Adversarial Machine Learning

Dr. Nicolas Müller, 09.01.2026

Fraunhofer
AISEC

Table of Contents I

Review: Backpropagation

Machine Learning is datadriven

- Given function space, e.g. $f(x_1, x_2) = w_1 \cdot x_1 + w_2 \cdot x_2$, $w_i \in \mathbb{R}$ and
- a dataset (X, y)

x_1	x_2	**y**
1	2.2	4
-2	4	0
1	1.5	3
5	-2	8
-1	-1	-3,8

- find $f : X \to Y$

Review: Backpropagation

Machine Learning is datadriven

- Given function space, e.g. $f(x_1, x_2) = w_1 \cdot x_1 + w_2 \cdot x_2$, $w_i \in \mathbb{R}$ and
- a dataset (X, y)

x_1	x_2	**y**
1	2.2	4
-2	4	0
1	1.5	3
5	-2	8
-1	-1	-3,8

- find $f : X \to Y$
- possible solution:

$$f(x_1, x_2) \approx 2x_1 + 1x_2$$

Fraunhofer AISEC

Training of Neural Networks

Review: Backpropagation

- Loss Function:
  - Goodness of fit $f(x)$ w.r.t. ground-truth y
  - The smaller, the better
  - Examples: L_1, L_2, (Binary) Cross Entropy, Cosine Similarity Loss, ...
- Example:

x_1	x_2	y	$y_{\mathbf{pred}} = f(x) = 2x_1 + x_2$	$L(x, y) = \|f(x) - y\|$
1	2.2	4		
-2	4	0		
1	1.5	3		
5	-2	8		

- Want to find weights w_i s.t. loss is minimized
- Also called θ

Training of Neural Networks

Review: Backpropagation

- Loss Function:
  - Goodness of fit $f(x)$ w.r.t. ground-truth y
  - The smaller, the better
  - Examples: L_1, L_2, (Binary) Cross Entropy, Cosine Similarity Loss, ...
- Example:

x_1	x_2	**y**	$y_{\mathbf{pred}} = f(x) = 2x_1 + x_2$	$L(x, y) = \lvert f(x) - y \rvert$
1	2.2	4	4.2	0.2
-2	4	0	0	0
1	1.5	3	3.5	0.5
5	-2	8	8	0

- Want to find weights w_i s.t. loss is minimized
- Also called θ

Fraunhofer AISEC

Review: Backpropagation

How to find $\theta^* \in \Theta$, i.e. parameters s.t. loss is minimized?

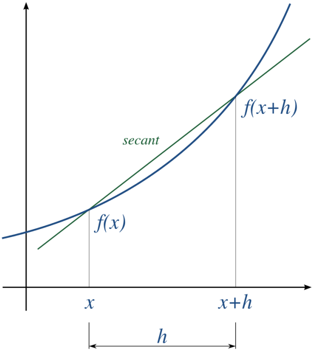
- Random Search over Θ (poor)
- Evoluationary algorithms (works)
- Backpropagation (performant)
  - needs *differentiable* ***f***

Backpropagation

Review: Backpropagation

A function f is *differentiable* if f continuous and $\forall x \in X$ is:

$$f'(x) = \lim_{h \nearrow 0} \frac{f(x+h) - f(x)}{h} = \lim_{h \searrow 0} \frac{f(x+h) - f(x)}{h}$$



Fraunhofer AISEC

Review: Backpropagation

Given:

- Dataset $(\boldsymbol{x}^{(i)}, \boldsymbol{y}^{(i)})$
- Neural Network $\boldsymbol{f}$ with weights θ
- Loss $\boldsymbol{L} : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}$

Task: Solve

$$\theta^* = \arg\min_{\theta} \frac{1}{N} \sum_{i=1}^{N} L\left(f_\theta(x^{(i)}), y^{(i)}\right)$$

Fraunhofer AISEC

Review: Backpropagation

- If f differentiable, can compute *Gradient* over N training samples

$$\nabla_\theta \frac{1}{N} \sum_{i=1}^{N} L\left(f_\theta(x^{(i)}), y^{(i)}\right)$$

- a $dim(\theta)$-dimensional vector
- pointing in the direction of largest increase of $\frac{1}{N}\sum_{i=1}^{N} L$
- Find parameters via iterative update

$$\theta_{t+1} = \theta_t - \alpha \nabla_\theta \frac{1}{N} \sum_{i=1}^{N} L$$

with θ_0 randomly initialized, $\alpha \in \mathbb{R}$ learning rate

Bibliography

Contact Information

Dr. Nicolas Müller

Department
Cognitive Security Technology

Fraunhofer-Institute for
Applied and Integrated Security (AISEC)

Address: Lichtenbergstrasse 11
85748 Garching (near Munich)
Germany
Internet: http://www.aisec.fraunhofer.de

Phone: +49 89 3229986-197
E-Mail: nicolas.mueller@aisec.fraunhofer.de

Fraunhofer AISEC