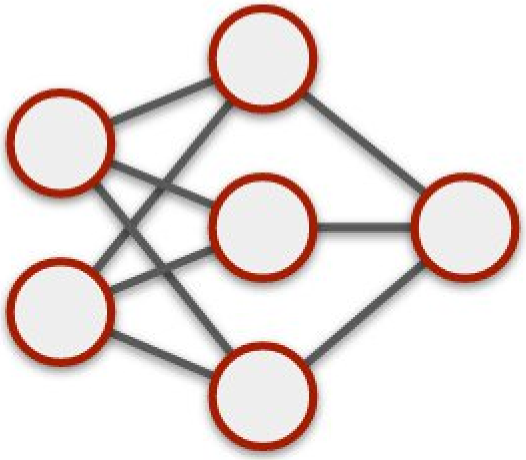


Neural Network Building Blocks (1/2)

Docent: Sebastian Wozniewski (Uni Göttingen)
Tutor: Steffen Korn (Uni Göttingen)

Deep Learning School — Basic Concepts
27.02.2023

About me

2016-2020 Master and PhD at the Karlsruhe Institute of Technology

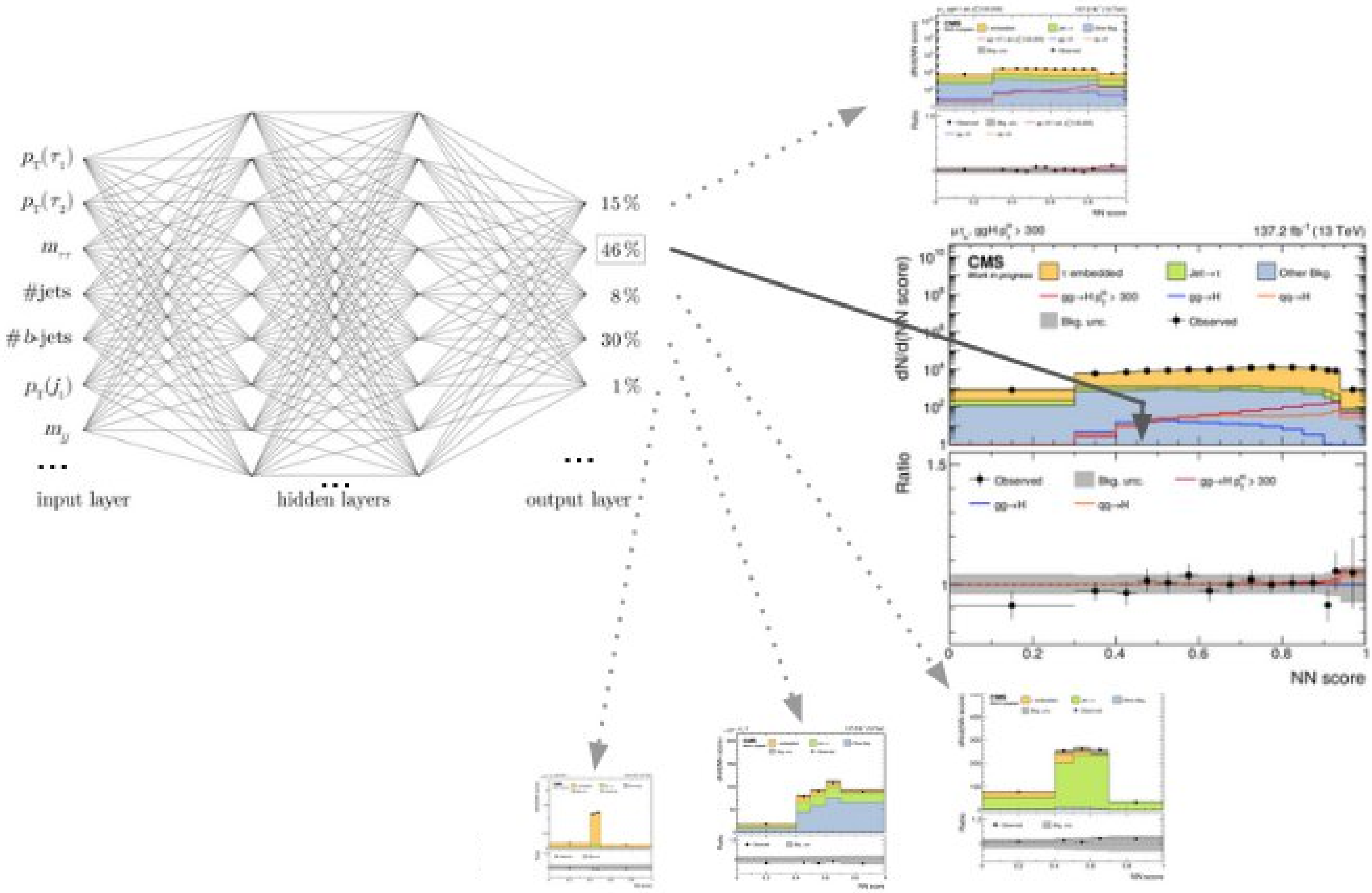
Data Analysis within the CMS collaboration:
Analyses of $H \to \tau\tau$ events using BDT / NN driven event classification

since 2021 PostDoc at University of Göttingen

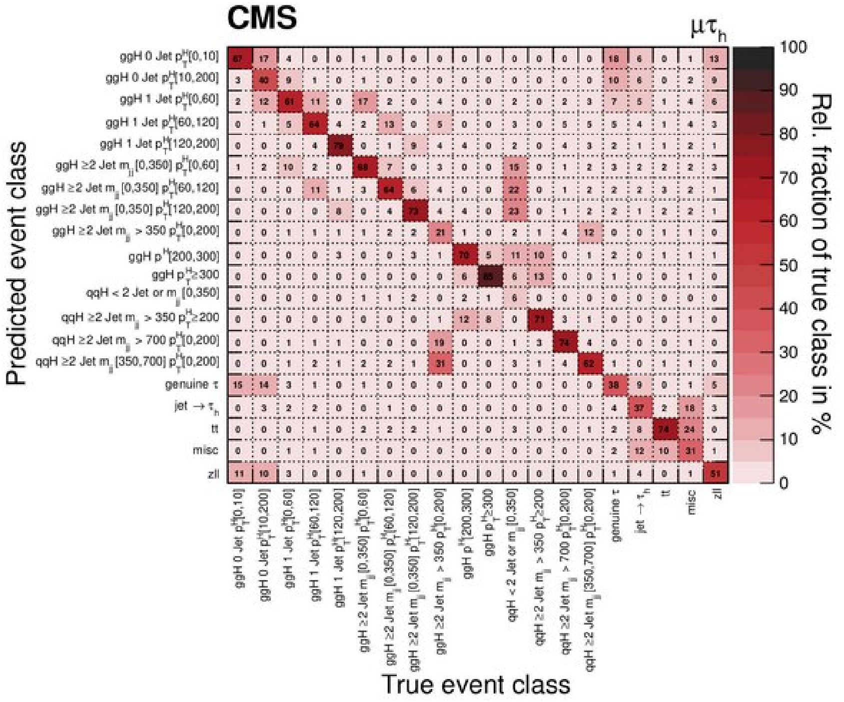
Grid Computing within the ATLAS collaboration

contact: sebastian.wozniewski@uni-goettingen.de

About me



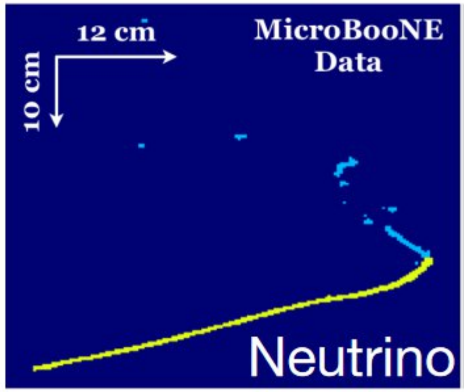
$p_T(\tau_1)$
$p_T(\tau_2)$
$m_{\tau\tau}$
#jets
#b-jets
$p_T(j_1)$
m_{jj}
...
input layer
...
hidden layers
15 %
46 %
8 %
30 %
1 %
...
output layer
CMS
Work in progress
137.2 fb^{-1} (13 TeV)
dN/d(NN score)
Ratio
Observed
Bkg. unc.
NN score



CMS
$\mu\tau_h$
Predicted event class
True event class
Rel. fraction of true class in %

Artificial Intelligence in Physics

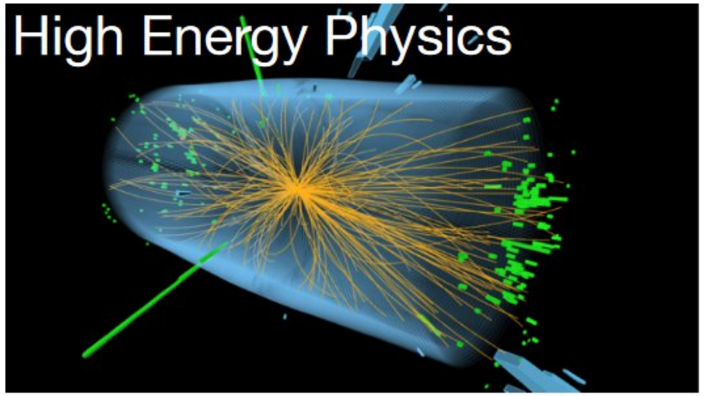
Sebastian Wozniewski — 27.02.23

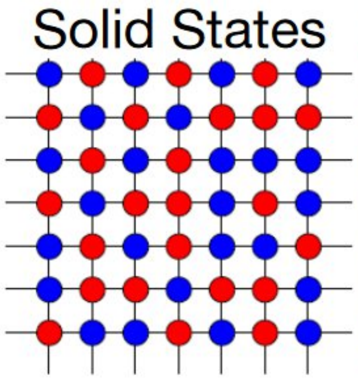


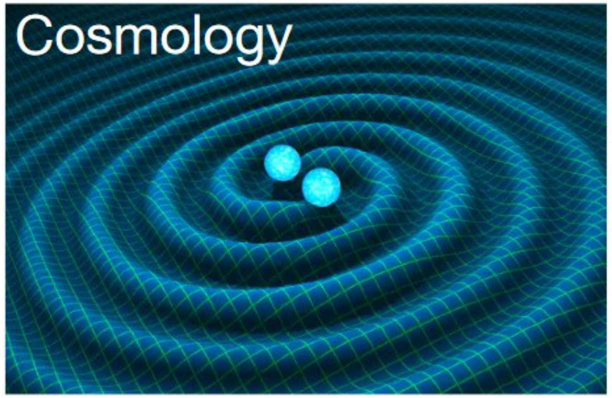
Complex data
- image-like data, multidimensional space of observables

Large amounts of data

Simulation







And more...

Outline

This evening:

- Introduction to Machine Learning
- Basic concepts and analogies
- Quick summary of other methods than Neural Networks
- Neural Networks
  - What is it?
  - Basic elements of a training
  - Architectures

Tomorrow morning:

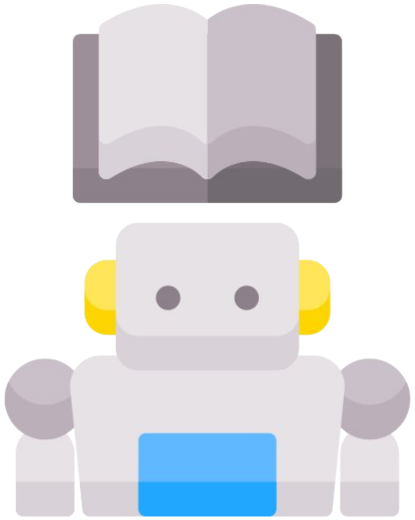
- Classifier results
- Class weights
- Performance metrics
- Dealing with large data sets
- Hands-On

Feel free to ask questions in between!

From the recent news: ChatGPT

Revolutionary AI abilities according to the latest news:

- communicating with you
- doing your homework
- writing ebooks to be sold on Amazon
- ...



OpenAI (developer) about limitations of ChatGPT:
(from https://openai.com/blog/chatgpt/ 22.02.2023)

ChatGPT sometimes writes **plausible-sounding but incorrect or nonsensical answers**. Fixing this issue is challenging, as: (1) during RL training, there's currently **no source of truth**; (2) training the model to be **more cautious causes it to decline questions that it can answer correctly**; and (3) supervised training misleads the model because the **ideal answer depends on what the model knows, rather than what the human demonstrator knows**.

Learning machines?

Learning is great, but even machines are “lazy”. Be aware of pitfalls!

Why is that a horse?

Why is that a painting by Monet?

Learning machines?

Learning is great, but even machines are “lazy”. Be aware of pitfalls!

Why is that a horse?

Because it’s standing
on green grass!

Why is that a painting by Monet?

Because his name is written here!

Or in other words: The machine might be more intelligent than you expect!

Some terms

Artificial intelligence - algorithm that mimics intelligent behavior

Machine learning - Procedure to tailor an intelligent algorithm in an automatized way based on ‘experience’ (input data)

Deep learning - Methods to train multilayer neural networks

History of Approaching complex problems

100 years ago:	Do it yourself
20 years ago	Invent and implement an algorithm
Today	Train a generalized algorithm that has already been invented and implemented

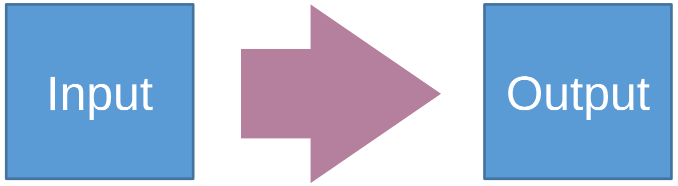
Driving motivation: Less work or more throughput/complexity for the same work!

History of Approaching complex problems

100 years ago:	Do it yourself
20 years ago	Invent and implement an algorithm
Today	Train a generalized algorithm that has already been invented and implemented

Driving motivation: Less work or more throughput/complexity for the same work!

Abstractly speaking it always is about a mapping:



The mapping should be precise but also efficient in computation.
→ Finding such a mapping, e.g. training a neural network, has many similarities to functional regression!

Analogies to functional regression

Supervised Machine Learning

- set of labeled data
- choose a ML method, network architecture, … based on a rough idea how the problem could be solved
- run a training with the labeled data
- map other data with the resulting model
- be careful with horses that do not stand on green grass

Functional Regression

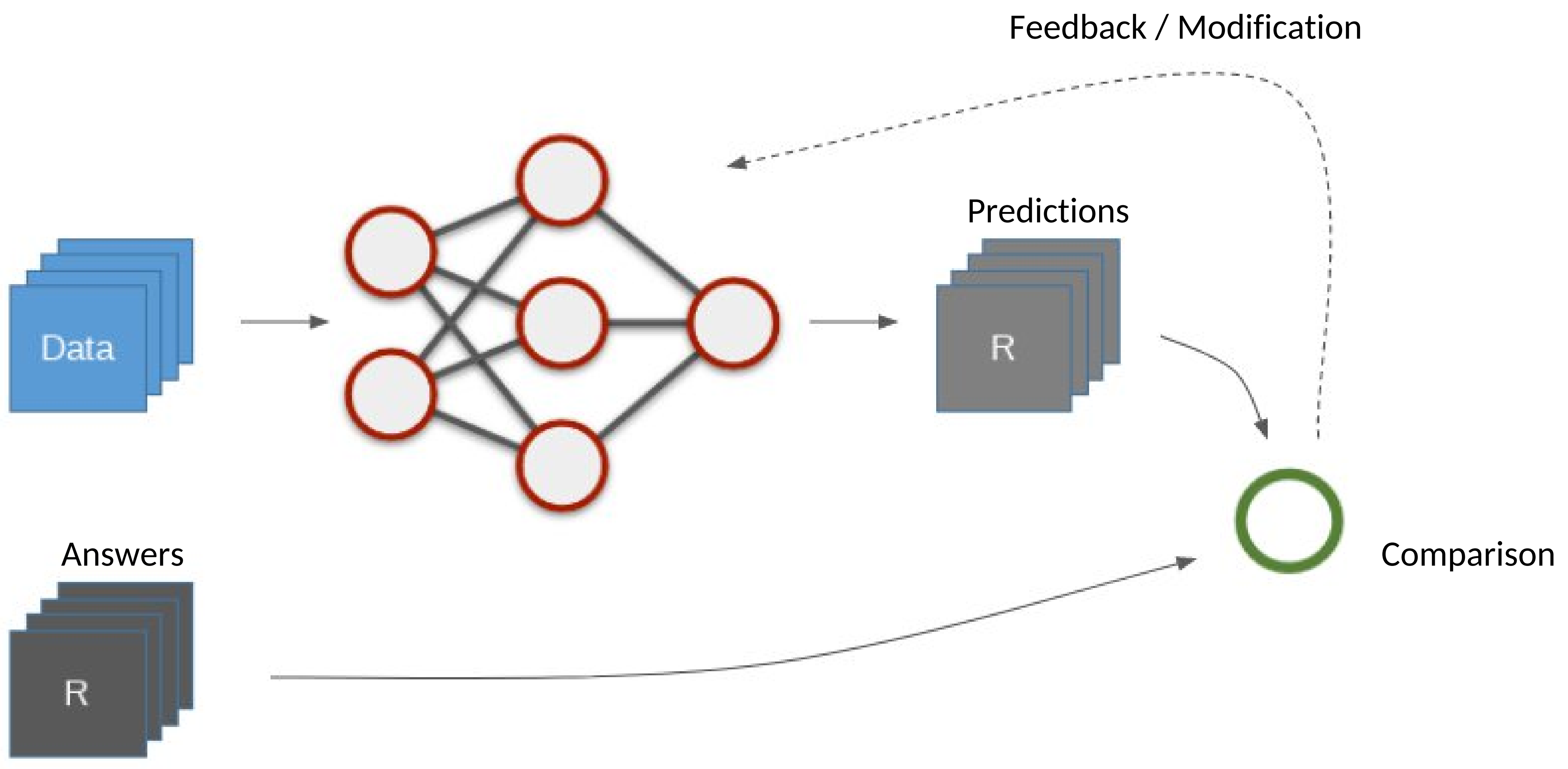
- set of known points (x, y)
- choose a parametrized function based on a rough idea what the relation of x and y is kind of
- run a fit to known points
- map other x with the resulting function
- be careful with extrapolations

Training Techniques in general

Aspect	Supervised Machine Learning	Unsupervised Machine Learning	Reinforcement Learning
Definition	Machine learns from well-known labeled data	Machine trained on unlabeled data	Machine interacts with environment and learns from positive or negative feedback
Type of problems	Regression, Classification	Clustering, Association, Dimension Reduction	Reward-based
Type of data	Labeled data	Unlabeled data	No predefined data
Training	External supervision	No supervision	No supervision
Approach	Maps the labeled input to output	Finds patterns	Trial-and-Error

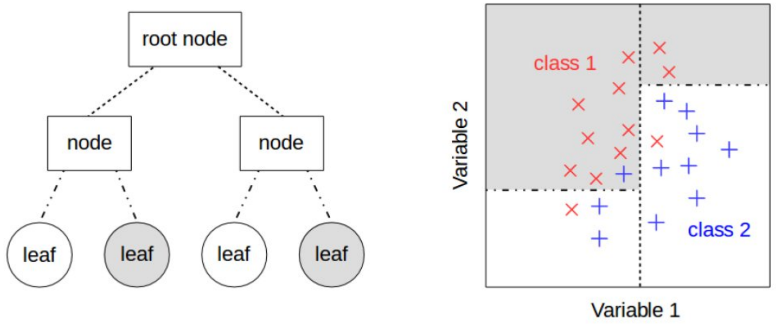
Our focus

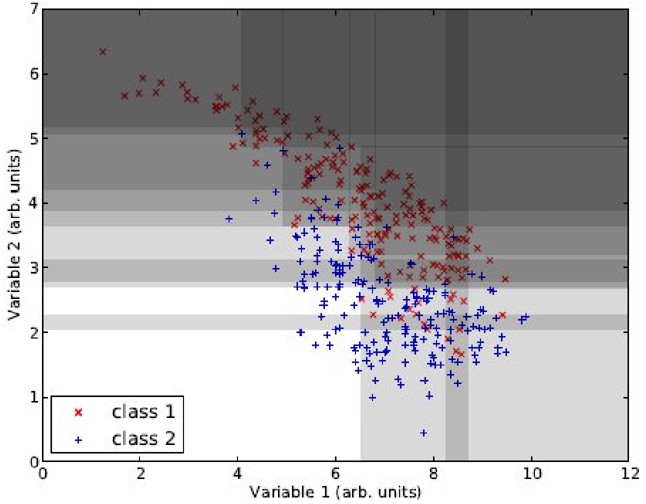
Supervised learning in a nut shell



Algorithms for supervised ML

- Linear regression
- Support Vector Machines
- Decision Trees
  - Random Forest
  - Boosted Decision Tree (BDT) => **(was) widely used in particle physics**
- Neural Networks => **widely and increasingly used in particle physics**
  - Multi-layer perceptron (fully connected, convolutional, deliberate compositions)
  - Recurrent NN
  - Graph NN
  - Generative Adversarial NN
- …



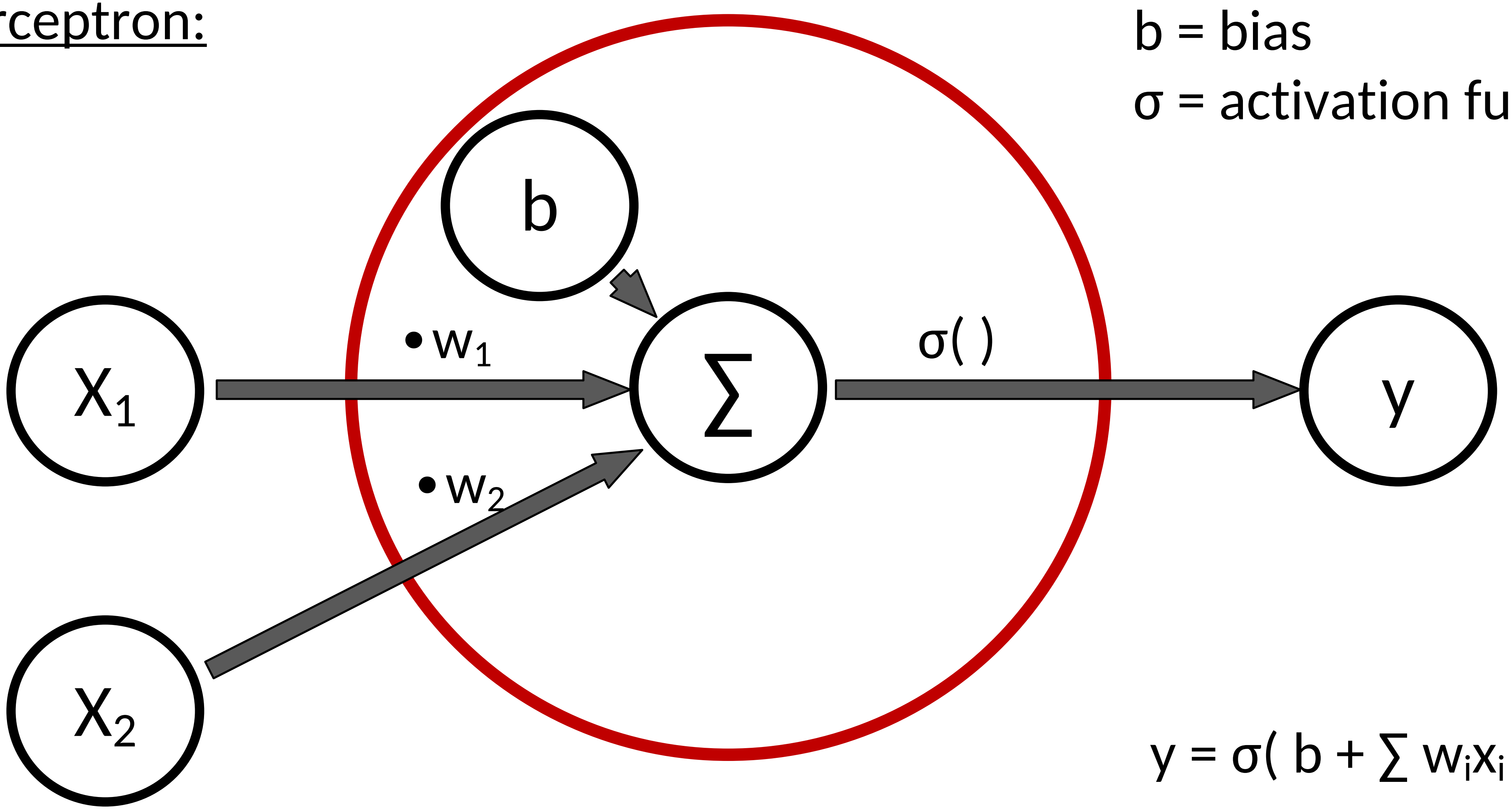


(f) 8 decision trees

Here, at the Deep Learning School we are going to learn about Neural Networks (with multiple layers)!

What is a neural network?

The perceptron:



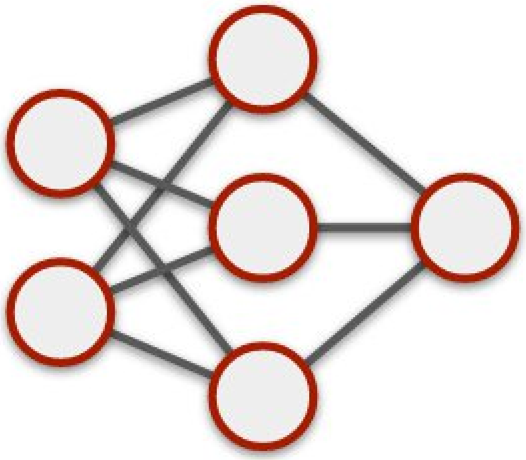
w = weights

b = bias

σ = activation function

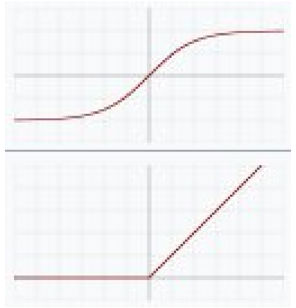
$$y = \sigma(\, b + \sum w_i x_i \,)$$

What is a neural network?



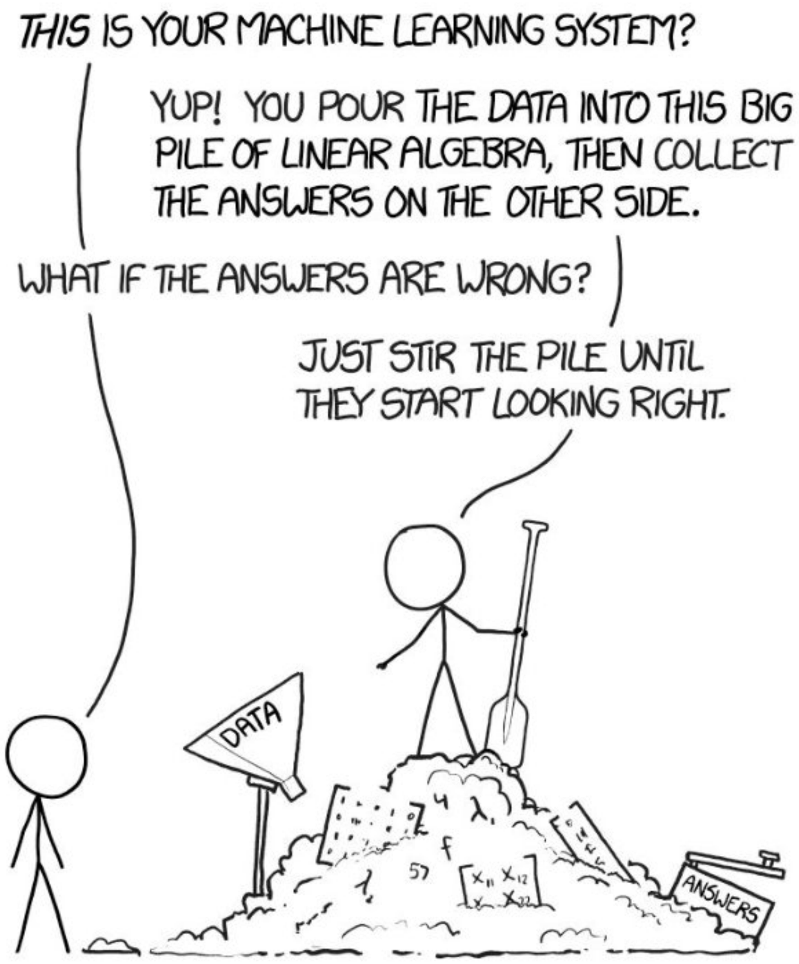
$$\begin{bmatrix} y_1 \\ y_2 \\ \dots \end{bmatrix} = \sigma\left(\begin{bmatrix} w_{11}\ w_{12}\ \dots \\ w_{21}\ w_{22}\ \dots \\ \dots \end{bmatrix} \bullet \begin{bmatrix} x_1 \\ x_2 \\ \dots \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \\ \dots \end{bmatrix} \right)$$

Activation function σ important, otherwise purely linear! More about that tomorrow…



Repeat this algebra for multiple layers

Now let's train this network!



xkcd.com

Training criterion: The loss function

Sebastian Wozniewski — 27.02.23

The loss function is a **measure for how much prediction and truth differ**.

Example from functional regression: Chi^2 would be the loss function (also applicable for ML driven regression)

$$L = \sum_i (y_i^{true} - y_i^{pred})^2$$

→ Training is supposed to minimize the loss function by adjusting weights and biases!

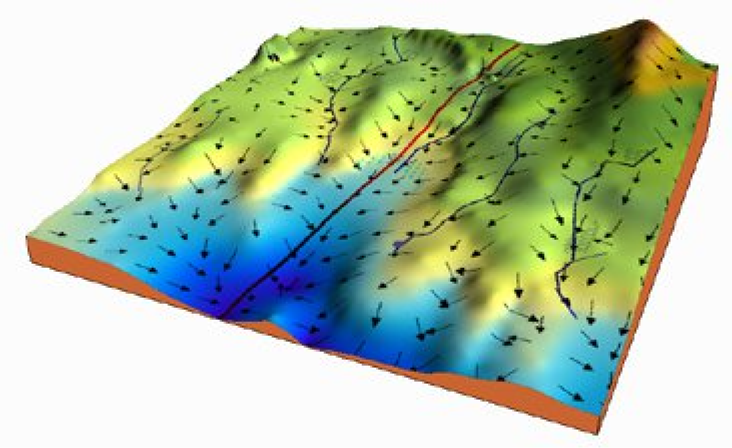
Training step: Gradient descent

Various ways of finding the best configuration of weights and biases.

In many cases **backpropagation / gradient descent** is the way to go!

Use invertibility or even derivability (depends on chosen activation functions) of the shown algebra to find beneficial adjustments. Tells you the local 'direction' to adapt the parameters but not the distance of direction of global minimum → **iterative procedure**.

Details tomorrow...



ml-cheatsheet.readthedocs.io

Overview of architectures

In theory a Neural Network can be designed deliberately large and as a fully connected network approximate deliberately complex mappings.
E.g. **Universal Approximation Theorem**: “A neural network with one hidden layer with a finite number of nodes can (in theory) approximate any reasonable function to arbitrary precision”
BUT: Large number of parameters that needs to be constrained by enough training data!

→ Invest some thoughts to design the architecture of your NN accordingly to the needs of your problem.
(like choosing a proper set of functions for a functional regression)

Selection of special types:

- Fully connected feed forward NN
- Convolutional NN (CNN): Image recognition, implements translational invariance
- Generative Adversarial Network (GAN): Training of two competitors for generation from random noise
- Recurrent NN (RNN): Include new sequential information e.g. over time
- Graph NN (GNN): Operating on data given in a graph format

Keep in mind: The data (format) determines your NN architecture!

Convolutional Neural Networks (CNN)

GEORG-AUGUST-UNIVERSITÄT GÖTTINGEN IN PUBLICA COMMODA SEIT 1737

Analysis of **image data**: Looking for certain shapes anywhere in a picture → translational invariance

Convolutional layers implement an image filter to be trained.

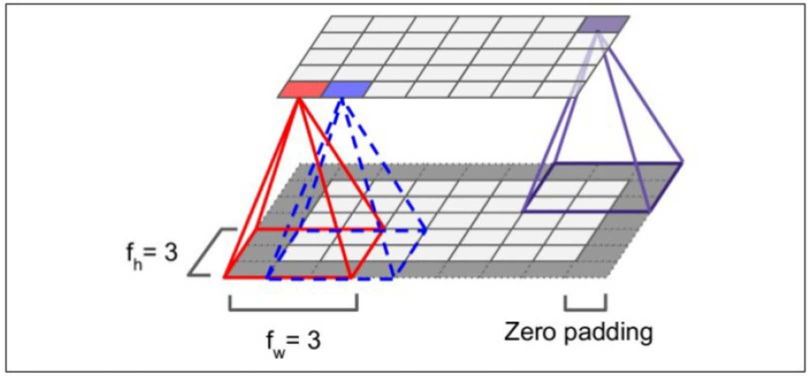


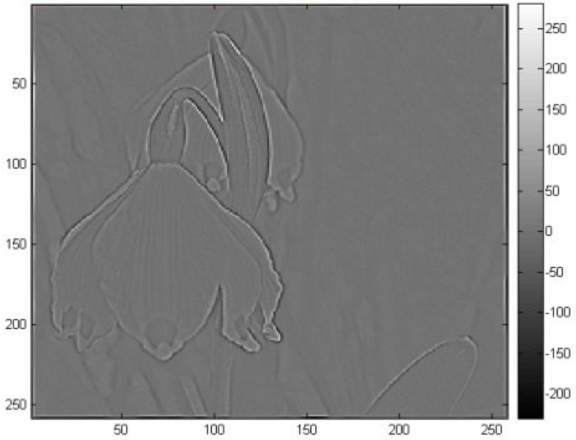
Figure 13-3. Connections between layers and zero padding

<u>Example of an image filter: Laplace filter</u>

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

Original

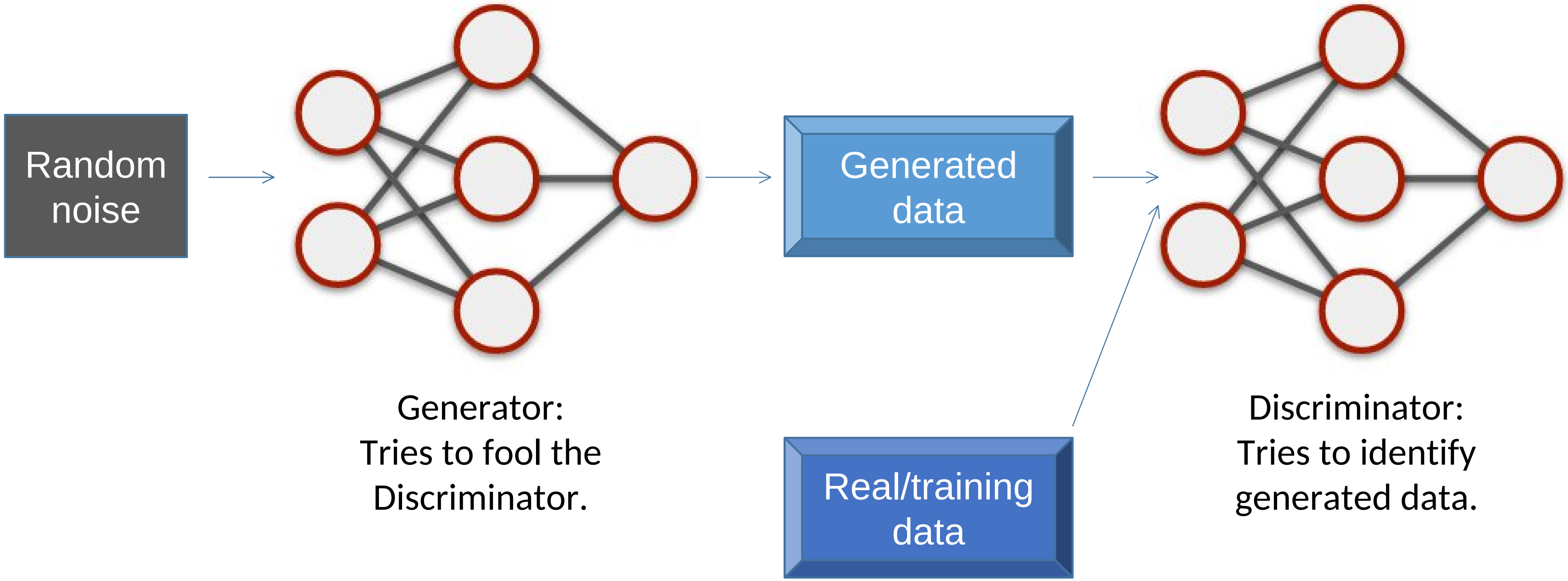
Laplace filter applied



More about CNNs on Wednesday!

Generative Adversarial Networks (GAN)

Method to **generate data** e.g. particle showers in a detector **from random noise input**.
Set of two ‘players’, i.e. the generator and the discriminator, being simultaneously trained against each other:

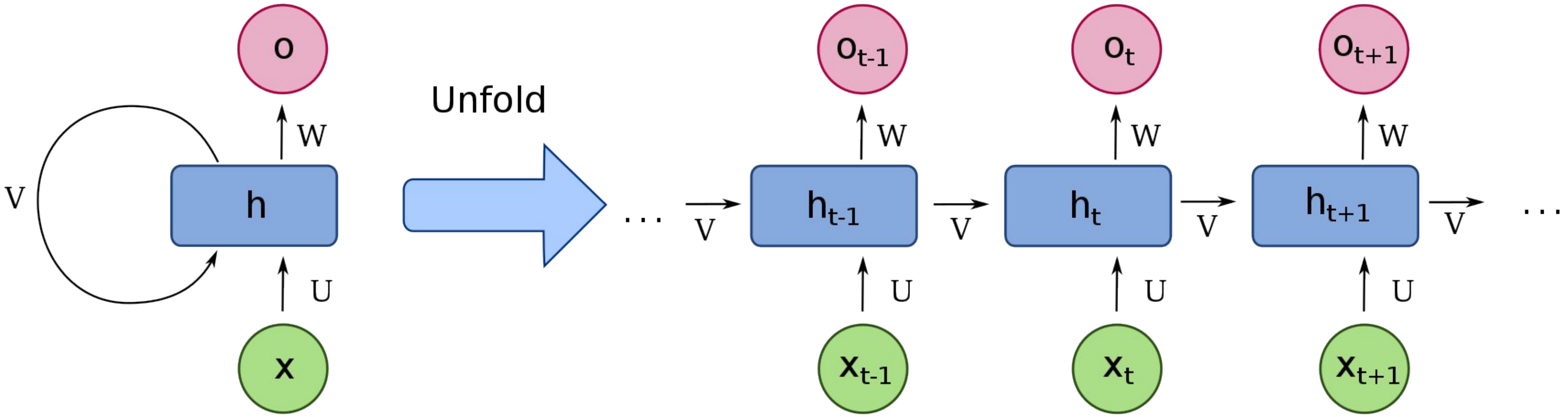


Generator:
Tries to fool the Discriminator.

Discriminator:
Tries to identify generated data.

Recurrent Neural Networks (RNN)

Output is fed back as input (loop) together with time dependent external input.
Applicable to **data sequences of variable length** e.g. for voice recognition.



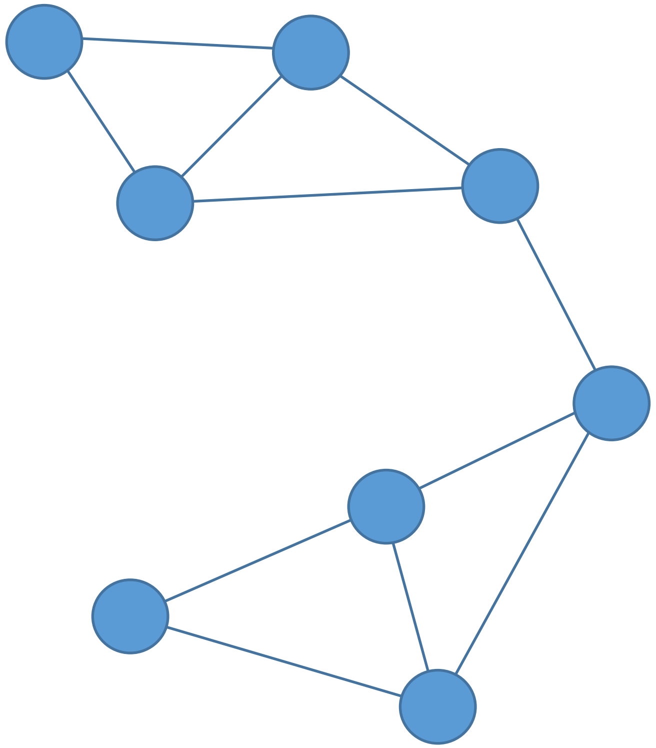
https://en.wikipedia.org/wiki/File:Recurrent_neural_network_unfold.svg

Graph Neural Networks (RNN)

Special about graph data:
- No fixed number of nodes
- Connections different in each graph

GNNs transform node and edge information (some set of quantities associated with these) of an input graph following its individual node connections (**"message passing"**).

No need to fit the graphs to the NN input layer the other way round.



Tomorrow’s lectures

Sebastian Wozniewski — 27.02.23

Morning: Part 2 of Neural Network Building Blocks

- Output layer in regression and classification
- Output usage
- Class weights
- Performance metrics
- Dealing with large datasets

Afternoon: Mastering Model Building

- Activation Functions
- Numerical stability
- Input Normalization
- Overtraining, Regularization
- ...