Westfälische Hochschule
Gelsenkirchen Bocholt Recklinghausen
University of Applied Sciences

Künstliche Intelligenz für Cyber-Sicherheit

- Vorlesung Cyber-Sicherheit -

Prof. Dr. (TU NN)
Norbert Pohlmann

Institut für Internet-Sicherheit – if(is)
Westfälische Hochschule, Gelsenkirchen
http://www.internet-sicherheit.de

if(is) internet-sicherheit.

KI für Cyber-Sicherheit
→ Inhalt

- Ziele und Ergebnisse der Vorlesung
- Einordnung
- Maschinelles Lernen
- Künstliche Neuronale Netze
- Anwendungen KI und Cyber-Sicherheit
- Angriffe auf maschinelles Lernen
- Herausforderungen
- Zusammenfassung

KI für Cyber-Sicherheit
→ Inhalt

- **Ziele und Ergebnisse der Vorlesung**
- Einordnung
- Maschinelles Lernen
- Künstliche Neuronale Netze
- Anwendungen KI und Cyber-Sicherheit
- Angriffe auf maschinelles Lernen
- Herausforderungen
- Zusammenfassung

Ziele und Ergebnisse der Vorlesung
→ KI für Cyber-Sicherheit

- Gutes Verständnis für die **Prinzipien des Maschinellen Lernens**.

- Erlangen der Kenntnisse über verschiedene **Verfahren des Maschinellen Lernen** und der **Künstlichen Intelligenz** und der **Angriffe** auf diese Verfahren.

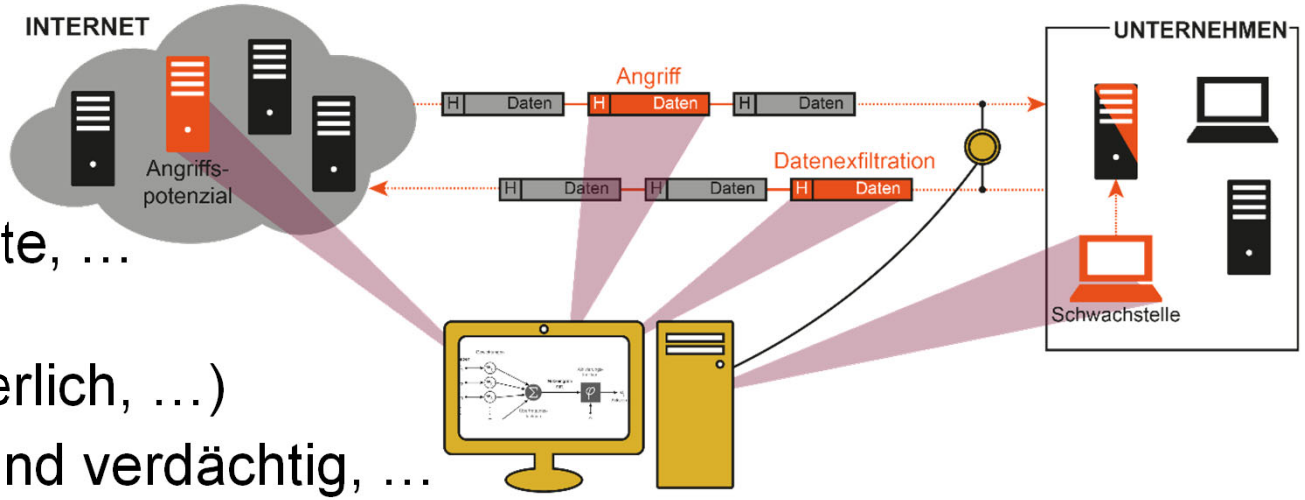
- Gewinnen von praktischen Erfahrungen durch die betrachtung von konkreten **Algorithmen** und **Anwendungen** von **KI in der Cyber-Sicherheit**.

KI für Cyber-Sicherheit
→ Inhalt

- Ziele und Ergebnisse der Vorlesung
- **Einordnung**
- Maschinelles Lernen
- Künstliche Neuronale Netze
- Anwendungen KI und Cyber-Sicherheit
- Angriffe auf maschinelles Lernen
- Herausforderungen
- Zusammenfassung

Künstliche Intelligenz
→ und Cyber-Sicherheit

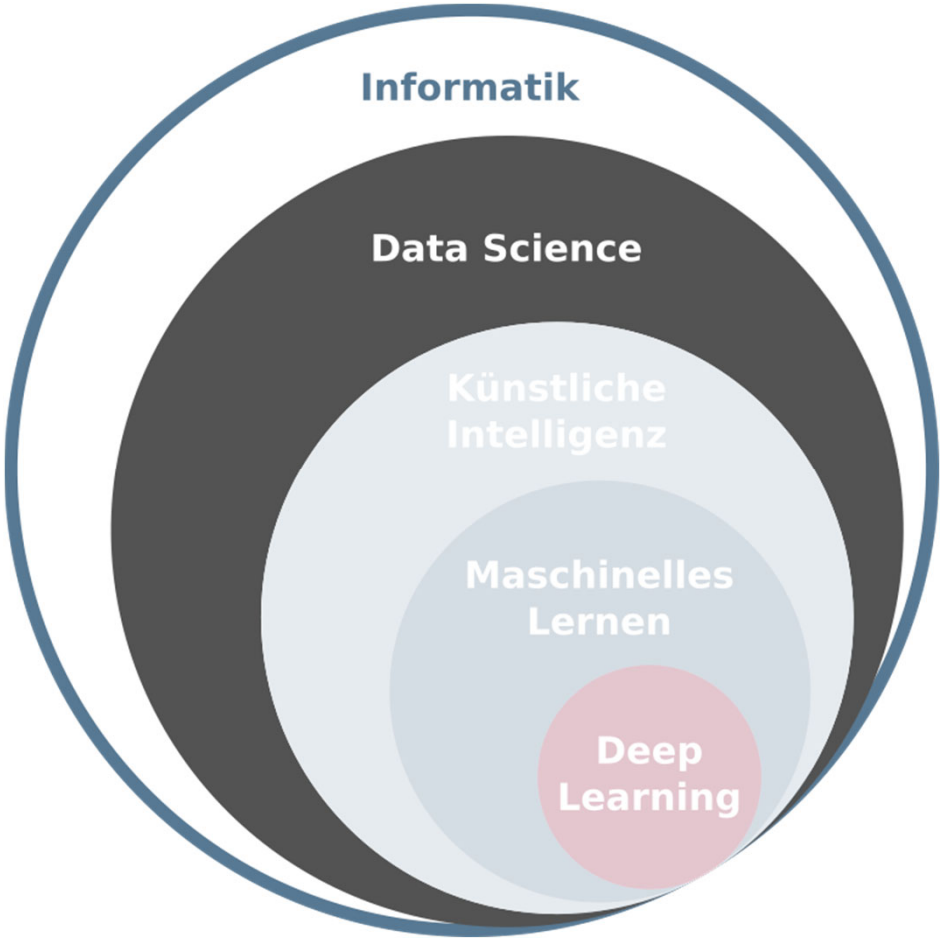
- Erhöhung der **Erkennungsrate** von **Angriffen**
  - Netzwerk, IT-Endgeräte, …
  - adaptive Modelle (selbständig, kontinuierlich, …)
  - Unterschied: normal und verdächtig, …



- **Unterstützung / Entlastung von Cyber-Sicherheitsexperten** *(von denen wir nicht genug haben)*
  - Erkennen von **wichtigen** sicherheitsrelevanten Ereignissen *(Priorisierung)*
  - **(Teil-)Autonomie** bei Reaktionen, … Resilienz, …

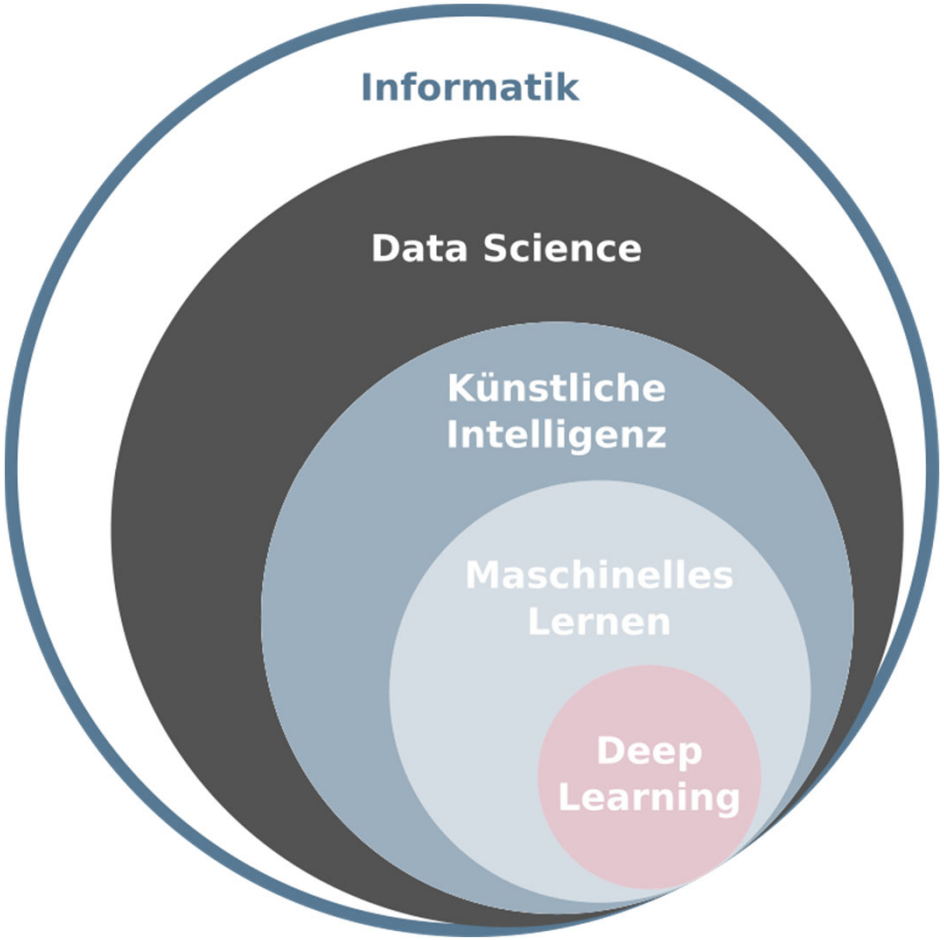
- **Verbesserungen** von bestehenden **Cyber-Sicherheitslösungen**
  - KI leistet einen Beitrag zu einer erhöhten Wirkung und Robustheit
  - Z.B.: Risikobasierte und adaptive Authentifizierung

Einordnung
→ Data Science



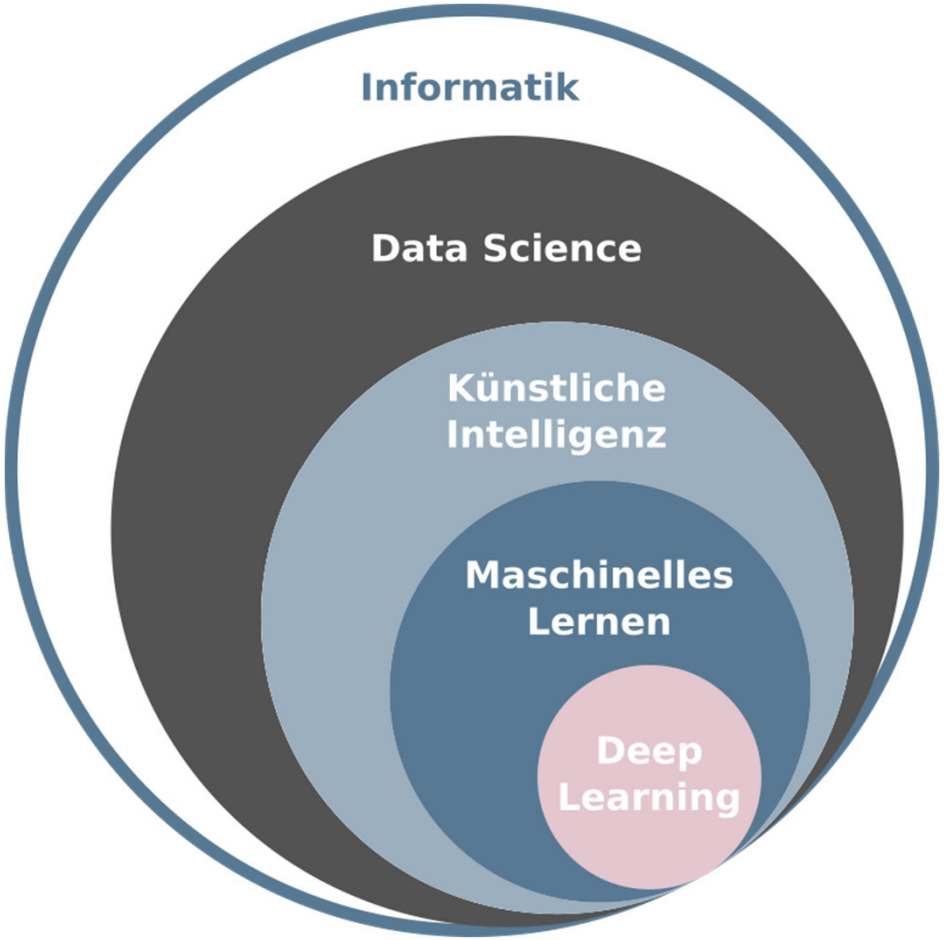
- **Data Science** bezeichnet generell die **Extraktion von Wissen aus Daten**.
- **Da es immer mehr Daten gibt, kann auch immer mehr Wissen daraus abgeleitet werden.** *(Wichtig: Daten müssen Informationen erhalten)*
- Abgrenzung zur künstlichen Intelligenz:
  - Statistiken
  - Kennzahlen
  - Datenerhebung

Einordnung
→ Künstliche Intelligenz



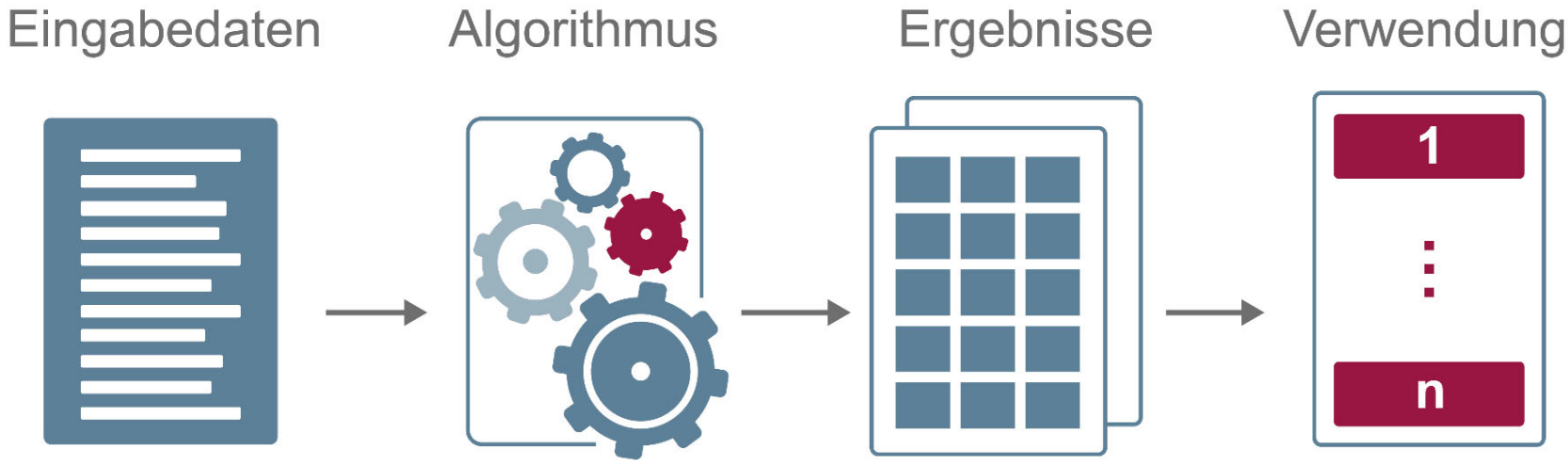
- **Künstliche Intelligenz** ist ein Fachgebiet der Informatik
- setzt intelligentes Verhalten in Algorithmen um
- (Ziel)
  - **automatisiert „menschenähnliche Intelligenz“ nachzubilden**.
  - **Starke „Künstliche Intelligenz“** (Zukunft)
    - Superintelligenz
    - **Singularität** („Maschine“ **verbessert sich selbst**, sind **intelligenter als Menschen**)

Einordnung
→ Maschinelles Lernen



- **Maschinelles Lernen** ist ein Begriff für die „künstliche“ **Generierung von Wissen aus Erfahrung** (in Daten) durch Computer.
- In **Lernphasen** lernen entsprechende ML-Algorithmen aus Beispielen (*alte Daten*) **Muster und Gesetzmäßigkeiten**.
- Daraus erstehende Verallgemeinerungen können auf neue Daten angewendet werden.
- **Schwache „Künstliche Intelligenz“** (wird heute erfolgreich umgesetzt)

Maschinellen Lernens
→ Workflow



Eingangsdaten

Qualität: Inhalt, Vollständigkeiten, Repräsentativität, … Aufbereitung

Algorithmen (ML)

Support-Vector-Machine (SVM), k-Nearest-Neighbor (kNN), … Deep Learning

Ergebnisse

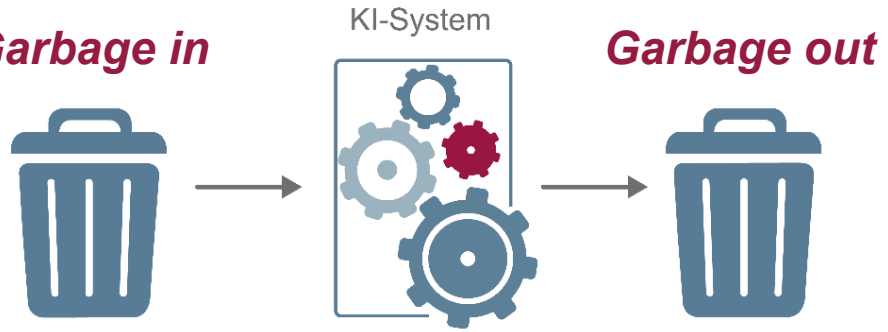
Ergebnisse aus der Verarbeitung (Algorithmus) der Eingangsdaten …

Verwendung

Die Anwendung entscheidet, wie Ergebnisse verwendet werden *(Vertrauen)*.

Vertrauenswürdigkeit
→ Qualität der Daten

Paradigma



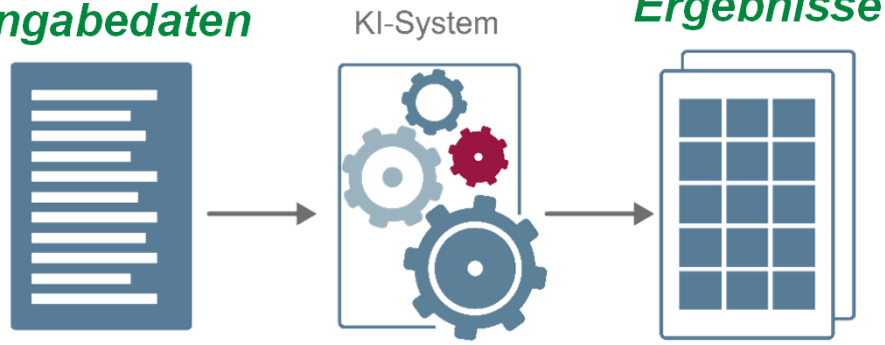
Standards für die Datenqualität:

→ Inhalthöhe der Daten und Korrektheit
→ Nachvollziehbarkeit (Datenquellen)
→ Vollständigkeit und Repräsentativität
→ Verfügbarkeit und Aktualität

Qualitativ hochwertige und sichere **Sensoren** motivieren

hohe Datenqualität der Eingabedaten

qualitative, vertrauenswürdige Ergebnisse



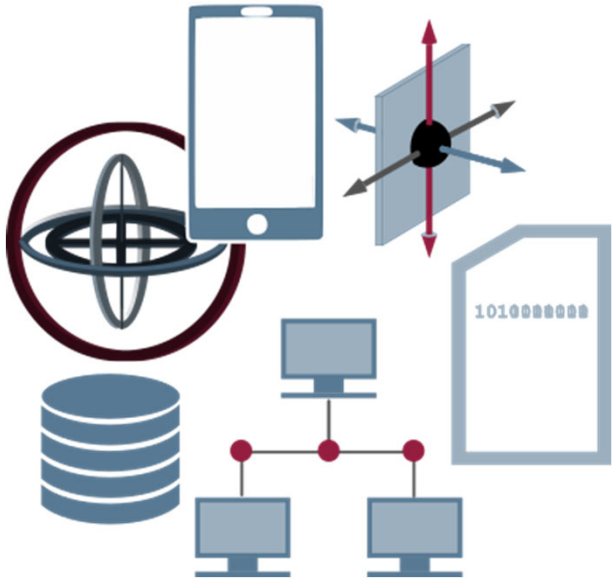
Weitere Aspekte zur Erhöhung der Qualität:

→ Datenpools etablieren
→ Austausch von Daten fördern
→ Interoperabilität schaffen
→ Open Data Strategie puschen

Erfolgsfaktoren – KI / ML
→ Eingabedaten (1/2)

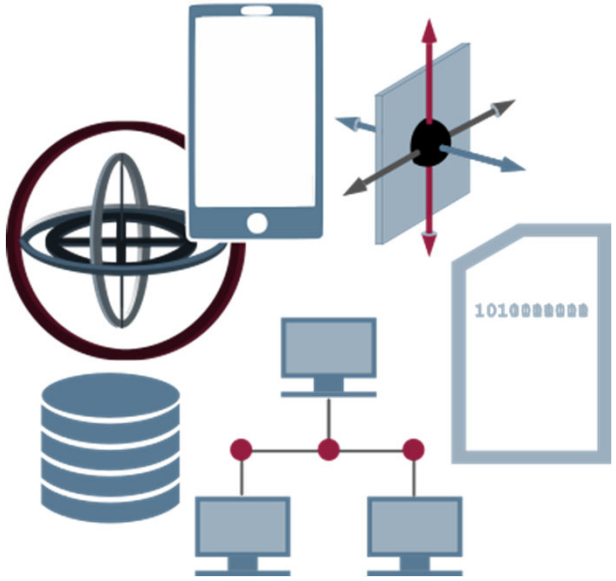
Erfolgsfaktor: Immer **mehr** vorhandene Daten

- **Smartphone, SmartWatch** (körpernah, personenorientiert)
  - Lage- und Beschleunigungssensoren, Nutzereingaben, Benutzerverhalten
- **Computer**
  - Nutzereingaben, Benutzerverhalten, Log Daten



Erfolgsfaktoren – KI / ML
→ Eingabedaten (2/2)

- **Netzwerke, Netzwerkkomponenten (Router, Firewall, …)**
  - Protokolldaten, Log Daten
- **Web-Dienste**
  - Benutzerverhalten, …
- **IoT (Internet of Things)**
  - Sensorik und Aktorik
- **Auto, …**



Erfolgsfaktoren – KI / ML
→ Leistungsfähige IT und Algorithmen

Erfolgsfaktor: **Leistungsfähigkeit** der IT-Systeme

- **enorme Steigerung** (CPU, RAM, …) 20 CPU Kerne, 64 GB Arbeitsspeicher,
  1 TB SSD, usw. Spezial-Hardware: GPUs, FPGA, TensorFlow PU (TPU),…

  … Parallelisierung, Kommunikationsgeschwindigkeiten,
  spezielle Software-Frameworks, …
- **leistungsfähige Cloud-Lösungen**, wie Amazon Web Services, Microsoft Azure, Google Cloud Platform und die IBM Cloud.

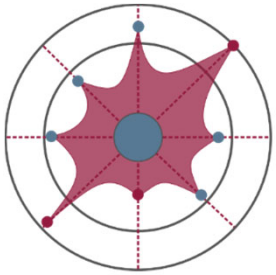
Erfolgsfaktor: **Algorithmen**

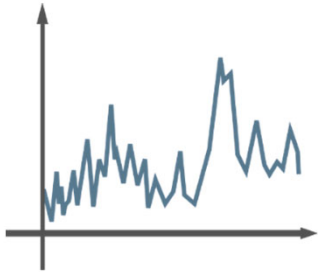
- Immer **bessere Algorithmen** (viel als OpenSource)
- Immer **mehr Erfahrungen** mit dem Umgang
- Immer **einfacherer Zugang** zu den Technologien und Diensten
- Beispiele: Support-Vector-Machine (SVM), k-Nearest-Neighbor (kNN), k-Means-Algorithmus, Hierarchische Clustering-Verfahren, Convolutional Neural Network

Künstliche Intelligenz
→ Ergebnisse und Verwendung

Ergebnisse sind **Modelle** zu den gelernten Eingabedaten

- **Nutzung** der Modelle führt zur konkreten **Anwendung**, z.B.:
  - **Klassifizierung** der Eingangsdaten, zur ***Erkennung von Angriffen***
  - **Numerische Werte**, wie Wahrscheinlichkeiten von ***normalen Verhalten***
  - **Binäre Werte**, wie eine ***erfolgreiche biometrischer Authentifizierung***



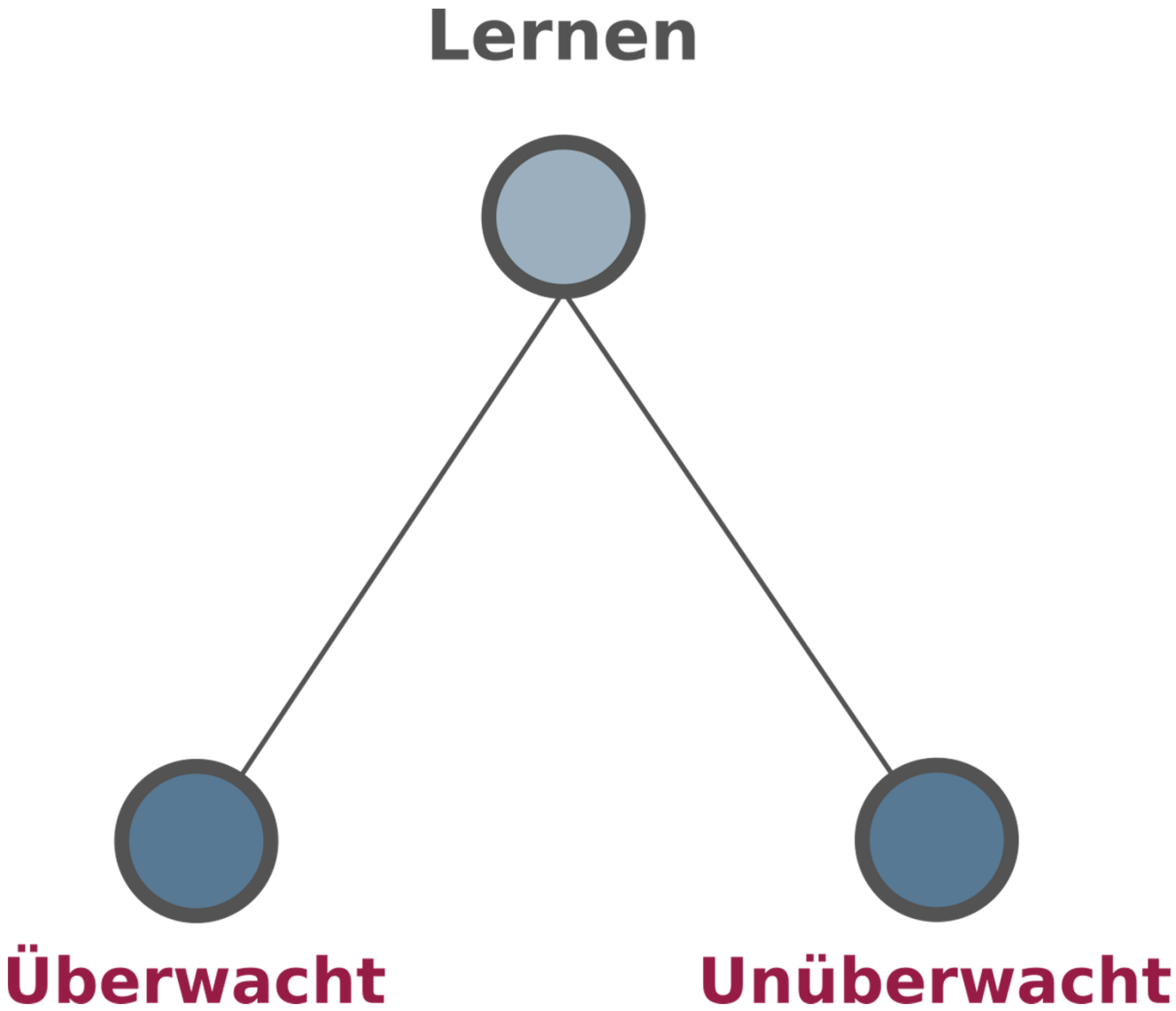


Verwendung: Policy, wie die Ergebnisse genutzt werden sollen.

KI für Cyber-Sicherheit
→ Inhalt

- Ziele und Ergebnisse der Vorlesung
- Einordnung
- **Maschinelles Lernen**
- Künstliche Neuronale Netze
- Anwendungen KI und Cyber-Sicherheit
- Angriffe auf maschinelles Lernen
- Herausforderungen
- Zusammenfassung

Maschinelles Lernen
→ Kategorien des Lernens



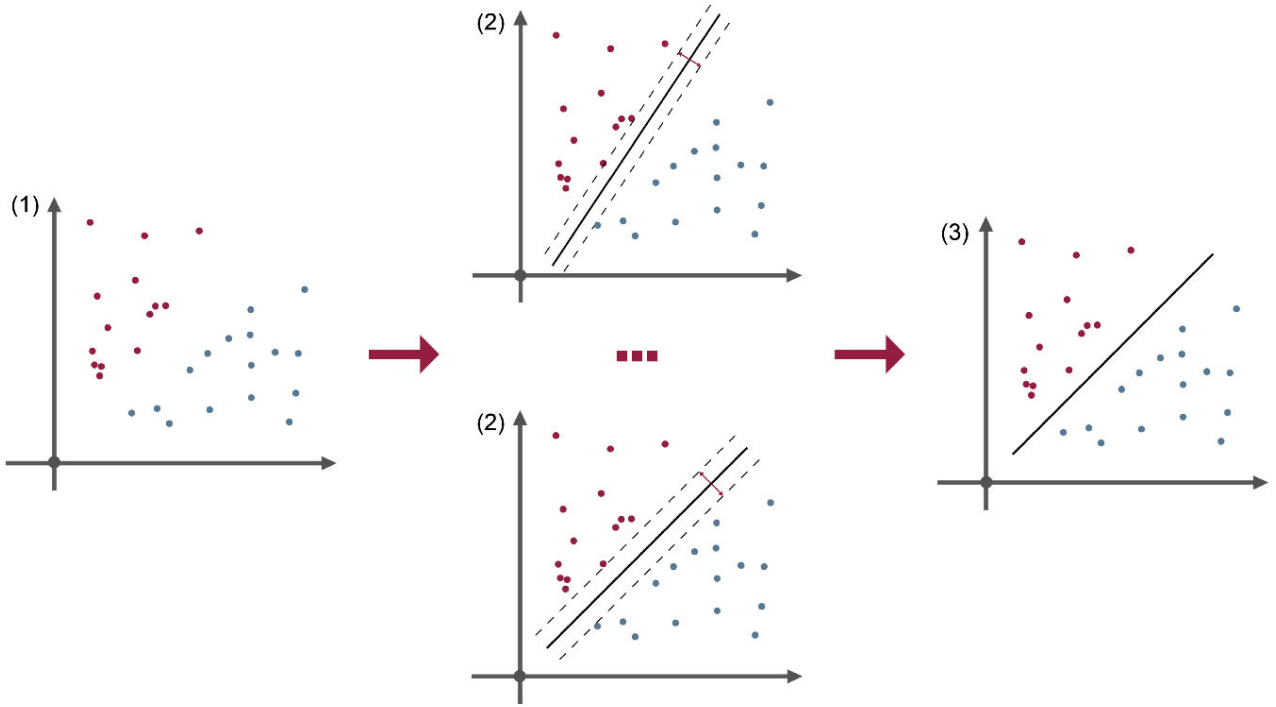
ML-Algorithmus
→ Überwachtes Lernen

- Ziele des überwachten Lernens
  - **Regression:** Vorhersagen von numerischen Werten
  - **Klassifizierung:** Einteilung von Daten in Klassen
- Beispiel: Erkennung von Spam-Mails
- Eingabedaten enthalten **erwartete Ergebnisse**
- **Einteilung der Daten in Trainings- und Testmengen** (*kontinuierlich* lernen)
- Ziel: Selbständig Ergebnisse generieren
- **ML-Algorithmus, z.B.:**
  - Support-Vector-Machine (SVM)
  - k-Nearest-Neighbor (kNN)

ML-Algorithmus

→ Support-Vector-Machine (SVM)/Training

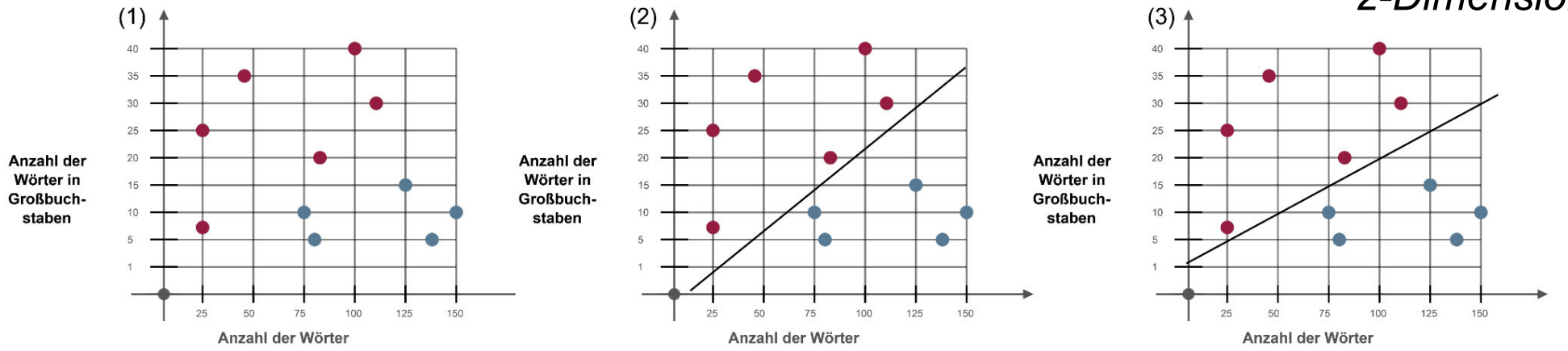
2-Dimensional



- **Input-Daten (1):**
  - bereits klassifizierte **Daten**
  - **Abstandsmaß**
- **ML-Algorithmus (2):**
  - **Ermitteln** von Geraden zur Trennung der Daten
  - **Bewertung** durch Abstand zu den Punkten
  - **Wahl** der Geraden mit maximalem Abstand zu beiden Klassen
- **Output (3):**
  - Gerade als **Modell** zur Klassifizierung

ML-Algorithmus
→ SVM - Beispiel Training (Spam)E-Mail

2-Dimensional



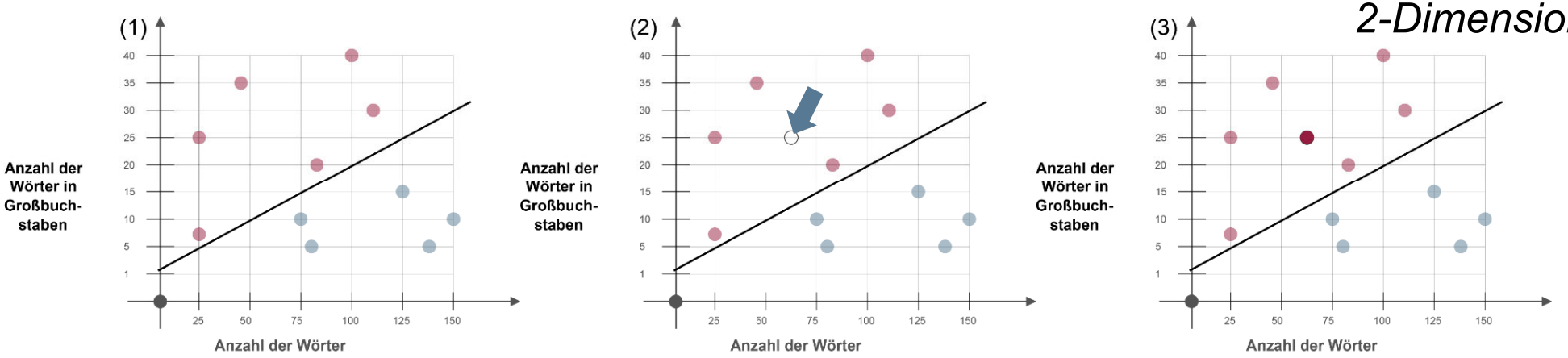
„Wissen aus Erfahrung“

Anzahl Wörter	25	25	47	75	79	82	100	110	125	140	150
Anzahl Wörter in Großbuchstaben	7	25	35	10	5	20	40	30	15	5	10
Spam-E-Mail	**ja**	**ja**	**ja**	**nein**	**nein**	**ja**	**ja**	**ja**	**nein**	**nein**	**nein**

- **Input-Daten (1):**
  - E-Mails mit entsprechender Klassifikation **Spam** / **kein Spam**
- **ML-Algorithmus (2):**
  - **Ermittlung** der Geraden, welche die Daten trennen
  - **Bestimmung** der besten Geraden
- **Output (3):**
  - Gerade als **Modell zur Klassifizierung** von E-Mails als **Spam** / **kein Spam**

ML-Algorithmus
→ SVM - Beispiel Spam - Erkennung

2-Dimensional

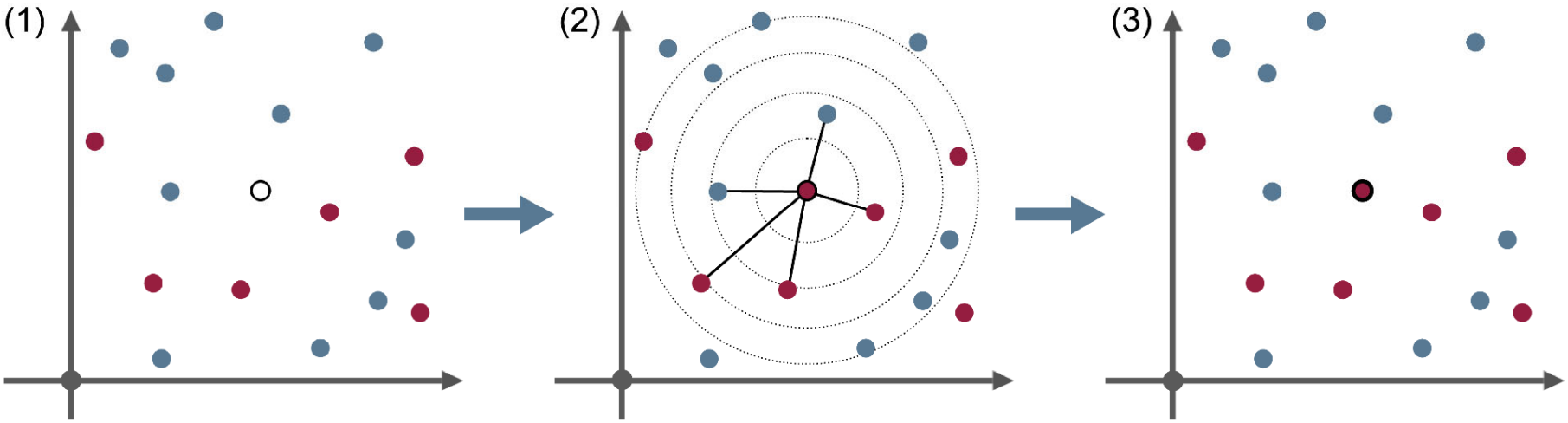


„auf neue Daten anwenden“

Anzahl Wörter	25	25	47	75	79	82	100	110	125	140	150	**63**
Anzahl Wörter in Großbuchstaben	7	25	35	10	5	20	40	30	15	5	10	**25**
Spam-E-Mail	ja	ja	ja	nein	nein	ja	ja	ja	nein	nein	nein	?

- **Input-Daten (1):**
  - **Modell** zur Erkennung von möglichen Spam-Mails
  - **zu beurteilende E-Mail** (z.B.: 63/25)
- **ML-Algorithmus (2):**
  - Berechnung der Lage der zu untersuchenden **E-Mail (63/25)**
- **Output (3):**
  - Lage der Punkte zum Modell klassifiziert die E-Mail als **Spam-Mail**

ML-Algorithmus
→ k-Nearest-Neighbor (kNN)



- **Input-Daten:**
  - Bereits klassifizierte Objekte
  - unklassifiziertes Objekt
  - Anzahl der zu betrachtenden Nachbarobjekte k
- **ML-Algorithmus:**
  - Berechnung der Distanz zu allen anderen Objekten
  - Betrachtung der k nächsten Nachbarobjekte
  - Zuordnung zur am häufigsten vorkommenden Klasse
- **Output:**
  - Klassifizierung des neuen Objekts

ML-Algorithmus
→ kNN – am Beispiel eines IDS (1/7)

- In diesem Beispiel werden die Systemaufrufe und deren Anzahl betrachtet.
  - Die unterschiedlichen Systemaufrufe werden durch kleine Buchstaben repräsentiert, hier „a“ bis „z“.

- Ein Prozess besteht aus einer beliebigen, festen Sequenz von Aufrufen.
  - Die Reihenfolge der Aufrufe wird in diesem Beispiel nicht berücksichtig.
  - Die Häufigkeit jedes Aufrufs wird für jeden normalen Prozess gespeichert.
  - Die Prozesse werden als P_1 bis P_4 dargestellt.
  - Die Sequenz der Vorkommen der Systemaufrufe steht hinter den jeweiligen Prozessen in Klammern.

ML-Algorithmus
→ kNN – am Beispiel eines IDS (2/7)

Prozess \ Systemaufruf	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
P_1("waafwz")	2	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	1
P_2("asdf")	1	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
P_3("axzb")	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
P_4("bbffe")	0	2	0	0	1	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

- Dann wird das Gleichheitsmaß und der Schwellenwert bestimmt, um zu definieren, was „normal“ ist bzw. „nicht normal“.
  - Die Funktion $sim(X, P_i)$ beschreibt das Ähnlichkeitsmaß des unbekannten Prozesses X zu dem jeweiligen bekannten Prozessen P_i.
  - Häufig verwendete Ähnlichkeitsmaße sind die Euklidische Distanzfunktion oder die Kosinus-Ähnlichkeit.

ML-Algorithmus
→ kNN – am Beispiel eines IDS (3/7)

- Die Auswahl oder Erstellung einer geeigneten Funktion für das Maß der Ähnlichkeit muss unter Berücksichtigung der zugrundeliegenden Problemstellung erfolgen.
  - Nicht jede Distanzfunktion ist per se für die Erfüllung einer speziellen Problemstellung geeignet.
  - In diesem Beispiel wird die Kosinus-Ähnlichkeit verwendet.

- Für zwei Vektoren X, P wird die Kosinus-Ähnlichkeit folgendermaßen berechnet:

$$sim(X,P) = \frac{\sum_{i=1}^{n} x_i p_i}{\sqrt{\sum_{i=1}^{n} x_i^2}\sqrt{\sum_{i=1}^{n} p_i^2}}, mit\ x_i, p_i\ Komponenten\ von\ X, P, 1 \leq i \leq n$$

- Die Vektoren X, P und deren Komponenten x_i, p_i ergeben sich in diesem Beispiel direkt aus den Systemaufrufen und deren Anzahl.

ML-Algorithmus
→ kNN – am Beispiel eines IDS (4/7)

- Für einen neuen Prozess $X_A("wasd")$, der analysiert werden soll, wird zuerst die Ähnlichkeit zu allen bereits gelernten Prozessen berechnet.
  - In diesem Beispiel sind die Eingabewerte für die Kosinus-Ähnlichkeit ausschließlich positiv.
  - Aus diesem Grund produziert die Funktion $sim(X, P)$ Ausgabewerte im Bereich von 0 bis 1 (einschließlich).
  - Ein Ausgabewert von 0 bedeutet, dass keine Ähnlichkeit zu einem gelernten Prozess vorliegt.
  - Ein Ausgabewert von 1 signalisiert, dass es sich um die gleichen Prozesse handelt.
  - Je näher der Ausgabewert an 1 liegt, desto ähnlicher sind sich die beiden betrachteten Prozesse.

ML-Algorithmus
→ kNN – am Beispiel eines IDS (5/7)

- Für die Klassifizierung werden die k nächsten Nachbarn mit der geringsten Distanz zu dem neuen Prozess betrachtet.
  - In unserem Beispiel sei der Einfachheit halber $k = 2$.

Prozess \ Systemaufruf	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	$sim(X_A, P_i)$
P_1("waafwz")	2	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	1	0,63
P_2("asdf")	1	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0,75
P_3("axzb")	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0,25
P_4("bbffe")	0	2	0	0	1	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0,00

X_A("wasd")	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	

ML-Algorithmus
→ kNN – am Beispiel eines IDS (6/7)

- Das Ähnlichkeitsmaß der beiden nächsten Prozesse wird in diesem Beispiel **gemittelt** und mit einem **vorher definierten Schwellenwert verglichen**.
  - Wird der Schwellenwert erreicht oder überschritten, wird der betrachtete Prozess als „normal“ Eingestuft.
  - Die Festlegung des Schwellenwertes kann auf vorher durchgeführten Untersuchungen (z.B. mittels Trainings- und Testdaten) oder auf Erfahrungswerten basieren.
  - In diesem Beispiel wurde der **Schwellenwert auf $\mathbf{0,65}$** festgelegt.
- Der gemittelte Wert beträgt $0{,}69$, welcher die Bedingung (**Schwellenwert**) für einen bekannten „normalen“ Prozess erfüllt.

$$\bar{x} = \frac{(0{,}63 + 0{,}75)}{2} = 0{,}69 \geq 0{,}65$$

ML-Algorithmus
→ kNN – am Beispiel eines IDS (7/7)

- Für einen weiteren unbekannten Prozess $X_B("cytq")$ ergeben sich mit der gleichen Vorgehensweise die folgenden berechneten Ähnlichkeiten:

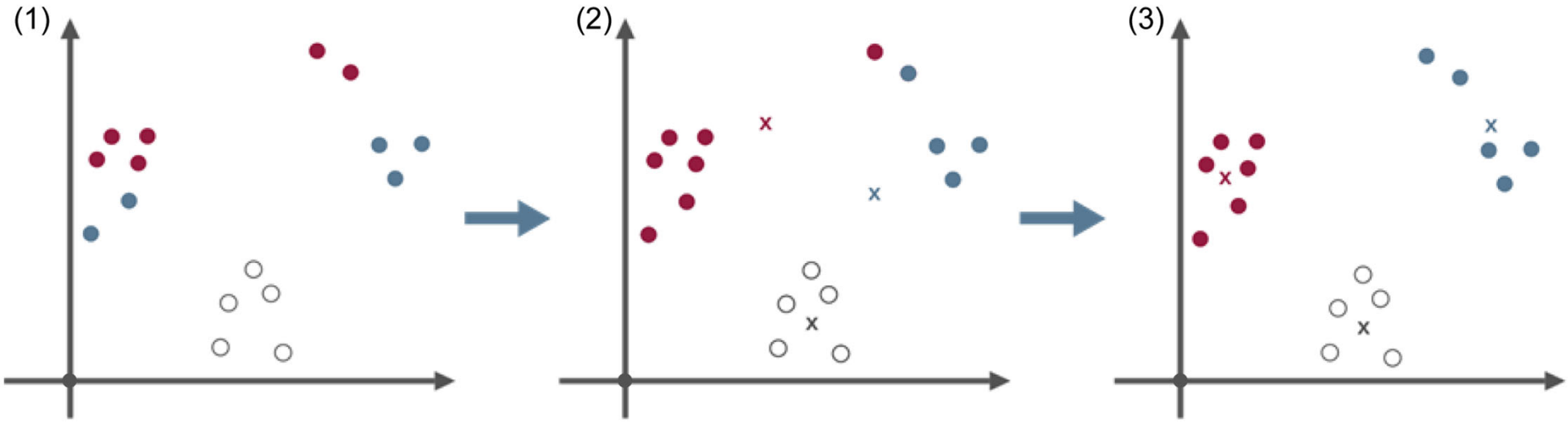
Prozess \ Systemaufruf	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	$sim(X_B, P_i)$
$P_1("waafwz")$	2	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	1	0,00
$P_2("asdf")$	1	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0,00
$P_3("axzb")$	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0,00
$P_4("bbffe")$	0	2	0	0	1	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0,00
$X_B("cytq")$	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	1	0	

- Die Berechnung der Kosinus-Ähnlichkeit hat ergeben, dass der Prozess X_B keinem der bekannten Datensätze ähnelt.
  - Folglich beträgt das arithmetische Mittel in jeder Kombination von zwei Nachbarn 0 und X_B wird als „nicht normal“ klassifiziert.

ML-Algorithmus
→ Unüberwachtes Lernen

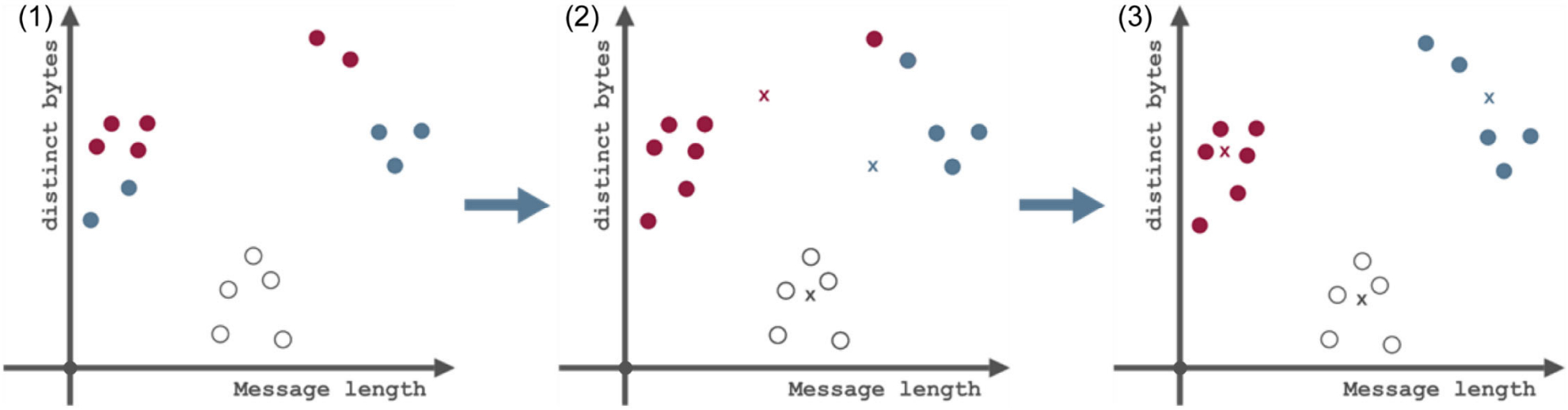
- **Stärke im Suchen nach Mustern in unklassifizierten Daten**
- Erwartungshaltung an diesen Ansatz:
  - Muster erkennen, die vorher **anders nicht greifbar waren** (Komplexität)
- ML-Algorithmus lernt selbstständig
- Klassische Fehler werden in diesem Sinne nicht produziert
- **ML-Algorithmus**
  - Clustering setzt ähnliche Datengruppen miteinander in Verbindung, z.B.:
    - k-Means-Algorithmus
    - Hierarchische Clustering-Verfahren
- **Problem:** Lernt der ML-Algorithmus in die gewünschte Richtung?

ML-Algorithmus
→ k-Means-Algorithmus



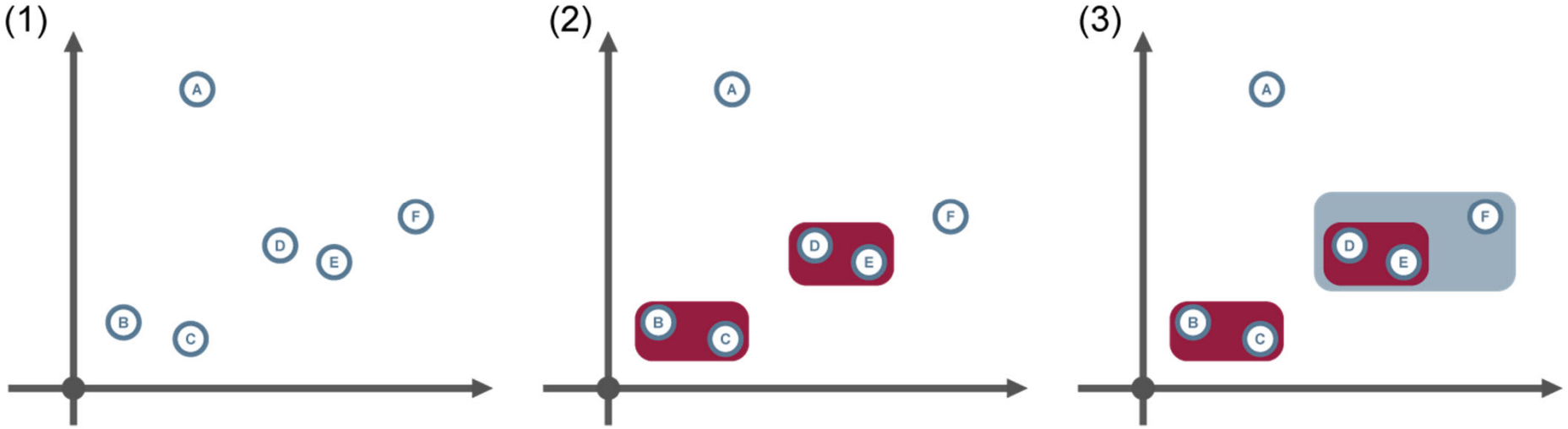
- **Input-Daten:**
  - beliebige Daten
  - Abstandsmaß
  - Anzahl k Cluster
  - Initiale Zuordnung der Elemente zu Clustern (z.B. zufällig)
- **ML-Algorithmus:**
  - Berechnung der **Schwerpunkte** (Zentroide)
  - Zuordnung der Elemente zu Cluster mit dem nächsten Zentroid
  - Neuberechnung der Zentroide und erneute Zuordnung
- **Output:**
  - **Einteilung** der Objekte in **k Cluster**

ML-Algorithmus
→ k-Means-Algorithmus - Beispiel



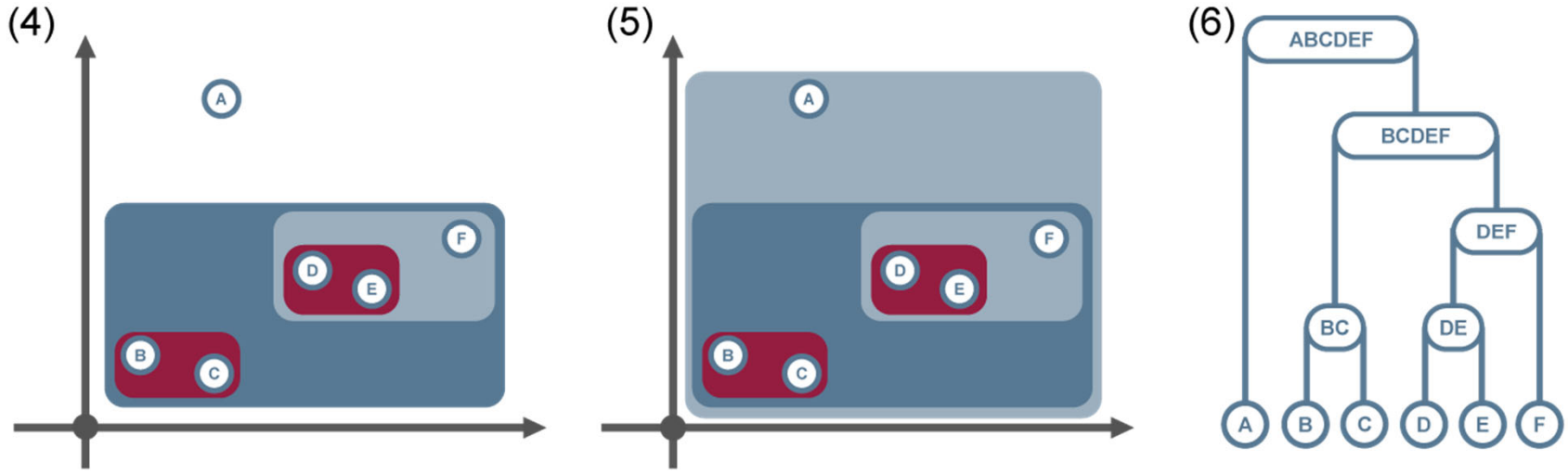
- **Input-Daten (1):**
  - Daten von Malware *(Palevo, Virut, Mariposa)*
  - Abstandsmaß
  - k = 3
  - Initiale Zuordnung nach Message length, distinct bytes
- **ML-Algorithmus (2):**
  - Berechnung der Durchschnitte
  - Zuordnung der Elemente zur Malwareart mit dem nächsten Zentroid
  - Neuberechnung der Zentroide und erneute Zuordnung
- **Output (3):**
  - Einteilung der Malware in die drei Malwarearten
    - Rot = Virut
    - Weiß = Palevo
    - Blau = Mariposa

ML-Algorithmus
→ Hierarchische Clustering-Verfahren (1)



- **Input-Daten (1):**
  - beliebige Daten
  - Ähnlichkeitsmaß
- **ML-Algorithmus (2 bis 5):**
  - jeder Datenpunkt ist ein eigenes Cluster
  - ähnlichste Cluster werden zuerst zusammengeführt
  - entstandene Cluster werden erneut als Eingabedaten verwendet
  - iteratives Zusammenführen der Cluster induziert eine hierarchische Struktur

ML-Algorithmus
→ Hierarchische Clustering-Verfahren (2)

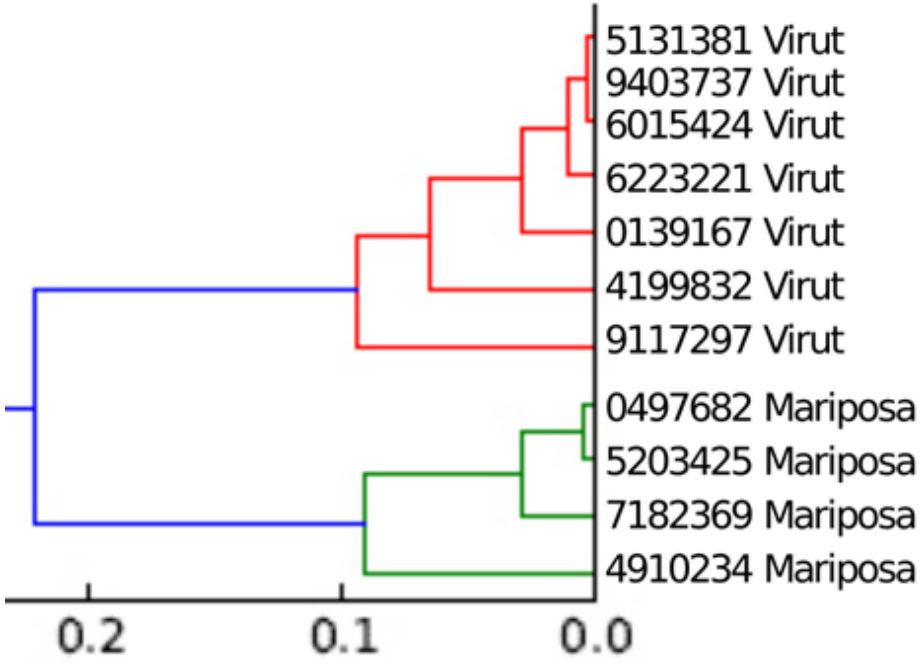


- **Output (6):**
  - Hierarchische Beziehungen zueinander in Form eines Binärbaums (Dendrogramm)

ML-Algorithmus

→ Hierarchische Clustering-Verfahren: Beispiel

- Clustering der Daten aus Botnet-Analyse
- Anwendung einer komplexen Distanzfunktion (Wertebereich [0, 1])
- Trennung der Familien-Cluster bei Distanz von ca. 0.1
- Einordnung der Daten in zwei Malware-Familien Virut und Mariposa

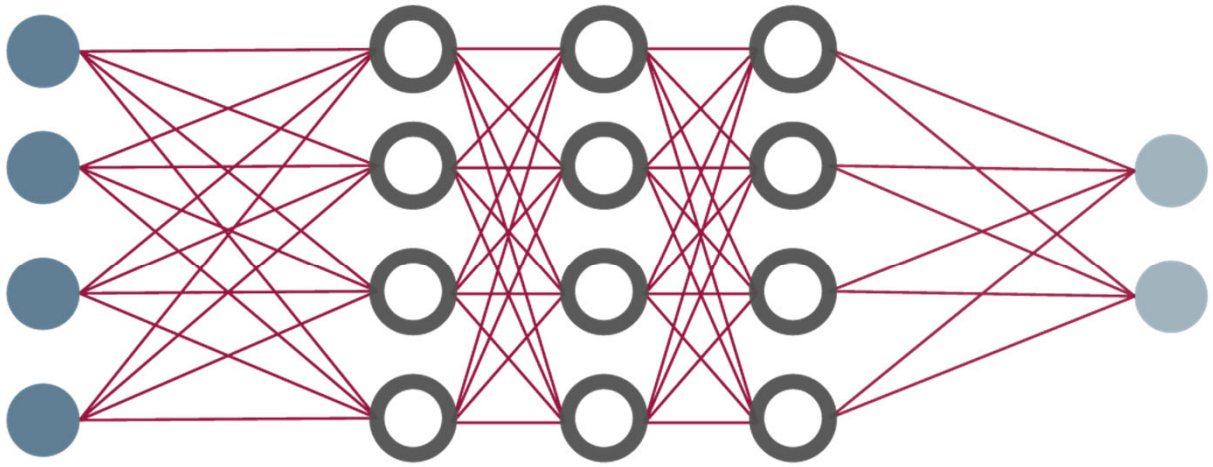


KI für Cyber-Sicherheit
→ Inhalt

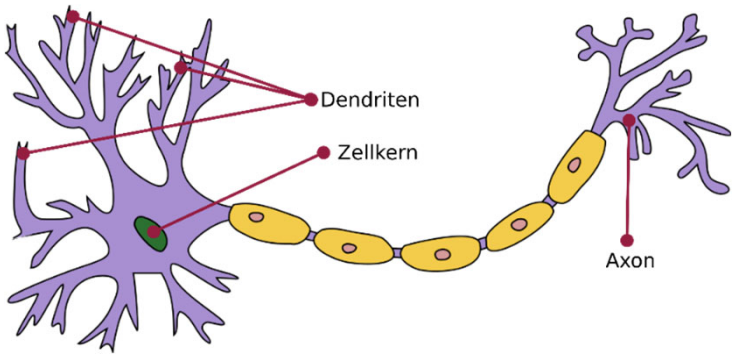
- Ziele und Ergebnisse der Vorlesung
- Einordnung
- Maschinelles Lernen
- **Künstliche Neuronale Netze**
- Anwendungen KI und Cyber-Sicherheit
- Angriffe auf maschinelles Lernen
- Herausforderungen
- Zusammenfassung

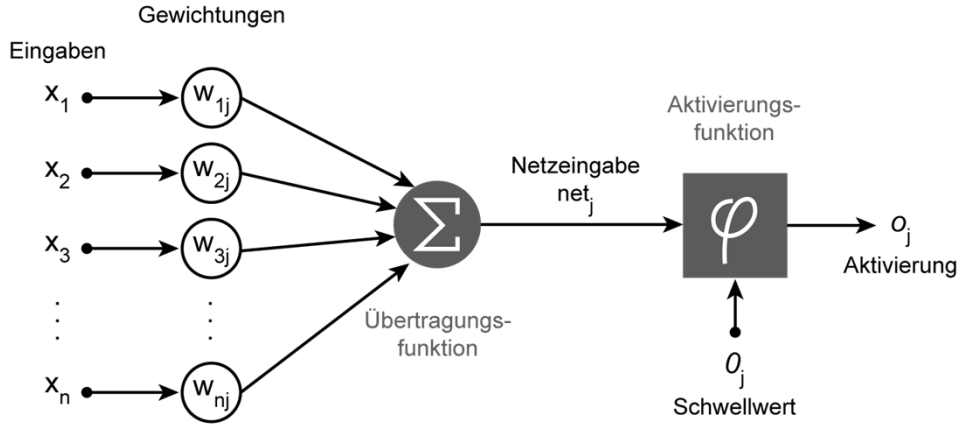
Künstlich Neuronale Netze
→ Netze aus künstlichen Neuronen (1/2)

- Vorlage ist die die biologische Struktur des Gehirns/Neurons
- Nutzen Gewichte und mathematische Funktionen (für die Informationsverarbeitung)
- Informationsverarbeitung über mehrere miteinander verbundene Schichten aus künstlichen Neuronen



Künstlich Neuronale Netze
→ Netze aus künstlichen Neuronen (2/2)

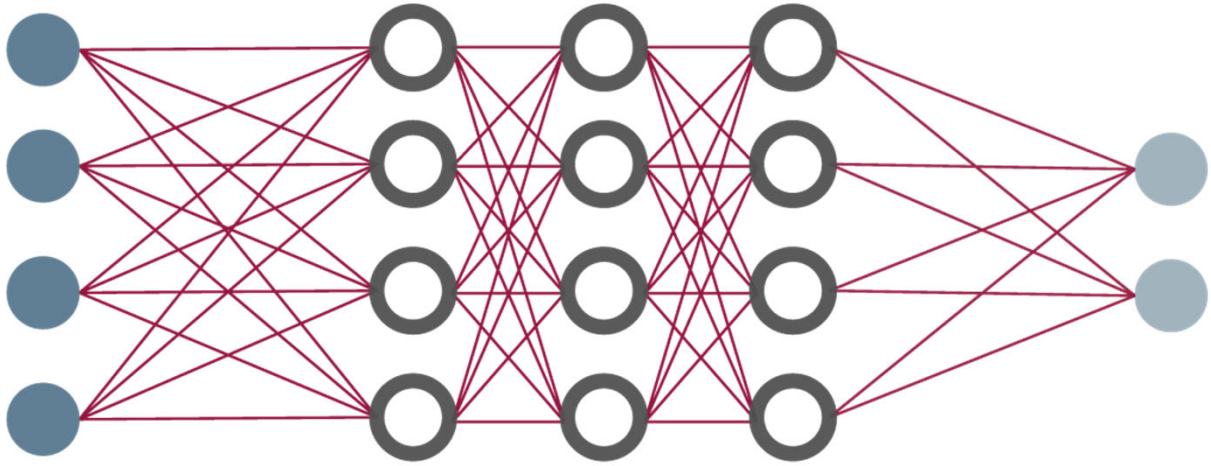




- **Biologisches Neuron:**
  - Dendriten:
    - Reizaufnahme (Signaleingang)
  - Axon:
    - Leitet die Informationen weiter (Signalausgang)
  - Zellkern:
    - Reizverarbeitung (Signalverarbeitung)

- **Künstliches Neuron:**
  - Übertragungsfunktion:
    - Berechnet anhand der Summe der Wichtungen, der Eingaben, die Netzeingabe
  - Aktivierungsfunktion/ Ausgabefunktion:
    - Ausgabe der Information
  - Schwellenwert:
    - Wert eines Reizes, bei dem das Neuron aktiviert wird

Künstlich Neuronale Netze
→ Schichten in einem KNN

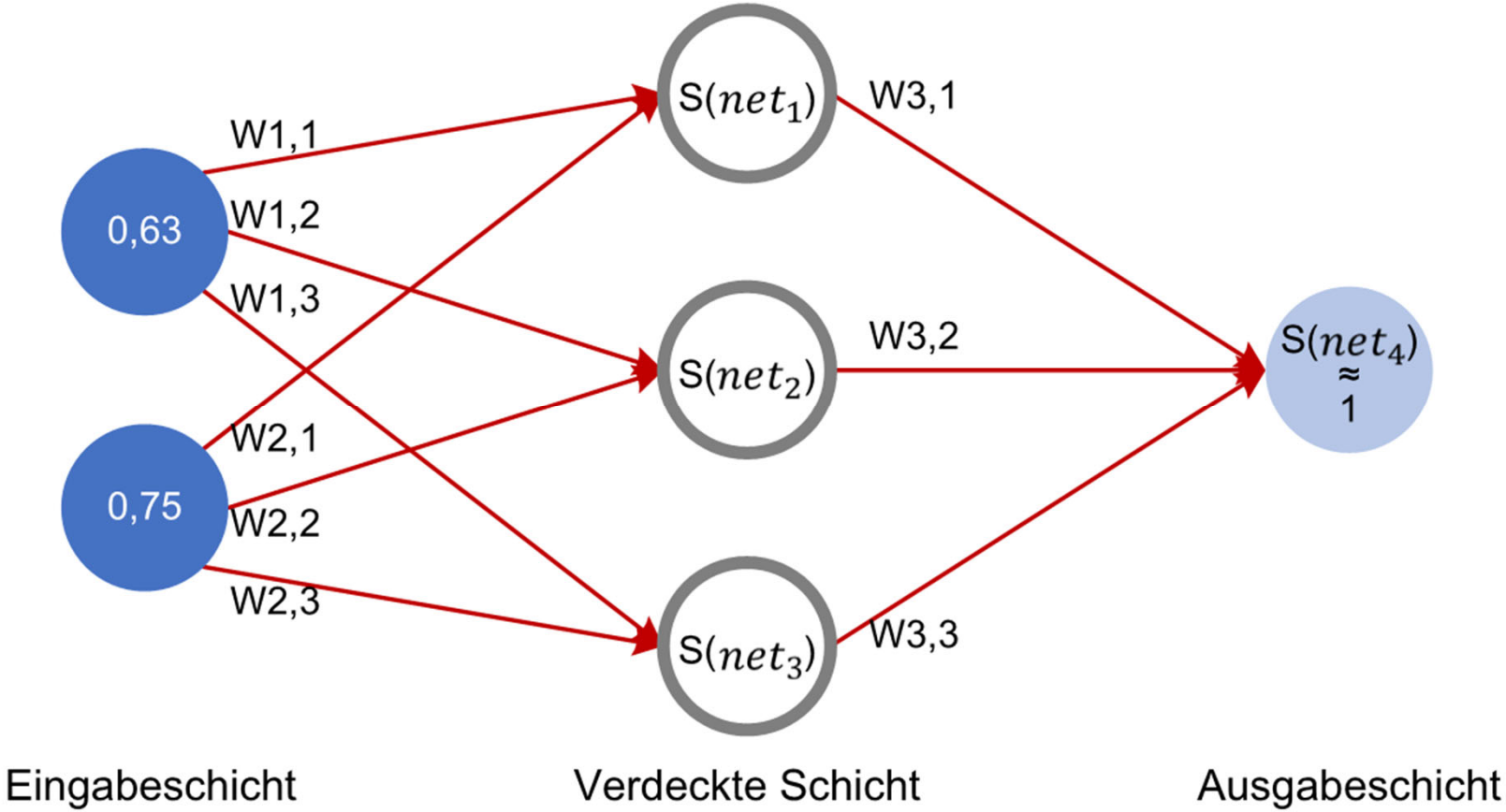


- **Eingabeschicht:**
  - Eingabeneuronen *(z.B. Ohren, Retina oder Haut)*
  - Eingabedaten werden in geeignete Repräsentation überführt
- **Verdeckte Schichten:**
  - Je nach Komplexität der Aufgabe 1-N verknüpfte Neuronen
  - Erkennung von simplen Mustern und Strukturen
  - Mit jeder Schicht werden immer komplexere Merkmale herausgefiltert
- **Ausgabeschicht:**
  - Ausgabe sämtlicher möglicher Repräsentationen der Ergebnisse

Künstlich Neuronale Netze
→ KNN-Beispiel (1/9)

- Nachfolgend wird anhand eines Rechenbeispiels dargestellt, wie ein KNN zu den bereitgestellten Ein- und Ausgabedaten ein Modell in mehreren Evaluationsrunden erstellt.
  - Als Eingabe werden die zwei höchsten Ähnlichkeitsmaße der verschiedenen Prozessaufrufe aus „kNN – am Beispiel eines IDS“ verwendet.
  - Basierend auf den Ähnlichkeitsmaßen soll das erzeugte KNN berechnen, ob ein Prozess „normal“ ist, also im Sinne der Cyber-Sicherheit ungefährlich ist.
  - In diesem Beispiel wird der Einfachheit halber nur ein Ausgabewert betrachtet.
  - Wenn das KNN eine 1 ausgibt, dann wird ein Prozess als „normal“ betrachtet.

Künstlich Neuronale Netze
→ KNN-Beispiel (2/9)



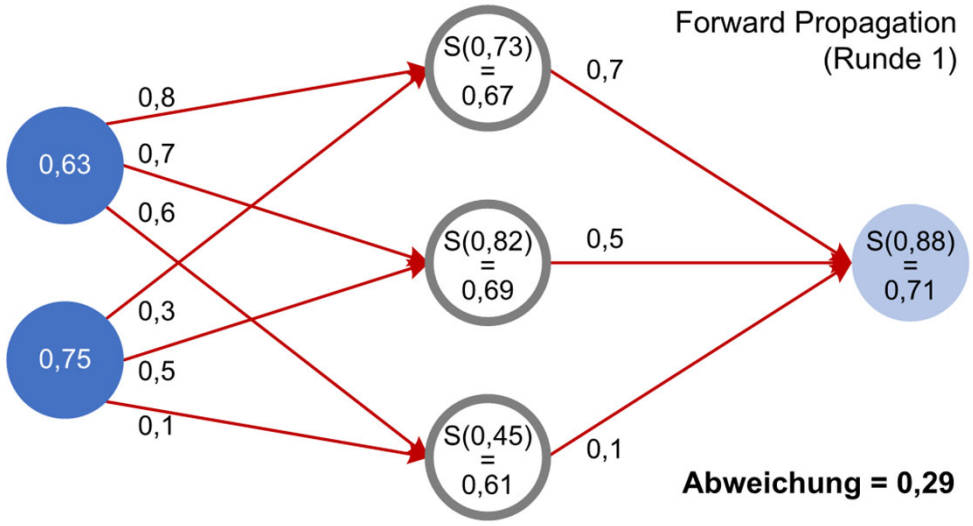
Künstlich Neuronale Netze
→ KNN-Beispiel (3/9)

- Die Berechnungen innerhalb des KNN lassen sich grundsätzlich in zwei Phasen unterteilen.
  - In der ersten Phase werden die Berechnungen von der Eingabeschicht in Richtung der Ausgabeschicht durchgeführt (**Forward Propagation**).
  - Abweichungen im daraus resultierenden Ergebnis werden anschließend durch eine rückwärts gerechnete Anpassung der Kantengewichte minimiert (**Back Propagation**).
- Nachdem die Kantengewichte angepasst wurden, werden die beiden Phasen erneut durchlaufen.

Künstlich Neuronale Netze
→ KNN-Beispiel (4/9)

- Diese Vorgehensweise wird so lange wiederholt, bis das Ergebnis in der Ausgabeschicht möglichst genau approximiert wurde.
  - Abhängig von der konkreten Problemstellung können mehrere tausend Runden nötig sein.
  - In diesem einfachen Rechenbeispiel werden nur zwei vorwärts gerichtete und eine rückwärtsgerichtete Runde betrachtet.

- In der ersten vorwärts gerichteten Runden wurden zufällige Kantengewichte gewählt.



Künstlich Neuronale Netze
→ KNN-Beispiel (5/9)

- Als Aktivierungsfunktion wird in diesem Beispiel die Sigmoidfunktion verwendet.

$$S(t) = \frac{1}{1 + e^{-t}}$$

- Bei der Forward Propagation werden die Netzeingaben für jedes Neuron auf Basis der Eingabewerte und den entsprechenden Kantengewichten folgendermaßen berechnet:

$$net_j = 0{,}63 * W1, j + 0{,}75 * W2, j, 1 \leq j \leq 3$$

$$net_4 = S(net_1) * W3{,}1 + S(net_2) * W3{,}2 + S(net_3) * W3{,}3$$

Künstlich Neuronale Netze
→ KNN-Beispiel (6/9)

- Die erste Forward Propagation hat eine Abweichung von 0,29 ergeben.
  - Diese Abweichung berechnet sich aus der Differenz von dem gewollten Ausgabewert (in diesem Beispiel der Wert 1 für einen „normalen“ Prozess) und dem aktivierten Ausgabewert des KNNs.

$$\text{Abweichung} = 1 - S(net_4)$$

- Diese Abweichung wird nun zurück gerechnet, damit die Kantengewichte entsprechend angepasst werden können.
  - In diesem Beispiel wird die folgende Ableitung der Sigmoidfunktion für die benötigte Änderungsrate der Kantengewichte verwendet:

$$S'(t) = \mathrm{S(t)} * (1 - \mathrm{S(t)})$$

Künstlich Neuronale Netze
→ KNN-Beispiel (7/9)

- Die konkrete Änderungsrate wird dann wie folgt berechnet:

$$\Delta = S'(net_4) * Abweichung$$

- Die neuen Kantengewichte zwischen der verdeckten Schicht und der Ausgabeschicht werden mit der folgenden Formel berechnet:

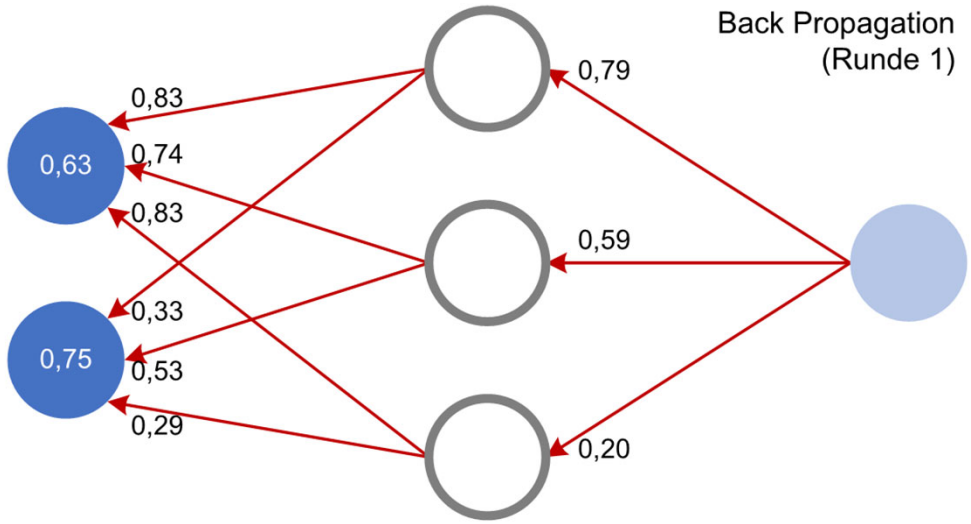
$$W3, \mathrm{j}_{new} = \mathrm{W3, j} + \frac{\Delta}{S(net_j)} 1 \leq j \leq 3$$

- Alle neuen Kantengewichte zwischen der Eingabeschicht und der verdeckten Schicht lassen sich nun folgendermaßen berechnen:

$$Wi, j_{new} = \frac{\Delta}{W3, j} * S'(net_j), 1 \leq i \leq 2{,}1 \leq j \leq 3$$

Künstlich Neuronale Netze
→ KNN-Beispiel (8/9)

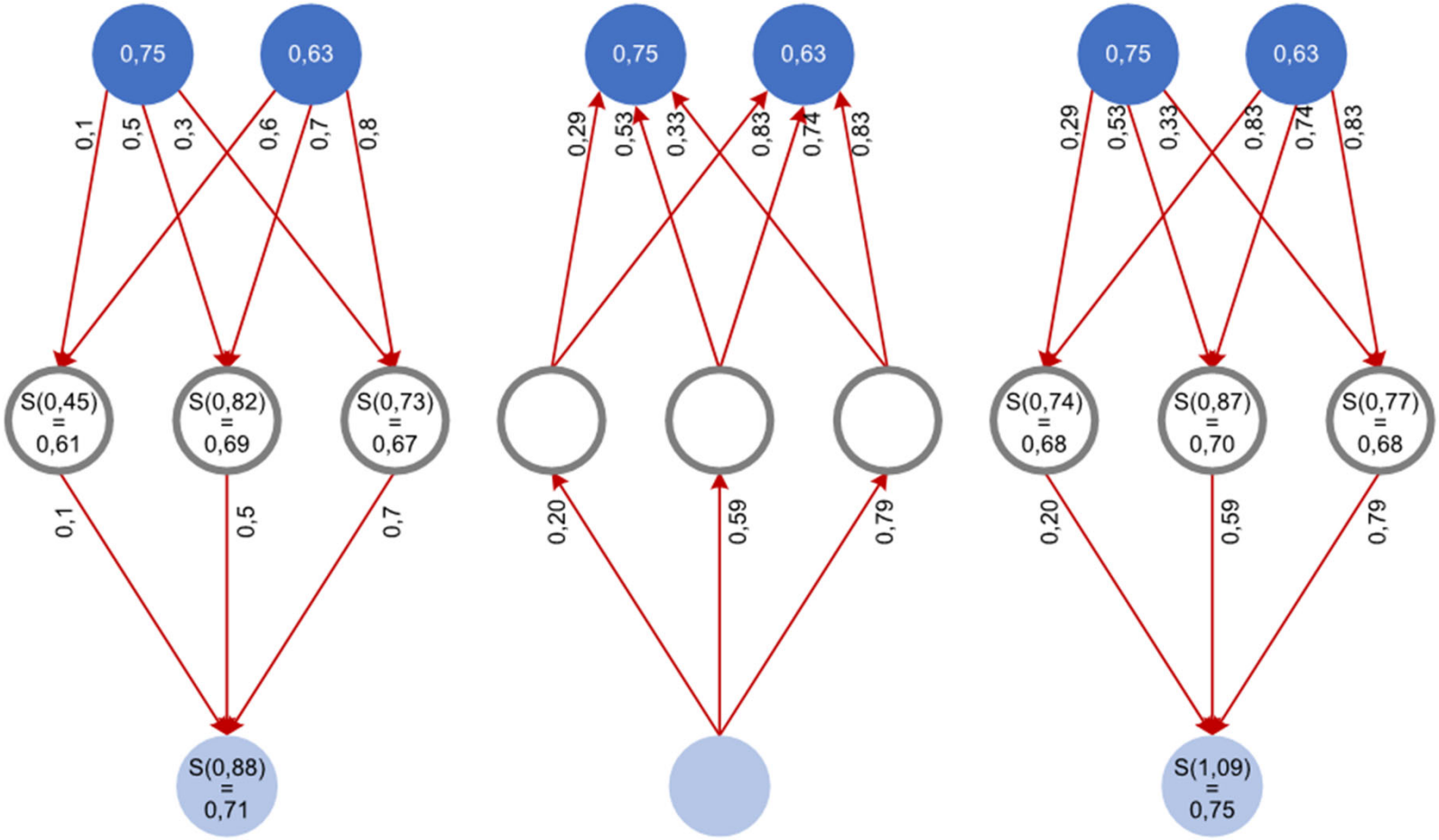
- Folgendes Ergebnis ergibt sich für die neu berechneten Kantengewichte:



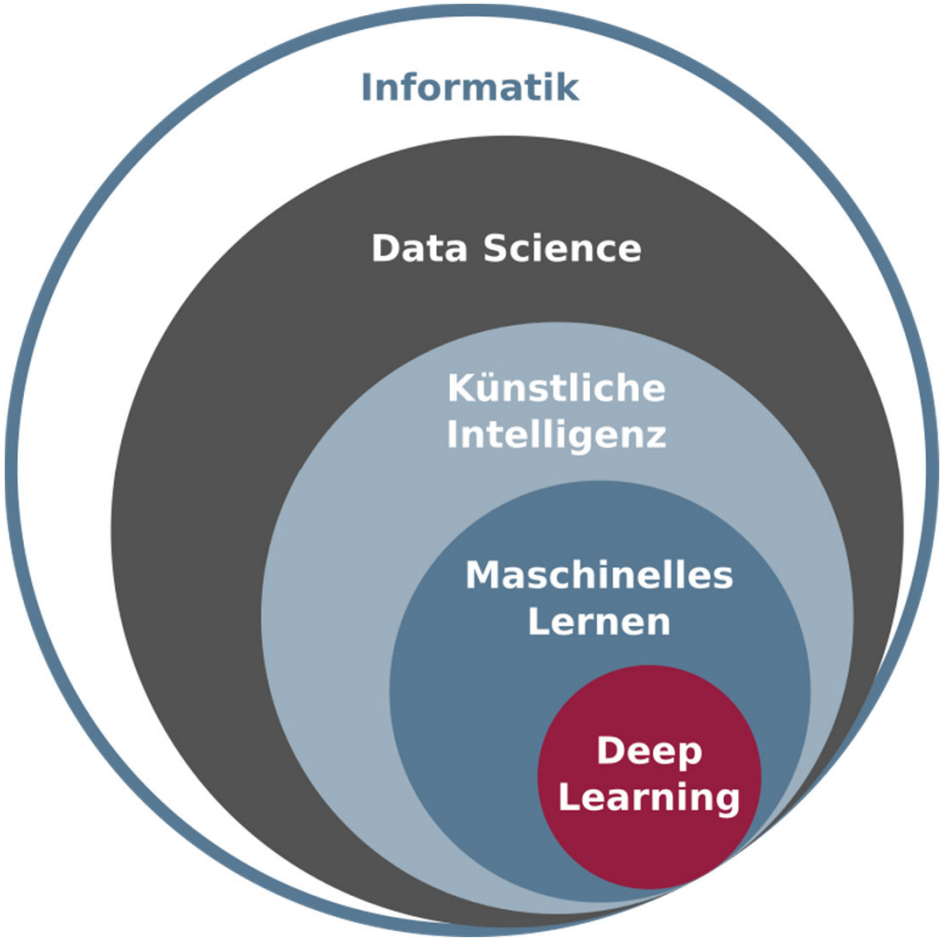
- Mit den neu berechneten Kantengewichten kann nun eine erneute Forward Propagation durchgeführt werden.

- In der zweiten Runde kann festgestellt werden, dass die Abweichung auf 0,25 reduziert werden konnte.

Künstlich Neuronale Netze
→ KNN-Beispiel (9/9)



Einordnung
→ Deep Learning



- Maschinelles Lernen wird noch effektiver durch:
  - **Deep Learning**
- Deep Learning ist eine Spezialisierung des maschinellen Lernens
- *Nutzt vorwiegend neuronale Netze*
  - ***Erlaubt unvollständige Daten***
  - ***Erlaubt Rauschen und Störungen***
- Kommt dem „menschlichen Gehirn“ am nächsten

Deep Learning
→ Architekturen (1/2)

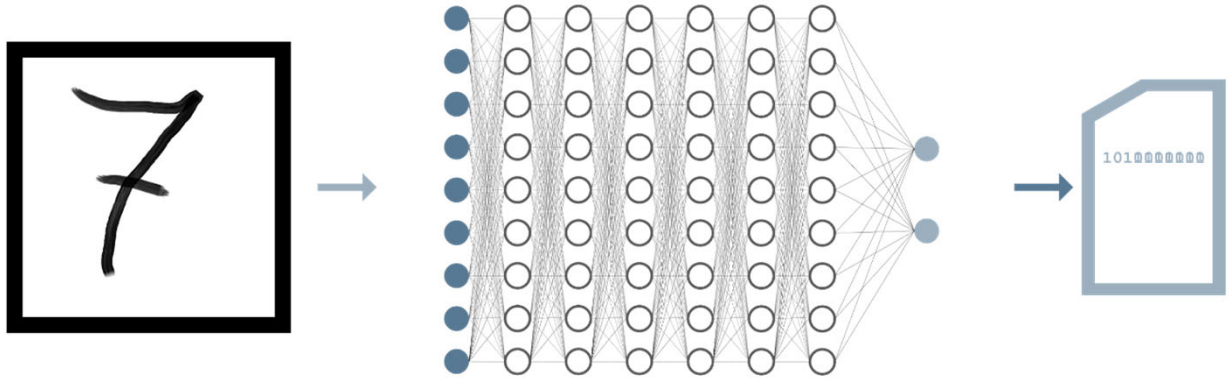
- Forschung durch **leistungsfähigere Hardware** und **steigende Datenverfügbarkeit** in letzten Jahren deutlich gestiegen
- Neben klassischen Feed-Forward-Netzen auch Recurrent Neural Networks handhabbar
  - Kanten können auch zu vorherigen Schichten zurückführen
- **Hohe Anzahl an Schichten**, welche nach Funktionsweise zusammengefasst werden können
- Verschiedene Architekturen haben sich für unterschiedliche Problemstellungen als besonders effektiv gezeigt
- **Bessere Skalierbarkeit**

Deep Learning
→ Architekturen (2/2)

- **Convolutional Neural Networks (CNN):**
  - Zweidimensionales „Fenster“ wird über Daten „geschoben“
  - Einfluss durch benachbarte Felder wird berücksichtigt
  - Besonders erfolgreich bei Computer Vision (z.B. Handschrift-Erkennung)

- **Long Short-Term Memory Networks (LSTM):**
  - Spezialform eines Recurrent Neural Networks
  - Neuronen können Zustände über einen längeren Zeitraum speichern
  - Besonders erfolgreich bei gesprochener Sprache (Alexa, Siri, usw.)

Deep Learning
→ Handschrifterkennung: Beispiel



Ziffer	0	1	2	3	4	5	6	**7**	8	9
Übereinstimmung	0 %	7 %	1%	0 %	4 %	0 %	0 %	**85 %**	0 %	3 %

- **Input-Daten (1):**
  - Bilddatei mit einer Zahl (7), die klassifiziert werden soll
- **ML-Algorithmus (2):**
  - Eingabedaten werden in den künstlichen Neuronen in den Schichten verarbeitet
  - Z.B. mit Hilfe eines Convolutional Neural Network (CNN)
- **Output (3):**
  - Tabelle mit einer Verteilung der **Wahrscheinlichkeiten** für eine Übereinstimmung mit **einer Ziffer**

KI für Cyber-Sicherheit
→ Inhalt

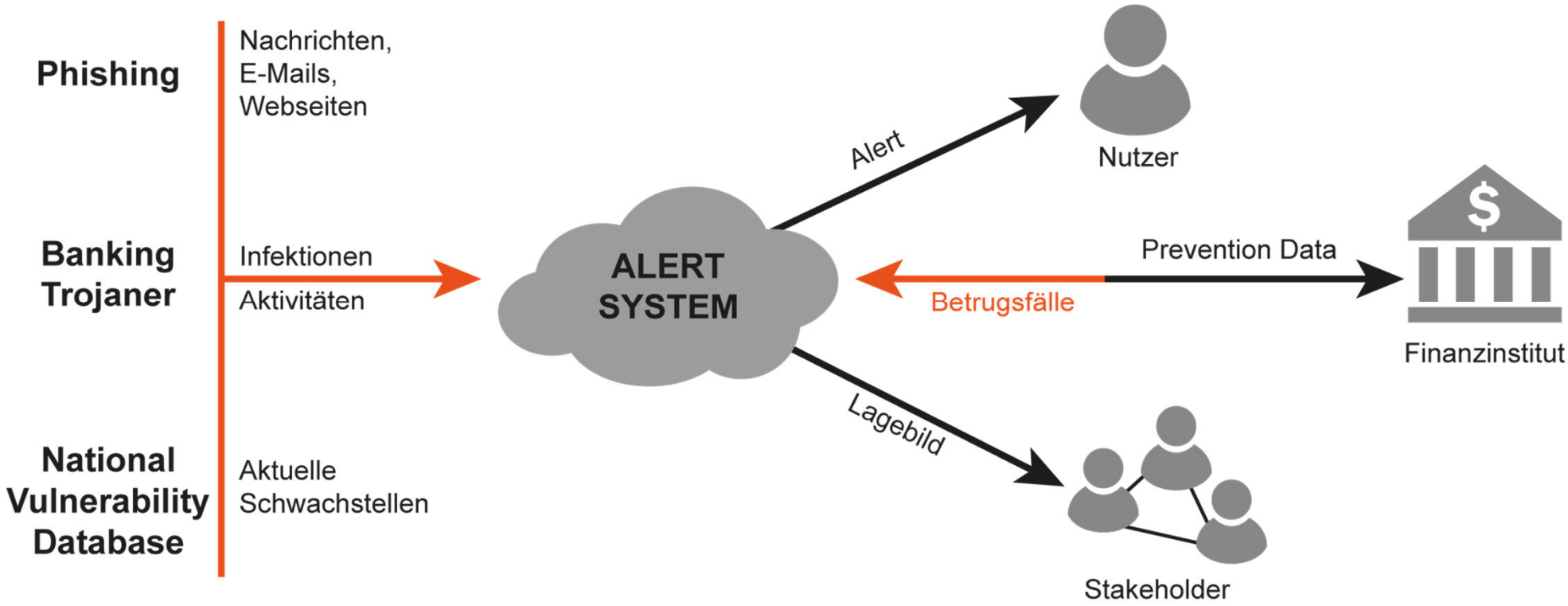
- Ziele und Ergebnisse der Vorlesung
- Einordnung
- Maschinelles Lernen
- Künstliche Neuronale Netze
- **Anwendungen KI und Cyber-Sicherheit**
- Angriffe auf maschinelles Lernen
- Herausforderungen
- Zusammenfassung

Anwendungen von KI und CS
→ Alert-System für Online-Banking

- Wie könnte eine Lösung aussehen?
  - Tagesaktuelle Warnungen bei erhöhter Gefahrenlage (Online-Banking)
    → damit der Bankkunde und die Bank reagieren können
  - Aufklärung der Nutzer, wenn Gefahren vorliegen
    → damit der Bankkunde sich „richtig“ verhalten kann

- Ansatz des Alert-Systems
  - ***Sicherheitskennzahlen*** zum Betrug identifizieren
  - Mittels KI ***Gefahrenlage bestimmen***
  - Nutzer und Bank ***Warnen***

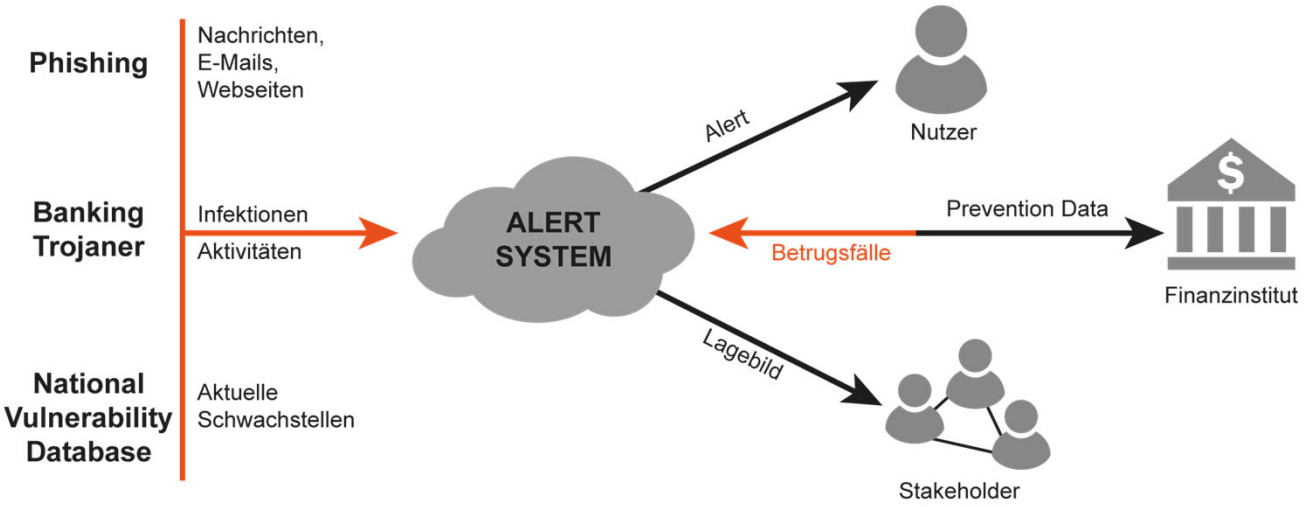
Alert-System für Online-Banking
→ Konzept



Alert-System für Online-Banking

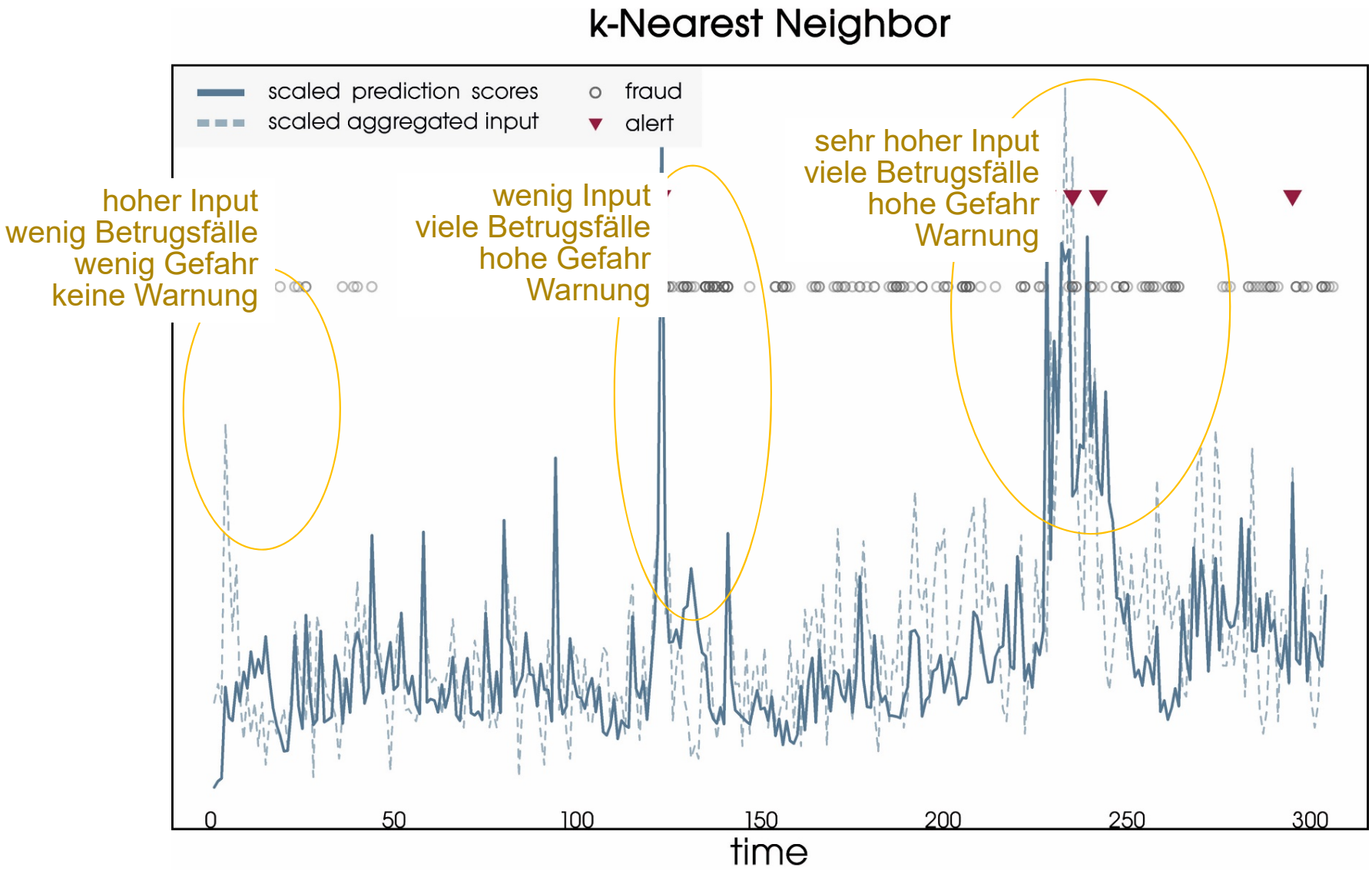
→ Zahlen für den Testzeitraum von 456 Tage

- 1.904 Nachrichten (Phishing-Angriff) – „Stackoverflow-Netzwerk“
- 5.589 **E-Mail** (Phishing-Angriff) – „Spam Archive“
- 2.776 Phishing-**Webseiten** – „PhishTank“
- 23.184 **Infektionen** von Banking-Trojaner (Malware) – Anti-Malwarehersteller
- 875 relevante **Schwachstellen** (NVD)
- 459 erfolgreiche **Betrugsfälle** im Online-Banking - Bankengruppe



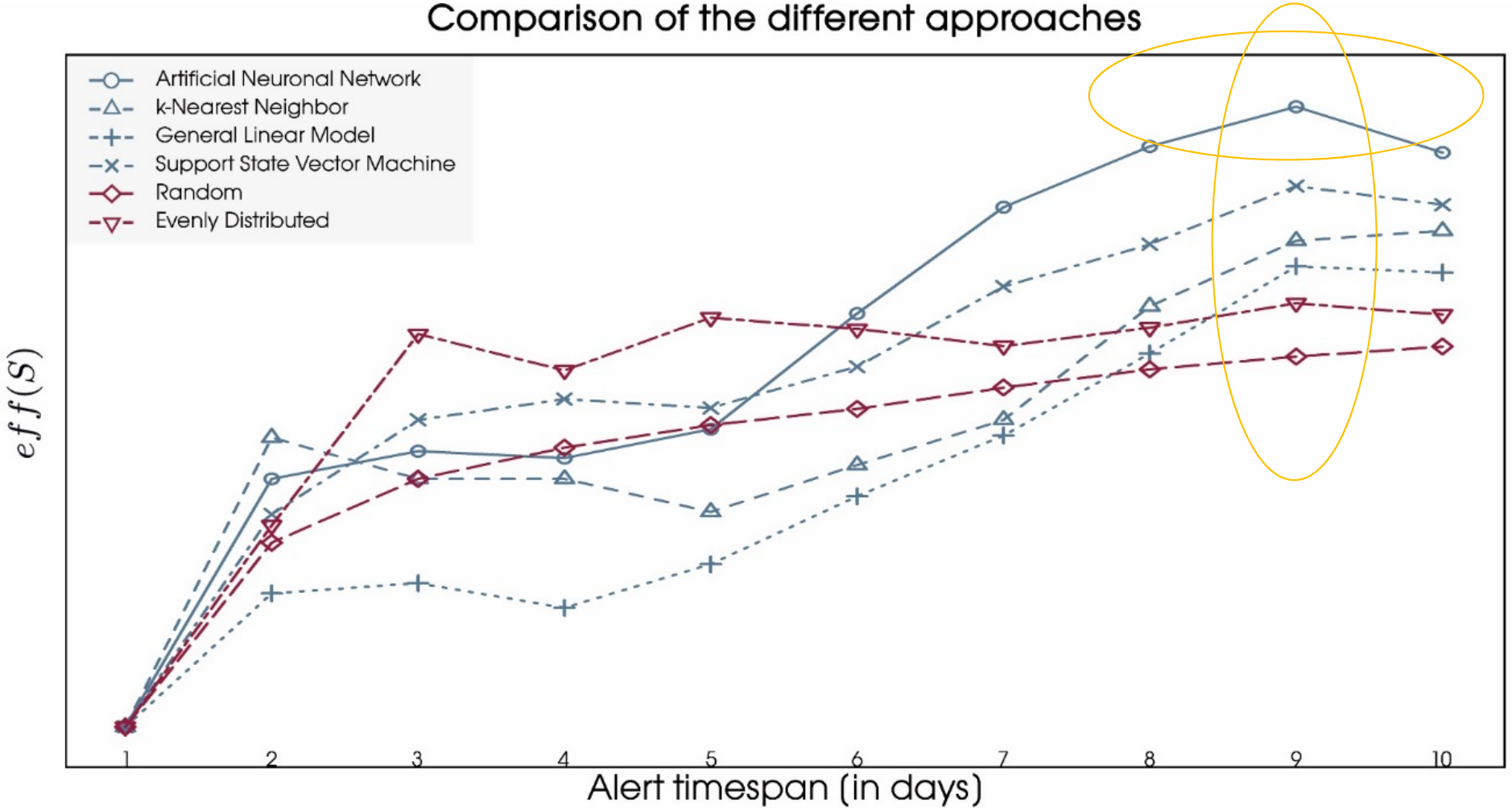
$^1/_3$ des Zeitraums zum Training (152 Tage) $^2/_3$ zur Evaluation (304 Tage)

Ergebnis einschätzen
→ k-Nearest Neighbor

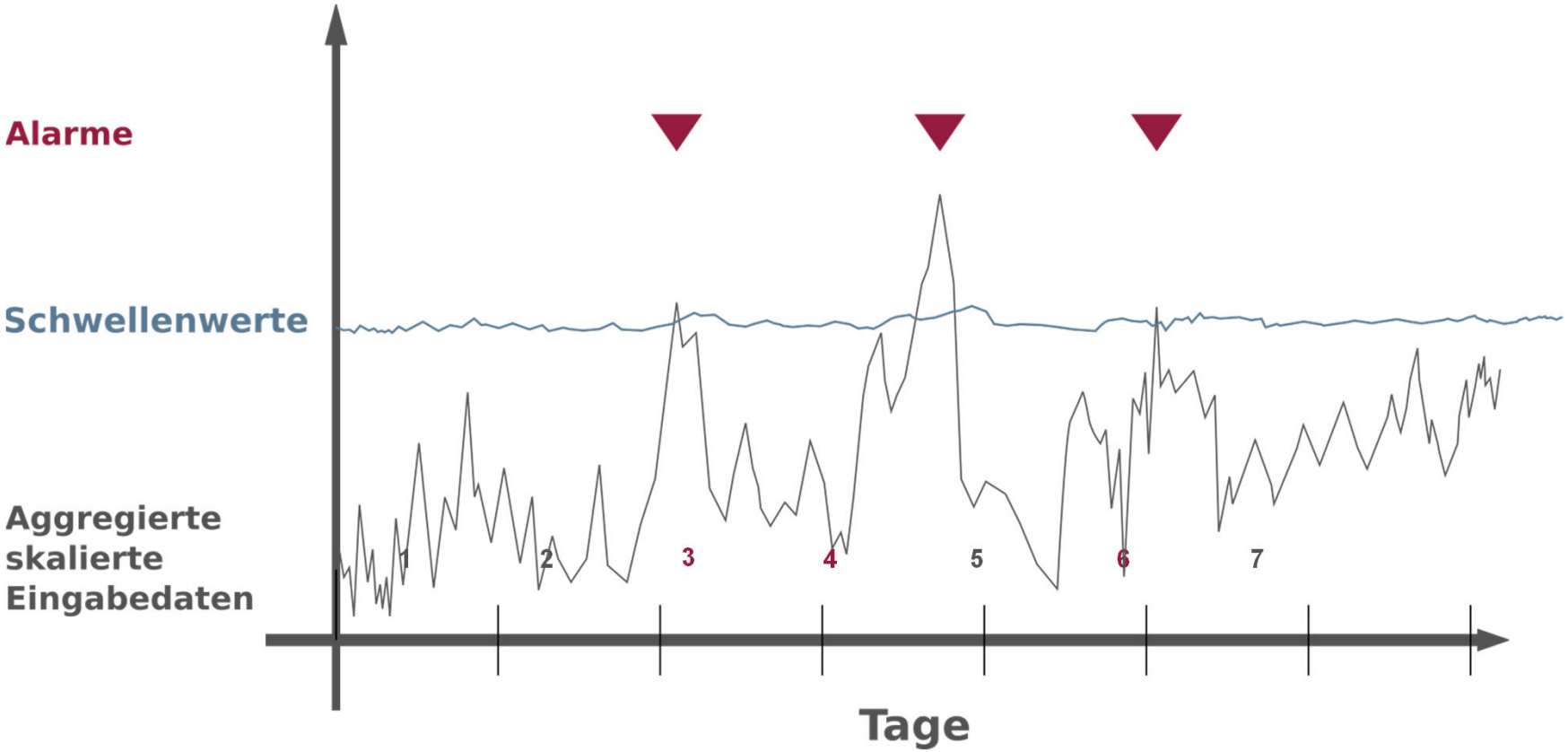


Ergebnisse
→ Vergleich der verschiedenen Verfahren

„Aber, drei Mal soviel Zeit für das Trainieren“



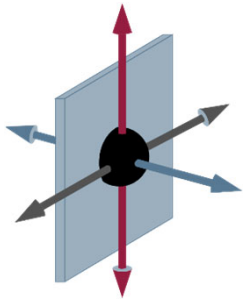
Alert-System für Online-Banking
→ Ergebnis

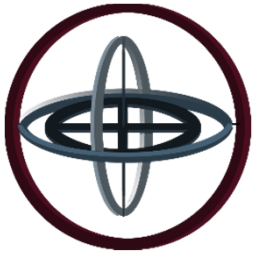


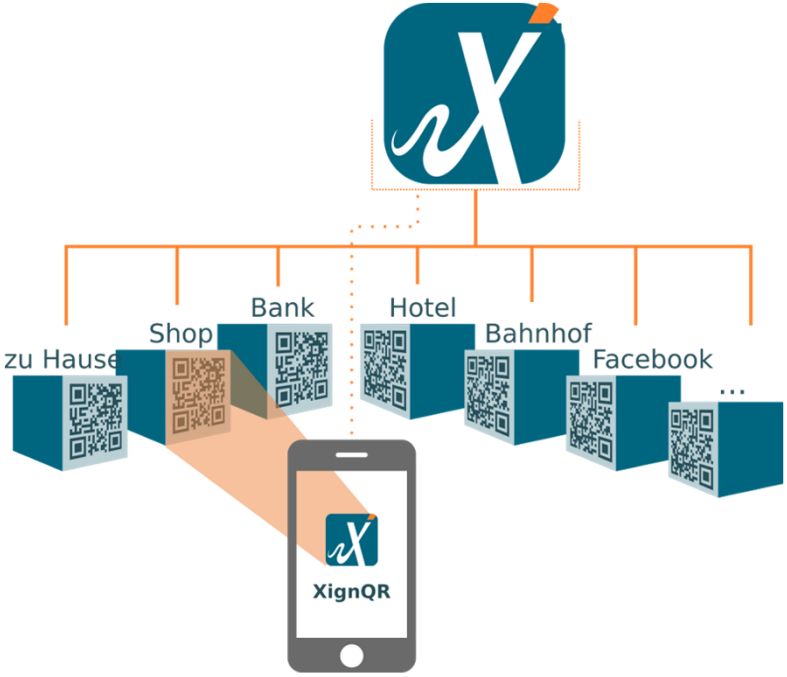
- **Output:**
  - Vorhergesagte Bedrohungswerte überschreiten an den Tagen 3, 4 und 6 den für dieses Alert-System eingestellten Schwellenwert
  - da Schwellenwert überschritten wurde, wird ein Alarm ausgelöst

Anwendungen von KI und CS (2/2)
→ Passive Authentifikation - XignQR

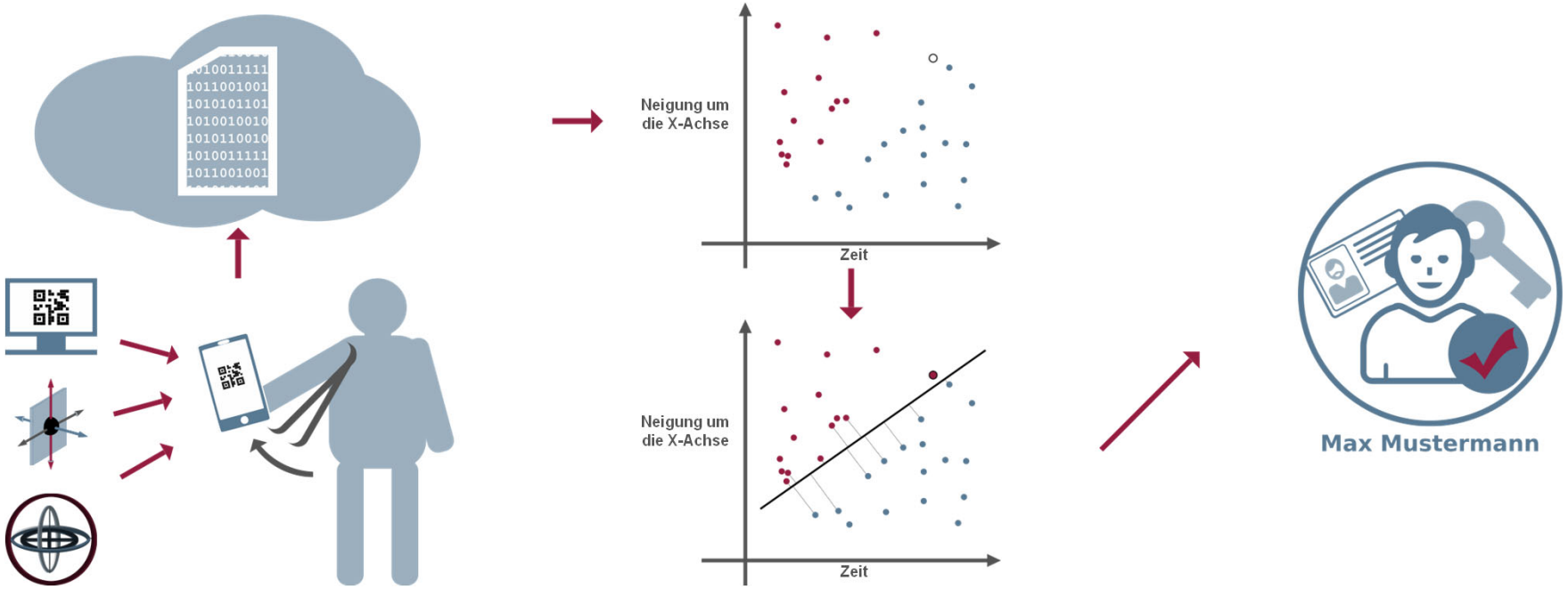
- Ein Nutzer wird automatisiert an der Art und Weise der Nutzung beim QR-Code Scannen erkannt.
- Während das gesamten Vorgangs werden passive biometrische Bewegungsdaten erfasst.
- Datenerfassung durch
  - **Beschleunigungssensor**
  - **Lagesensor**





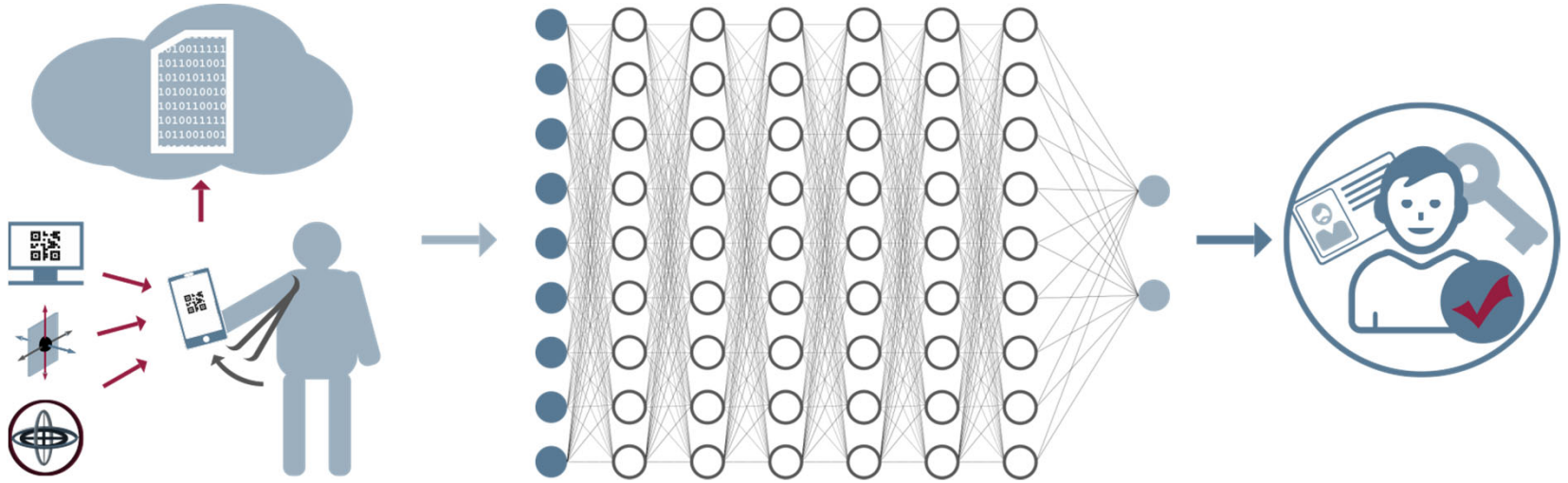


Passive Authentifikation - XignQR
→ Support-Vector-Machine (SVM)



- **Input-Daten:**
  - Nutzer holt Gerät aus Hosentasche
  - Erfassen von **Lage** und **Beschleunigung** des Smartphones
- **ML-Algorithmus:**
  - Daten werden anhand der Hyperebene/des Modell klassifiziert
  - rote Übereinstimmung ist positive Klassifizierung
  - blau eine negative Klassifizierung (bspw. anderer Nutzer)
- **Output:**
  - Authentisierung ist entweder erfolgreich oder schlägt fehl (**95 %**)

Passive Authentifikation - XignQR
→ Neuronales Netz



- **Input-Daten:**
  - Lage und Beschleunigungsdaten des Nutzers werden erzeugt
- **ML-Algorithmus:**
  - Eingabedaten werden in den künstlichen Neuronen in den Schichten verarbeitet
- **Output:**

Nutzer	Übereinstimmung
0	0,059 %
1	99,85 %
2	0,087 %

```
time, type, x, y, z
271, Accelerometer, -0.07606506, 9.173798, 3.6333618
277, Accelerometer, 1.0681152E-4, 9.146423, 3.5619507
279, Gyroscope, 0.027664185, 0.06774902, 0.02182006
...
```

```
[[5.9110398e-04 9.9853361e-01 8.7528664e-04]]

Predicted Class [1]

Predicted Person: Sandra Kreis
```

KI für Cyber-Sicherheit
→ Weitere Beispiele

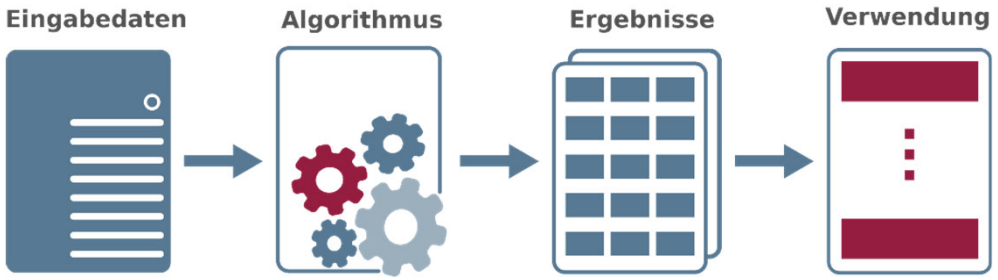
- Logdatenanalyse
- Malware-Erkennung
- Security Information and Event Management (SIEM)
- Threat Intelligence
- Spracherkennung
- Bilderkennung (Ausweis, Video, …)
- Authentifikationsverfahren
- Fake-News
- IT-Forensik
- Sichere Softwareentwicklung
- …

KI für Cyber-Sicherheit
→ Inhalt

- Ziele und Ergebnisse der Vorlesung
- Einordnung
- Maschinelles Lernen
- Künstliche Neuronale Netze
- Anwendungen KI und Cyber-Sicherheit
- **Angriffe auf maschinelles Lernen**
- Herausforderungen
- Zusammenfassung

Künstliche Intelligenz / ML
→ Angriffe

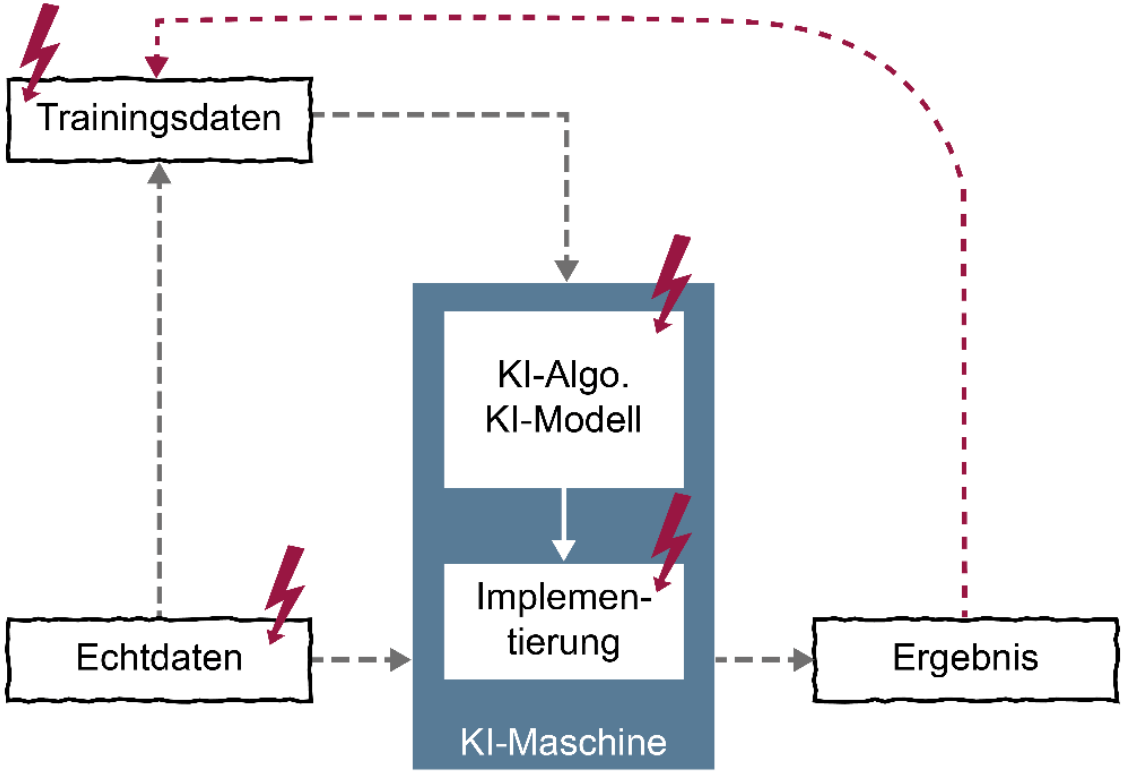
- **„Hacker“ greifen an und manipulieren den Workflow**
  - die Eingabedaten (Input)
    - gezielte Manipulation
  - die Algorithmen
  - die Ergebnisse (Output)
  - die Verwendung



- **Angriffe auf die Privatsphäre**
  (personenorientierte Daten, die verwendet werden)

Vertrauenswürdigkeit
→ Qualität der Umsetzung

Stand der Technik *an*
IT-Sicherheitsmaßnahmen
zum ***Schutz***
→ *der* ***Daten*** *(Training, Echt, Ergebnis),*
→ *der* ***KI-Maschine*** *und*
→ *der* ***Anwendung***



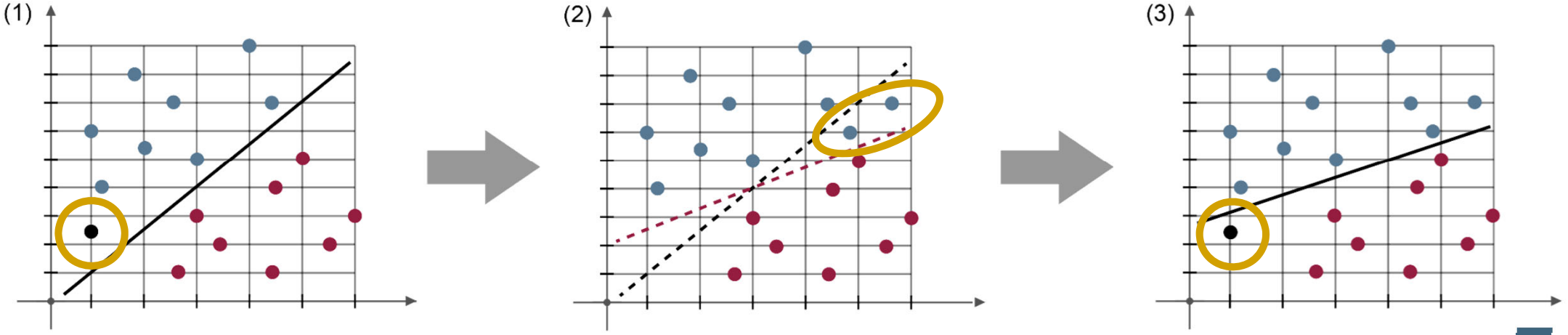
Schutzziele:
→ ***Integrität***
(Erkennen von Manipulation der Daten)
→ ***Vertraulichkeit***
(Wahrung von Geschäftsgeheimissen)
→ ***Datenschutz***
(Schutz von personenbezogenen Daten)
→ ***Verfügbarkeit***
(der Anwendung und Ergebnisse)

Nutzung *einer*
qualitativ hochwertigen
KI-Technologie

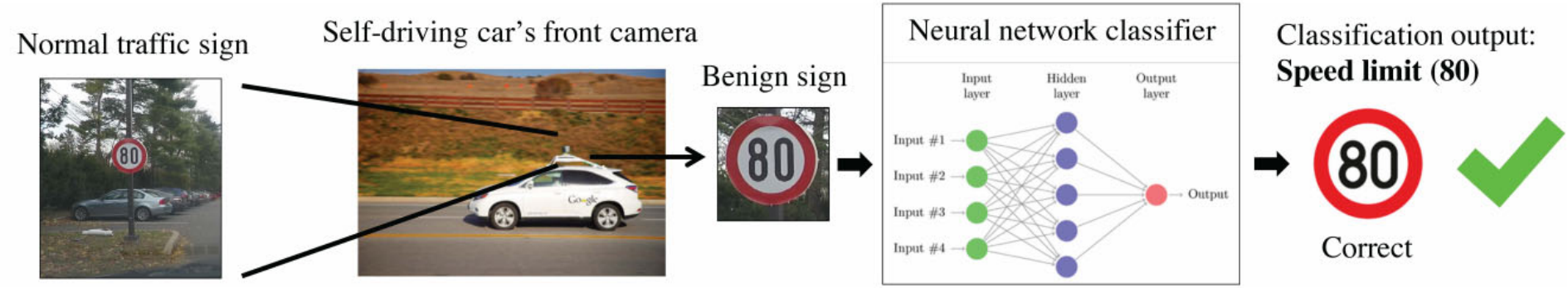
Zusammenarbeit *von* ***erfahrenen***
KI- *und*
Anwendungsexperten

Angriffe auf maschinelles Lernen
→ Manipulation von Trainingsdaten

- (1) **Normale Klassifizierung** eines neuen Inputs.
  (neuer schwarzer Punkt gehört zur blauen Klasse)
- (2) **Beispiel: Manipulation von Trainingsdaten**
  - Falsch klassifizierte Daten werden in den Trainingsprozess als Angriff einschleusen *(zwei weitere blaue Punkte)*.
  - Dadurch wird die Gerade des Modells zur Klassifizierung manipuliert *(Gerade wird flacher)*.
- (3) Damit kann ein Angreifer für **falsche Klassierungen** sorgen.
  (jetzt gehört der neuer schwarzer Punkt zur roten Klasse)



Angriffe auf maschinelles Lernen
→ Manipulation von Verkehrszeichen



(a) Operation of the computer vision subsystem of an AV under *benign conditions*

(b) Operation of the computer vision subsystem of an AV under *adversarial conditions*

Fig. 1. **Difference in operation of autonomous cars under benign and adversarial conditions.** Figure 1b shows the classification result for a drive-by test for a physically robust adversarial example generated using our Adversarial Traffic Sign attack.

Quelle: https://arxiv.org/abs/1802.06430

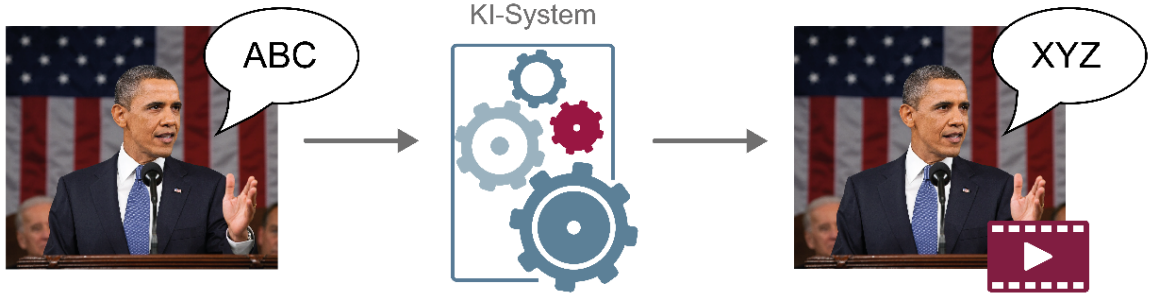
KI für Cyber-Sicherheit
→ Inhalt

- Ziele und Ergebnisse der Vorlesung
- Einordnung
- Maschinelles Lernen
- Künstliche Neuronale Netze
- Anwendungen KI und Cyber-Sicherheit
- Angriffe auf maschinelles Lernen
- **Herausforderungen**
- Zusammenfassung

Künstliche Intelligenz
→ Angreifer verwenden KI

„Hacker“ verwenden KI ebenfalls für ihre Zwecke (Dual-Use)

- Schnelle Schwachstellensuche (schneller Angreifen, neue Angriffsvektoren)
- Social-Engineering (Chatbots, …)
- Passwortknacker
- Neue Angriffsstrukturen und Vorgehensweisen
- Videomanipulation (Deep-Fake)
  - „Fake Obama Video“
  - „Make Putin Smile Video“



Künstliche Intelligenz
→ Allgemeine Herausforderungen

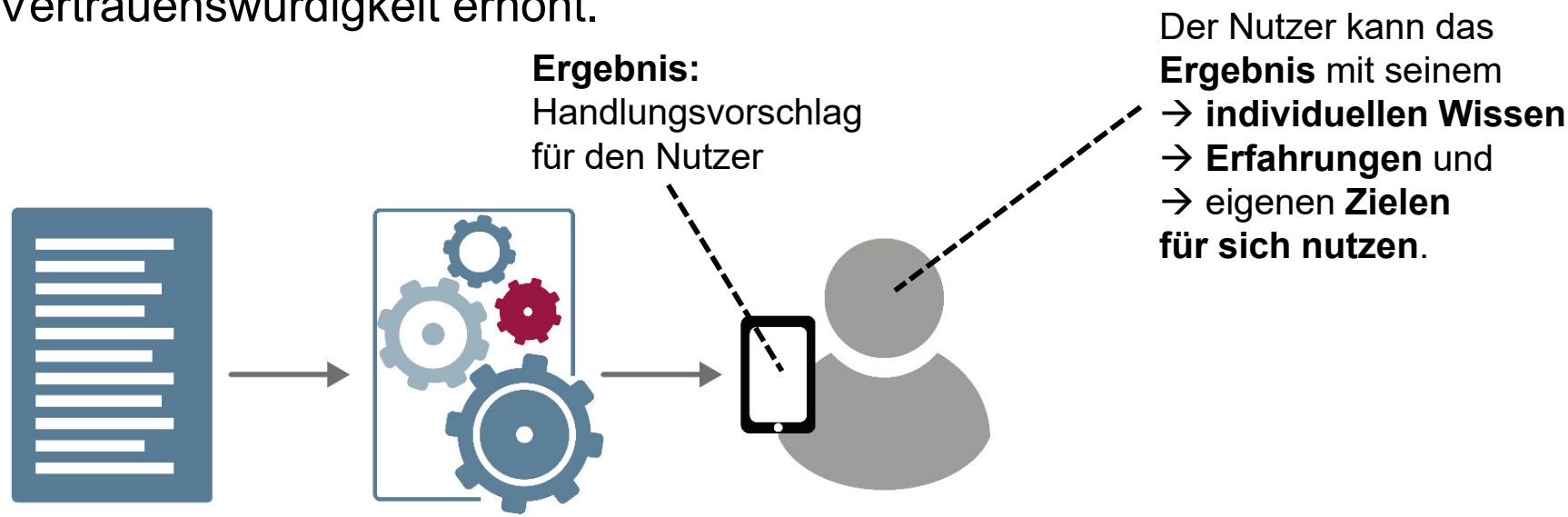
- **Datenschutz** (persönliche Daten … Europäische Datenschutz-Grundverordnung)
- **Selbstbestimmung** („human in the loop“)
- **Diskriminierung** (ausgeglichene Daten … Problem: gibt es nicht)
  → Frau/Mann, Herkunft, Ausbildung, …
- **Vertrauenswürdigkeit** der Daten und Ergebnisse
  → KI-Siegel
- …

Intelligente Algorithmen
→ Chancen und Risiken

- **Individuelles Wissen** und **Komplexität des denkenden** Menschen sind Algorithmen überlegen! **+**
- **Algorithmen können schneller** Wissen aus vorhandenen **Daten auswerten! +**
- Individuelles Wissen + Algorithmen Wissen = **+++**
- **Praktische Probleme:** Medizin / Watson
  - Diagnostik *(Maschine)*
  - Haftung *(Mensch)*

Vertrauenswürdigkeit
→ Nachvollziehbarkeit der Ergebnisse

- **„Keep the human in the loop“**
  - KI-Ergebnis muss als **Handlungsempfehlung** für den **Nutzer** verstanden werden.
  - Damit wird die **Selbstbestimmtheit** der Nutzer gefördert und die Vertrauenswürdigkeit erhöht.



- **Automatisierte Anwendungen** (z.B. autonomes Fahren)
  - Simulation, Test und **Validierung**
  - Verantwortung, **Haftung** und Versicherung

Forschungsfragen

→ Sicherheit/Vertrauenswürdigkeit von KI (1)

- **Sicherheit und Vertrauenswürdigkeit der verwendeten Daten:**
  - Sicherheitsinfrastruktur für
    - **Integrität** (Erkennung von Manipulationen an Daten)
    - **Vertraulichkeit** (Schutz von Geschäftsgeheimnissen)
    - **Datenschutz** (Schutz von persönlichen Daten)
    - **Verfügbarkeit** (der Anwendung und Ergebnisse)

- **Sichere und vertrauenswürdige Implementierung:**
  - Cybersicherheitsmechanismen für den Schutz von
    - Daten,
    - KI-Algorithmen und
    - Anwendungen

Forschungsfragen

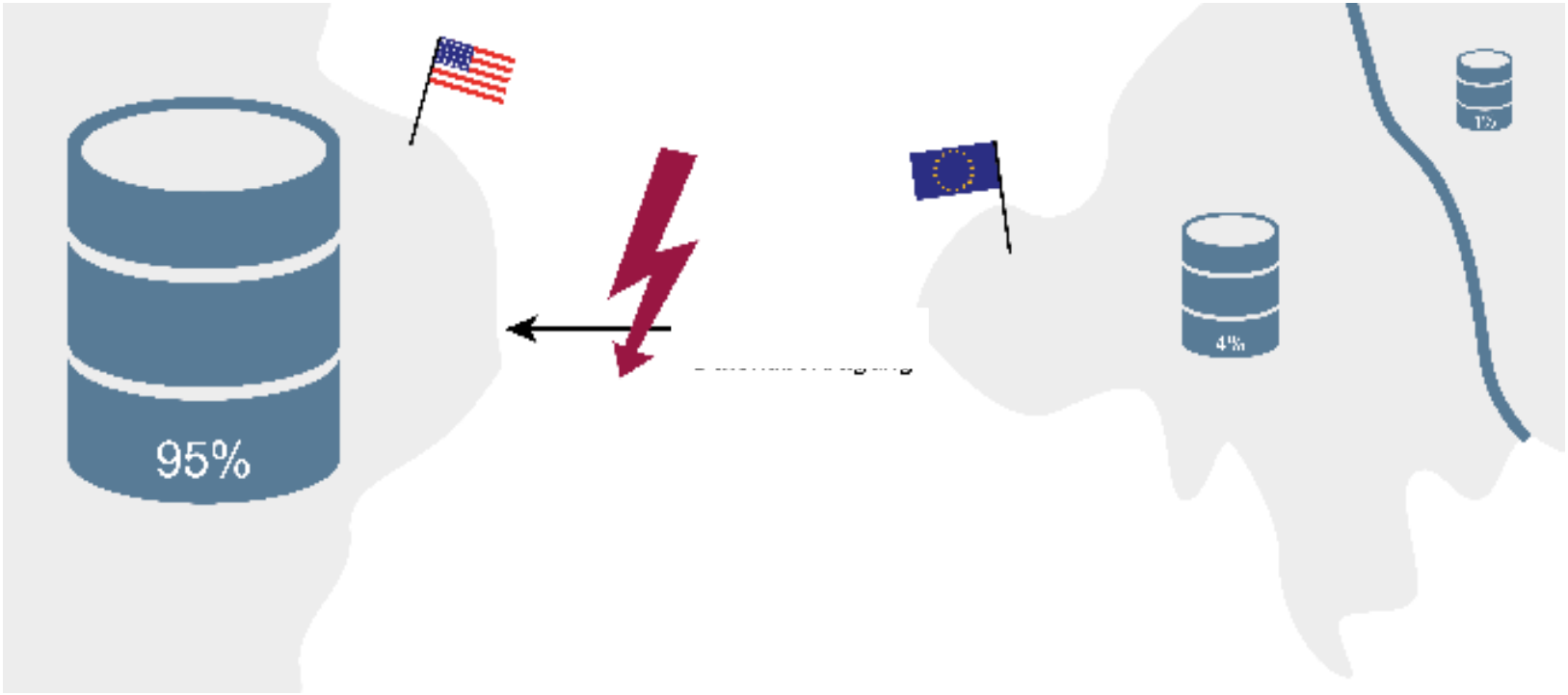
→ Sicherheit/Vertrauenswürdigkeit von KI (2)

- **Nachvollziehbarkeit von Entscheidungen**
  - Infrastrukturen für die Überprüfung von Verantwortungen (Blockchain, PKI, …)

Forschungsfragen
→ Souveränität

- Wir brauchen leistungsfähige KI-Infrastrukturen, um die digitale Souveränität aufrechtzuerhalten.

- Verfügbarkeit von Daten.



Forschungsfragen

→ Austausch von sicherheitsrelevanten Daten

- Nützlich für bessere Ergebnisse!
- Wie kann der Austausch attraktiv gestaltet warden?
- Was sind die Nachteile?
- …

KI für Cyber-Sicherheit
→ Inhalt

- Ziele und Ergebnisse der Vorlesung
- Einordnung
- Maschinelles Lernen
- Künstliche Neuronale Netze
- Anwendungen KI und Cyber-Sicherheit
- Angriffe auf maschinelles Lernen
- Herausforderungen
- **Zusammenfassung**

KI für Cyber-Sicherheit
→ Zusammenfassung (1/2)

- **KI/ML ist eine wichtige Technologie für die Zukunft, auch für Cyber-Sicherheit**
  - Erkennen von Bedrohungen, Schwachstellen, Angriffen, …
  - Erkennen von Nutzern (Authentifikation)
  - Unterstützung von Cyber-Sicherheitsexperten
  - …
- **Sehr gute Daten sind das Wichtigste**
  - Neue, bessere Sensoren (Daten mit sehr gutem Inhalt)
  - Zusammenarbeit und Austausch von Daten
  - …
- **Technologische- und Daten-Souveränität wird immer wichtiger**

Westfälische Hochschule
Gelsenkirchen Bocholt Recklinghausen
University of Applied Sciences

Künstliche Intelligenz für Cyber-Sicherheit

- Vorlesung Cyber-Sicherheit -

Prof. Dr. (TU NN)
Norbert Pohlmann

Institut für Internet-Sicherheit – if(is)
Westfälische Hochschule, Gelsenkirchen
http://www.internet-sicherheit.de

if(is) internet-sicherheit.

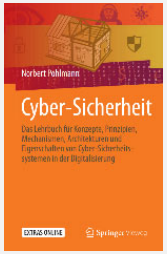
Anhang / Credits

Wir empfehlen

- **Kostenlose App securityNews**

securityNews

- **7. Sinn im Internet (Cyberschutzraum)**
  https://www.youtube.com/cyberschutzraum
- **Master Internet-Sicherheit**
  https://it-sicherheit.de/master-studieren/
- **Cyber-Sicherheit**
  Das **Lehrbuch** für Konzepte, Mechanismen, Architekturen und Eigenschaften von Cyber-Sicherheitssystemen in der Digitalisierung“, Springer Vieweg Verlag, Wiesbaden 2019
- https://norbert-pohlmann.com/cyber-sicherheit/



Besuchen und abonnieren Sie uns :-)

WWW
https://www.internet-sicherheit.de

Facebook
https://www.facebook.com/Internet.Sicherheit.ifis

Twitter
https://twitter.com/_ifis

YouTube
https://www.youtube.com/user/InternetSicherheitDE/

Prof. Norbert Pohlmann
https://norbert-pohlmann.com/

Quellen Bildmaterial

Eingebettete Piktogramme:
- Institut für Internet-Sicherheit – if(is)

Der Marktplatz IT-Sicherheit

(IT-Sicherheits-) Anbieter, Lösungen, Jobs, Veranstaltungen und Hilfestellungen (Ratgeber, IT-Sicherheitstipps, Glossar, u.v.m.) leicht & einfach finden.
https://www.it-sicherheit.de/

Literatur
→ Artikel / Bücher

C. Paulisch, N. Pohlmann, R. Riedel, T. Urban: „Sei gewarnt! Vorhersage von Angriffen im Online-Banking“. In Proceedings der „DACH Security 2018 Konferenz“, syssec Verlag, 2018
https://norbert-pohlmann.com/wp-content/uploads/2019/02/384-Sei-gewarnt-Vorhersage-von-Angriffen-im-Online-Banking-Prof.-Norbert-Pohlmann.pdf

N. Pohlmann: „Künstliche Intelligenz und Cybersicherheit“, Diskussionsgrundlage für den Digitalgipfel, 2018
https://norbert-pohlmann.com/wp-content/uploads/2019/02/Künstliche-Intelligenz-und-Cybersicherheit-Diskussionsgrundlage-für-den-Digitalgipfel-2018-Prof.-Norbert-Pohlmann.pdf

N. Pohlmann: „Künstliche Intelligenz und Cybersicherheit - Unausgegoren aber notwendig“, IT-Sicherheit – Fachmagazin für Informationssicherheit und Compliance, DATAKONTEXT-Fachverlag, 1/2019
https://norbert-pohlmann.com/wp-content/uploads/2019/04/393-Künstliche-Intelligenz-und-Cybersicherheit-Unausgegoren-aber-notwendig-Prof.-Norbert-Pohlmann.pdf

U. Coester, N. Pohlmann: „Ethik und künstliche Intelligenz - Wer macht die Spielregeln für die KI?“, IT & Production – Zeitschrift für erfolgreiche Produktion, TeDo Verlag, 2019
https://norbert-pohlmann.com/wp-content/uploads/2019/08/406-Ethik-und-künstliche-Intelligenz-Wer-macht-die-Spielregeln-für-die-KI-Prof.-Norbert-Pohlmann.pdf

N. Pohlmann: „Sicherheit und Vertrauenswürdigkeit von KI-Systemen“, Thesen und Handlungsempfehlungen, Thesenpapier für die Enquete-Kommission KI des Deutschen Bundestagen, 2019
https://norbert-pohlmann.com/wp-content/uploads/2019/07/Thesenpapier-Enquete-Kommission-KI-Datensicherheit-Prof.-Norbert-Pohlmann-03_06_19.pdf

N. Pohlmann: "Cyber-Sicherheit – Das Lehrbuch für Konzepte, Mechanismen, Architekturen und Eigenschaften von Cyber-Sicherheitssystemen in der Digitalisierung“, ISBN 978-3-658-25397-4; 594 Seiten, Springer-Vieweg Verlag, Wiesbaden 2019
https://norbert-pohlmann.com/cyber-sicherheit/