

Übung 7

Institutsleitung
Prof. Dr.-Ing. J. Becker
Prof. Dr.-Ing. E. Sax
Prof. Dr. rer. nat. W. Stork

Übung zu Informationstechnik II und Automatisierungstechnik – Nathalie Brenner

Prof. Dr.-Ing. Eric Sax

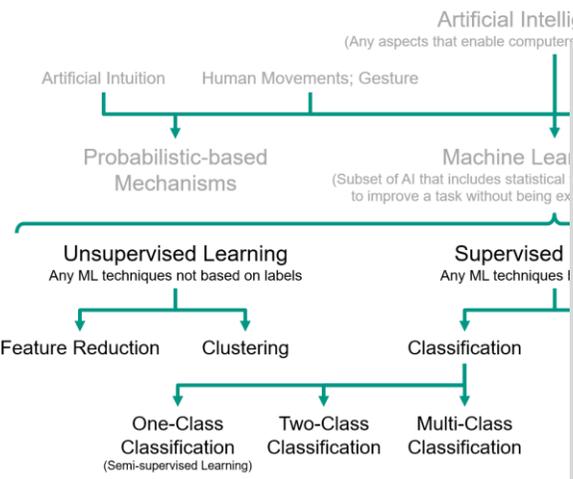


WIEDERHOLUNG ÜBUNG 6



Supervised Learning

Modelbildung – Übersicht über Verfahren



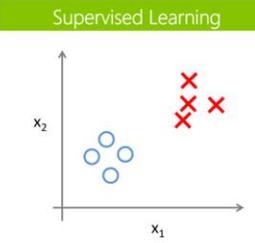
The flowchart shows the hierarchy of AI and ML. AI (Artificial Intelligence) is defined as 'Any aspects that enable computers to mimic human intelligence'. It branches into 'Artificial Intuition' and 'Human Movements; Gesture'. 'Artificial Intuition' leads to 'Probabilistic-based Mechanisms'. 'Human Movements; Gesture' leads to 'Machine Learning (Subset of AI that includes statistical methods to improve a task without being explicitly programmed)'. Machine Learning is further divided into 'Unsupervised Learning (Any ML techniques not based on labels)' and 'Supervised Learning (Any ML techniques based on labels)'. Unsupervised Learning includes 'Feature Reduction' and 'Clustering'. Supervised Learning includes 'Classification', which is further divided into 'One-Class Classification (Semi-supervised Learning)', 'Two-Class Classification', and 'Multi-Class Classification'.

Modeling – Einteilung der Algorithmen

Group by Learning Style

Supervised Learning

- gelabelte Daten
- Lernen/Vorhersagen von Output aus Input-Daten
- Herausforderung:
 - extrapolieren
 - generalisieren
- Beispiele
 - Klassifizierung
 - Regression



The scatter plot shows a 2D coordinate system with axes x_1 and x_2 . There are four blue circles clustered in the lower-left quadrant and four red crosses clustered in the upper-right quadrant, representing two distinct classes of data points.

Unsupervised Learning

ungelabelte Daten

Regression und Klassifikation

Gemeinsamkeiten und Unterschiede

Überwachtes Lernen

- Bei Trainingsdaten ist das Vorhersageattribut bekannt
- Zielgröße neuer Datensätze werden auf Basis des gelernten Modells vorhergesagt

Regressionsprobleme

- Idee
 - Bestimmung eines unbekanntes **numerischen** Attributwertes (ordinal oder kategorisch durch Schwellwertsetzung)
 - Unter Benutzung beliebiger Attributwerte
- Beispiele:
 - Vorhersage von Kosten, Aufwand, etc.
 - Vorhersage von Kundenverhalten (Kündigungszeitpunkt)
 - Vorhersage zu Verkaufszahlen
 - uvm

Klassifikationsprobleme

- Idee
 - Bestimmung eines unbekanntes **kategorischen** Attributwertes (ordinal mit Einschränkungen)
 - Unter Benutzung beliebiger Attributwerte
- Beispiele:
 - Klassifikation von Spam
 - Vorhersage von Kundenverhalten (Kündigung)
 - Vorhersage von Kreditwürdigkeit
 - uvm

Wiederholung Übung 6

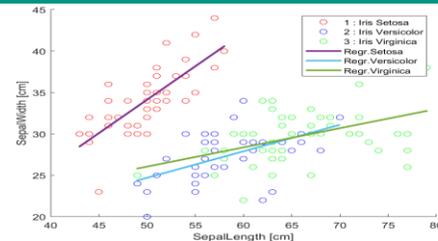
Maschinelle Lernmethoden

Regression

Bestimmung eines unbekannten numerischen Attributwertes

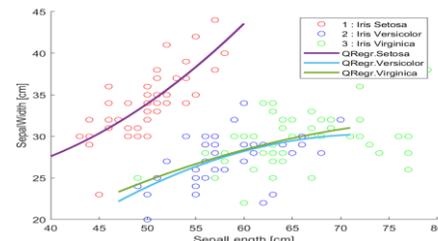
Regressionsgerade

$$y = \frac{S_{xy}}{S_{xx}}x + \bar{y} - \frac{S_{xy}}{S_{xx}}\bar{x}$$



Regressionskurve

$$y = ax^2 + bx + c$$



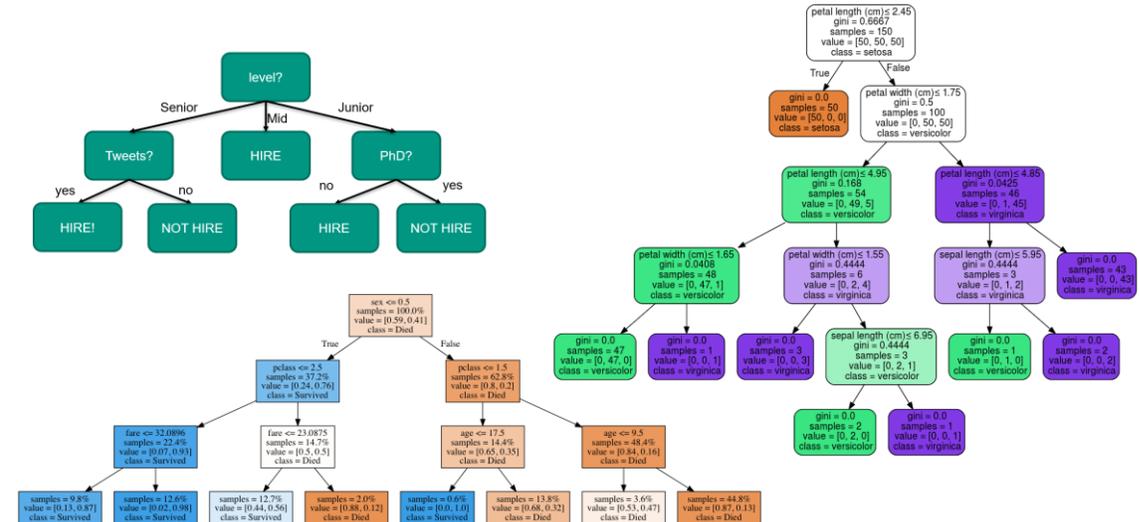
K-fold Cross Validation

Methodenauswahl

Supervised Learning
Methodenauswahl – k-fold Cross Validation

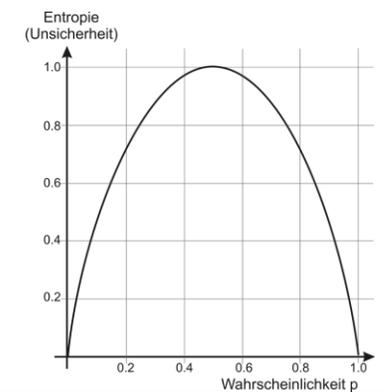
- Die Trainingsdaten werden in k Teile unterteilt
- Auf Basis von k-1 Teilen wird das Modell berechnet und im Anschluss mit dem verbleibenden Teil validiert.
- Wiederholen bis alle Teile einmalig der Validierung dienen
- MSE berechnen und merken
- Wiederholung der Punkte 1.-4. für jede Methode
- Auswahl der Methode mit dem geringsten MSE
- Modellbildung auf Basis der gesamten 70% anhand der ausgewählten Methode

Klassifikation – Decision Tree



Entropie

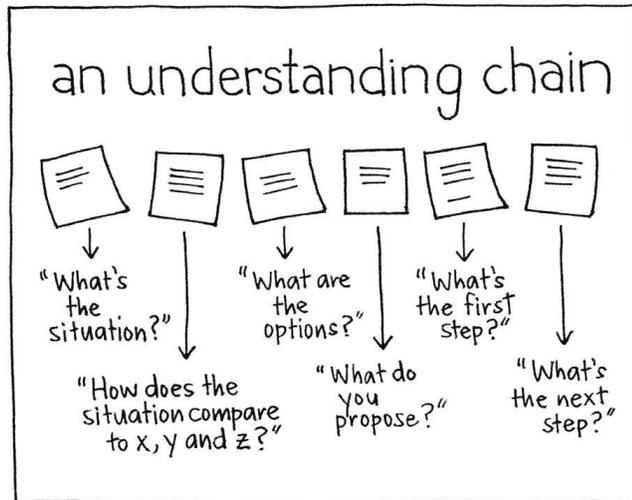
$$H(S) = -p_1 \log p_1 - p_2 \log p_2 - \dots - p_n \log p_n$$



INHALT ÜBUNG 7



Data Understanding



Data Preparation



Modeling



Ziele der heutigen Übung



- Nach der heutigen Übung können Sie....

• ...Ansätze zur Verwaltung und Analyse großer Datenbestände hinsichtlich ihrer Anwendbarkeit und Wirksamkeit einschätzen

1

• ... Modelle berechnen unter Anwendung überwachter maschineller Lernmethoden

2

• ... Merkmale und Eigenschaften von KNN benennen, bewerten und abgrenzen

3

• ... überwachte Lernverfahren hinsichtlich ihrer Performanz bewerten

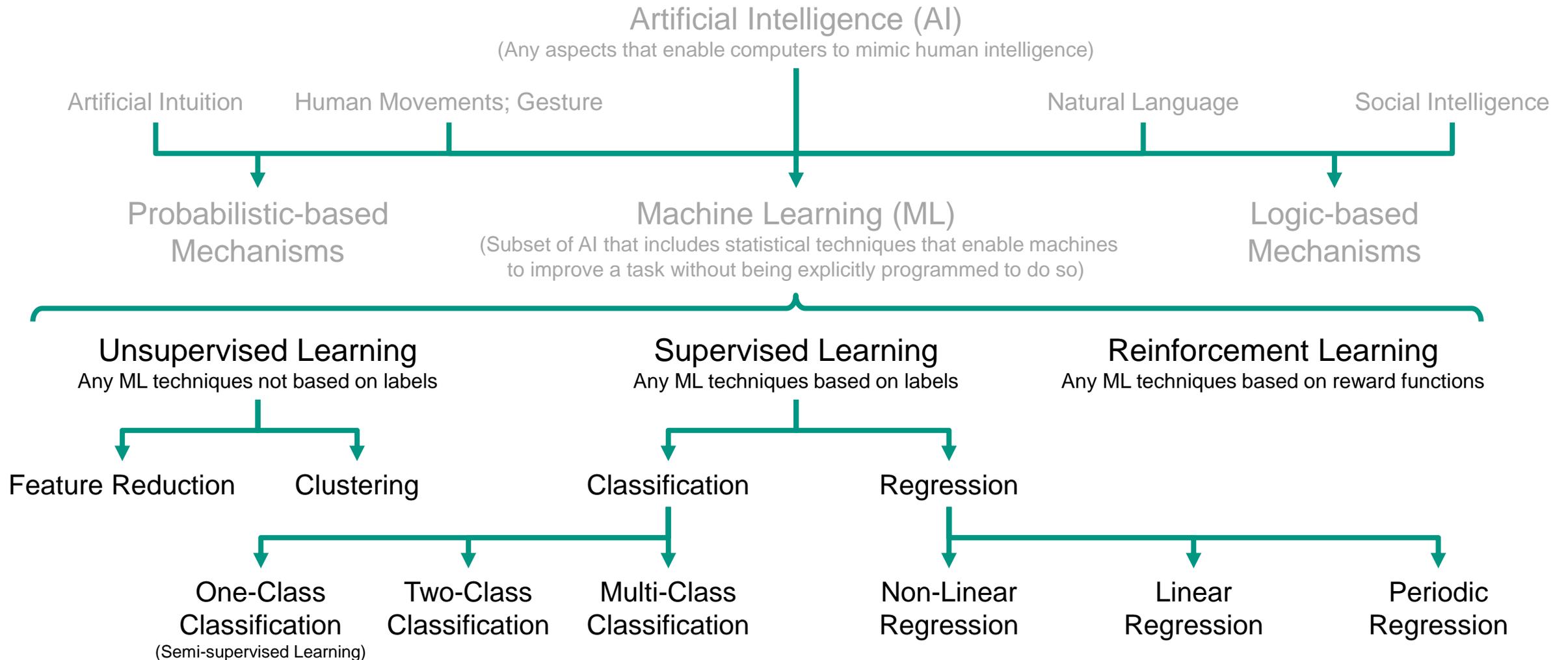
4

• ... Modelle unter Anwendung unüberwachter maschineller Lernverfahren aufstellen

SUPERVISED LEARNING KLASSIFIKATION



Modelbildung – Übersicht über Verfahren



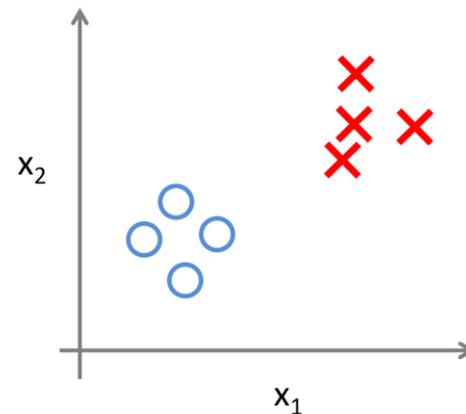
Modeling – Einteilung der Algorithmen

Group by Learning Style

Supervised Learning

- gelabelte Daten
- Lernen/Vorhersagen von Output aus Input-Daten
- Herausforderung:
 - extrapolieren
 - generalisieren
- Beispiele
 - Klassifizierung
 - Regression

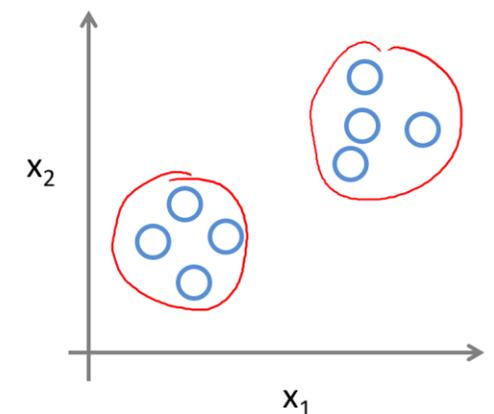
Supervised Learning



Unsupervised Learning

- ungelabelte Daten
- Auffinden von versteckten Strukturen in Daten
- Herausforderung:
 - Subjektiver als SL
 - Validierung
- Beispiele
 - Clustering
 - Dimensionsreduktion

Unsupervised Learning



Regression und Klassifikation

Gemeinsamkeiten und Unterschiede

Überwachtes Lernen

- Bei Trainingsdaten ist das Vorhersageattribut bekannt
- Zielgröße neuer Datensätze werden auf Basis des gelernten Modells vorhergesagt

Regressionsprobleme

- Idee
 - Bestimmung eines unbekanntes **numerischen** Attributwertes (ordinal oder kategorisch durch Schwellwertsetzung)
 - Unter Benutzung beliebiger Attributwerte
- Beispiele:
 - Vorhersage von Kosten, Aufwand, etc.
 - Vorhersage von Kundenverhalten (Kündigungszeitpunkt)
 - Vorhersage zu Verkaufszahlen
 - uvm

Klassifikationsprobleme

- Idee
 - Bestimmung eines unbekanntes **kategorischen** Attributwertes (ordinal mit Einschränkungen)
 - Unter Benutzung beliebiger Attributwerte
- Beispiele:
 - Klassifikation von Spam
 - Vorhersage von Kundenverhalten (Kündigung)
 - Vorhersage von Kreditwürdigkeit
 - uvm

Klassifikation - Algorithmen

K-Nearest-Neighbors

- K-Nearest-Neighbor/Nächste-Nachbarn-Klassifikation

- Wahrscheinlichkeitsdichtefunktion
- Klassenzuordnung über k nächste Nachbarn

- **Einfache** Nearest-Neighbor Klassifikation (1NN)

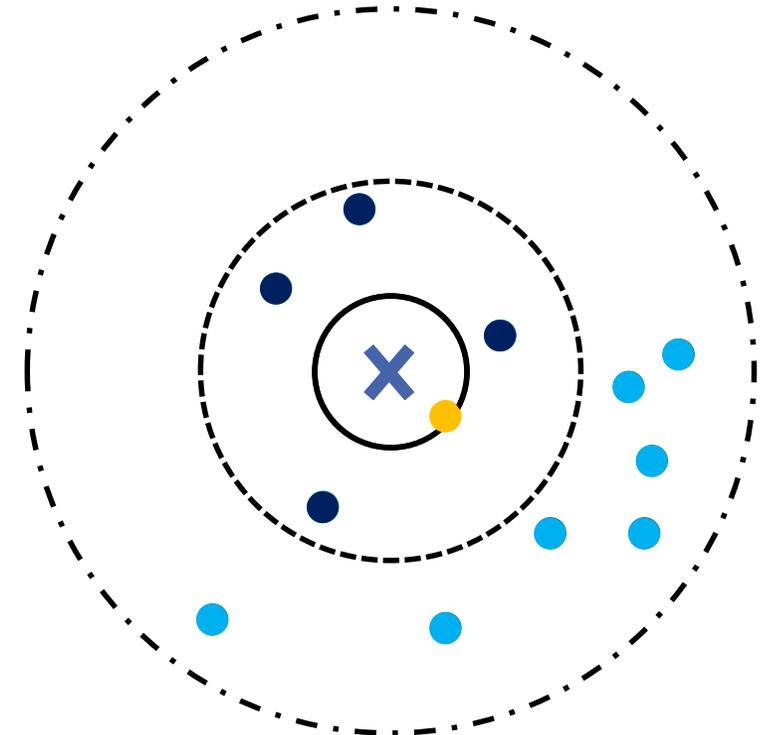
Bestimmung nächster Nachbar über Distanz d

$$d(x, x_1) = \min(d(x, x_1))$$

- **K**- Nearest Neighbors Klassifikation (KNN)

Wahl des Parameters k

- Zu kleines k :
hohe Sensitivität gegenüber Ausreißern
- Zu großes k :
viele Objekte aus anderen Clustern in Entscheidungsmenge
- Mittleres k :
höchste Klassifikationsgüte



Klassifikation - Algorithmen

K-Nearest-Neighbors

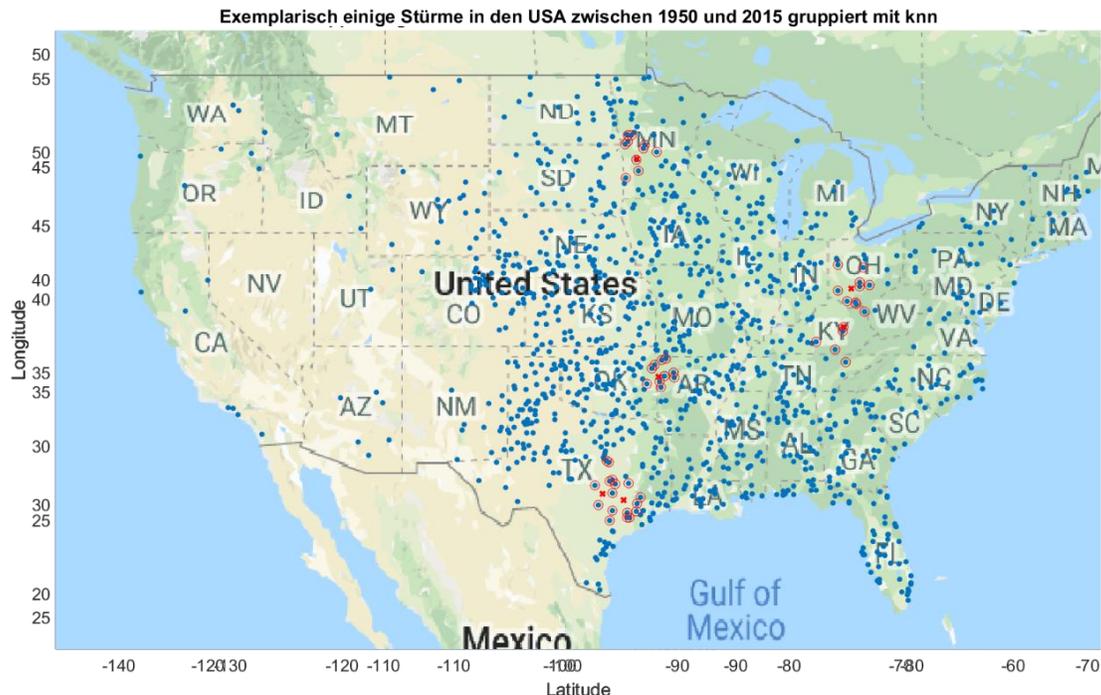
■ K-Nearest-Neighbor Klassifikation (k-NN)

Bestimmung der nächsten Nachbarn über

$$d(x, x_1) \leq d(x, x_2) \leq \dots \leq d(x, x_n)$$

x der Klasse zugeordnet, die unter k nächsten Nachbarn am häufigsten vorkommt

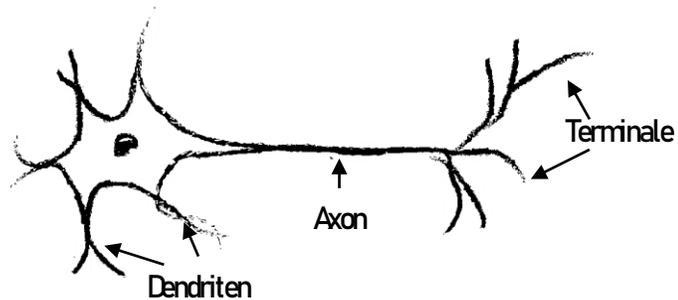
Klassenwahrscheinlichkeit: $\hat{P}(y = g|x) = 1/k \sum_{i=1}^k I(y_i = g)$
 y_i : Klassenzugehörigkeit i -ter Nachbar
 g : relative Häufigkeit einer Klasse



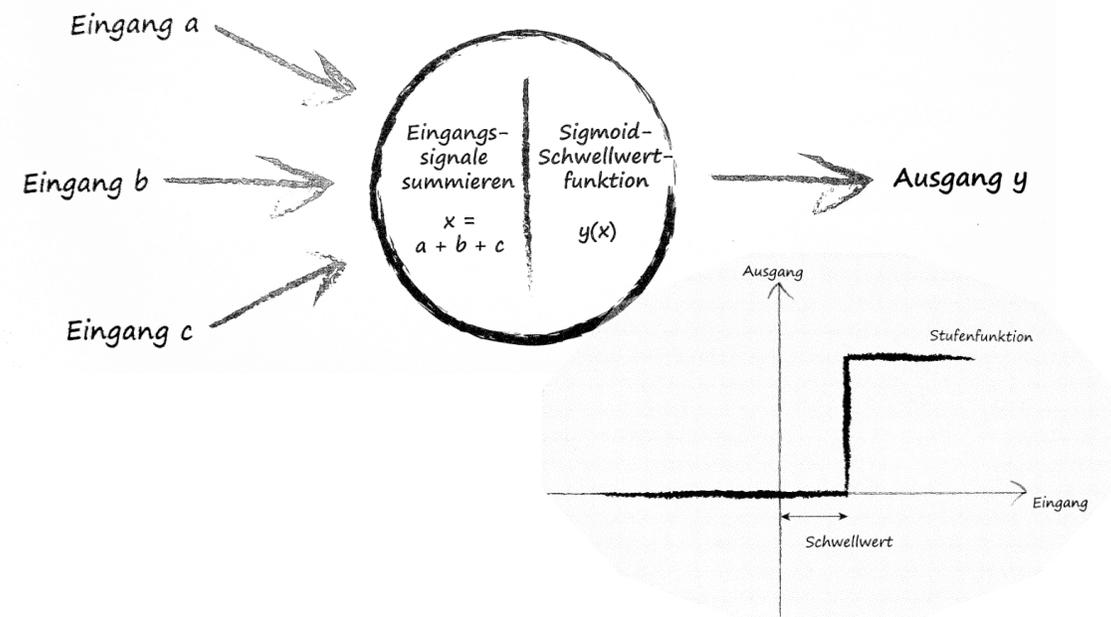
Klassifikation

Künstliche Neuronale Netze - Einführung

- Die Idee von Künstlichen Neuronalen Netzen beruht auf den Neuronennetzen des menschlichen Gehirns und finden Anwendung in zahlreichen Gebieten, wie z.B. Handschrifterkennung, Gesichtsdetektion, uvm.
- Aufteilung in zwei Teilbereiche
 - KNN, die modelliert werden um menschliches Verhalten nachzuahmen/ zu verstehen
 - KNN, modelliert um ein konkretes Anwendungsproblem aus Bereichen, wie beispielsweise der Statistik, zu lösen



- KNN bestehen aus
 - Input-Units: Signale von der „Außenwelt“
 - Hidden-Units: interne Repräsentation der „Außenwelt“, Eingang berechnet sich aus Übertragungsfunktion, Ausgang wird mit Aktivierungsfunktion belegt
 - Output-Units: Signale an die „Außenwelt“



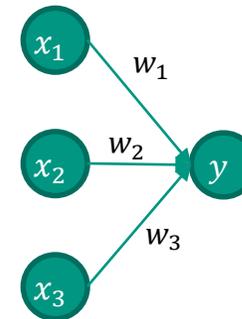
Künstliche Neuronale Netze

Übertragungs- & Aktivierungsfunktionen

- Perceptron:
einzelnes künstliches Neuron mit anpassbaren Gewichten und Schwellenwerten

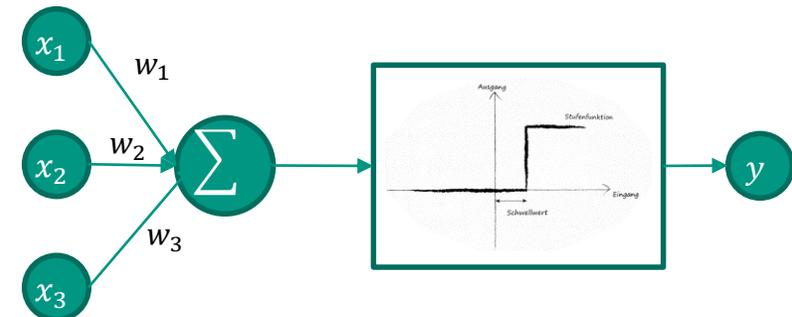
- Übertragungsfunktion Perceptron, ohne Schwellenwert:

$$y = \sum_{i=1}^n w_i x_i$$



- Übertragungsfunktion Perceptron, mit Schwellenwert:

$$y = \phi\left(\sum_{i=1}^n w_i x_i\right)$$



Künstliche Neuronale Netze

Übertragungs- & Aktivierungsfunktionen

- Aktivierungsfunktion stellt einen Hyperparameter dar
 - Hyperparameter: „ein Parameter, der zur Steuerung des Trainingsalgorithmus verwendet wird und dessen Wert im Gegensatz zu anderen Parametern nicht im eigentlichen Training des Modells gelernt wird“.
 - Aktivierungsfunktion: „eine Funktion, die das Eingangssignal übernimmt, ein Ausgangssignal generiert und dabei aber eine Art Schwellwert berücksichtigt“

- Verschiedene Arten von Aktivierungsfunktionen:

- Stufenfunktion

$$y = \begin{cases} 0: x < 0 \\ 1: x \leq 1 \end{cases}$$

- Sigmoid-Funktion

$$y = \frac{1}{1+e^{-x}}$$

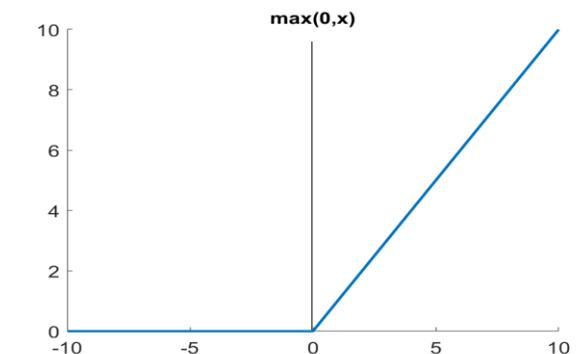
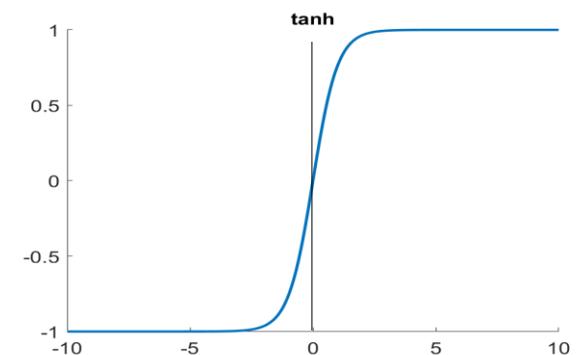
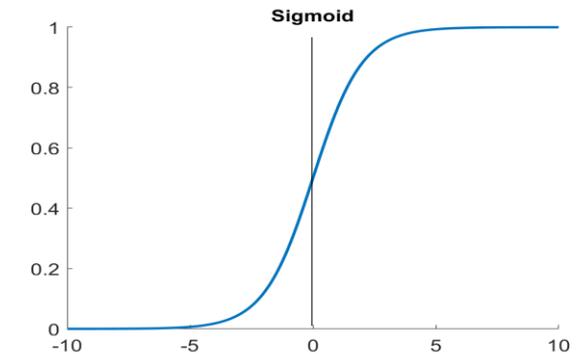
- Tangens-Hyperbolicus-Funktion

$$y = \tanh(x)$$

- Maximum-Funktion

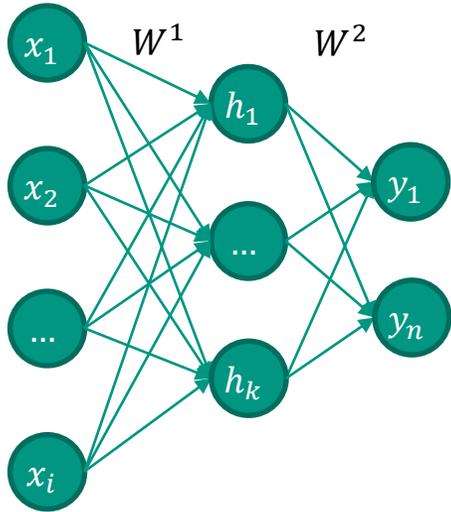
$$y = \max(0, x)$$

- uvm



Künstliche Neuronale Netze

Feedforward NN - Berechnung



$$\bar{x} = [x_1 \ x_2 \ \dots \ x_i]$$

$$W^1 = \begin{bmatrix} w_{11}^1 & \dots & w_{1k}^1 \\ \dots & \dots & \dots \\ w_{i1}^1 & \dots & w_{ik}^1 \end{bmatrix}$$

$$\bar{h}' = [h'_1 \ h'_2 \ \dots \ h'_k] = \bar{x} * W^1 = [x_1 \ x_2 \ \dots \ x_i] * \begin{bmatrix} w_{11}^1 & \dots & w_{1k}^1 \\ \dots & \dots & \dots \\ w_{i1}^1 & \dots & w_{ik}^1 \end{bmatrix}$$

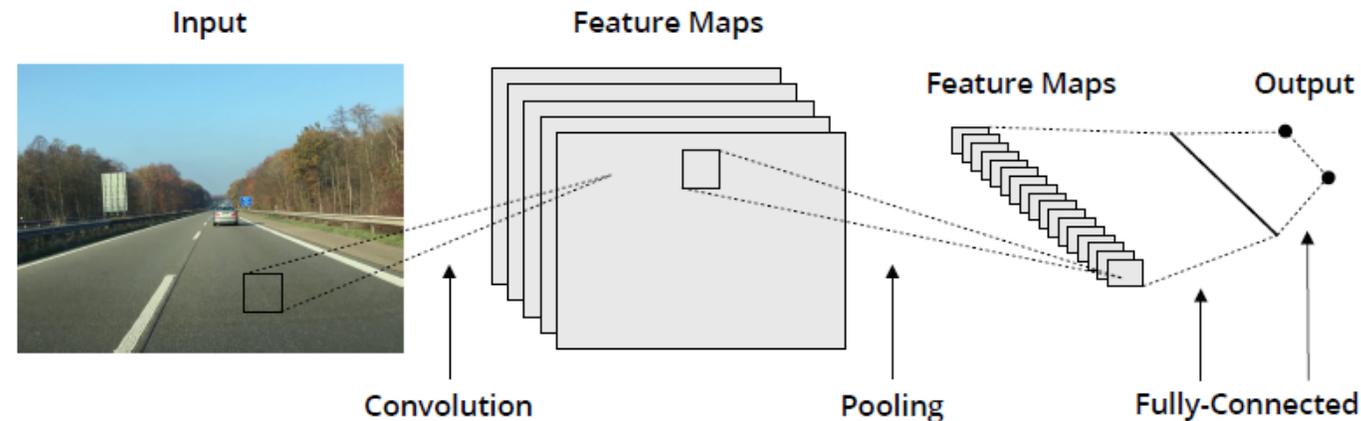
$$\bar{h} = \phi(\bar{h}') = \phi(\bar{x} * W^1), \text{ mit } \phi \text{ als Aktivierungsfunktion}$$

$$\bar{y} = [y_1 \ \dots \ y_n] = \bar{h} * W^2 = \phi([x_1 \ x_2 \ \dots \ x_i] * \begin{bmatrix} w_{11}^1 & \dots & w_{1k}^1 \\ \dots & \dots & \dots \\ w_{i1}^1 & \dots & w_{ik}^1 \end{bmatrix}) * \begin{bmatrix} w_{11}^2 & \dots & w_{1n}^2 \\ \dots & \dots & \dots \\ w_{k1}^2 & \dots & w_{kn}^2 \end{bmatrix}$$

Künstliche Neuronale Netze

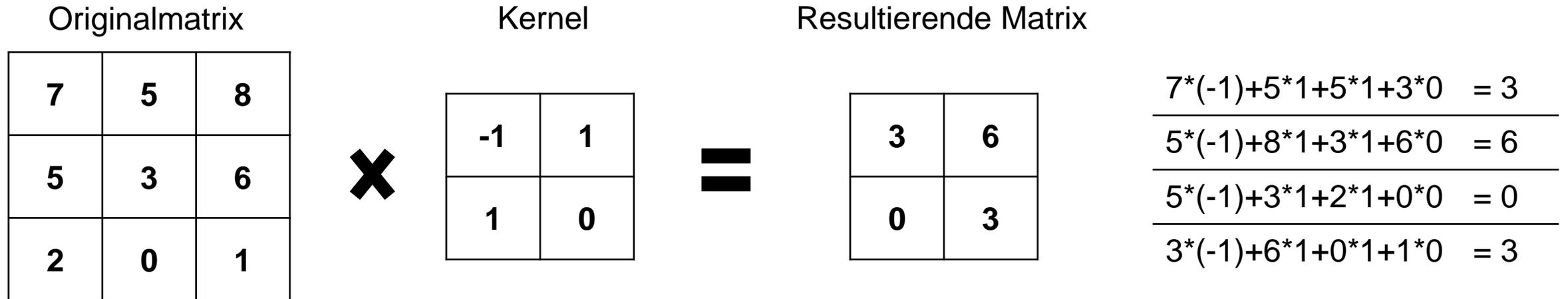
CNN – faltendes künstliches neuronales Netz

- Convolutional Neural Network besteht aus unterschiedlichen Layern
 - Convolutional Layer
Neuronenaktivität wird über diskrete Faltung berechnet
 - Pooling Layer
Verwerfen überflüssiger Informationen
 - Fully-connected Layer
abschließende Klassifikation



Künstliche Neuronale Netze

CNN - Faltung



■ Faltung

Kernelmatrix wird mit vordefinierter Schrittgröße („stride“) über die Originalmatrix geschoben

■ Padding

Je nach Schrittgröße kann die Kernelmatrix über den Rand hinaus geschoben werden. Der Umgang mit dieser Situation wird „Padding“ genannt.

VALID-Padding: kein Überschreiten der Ränder

SAME-Padding: z.B. mit zero-Padding: Der Originalmatrix wird außerhalb der Ränder „0“ zugewiesen“

Klassifikation

Bewertung des trainierten Modells - Confusion Matrix

$$TPR = \frac{\sum \text{True positives}}{\sum \text{Condition positive}}$$

$$FPR = \frac{\sum \text{False positives}}{\sum \text{Condition negative}}$$

$$FNR = \frac{\sum \text{False negatives}}{\sum \text{Condition positive}}$$

$$TNR = \frac{\sum \text{True negatives}}{\sum \text{Condition negative}}$$

		Actual value	
		Yes	No
Predicted value	Positive	True positive (TP) „Vorhersage richtig, Objekt vorhanden“ 	False positive (FP) „Vorhersage falsch, Objekt nicht vorhanden“ 
	Negative	False negative (FN) „Vorhersage falsch, Objekt vorhanden“ 	True negative (TN) „Vorhersage richtig, Objekt nicht vorhanden“ 

		„Legende“
True		Algorithmus Vorhersage richtig
False		Algorithmus Vorhersage falsch
Predictive value:		<u>Entscheidung</u> des Algorithmus
		Pos Gesuchtes Objekt (Husky)
		Neg Nicht das gesuchte Objekt (kein Husky)
Actual value		Yes: gesuchtes Objekt dargestellt No: Gesuchtes Objekt nicht dargestellt

Gütekriterium:

Klassifikationsrate: $P(\text{richtig klassifiziert}) = \frac{t_p + t_n}{n_{ges}}$
 (accuracy) $P(\text{falsch klassifiziert}) = \frac{f_p + f_n}{n_{ges}}$

True positive	False positive
False negative	True negative

Confusion Matrix

Zwischenübung



$$\text{TPR} = \frac{\sum \text{True positives}}{\sum \text{Condition positive}}$$

$$\text{FPR} = \frac{\sum \text{False positives}}{\sum \text{Condition negative}}$$

$$\text{FNR} = \frac{\sum \text{False negatives}}{\sum \text{Condition positive}}$$

$$\text{TNR} = \frac{\sum \text{True negatives}}{\sum \text{Condition negative}}$$

<p>True positive (TP) „Vorhersage richtig, Objekt vorhanden“</p> <p>$t_p = 80$</p>	<p>False positive (FP) „Vorhersage falsch, Objekt nicht vorhanden“</p> <p>$f_p = 5$</p>
<p>False negative (FN) „Vorhersage falsch, Objekt vorhanden“</p> <p>$f_n = 10$</p>	<p>True negative (TN) „Vorhersage richtig, Objekt nicht vorhanden“</p> <p>$t_n = 50$</p>

TPR = ?

FPR = ?

FNR = ?

TNR = ?

Untersuchte Bilder

n_{ges}

145

Tatsächlich Husky

n_{Husky}

90

Tatsächlich kein Husky

$n_{noHusky}$

55

Alg behauptet Husky

n_{AHusky}

85

Alg behauptet kein Husky

$n_{AnoHusky}$

60

Berechnen Sie:

- TPR
- FPR
- FNR
- TNR

Confusion Matrix

Zwischenübung - Lsg



$$\text{TPR} = \frac{\sum \text{True positives}}{\sum \text{Condition positive}}$$

$$\text{FPR} = \frac{\sum \text{False positives}}{\sum \text{Condition negative}}$$

$$\text{FNR} = \frac{\sum \text{False negatives}}{\sum \text{Condition positive}}$$

$$\text{TNR} = \frac{\sum \text{True negatives}}{\sum \text{Condition negative}}$$

<p>True positive (TP) „Vorhersage richtig, Objekt vorhanden“</p> <p>$t_p = 80$</p>	<p>False positive (FP) „Vorhersage falsch, Objekt nicht vorhanden“</p> <p>$f_p = 5$</p>
<p>False negative (FN) „Vorhersage falsch, Objekt vorhanden“</p> <p>$f_n = 10$</p>	<p>True negative (TN) „Vorhersage richtig, Objekt nicht vorhanden“</p> <p>$t_n = 50$</p>

TPR =

FPR =

FNR =

TNR =

Untersuchte Bilder	n_{ges}	145
Tatsächlich Husky	n_{Husky}	90
Tatsächlich kein Husky	$n_{noHusky}$	55
Alg behauptet Husky	n_{AHusky}	85
Alg behauptet kein Husky	$n_{AnoHusky}$	60

Berechnen Sie:

- TPR
- FPR
- FNR
- TNR

- Supervised Learning
 - K-NN
 - NN
 - Confusion Matrix

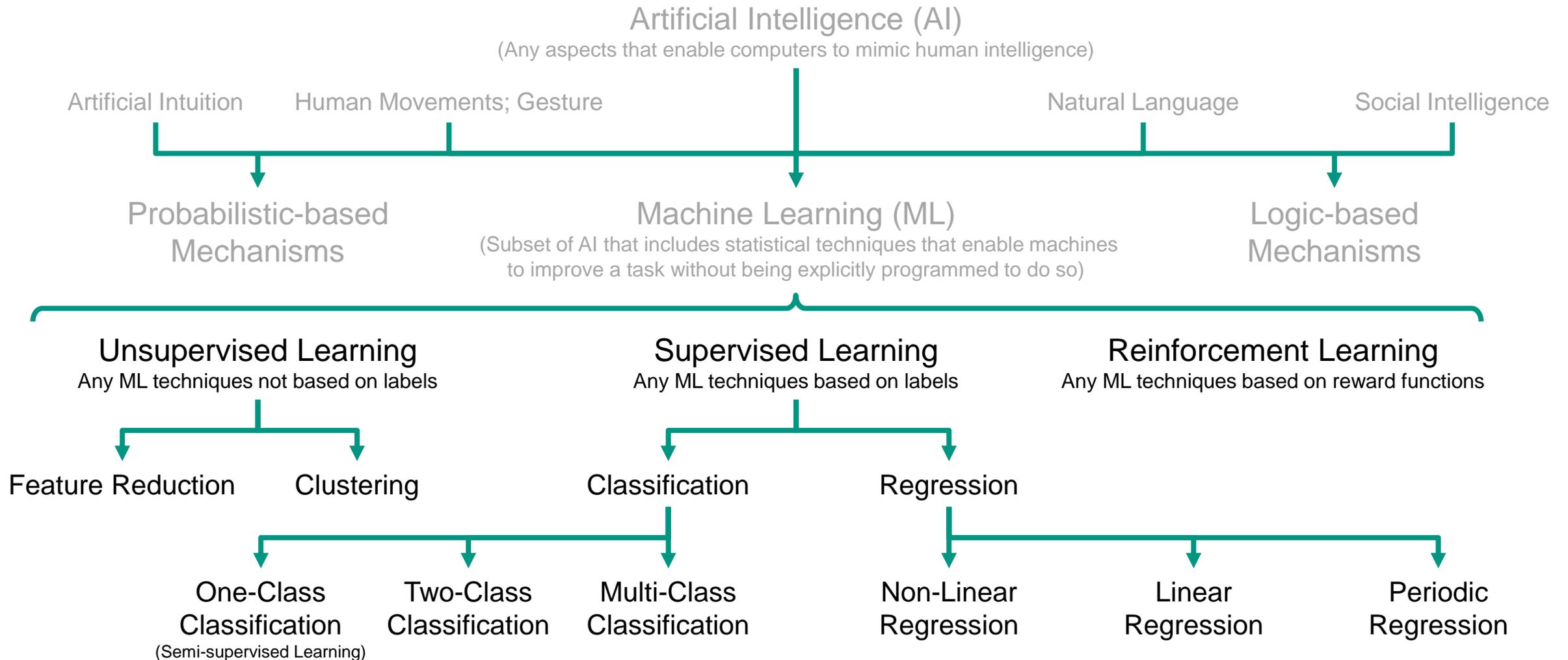


Fragen?

UNSUPERVISED LEARNING CLUSTERING

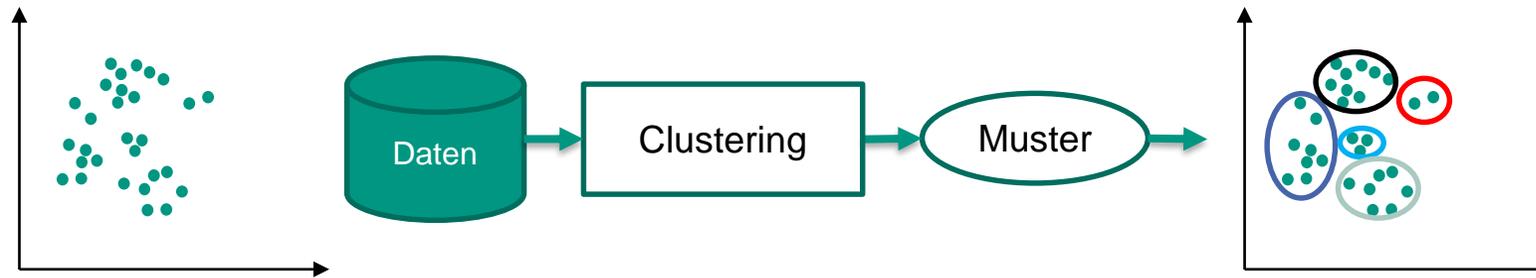


Modelbildung – Übersicht über Verfahren



Unsupervised Learning

Clustering



Gruppierung von gleichartigen Situationen/ Datenpunkten/ Merkmalen

■ Konzepte

- Berechnung Ähnlichkeiten von durch Daten beschriebene Objekte
- Ähnlichkeit für Vorhersagen verwenden
- Clustering als auf Ähnlichkeit beruhende Segmentierung

■ Verfahren

- Suche nach ähnlichen Objekten
- K-means
- Distanzmaß für Berechnung von Ähnlichkeit

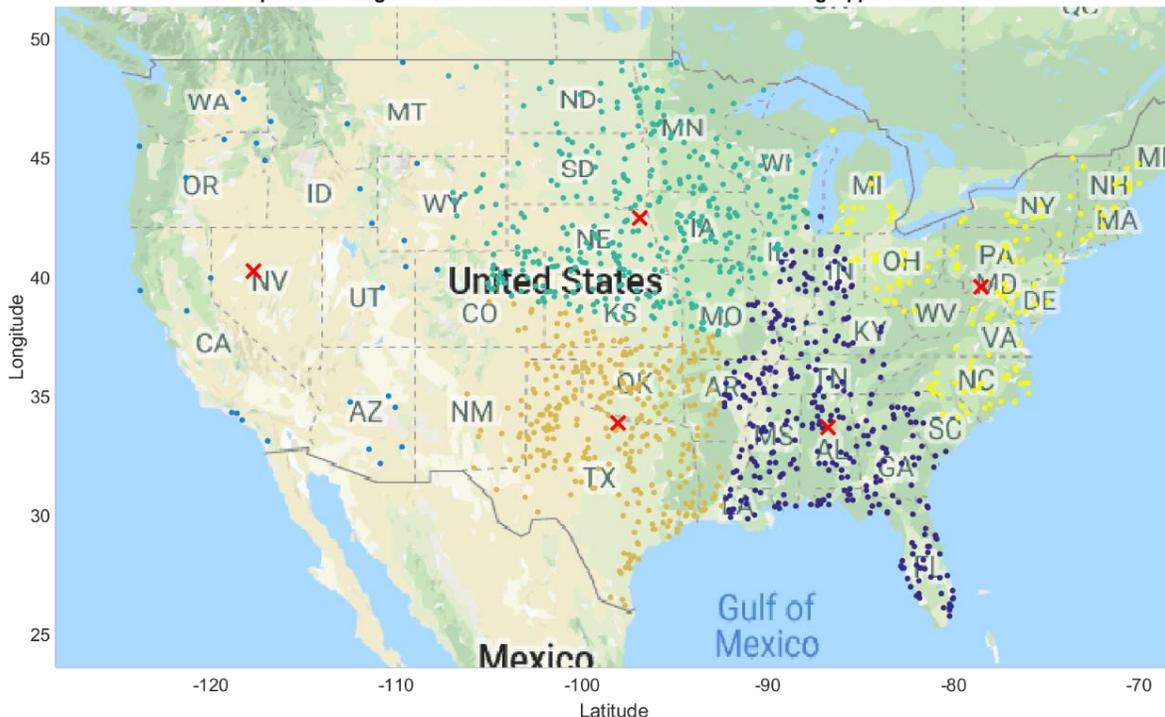
Clustering - Algorithmen

K-means

- Verfahren der Vektorquantisierung
- Bildung Mengen von ähnlichen Objekten in k Gruppen
möglichst geringe Varianz, Gruppen ähnlicher Größen

Datensatz in k Partitionen teilen, so dass Summe der quadrierten Abweichungen von Cluster-Schwerpunkten minimal wird

Exemplarisch einige Stürme in den USA zwischen 1950 und 2015 gruppiert mit k-means



$$J = \sum_{i=1}^k \sum_{x_j \in S_i} \|x_j - \mu_i\|^2$$

x_j : Datenpunkt
 S_i : Cluster
 μ_i : Schwerpunkt

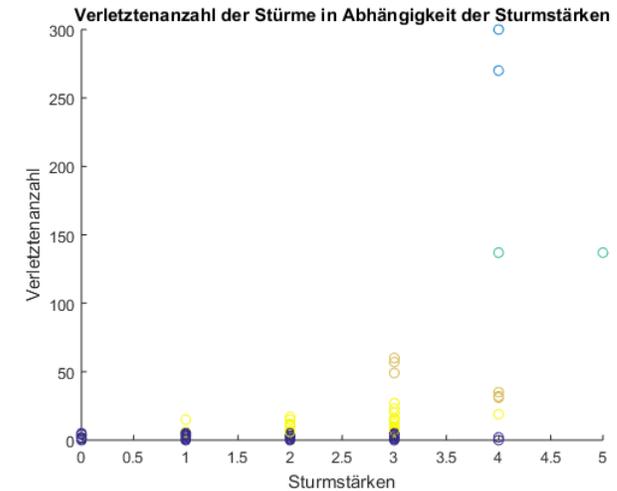
Lloyd Algorithmus:

1. k unterschiedliche Zentren c_1, c_2, \dots, c_k
2. Solange sich die Zielfunktion verbessert:
Partitioniere P in Cluster S_1, S_2, \dots, S_k dass S_i die Punkte aus P enthält, deren nächstgelegenes Zentrum c_i ist
Für jedes $1 \leq i \leq k$ sei $c_i \leftarrow \mu(C_i)$

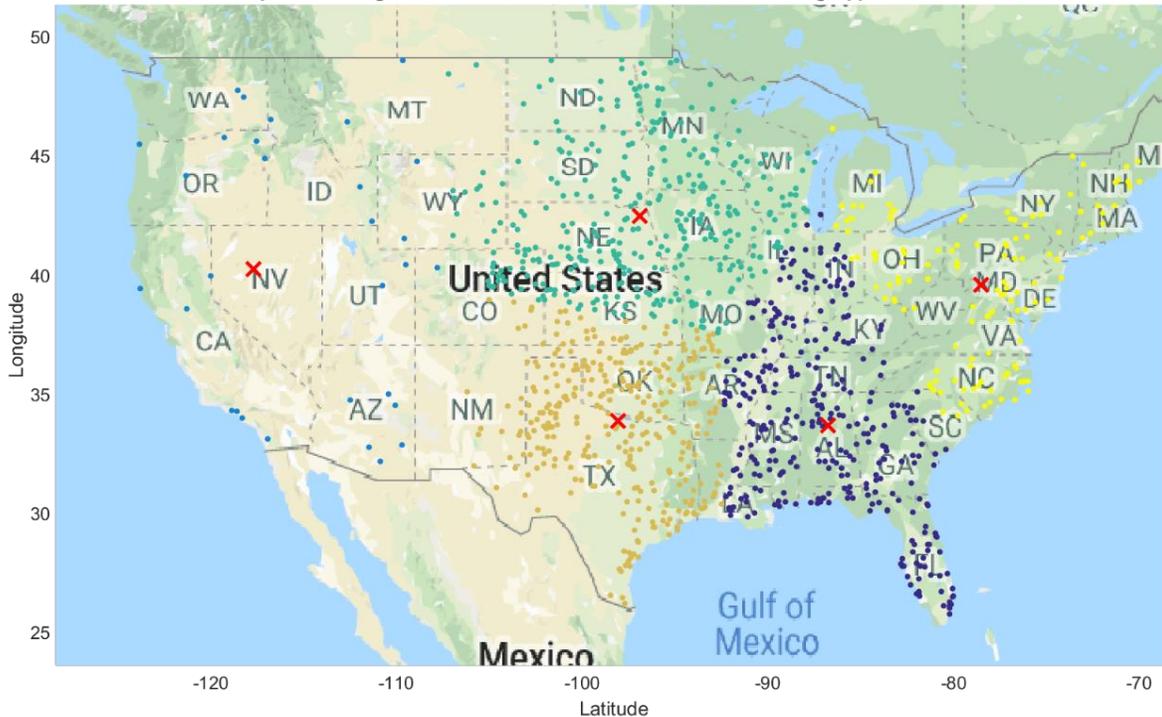
Clustering

K-means

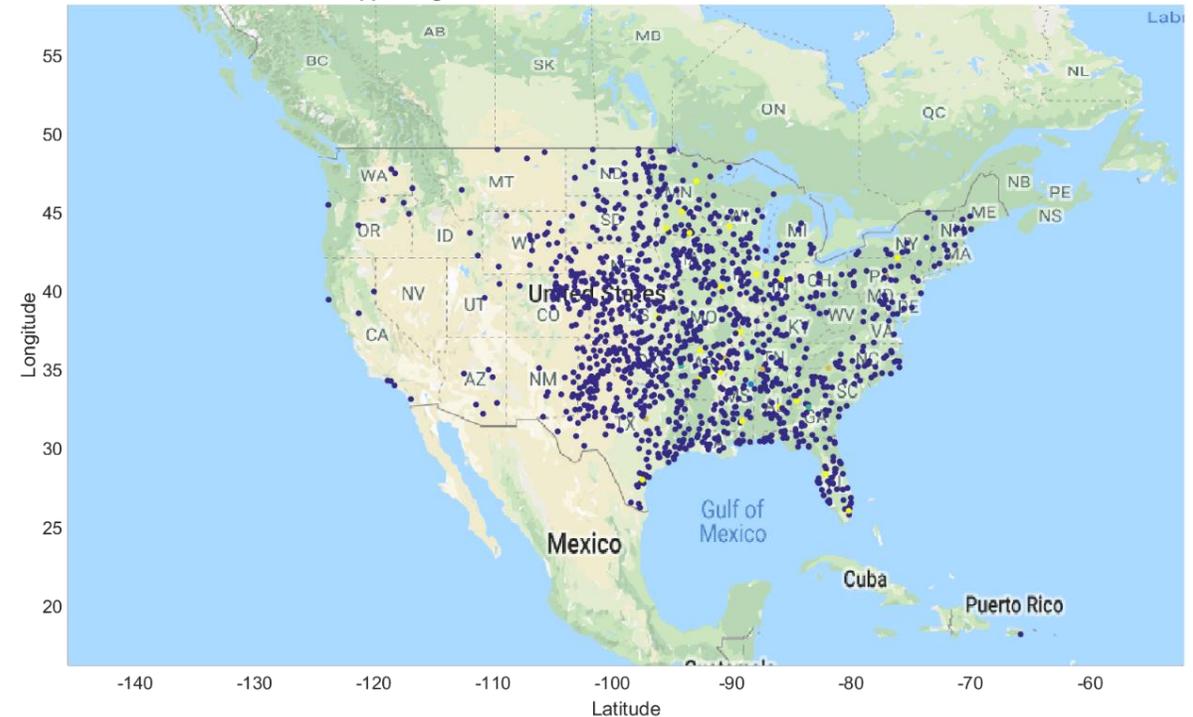
- Links:
Clusterbildung nach GPS-Daten
- Rechts:
Clusterbildung nach Sturmstärken und Verletztenanzahl



Exemplarisch einige Stürme in den USA zwischen 1950 und 2015 gruppiert mit k-means



Exemplarisch einige Stürme in den USA zwischen 1950 und 2015
Gruppierung nach Sturmstärke und Verletztenanzahl mit k-means



Clustering

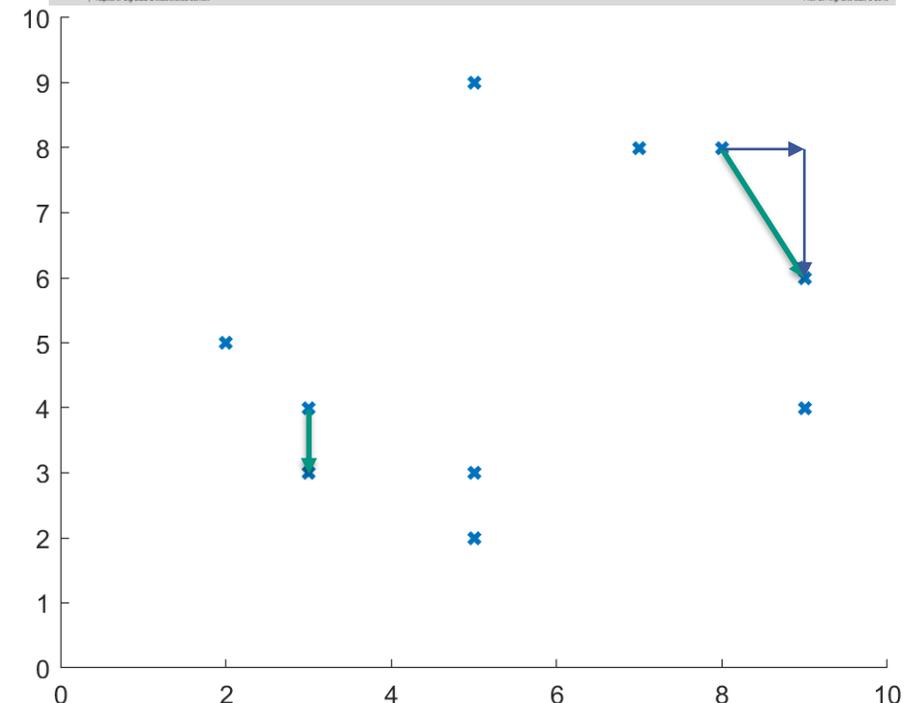
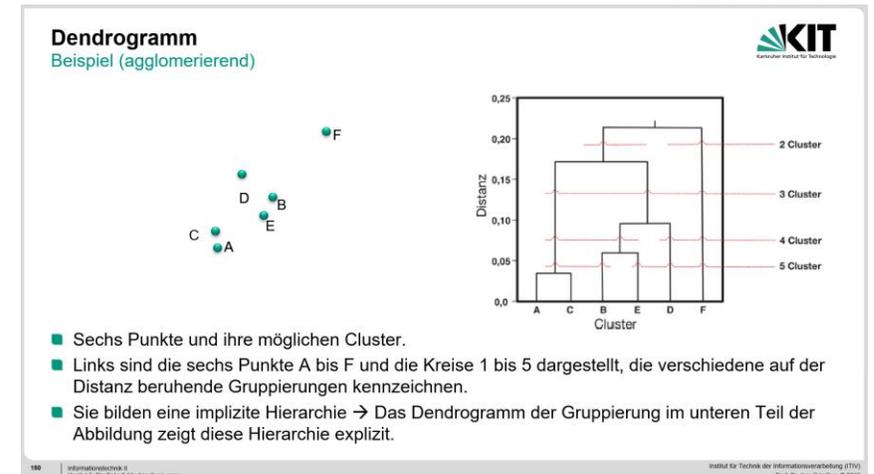
Dendrogramme

- Visualisierung hierarchischer Baumstruktur
 - Jede Wurzel repräsentiert ein einzelnes Cluster, welches die komplette Datenmenge umfasst
 - Jedes Blatt repräsentiert ein Cluster, in denen sich lediglich ein Objekt befindet
 - Jeder Knoten repräsentiert die Vereinigung aller Kindknoten/Blätter
 - Die Verbindungen stellen Ähnlichkeitsmaß dar

Grafische Erstellung über Entfernung
Gesucht Abstand zwischen Datenpunkten

$$|\overline{x_a}| = \begin{vmatrix} x_{a1} \\ x_{a2} \end{vmatrix} = \sqrt{x_{a1}^2 + x_{a2}^2} \downarrow$$

Vektor zwischen zwei Punkten:
Linearkombination $\overline{x_a} = \overline{x_1} + \overline{x_2}$



Clustering

Dendrogramm

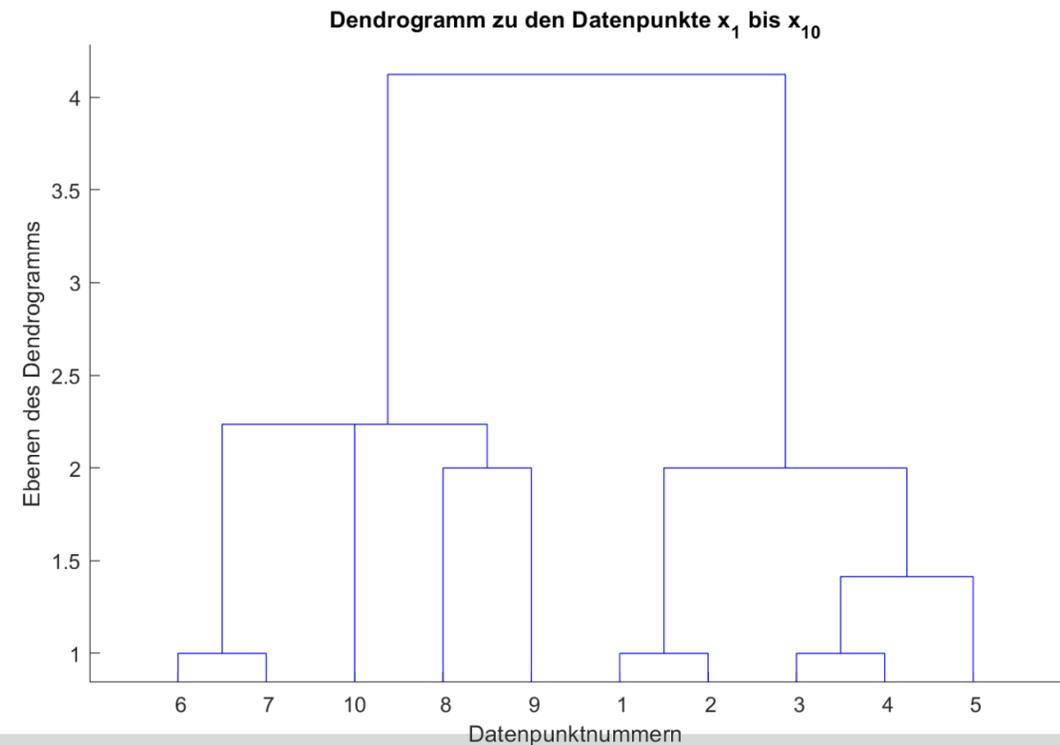
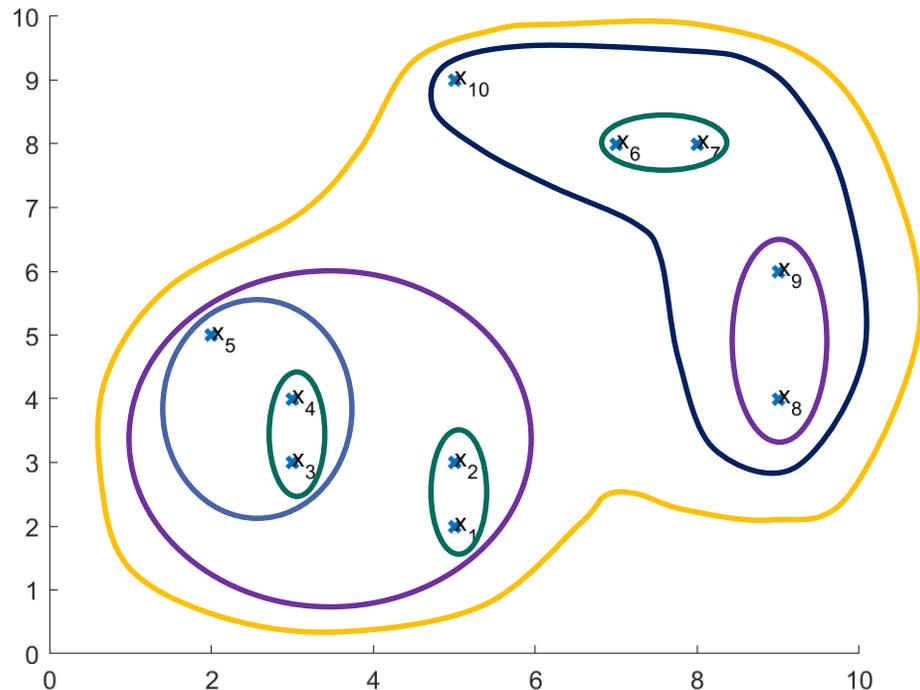
Ebene

Abstand

Punkte

Vektor zwischen zwei Punkten: $\overline{x_a} = \overline{x_1} + \overline{x_2}$

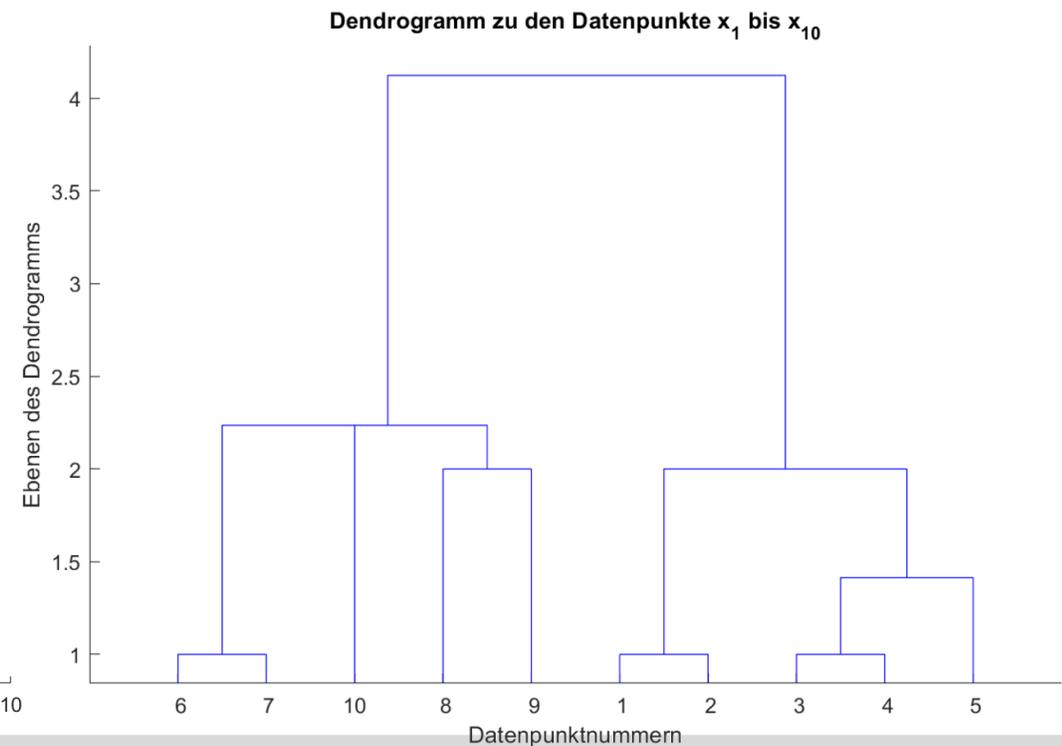
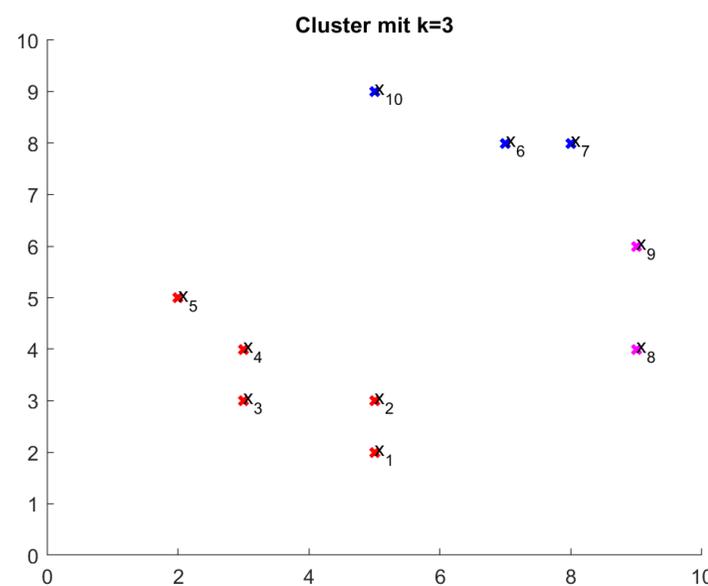
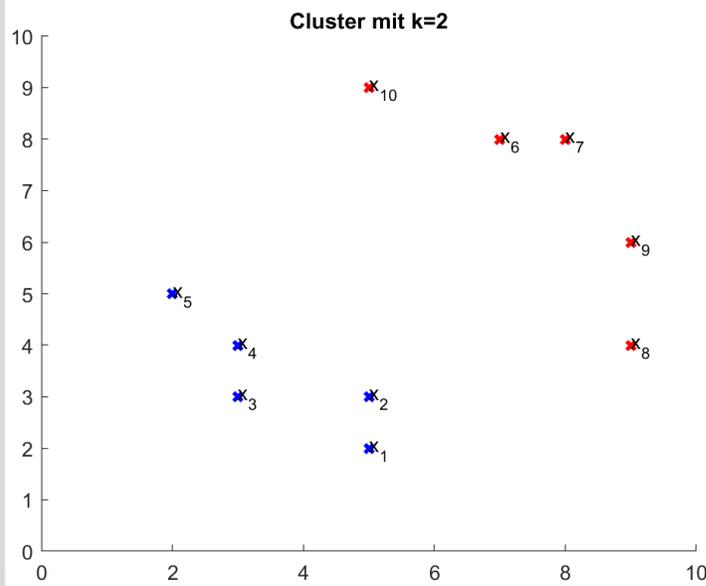
$$\text{Abstand: } |\overline{x_a}| = \begin{vmatrix} x_{a1} \\ x_{a2} \end{vmatrix} = \sqrt{x_{a1}^2 + x_{a2}^2}$$



Clustering

K-means mit unterschiedlichen k

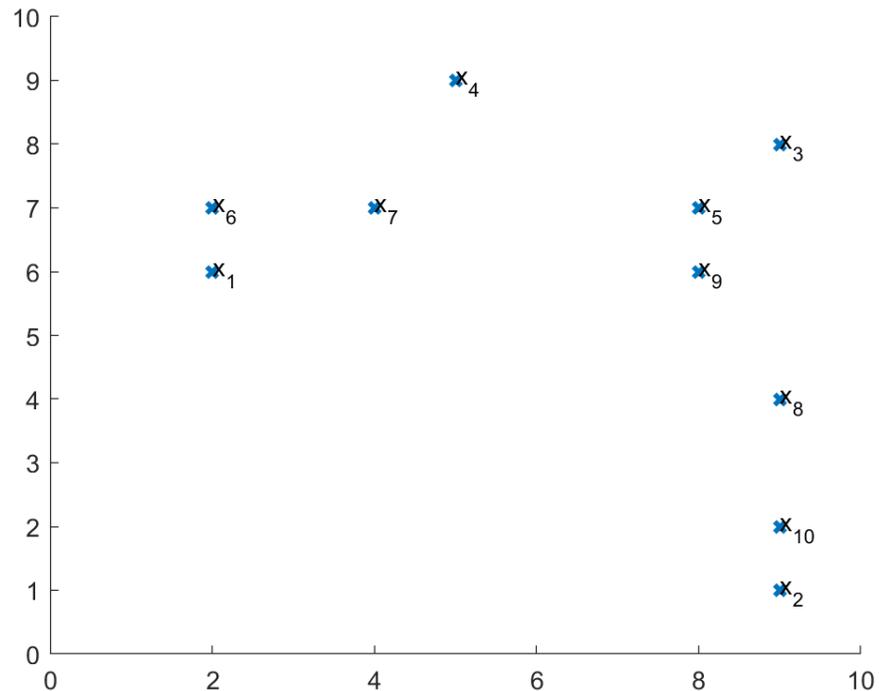
- Dendrogramm gibt Aufschluss über die zu wählende Anzahl der Cluster



K-means und Dendrogramm

Zwischenübung

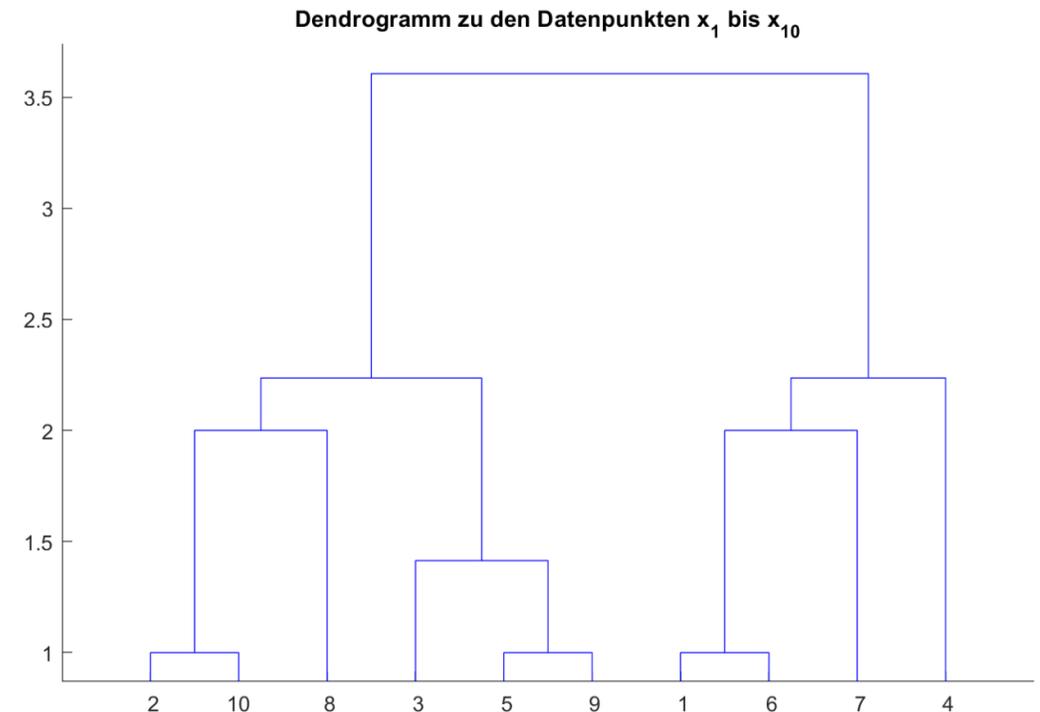
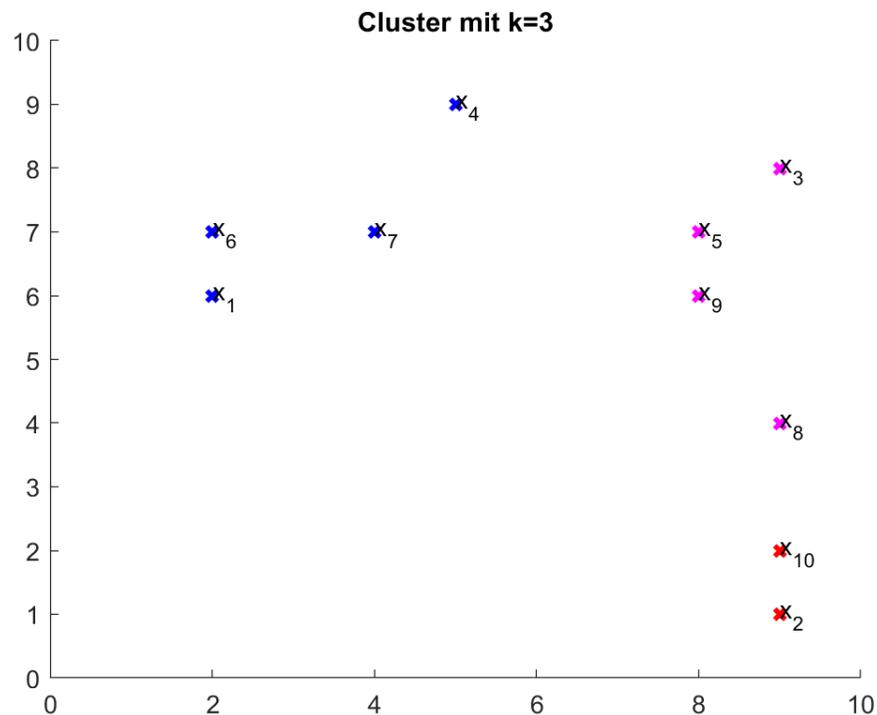
- Zeichnen Sie das zugehörige Dendrogramm zu den gegebenen Datenpunkten und geben Sie auf Basis dessen ein geeignetes k für einen k-means Algorithmus an



K-means und Dendrogramm

Zwischenübung - Lsg

- Zeichnen Sie das zugehörige Dendrogramm zu den gegebenen Datenpunkten und geben Sie auf Basis dessen ein geeignetes k für einen k-means Algorithmus an



- Unsupervised Learning
 - Clustering
 - K-means
 - Dendrogramme



Ziele der heutigen Übung



■ Nach der heutigen Übung können Sie....

• ...Ansätze zur Verwaltung und Analyse großer Datenbestände hinsichtlich ihrer Anwendbarkeit und Wirksamkeit einschätzen

1

• ... Modelle berechnen unter Anwendung überwachter maschineller Lernmethoden

2

• ... Merkmale und Eigenschaften von KNN benennen, bewerten und abgrenzen

3

• ... überwachte Lernverfahren hinsichtlich ihrer Performanz bewerten

4

• ... Modelle unter Anwendung unüberwachter maschineller Lernverfahren aufstellen