Decision Support Systems
Entscheidungsunterstützungssystem

Fabian Karl

Arbeitsbereich Wissenschaftliches Rechnen
Fachbereich Informatik
Fakultät für Mathematik, Informatik und Naturwissenschaften
Universität Hamburg

Gliederung

Was ist das?

Ein künstliches System, dass Menschen hilft gute und optimale Entscheidungen zu treffen.

Wann wird das gebraucht?

- Wenn es zu viele Daten gibt
- Wenn Daten zu unstrukturiert, zu hoch-dimensional oder zu groß werden
- Wenn man als Mensch nicht mehr zurückverfolgen kann, wie sich die Daten beeinflussen

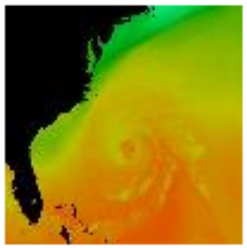
Wie kann eine Decision Support System aussehen?

- Systeme, die Menschen bei deren Entscheidungen helfen:
  - Visualisieren der Daten
  - Umstrukturieren der Daten
  - Vergleiche erstellen
  - Clustern
  - Mittelwerte berechnen
- Systeme, die eine Entscheidung treffen bzw. vorschlagen:
  - Supervised Learning (Überwachtes Lernen)

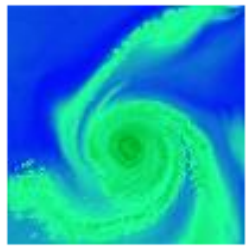
Beispielanwendung: Datenkompression

Finde einen optimalen Kompressionsalgorithmus für neue und unbekannte wissenschaftliche Daten.

Isabel: Ein Klimadatensatz



Hurricane Isabel WRF Model Data

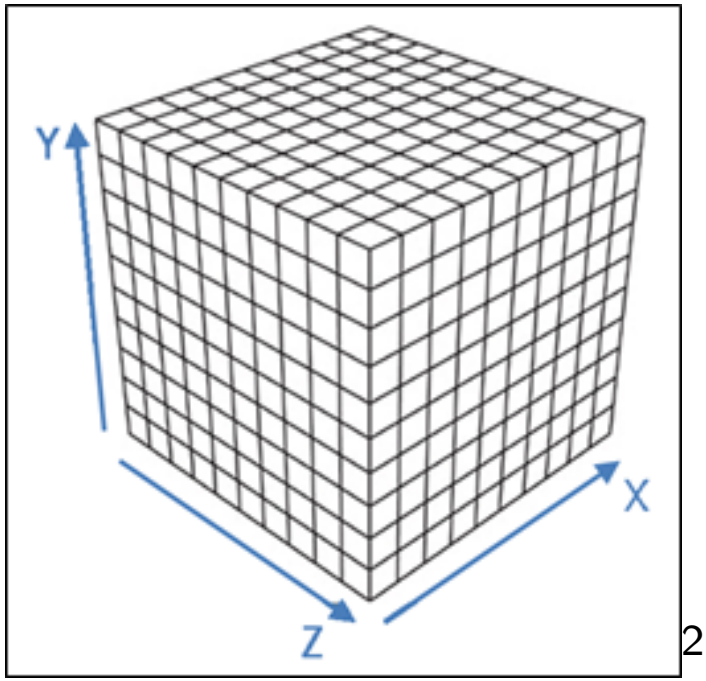


- Daten vom Hurricane Isabel im September 2003[1]
- 13 verschiedene Messeinheiten, die über 48 Zeitschritte (time) gemessen wurden
- Jeder Zeitschritt hat 500x500x100 Gleitkommazahlen (longitude x latitude x vertical)

[1] http://www.vets.ucar.edu/vg/isabeldata/

Ein Datenpunkt

Abbildung: 3-Dimensionale Matrix



[2]

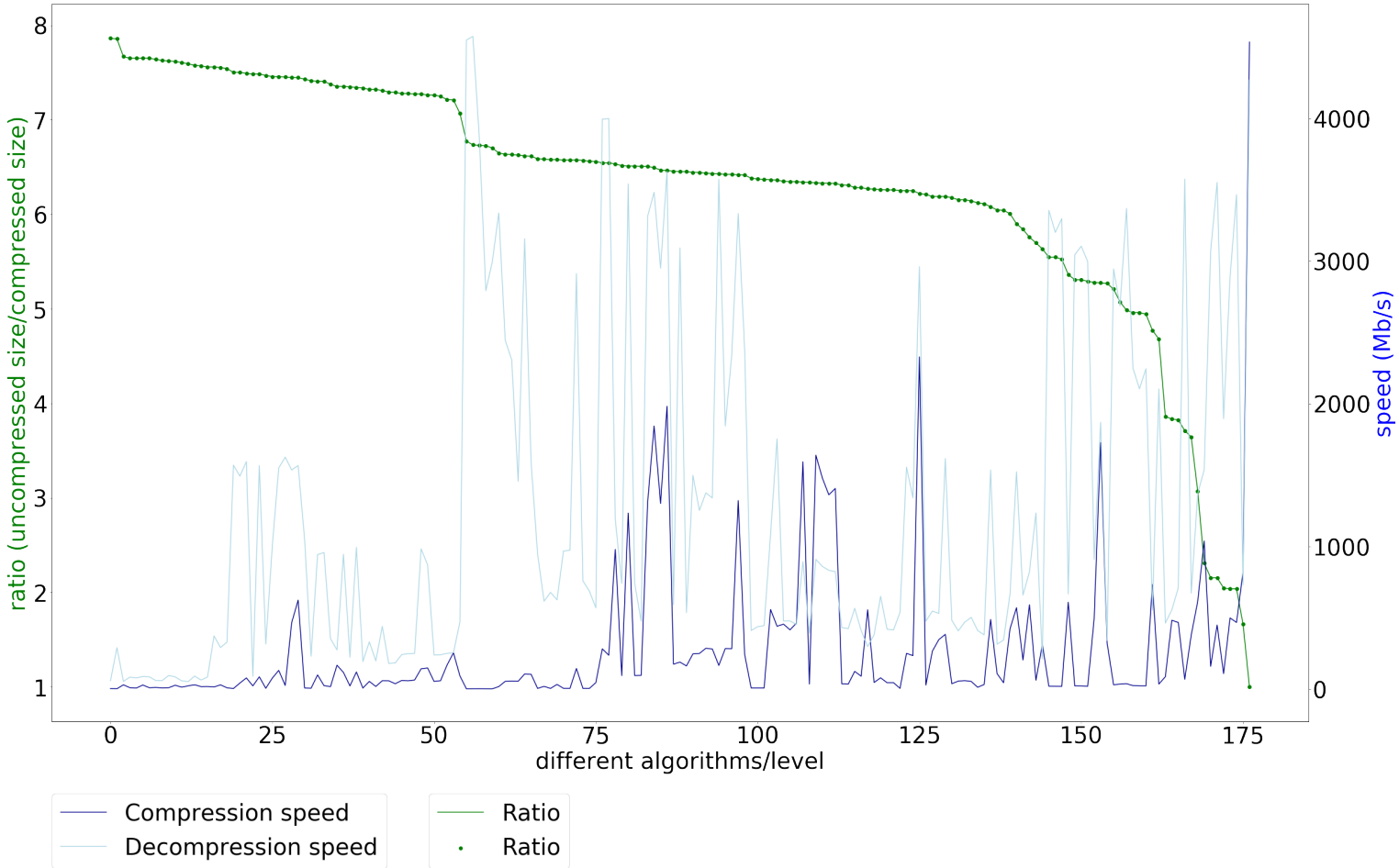
[2] https://www.quora.com/I-have-a-hypothesis-I-believe-the-singularity-of-a-black-hole-is-a-solid-and-a-Bose%E2%80%93Einstein-condensate-am-I-correct-or-close

Tabelle: LzBench: Ein Kompressionsbenchmark

Nummer	Name	Version/Datum	Nummer	Name	Version/Datum
0	blosclz	10.11.2015	19	lzmat	v1.01
1	brieflz	01.01.2000	20	lzo	v2.09
2	brotli	12.12.2017	21	lzrw	15.07.1991
3	crush	v1.0	22	lzsse	14.05.2016
4	csc	13.10.2016	23	pithy	24.12.2011
5	density	v0.12.5 beta	24	quicklz	v1.5.0
6	fastlz	v0.1	25	shrinker	v0.1
7	gipfeli	13.07.2016	26	slz	v1.0.0
8	glza	v0.8	27	snappy	v1.1.4
9	libdeflate	v0.7	28	tornado	v0.6a
10	lizard	v1.0	29	ucl	v1.03
11	lz4/lz4hc	v1.8.0	30	wflz	16.09.2015
12	lzf	v3.6	31	xpack	02.06.2016
13	lzfse/lzvn	08.03.2017	32	xz	v5.2.3
14	lzg	1.v0.8	33	yalz77	19.09.2015
15	lzham	v1.0	34	yappy	22.03.2014
16	lzjb	2010	35	zlib	v1.2.11
17	lzlib	v1.8	36	zling	10.04.2016
18	lzma	v16.04	37	zstd	v1.3.3

Ergebnis: Für eine Datei

Abbildung: Ergebnis für eine Datei aus Isabel



Ergebnis: Mehrere verschiedene Daten

Tabelle: Die besten drei Algorithmen pro Datenpunk

Data sample	best ratio	second best ratio	third best ratio
QRAINf02	brotli 2017-03-10 -11	lzlib 1.8 -9	xz 5.2.3 -6
QRAINf04	lzlib 1.8 -9	brotli 2017-03-10 -11	lzma 16.04 -9
QRAINf05	lzlib 1.8 -9	brotli 2017-03-10 -11	xz 5.2.3 -6
QRAINf14	lzlib 1.8 -9	lzlib 1.8 -6	lzma 16.04 -5
QRAINf30	lzlib 1.8 -9	lzlib 1.8 -6	xz 5.2.3 -6
QRAINf31	lzlib 1.8 -9	xz 5.2.3 -6	lzlib 1.8 -6
QSNOWf01	lzlib 1.8 -9	lzlib 1.8 -6	lzma 16.04 -5
QSNOWf02	lzlib 1.8 -9	xz 5.2.3 -6	xz 5.2.3 -9
QSNOWf04	lzlib 1.8 -9	xz 5.2.3 -6	lzma 16.04 -9
QSNOWf07	lzlib 1.8 -9	xz 5.2.3 -6	lzlib 1.8 -6
QSNOWf29	lzlib 1.8 -6	lzlib 1.8 -9	xz 5.2.3 -6
QSNOWf31	lzlib 1.8 -6	xz 5.2.3 -6	lzlib 1.8 -9
QICEf01	lzlib 1.8 -9	xz 5.2.3 -6	lzlib 1.8 -3
QICEf05	brotli 2017-03-10 -11	lzlib 1.8 -9	lzlib 1.8 -3
QICEf06	brotli 2017-03-10 -11	lzlib 1.8 -3	lzlib 1.8 -9
QICEf08	brotli 2017-03-10 -11	lzlib 1.8 -9	lzlib 1.8 -6
QICEf09	brotli 2017-03-10 -11	lzlib 1.8 -9	xz 5.2.3 -9
QICEf19	brotli 2017-03-10 -11	lzlib 1.8 -9	lzlib 1.8 -0

Lösung durch Testen

- Durch Testen eines jeden Datenpunkts wird das Problem gelöst
- Daten können nach verschiedenen Metriken geordnet werden
- Der optimale Algorithmus kann ausgewählt werden

Vor- und Nachteile von naivem Testen

+ korrekter Weg um den optimalen Lösung zu finden
+ Ergebnisse können wieder verwendet werden

– sehr hoher Zeitaufwand
– Bei anderen Problemen noch viel Aufwändiger oder unmöglich (Bsp: Bilderkennung, Aktienkurse)
– Expertenwissen wird benötigt

Verwenden einer Heuristik

- Eventuell wurde bereits ein sehr ähnlicher Datensatz getestet und gespeichert
- Finde über eine Heuristik einen bereits getesteten Datensatz, der dem unbekannten am ähnlichsten ist
- Statt jeden neuen, unbekannten Datensatz zu testen

- Zeitgewinn aber Verlust an Genauigkeit und Sicherheit

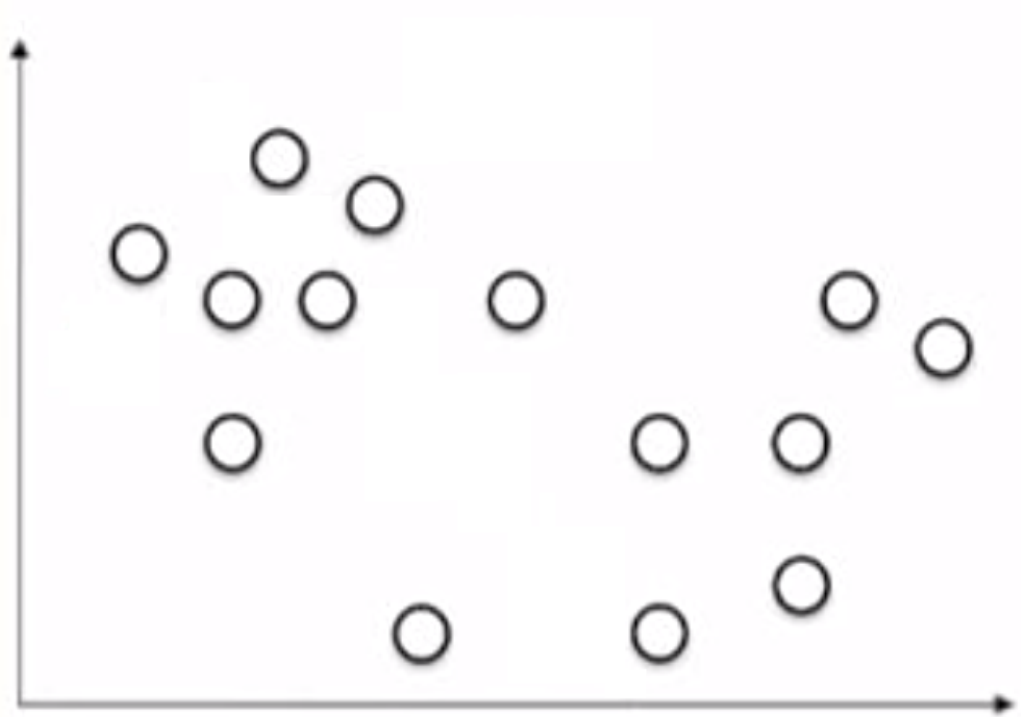
Kategorisierung

- Jedem Datenpunkt eine Gruppe zuweisen
- Zum Beispiel: Jedem Datensatz einen Kompressionsalgorithmus zuweisen
- Bei minimalem Wissen (noch keine getesteten Daten): Clustering

Clustering

- Clustering ist eine Form von Unsupervised Learning
- Datenpunkte werden aufgrund einer unterliegenden Struktur in Gruppen (Cluster) eingeteilt
- Algorihtmen: **K-Means** [4], Expectation-Maximization, Self Organizing Maps

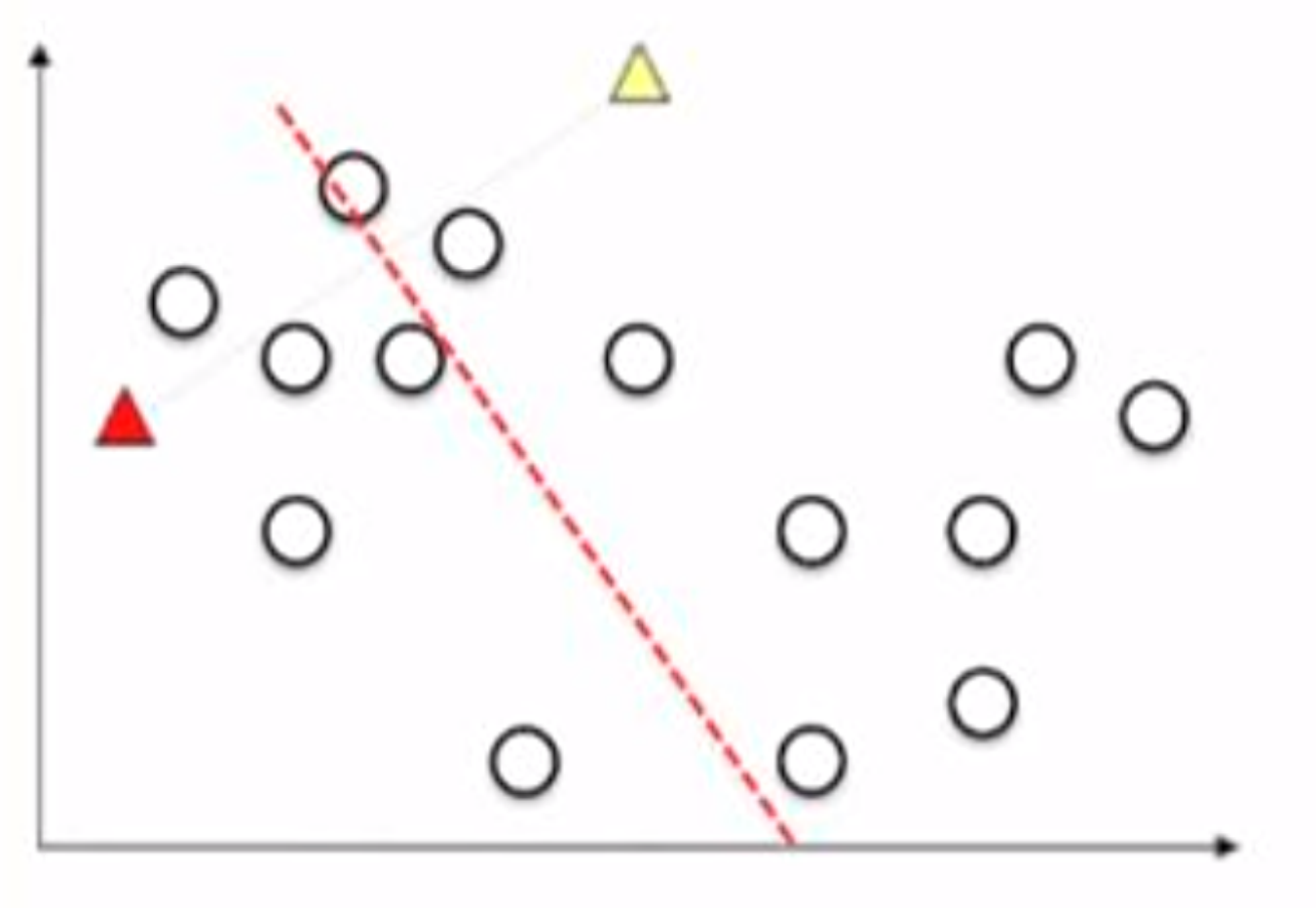
k-Means



[3]

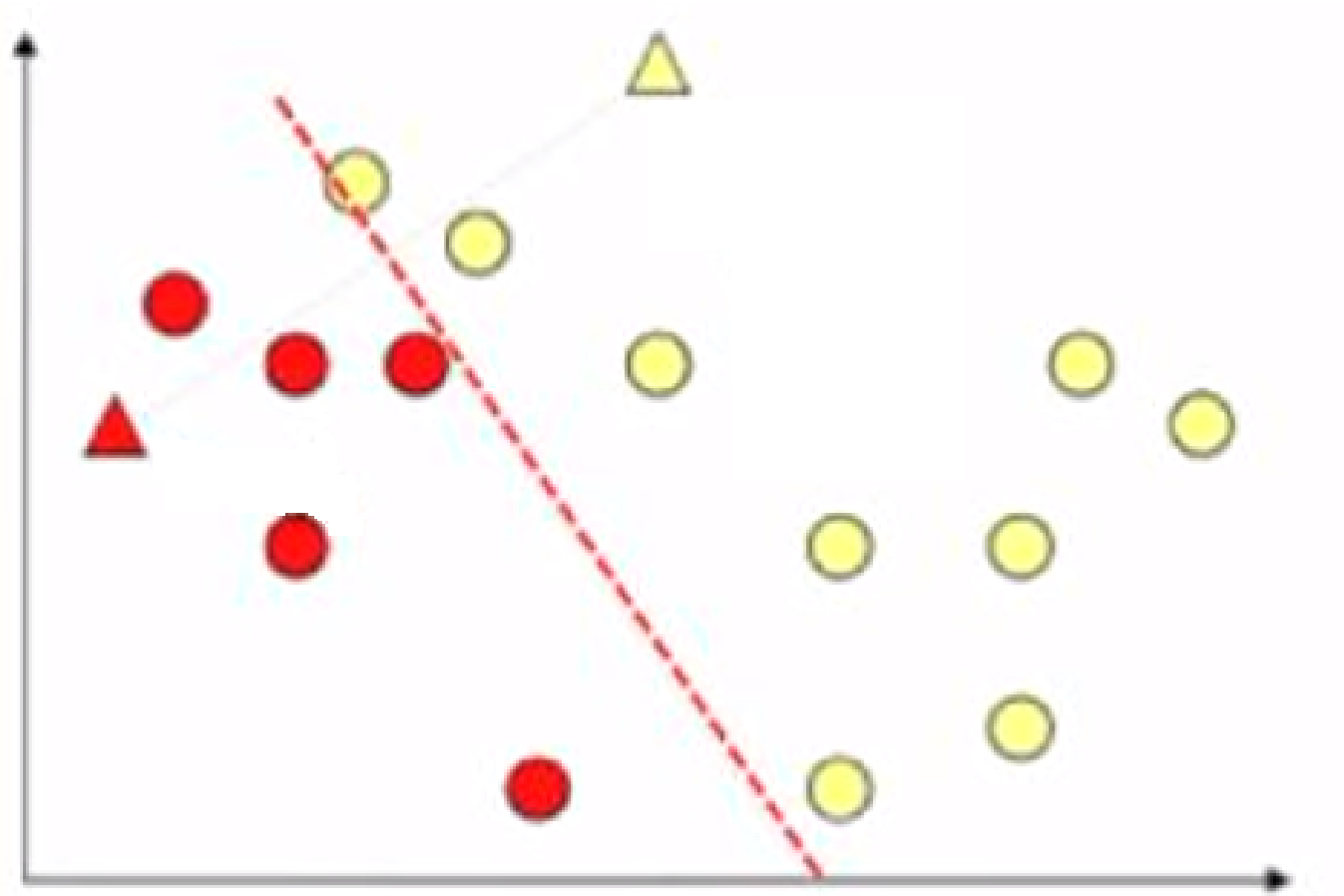
[3] https://www.youtube.com/watch?v=_aWzGGNrcic&t=312s by Victor Lavrenko

Abbildung: Zufällig k (hier 2) Centroids bestimmen



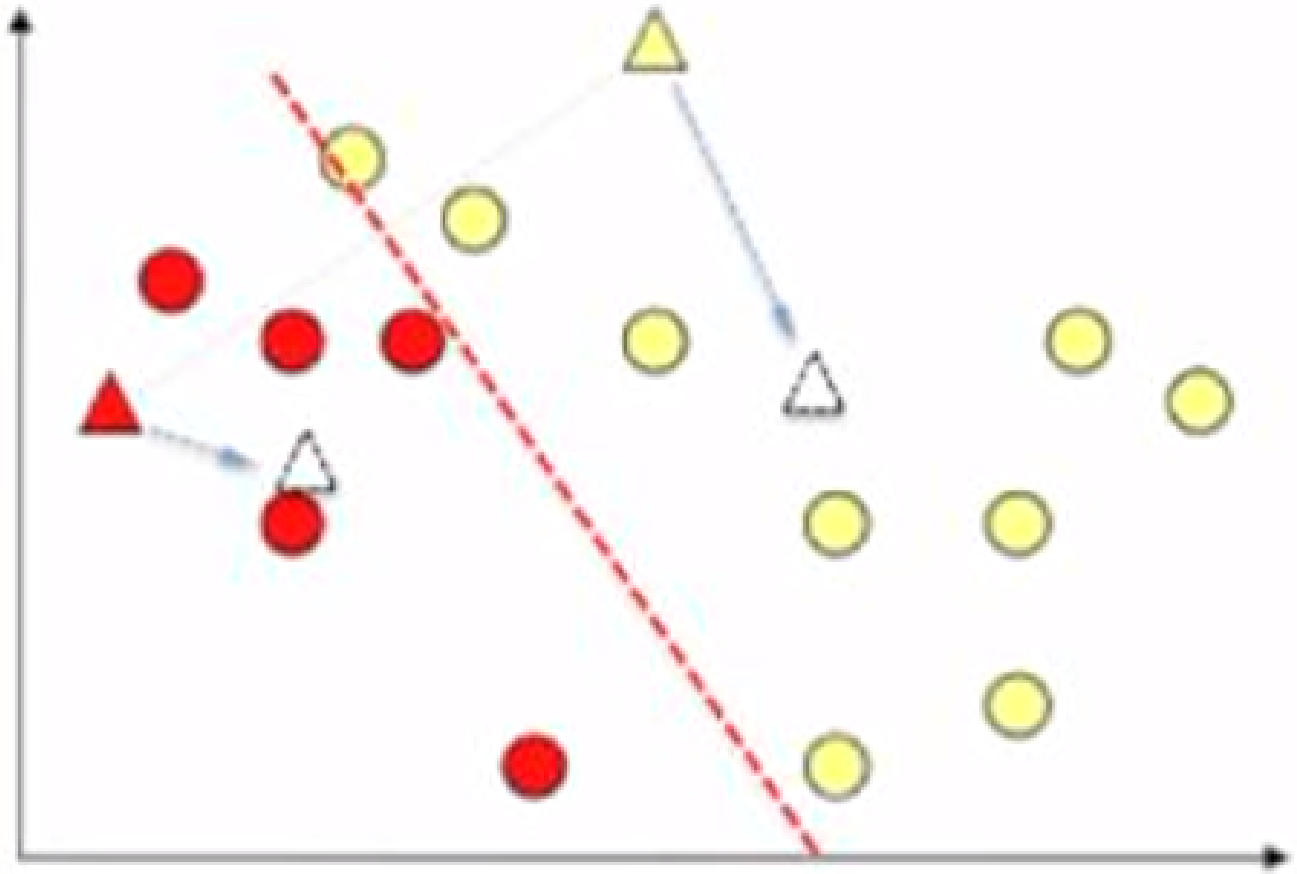
7

Abbildung: Jeder Datenpunkt wird seinem nächsten Centroid zugewiesen



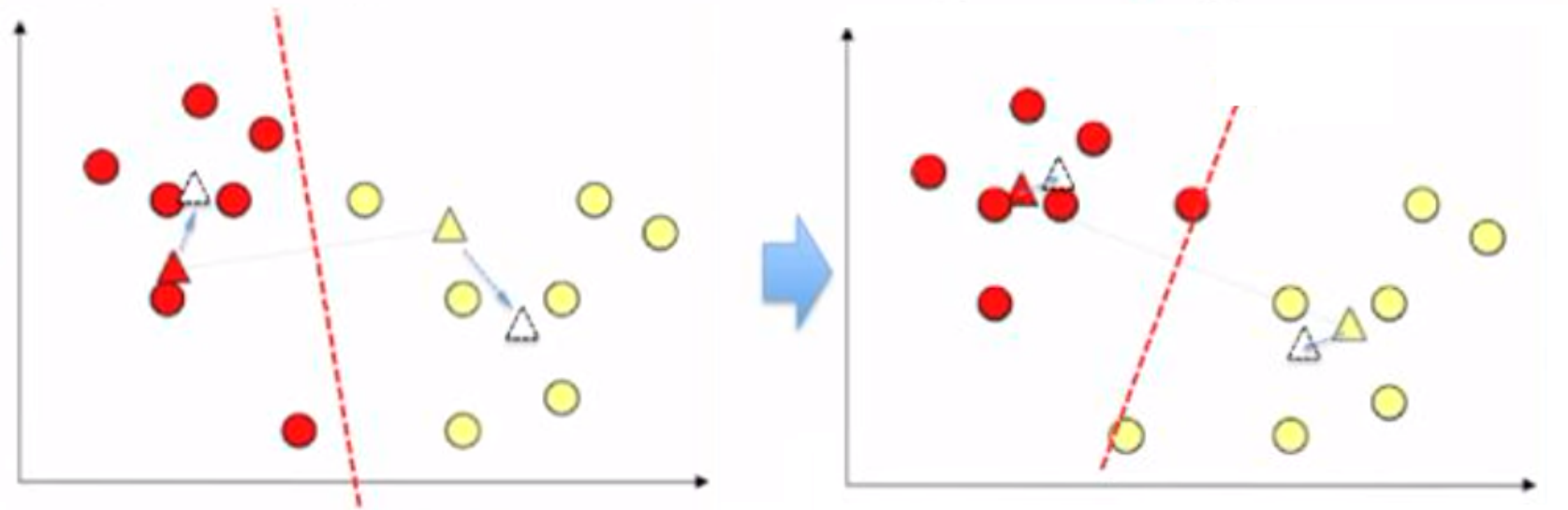
7

Abbildung: Die Centroide wandern in den Mittelpunkt ihrer zugewiesenen Punkte. Dann wird jeder Punkt neu zugewiesen.



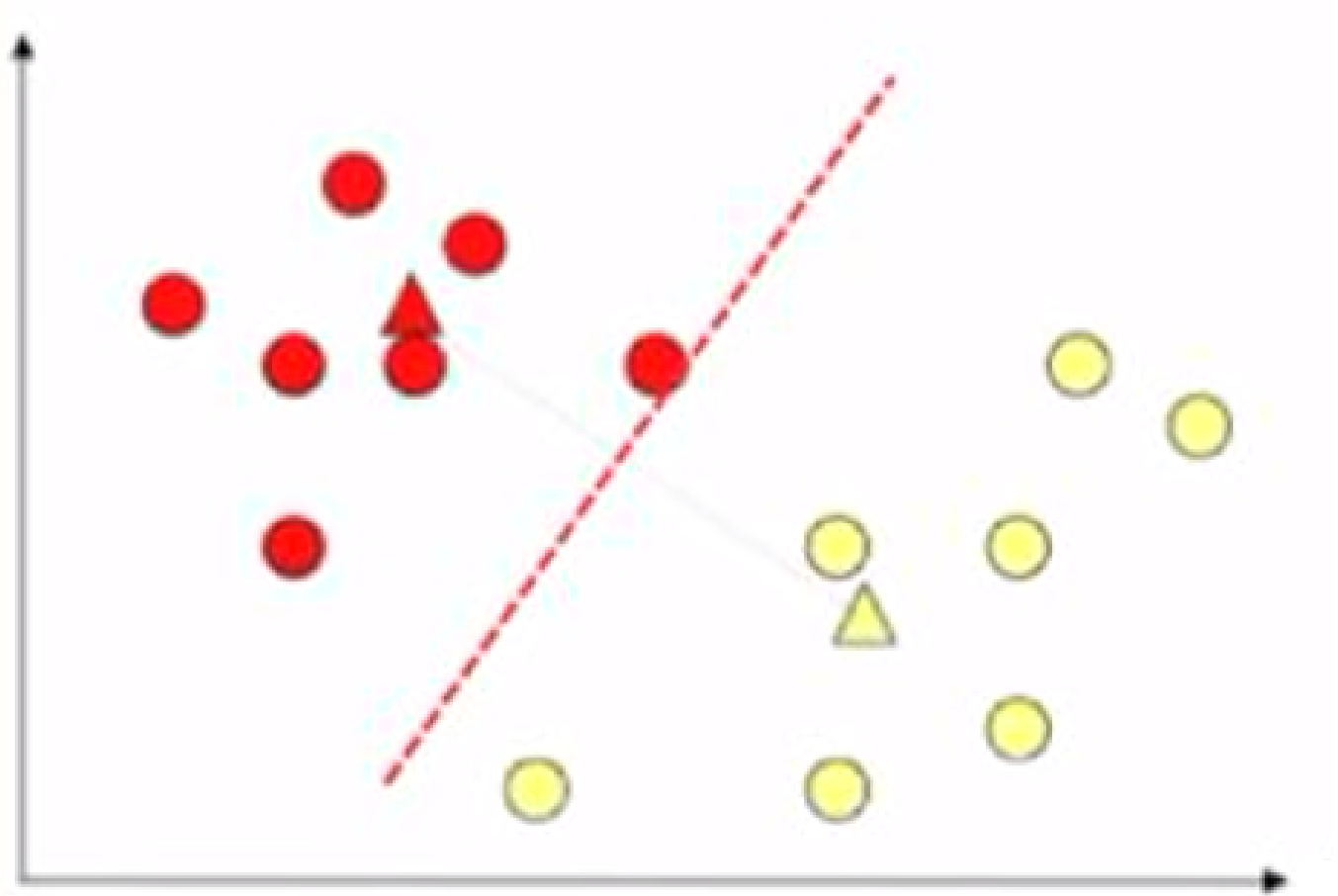
7

Abbildung: Dieser Vorgang wir bis zur Konvergenz wiederholt



7

Abbildung: Die Datenpunkte wurden in k=2 Cluster unterteilt



7

k-Means: Vor- und Nachteile

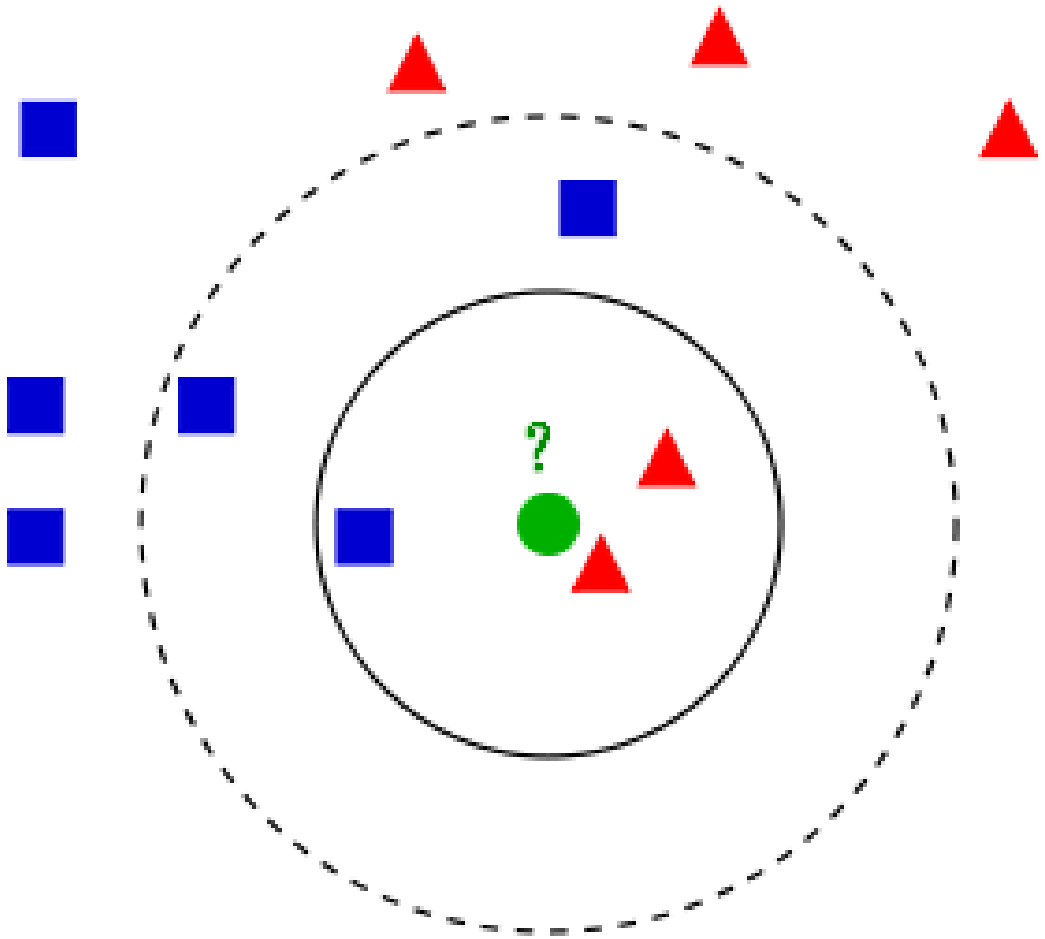
+ Sehr leicht zu implementieren
+ Gut nachvollziehbar
+ Oft gute Ergebnisse

– Beruht auf der Annahme: Distanz = Ähnlichkeit
– k muss gewählt werden

Einfachste Heurisitk: Distanz

- unter der selben Annahme:
  Distanz ist ein Indiz für Kompressionsalgorithmus.
- Es existieren bereits Gruppen mit ein Label
- Ein Label für einen neuen Datenpunk kann bestimmt werden
- **k-Nearest-Neighbour Algorithmus** kann verwendet werden

Abbildung: k-Nearest-Neigbour Algorithmus für k=3 mit zwei Klassen

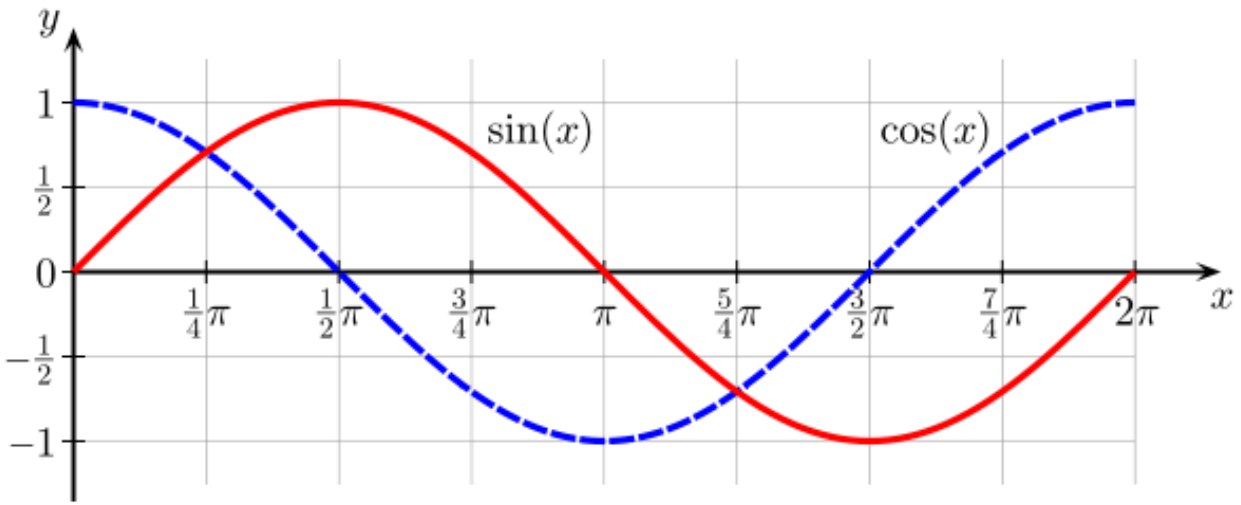


4

[4] https://commons.wikimedia.org/wiki/File:KnnClassification.svg

Wie richtig ist unsere Annahme?

Abbildung: Daten mit ähnlicher Struktur können eine große Distanz besitzen



5

[5] https://de.wikipedia.org/wiki/Sinus_und_Kosinus#/media/File:Sine_cosine_one_period.svg

Vergleich Bilderkennung

Abbildung: Ähnlichkeit durch Vergleichen von Pixeln

6 7

[6] https://www.gala.de
[7] https://www.travelworks.at

Vergleich Bilderkennung

Abbildung: Ähnlichkeit durch Vergleichen von Pixeln

8 9

[8] https://www.gala.de
[9] http://www.trendystickers.dk

Idee: Supervised Learning

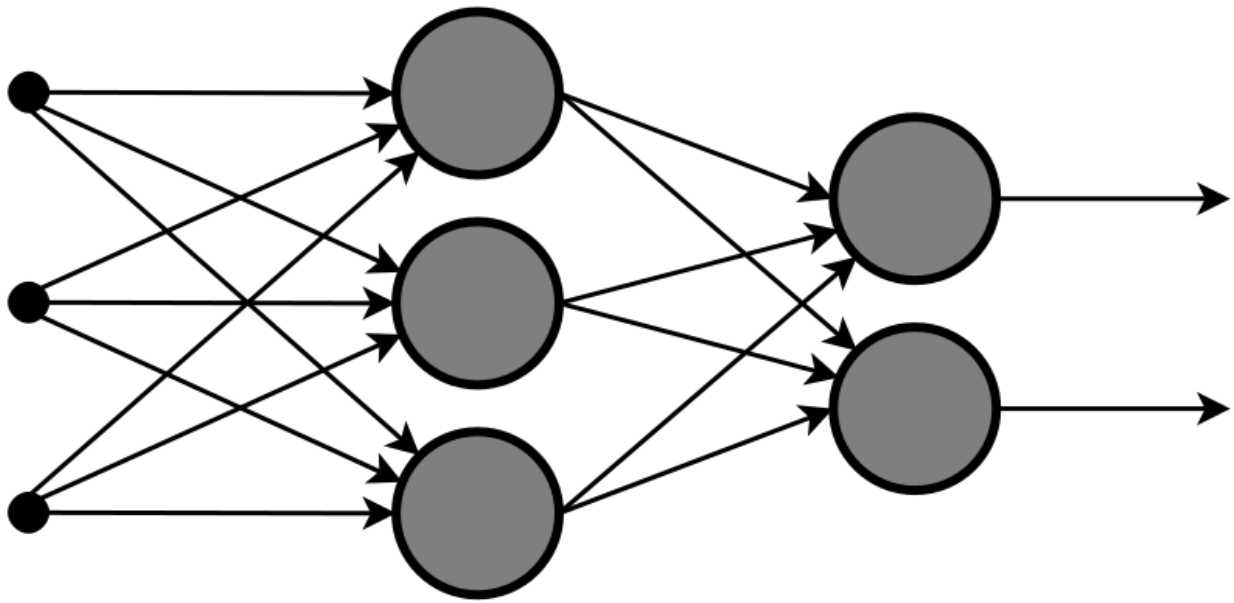
- Verwendung von Target Variable (Label bei Klassifizierung)
- Wenn ein Target existiert, kann bei einer Abweichung dieses Targets ein Fehler berechnet werden
- Dadurch kann das System trainiert (optimiert) werden

Idee: Supervised Learning

- Eine Funktion von Input auf Label wird gelernt
- F(Bild) $\rightarrow$ Tiername bzw.
  F(Datensatz) $\rightarrow$ Kompressionsalgorithmus
- Verschiedene Algorithmen existieren: Entscheidungsbaum, Support Vector Machine, Neuronale Netzwerke, Bayesian Classifiers, ...

Neuronale Netzwerke [3]

Abbildung: Feed Forward Neural Network mit 3 Input- und 2 Output-Knoten

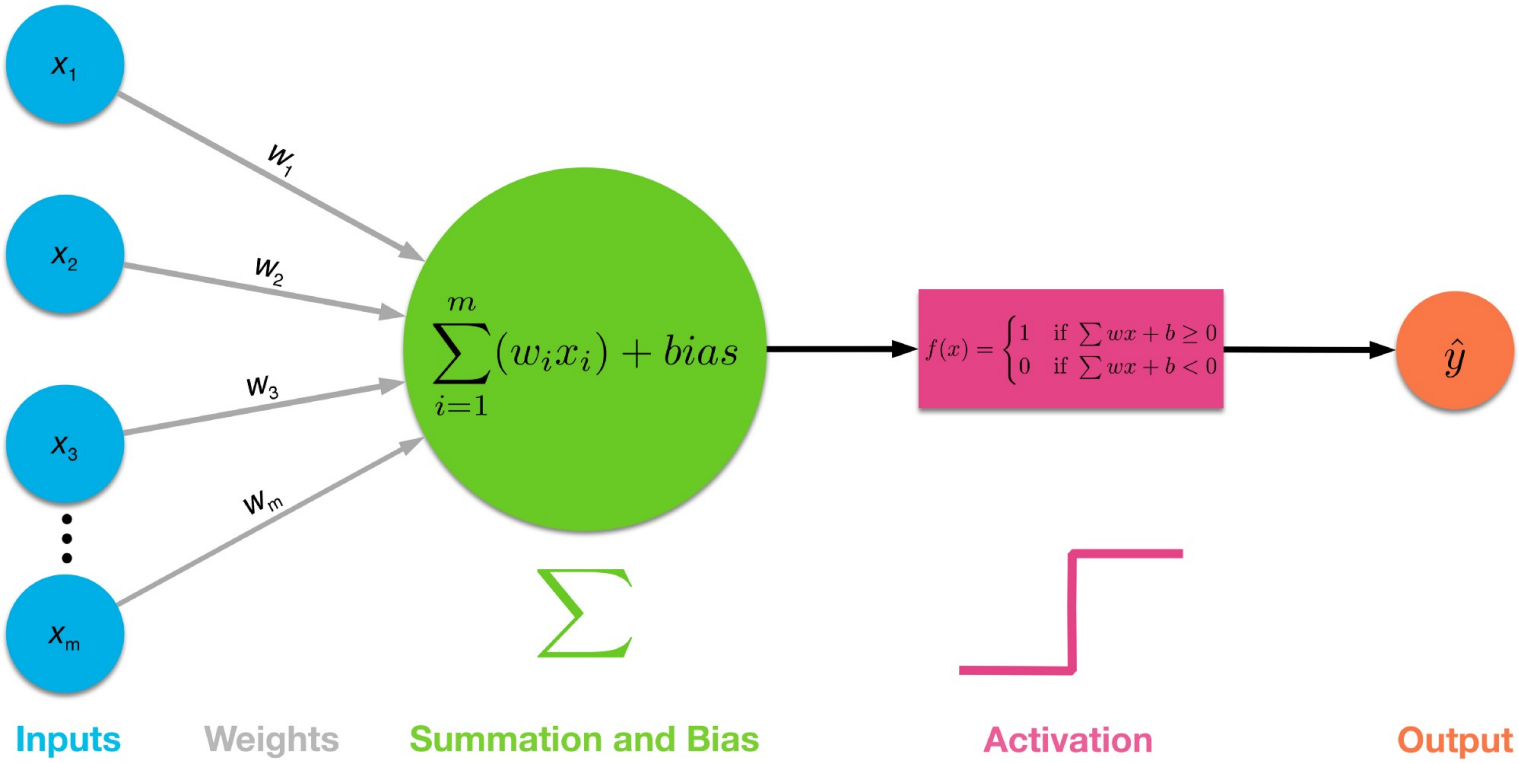


10

[10] https://en.wikibooks.org/wiki/Artificial_Neural_Networks/Feed-Forward_Networks

Neuronale Netzwerke

Abbildung: Perceptron: Veranschaulicht was in jedem Knoten passiert

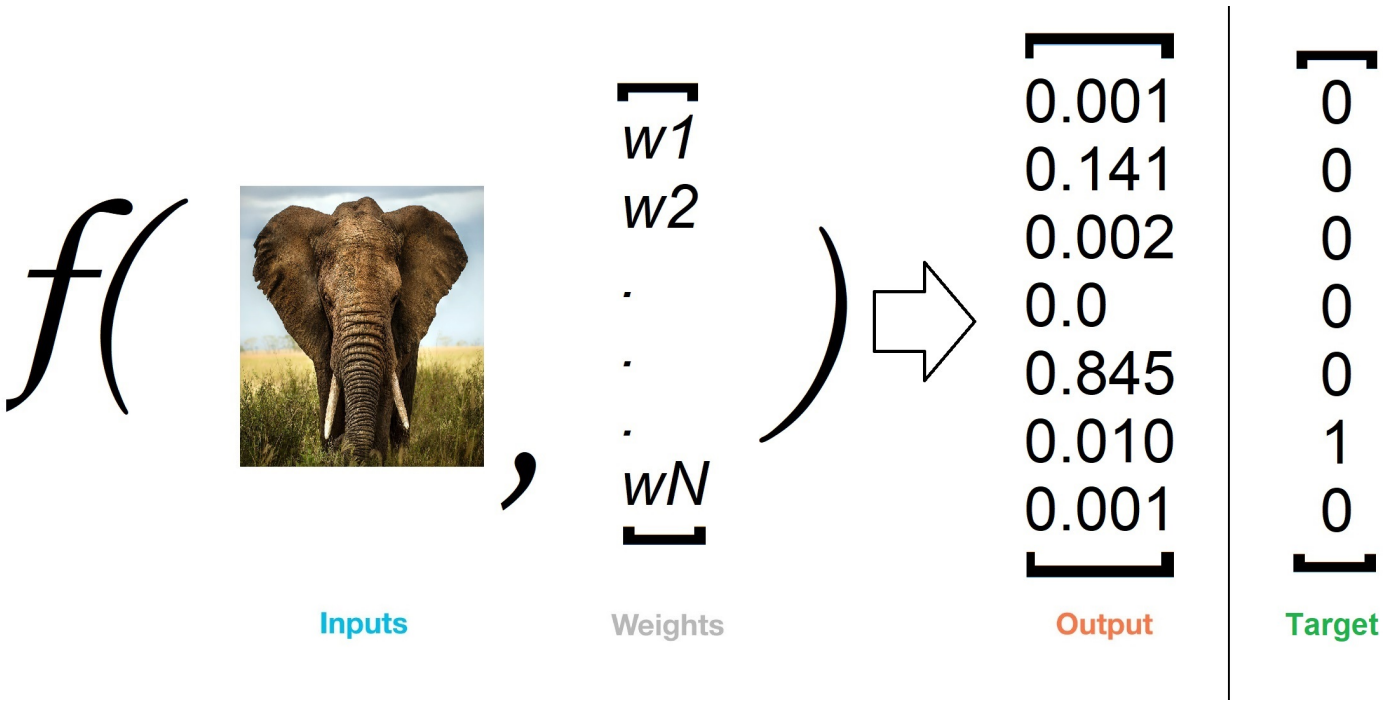


[11]

[11] https://towardsdatascience.com/multi-layer-neural-networks-with-sigmoid-function-deep-learning-for-ro

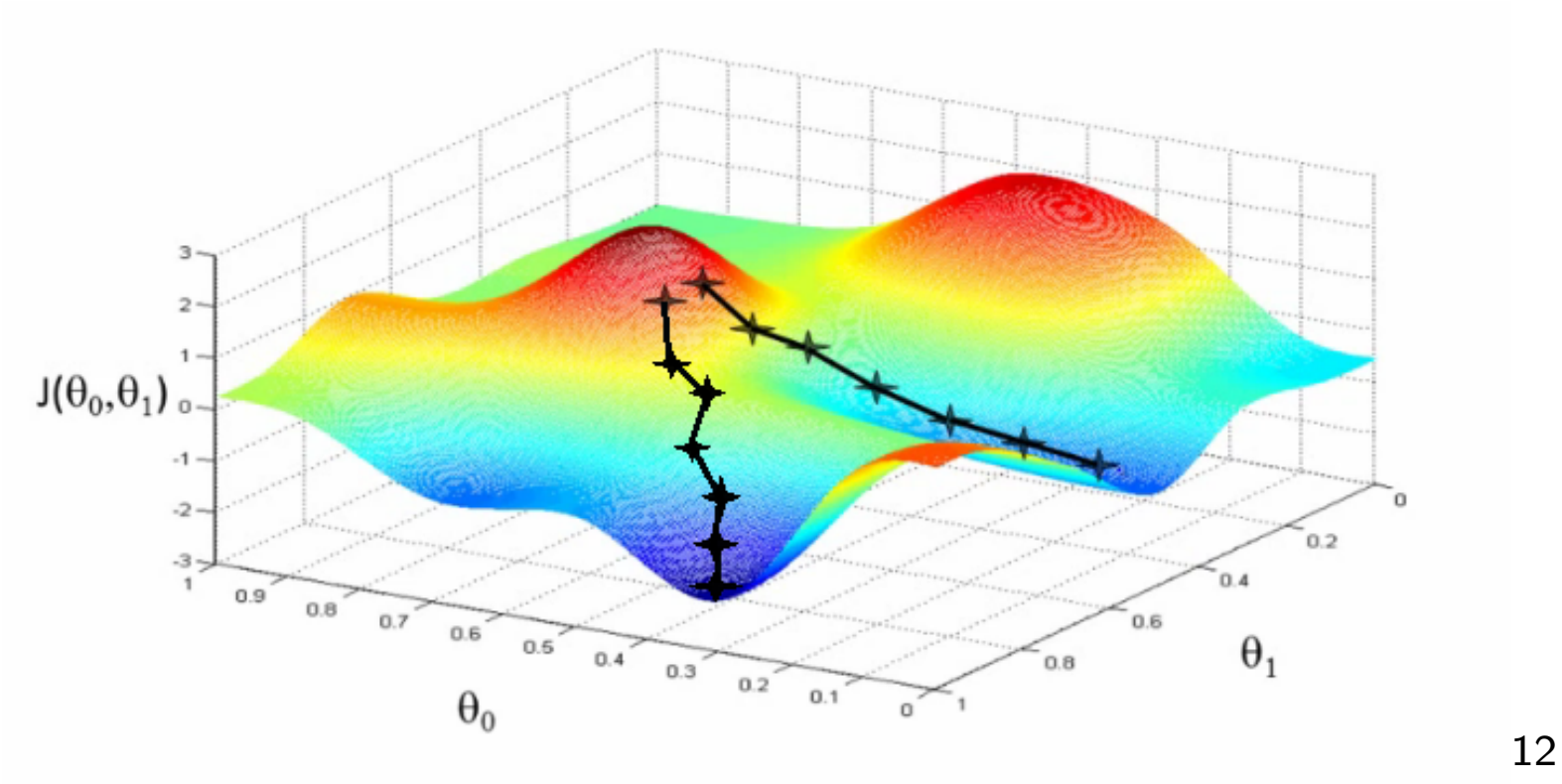
Neuronale Netzwerke

Abbildung: Das Netzwerk lässt sich durch eine Funktion darstellen



Optimierung durch Gradient Decent

Abbildung: Der Fehler in Abhängigkeit von den Gewichten



12

[12] http://blog.datumbox.com/tuning-the-learning-rate-in-gradient-descent/

Beispiel Image Classification [1]

13

[13] http://www.cs.cmu.edu/~aarti/Class/10701_Spring14/slides/

Ergebnis: Neuronale Netzwerke

+ Sehr komplexe Funktionen können gut modelliert werden
+ Generalisieren gut
+ Sind heutzutage leicht zu benutzen (z.B. Keras)[14]

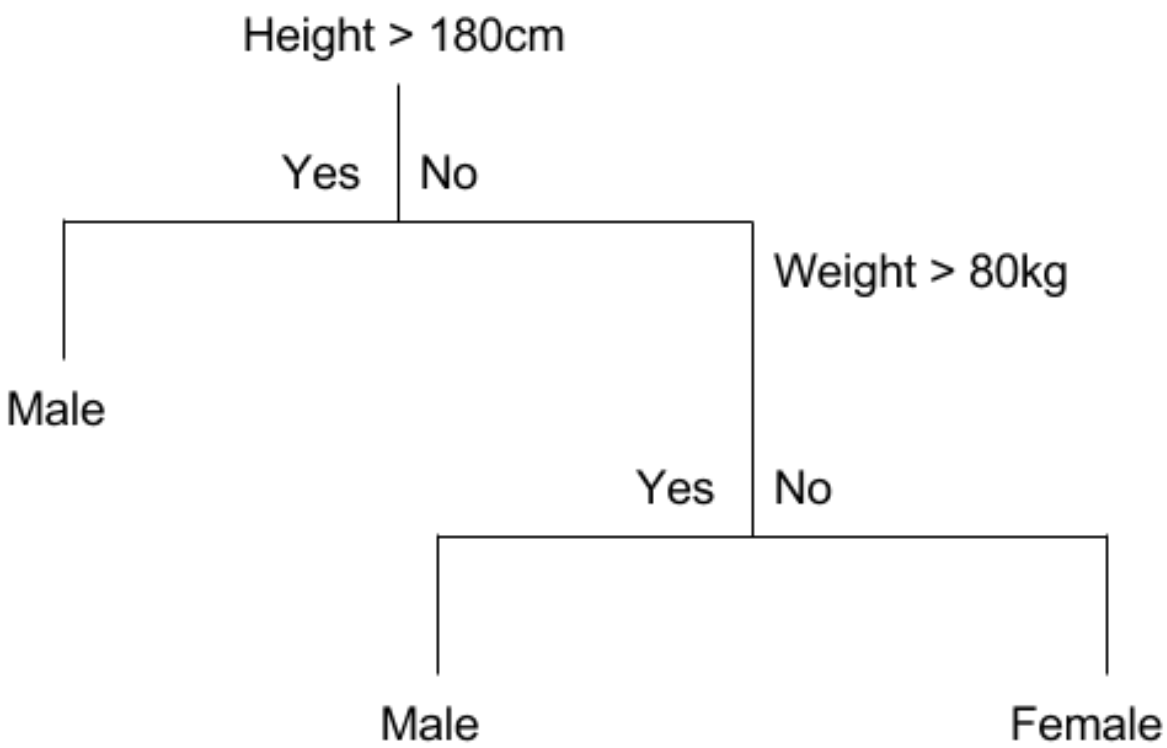
– Relativ viele Hyperparameter existieren
– Black Box
– Kann sehr rechenintensiv sein

[14] https://keras.io/

Decision Tree [2]

- Automatischer Aufbau eines Entscheidungsbaums
- Die Entscheidungsbedingungen werden gelernt
- Wird auch supervised trainiert

Decision Tree [2]



15

[15] https://machinelearningmastery.com/classification-and-regression-trees-for-machine-learning/

Vergleich von Decision Tree zu Neuronalem Netzwerk

\+ Einfacher zu interpretieren und zu verstehen

\+ Keine Normalisierung des Inputs notwendig

\– Probleme können zu komplex werden

Zurück zu Datenkompression

- genug Datensätze wurden getestet und ihre optimalen Kompressionsalgorithmus wurden gefunden
- optimaler Kompressionsalgorithmus als Label für jeden Datensatz
- für neue, unbekannte Daten kann ein Kompressionsalgorithmus vorhergesagt werden

Ergebnis: Supervised Training

+ zuverlässige Vorhersage bei genug Training
+ schnelle Vorhersage für neue Daten
+ erkennt wiederkommende Features und Strukturen in den Daten

– benötigt viele Trainingsdaten um gut zu funktionieren
– Training benötigt Zeit und Resourcen
– Daten müssen die selben Dimensionen haben (cutting, padding)

Zusammenfassung: Decision support System

Verschiedene Ansätze um eine optimale Entscheidung zu finden:

- Oft sind Probleme theoretisch lösbar, aber nur unter Einsatz von sehr viel Zeit
- Clustering kann verwendet werden, um unlabeled Daten zu gruppieren
- Distanz als einfache Heuristik verwenden: einfach und schnell aber kann sehr ungenau sein
- Supervised Learning: Sehr gute Vorhersagen bei genug Trainingsdaten aber Lable nicht immer vorhanden

Literatur

[1] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015.

[2] J. Ross Quinlan. Induction of decision trees. *Machine learning*, 1(1):81–106, 1986.

[3] Jürgen Schmidhuber. Deep learning in neural networks: An overview. *Neural networks*, 61:85–117, 2015.

[4] Pang-Ning Tan, Michael Steinbach, and Vipin Kumar. *Introduction to Data Mining, (First Edition)*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2005.