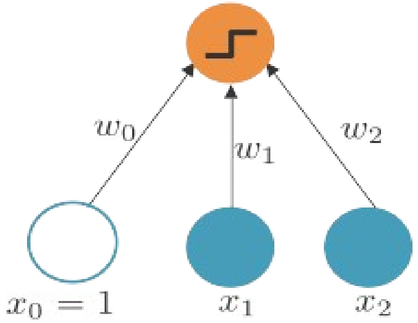
Q: What is a potential problem with perceptrons?

- They can only return one output, so only work for binary problems
- **They are linear machines, so can only solve linear problems**
- They can only work for vector inputs
- They are too complex to train, so they can work with big computers only

Perceptron - Example

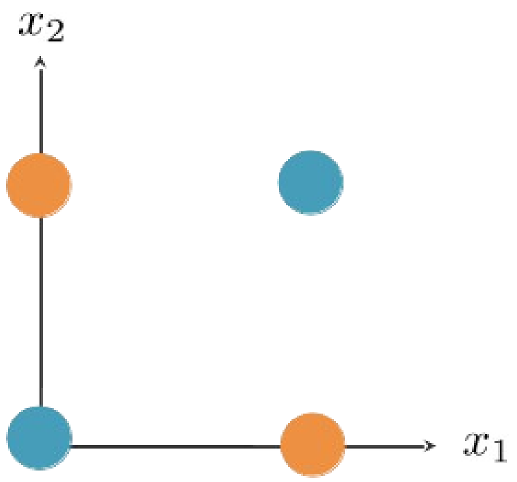
- Example of Perceptron: Learn the operation XOR

x1	x2	y
0	0	0
0	1	1
1	0	1
1	1	0



[Minsky and Papert, 1969]
There is no combination:

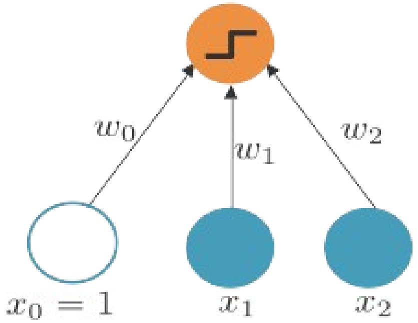
$$f(x) = s(w_0 + w_1 x_1 + w_2 x_2)$$



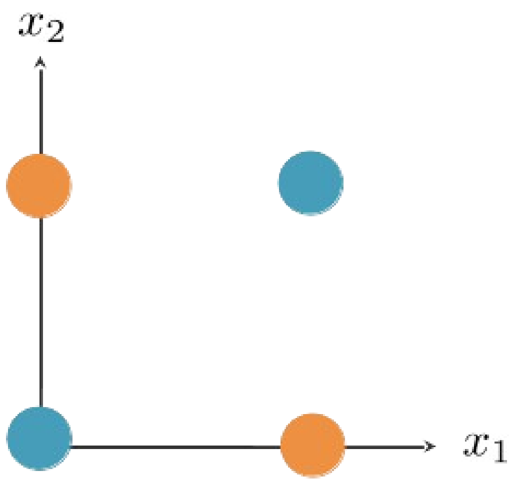
Perceptron - Example

- Example of Perceptron: Learn the operation XOR

x1	x2	y
0	0	0
0	1	1
1	0	1
1	1	0



$$f(x) = s(w_0 + w_1 x_1 + w_2 x_2)$$



[Minsky and Papert, 1969]
There is no combination:

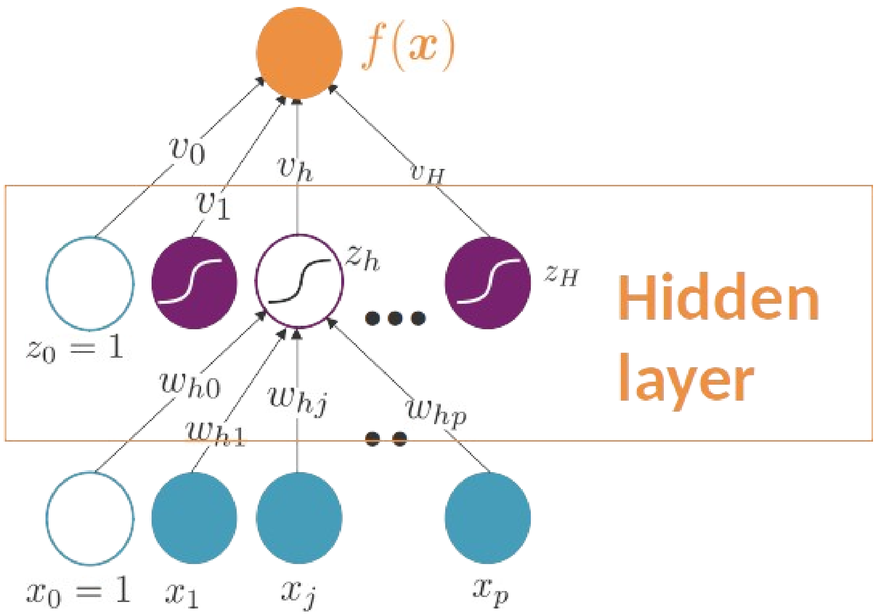
$$\begin{aligned} w_0 &\leq 0 \\ w_0 + w_2 &> 0 \\ w_0 + w_1 &> 0 \\ w_0 + w_1 + w_2 &\leq 0 \end{aligned}$$

Minsky & Multi-layer perceptrons

- Interestingly, Minsky **never said** XOR is unsolvable by neural networks
  - Only that XOR cannot be solves with **1 layer** perceptrons
- Multi-layer perceptrons can solve XOR
  - 9 year earlier Minsky built such a multi-layer perceptron
  - Any continuous function on a compact subset of $\mathbb{R}^n$ can be approximated to any arbitrary degree of precision by a feed-forward multi-layer perceptron with a single hidden layer containing a finite number of neurons.

[Cyberno 1989, Hornik, 1991]

- However, how to train a multi-layer perceptron?
- Rosenblatt's algorithm not applicable

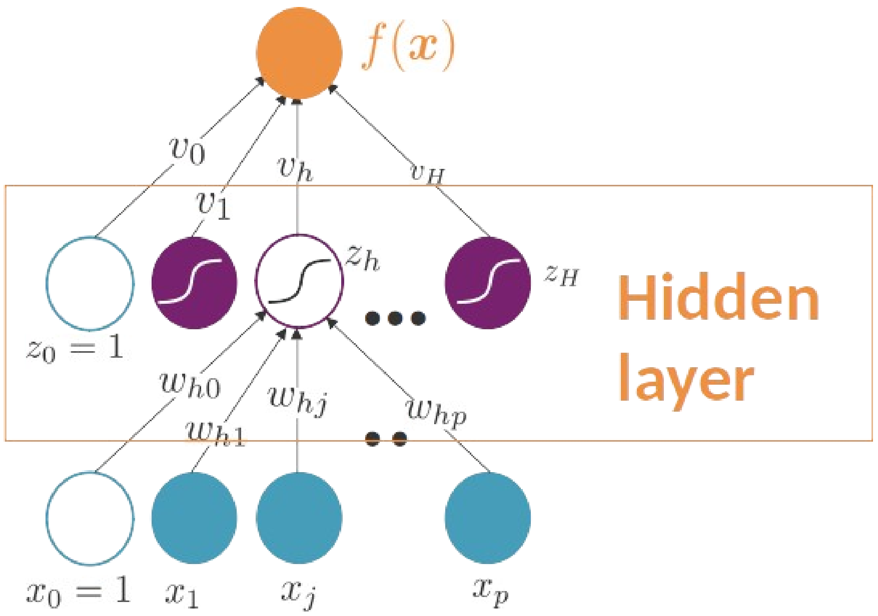


Minsky & Multi-layer perceptrons

- Interestingly, Minsky **never said** XOR is unsolvable by neural networks
  - Only that XOR cannot be solves with **<u>1 layer</u>** perceptrons
- <u>Multi</u>-layer perceptrons <u>can solve</u> XOR
  - 9 year earlier Minsky built such a multi-layer perceptron
  - Any continuous function on a compact subset of $\mathbb{R}^n$ can be approximated to any arbitrary degree of precision by a feed-forward multi-layer perceptron with a single hidden layer containing a finite number of neurons.

[Cyberno 1989, Hornik, 1991]

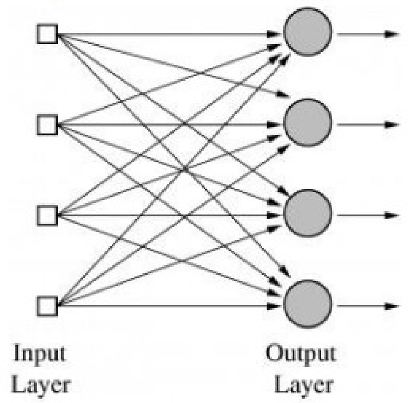
- Problem: how to train a multi-layer perceptron?
  - Rosenbatt's algorithm not applicable
  - It expects to know the ground truth **z_h'** for the **z_h**
  - For the output layers we have the ground truth labels
  - For intermediate hidden layers we don't



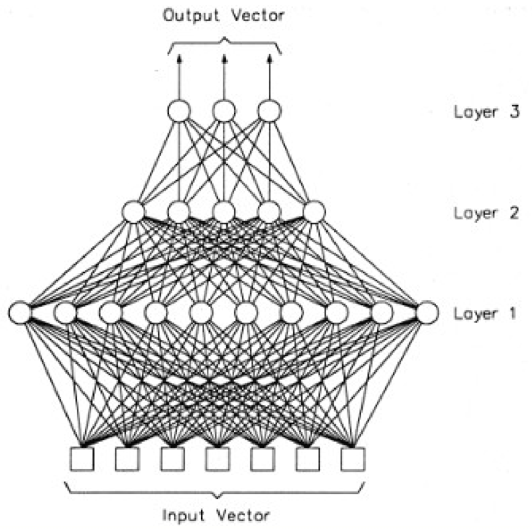
From a single layer to multiple layers

- 1 perceptron == 1 decision
- What about multiple decisions?
  - E.g. digit classification
- Stacks as many outputs as the possible outcomes into a layer
  - Neural Networks
- Use one layer as input to the next layer
  - Add nonlinearities between layers
  - Multi-layer perceptron (MLP)

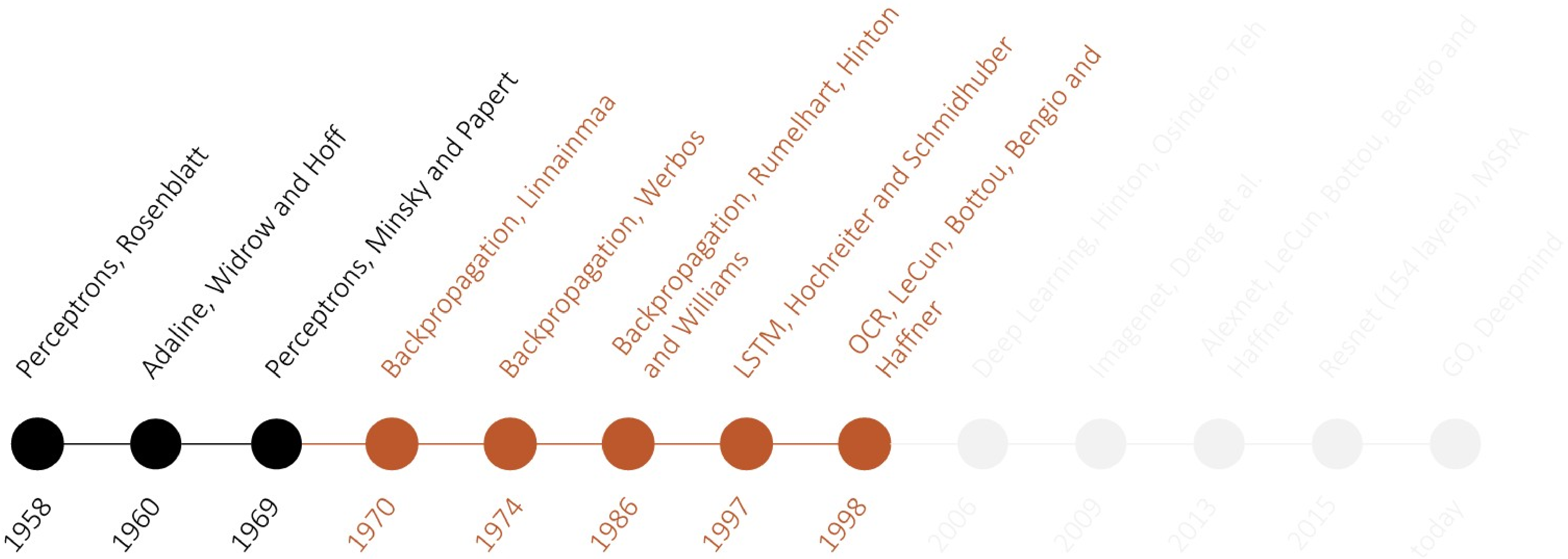
1-layer neural network



Multi-layer perceptron



The "AI winter" despite notable successes



The first ''AI winter''

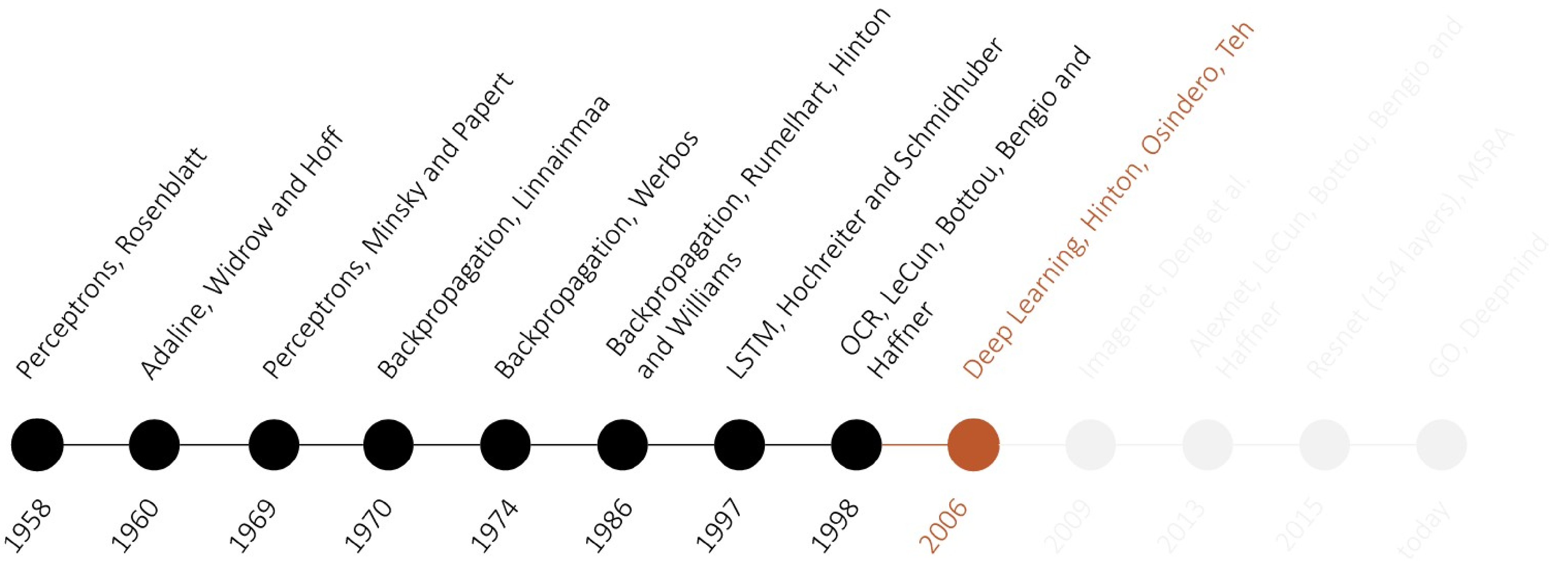
- What everybody thought: "If a perceptron cannot even solve XOR, why bother?"
- Results not as promised (too much hype!) -> no further funding -> AI Winter
- Still, significant discoveries were made in this period
  - Backpropagation -> Learning algorithm for MLPs
  - Recurrent networks -> Neural Networks for infinite sequences

The second ''AI winter''

- Concurrently with Backprop and Recurrent Nets, new and promising Machine Learning models were proposed

- Kernel Machines & Graphical Models
  - Similar accuracy with better math and proofs and fewer heuristics
  - Neural networks could not improve beyond a few layers

The thaw of the "AI winter"



Neural Network problems a decade ago

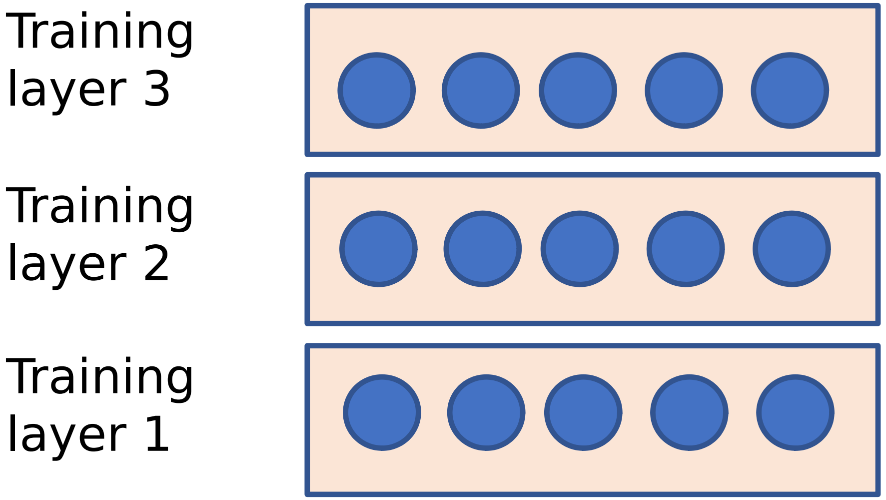
- Lack of processing power
- Lack of data
- Overfitting
- Vanishing gradients
- Experimentally, training multi-layer perceptrons was not that useful
  - Accuracy didn't improve with more layers
  - Are 1-2 hidden layers the best neural networks can do?

Deep Learning arrives

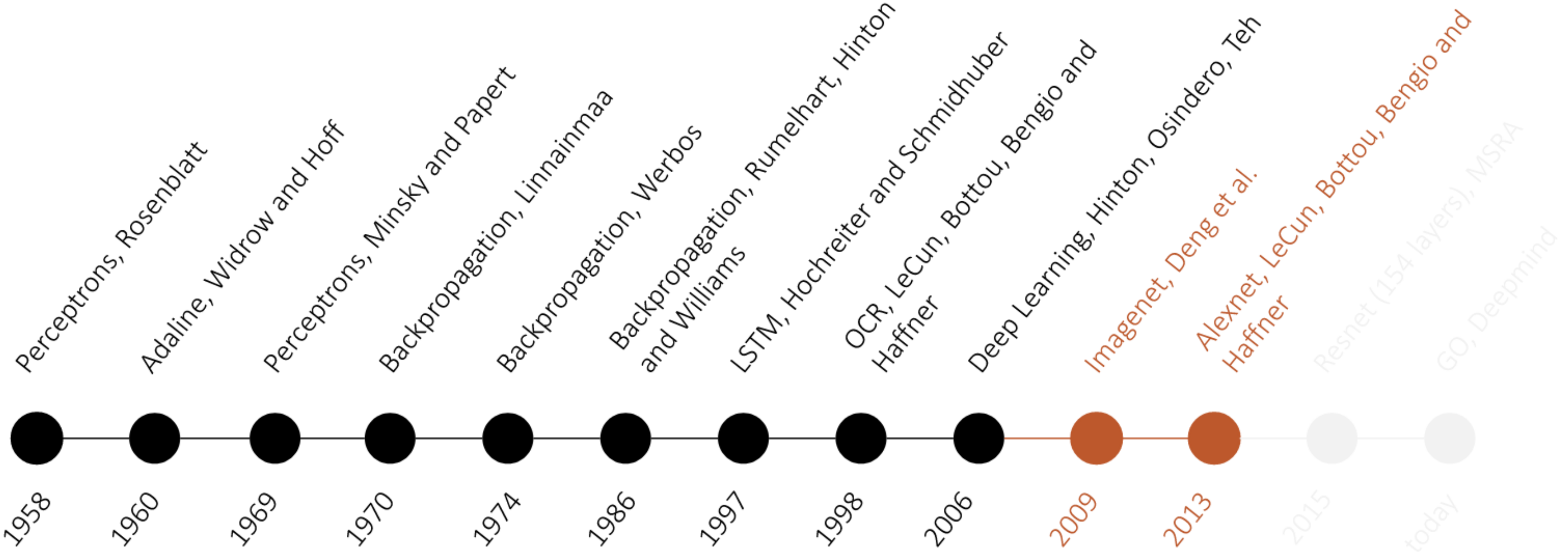
- Layer-by-layer training

- Training multi-layered neural networks became easier

- Per-layer trained parameters initialize further training using constructive divergence



Deep Learning Renaissance

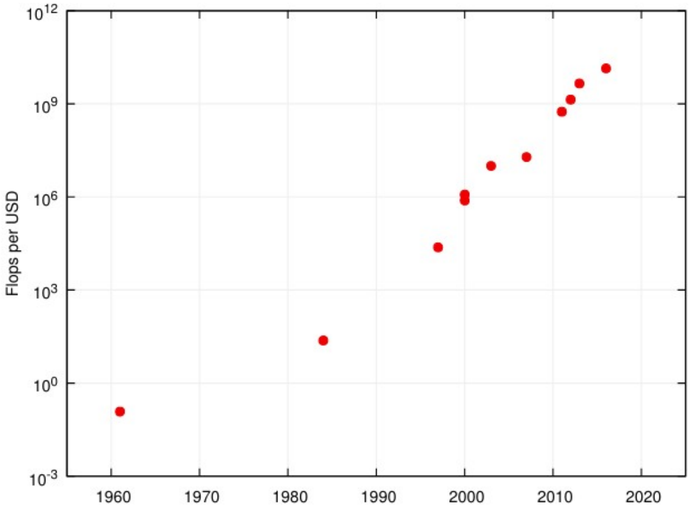


Why now?

- Five decades of research in machine learning
  - A taxonomy of ML concepts (classification, generative models, clustering, kernels, linear embeddings, etc)
  - A sound statistical formalization (Bayesian estimation, PAC)
  - A clear picture of fundamental issues (bias/variance dilemma, VC dimension, generalization bounds, etc.)
  - A good understanding of optimization issues
  - Efficient large-scale algorithms

Why now?

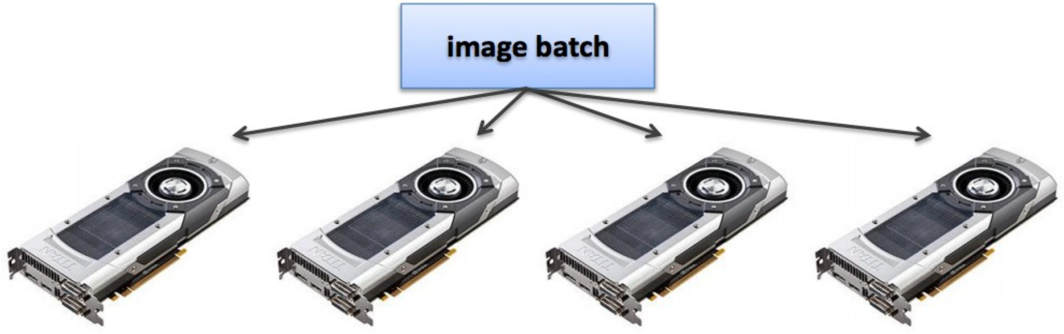
- Five decades of research in machine learning
- CPUS/GPUs/storage developed for other purposes



(Wikipedia "FLOPS")

	TFlops (10^{12})	Price	GFlops per $
Intel i7-6700K	0.2	$344	0.6
AMD Radeon R-7 240	0.5	$55	9.1
NVIDIA GTX 750 Ti	1.3	$105	12.3
AMD RX 480	5.2	$239	21.6
NVIDIA GTX 1080	8.9	$699	12.7

- Multiple GPU support
  - 4 x NVIDIA Titan, off-the-shelf workstation
  - data parallelism for training and testing
  - ~3.75 times speed-up, 2-3 weeks for training



Why now?

- Five decades of research in machine learning
- CPUS/GPUs/storage developed for other purposes
- Lots of data from "the internet"

Data-set	Year	Nb. images	Resolution	Nb. classes
MNIST	1998	6.0×10^4	28×28	10
NORB	2004	4.8×10^4	96×96	5
Caltech 101	2003	9.1×10^3	$\simeq 300 \times 200$	101
Caltech 256	2007	3.0×10^4	$\simeq 640 \times 480$	256
LFW	2007	1.3×10^4	250×250	–
CIFAR10	2009	6.0×10^4	32×32	10
PASCAL VOC	2012	2.1×10^4	$\simeq 500 \times 400$	20
MS-COCO	2015	2.0×10^5	$\simeq 640 \times 480$	91
ImageNet	2016	14.2×10^6	$\simeq 500 \times 400$	$21,841$
Cityscape	2016	25×10^3	$2,000 \times 1000$	30

Deep Learning is Data Hungry

- In 2009 the ImageNet dataset was published [Dong et al.,2009]
  - Collected images for each of the 100K terms in Wordnet (16M images in total)
  - Terms organized hierarchically: "Vehicle" -> "Ambulance"
- ImageNet Large Scale Visual Recognition Challenge (ILSVRC)
  - 1 million images
  - 1.000 classes
  - Top-5 and top-1 error measured

Some statistics for ImageNet

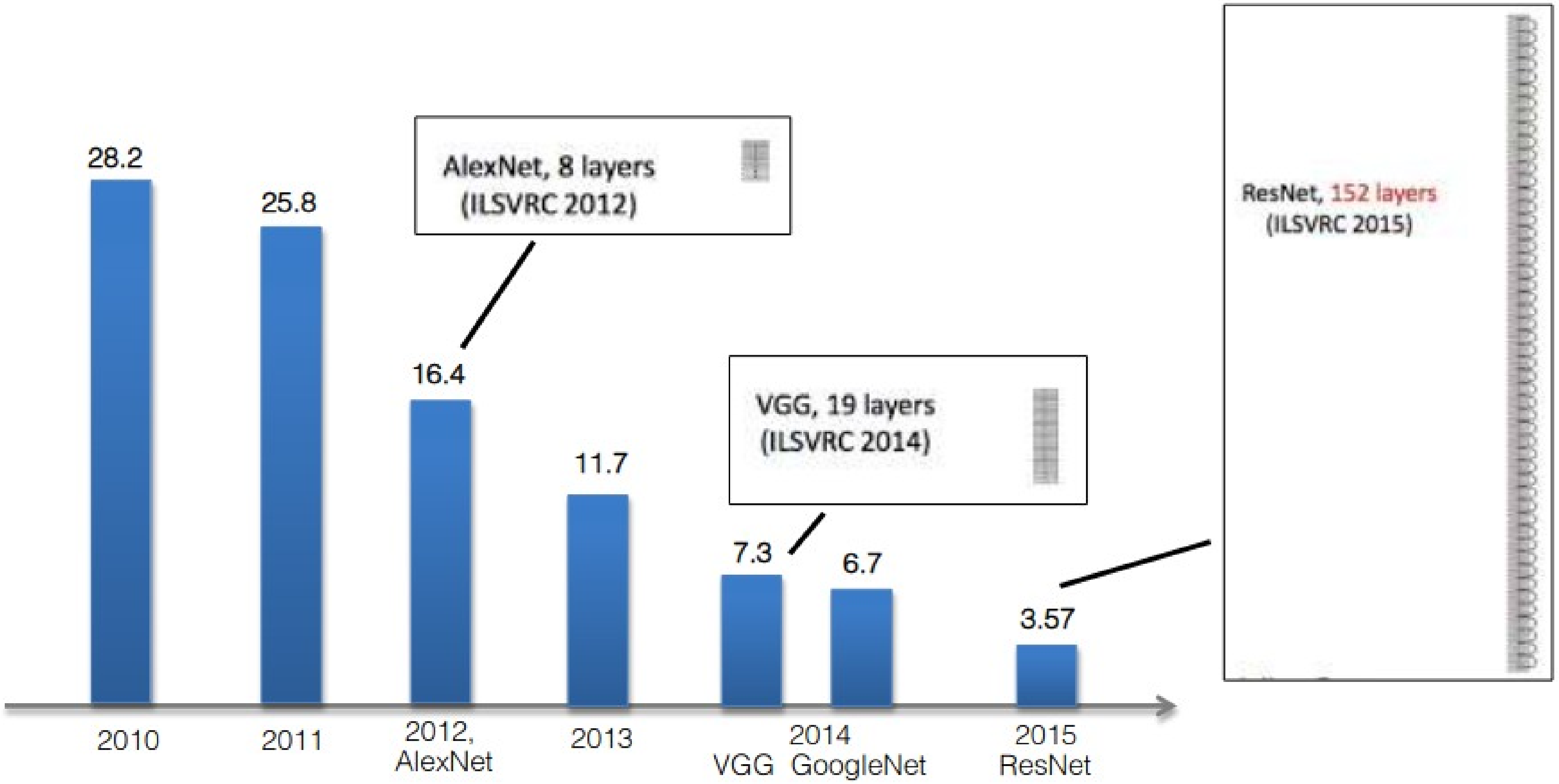
- July 2008: 0 images
- Dec 2008: 3 million images, 6K+ synsets
- April 2010: 11 million images, 15K+ synsets
- Currently: 14 million images, 21K sunsets indexed
- Today a Kaggle competition

- *2012: AlexNet the first deep learning based method, it boosts 10% the performance on Imagenet*

Some statistics for ImageNet

- July
- Dec
- Apri
- Curr
- Toda

- *201.* *t*
  boo:



Classification task on ImageNet challenge top-5 error

Why now?

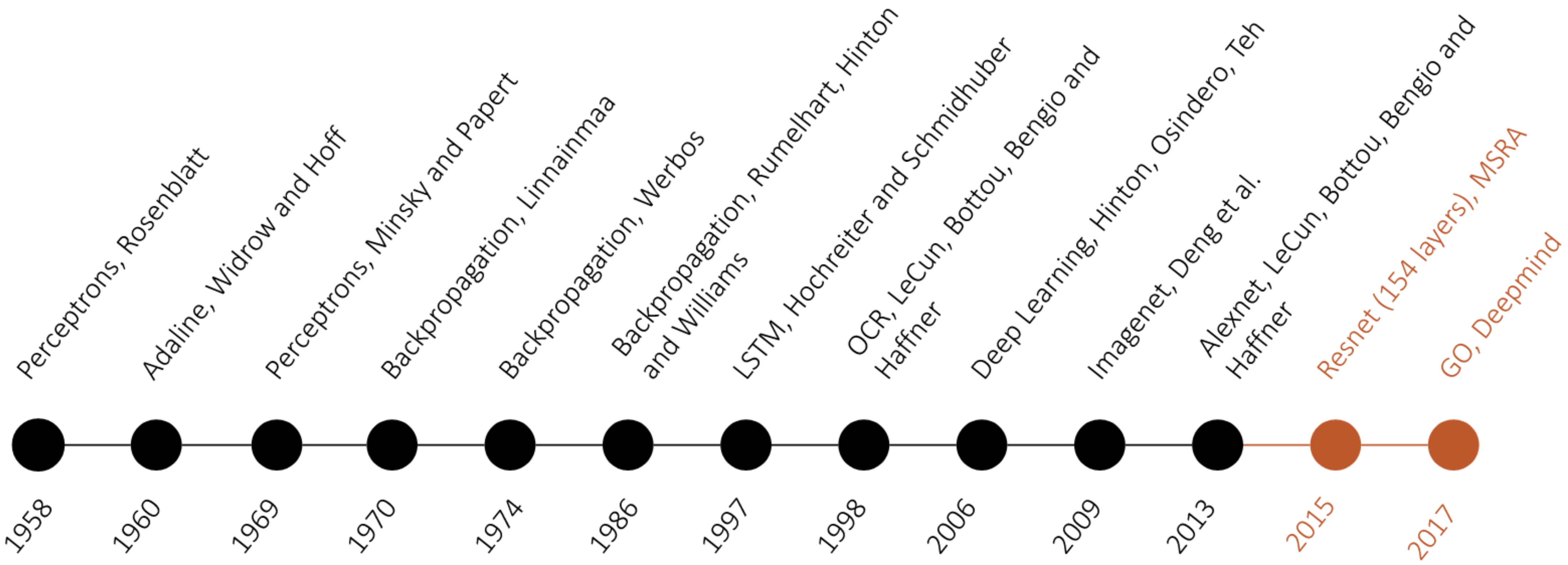
- Five decades of research in machine learning
- CPUS/GPUs/storage developed for other purposes
- Lots of data from "the internet"
- Tools and culture of collaborative and reproducible science

	Language(s)	**License**	**Main backer**
PyTorch	**Python**	BSD	Facebook
Caffe2	C++, Python	Apache	Facebook
TensorFlow	Python, C++	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

Why now?

- Five decades of research in machine learning
- CPUS/GPUs/storage developed for other purposes
- Lots of data from "the internet"
- Tools and culture of collaborative and reproducible science
- Resources and efforts from large corporations

Deep Learning Golden Era



Deep Learning – The main idea

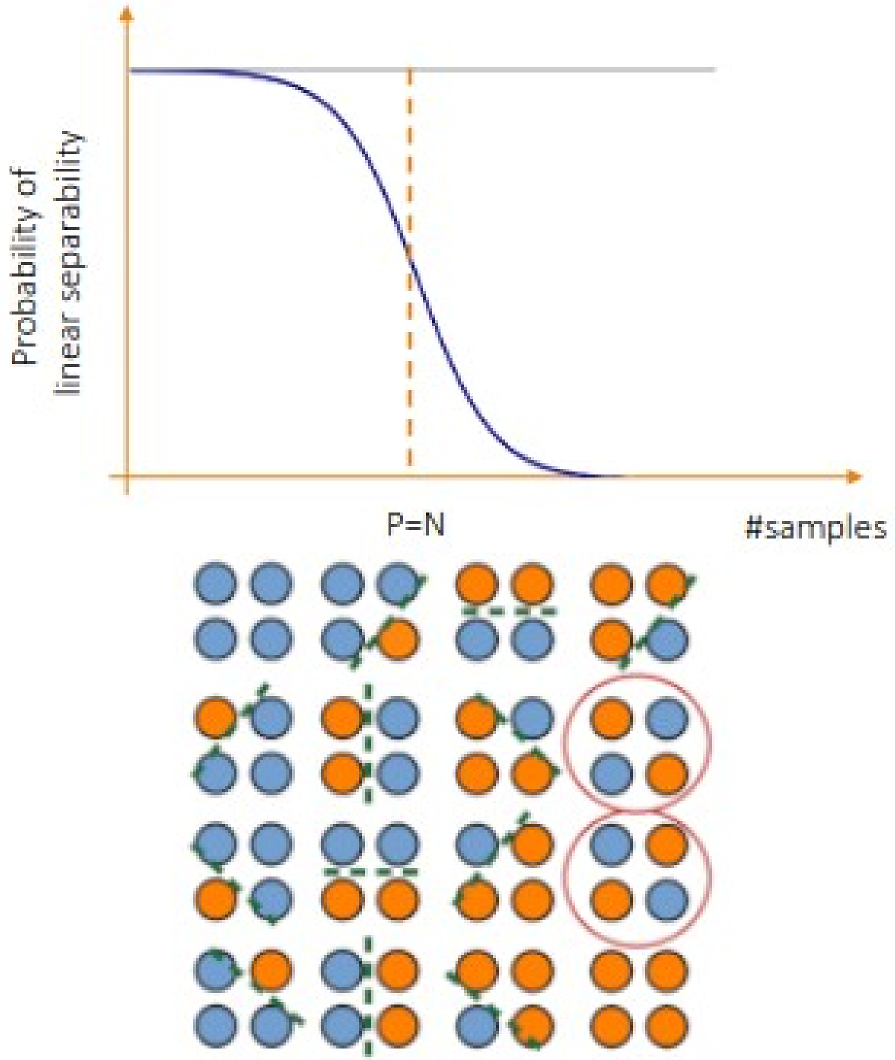
- A family of parametric, non linear and hierarchical representation learning functions, which are massively optimized with stochastic gradient descent to encode domain knowledge, i.e. domain invariances, stationarity.
- $h_1 = g(W_1x+b_1)$
- $a_L(x;\theta_{1,\ldots L}) = h_L(h_{L-1}(\ldots h_1(x,\theta_1),\theta_{L-1}),\theta_L)$
  - X: input, θ_L: parameters for layer l, $a_L = h_L(x, \theta_L)$: (non)linear function
- Given train in corpus {X,Y} find optimal parameters

$$\theta^* \leftarrow \arg\min_\theta \sum_{(x,y)\subseteq(X,Y)} \ell(y, a_L(x;\theta_{1,\ldots,L}))$$

Non-separability of linear machines

- $X = \{x_1, x_2, \dots, x_n\} \in \mathcal{R}^d$
- Given the n points there are in total 2^n dichotomies
- Only about d are linearly separable
- With n > d the probability X is linear separable converges to 0 very fast.
- The chances that a dichotomy is linearly separable is very small

- “Function Counting Theorem”. Cover, 1965



Non-linearizing linear machines

- Most data distributions and tasks are non-linear
- A linear assumption is often convenient, but not necessarily truthful
- **Problem:** How to get non-linear machines without too much effort?

Non-linearizing linear machines

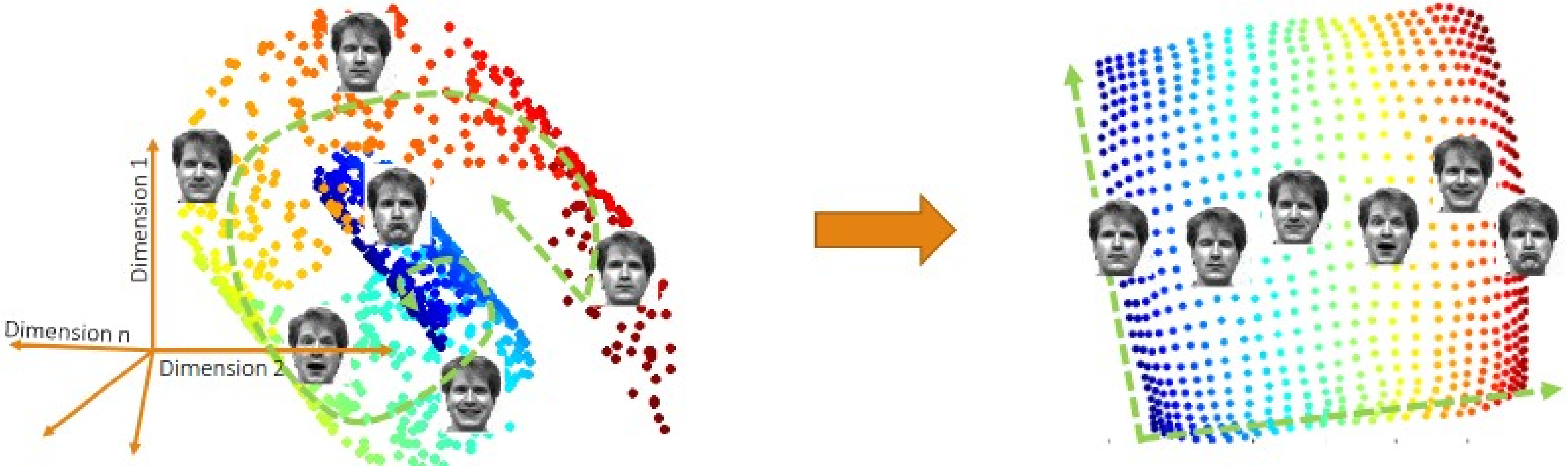
- Most data distributions and tasks are non-linear
- A linear assumption is often convenient, but not necessarily truthful
- **Problem:** How to get non-linear machines without too much effort?
- **Solution:** Make features non-linear
- What is a good non-linear feature?
  - Non-linear kernels, e.g. polynomial. RBF, etc
  - Explicit design of features (SIFT, HOG)?

Good features

- Invariant … but not too invariant
- Repeatable … but not bursty
- Discriminative … but not too class-specific
- Robust … but sensitive enough

Manifolds

- Raw data live in huge dimensionalities
- But, effectively lie in lower dimensional manifolds
- Can we discover this manifold to embed our data on?

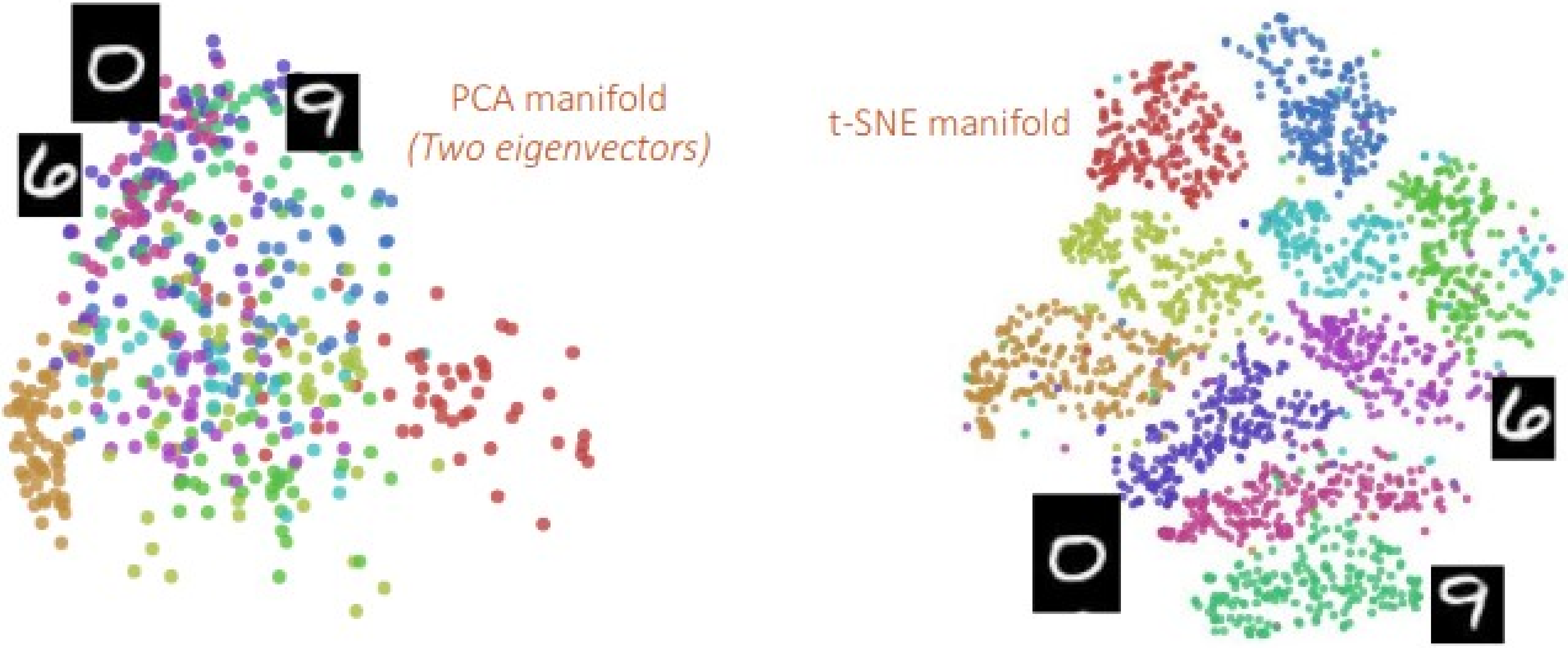


How to get good features?

- Goal: discover these lower dimensional manifolds
  - These manifolds are most probably highly non-linear
- First hypothesis: Semantically similar things lie closer together than semantically dissimilar things
- Second hypothesis: A face (or any other image) is a point on the manifold
  - Compute the coordinates of this point and use them as a feature
  - Face features will be separable

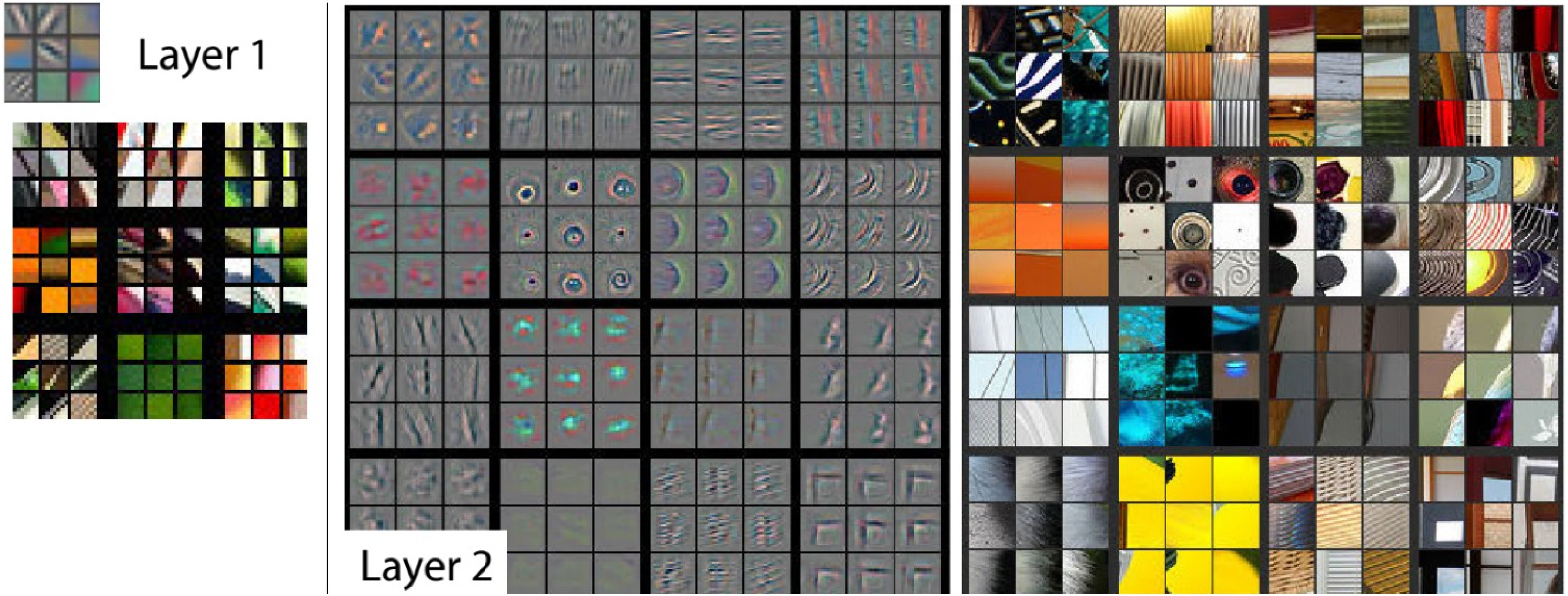
The digits manifolds

- There are good features (manifolds) and bad features
- 28 pixels x 28 pixels = 784 dimensions



End-to-end learning of feature hierarchies

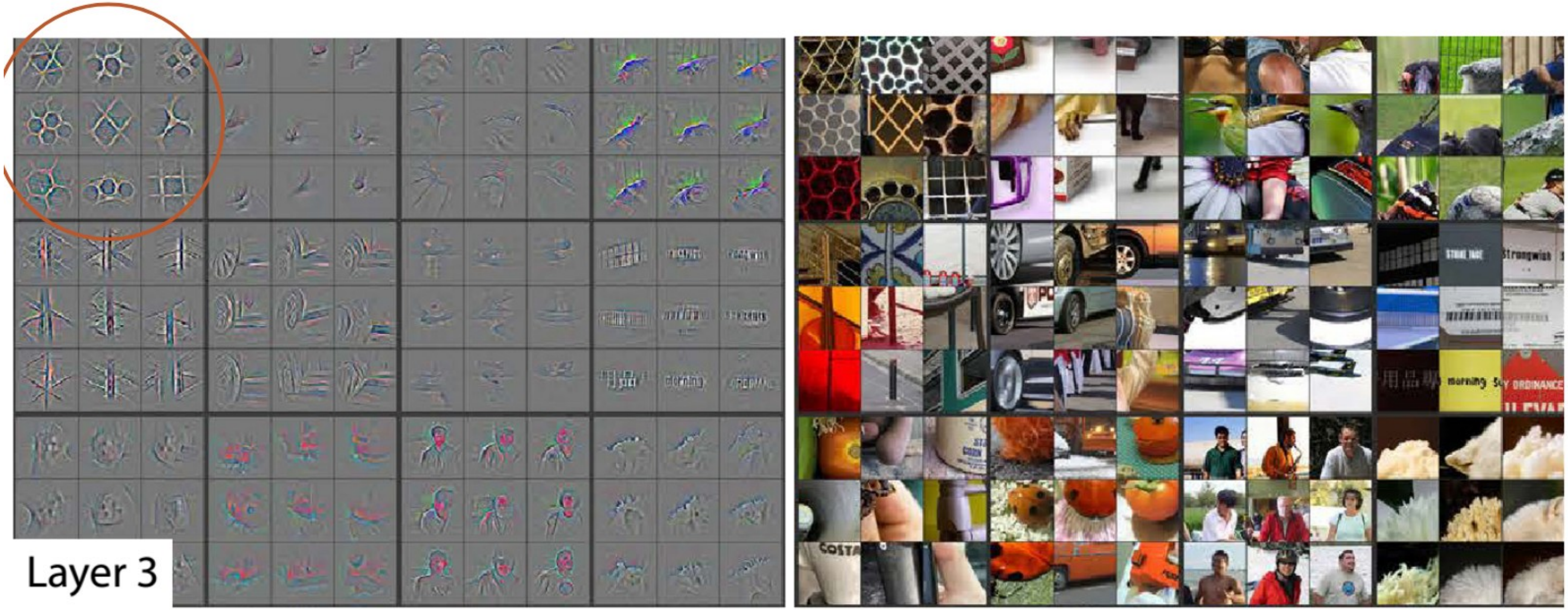
- Given a random feature map what are the top 8 activations?



End-to-end learning of feature hierarchies

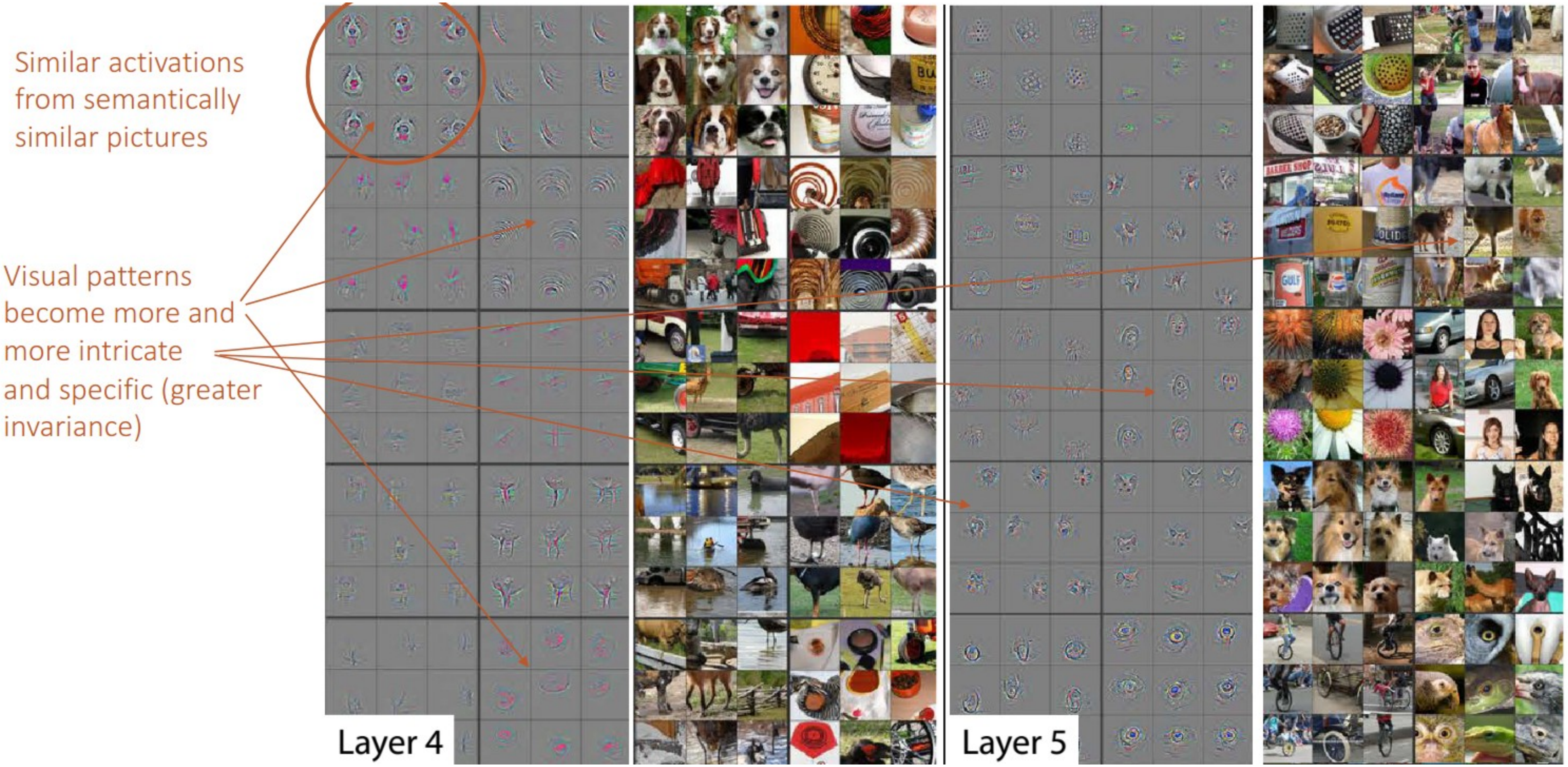
- Given a random feature map what are the top 8 activations?

Similar activations from lower level visual patterns



End-to-end learning of feature hierarchies

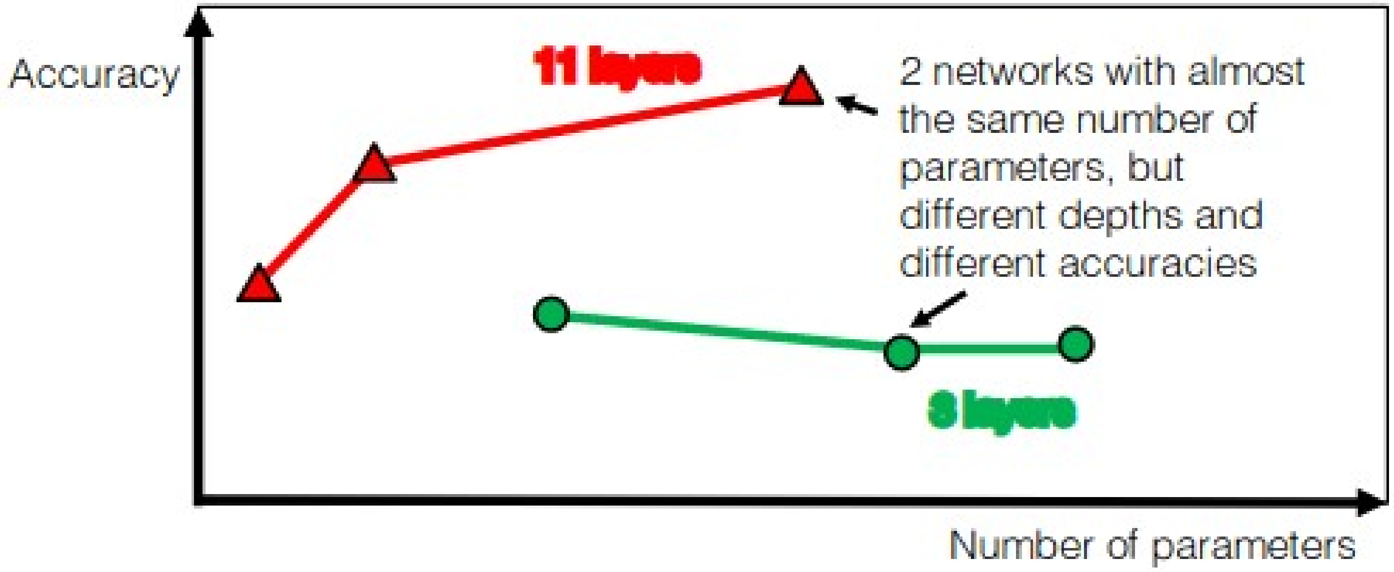
- Given a random feature map what are the top 8 activations?



So, why deep and not shallow?

- Although with two-layer (shallow) network, we can approximate all possible functions
  - Given the network layers are wide enough
- Deep architectures tend to be more efficient
  - Or otherwise, the network capacity given number of parameters is larger
- Also, deep and narrow architectures tend to generalize better than shallow and wide architectures

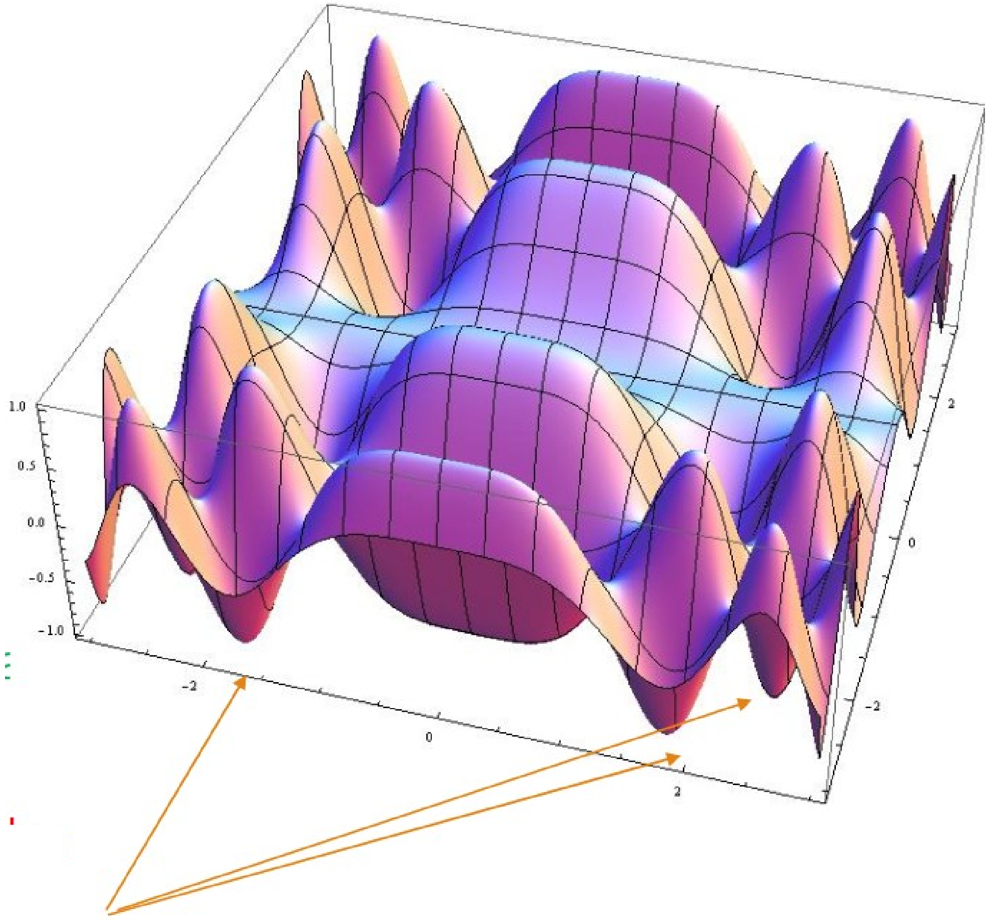
Deeper Networks Perform Better for a Given Number of Parameters



I. Goodfellow, Y. Bengio, A. Courville. *Deep Learning*. 2016.

(Non)- convexity

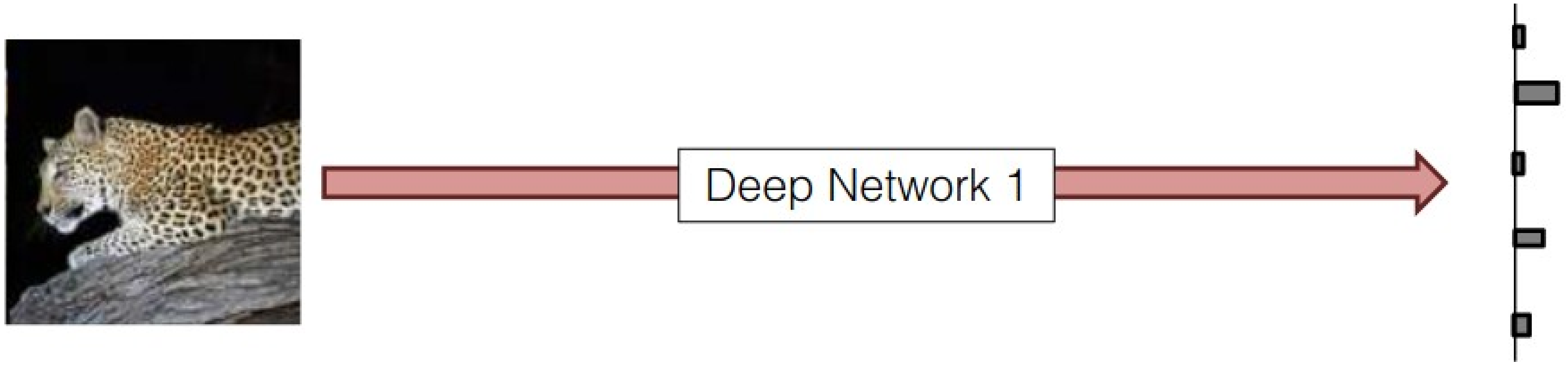
- High non-convex
  - Neural networks are stable and accurate enough though
  - So, it this more of a real problem or an interesting observation we should explain?
- Whether you have one set of parameters or another matters litter in practice
- Often ensembles of models are preferred anyway
- You cannot know if your local optimum is near the global optimum



Possible minimum

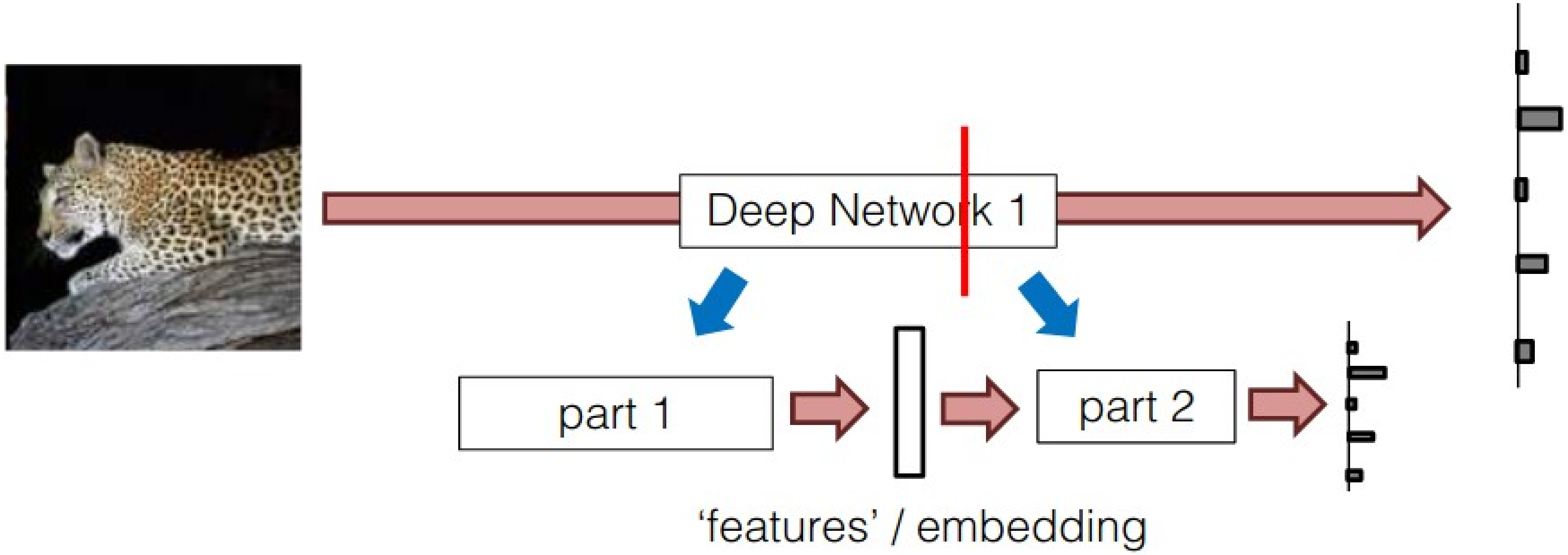
Deeper Features as Embedding

- Training a deep network on a problem where a large amount of data is available:



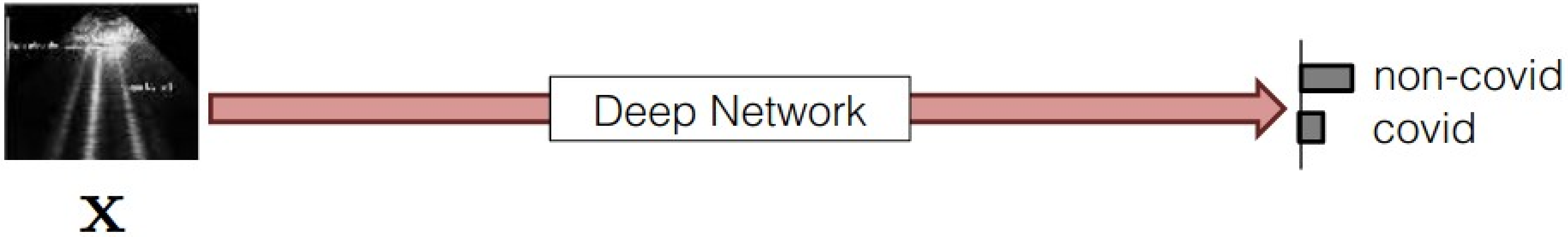
Deeper Features as Embedding

- Training a deep network on a problem where a large amount of data is available.
- Cut this network into two parts (after training):

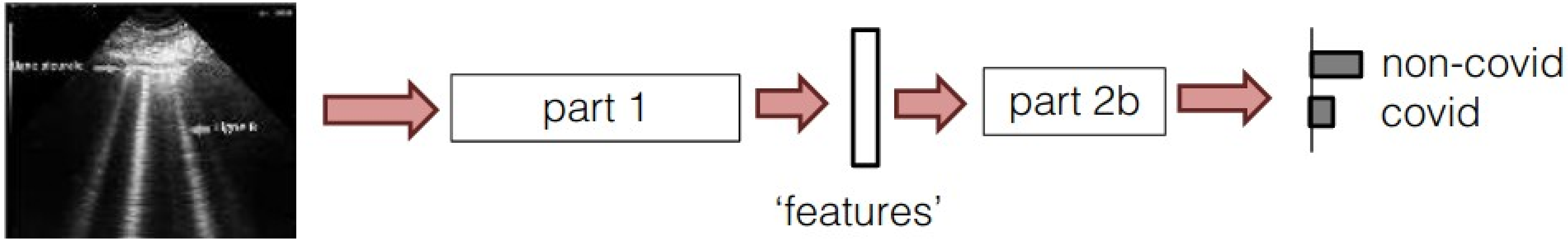


Transfer Learning/ Domain Transfer

- A simple method for transfer learning:

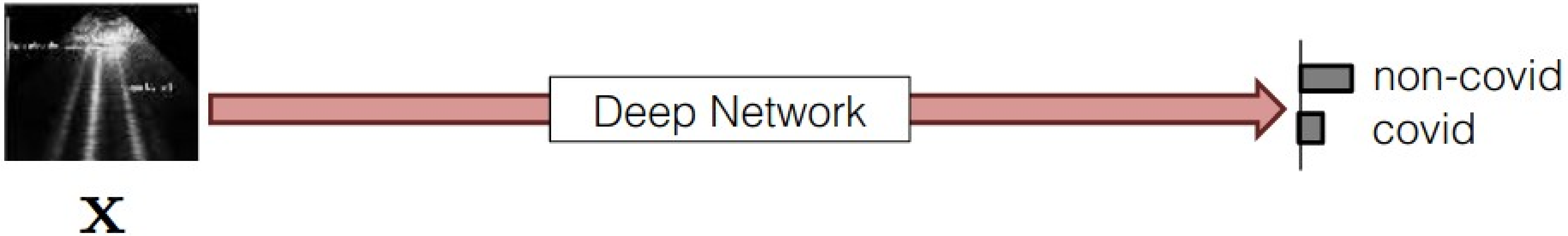


- Keep the parameters of Part 1, initialize randomly Part 2b with the new number of classes

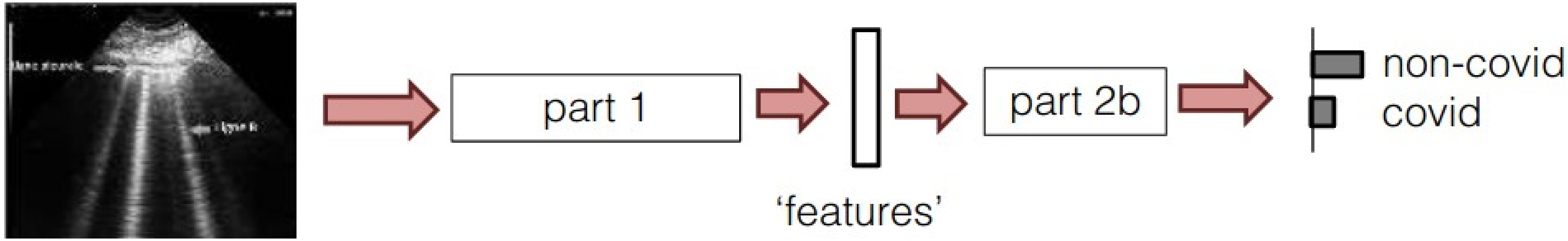


Transfer Learning/ Domain Transfer

- A simple method for transfer learning:

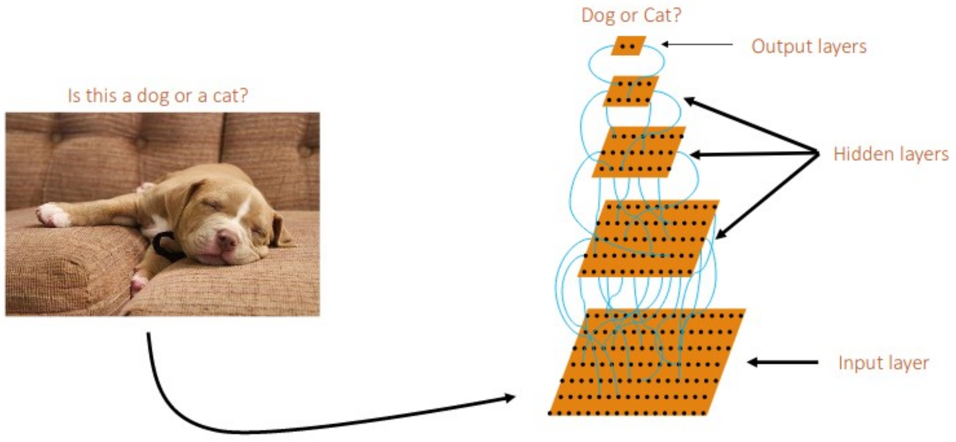


- Keep the parameters of Part 1, initialize randomly Part 2b with the new number of classes
- Alternatively, we can ‘fine-tune’ the parameters of Part 1.

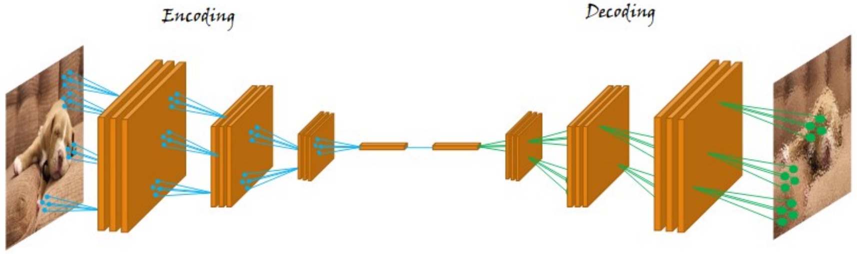


Types of learning

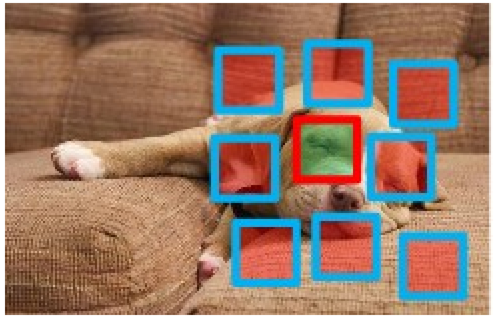
- Supervised learning, e.g. Convolutional Networks



- Unsupervised learning, e.g. Autoencoders



- Self-supervised learning
  - A mix of supervised and unsupervised learning



- Reinforcement learning
  - Agent perform actions in an environment and gets rewards

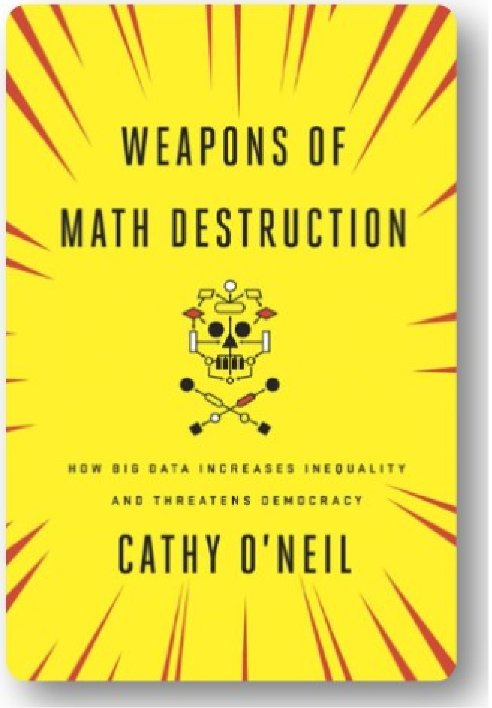
Many unanswered theoretical questions

- Theory of unsupervised learning equivalent to statistical learning theory? How to do best unsupervised learning?
- Several intractable deep network losses
- Deep structured outputs
- Combining external static knowledge (e.g. Wikipedia knowledge) with stochastic methods like neural nets
- And many more …

- Hopefully, some will answered by you in the near future ;)

AI Models in the Wild

- More and more AI models are publicly available (e.g. ChatGPT, LAMA, SAM) or they are introduced in different sections of private or public services
- However, there are a number of issues inherent issues with such models:
  - Not transparent: The input-output mapping cannot be clearly explained.
  - Huge scale: The models are used in a much larger scale than they were tested for.
  - Unmeasurable potential damages: Not quite clear hoe to quantify the effects of different aspects of the society.



Fairness in AI

nature

Subscribe Search Login

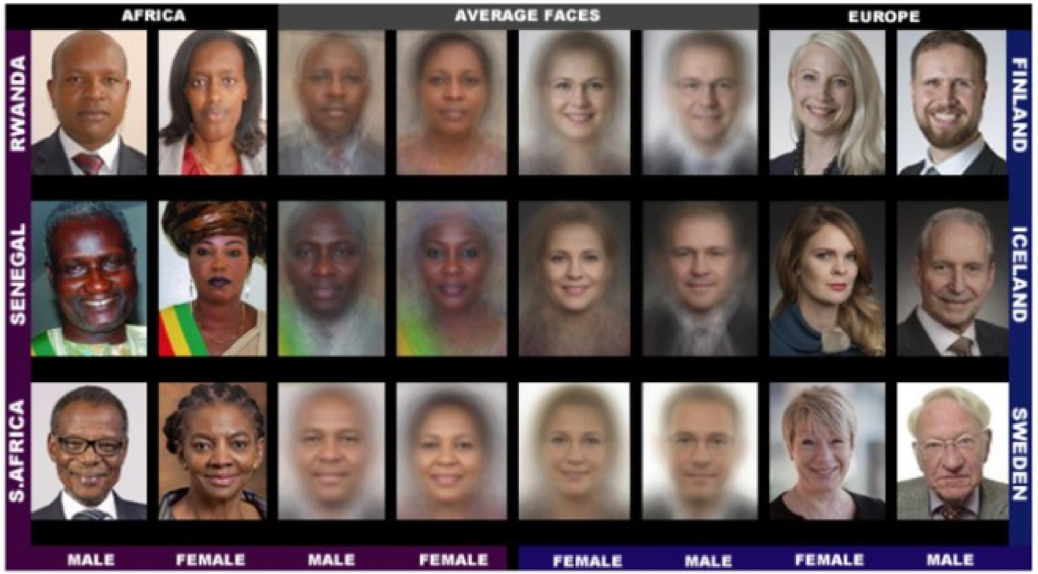
COMMENT · 18 JULY 2018

AI can be sexist and racist — it's time to make it fair

Computer scientists must identify sources of bias, de-bias training data and develop artificial-intelligence algorithms that are robust to skews in the data, argue James Zou and Londa Schiebinger.

James Zou & Londa Schiebinger

Racial bias in face recognition systems

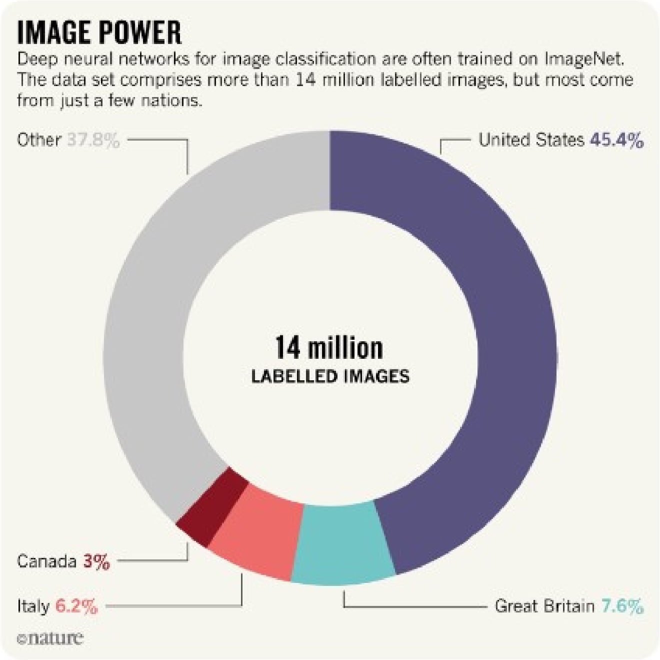


Publicly available commercial face recognition online services provided by Microsoft, Face++, and IBM respectively are found to suffer from achieving much lower accuracy on females with darker skin color(see Fig4, Buolamwini and Gebru, 2018).

Classifier	Metric	All	F	M	Darker	Lighter	DF	DM	LF	LM
MSFT	PPV(%)	93.7	89.3	97.4	87.1	99.3	79.2	94.0	98.3	**100**
	Error Rate(%)	6.3	10.7	2.6	12.9	0.7	**20.8**	6.0	1.7	0.0
	TPR (%)	93.7	96.5	91.7	87.1	99.3	92.1	83.7	**100**	98.7
	FPR (%)	6.3	8.3	3.5	12.9	0.7	**16.3**	7.9	1.3	0.0
Face++	PPV(%)	90.0	78.7	99.3	83.5	95.3	65.5	**99.3**	94.0	99.2
	Error Rate(%)	10.0	21.3	0.7	16.5	4.7	**34.5**	0.7	6.0	0.8
	TPR (%)	90.0	98.9	85.1	83.5	95.3	98.8	76.6	**98.9**	92.9
	FPR (%)	10.0	14.9	1.1	16.5	4.7	**23.4**	1.2	7.1	1.1
IBM	PPV(%)	87.9	79.7	94.4	77.6	96.8	65.3	88.0	92.9	**99.7**
	Error Rate(%)	12.1	20.3	5.6	22.4	3.2	**34.7**	12.0	7.1	0.3
	TPR (%)	87.9	92.1	85.2	77.6	96.8	82.3	74.8	**99.6**	94.8
	FPR (%)	12.1	14.8	7.9	22.4	3.2	**25.2**	17.7	5.20	0.4

Buolamwini & Gebru, 2018. Conference on Fairness, Accountability and Transparency

Lack of data diversity and underrepresentation



"Biases in the data often reflect deep and hidden imbalances in institutional infrastructures and social power relations."

Zou, James & Londa Schiebinger, 2018. "AI can be sexist and racist it's time to make it fair." Nature

Recap

- We have almost 2 month and a lot of material to cover
- After the course you should
  - Have theoretical background about the fundamentals of deep learning
  - Hands-on experience on generating your own networks
- Please try to follow the most of the lectures/ TPs and be active with questions/ recommendations.
  - Work together
- The course can give you very very important skills for your next career steps, try to take as much as you can from it