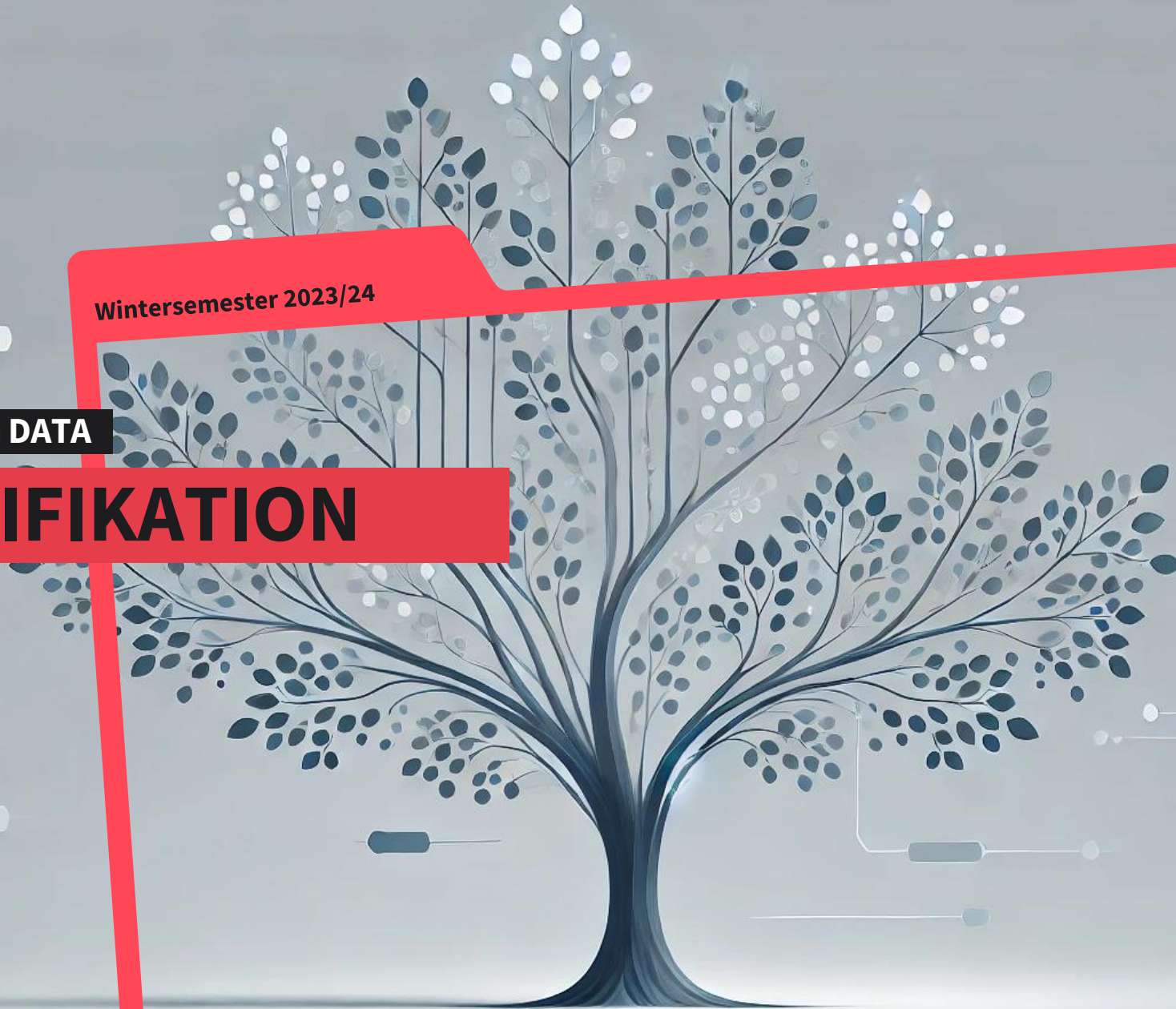


Wintersemester 2023/24

DATA ANALYTICS & BIG DATA

03: KLASSIFIKATION

Prof. Dr. Frank Krickel



Semester- plan

ID	Was?	Wann?
01 ASSOZIATION	Assoziationsanalyse, per Hand, in Java, in KNIME	17.10.
02 STATISTIK	Statistik-Grundlagen, Lineare Regression	24.10.
03 KLASSIFIKATION	Regression (Forts.), Klassifikationsalgorithmen	7.11.
04 CLUSTERING	Klassifikation (Forts.), Cluster-Analyse	15.11.
05 KI	Neuronale Netze, Deep Learning	28.11.
06 TEXT	Text- und Bild-Analyse, Themen für Fallstudie	5.12.
07 WEB	Web-Analyse	9.1.
08 UEBUNG	Machine Learning anwenden	10.1.
09 BIG DATA	Big-Data-Architekturen: Datenhaltungskonzepte	16.1.
10 BIG DATA 2	Big-Data-Architekturen: Fortsetzung	23.1.
11 PUFFER	Puffer/Arbeit an Fallstudie	30.1.
12 FALLSTUDIE	Präsentation der Fallstudie	6.2.

WIEDERHOLUNG/ AUFGABEN LINEARE REGRESSION

04 CLUSTERING

1. **Wiederholung/Aufgaben**
Lineare Regression
2. Wiederholung/Aufgaben
Logistische Regression
3. Klassifikation: KNN-Algorithmus
4. Klassifikation: Entscheidungsbäume
5. Clustering: k-means-Algorithmus

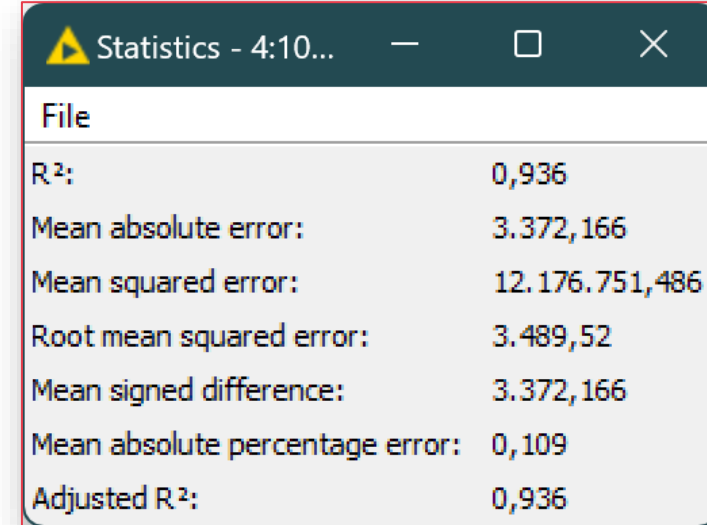
MACHINE LEARNING – „ARTEN“ – **STAND HEUTE**

Oberbegriff	Überwachtes Lernen (supervised learning)		Unüberwachtes Lernen (unsupervised learning)			Reinforcement Learning ...
Ziel:	✓ „Regression“: Vorhersage eines metrischen Wertes	Klassifikation: Vorhersage eines kategorialen Wertes	Clustering: Bildung von Klassen gleicher Objekte	✓ Assoziation: Suchen von wenn- dann-Beziehungen	<i>Anomaly Detection</i>	
Anwendbare Verfahren:	✓ Einfache Lineare Regression	✓ Logistische Regression	K-means-Algorithmus	✓ Apriori Algorithmus		
	✓ Multilineare Regression	kNN-Algorithmus	<i>Hierarchisches Clustering</i>	...		
	✓ Polynomiale Regression	Entscheidungsbaum (Decision Tree)	DBSCAN (Density-Based Spatial ...)			
	Logistische Regression ...	<i>Support Vector Maschine</i> <i>Random Forest</i> Naive Bayes Neural Networks 			

Legende:
 Wird in der Vorlesung behandelt
 Wird NICHT in der Vorlesung behandelt
 ✓ Bereits behandelt
 Thema heute

GÜTEMAßE (ODER: FEHLERMAß, LOSS FUNCTIONS))

Gütemaße	Erklärung
R-squared (R^2)	Auch bekannt als Bestimmtheitsmaß. Zwischen 0 und 1. Je näher an 1, um so besser. Vorsicht: kann bei mehreren Prädiktoren steigen, auch wenn das Modell nicht besser wird
Mean Absolute Error (MAE)	Mittelwert der absoluten Abstände (selbe Skala wie der Zielwert)
Mean Squared Error (MSE)	Mittelwert der Quadrate der Abstände (sehr gebräuchlich)
Root Mean Squared Error (RMSE)	Wurzel aus MSE (selbe Skala wie der Zielwert, gewichtet hohe Abweichungen stärker)
Mean signed difference (MSD)	Zeigt plus/minus der Abweichung an
Mean absolute percentage error (MAPE)	Mittelwert der prozentualen Abweichungen
Adjusted R2	Häufig besser geeignet und stets niedrigerer als R^2 . Wird mittels MSE und Varianz ermittelt

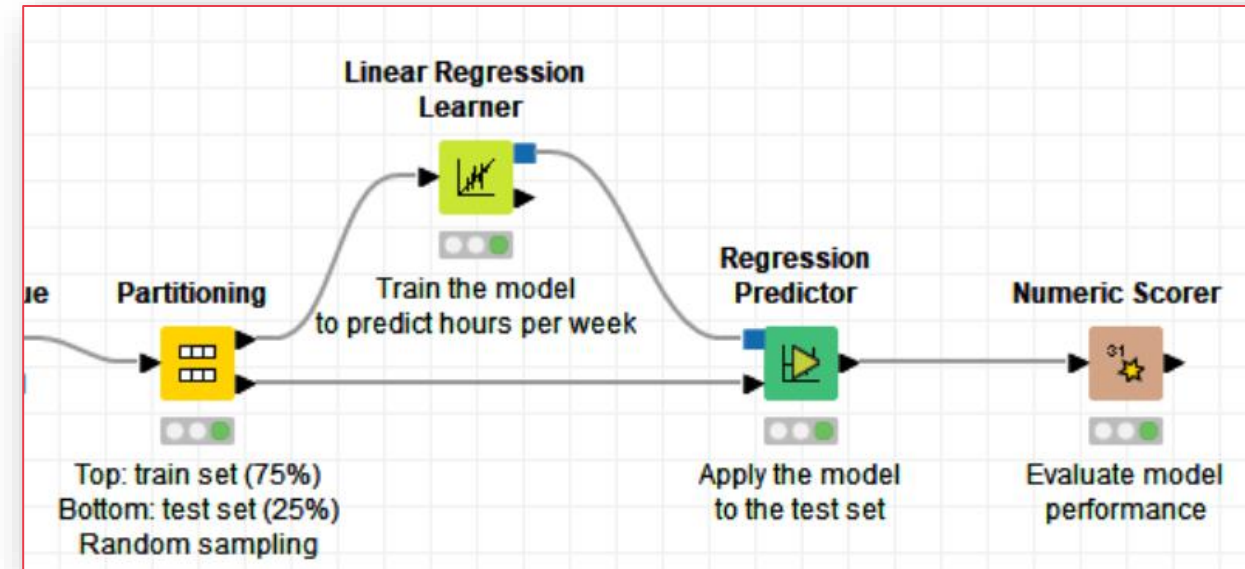
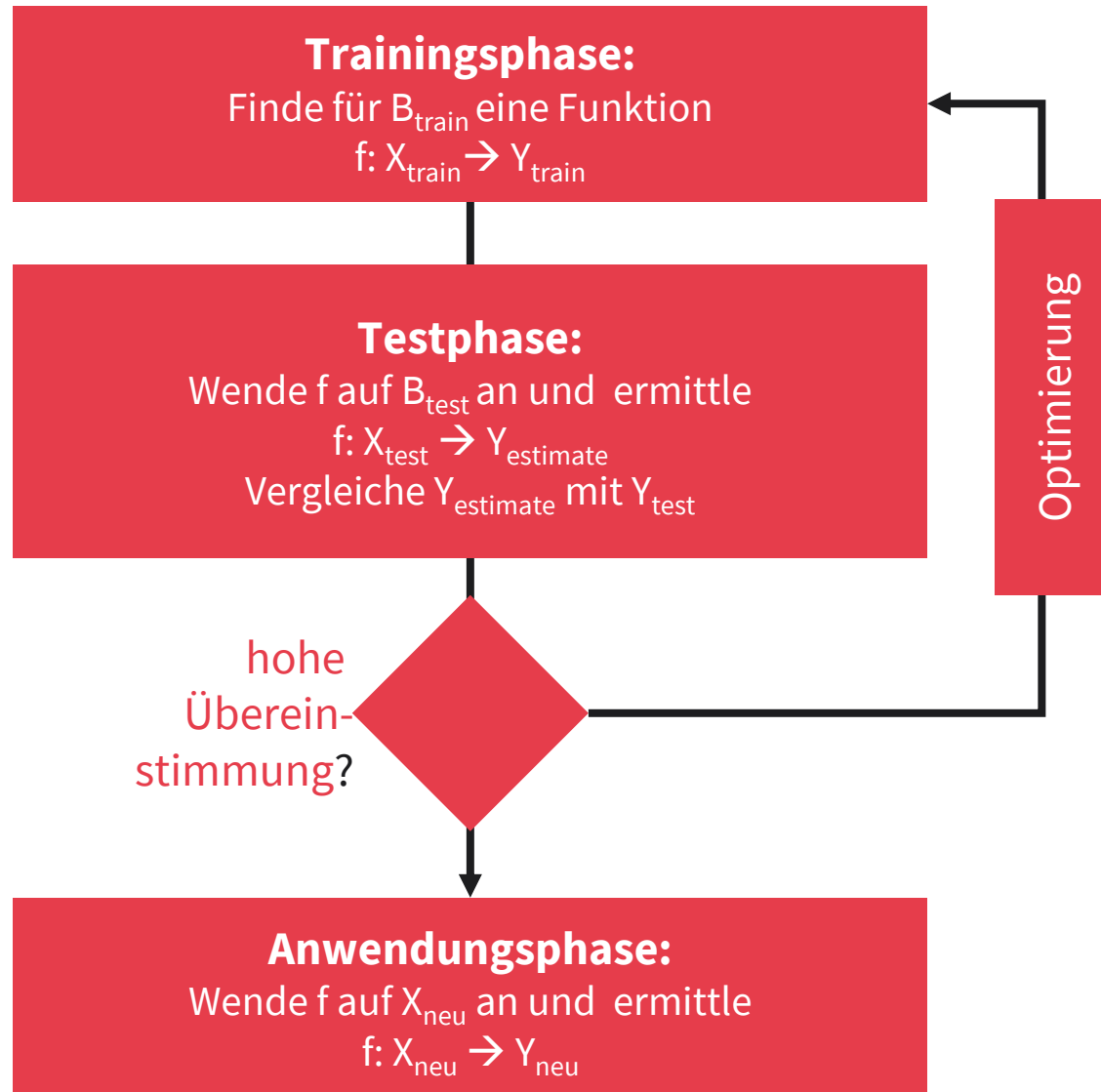


File	
R^2 :	0,936
Mean absolute error:	3.372,166
Mean squared error:	12.176.751,486
Root mean squared error:	3.489,52
Mean signed difference:	3.372,166
Mean absolute percentage error:	0,109
Adjusted R^2 :	0,936

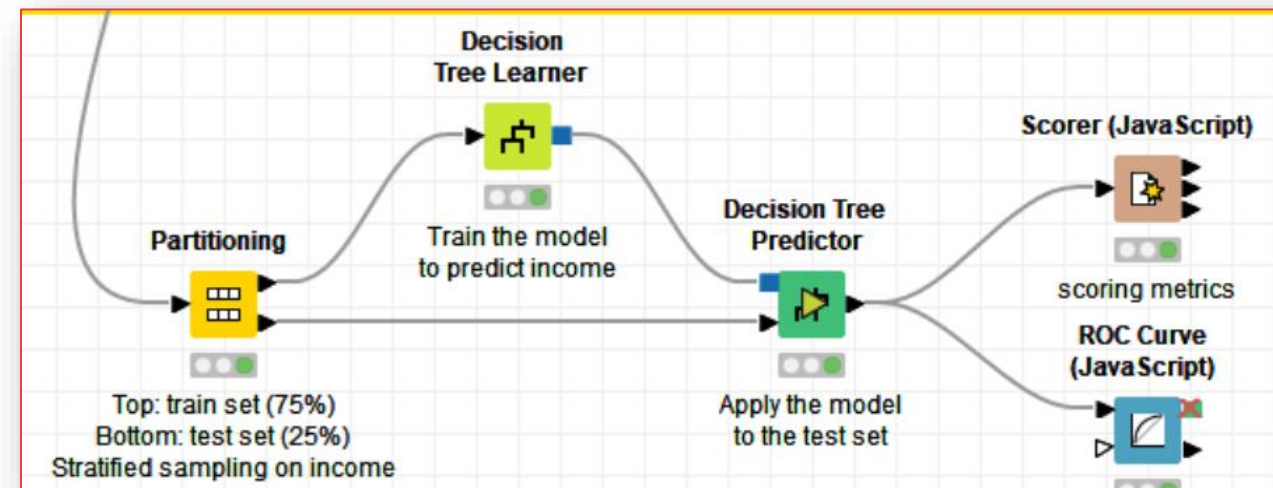
Empfehlung (nicht immer anwendbar):

1. Schaut Euch **zuerst R^2 bzw. Adj. R^2** an: Wenn die Werte zwischen 0,8 und 1 liegen, ist die Schätzung recht gut.
2. Betrachtet **auch den RMSE und den MAE (MAPE)**, um die Auswirkung sehr großer Fehler einzuschätzen. Kleinere Werte sind immer besser.

PROGNOSEVERFAHREN UND ÜBERWACHTES LERNEN IN KNIME



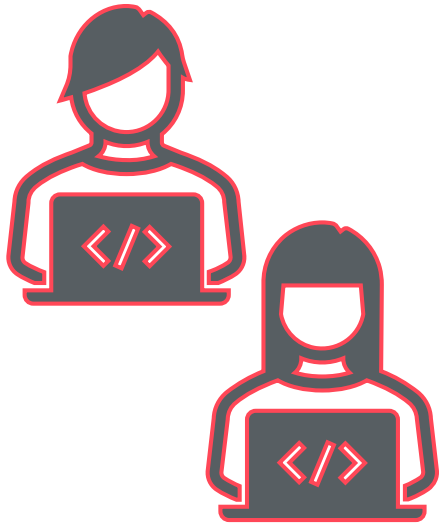
— B wird getrennt in B_{train} und B_{test}



ÜBUNG

OVERFITTING / UNDERFITTING

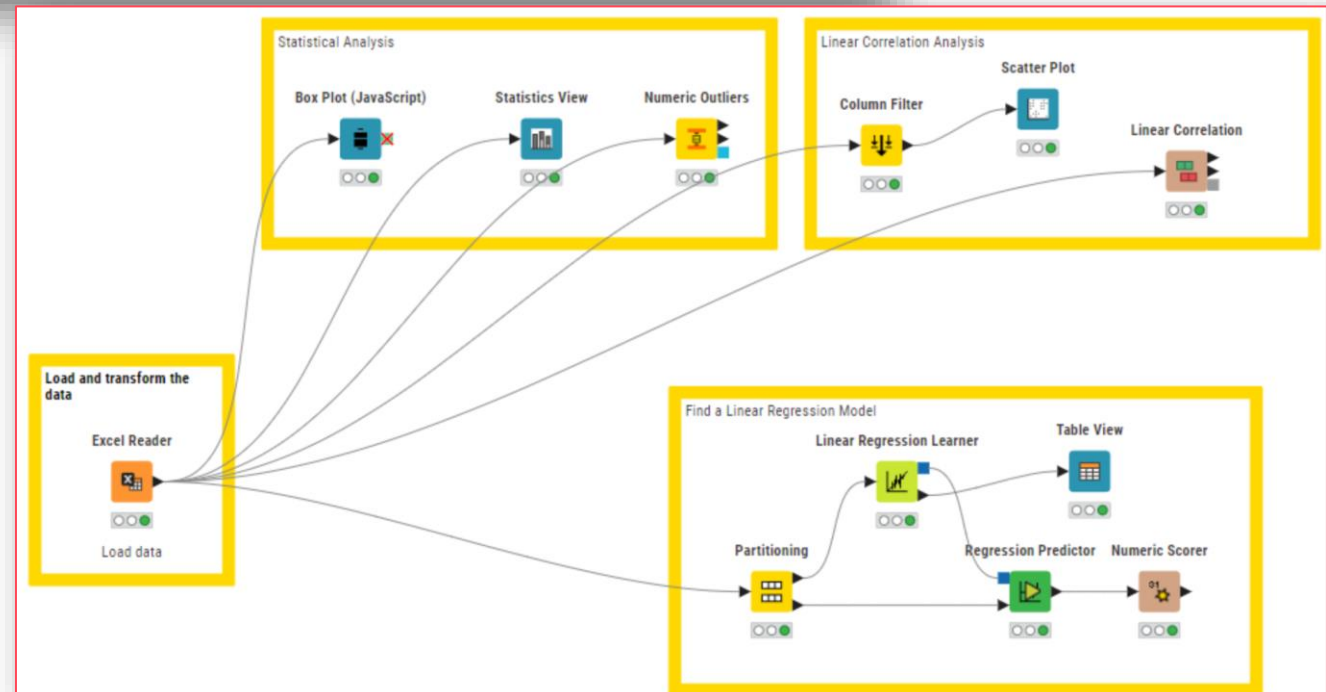
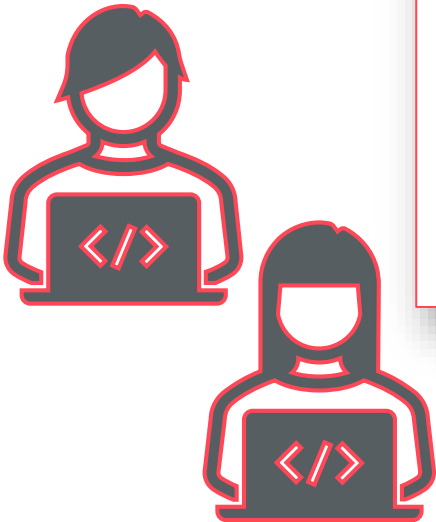
– Sucht im Internet nach einer einfachen, bildhaften Erklärung



	Underfitting	Just right	Overfitting
Symptoms	<ul style="list-style-type: none"> - High training error - Training error close to test error - High bias 	<ul style="list-style-type: none"> - Training error slightly lower than test error 	<ul style="list-style-type: none"> - Low training error - Training error much lower than test error - High variance
Regression			
Classification			
Deep learning			
Remedies	<ul style="list-style-type: none"> - Complexify model - Add more features - Train longer 	https://aiml.com/what-is-underfitting/	<ul style="list-style-type: none"> - Regularize - Get more data

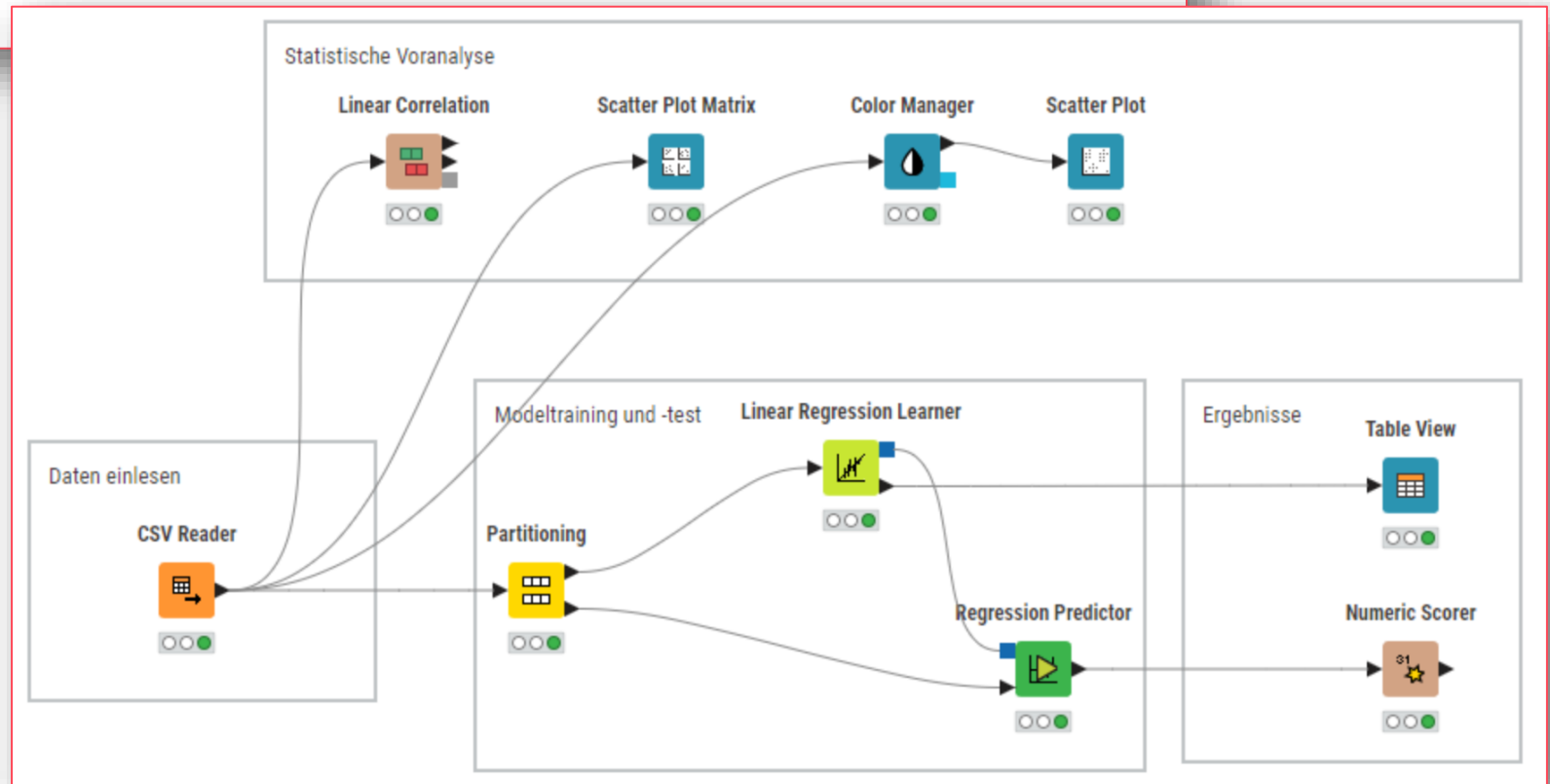
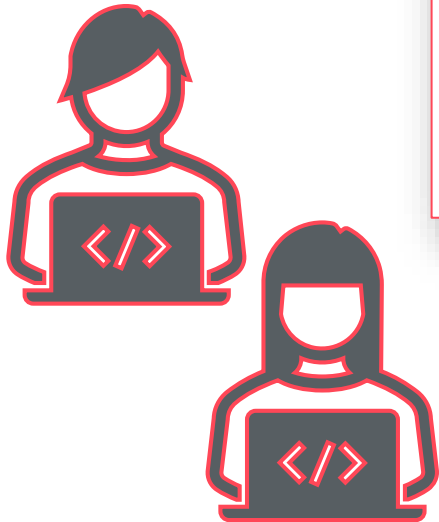
Aufgabe 1: Lineare Regression (Wdh. aus Vorlesung)

- Lade die Datei *Einkommen.xlsx* in KNIME
- Erstelle ein Linear Regression-Modell, in dem du das Einkommen aus folgenden Parametern prognostizierst:
 - IQ
 - Alter
 - Höchster Abschluss
 - Größe
 - Gender
- Versuche Kombinationen aus diesen Prädiktoren
- Welche Kombination hat den höchsten r-squared bzw. adjusted r-squared Wert?



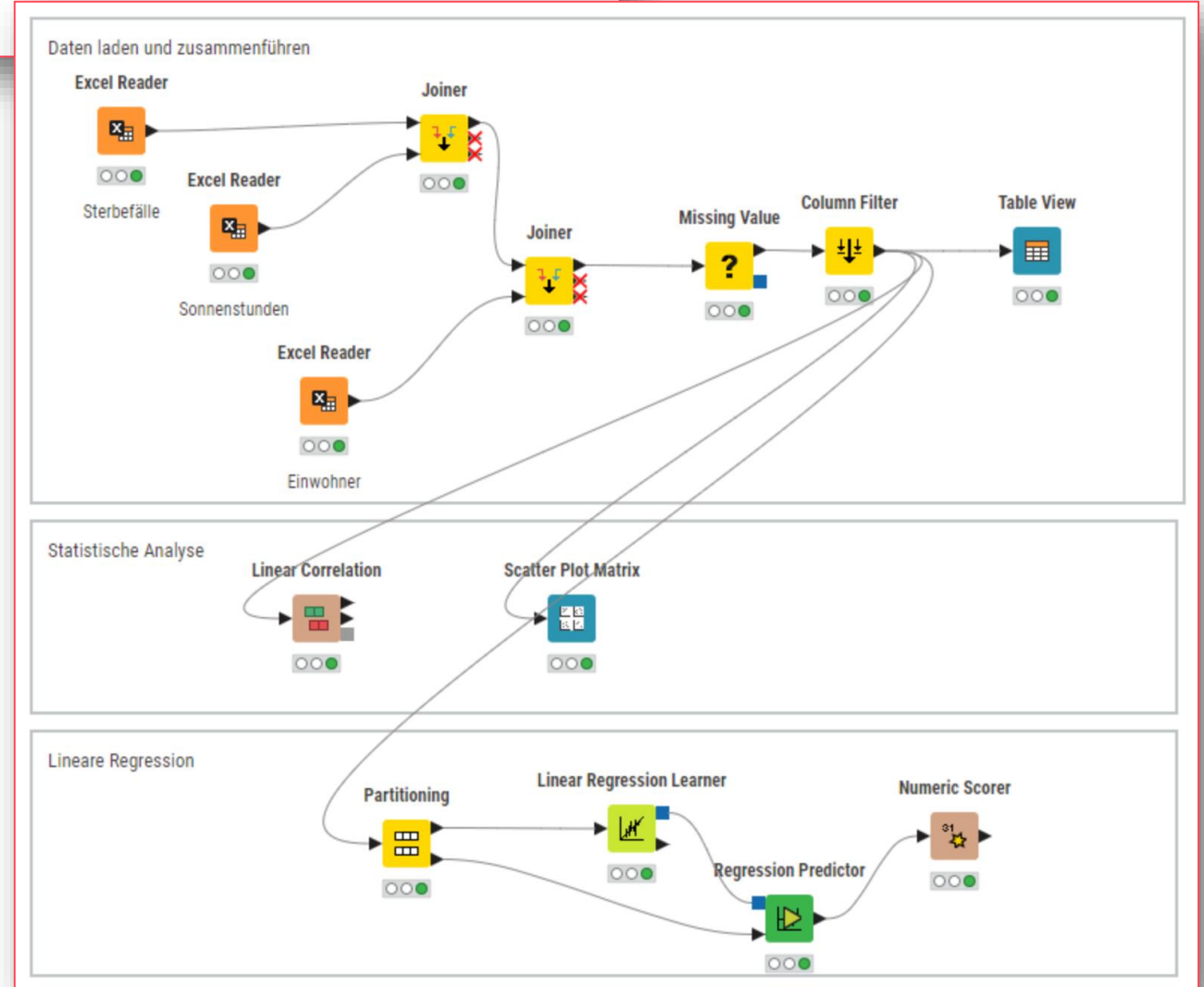
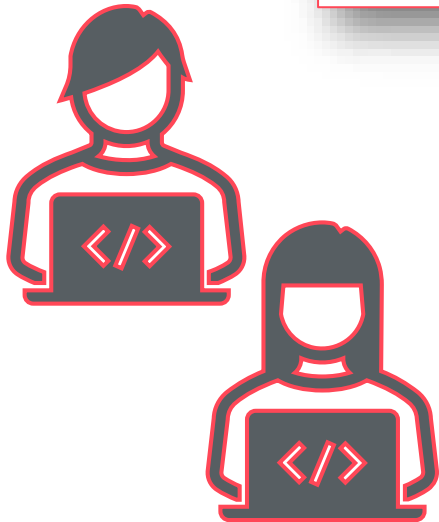
Aufgabe 2: Lineare Regression (Transfer)

- Lade die Datei *FuelConsumptionCo2.csv* in KNIME
- Erstelle ein Linear Regression-Modell, in dem du die CO2-Emissionen aus folgenden Parametern prognostizierst:
 - Fuel Consumption
- Versuche weitere Prädiktoren
- Welcher Prädiktor hat den höchsten r-squared bzw. adjusted r-squared Wert?
-



Aufgabe 3: Lineare Regression (Vertiefung)

- Finde Daten zu den deutschen Bundesländern. Z.B. Fläche, Einwohnerzahl, BIP, uvm. Versuche herauszufinden, ob es „gute“ lineare Korrelationen zwischen diesen Merkmalen gibt.



WIEDERHOLUNG/ AUFGABEN LOGISTISCHE REGRESSION

04 CLUSTERING

1. Wiederholung/Aufgaben
Lineare Regression
2. **Wiederholung/Aufgaben**
Logistische Regression
3. Klassifikation: KNN-Algorithmus
4. Klassifikation: Entscheidungsbäume
5. Clustering: k-means-Algorithmus

Gemeinsamkeiten

- sind Prognoseverfahren
- d.h. sie versuchen aus einer Menge von Beobachtungen
- einen "Weg" zu finden, eine Vorhersage über Nicht-Beobachtetes zu machen
- sind Verfahren des "überwachten" Lernens (supervised learning):
 - Trainieren auf Basis von bekannten Beobachtungen
 - Testen auf Basis von bekannten Beobachtungen
 - ggf. Optimieren
 - Anwenden auf neue, nicht beobachtete Einheiten

Unterschiede

- Regression versucht einen mathematischen Zusammenhang zu finden, um metrische Werte für eine Vorhersage zu finden
 - z.B. Einkommen in Abhängigkeit von IQ, Ausbildung, ...
- Klassifikation versucht hingegen, für kategoriale Werte eine Aussage zu finden, ob die Beobachtungseinheit zu einer bestimmten Klasse gehört
 - z.B. Einkommenskategorien {hoch, mittel, niedrig} finden,
 - z.B. Katze oder Hund
 - Z.B. Kreditwürdigkeit (ja, nein)

- Ist eigentlich ein Regressionsverfahren, das Prädiktoren einen metrischen Wert zwischen 0 und 1 zuweist.
- Dieser lässt sich als Wahrscheinlichkeitswert interpretieren
- De facto wird die logistische Regression aber vor allem für Klassifikationsaufgaben eingesetzt mit bi-nominalen Werten:
 - Ja/Nein
 - 0/1
 - Katze/Hund
 - Kreditwürdig/nicht kreditwürdig
 - Besteht Prüfung / Besteht nicht

ANWENDUNG DER LOGISTISCHEN FUNKTION

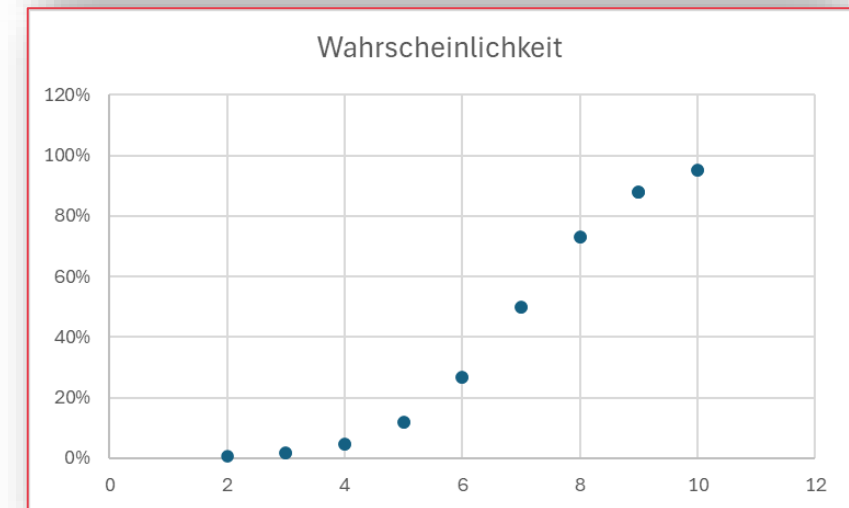
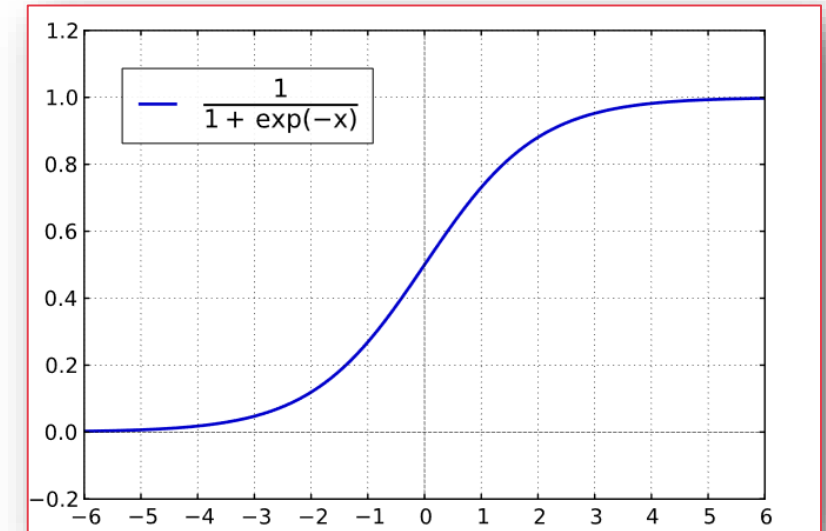
- Eine Sigmoidfunktion (genauer gesagt: logistische Funktion) ordnet beliebigen negativen oder positiven Werten einen Wert im Intervall $[0 \dots 1]$ zu
- Die Formel der Funktion lautet:

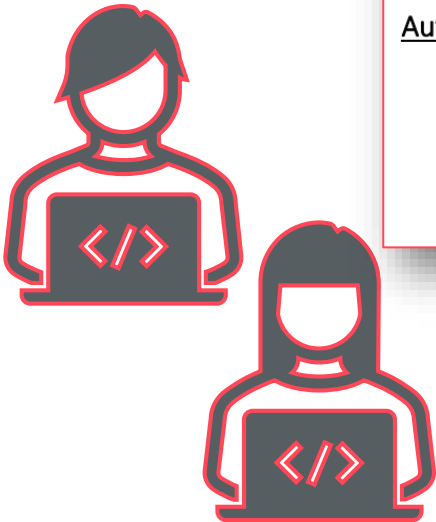
$$F(x) = \frac{e^x}{1 + e^x} = \frac{1}{1 + e^{-x}}, \quad x \in \mathbb{R}$$

- Wenn wir die logistische Regression nutzen, suchen wir also die Koeffizienten a und b einer Funktion der Form:

$$P(\text{Bestehen}) = \frac{1}{1 + e^{-(a+b \times \text{Tage})}}$$

- Setzen wir z.B. $a = -5$ und $b = 1$, erhalten wir eine erste Annäherung
- Diese muss nun optimiert werden ... das überlassen wir dem Algorithmus 😊





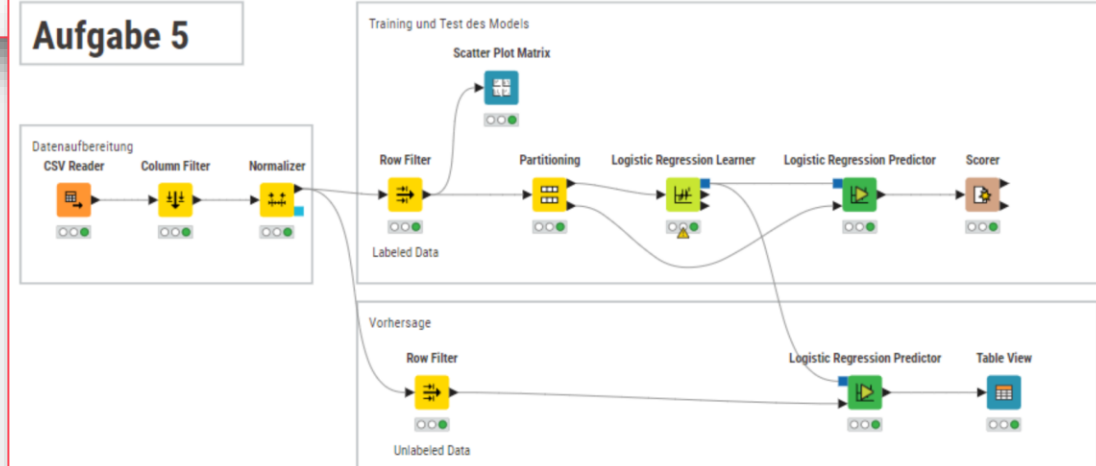
Aufgabe 5: Logistische Regression

- Lade die Datei *Einkommen2.csv* in KNIME
- Denke an die Normalisierung
- Erstelle ein Logistic Regression-Modell, in dem du die Einkommenshöhe (niedrig, mittel, hoch) prognostizierst:

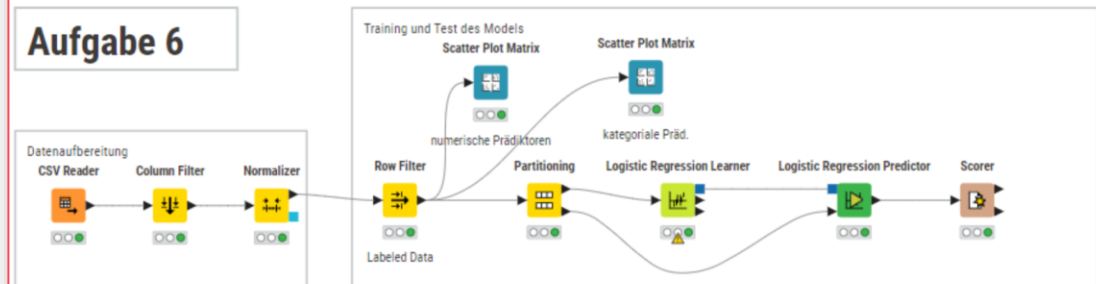
Aufgabe 6: Logistische Regression

- Lade die Datei *Einkommen3.csv* in KNIME
- Denke an die Normalisierung
- Erstelle ein Logistic Regression-Modell, in dem du das Income prognostizierst:
-

Aufgabe 5



Aufgabe 6



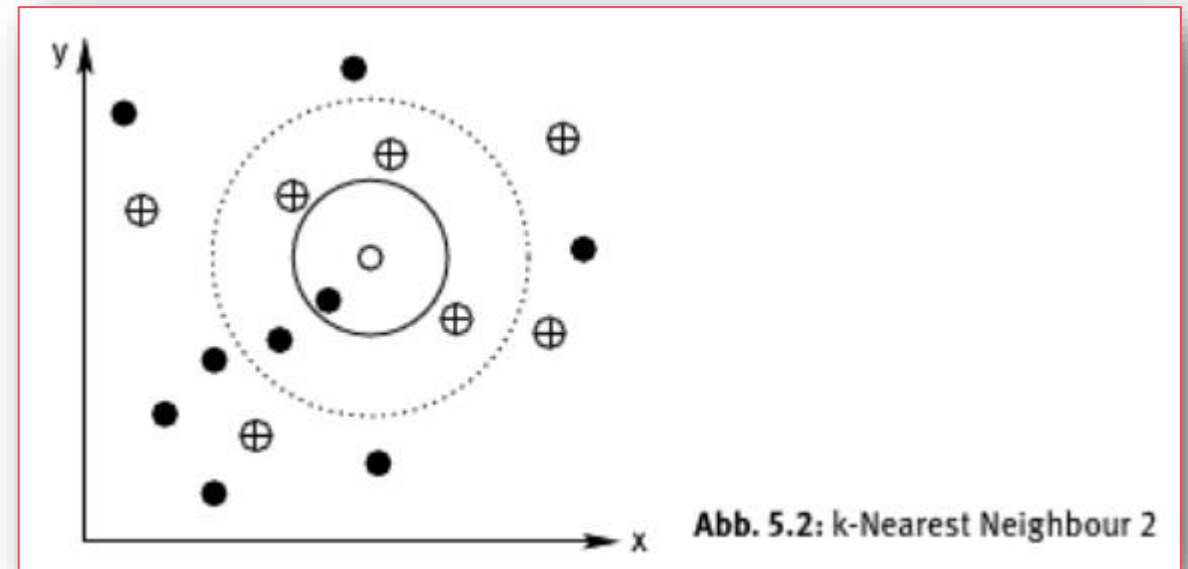
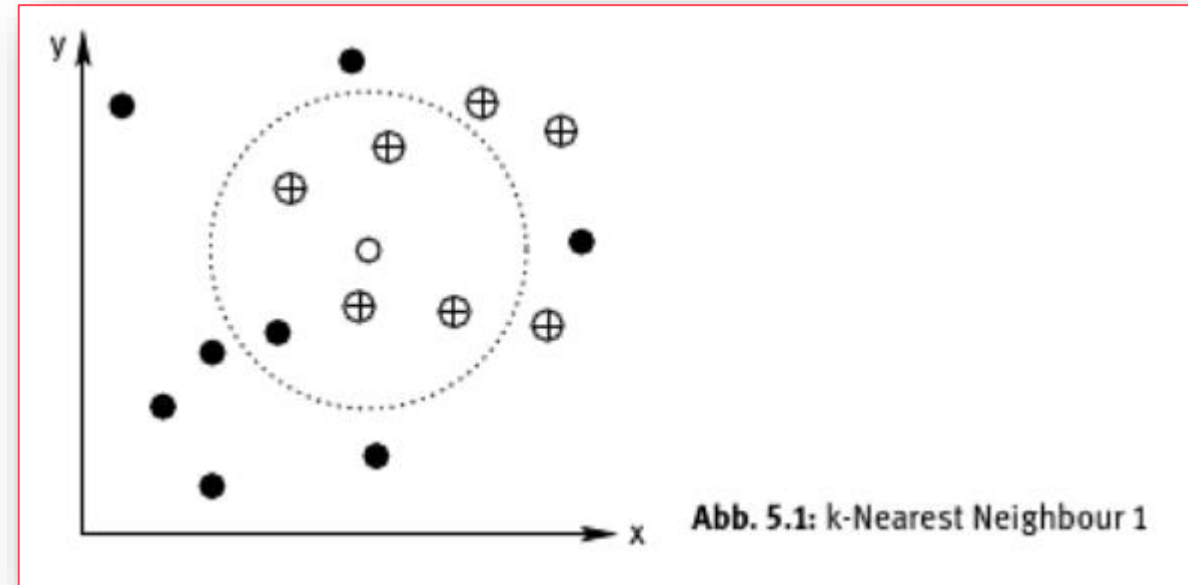
KLASSIFIKATION: KNN-ALGORITHMUS

04 CLUSTERING

1. Wiederholung/Aufgaben
Lineare Regression
2. Wiederholung/Aufgaben
Logistische Regression
3. **Klassifikation: KNN-Algorithmus**
4. Klassifikation: Entscheidungsbäume
5. Clustering: k-means-Algorithmus

DER KNN-ALGORITHMUS

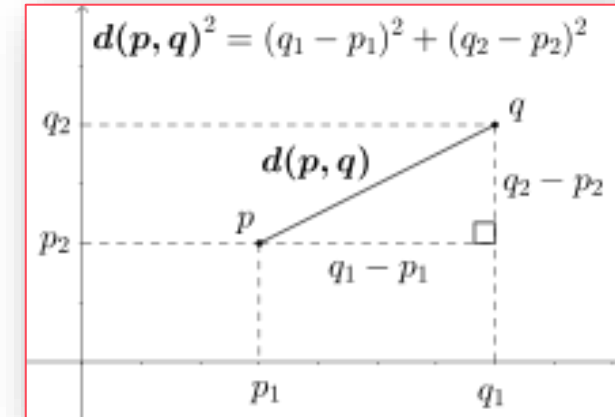
- kNN: k-Nearest-Neighbour
- einfacher Algorithmus zur Klassifikation
- Klassifikation: Zuweisung eines Objekts zu einer Klasse von Objekten
- Voraussetzung: ein Abstandsmaß ist definierbar
- instanzenbasierter Algorithmus: es wird kein "unabhängiges" Modell entwickelt, sondern das zu klassifizierende Objekt wird mit den k "nächsten Nachbarn", d.h. mit den k bisherigen Instanzen, die den geringsten Abstand haben, verglichen und nach der häufigsten Klasse vorhergesagt.
- Dabei ergeben sich ggf. unterschiedliche Klassifikationen (siehe Beispiel rechts)



BEISPIELE VON ABSTANDSFUNKTIONEN

Euklidischer Abstand:

- Seien p und q Punkte im n -dimensionalen Raum
- $Abstand_{euklid}(p, q) = \sqrt{\sum_{i=1}^n (p_i - q_i)^2}$
- Für $n = 2$ entspricht das dem bekannten Satz von Pythagoras



Hamming-Abstand:

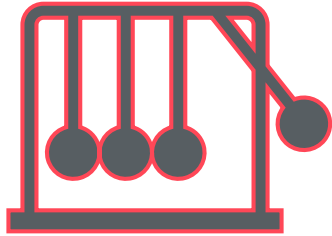
- Anzahl nicht übereinstimmender Merkmalsausprägungen zwischen zwei Objekte

	Merkmal 1	Merkmal 2	Merkmal 3	Merkmal 4	Merkmal 5	
Objekt 1	ja	ja	nein	nein	ja	
Objekt 2	ja	nein	ja	nein	ja	Hamming-Distanz
Diff-Werte	0	1	1	0	0	2

Für eine KNN-Analyse sollten die Daten numerisch und normalisiert sein.

DEMO

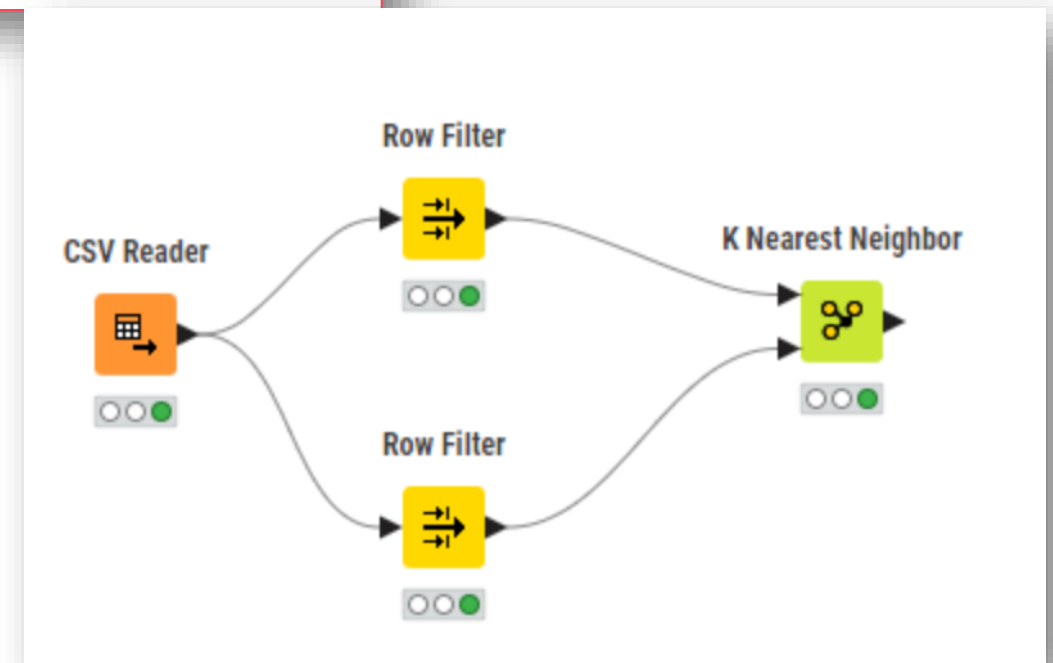
KNN IN EXCEL UND IN KNIME

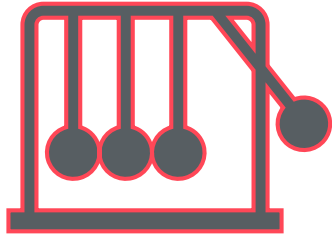


Tab. 5.1: Einkommen-Tabelle

Nr	Alter	verheiratet	Eigenheim	Akademiker	Einkommen
neu	26	1	0	1	?
1	59	1	1	1	hoch
2	55	1	0	0	gering
3	40	0	0	0	gering
4	37	1	1	1	hoch
5	26	0	0	0	gering
6	24	1	0	0	mittel
7	22	1	1	1	mittel
8	53	0	1	0	hoch

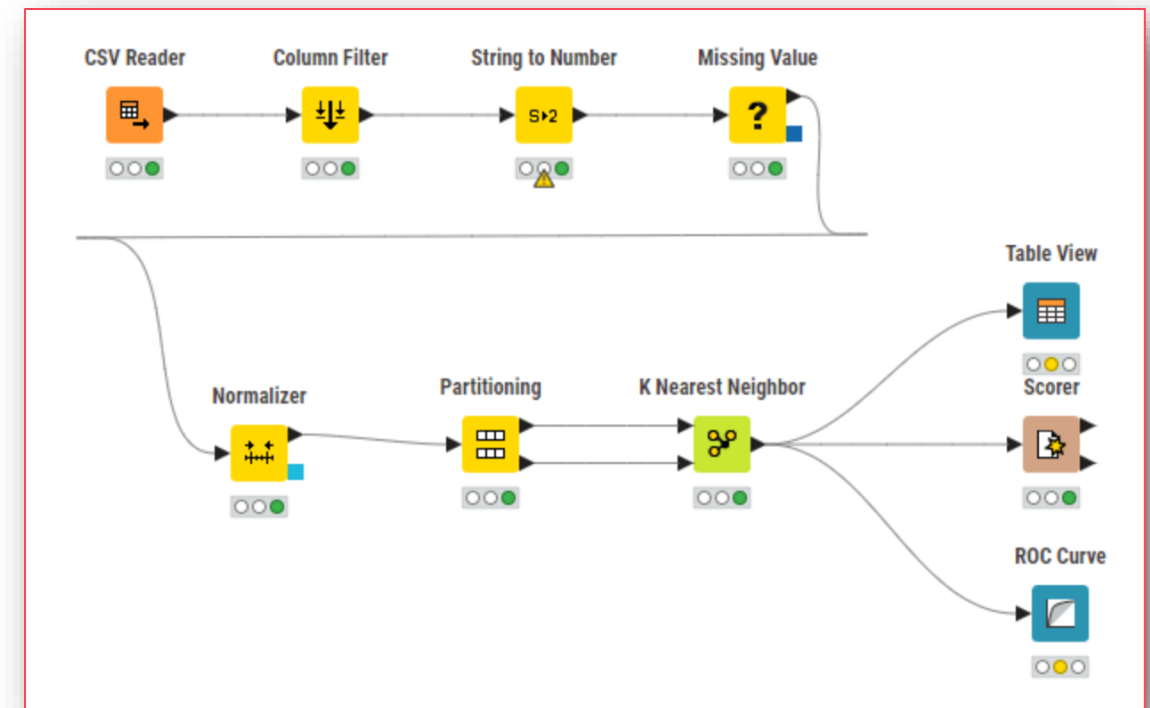
ID	Alter	Alter (gew.)	Verheiratet	Eigenheim	Akademiker	Einkommen	Euklidischer Abstand
6	24	0,05	1	0	0	hoch	1,001
7	22	0,00	1	1	1	mittel	1,006
4	37	0,41	1	1	1	mittel	1,043
2	55	0,89	1	0	0	gering	1,271
1	59	1,00	1	1	1	hoch	1,340
5	26	0,11	0	0	0	gering	1,414
3	40	0,49	0	0	0	gering	1,464
8	53	0,84	0	1	0	hoch	1,879
999	26	0,11	1	0	1	???	





'Id'	'Clump_thickness'	'Uniformity_Cell_Size'	'Uniformity_Cell_Shape'	'Marginal_Adhesion'	'Single_Epithelial_Cell_Size'	'Bare_Nuclei'	'Bland_Chromatin'	'Normal_Nucleoli'	'Mitoses'	'Class'
1000025	5	1	1	1	2	1	3	1	1	2
1002945	5	4	4	5	7	10	3	2	1	2
1015425	3	1	1	1	2	2	3	1	1	2
1016277	6	8	8	1	3	4	3	7	1	2
1017023	4	1	1	3	2	1	3	1	1	2
1017122	8	10	10	8	7	10	9	7	1	4
1018099	1	1	1	1	2	10	3	1	1	2
1018561	2	1	2	1	2	1	3	1	1	2
1033078	2	1	1	1	2	1	1	1	5	2
1033078	4	2	1	1	2	1	2	1	1	2
1035283	1	1	1	1	1	1	3	1	1	2
1036172	2	1	1	1	2	1	2	1	1	2
1041801	5	3	3	3	2	3	4	4	1	4
1043999	1	1	1	1	2	3	3	1	1	2
1044572	8	7	5	10	7	9	5	5	4	4

- Ein häufig verwendet Datensatz der Universität Wisconsin
- Er enthält 10 Merkmale, die bei einer Vorsorgeuntersuchung erfasst werden.
- Das Ergebnis besteht aus zwei Werten:
 - 2: gutartig
 - 4: bösartig



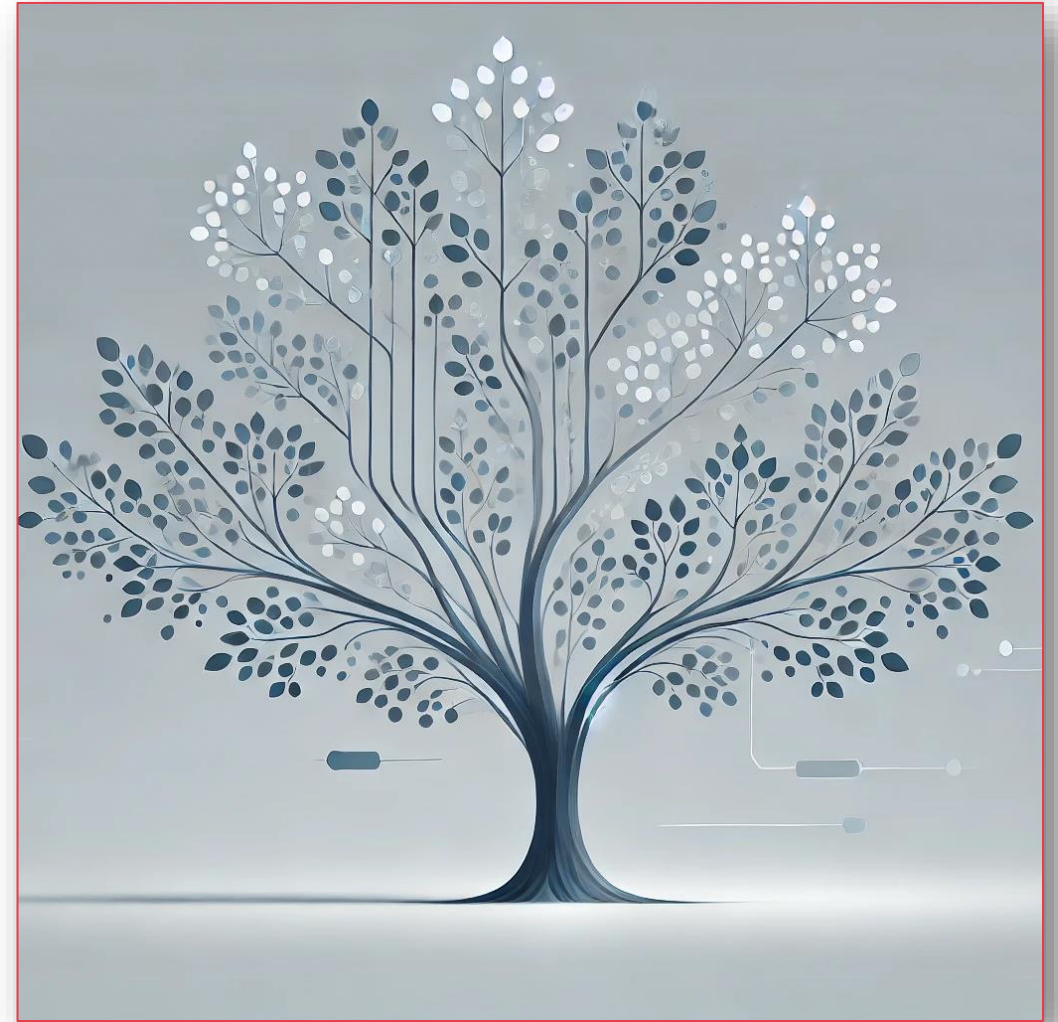
KLASSIFIKATION: ENTSCHEIDUNGSBÄUME

04 CLUSTERING

1. Wiederholung/Aufgaben
Lineare Regression
2. Wiederholung/Aufgaben
Logistische Regression
3. Klassifikation: KNN-Algorithmus
4. **Klassifikation: Entscheidungsbäume**
5. Clustering: k-means-Algorithmus

ENTSCHEIDUNGSBÄUME

- Entscheidungsbaum-Verfahren (Decision Trees)
- Wichtiger Algorithmus zur Klassifikation
- Abstandsmaß ist nicht notwendig
- modellbasierter Algorithmus: es wird ein unabhängiges Modell entwickelt, das in Form eines Entscheidungsbaums versucht, eine Klassifikation zu erreichen
- Weitere Erklärung an Hand eines Beispiels...



SHALL I PLAY TENNIS → LOOK AT THE WEATHER FORECAST ☺

- Ausgangspunkt: Ein Datensatz zu Wetterdaten und (davon abhängig) der Entscheidung Tennis zu spielen
- Das Beispiel besteht nur aus Nominalskalen
- Grundidee:
 - Wir wählen ein Merkmal aus (z.B. outlook) und prüfen, ob wir je nach Merkmalsausprägung bereits eine Entscheidung treffen können:

Tab. 5.6: Daten Wetter-Beispiel

Tag	outlook	temperature	humidity	windy	play
1	sunny	hot	high	false	no
2	sunny	hot	high	true	no
3	overcast	hot	high	false	yes
4	rainy	mild	high	false	yes
5	rainy	cool	normal	false	yes
6	rainy	cool	normal	true	no
7	overcast	cool	normal	true	yes
8	sunny	mild	high	false	no
9	sunny	cool	normal	false	yes
10	rainy	mild	normal	false	yes
11	sunny	mild	normal	true	yes
12	overcast	mild	high	true	yes
13	overcast	hot	normal	false	yes
14	rainy	mild	high	true	no

SCHRITT 1: WÄHLE OUTLOOK

Tag	outlook	temperature	humidity	windy	play
3	overcast	hot	high	false	yes
7	overcast	cool	normal	true	yes
12	overcast	mild	high	true	yes
13	overcast	hot	normal	false	yes
4	rainy	mild	high	false	yes
5	rainy	cool	normal	false	yes
6	rainy	cool	normal	true	no
10	rainy	mild	normal	false	yes
14	rainy	mild	high	true	no
1	sunny	hot	high	false	no
2	sunny	hot	high	true	no
8	sunny	mild	high	false	no
9	sunny	cool	normal	false	yes
11	sunny	mild	normal	true	yes

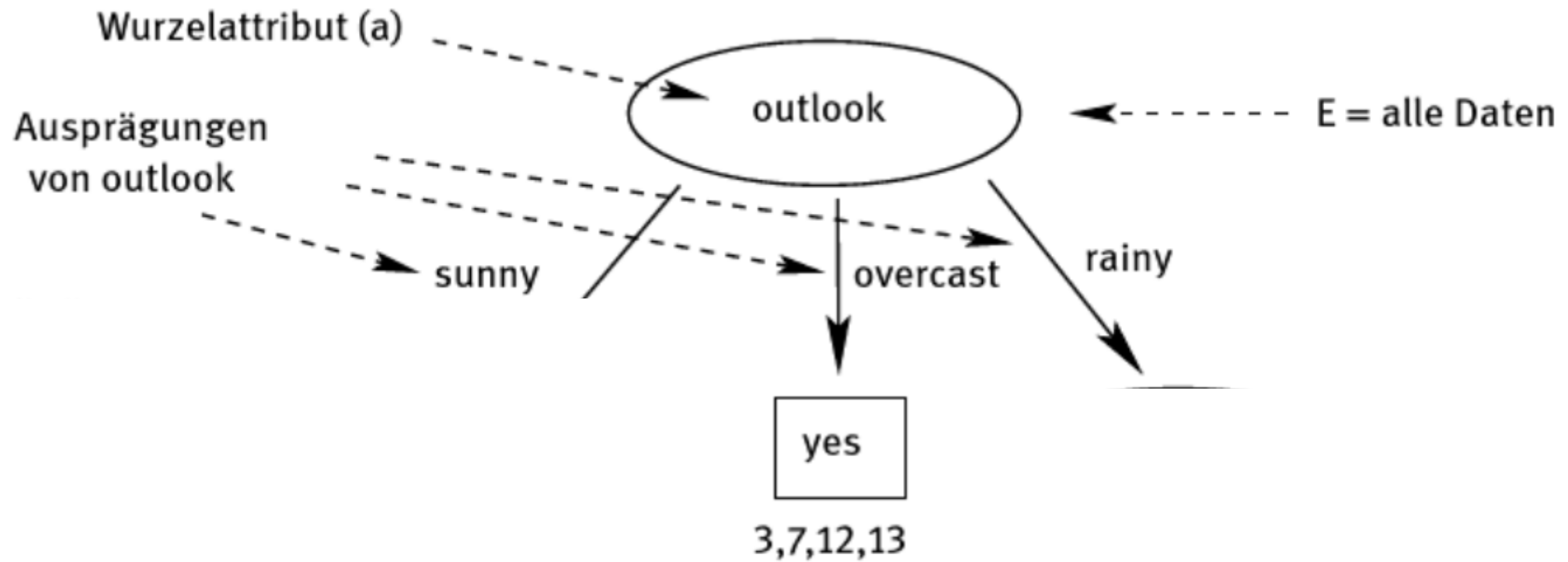
outlook = overcast → play tennis

Tag	outlook	temperature	humidity	windy	play
4	rainy	mild	high	false	yes
5	rainy	cool	normal	false	yes
6	rainy	cool	normal	true	no
10	rainy	mild	normal	false	yes
14	rainy	mild	high	true	no
1	sunny	hot	high	false	no
2	sunny	hot	high	true	no
8	sunny	mild	high	false	no
9	sunny	cool	normal	false	yes
11	sunny	mild	normal	true	yes

outlook = rainy/sunny → weiter analysieren

- Bei Auswahl von outlook ergibt sich:
 - Klare Empfehlung falls outlook = overcast
 - Alle anderen Ausprägungen: weitere Analyse notwendig

S



- Klare Empfehlung falls outlook = overcast
- Alle anderen Ausprägungen: weitere Analyse notwendig

SCHRITT 2: ANALYSE FÜR OUTLOOK = SUNNY UND HUMIDITY

Tag	outlook	temperature	humidity	windy	play
1	sunny	hot	high	false	no
2	sunny	hot	high	true	no
8	sunny	mild	high	false	no
9	sunny	cool	normal	false	yes
11	sunny	mild	normal	true	yes

humidity = high → don't play tennis

humidity = normal → play tennis

- Bei Auswahl von humidity ergibt sich:
 - Klare Empfehlung falls
 - humidity = high
 - humidity = normal
 - keine weitere Analyse notwendig

S

Ta

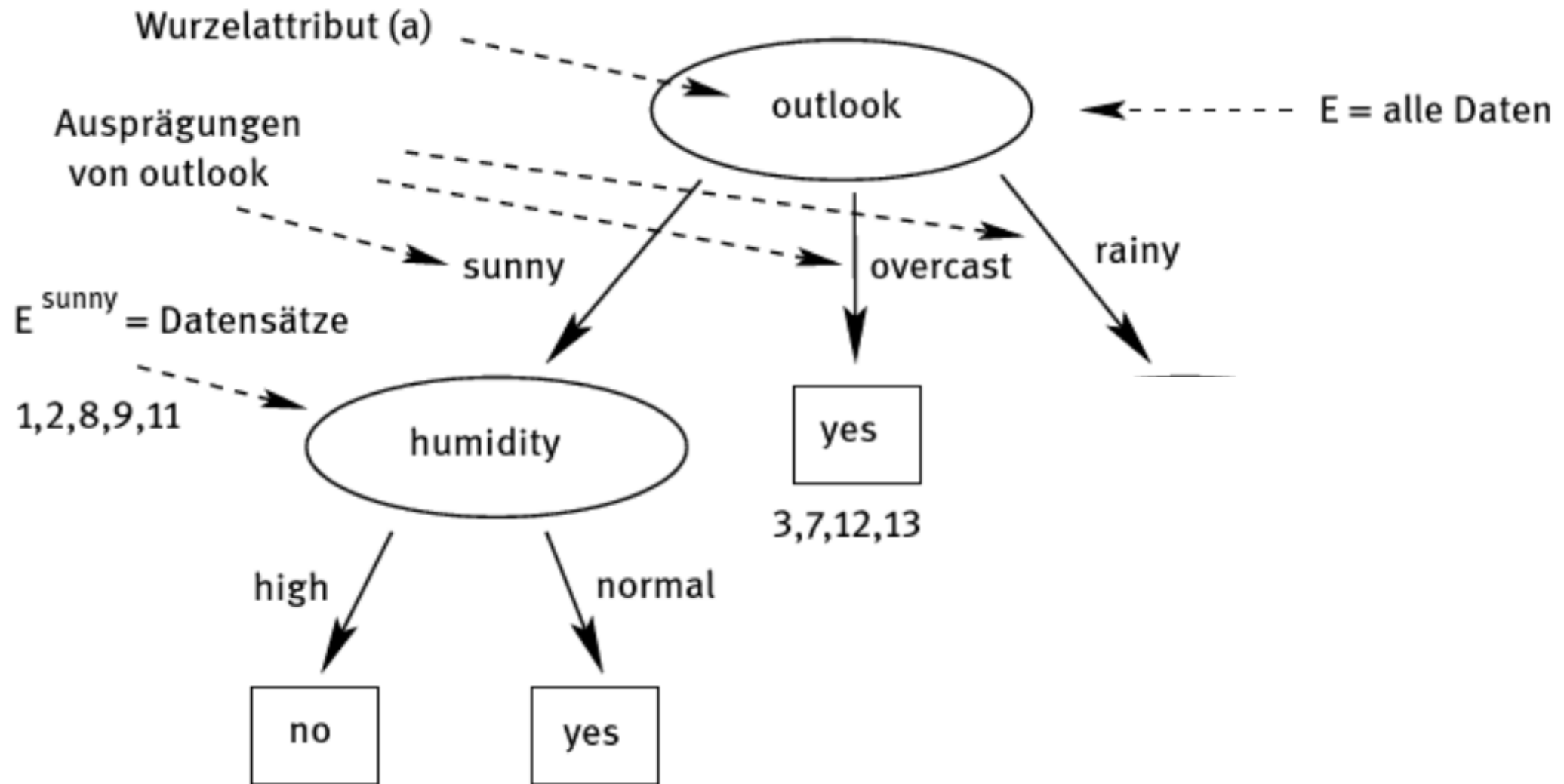
2

3

8

9

1



keine weitere Analyse notwendig

SCHRITT 2: ANALYSE FÜR OUTLOOK = RAINY UND WINDY

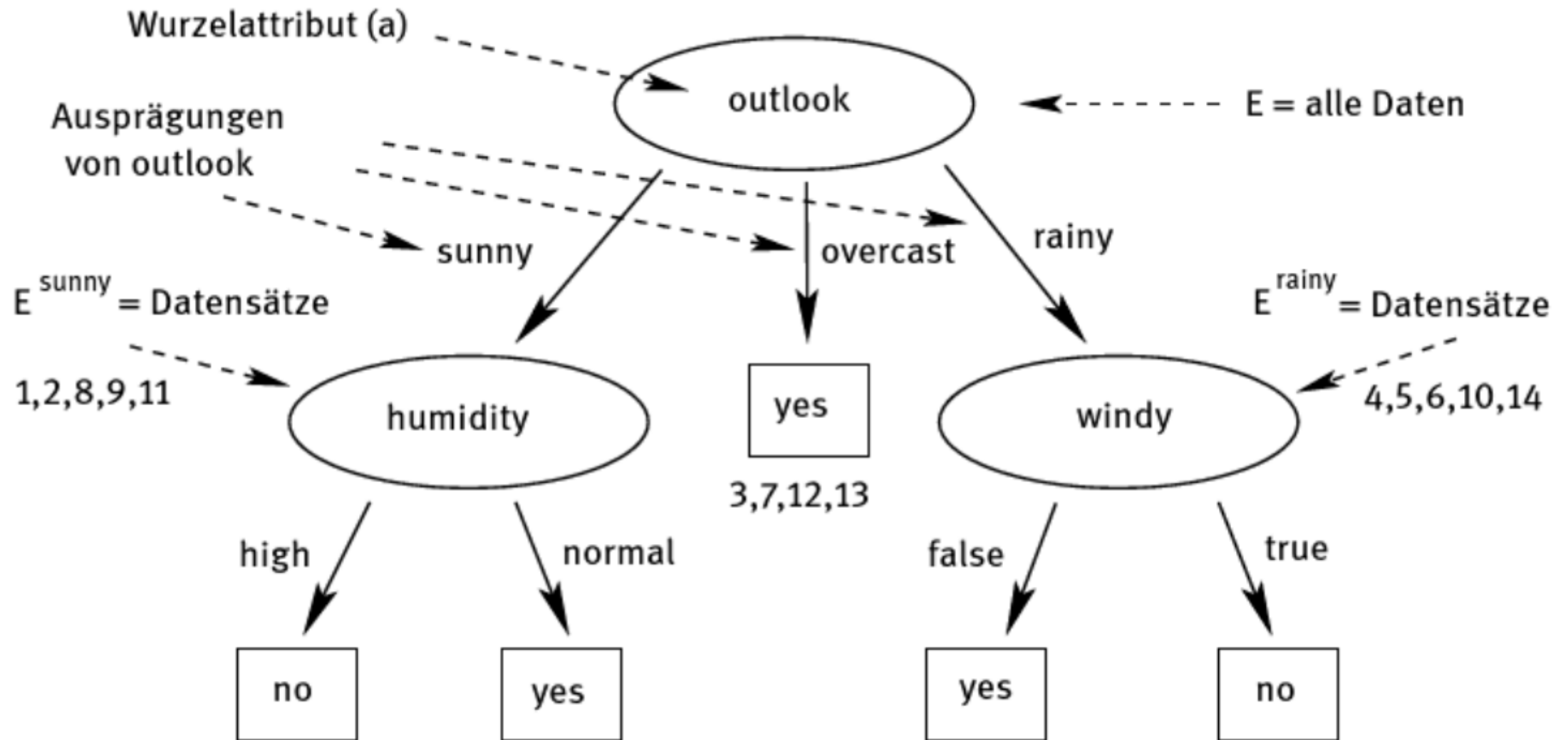
Tag	outlook	temperature	humidity	windy	play
4	rainy	mild	high	false	yes
5	rainy	cool	normal	false	yes
10	rainy	mild	normal	false	yes
6	rainy	cool	normal	true	no
14	rainy	mild	high	true	no

} windy = false → play tennis
 } windy = true → don't play tennis

- Bei Auswahl von windy ergibt sich ebenfalls klare Empfehlungen:
- Keine weitere Analyse notwendig

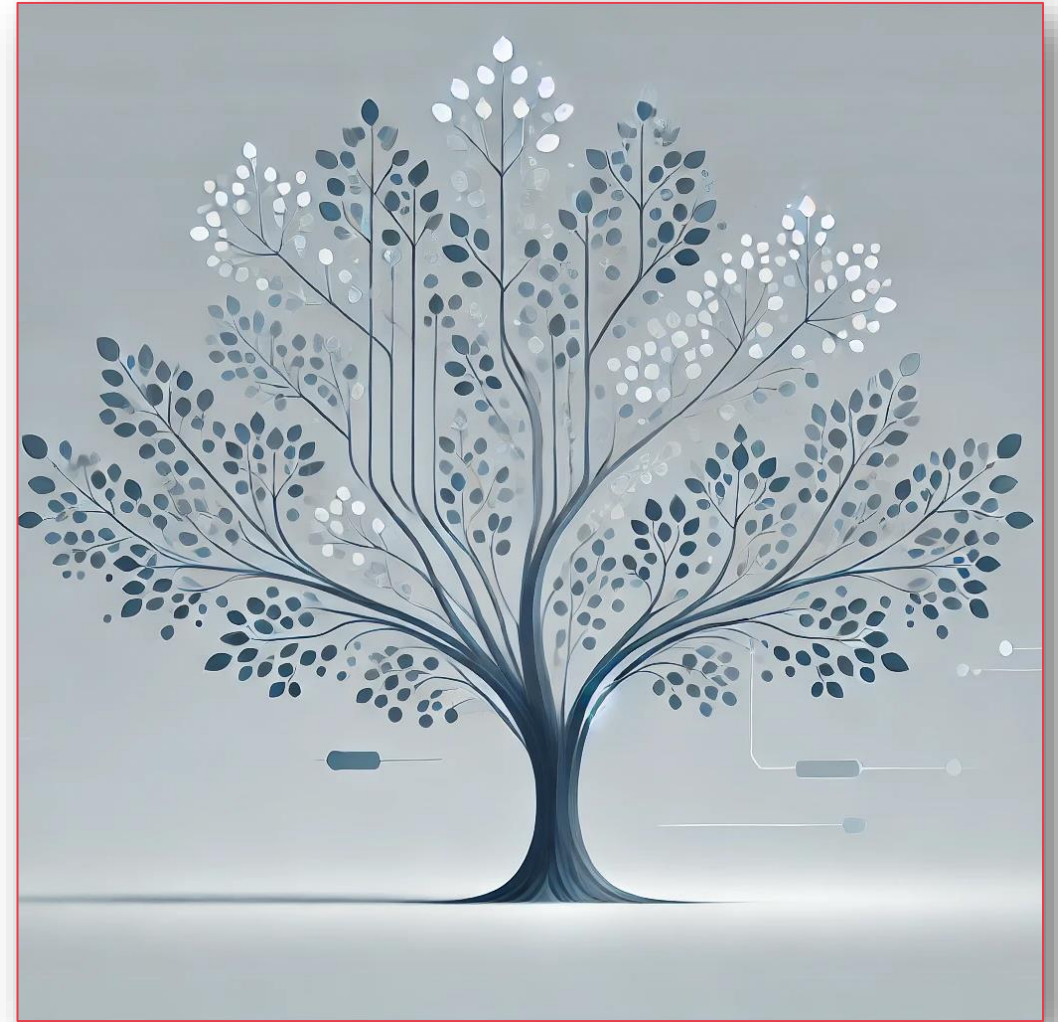
S

Tag
4
5
10
6
14



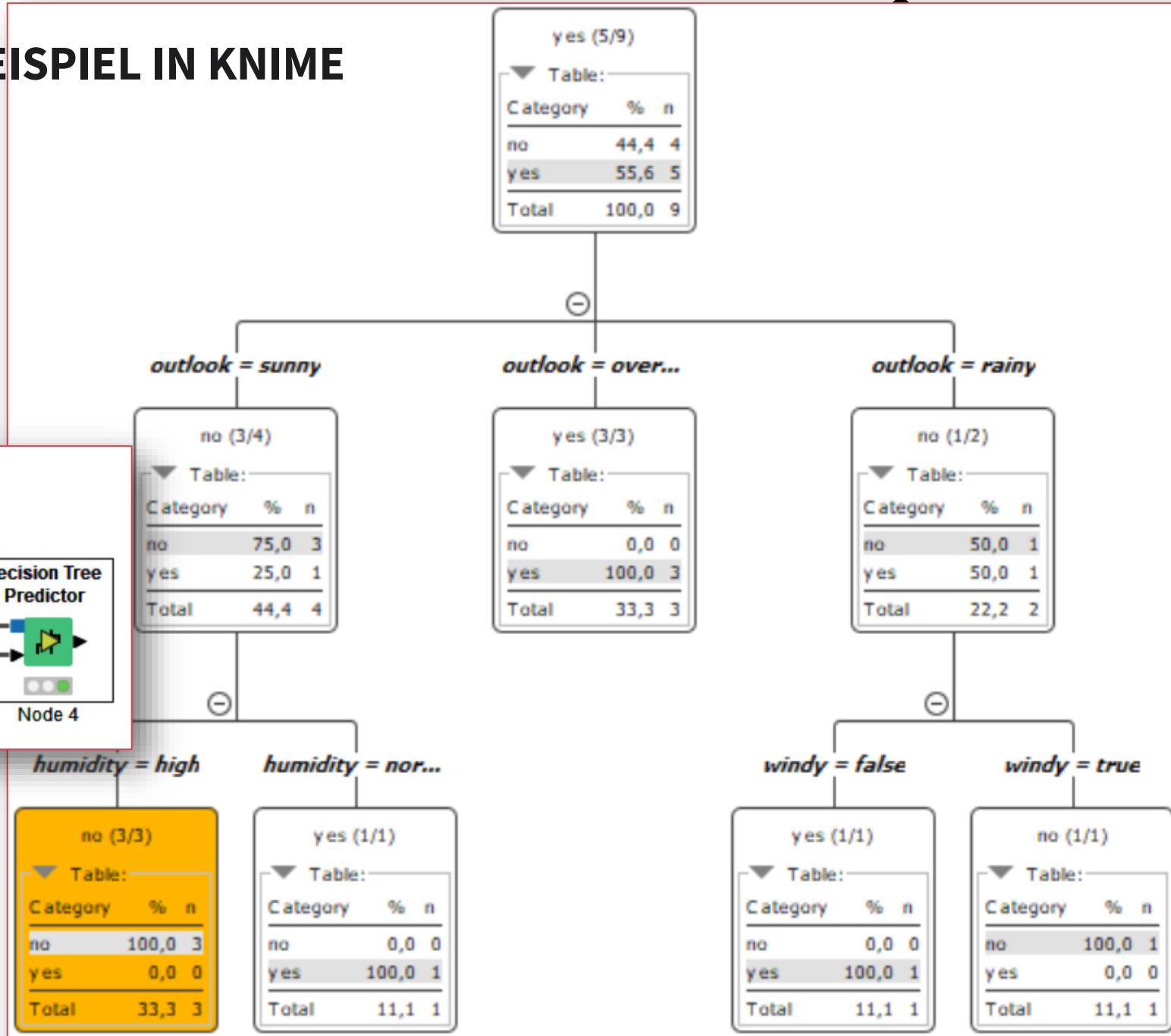
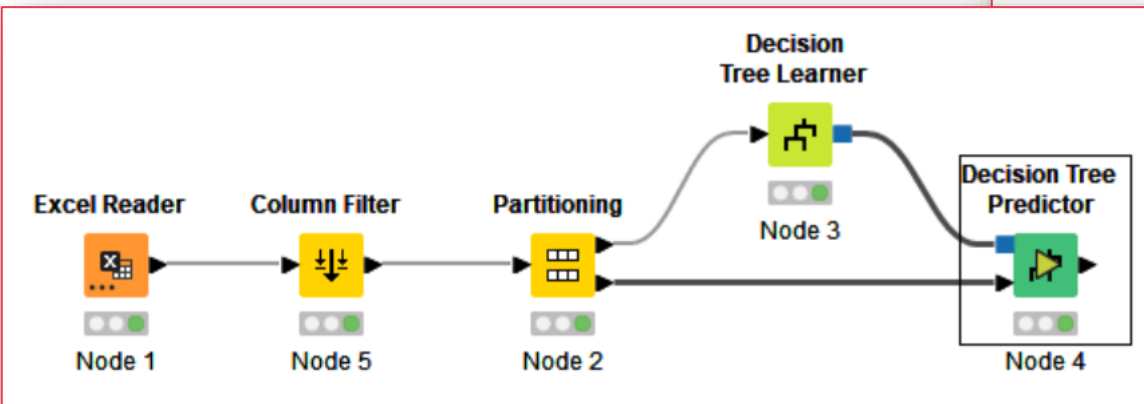
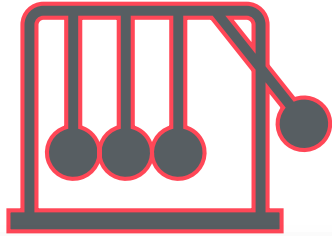
BEWERTUNG:

- Entscheidend ist hier offensichtlich die richtige Auswahl der Merkmale
- Diese kann erfolgen:
 - Händisch → nicht realistisch bei größeren Datensätzen
 - Zufällig → kann zu guten oder schlechten Ergebnissen führen
 - per "Algorithmus"



DEMO

WETTERBEISPIEL IN KNIME



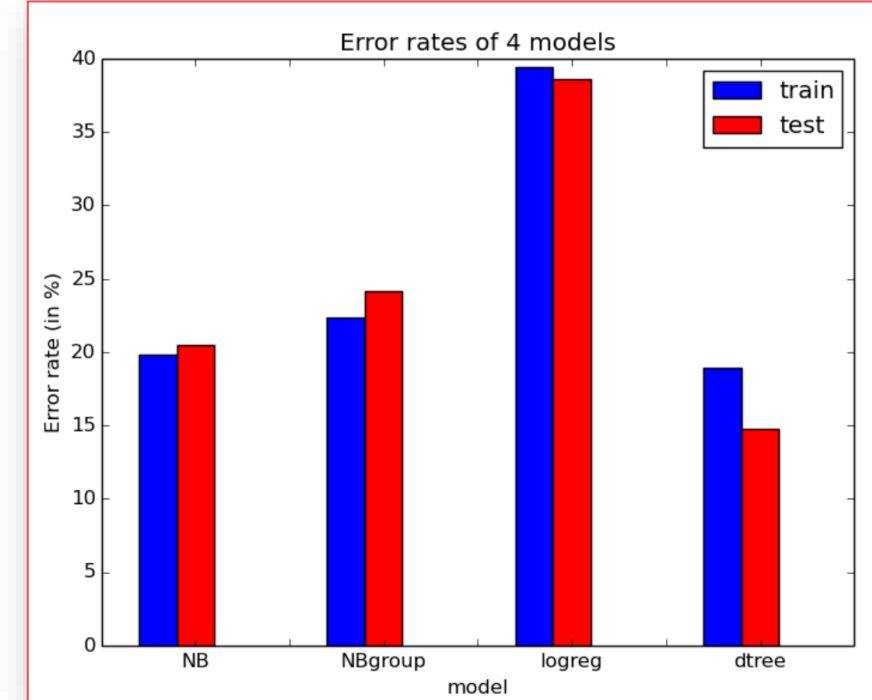
MACHINE LEARNING – „ARTEN“ – ZWISCHENSTAND

Oberbegriff	Überwachtes Lernen (supervised learning)		Unüberwachtes Lernen (unsupervised learning)			Reinforcement Learning ...
Ziel:	✓ „Regression“: Vorhersage eines metrischen Wertes	Klassifikation: Vorhersage eines kategorialen Wertes	Clustering: Bildung von Klassen gleicher Objekte	✓ Assoziation: Suchen von wenn- dann-Beziehungen	<i>Anomaly Detection</i>	
Anwendbare Verfahren:	✓ Einfache Lineare Regression	✓ Logistische Regression	K-means-Algorithmus	✓ Apriori Algorithmus		
	✓ Multilineare Regression	✓ kNN-Algorithmus	<i>Hierarchisches Clustering</i>	...		
	✓ Polynomiale Regression	✓ Entscheidungsbaum (Decision Tree)	<i>DBSCAN (Density-Based Spatial ...)</i>			
	Logistische Regression ...	<i>Support Vector Maschine</i> <i>Random Forest</i> Naive Bayes Neural Networks 			

Legende:
 Wird in der Vorlesung behandelt
 Wird NICHT in der Vorlesung behandelt
 ✓ Bereits behandelt
 Thema heute

WANN MACHE ICH WAS?

- Klassifikationsalgorithmen verwenden wir, wenn wir „labeled data“ haben, und der Zielwert kategorial ist.
- Faustregeln: Verwende ...
 - Logistische Regression: Wenn du lineare Beziehungen und Interpretierbarkeit brauchst.
 - k-NN: Wenn die Daten klein und einfach strukturiert sind, ohne lineare Annahmen.
 - Decision Tree: Wenn die Daten komplex sind und du nichtlineare Beziehungen und Interaktionen modellieren willst.
- Am besten testet ihr alle Algorithmen auf demselben Datensatz.
- Als Beispiel sehr hilfreich: Lest die Datei *income analysis Lemon et al.pdf* (findet ihr in Teams)



Classifier	Train Error	Test Error
Baselines	24.008%	23.623%
Naive Bayes	19.893%	20.432%
Naive Bayes (Grouped)	22.353%	24.128%
Logistic Regression	39.370%	38.612%
Decision Tree	18.940%	14.778%

WANN IST MEIN MODEL „GUT“?

– Die wichtigsten Gütemaße kann man der Confusion Matrix entnehmen:

Die Confusion-Matrix ist eine Tabelle, die hilft, die Leistung eines Klassifikationsmodells zu bewerten. Sie zeigt, wie oft das Modell richtig oder falsch liegt, und teilt die Ergebnisse in vier Felder auf:

	Tatsächlich Positiv	Tatsächlich Negativ
Vorhergesagt Positiv	True Positives (TP): richtig positiv	False Positives (FP): falsch positiv
Vorhergesagt Negativ	False Negatives (FN): falsch negativ	True Negatives (TN): richtig negativ

Erklärung der Begriffe:

- **True Positives (TP):** Fälle, bei denen das Modell „Positiv“ vorhergesagt hat und es auch tatsächlich positiv ist.
- **False Positives (FP):** Fälle, bei denen das Modell „Positiv“ vorhergesagt hat, aber es tatsächlich negativ ist (auch „False Alarm“ genannt).
- **False Negatives (FN):** Fälle, bei denen das Modell „Negativ“ vorhergesagt hat, aber es tatsächlich positiv ist.
- **True Negatives (TN):** Fälle, bei denen das Modell „Negativ“ vorhergesagt hat und es auch tatsächlich negativ ist.

Confusion Matrix - 3:9 - Scorer		
File Hilite		
'Class' \ Cl...	2	4
2	128	1
4	8	68
Correct classified: 196		
Wrong classified: 9		
Accuracy: 95,61%		
Error: 4,39%		
Cohen's kappa (κ): 0,904%		

1. Accuracy (Genauigkeit)

- **Definition:** Anteil der richtig klassifizierten Fälle an der Gesamtanzahl der Fälle.

$$\text{Accuracy} = \frac{\text{True Positives (TP)} + \text{True Negatives (TN)}}{\text{Gesamtanzahl der Instanzen}}$$

- Zeigt die allgemeine Modellleistung, kann jedoch bei unausgebalancierten Klassen irreführend sein.

2. Precision (Präzision)

- **Definition:** Anteil der korrekt als positiv vorhergesagten Fälle an allen positiven Vorhersagen.

$$\text{Precision} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Positives (FP)}}$$

- Wichtig, wenn falsch-positive Vorhersagen vermieden werden sollen, z.B. bei Diagnosen.

3. Cohen's Kappa

- Maß zur Bewertung der Übereinstimmung zwischen Modellvorhersagen und tatsächlichen Werten, unter Berücksichtigung zufälliger Übereinstimmungen.
- Hilfreich bei unausgebalancierten Klassen, da es eine robustere Einschätzung der Modellleistung bietet.

CLUSTERING: K-MEANS-ALGORITHMUS

04 CLUSTERING

1. Wiederholung/Aufgaben
Lineare Regression
2. Wiederholung/Aufgaben
Logistische Regression
3. Klassifikation: KNN-Algorithmus
4. Klassifikation: Entscheidungsbäume
5. **Clustering: k-means-Algorithmus**

Klassifikation

- kategoriale Ergebniswerte
- ist ein prädiktives Verfahren
 - d.h. versucht auf Basis von Beobachtungsdaten
 - ein Modell zu finden,
 - das eine Vorhersage zur Zugehörigkeit eines unbeobachteten Objekts x
 - zu einer bekannten Klasse C zu machen
 - Kernfrage: Ist $x \in C$?
- in der Regel: überwachtes Lernen

Clustering

- ebf.: kategoriale Ergebniswerte
- ist ein deskriptives Verfahren
 - d.h. v versucht auf Basis von Beobachtungsdaten
 - sinnvolle Klassen innerhalb der Beobachtungseinheiten zu finden,
 - die "ähnliche" Objekte beinhalten
 - Kernfrage: Gibt es sinnvolle Klassen $C_i \subseteq C$?
- in der Regel: unüberwachtes Lernen

Wir betrachten zunächst den k-Means-Algorithmus als Beispiel

DER K-MEANS-ALGORITHMUS

Listing 6.2 (k-Means – Basis-Variante).

PROCEDURE k-Means

Erzeuge (zufällig) k Anfangscluster C_i

//Alle Objekte x werden (zufällig) einem Cluster zugeordnet

REPEAT

Tausch_erfolgt := false

Bestimme die Centroide $\bar{x}_1, \bar{x}_2, \dots, \bar{x}_k$ der Cluster

Für alle x aus den Eingabedaten: Weise x demjenigen Cluster C_i zu,
zu dessen Centroiden \bar{x}_i x die geringste Distanz hat

IF ein x wird einem anderen Cluster zugewiesen **THEN** Tausch_erfolgt := true

UNTIL NOT Tausch_erfolgt

END k-Means

Die Anzahl der (gesuchten) Klassen k wird vorgegeben.

Eine Abstandsfunktion muss definiert sein (siehe k-Nearest-Neighbour)

Der Algorithmus versucht nun k Teilmengen zu finden, ...

... bei denen die Summen der Abstände zu einem jeweils gemeinsamen Punkt (dem Centroid) minimal ist

- Dabei wird mit einer zufälligen Verteilung angefangen ...
- ... um dann solange Elemente einer anderen Menge zuzuordnen
- ... bis keine Optimierung mehr möglich ist.

DER K-MEANS-ALGORITHMUS "IN BILDERN"

