Vorlesung Machine Learning II

Jonas Pleyer
FDM UNI Freiburg

06.07.2022
SR 404
Math. Institut

Theoretical Foundations

Structure of Neural Networks (Simple)

Definition - Training Set

We have pairs (x_i, y_i) of input/output values and assume that the output y_i was generated by some function f^* depending on x_i. The set of these values is the training set S

$$S = \{(x_i, y_i = f^*(x_i))\}$$

With this definition, we neglect any measurement noise since f^* is assumed to be deterministic.

Definition - Artificial Neural Network (ANN)

An ANN is comprised of an input layer with vector input x_i an output layer with output vector y_j and L hidden layers with activation functions f^k. The layers are connected ascendingly and the input t^{k+1} of the next layer is the weighted sum of output of the previous one

$$t^{k+1} = W^k t^k$$

$$t_i^{k+1} = \sum_{j=1}^{n_k} w_{ij}^k t_j^k$$

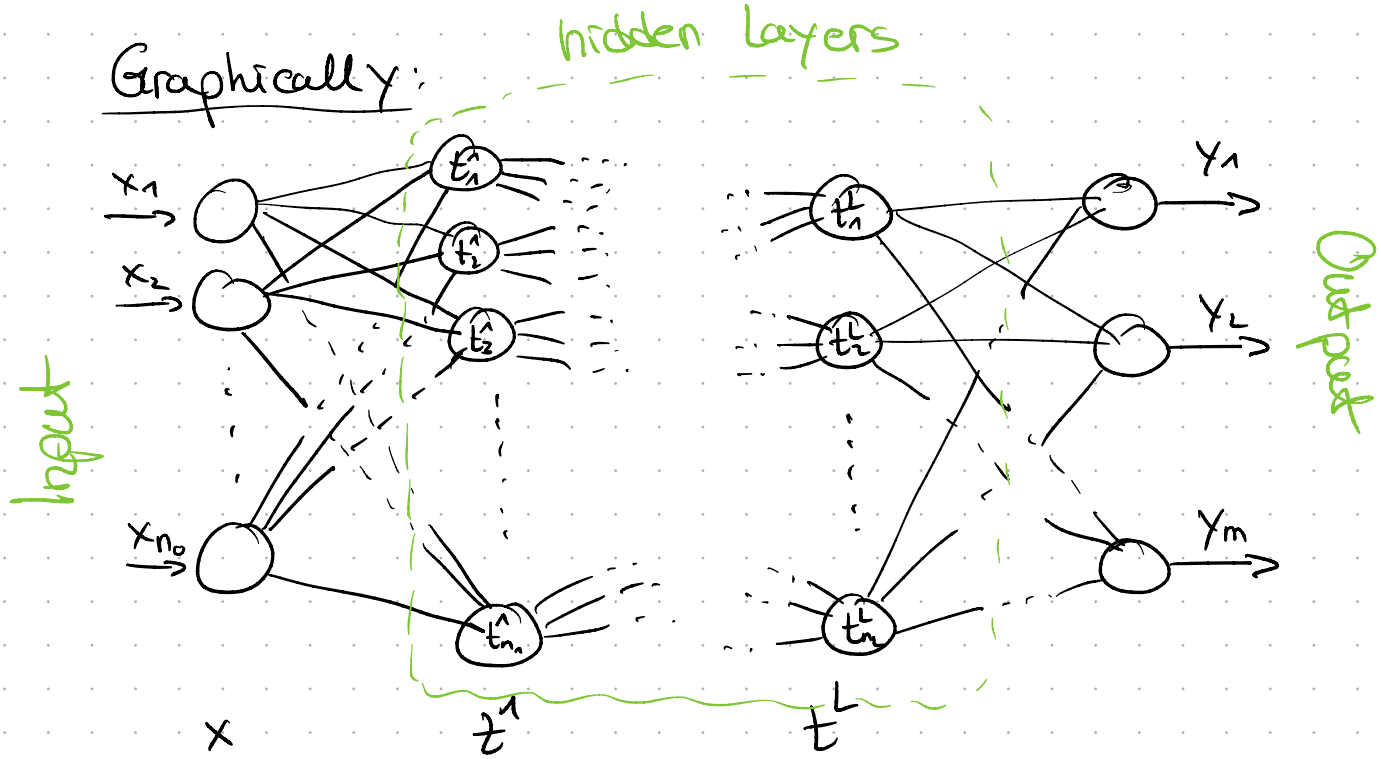
where $t^k = (t^k_1, \dots, t^k_{n_k})$ is a vector and

$$W^k = \begin{pmatrix} w^k_{11} & w^k_{12} & w^k_{13} & \dots \\ w^k_{21} & w^k_{22} & \ddots & \\ \vdots & & & \end{pmatrix}$$

is the matrix containing the weights w^k_{ij} between layer k and $k+1$ and node i of layer $k+1$ and node j of layer k. The full out ut of the network g can be computed via

$$g(x) = f^L(W^L(f^{L-1}(W^{L-1} \dots f^1(W^1(x)) \dots)))$$

Graphically:



Definition — Activation Function

The term "function" is used loosely here. These can contain intermediate state and have non-trivial feedbacks. An activation function f takes a scalar value and produces a scalar value.

Examples

1) Step-function

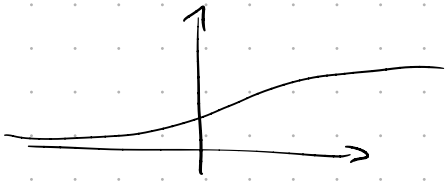
$$f(x) = \begin{cases} 1 & \text{if } x > \text{threshold} \\ 0 & \text{if } x \leq \text{threshold} \end{cases}$$

2) Linear function

$$f(x) = ax$$

3) Sigmoid

$$f(x) = \frac{1}{1+e^{-x}}$$

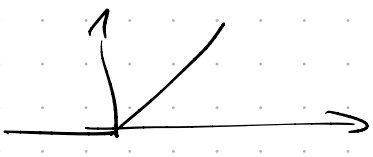


4) Tanh

$$\begin{aligned} f(x) &= 2\,\text{sigmoid}(x) - 1 \\ &= \frac{2}{1+e^{-x}} - \frac{1+e^{-x}}{1+e^{-x}} \\ &= \frac{1-e^{-x}}{1+e^{-x}} \end{aligned}$$

5) ReLU (Rectified Linear Unit)

$$f(x) = \max(0, x)$$



Definition – Loss / Cost function

The loss-function C takes two output values (vectors) and compares them against each other. It returns a non-negative scalar value and is zero if and only if

$$C(y,z) = 0 \iff y = z$$

Examples:

Let $y, z \in \mathbb{R}^m$ be vectors.

1) Choose C such that

$$C(y,z) = \frac{1}{m} \sum_{i=1}^{m} (y_i - z_i)^2$$

the mean-squared error (MSE).

> Comment:
> General purpose cost-function. Is most often chosen.

2) Assume the variables y and z are parameters for polynoms on a interval I. Then we define

$$C(x,y) = \max_{x \in I} \left| \sum_{i=1}^{m} y_i x^i - z_i x^i \right|$$

the supremum norm.

Backpropagation

Questions:

i) How do Neural Networks learn?

ii) How do we adjust parameters?

iii) Are there different techniques?

Definition – Gradient descent

Consider a differentiable function F.
The gradient of F at point x_0 points into the direction of steepest ascend while its negative value points to the largest direction of descent. Starting from the initial x_0, we can obtain the next point x_1 by

$$x_1 = x_0 - \gamma \nabla F(x_0)$$

and thus $F(x_1) \leq F(x_0)$. Iteratively, we obtain

$$x_{n+1} = x_n - \gamma \nabla F(x_n)$$

and hope for convergence as $n \to \infty$.
The parameter γ controls the speed of learning and is thus called the learning rate.

To use the gradient-descent method in ANNs, we need to compute the gradient of the cost-function with respect to the weights in the ANN.

Calculation

$$\frac{\partial C}{\partial w_{ij}^L} = ???$$

we use chain-rule applied to

$$C(y, g(x)) = C(y, f^L(w^L(f^{L-1}(w^{L-1} \dots f^1(w^1(x)) \dots))))$$

We denoted

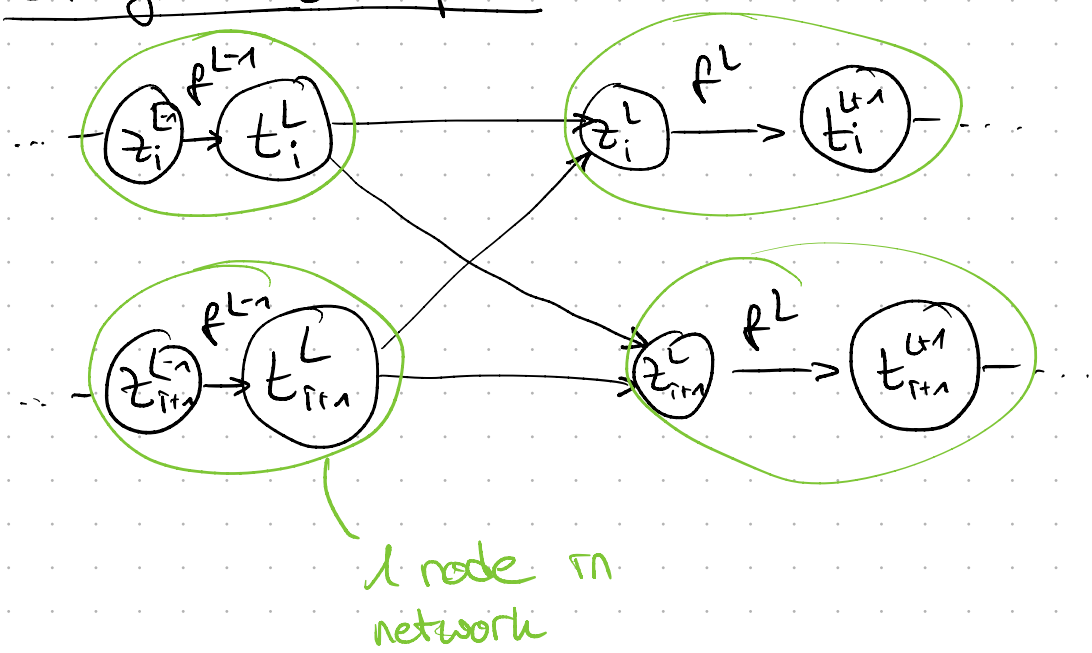
$$\begin{aligned} t^{L+1} &= f^L(w^L t^L) \\ &= f^L(w^L(f^{L-1}(w^{L-1} t^{L-1}))) \\ &\vdots \\ &= f^L(w^L(\dots f^1(w^1 x) \dots)) \end{aligned}$$

as the output of the layer L.

Define the weighted input of each layer

$$z^L = w^L t^L$$

Diagram (Example)



$$z^{l+1} = f^l(z^l_{i+1}) = f^l(w^l t^l)$$

$$z^l = w^l t^l$$

Now we can rewrite the cost-function

$$\nabla_x C(y, g(x))$$

$$= \nabla_x g(x) \cdot \nabla_2 C(y, g(x))$$

scalar

vector vector → matrix

vector vector → vector

$$= \left(\nabla_x z^l \cdot \nabla_{z^l} f^l\right) \nabla_{t^l} C$$

$$= \left(\left(\nabla_x t^l \underbrace{\nabla_{t^l} z^l}_{w^l}\right) \nabla_{z^l} f^l\right) \nabla_{t^l} C$$

$$= \left(\left(\left(\nabla_x z^{l-1} \nabla_{z^{l-1}} f^{l-1}\right) w^l\right) \nabla_{z^l} f^l\right) \nabla_{t^l} C$$

$= \dots$

$= W^1 \nabla_{z^1} f^1 \; W^2 \nabla_{z^2} f^2 \dots \; W^L \nabla_{z^L} f^L \; \nabla_{t^L} C$

Define

$$\delta^L := \nabla_{z^L} f^L \; W^{L+1} \dots \; W^L \nabla_{z^L} f^L \; \nabla_{t^L} C$$

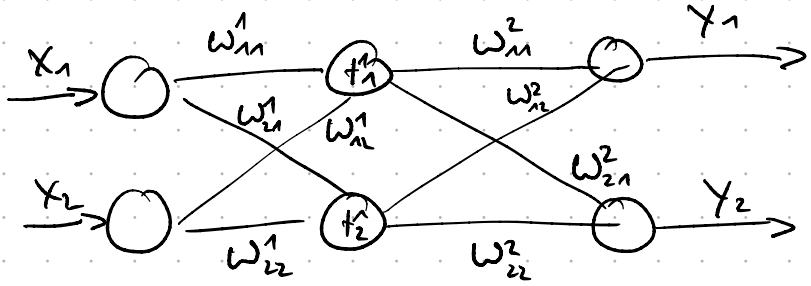
then the gradient at layer L becomes

$$\nabla_{W^L} C = t^{L-1} \delta^L$$

$$\boxed{\delta^{L-1} = \nabla_{z^{L-1}} f^{L-1} \; W^L \delta^L}$$

← backwards-propagation

Example



suppose that $f(x) = ax$

$$C(z, y) = \frac{(z_1 - y_1)^2 + (z_2 - y_2)^2}{2}$$

$g(x) = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}$

$$C(z, g(x)) = \frac{(z_1 - g_1(x))^2 + (z_2 - g_2(x))^2}{2}$$

$$\nabla_y C = \begin{pmatrix} \frac{\partial C}{\partial y_1} \\ \frac{\partial C}{\partial y_2} \end{pmatrix} = \begin{pmatrix} -(z_1 - y_1) \\ -(z_2 - y_2) \end{pmatrix}$$

$$\delta^2 = \underbrace{\nabla_{z^2} f}_{a} \nabla_y C$$

$$= a \nabla_y C = a \begin{pmatrix} y_1 - z_1 \\ y_2 - z_2 \end{pmatrix}$$

$$\Rightarrow \delta^1 = \underbrace{\nabla_{z^2} f}_{} w^2 \delta^2$$

$$= a^2 w^2 \begin{pmatrix} y_1 - z_1 \\ y_2 - z_2 \end{pmatrix}$$

$$= a^2 \begin{pmatrix} w^2_{11} & w^2_{12} \\ w^2_{21} & w^2_{22} \end{pmatrix} \begin{pmatrix} y_1 - z_1 \\ y_2 - z_2 \end{pmatrix}$$

$$= a^2 \begin{pmatrix} w^2_{11}(y_1 - z_1) + w^2_{12}(y_2 - z_2) \\ w^2_{21}(y_1 - z_1) + w^2_{22}(y_2 - z_2) \end{pmatrix}$$

$$\nabla_{w^L} = t^L \delta^L$$

$$\nabla_{w^1} = t^1 \delta^1$$

$$= f(w^1 x) \delta^1$$

$$= a \begin{pmatrix} w^1_{11} & w^1_{12} \\ w^1_{21} & w^1_{22} \end{pmatrix} (\dots)$$

$$= a^3 \begin{pmatrix} w^1_{11}(w^2_{11}(y_1 - z_1)) + w^2_{12}(y_2 - z_2)) + w^1_{12}(\dots) \\ w^1_{21}(w^2_{21}(y_1 - z_1)) + w^2_{22}(y_2 - z_2)) + w^1_{22}(\dots) \end{pmatrix}$$

$$\nabla_{W^2} = t^2 \delta^2$$

$$= f(W^2 \underbrace{f(W^1 x)}) \delta^2$$

already calculated

$$\delta^2 = a \begin{pmatrix} y_1 - z_1 \\ y_2 - z_2 \end{pmatrix}$$