Deep Learning for UvA Data Scientists

Data Science Centre
11th March 2022

UNIVERSITEIT VAN AMSTERDAM

A bit about me

- *“Thiv”*
- PhD students @ UvA
- Background: Chemistry
- Research Interest: Neurosymbolic Artificial Intelligence

Thiviyan Thanapalasingam
t.singam@uva.nl
https://twitter.com/ThiviyanSingam
INDE lab
INtelligent Data Engineering

Credits for slides:
Dr. Peter Bloem
https://twitter.com/pbloemesquire
https://mlvu.github.io/

Before we start…

I assume you know:

- Python Programming
- Data Science Toolkits
- Basic machine learning (e.g. training/validation/test data)
- Basic probability
- Some linear algebra terms (e.g. vectors, matrices, tensors)

Workshop Outline - Part 1

11:10 am

Introduction to Deep Learning

- Fundamental concepts
- Applications of Deep Learning

Using Deep Learning Libraries

- Working with PyTorch to build Deep Learning models
- Essential concepts
- Using Google Colab to build and train models

11:30 am

Interactive Session #1 (20 mins)

Workshop Outline - Part 2

11:50 am

Training Deep Learning Models

- Practical tips for building and testing models
- Hyperparameter Tuning

Overview of common Deep Learning Architectures

- Convolutional Neural Networks, Recurrent Networks, Graph Neural Networks, Generative Models

12:10 pm

Interactive Session #2 (20 mins)

Workshop Outline - Part 3

12:30 pm

Current State of Deep Learning

- Limitations of Deep Learning

12:50 pm

Q&A Session

Introduction to Deep Learning

What is Deep Learning?

What is Deep Learning?

trad. Machine Learning

raw data

feature extraction (hardcoded)

↓

features

learning

↓

output

specific learning method for each model

classification, regression, clustering

What is Deep Learning?

trad. Machine Learning

raw data

↓ feature extraction (hardcoded)

features

↓ learning

output

specific learning method for each model

classification, regression, clustering

Deep Learning

raw data

↓

tensors

↓ learning

hidden layers

↓ learning

output

always uses (some form of) gradient descent

much more flexible, less limited to abstract tasks

End-to-end Learning


The New York Times.

ARMISTICE SIGNED, END OF THE WAR!
BERLIN SEIZED BY REVOLUTIONISTS;
NEW CHANCELLOR BEGS FOR ORDER;
OUSTED KAISER FLEES TO HOLLAND

$\xrightarrow{\text{OCR}}$ $\xrightarrow{\text{tokenization}}$ $\xrightarrow{\text{named entity recognition}}$ $\xrightarrow{\text{relation extraction}}$

“When did Obama become president?”

End-to-end Learning

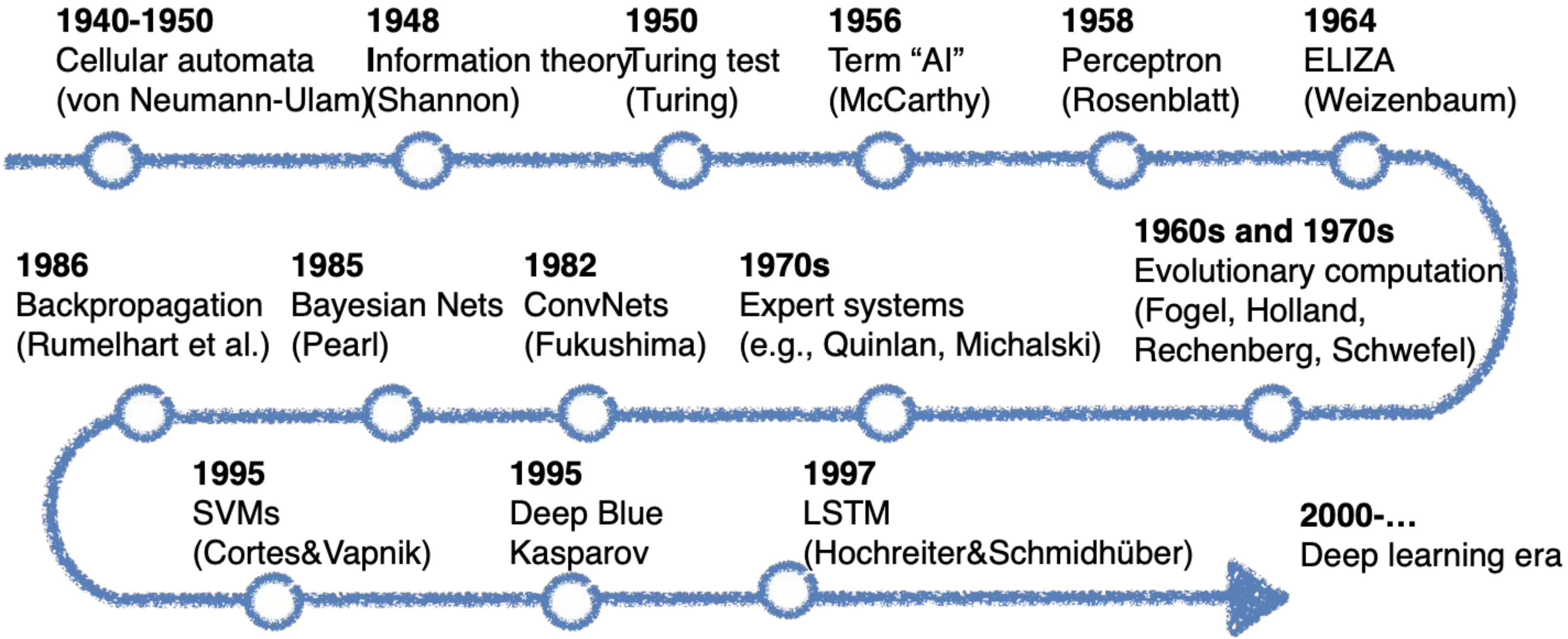

The New York Times.
ARMISTICE SIGNED, END OF THE WAR!
BERLIN SEIZED BY REVOLUTIONISTS;
NEW CHANCELLOR BEGS FOR ORDER;
OUSTED KAISER FLEES TO HOLLAND

OCR → tokenization → named entity recognition → relation extraction → "When did Obama become president?"

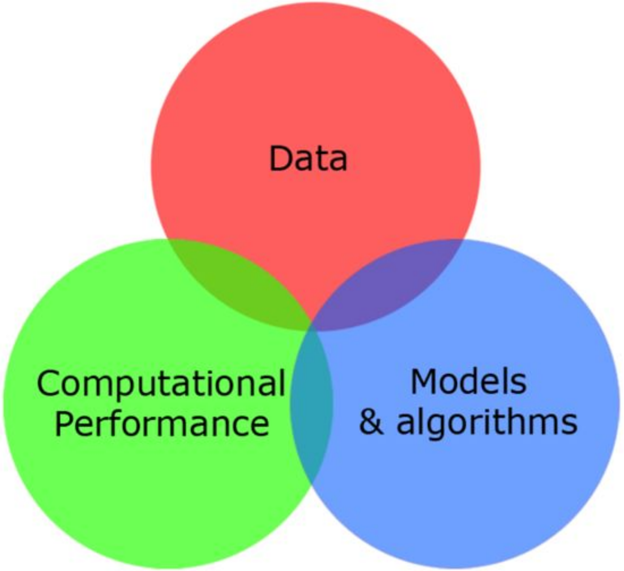
← ← ← ←

error signal

Historical Perspective



Why is Deep Learning Successful?



Accessible hardware

Powerful hardware

Intuitive programming languages

Specialized packages

Components of AI Systems

Knowledge representation

How to represent & process data?

Knowledge acquisition (learning objective & algorithms)

How to extract knowledge?

Learning problems

What kind of problems can we formulate?

Learning as Optimisation

For given **data**, find the **best data representation** from a given **class of representations** that minimizes given **learning objective (loss)**.

$$\min_{x \in \mathbb{X}} f(x; \mathscr{D})$$

$$\text{s.t. } x \in \mathbb{Y} \subseteq \mathbb{X}$$

Optimization algorithm = learning algorithm.

Learning Tasks

Learning Tasks

Supervised Learning

- We distinguish **inputs** and **outputs**.
- We are interested in **prediction**.
- We minimize a **prediction error.**

Learning Tasks

Supervised Learning

- We distinguish **inputs** and **outputs**.
- We are interested in **prediction**.
- We minimize a **prediction error.**

Unsupervised learning

- **No** distinction among variables.
- We look for a **data structure**.
- We minimize a **reconstruction error**, **compression rate**, …

Learning Tasks

Supervised Learning

- We distinguish **inputs** and **outputs**.
- We are interested in **prediction**.
- We minimize a **prediction error.**

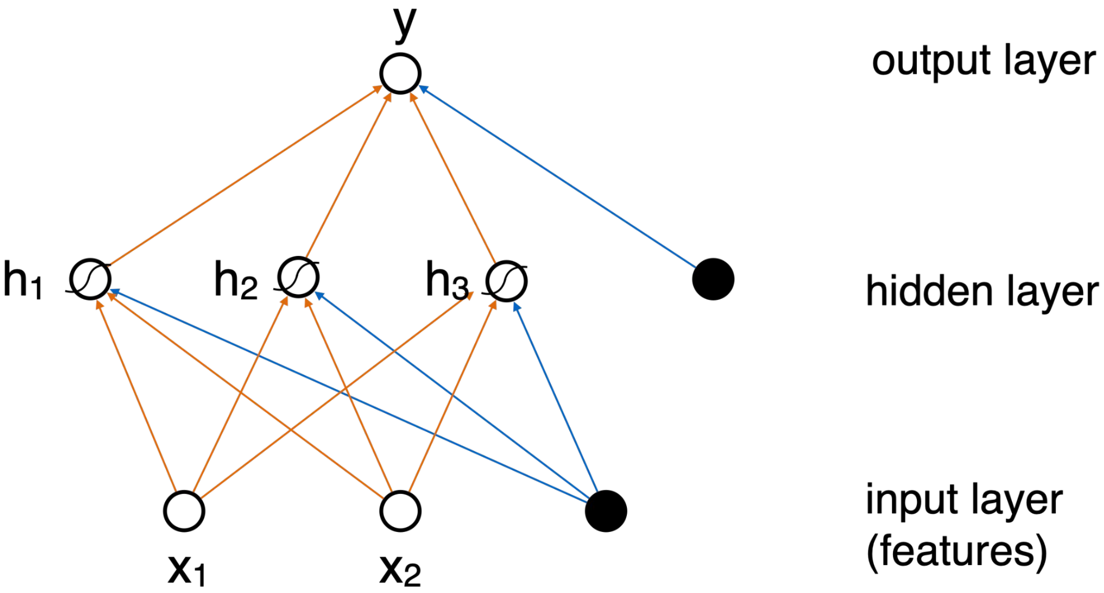
Unsupervised learning

- **No** distinction among variables.
- We look for a **data structure**.
- We minimize a **reconstruction error**, **compression rate**, …

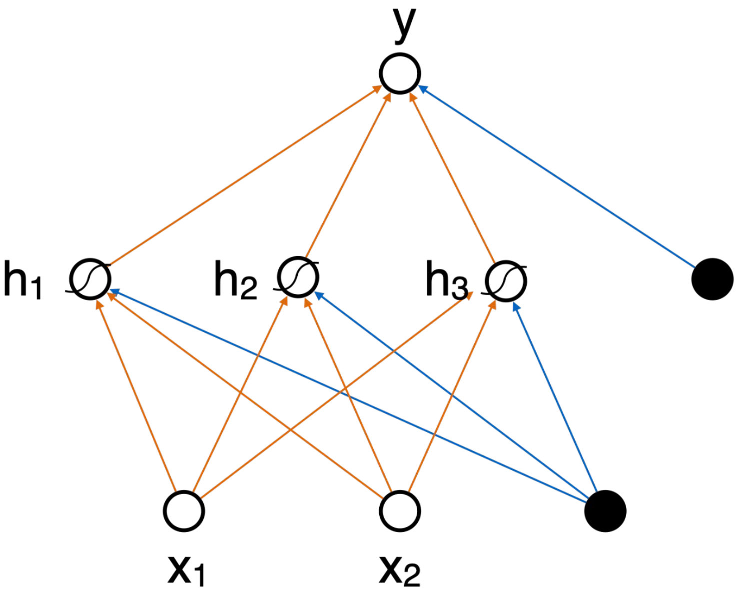
Reinforcement learning

- An **agent** interacts with an **environment**.
- We want to learn a **policy**.
- Each **action** is **rewarded.**
- We maximize a **total reward**.

Neural Networks



Neural Networks

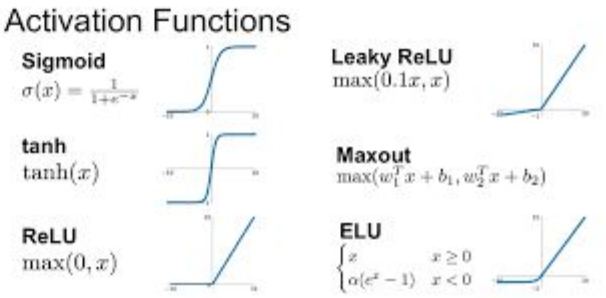


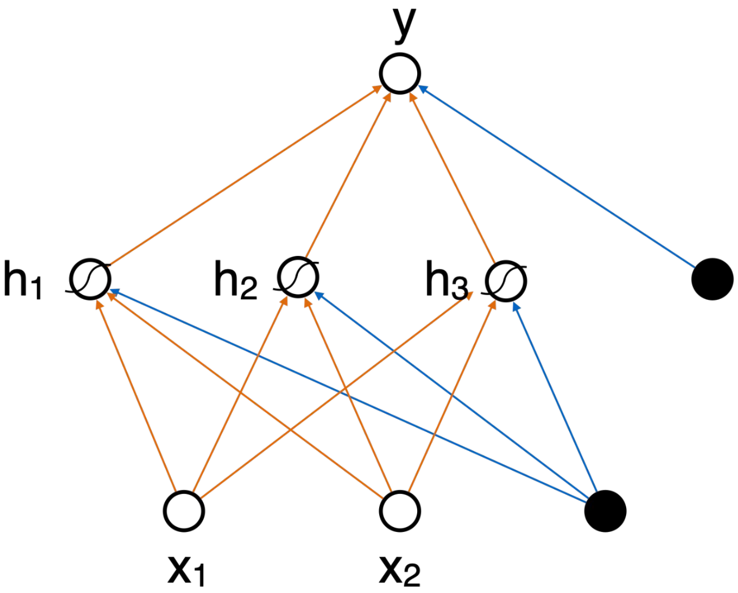
output layer

hidden layer

input layer
(features)

Neural Networks



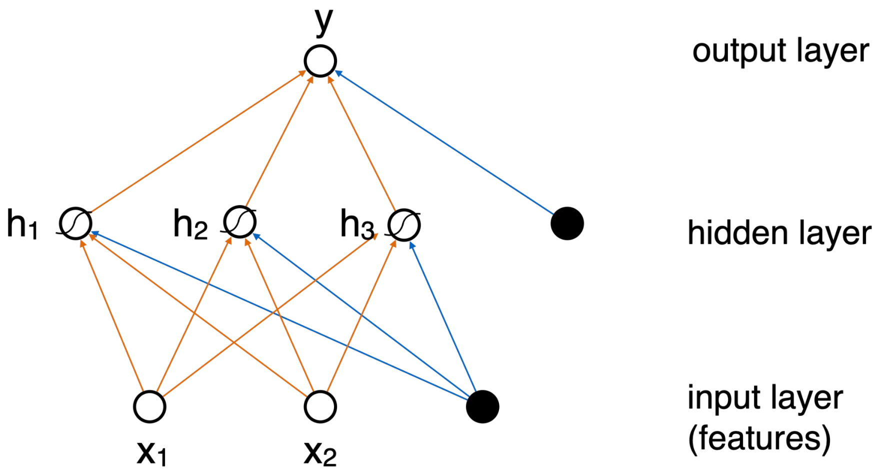


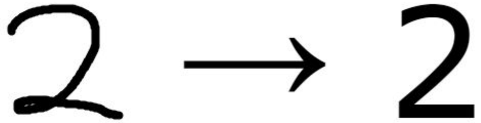
output layer

hidden layer

input layer
(features)

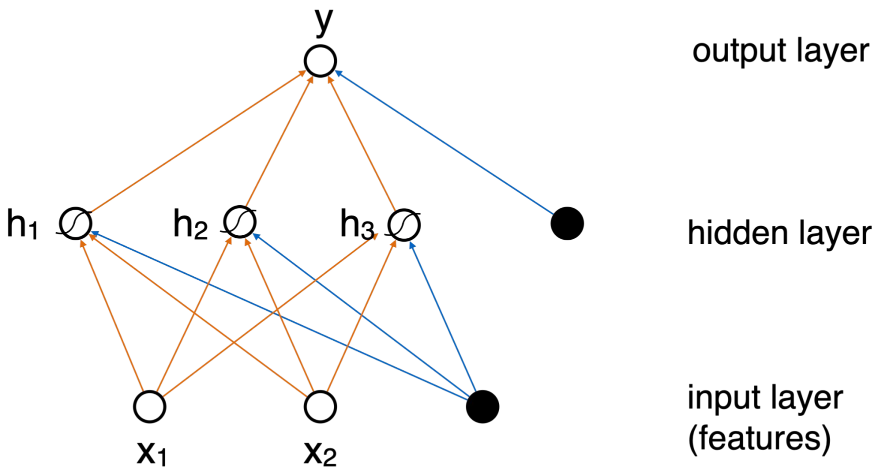
Neural Networks: Function approximation

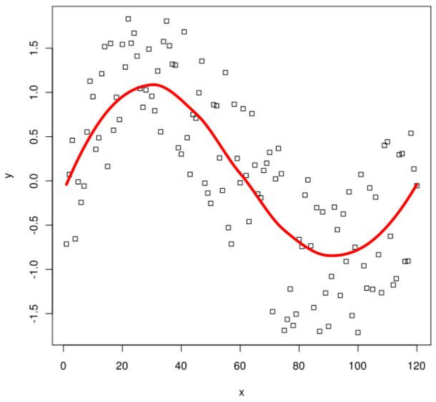




Assumption: There is a mathematical function for the task you are trying to learn
Goal: To learn how to approximate the function

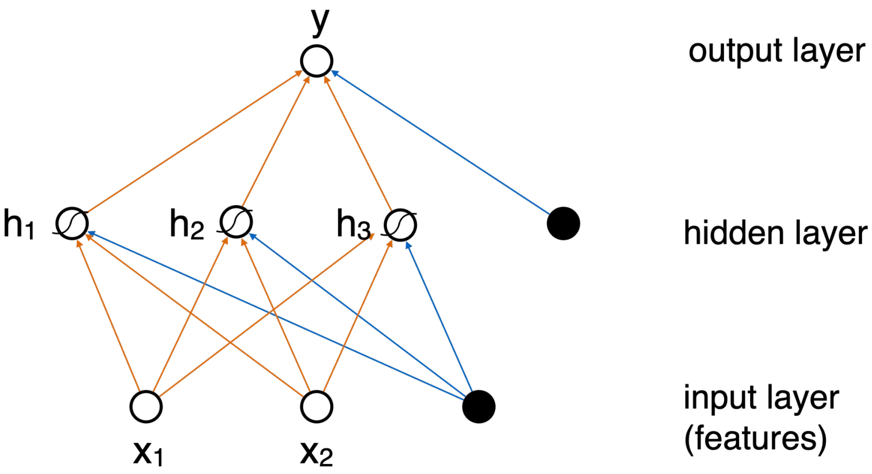
Neural Networks: Function approximation

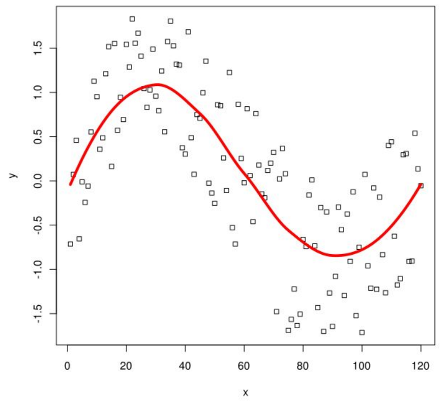




<u>Assumption:</u> There is a mathematical function for the task you are trying to learn
<u>Goal:</u> To learn how to approximate the function

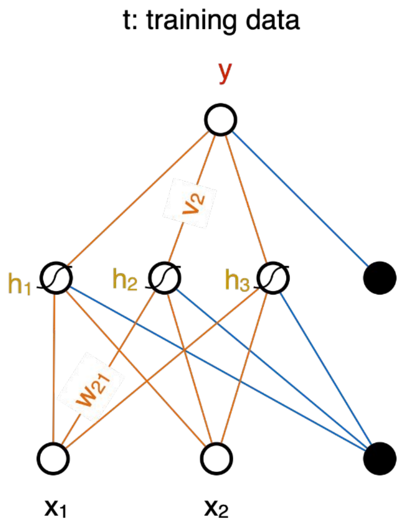
Neural Networks: Function approximation

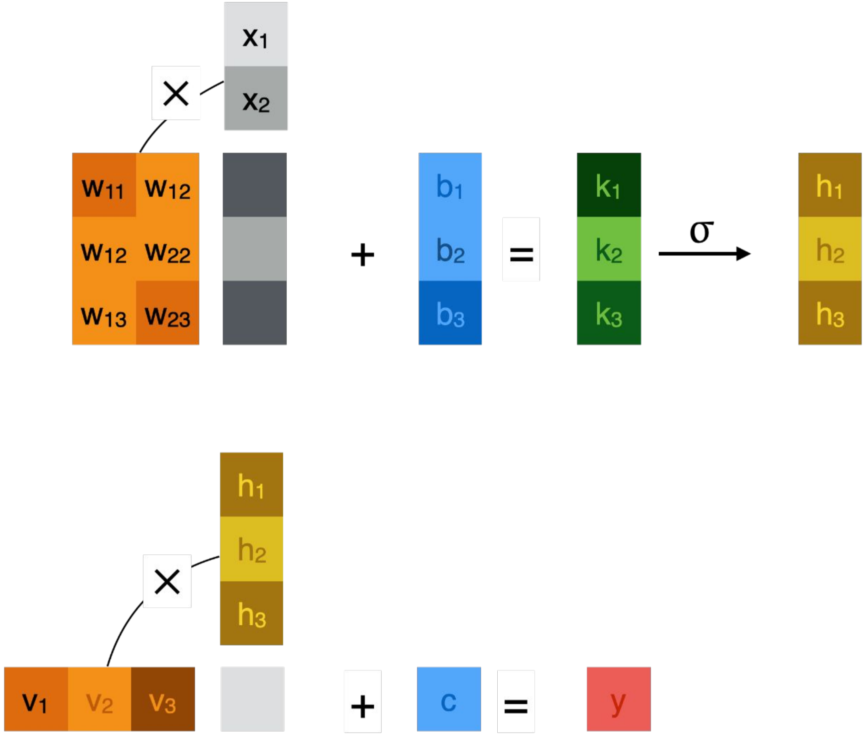




Universal function approximators!

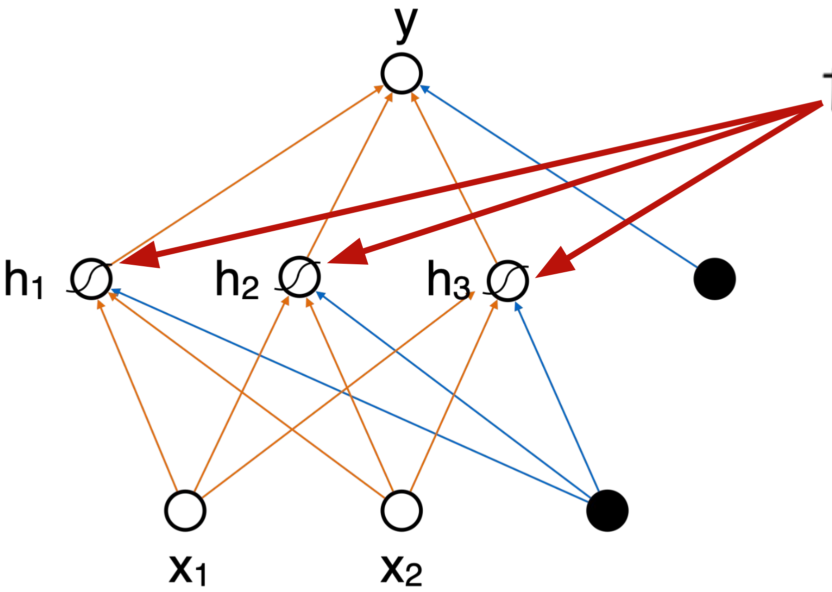
Neural Networks: it's all just linear algebra





x

Neural Networks: it's all just linear algebra



$$f(\mathbf{x}) = \mathbf{V}\,\sigma(\mathbf{W}\mathbf{x} + \mathbf{b}) + \mathbf{c}$$

Application of Deep Learning

- Computer Vision
- Information Retrieval
- Recommendation Systems
- Natural Language Processing
- Machine Translation
- Speech Recognition & Synthesis
- Bioinformatics
- Medical Image Analysis
- and more …

Deep Learning Libraries

Deep Learning Frameworks

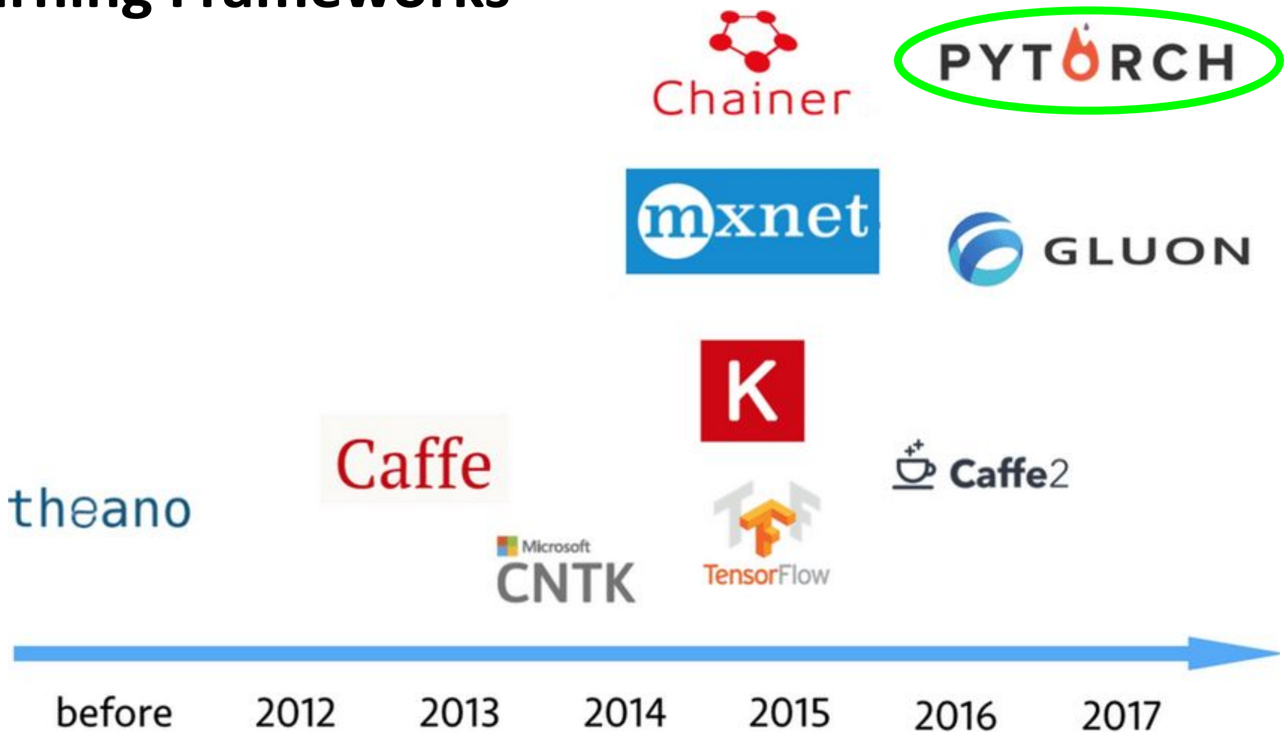


Image Source: https://www.researchgate.net/publication/349108714/figure/fig1/AS:988814684217344@1612763200002/Timeline-of-deep-learning-frameworks.png

PyTorch

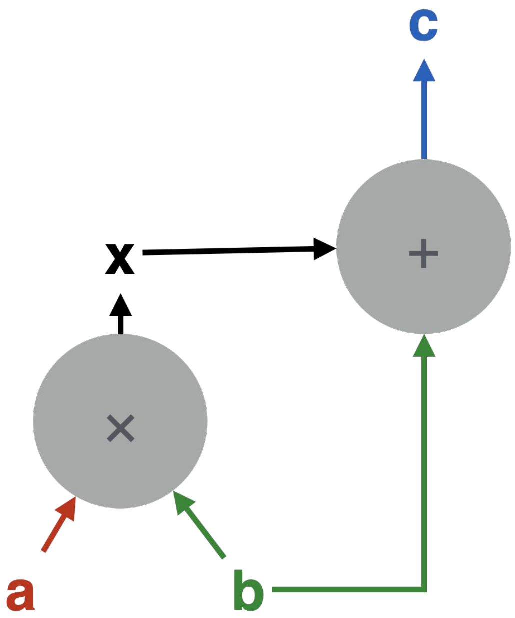
PyTorch

- Created by engineers at Facebook (now Meta)
- Open-source Python library
- Written to be very performant
- Distributed training across multiples computers
- Preferred by academics for machine learning research

Computation Graph

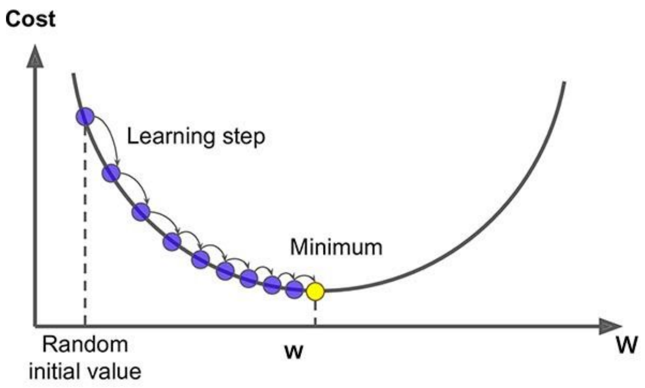
$$(a \times b) + b = c$$

- Directed a graphs that represent a mathematical expression
- Deep learning systems use computation graphs to:
  - execute a given computation (**the forward pass**)
  - compute the gradients for the data nodes with respect to the output using the backpropagation algorithm (**the backward pass**)



Gradient Descent

- Iteratively updates the parameters of the models to converge loss towards zero
- **Goal:** Find the global minimum of the loss curve



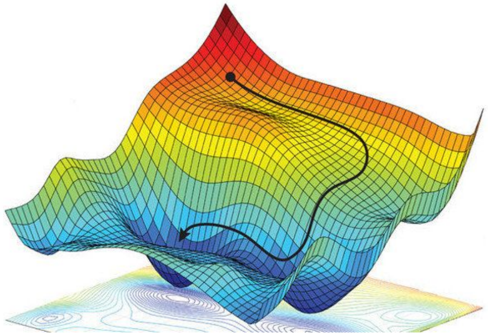


Image Source:
https://www.researchgate.net/profile/Alexander-Amini/publication/325142728/figure/fig1/AS:766109435326465@1559666131320/Non-convex-optimization-We-utilize-stochastic-gradient-descent-to-find-a-local-optimum.jpg

Gradient Descent

- Iteratively updates the parameters of the models to converge loss towards zero
- **Goal:** Find the global minimum of the loss curve

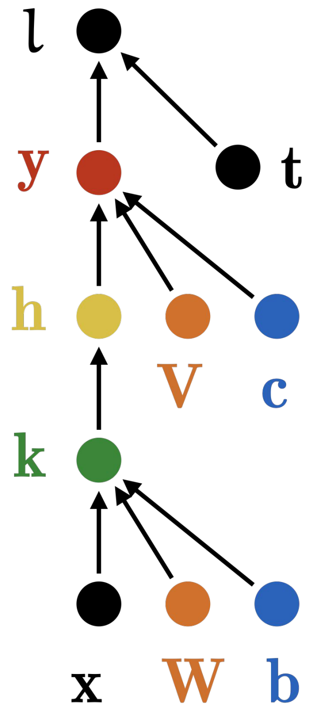
Gradient descent algorithm

$$
\begin{aligned}
&\text{repeat until convergence } \{ \\
&\quad \theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1) \\
&\qquad \text{(for } j = 1 \text{ and } j = 0) \\
&\}
\end{aligned}
$$

- Initial model parameters
- Learning rate
- Gradients of the parameters
- Updated model parameters

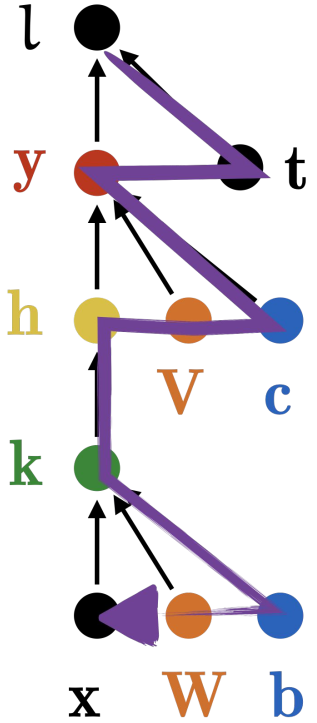
Backpropagation Algorithm

- Facilitates ***learning*** by correcting the output of model using the error signal
- Computes gradients for the parameters of the model



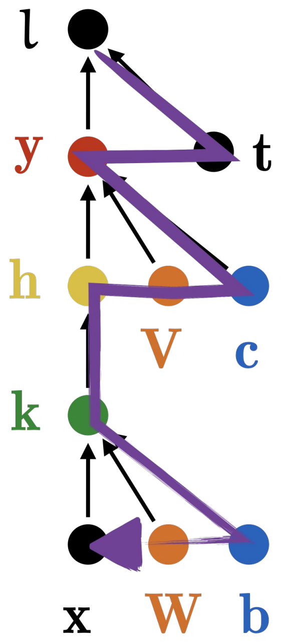
Backpropagation Algorithm

- Facilitates ***learning*** by correcting the output of model using the error signal
- Computes gradients for the parameters of the model
- Walk down the computation graph from the loss to the inputs
- In PyTorch, you can simply use `.backward()` to backpropagate through the model



Automatic Differentiation

- Perform computation by chaining functions
- Keep track of all computation in a computation graph
- When the computation is finished, walk backward through the computation graph to perform backpropagation



PyTorch: Basic Training Code

```
parameters = list(model.parameters()) # retrieve parameters of models
optimiser = Adam(lr=0.0003, params=parameters)
optimiser.zero_grad()

epochs = 5
for epoch in range(epochs):
      opt.zero_grad()

      y = model(x)  # forward pass
      loss = F.binary_cross_entropy(x, y, reduction='none').sum()

      loss.backward() # backward pass

      optimiser.step()
```

Interactive Session #1

https://bit.ly/35B40TH

https://bit.ly/35B40TH

Google Colab

- Interactive interface for coding
  - Similar to Jupyter Notebooks
- Uses Google's Cloud Infrastructure to train deep learning models
- Pre-installed with important machine learning packages

Instructions:

- Clone the file ***"File" -> "Save as a copy in Drive"***
- Run the code in the cells ***"SHIFT + ENTER"***
- Experiment with the code
- Complete the task at the bottom of the notebook
- If you need help, refer to the documentation https://pytorch.org/docs/stable/index.html

Building & Testing Deep Learning Models

Common Pitfalls

- Not the correct Deep Learning Architecture
- Not enough data
- Not enough features
- Not tuned for the task
- Test data may come from a different distribution!
- Not the right signals are used for training

- May be the task can not be learned!

Class Imbalance

Class Imbalance

- Use a big test set
- *Resample* your training data
- Don't rely on accuracy. Try ROC plots, precision-recall plots, AUC, etc. Look at the confusion matrix
- Use data augmentation for the minority class

Oversampling

Undersampling

Undersampling

Both approach change the data distribution!

Data Augmentation

- Idea: Create new data points by augmenting existing data points
- Image domain
  - rotate or translate images in the minority class
  - add gaussian noise
- Remember: Only augment the training data. Keep the test data as is.

Practical Tips

- Pay attention to loss curves
- Be Cautious with the Test data
  - Leave it untouched until the end
- Overcoming vanishing gradients
  - Proper initialisation, ReLUs over sigmoids
- Minibatch gradient descent
- Optimizers
  - (Nesterov) momentum, Adam
- Regularizers
  - L1, L2, Dropout

Hyperparameter Tuning

- Hyperparameters should be fine-tuned
  - To get best performance on *cross-validation dataset*
- Time consuming and expensive process
- Should be systematic
  - Only change one hyperparameters
- Prioritise hyperparameters for tuning
  - Learning-rate
  - Epochs
  - Hidden units
  - Hidden layers
  - ….
  - Optimiser

Multi-Layer Perceptron (MLP)

$$f(\mathbf{x}) = \mathbf{V}\,\sigma(\mathbf{W}\mathbf{x} + \mathbf{b}) + \mathbf{c}$$

- Simple and effective
  - Universal function approximators
- Specialised deep learning architectures can be more effective at solving more complex problems
  - Useful built-in inductive biases
  - Less data required for learning *(in theory)*

Overview of popular Deep Learning Architectures

Convolutional Neural Networks (CNNs)

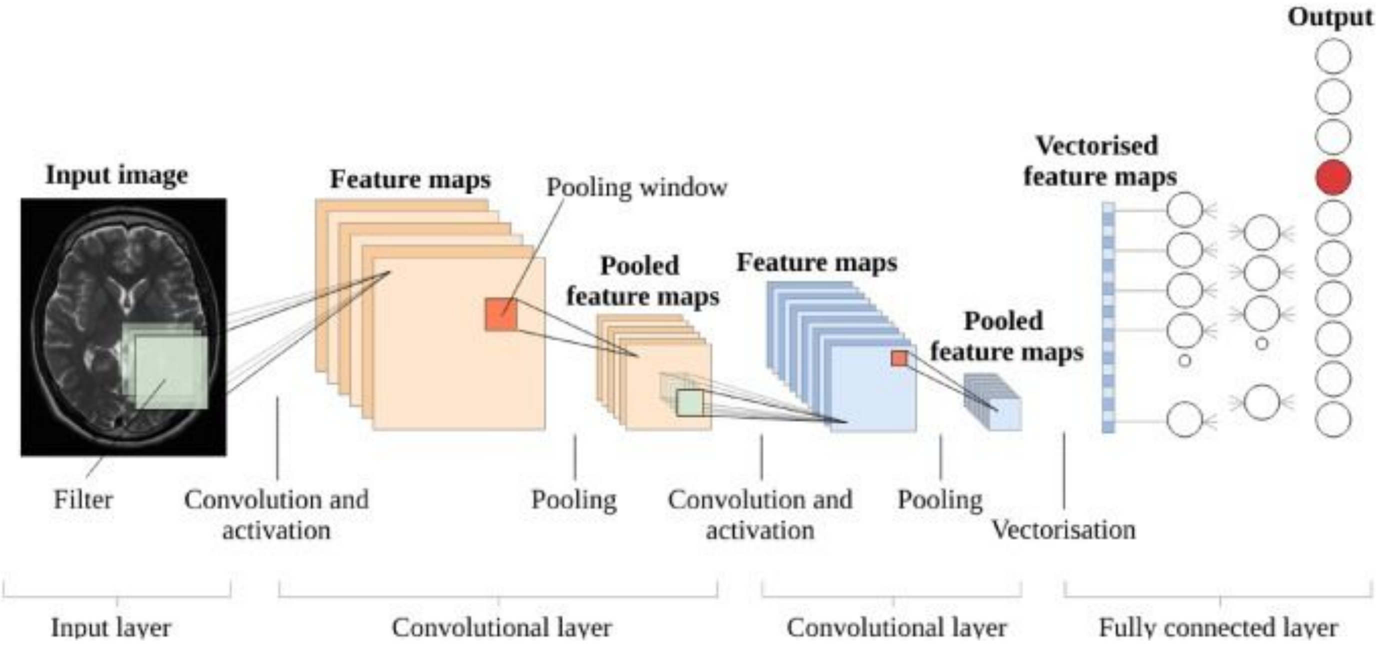


Image Source: https://ars.els-cdn.com/content/image/1-s2.0-S0939388918301181-gr2.jpg

Convolutional Neural Networks (CNNs)

Feature maps

Input Features
(e.g. Image pixels)

Image Source: https://miro.medium.com/max/1400/1*ulfFYH5HbWpLTIfuebj5mQ.gif

Convolutional Neural Networks (CNNs)

- Translation Invariance
- **Intuition:** The position of an object in an image should not influence how it is classified

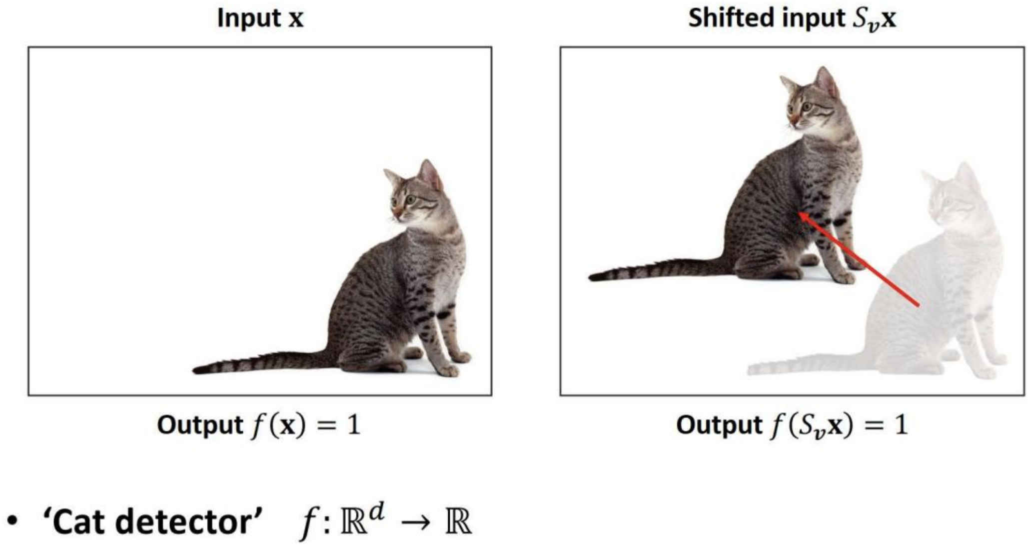
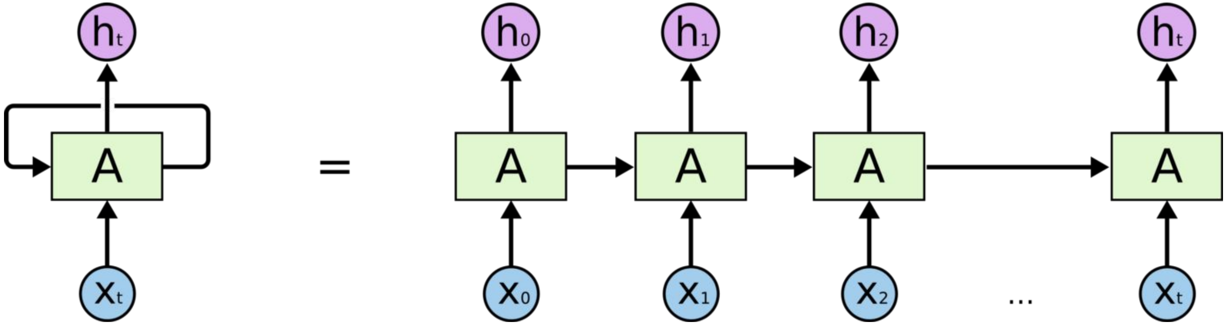


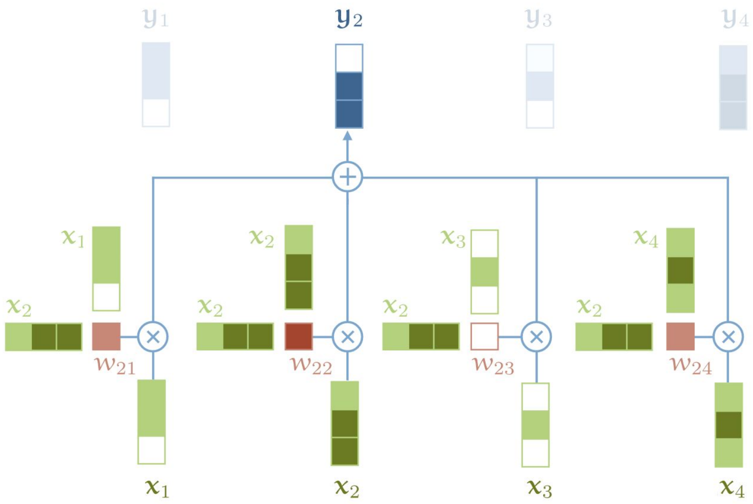
Image Source: https://www.doc.ic.ac.uk/~bkainz/teaching/DL/notes/equivariance.pdf

Recurrent Neural Networks (RNNs)



- Unlike MLP, RNNs take decisions based on current and previous inputs
- Sequential data
  - Time series
  - Natural Language

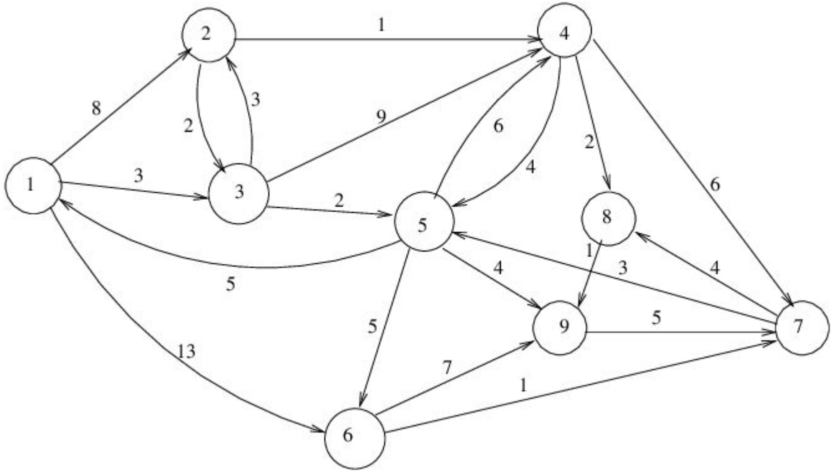
Transformers



- Self-Attention Mechanism
  - Takes a weighted average over all the input vectors in a sequence
- Achieve human-level performance across many tasks
  - Text generation
  - Text comprehension
  - Question Answering
  - ….

Recommended Reading: http://peterbloem.nl/blog/transformers

Graph Neural Networks (GNNs)



Applications

- Link Prediction
- Molecular Property Prediction
- Social Network Analysis
- Recommender system

Graph Neural Networks (GNNs)

- Averaging vector representation of neighboring nodes
- **Intuition:** Nodes in the graph share information with their neighboring nodes (message passing)
- **Goal:** Every node has a unique node vector representation

Libraries for Graph Representation Learning

- PyTorch Geometric (https://pytorch-geometric.readthedocs.io)
- Deep Graph Learning (https://www.dgl.ai)

Autoencoder

- Unsupervised learning
- Learning latent vectors (*z*) for data

Two components:

- ***Encoder Network*** translates the original high-dimensional input into the latent low-dimensional vectors
- ***Decoder Network*** recovers the data from the code, likely with larger and larger output layers

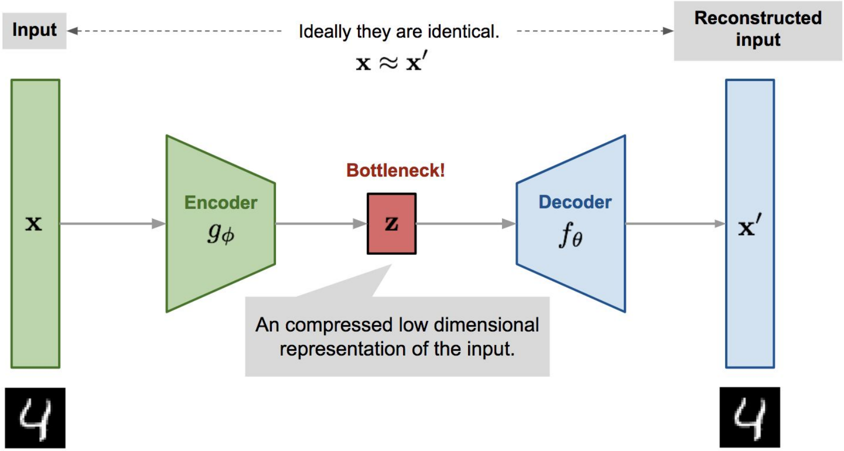
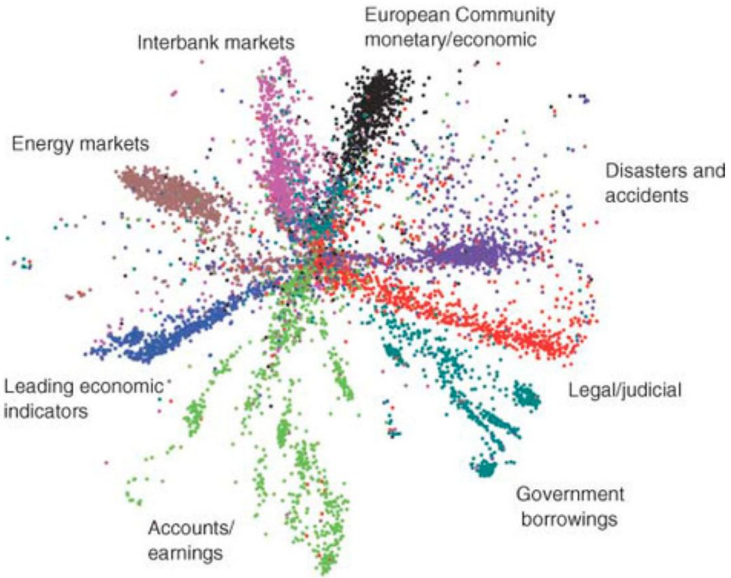


Image Source: https://lilianweng.github.io/posts/2018-08-12-vae/autoencoder-architecture.png

Autoencoder



Can you spot the *real* faces?

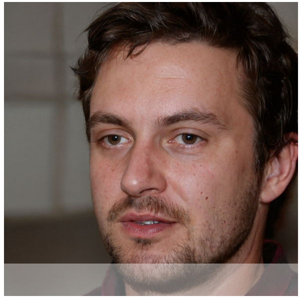







Can you spot the *real* faces?






None of these people exist!

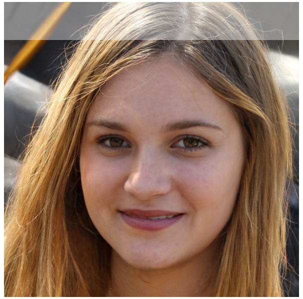
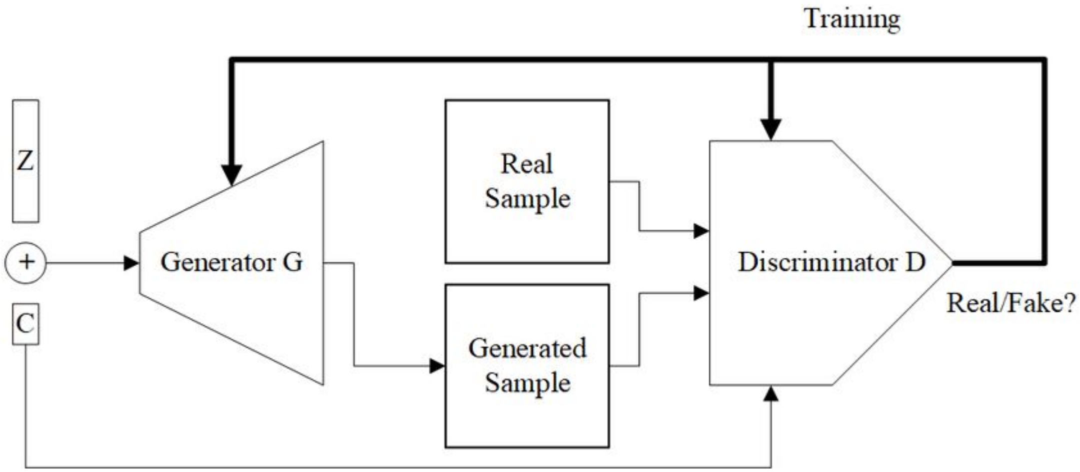



Image Source: https://this-person-does-not-exist.com/en

Generative Models



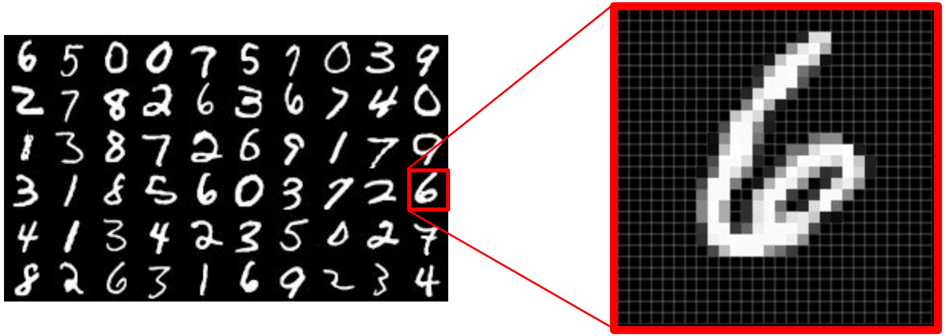
Generative Adversarial Network

- ***Discriminator Network*** is optimized to tell the fake samples from the real ones
- ***Generator Network*** is trained to capture the real data distribution so that its generative samples can be as real as possible and trick the discriminator.

Image Source:
https://www.researchgate.net/profile/Alptekin-Temizel/publication/330593980/figure/fig1/AS:736664213274624@1552645843455/a-Standard-GAN-and-b-conditional-GAN-architectures.ppm

Interactive Session #2

https://bit.ly/3vZy1XP



https://bit.ly/3vZy1XP

Google Colab

- Interactive interface for coding
  - Similar to Jupyter Notebooks
- Uses Google's Cloud Infrastructure to train deep learning models
- Pre-installed with important machine learning packages

Instructions:

- Clone the file ***"File" -> "Save as a copy in Drive"***
- Run the code in the cells ***"SHIFT + ENTER"***
- Experiment with the code
- Complete the task at the bottom of the notebook
- If you need help, refer to the documentation https://pytorch.org/docs/stable/index.html

Current State of Deep Learning

TechRepublic.

INNOVATION

Google uses AI, deep learning to predict cardiovascular risk from retina scans

Google's deep learning algorithm could more accurately detect a patient's risk of heart disease and stroke using a scan of their retina.

By Alison DeNisco Rayome | February 20, 2018, 5:54 AM PST

0

BBC | Home | News | Sport | More

NEWS

Business

What is 'deep learning'?

What is artificial intelligence?

The rob like a ch

What is 'deep learning'?

Every day we create billions of bits of data. Ever faster and more powerful computers can use that big data to learn, predict events

Home | The New York Times Magazine | Share

The Great A.I. Awakening

How Google used artificial intelligence to transform Google Translate, one of its more popular services — and how machine learning is poised to reinvent computing itself.

BY GIDEON LEWIS-KRAUS DEC. 14, 2016

News | Opinion | Sport | Culture | Lifestyle | More

World UK Science Cities Global development Football Tech Business Environment Obituaries

Science

'It's able to create knowledge itself': Google unveils AI that learns on its own

In a major breakthrough for artificial intelligence, AlphaGo Zero took just three days to master the ancient Chinese board game of Go ... with no human help

Ian Sample
Science editor

@iansample
Wed 18 Oct 2017
18.00 BST

This article is 4 months old

The AI Hype

- Achievement of deep learning is sometimes overblown in the media
- General Public have high expectation
- Realistic view of current state of the art is important!

NEWS ANALYSIS

Did IBM overhype Watson Health's AI promise?

IBM's Watson Health division has been under fire for not delivering on its promise to use AI to enable smarter, more personalized medicine. But IBM officials maintain that hospitals are seeing benefits.

By Lucas Mearian

Senior Reporter, Computerworld | 14 NOVEMBER 2018 12:00 CET

In recent weeks, IBM has changed leadership at its Watson Health division and announced a new business strategy for deployment that relies on a hybrid cloud, not a public- or private-cloud only model.

Over the past year, Watson Health – particularly Watson for Oncology – has come under criticism for not meeting expectations or even offering physician users inaccurate advice. (Watson for Oncology is IBM's commercial cognitive computing cloud platform that analyzes large volumes of patient healthcare data and published medical studies to offer physicians cancer treatment options.)

[**Further reading:** A.I. and speech advances bring virtual assistants to work]

Laura Craft, a vice president of research for Gartner's Healthcare Strategy business, said IBM's Cognitive Computing Division did not do well in recent third-quarter results, "and that was largely driven by the healthcare component."

Craft also pointed to the recent leadership changes as indicative of internal problems.

Reports of trouble, and IBM's defense

In July, the healthcare news publication *Stat* published a report claiming "internal IBM documents" showed the Watson supercomputer often spit out erroneous cancer treatment advice and that company medical specialists and customers identified "multiple examples of unsafe and incorrect treatment recommendations," even as IBM was promoting its AI technology.

News Source: Did IBM overhype Watson Health's AI promise?

The AI Hype

- Achievement of deep learning is sometimes overblown in the media
- General Public have high expectation
- Realistic view of current state of the art is important!

Using AI to screen live video of terrorism is ‘very far from being solved,’ says Facebook AI chief

Live-streamed attacks like Christchurch shooting require human moderation

By James Vincent | May 20, 2019, 7:12am EDT

SHARE

Illustration by Alex Castro / The Verge

Listen to this article

When faced with hard questions about how Facebook will remove terrorist content from its platforms, CEO Mark Zuckerberg offers a simple answer: artificial intelligence will do it. But according to Facebook’s chief AI scientist, Yann LeCun, AI is years away from being able to fully shoulder the burden of moderation, particularly when it comes to screening live video.

Speaking at an event at Facebook’s AI Research Lab in Paris last week, LeCun said Facebook was years away from using AI to moderate live video at scale, reports *Bloomberg News*.

“This problem is very far from being solved,” said LeCun, who was recently awarded the Turing Prize, known as the Nobel Prize of computing, along with other AI luminaries.

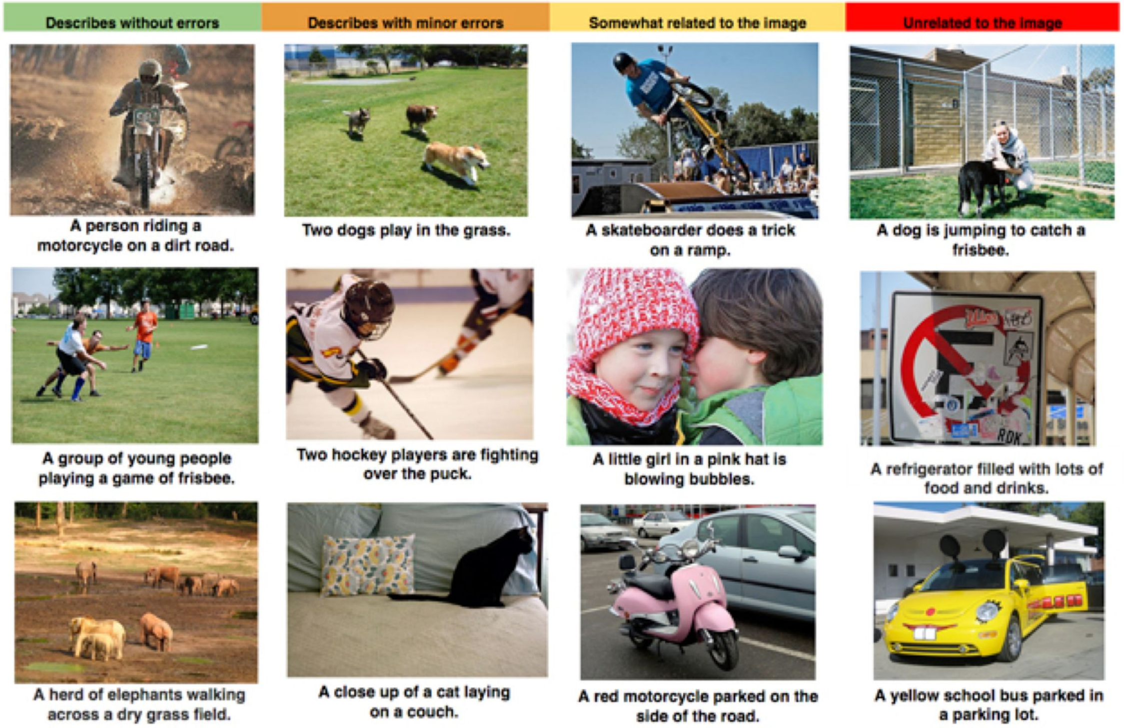
Screening live video is a particularly pressing issue at a time where terrorists commit atrocities with the aim of going viral. Facebook’s inability to meet this challenge became distressingly clear in the aftermath of the Christchurch shooting in New Zealand this year. The attack was streamed live on Facebook, and although the company claims it was seen by fewer than 200 people during its broadcast, it was this stream that was then downloaded and shared across the rest of the internet.

The inability of automated systems to understand and block content like this isn’t news for AI experts like LeCun. They’ve long

AI CAN REMOVE UNWANTED CONTENT, BUT ONLY AFTER A

News Source: [Facebook AI is Far From Able to Moderate Live Video]

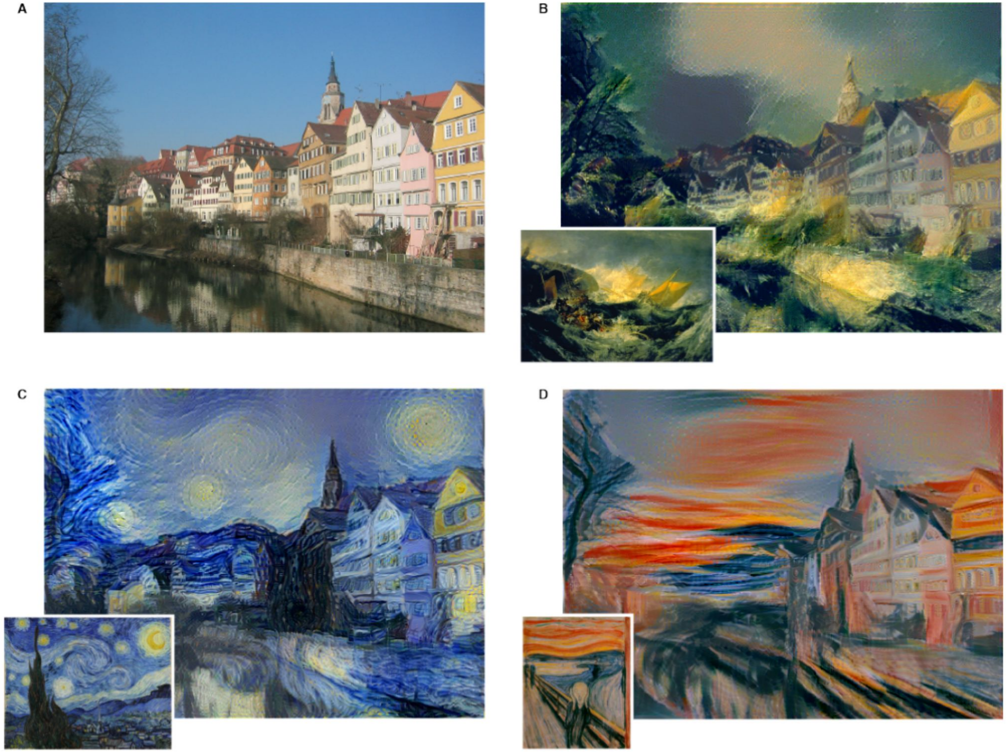
Natural Language Description



2014

Source: https://www.engadget.com/2014/11/18/google-natural-language-image-description/

Style Transfer



2015

Source: https://research.googleblog.com/2016/10/supercharging-style-transfer.html

Fake Face Generation

Ian Goodfellow
@goodfellow_ian

Following

4.5 years of GAN progress on face generation. arxiv.org/abs/1406.2661 arxiv.org/abs/1511.06434 arxiv.org/abs/1606.07536 arxiv.org/abs/1710.10196 arxiv.org/abs/1812.04948

Source: https://twitter.com/goodfellow_ian/status/1084973596236144640

Limitations of Deep Learning

- Transparency
- Requirement for modules to be differentiable
- Correlation ≠ Causation
- Large amounts of data
- Large compute infrastructure (e.g. GPU’s & TPU’s)
- Hidden biases in data
  - ***Explainability***
- Social Impact

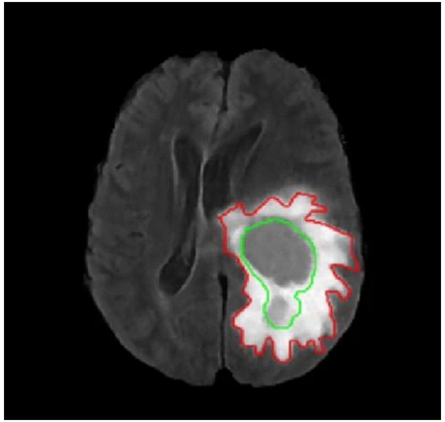
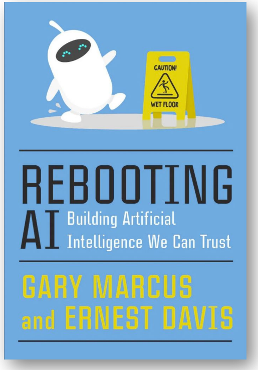
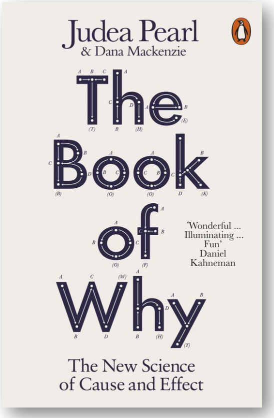


Image Source:
https://www.researchgate.net/profile/Jason-Corso/publication/5797261/figure/fig2/AS:601719062921223@1520472412251/An-example-of-the-brain-tumor-segmentation-Left-case-is-from-the-training-set-and-right.png

Limitations of Deep Learning

- Transparency
- Requirement for modules to be differentiable
- Correlation ≠ Causation
- Large amounts of data
- Large compute infrastructure (e.g. GPU's & TPU's)
- Hidden biases in data
  - *Explainability*
- Social Impact





Limitations of Deep Learning

- Transparency
- Requirement for modules to be differentiable
- Correlation ≠ Causation
- Large amounts of data
- Large compute infrastructure (e.g. GPU’s & TPU’s)
- Hidden biases in data
  - ***Explainability***
- Social Impact

What is the social impact of your work?

Key Takeaways

- Learning low dimensional vector representations for a specific downstream application
- Be cautious about *data distribution*
- Deep learning frameworks make it quick to prototype and test models
- Different deep learning architectures have different *inductive biases* built into them

Resources

- *Pytorch 60 minute blitz*
  https://pytorch.org/tutorials/beginner/deep_learning_60min_blitz.html
- *Learning Pytorch with examples*
  https://pytorch.org/tutorials/beginner/pytorch_with_examples.html
- *Machine Learning Course* at Vrije Universiteit Amsterdam
  https://mlvu.github.io/
- *Deep Learning Course* at University of Amsterdam
  https://uvadlc.github.io/
- *Probabilistic Machine Learning* by Kevin P. Murphy
  https://probml.github.io/pml-book/

Solutions

- Interactive Session Part 1
  https://bit.ly/3CzjKmc
- Interactive Session Part 2
  https://bit.ly/3HZFR6m

Thank you for your attention!

t.singam@uva.nl
https://www.linkedin.com/in/thiviyan-thanapalasingam-b16727133/

Questions

- Information in general: what can we expect from deep learning and what not?
- How to deal with low number of objects/samples
- Transfer learning
- Incorporate prior knowledge
- Explainability in deep learning
- Text classification tasks especially when the training set is rather limited