NPFL114, Lecture 01

Introduction to Deep Learning

Milan Straka

Notation

- a, $\boldsymbol{a}$, $\boldsymbol{A}$, A: scalar (integer or real), vector, matrix, tensor
- a, $\mathbf{a}$, $\mathbf{A}$: scalar, vector, matrix random variable

Notation

- a, $\boldsymbol{a}$, $\boldsymbol{A}$, A: scalar (integer or real), vector, matrix, tensor
- a, $\mathbf{a}$, $\mathbf{A}$: scalar, vector, matrix random variable
- $\frac{df}{dx}$: derivative of f with respect to x
- $\frac{\partial f}{\partial x}$: partial derivative of f with respect to x

Notation

- a, $\boldsymbol{a}$, $\boldsymbol{A}$, A: scalar (integer or real), vector, matrix, tensor
- a, $\mathbf{a}$, $\mathbf{A}$: scalar, vector, matrix random variable
- $\frac{df}{dx}$: derivative of f with respect to x
- $\frac{\partial f}{\partial x}$: partial derivative of f with respect to x
- $\nabla_{\boldsymbol{x}} f$: gradient of f with respect to $\boldsymbol{x}$, i.e., $\left(\frac{\partial f(\boldsymbol{x})}{\partial x_1}, \frac{\partial f(\boldsymbol{x})}{\partial x_2}, \ldots, \frac{\partial f(\boldsymbol{x})}{\partial x_n}\right)$

Random Variables

A random variable x is a result of a random process. It can be discrete or continuous.

Random Variables

A random variable x is a result of a random process. It can be discrete or continuous.

Probability Distribution

Probability distribution describes how likely are individual values a random variable can take.

The $\mathrm{x} \sim P$ denotes a random variable x has distribution P.

For discrete variables, probability that x takes a value x is denoted as $P(x)$ or explicitly as $P(\mathrm{x} = x)$.

For discrete variables, probability that value of x lies in the interval $[a, b]$ is given by $\int_a^b p(x)\,\mathrm{d}x$.

Random Variables

Expectation

An expectation of a function $f(x)$ with respect to discrete probability distribution $P(x)$ is defined as:

$$\mathbb{E}_{\mathrm{x}\sim P}[f(x)] \overset{\text{def}}{=} \sum_x P(x) f(x)$$

For continuous variables it is computed as:

$$\mathbb{E}_{\mathrm{x}\sim p}[f(x)] \overset{\text{def}}{=} \int_x p(x) f(x) \,\mathrm{d}x$$

Expectation is linear, i.e.,

$$\mathbb{E}_{\mathrm{x}}[\alpha f(x) + \beta g(x)] = \alpha\mathbb{E}_{\mathrm{x}}[f(x)] + \beta\mathbb{E}_{\mathrm{x}}[g(x)]$$

Random Variables

Variance

Variance measures how much the values of a random variable differ from the expectation.

$$\operatorname{Var}(f(x)) \overset{\text{def}}{=} \mathbb{E}\left[f(x) - \mathbb{E}[f(x)]^2\right]$$

Common Probability Distributions

Bernoulli Distribution

Bernoulli distribution is a distribution over a binary random variable. It has one parameter $\varphi \in [0, 1]$, which specifies the probability of the random variable being equal to 1.

Common Probability Distributions

Bernoulli Distribution

Bernoulli distribution is a distribution over a binary random variable. It has one parameter $\varphi \in [0, 1]$, which specifies the probability of the random variable being equal to 1.

Categorical Distribution

Extension of Bernoulli distribution to random variables taking k different variables. It is parametrized by a $p \in [0, 1]^k$, such that $\sum p(i) = 1$.

Information Theory

Self Information

Amount of when a random variable is sampled.

- Should be zero for events with probability 1.
- Less likely events are more surprising.
- Independent events should have additive information.

$$I(x) \stackrel{\text{def}}{=} -\log P(x) = \log \frac{1}{P(x)}$$

Information Theory

Self Information

Amount of when a random variable is sampled.

- Should be zero for events with probability 1.
- Less likely events are more surprising.
- Independent events should have additive information.

$$I(x) \stackrel{\text{def}}{=} -\log P(x) = \log \frac{1}{P(x)}$$

Entropy

Amount of in the whole distribution.

$$H(P) \stackrel{\text{def}}{=} \mathbb{E}_{\mathrm{x}\sim P}[I(x)] = -\mathbb{E}_{\mathrm{x}\sim P}[\log P(x)]$$

- for discrete P: $H(P) = -\sum_x P(x) \log P(x)$
- for continuous P: $H(P) = -\int P(x) \log P(x) \,\mathrm{d}x$

Information Theory

Cross-Entropy

$$H(P,Q) \overset{\text{def}}{=} -\mathbb{E}_{\mathrm{x}\sim P}[\log Q(x)]$$

- Gibbs inequality
  - $H(P,Q) \geq H(P)$
  - $H(P) = H(P,Q) \Leftrightarrow P = Q$
- generally $H(P,Q) \neq H(Q,P)$

Information Theory

Cross-Entropy

$$H(P, Q) \overset{\text{def}}{=} -\mathbb{E}_{\mathrm{x} \sim P}[\log Q(x)]$$

- Gibbs inequality
  - $H(P, Q) \geq H(P)$
  - $H(P) = H(P, Q) \Leftrightarrow P = Q$
- generally $H(P, Q) \neq H(Q, P)$

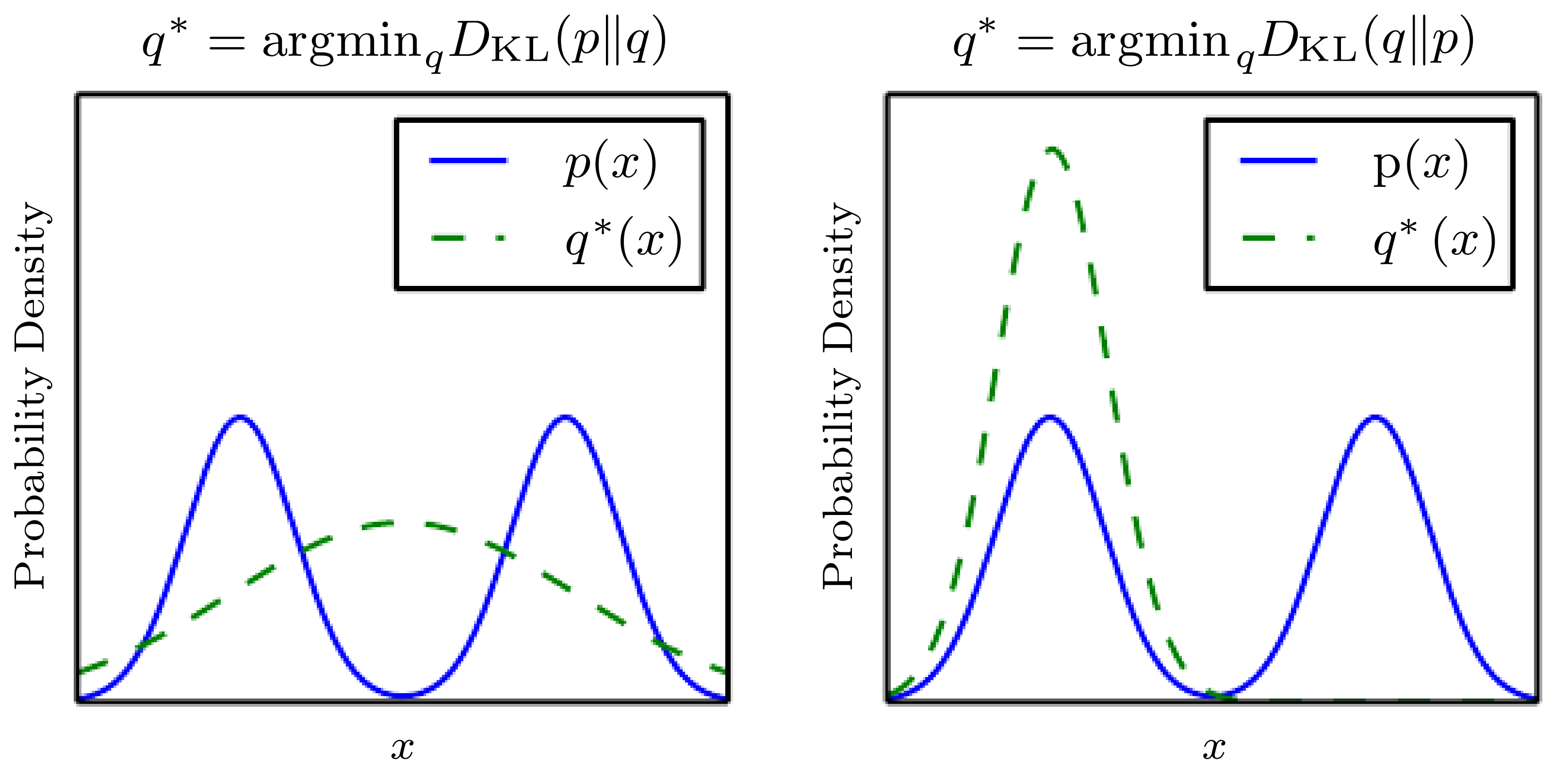
Kullback-Leibler Divergence (KL Divergence)

Sometimes also called .

$$D_{\mathrm{KL}}(P||Q) \overset{\text{def}}{=} H(P, Q) - H(P) = \mathbb{E}_{\mathrm{x} \sim P}[\log P(x) - \log Q(x)]$$

- consequence of Gibbs inequality: $D_{\mathrm{KL}}(P||Q) \geq 0$
- generally $D_{\mathrm{KL}}(P||Q) \neq D_{\mathrm{KL}}(Q||P)$

Nonsymmetry of KL Divergence



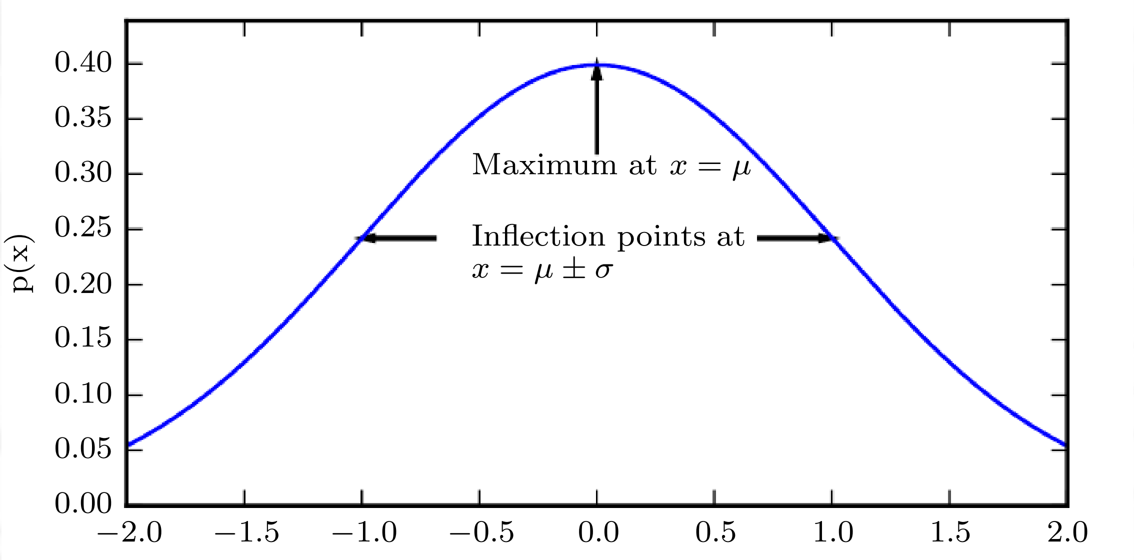
Common Probability Distributions

Normal (or Gaussian) Distribution

Distribution over real numbers, parametrized by a mean μ and variance σ^2:

$$\mathcal{N}(x; \mu, \sigma^2) = \sqrt{\frac{1}{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

For standard values $\mu = 0$ and $\sigma^2 = 1$ we get $\mathcal{N}(x; 0, 1) = \sqrt{\frac{1}{2\pi}} e^{-\frac{x^2}{2}}$.



Why Normal Distribution

Central Limit Theorem

A sum of independent identically distributed random variables with a limited variance converges to normal distribution.

Why Normal Distribution

Central Limit Theorem

A sum of independent identically distributed random variables with a limited variance converges to normal distribution.

Distribution with Maximal Entropy

Consider distributions with a given mean and variance. It can be proven (using variational inference) that such distribution with   is exactly the normal distribution.

A distribution with maximal entropy can be considered the most general one, containing as little additional assumptions as possible.

Machine Learning

A possible definition of learning from Mitchell (1997):

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

Machine Learning

A possible definition of learning from Mitchell (1997):

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

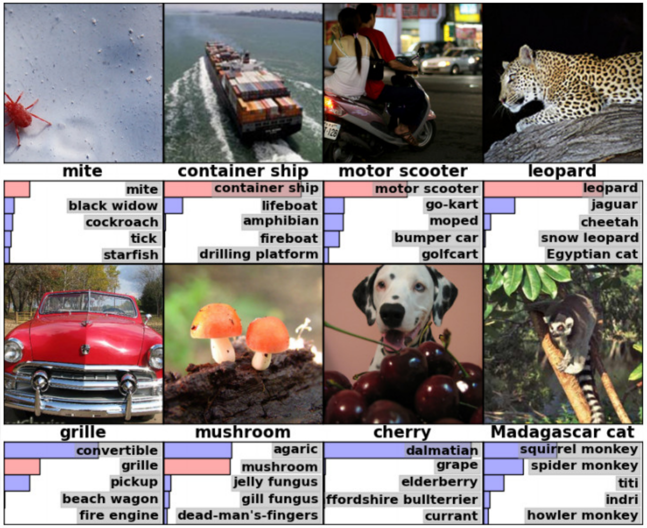
- Task T
  - : assigning one of k categories to a given input
  - : producing a number $x \in \mathbb{R}$ for a given input
  - , , , …
- Experience E
  - : usually a dataset with desired outcomes (or)
  - : usually data without any annotation (raw text, raw images, …)
  - , , …
- Measure P
  - , , , …

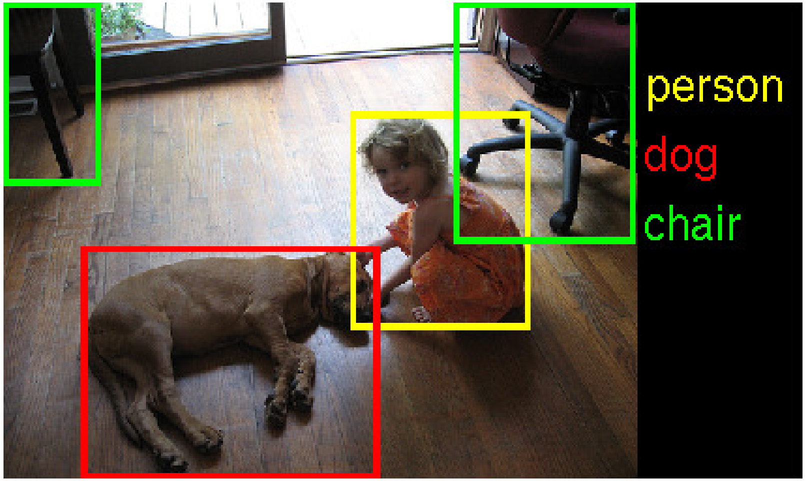
Well-known Datasets

MNIST	Images (28x28, grayscale) of handwritten digits.	60k
CIFAR-10	Images (32x32, color) of 10 classes of objects.	50k
CIFAR-100	Images (32x32, color) of 100 classes of objects (with 20 defined superclasses).	50k
ImageNet	Labeled object image database (labeled objects, some with bounding boxes).	14.2M
ImageNet-ILSVRC	Subset of ImageNet for Large Scale Visual Recognition Challenge, annotated with 1000 object classes and their bounding boxes.	1.2M
COCO	: Complex everyday scenes with descriptions (5) and highlighting of objects (91 types).	2.5M

Well-known Datasets

ImageNet-ILSVRC



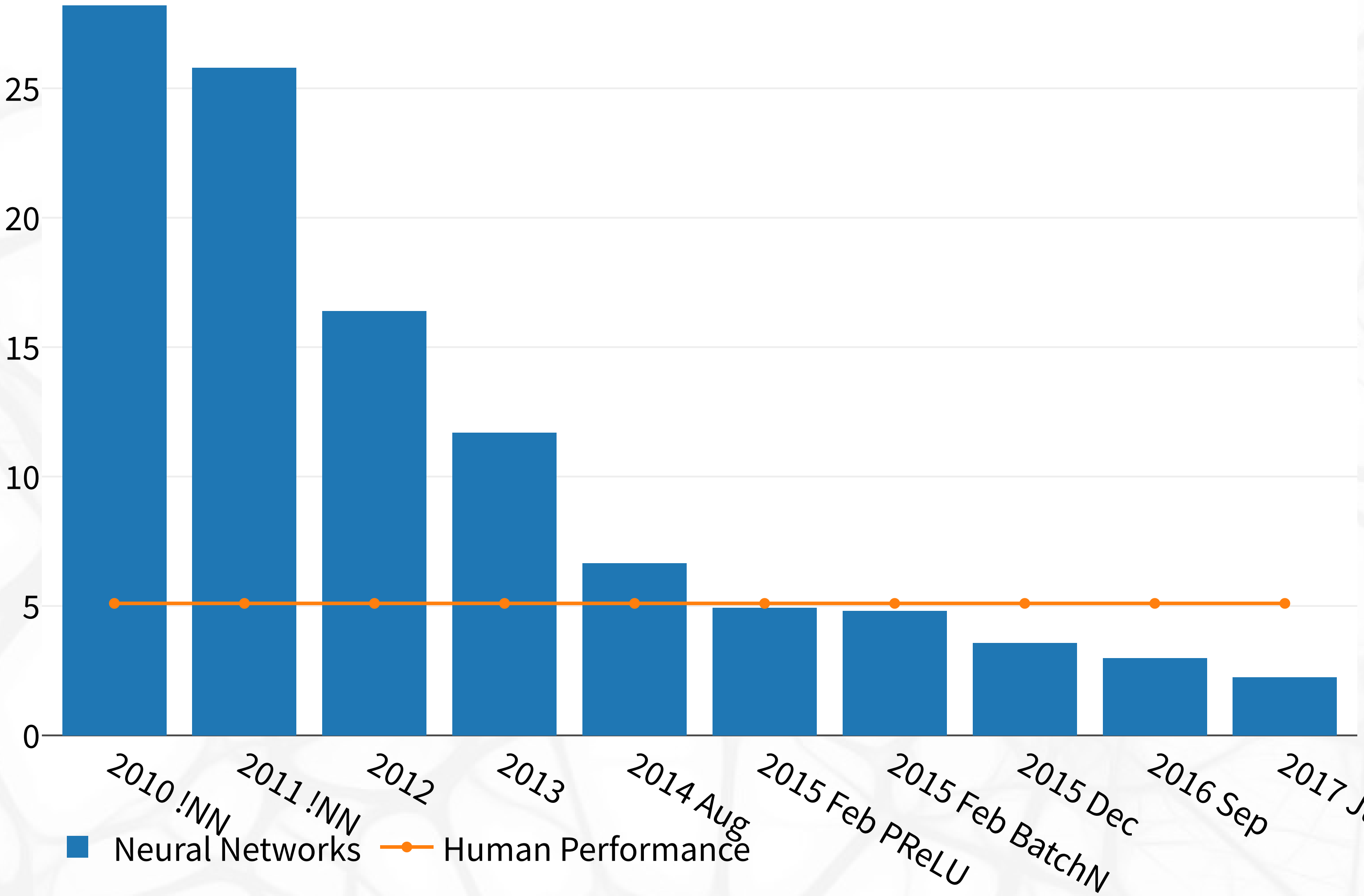


COCO

Well-known Datasets

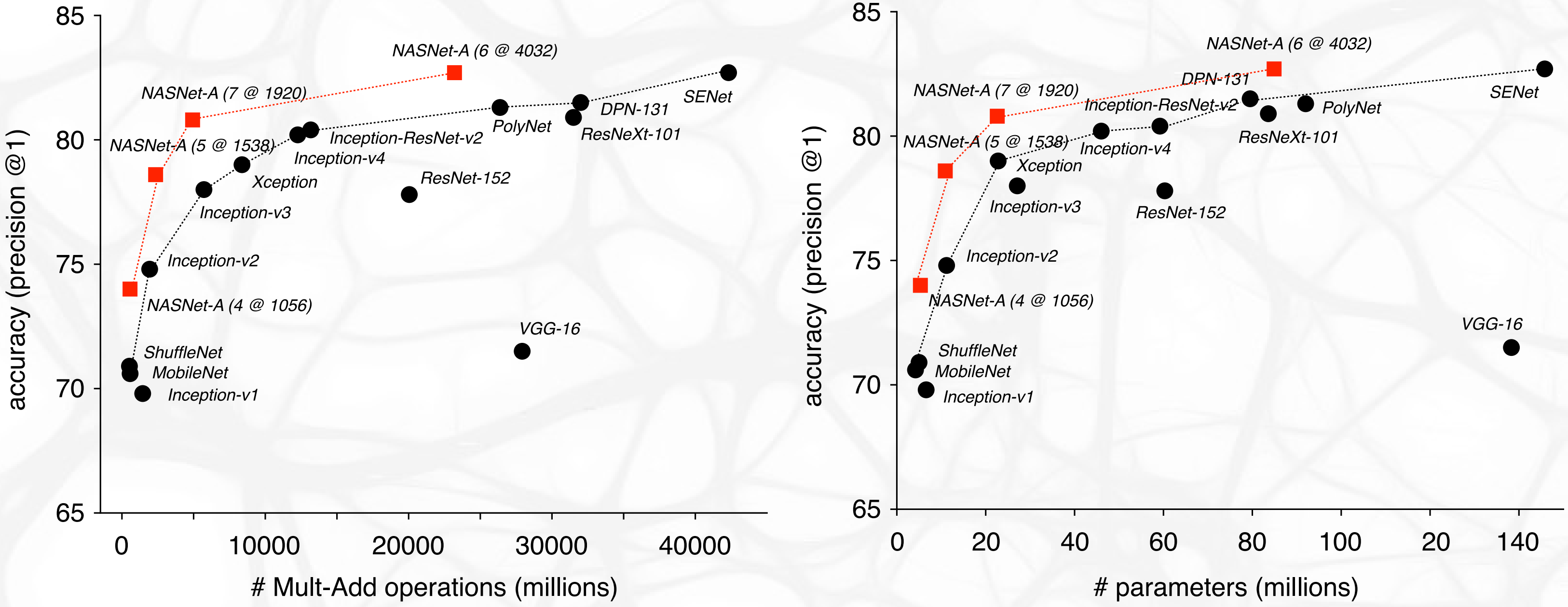
IAM-OnDB	Pen tip movements of handwritten English from 221 writers.	86k words
TIMIT	Recordings of 630 speakers (10 sentences each) of 8 major dialects of American English.	6.3k sentences
CommonVoice	Nearly 400,000 recordings from 20,000 different people, around 500 hours of speech.	400k
PTB	: 2500 stories from Wall Street Journal, with POS tags and parsed into trees.	1M words
PDT	: Czech sentences annotated on 4 layers (word, morphological, analytical, tectogrammatical).	1.9M words
UD	: Treebanks of 60 languages with consistent annotation of lemmas, POS tags, morphology and syntax.	102 treebanks
WMT	Aligned parallel sentences for machine translation.	gigawords

ILSVRC Image Recognition Accuracies

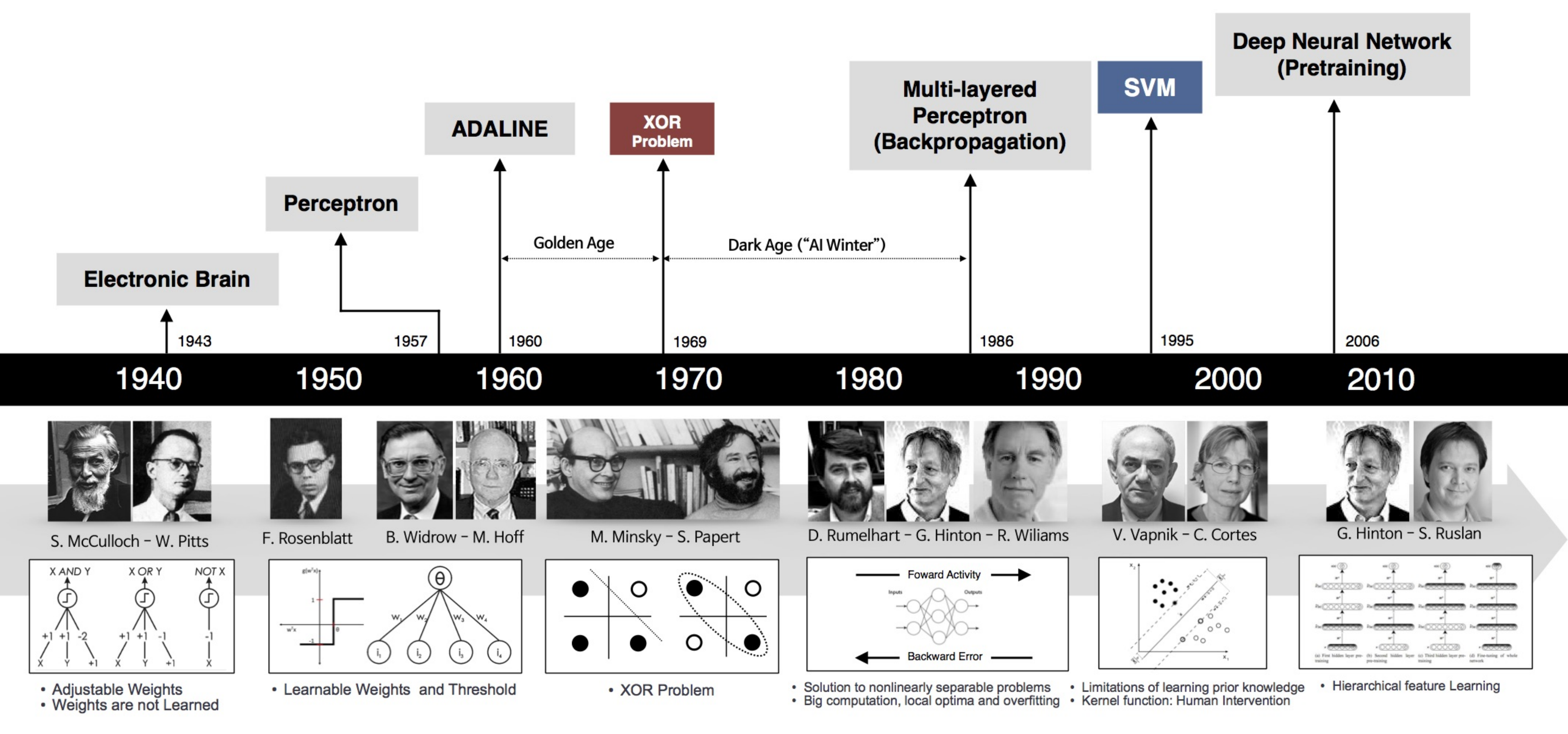


ILSVRC Image Recognition Accuracies

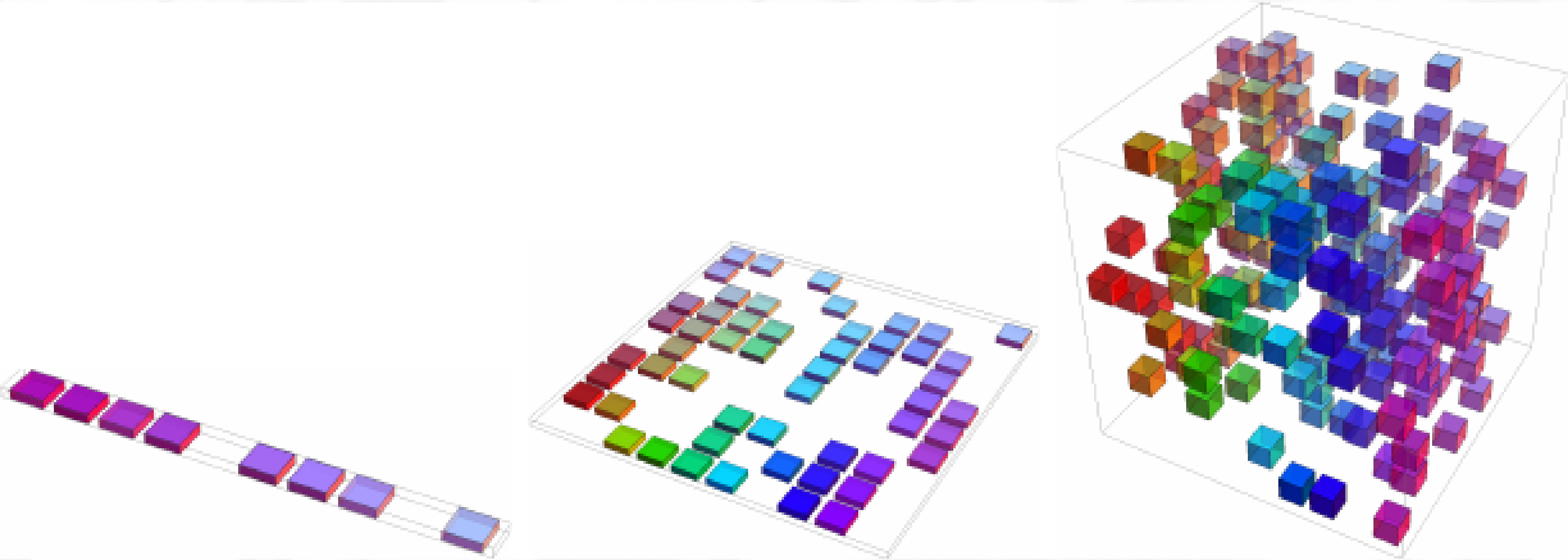
Recently (summer 2017), a paper came out describing automatic generation of neural architectures using reinforcement learning.



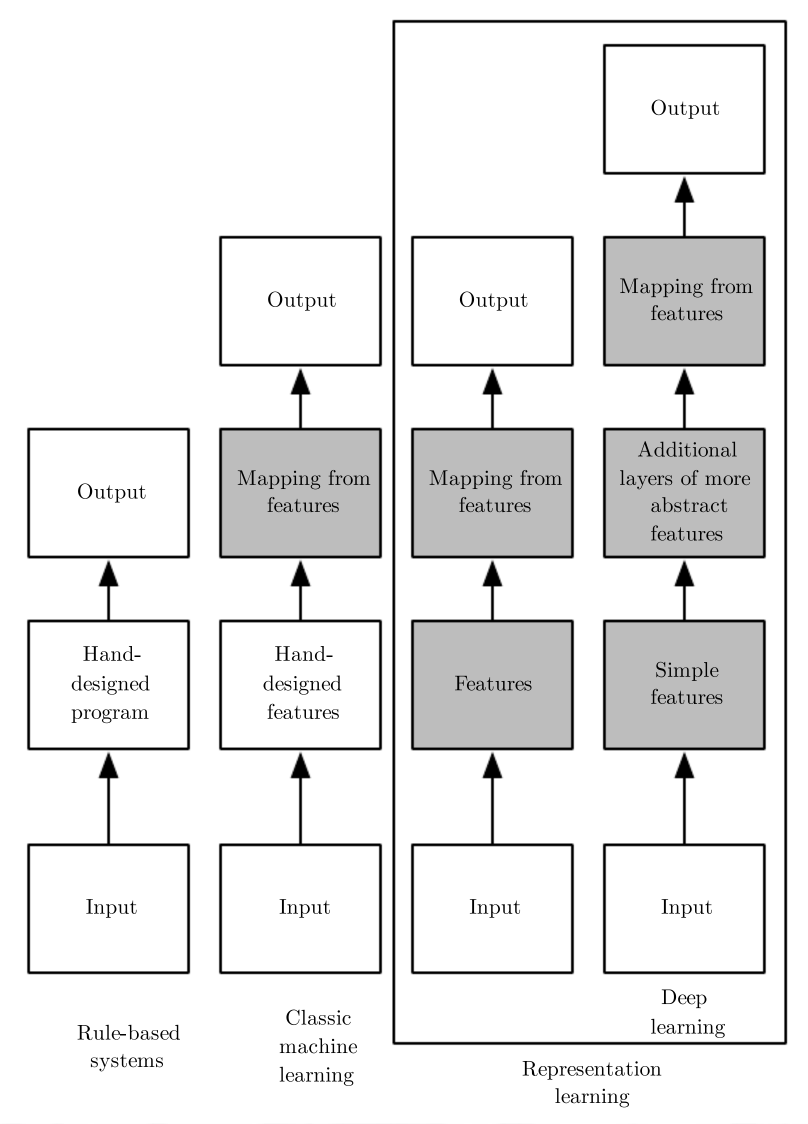
Introduction to Machine Learning History


Electronic Brain
Perceptron
ADALINE
XOR Problem
Multi-layered Perceptron (Backpropagation)
SVM
Deep Neural Network (Pretraining)
Golden Age
Dark Age ("AI Winter")
1943
1957
1960
1969
1986
1995
2006
1940
1950
1960
1970
1980
1990
2000
2010
S. McCulloch – W. Pitts
F. Rosenblatt
B. Widrow – M. Hoff
M. Minsky – S. Papert
D. Rumelhart – G. Hinton – R. Wiliams
V. Vapnik – C. Cortes
G. Hinton – S. Ruslan
X AND Y
X OR Y
NOT X
Foward Activity
Inputs
Outputs
Backward Error
• Adjustable Weights
• Weights are not Learned
• Learnable Weights and Threshold
• XOR Problem
• Solution to nonlinearly separable problems
• Big computation, local optima and overfitting
• Limitations of learning prior knowledge
• Kernel function: Human Intervention
• Hierarchical feature Learning

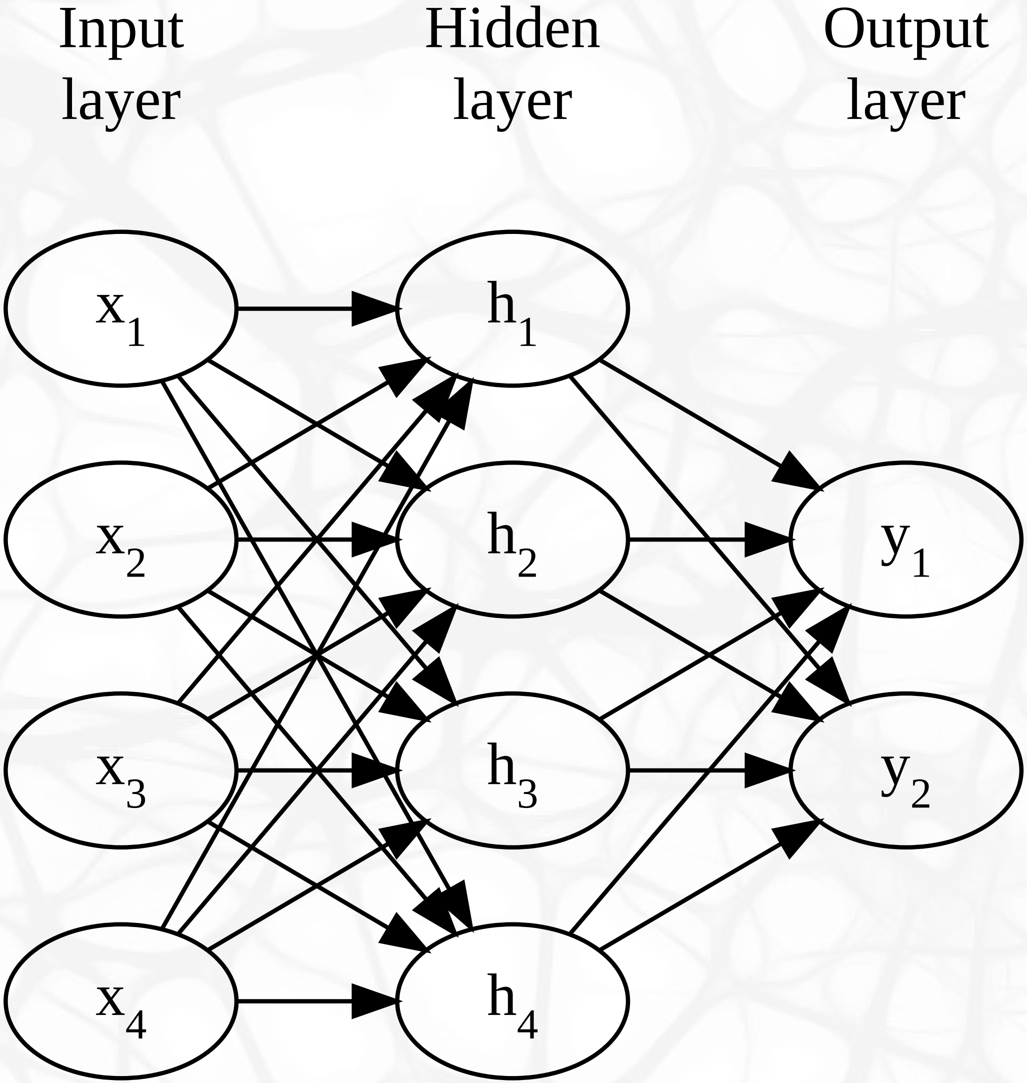
Curse of Dimensionality



Machine and Representation Learning



Neural Network Architecture à la '80s



Neural Network Architecture

There is a weight on each edge, and an activation function f is performed on the hidden layers, and optionally also on output layer.

$$h_i = f\left(\sum_j w_{i,j} x_j\right)$$

If the network is composed of layers, we can use matrix notation and write:

$$\boldsymbol{h} = f\left(\boldsymbol{W}\boldsymbol{x}\right)$$

Neural Network Activation Functions

Output Layers

- none (linear regression if there are no hidden layers)
- σ (sigmoid; logistic regression if there are no hidden layers)

$$\sigma(x) \stackrel{\text{def}}{=} \frac{1}{1+e^{-x}}$$

- $\operatorname{softmax}$ (maximum entropy markov model if there are no hidden layers)

$$\operatorname{softmax}(\boldsymbol{x}) \propto e^{\boldsymbol{x}}$$

$$\operatorname{softmax}(\boldsymbol{x})_i \stackrel{\text{def}}{=} \frac{e^{x_i}}{\sum_j e^{x_j}}$$

Neural Network Activation Functions

Hidden Layers

- none (does not help, composition of linear mapping is a linear mapping)
- σ (but works badly – nonsymmetrical, $\frac{d\sigma}{dx}(0) = 1/4$)
- $\tanh$
  - result of making σ symmetrical and making derivation in zero 1
  - $\tanh(x) = 2\sigma(2x) - 1$
- ReLU
  - $\max(0, x)$

Universal Approximation Theorem '89

Let $\varphi(x)$ be nonconstant, bounded and monotonically increasing continuous function.

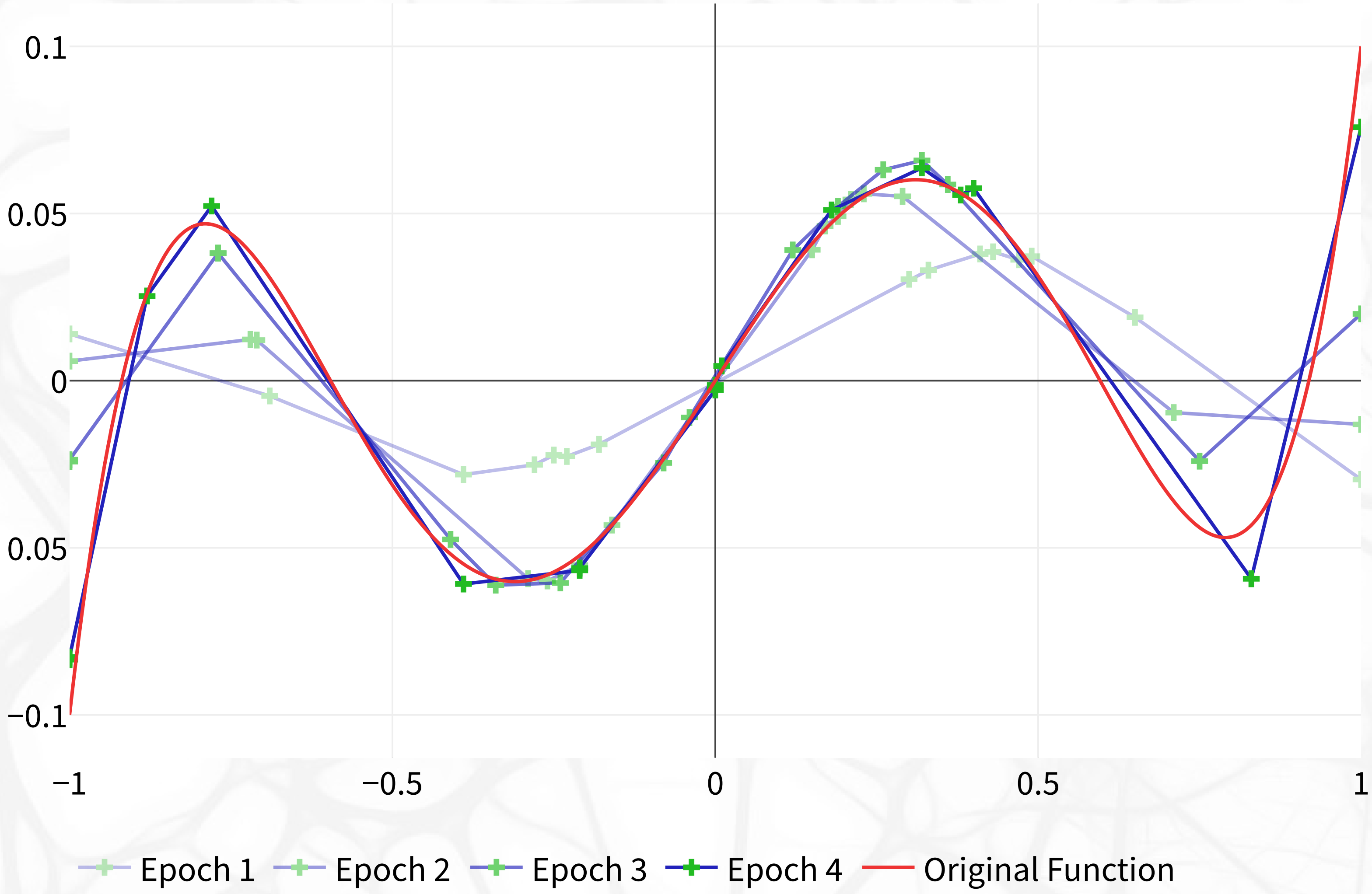
Then for any $\varepsilon > 0$ and any continuous function f on $[0,1]^m$ there exists an $N \in \mathbb{N}$, $v_i \in \mathbb{R}$, $b_i \in \mathbb{R}$ and $\boldsymbol{w_i} \in \mathbb{R}^m$, such that if we denote

$$F(\boldsymbol{x}) = \sum_{i=1}^{N} v_i \varphi(\boldsymbol{w_i} \cdot \boldsymbol{x} + b_i)$$

then for all $x \in [0,1]^m$

$$|F(\boldsymbol{x}) - f(\boldsymbol{x})| < \varepsilon.$$

Evolving ReLU Approximation


Epoch 1
Epoch 2
Epoch 3
Epoch 4
Original Function