INTELLIGENTE DATENANALYSE IN MATLAB

Überwachtes Lernen: Entscheidungsbäume

Literatur

- Stuart Russell und Peter Norvig: Artificial Intelligence.
- Andrew W. Moore: http://www.autonlab.org/tutorials.

Überblick

- Schritte der Datenanalyse:

Daten-vorverarbeitung → Problemanalyse → Problemlösung → Anwendung der Lösung

Daten-vorverarbeitung

Aggregation und Selektion von Daten.

Integration und Säuberung der Daten.

Feature-Extraktion.

Problemanalyse

Bestimmen von gegeb./gesuchten Größen.

Wahl des Performanzmaß/ Zielkriteriums.

Modellraum und Modellannahmen.

Problemlösung

Algorithmen für das Optimieren des Zielkriteriums finden.

Implementieren der Algorithmen.

Anwendung der Lösung

Modell-Selektion & -Anpassung.

Training & Evaluation des Modells auf gegebenen Daten.

Vorhersage für neue Daten.

Überwachtes Lernen

Problemstellung

- Gegeben: Trainingsdaten mit <u>bekannten</u> Zielattributen (gelabelte Daten).
- Eingabe: Instanz (Objekt, Beispiel, Datenpunkt, Merkmalsvektor) = Vektor mit Attribut-Belegungen.
- Ausgabe: Belegung des/der Zielattribut(e).
  - Klassifikation: Nominaler Wertebereich des Zielattributs.
  - Ordinale Regression: Ordinaler Wertebereich des Zielattributs.
  - Regression: Numerischer Wertebereich des Zielattributs.
- Gesucht: Modell $f : \mathbf{x} \mapsto y$.

Überwachtes Lernen

Beispiel

- Beispiel *binäre Klassifikation*:

Tag	Bewölkung	Temperatur	Luftfeuchtigkeit	Wind	Tennis spielen?
1	sonnig	warm	hoch	wenig	nein
2	sonnig	warm	hoch	stark	nein
3	bedeckt	warm	hoch	wenig	ja
4	Regen	mild	hoch	wenig	ja
5	Regen	kühl	normal	wenig	ja
6	Regen	kühl	normal	stark	nein
7	bedeckt	kühl	normal	stark	ja
8	sonnig	mild	hoch	wenig	nein
9	sonnig	kühl	normal	wenig	ja
10	Regen	mild	normal	wenig	ja
11	sonnig	mild	normal	stark	?
12	bedeckt	mild	hoch	stark	?
13	bedeckt	warm	normal	wenig	?
14	Regen	mild	hoch	stark	?

Trainingsdaten

Testdaten

Zielgröße

Überwachtes Lernen

Arten von Modellen

- Entscheidungsbäume/Regelsysteme:
  - Klassifikations-, Regressions-, Modellbaum.
- Lineare Modelle:
  - Trennebenen, Regressionsgerade.
- Nicht-lineare Modelle, linear in den Parametern:
  - Nicht-lineare Datentransformation + lineares Modell.
  - Kernel-Modell.
  - Probabilistisches Modell.
- Nicht-lineare Modelle, nicht-linear in den Parametern:
  - Neuronales Netz.

Entscheidungsbäume

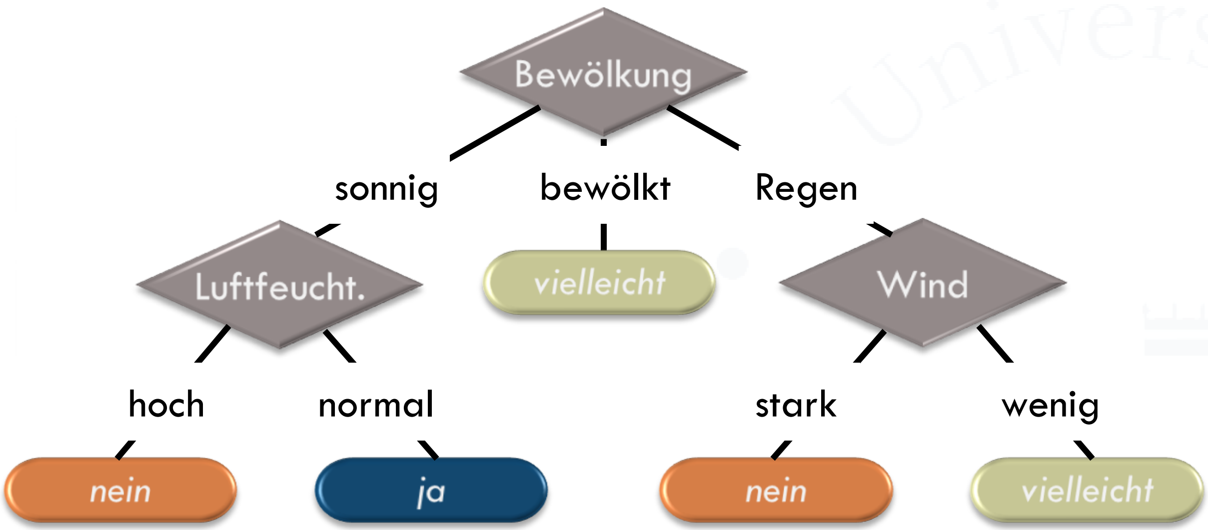
Definition

- Entscheidungsbaum: Graph-Darstellung von Regeln zur Klassifikation/Regression.
- Innere Knoten: Attributbelegung prüfen.
  - Nominale/binäre Attribute (prüfen auf gleich/ungleich).
  - Numerische/ordinale Attribute (prüfen auf größer/kleiner).
- Kanten: Attributbelegung.
- Blätter: Liefern Klassenlabel/Regressionswerte.

Entscheidungsbäume

Beispiel

- Beispiel „Tennis spielen“:



Regel: (Bewölkung = sonnig) ^ (Luftfeuchtigkeit = normal) $\Rightarrow$ ja

Entscheidungsbäume

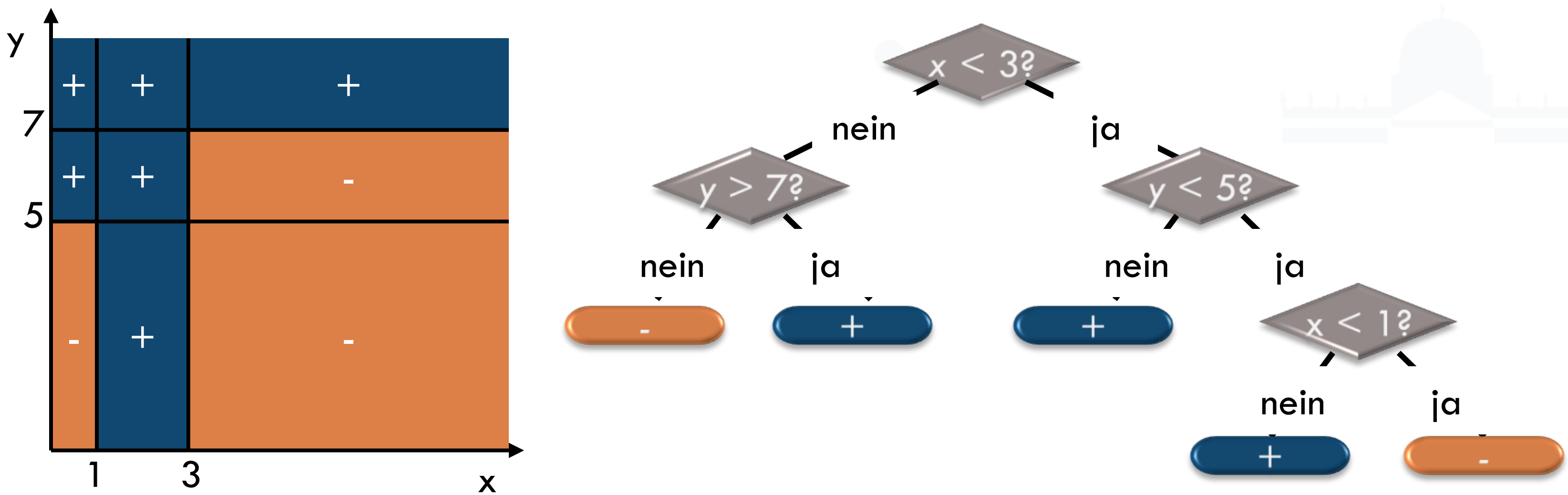
Besonders geeignet …

- Für kleine und mittelgroße Klassifikationsprobleme.
- Falls Interpretierbarkeit/Begründung der Entscheidung notwendig bzw. falls explizite Abhängigkeiten/Regeln zwischen den Attributen erkannt werden sollen.
- Für Daten mit wenig, überwiegend ordinalen/binären Attributen.
- Für Daten mit fehlerhaften/fehlenden Attributen.
- Falls schnelles & einfaches Verfahren gewünscht.

Entscheidungsbäume

Beispiel

- Instanzen (Vektoren mit *m* Einträgen) liegen im *m*-dimensionalen Eingaberaum.
- Dieser wird in achsenparallele Rechtecke zerlegt.



Entscheidungsbäume

Lernen von Entscheidungsbäumen

- Ziel: Optimaler Entscheidungsbaum.
  - Maximum Likelihood (ML): Daten möglichst gut erklären.
  - Maximum A-Posteriori (MAP): Daten möglichst gut erklärt & geringe Komplexität (wenig Knoten/Kanten).
    - Leichter interpretierbar.
    - Bessere Generalisierungsfähigkeit.
    - Entscheidungen in den Blättern stützen sich auf mehr Beispiele.
- Problem: Anzahl binärer Entscheidungsbäume der Tiefe t für m Attributen ist $O(m^{2^t}) \Rightarrow$ kleinsten konsistenten Baum finden ist NP-hart!

Entscheidungsbäume

Lernen von Entscheidungsbäumen (nominale Attribute)

- Ansatz: Greedy-Suche (z.B. Top-Down-Suche).
- Beispiel: Klassifikationsbaum für nominale Attribute.

BuildTree(*Instanzen*,*Attribute*)

```
IF Alle Instanzen sind aus einer Klasse THEN
   RETURN (Blatt mit Klasse)
IF Attribute = Ø THEN
   RETURN (Blatt mit häufigster Klasse)
Wähle bestes Attribut A ∈ Attribute als Wurzel W
FOR Alle Belegungen v von A
   Erzeuge Kante (mit Bedingung A = v) zwischen Wurzel und
   BuildTree({x ∈ Instanzen mit x.A = v}, Attribute \ {A})
RETURN (Baum mit Wurzel W)
```

Welches Attribut ist das Beste?

Entscheidungsbäume

Lernen von Entscheidungsbäumen (nominale Attribute)

- Splitting-Kriterium („bestes Attribut“ finden):
  - Daten $D = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_n, y_n)\}$ partitionieren bzgl. Attribut A (mit k möglichen Belegungen): $\{D_1, D_2, \ldots, D_k\}$.
  - Ziel: „Unsicherheit“ (Entropie) über das Klassenlabel durch Partitionierung verringern.
- Entropie: $\mathrm{H}_Z = \mathrm{E}[h(Z)]$
  - Theoretische Verteilung der Zufallsvariable Z (gegeben durch Verteilungsfunktion): $\mathrm{H}_Z = -\int p(z) \log p(z) \partial z$
  - Empirische Verteilung der Zufallsvariable Z (gegeben durch Datenpunkte/Instanzen): $\mathrm{H}_Z = -\sum_z p(z) \log p(z)$

Entscheidungsbäume

Lernen von Entscheidungsbäumen (nominale Attribute)

- Entropie über Klassenlabel <u>vor</u> Partitionierung:

$$H_{Y\sim D} = -\sum_{y\in\{+1,-1\}} p_D(y)\log p_D(y)$$

- Erwartete Entropie über Klassenlabel <u>nach</u> Partitionierung:

$$E\left[H_{Y\sim D_i}\right] = \sum_{i=1}^{k} \frac{|D_i|}{|D|} H_{Y\sim D_i}$$

- Erwartete Verringerung der Entropie (*Information Gain*):

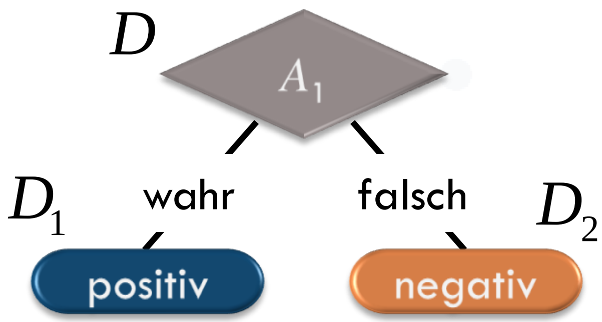
$$IG_{Y,\{D_1,\dots,D_k\}} = H_{Y\sim D} - E\left[H_{Y\sim D_i}\right]$$

Entscheidungsbäume

Lernen von Entscheidungsbäumen (nominale Attribute)

- Beispiel: Partitionierung der Daten D bzgl. binären Attributs A_1 in D_1 und D_2.

$$H_{Y\sim D} = -\sum_{y\in\{+1,-1\}} p_D(y)\log p_D(y)$$



$$H_{Y\sim D_1} = -\sum_{y\in\{+1,-1\}} p_{D_1}(y)\log p_{D_1}(y)$$

$$H_{Y\sim D_2} = -\sum_{y\in\{+1,-1\}} p_{D_2}(y)\log p_{D_2}(y)$$

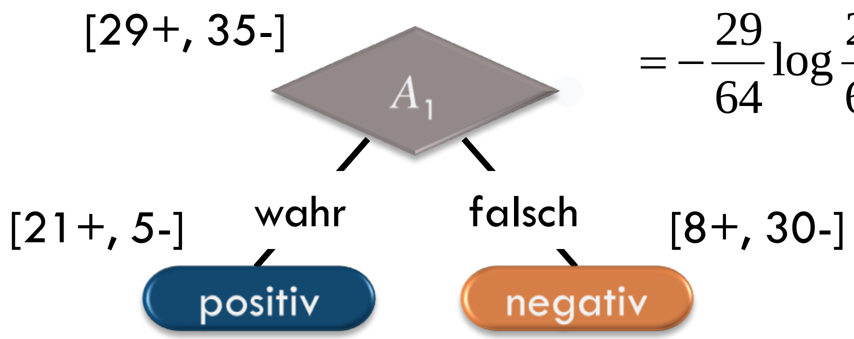
Entscheidungsbäume

Lernen von Entscheidungsbäumen (nominale Attribute)

- Beispiel: Partitionierung der Daten D bzgl. binären Attributs A_1 in D_1 und D_2.

$$\mathrm{H}_{Y \sim D} = -\sum_{y \in \{+1,-1\}} p_D(y) \log p_D(y)$$

$$= -\frac{29}{64} \log \frac{29}{64} - \frac{35}{64} \log \frac{35}{64} = 0{,}994$$



$$\mathrm{H}_{Y \sim D_1} = -\sum_{y \in \{+1,-1\}} p_{D_1}(y) \log p_{D_1}(y)$$

$$= -\frac{21}{26} \log \frac{21}{26} - \frac{5}{26} \log \frac{5}{26} = 0{,}706$$

$$\mathrm{H}_{Y \sim D_2} = -\sum_{y \in \{+1,-1\}} p_{D_2}(y) \log p_{D_2}(y)$$

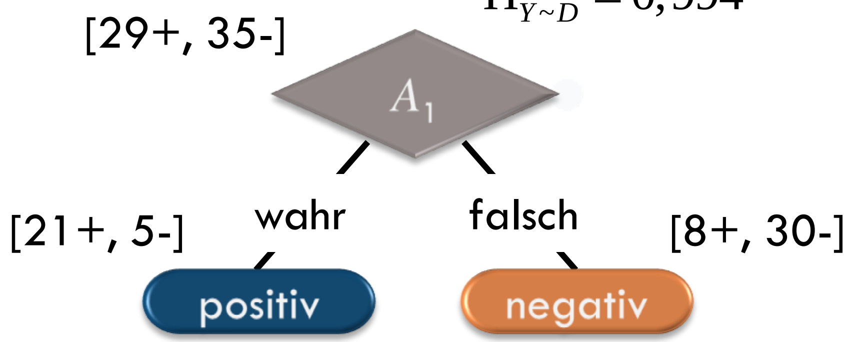
$$= -\frac{8}{38} \log \frac{8}{38} - \frac{30}{38} \log \frac{30}{38} = 0{,}742$$

Entscheidungsbäume

Lernen von Entscheidungsbäumen (nominale Attribute)

- Beispiel: Partitionierung der Daten *D* bzgl. binären Attributs A_1 in D_1 und D_2.

$$\mathrm{H}_{Y\sim D}=0,994$$



$$\mathrm{H}_{Y\sim D_1}=0,706 \qquad \mathrm{H}_{Y\sim D_2}=0,742$$

$$\mathrm{IG}_{Y,\{D_1,\ldots,D_k\}}=\mathrm{H}_{Y\sim D}-\mathrm{E}\left[\mathrm{H}_{Y\sim D_i}\right]=0,994-\left(\frac{26}{64}\cdot 0,706+\frac{38}{64}\cdot 0,742\right)=0,266$$

Entscheidungsbäume

Lernen von Entscheidungsbäumen (nominale Attribute)

- *Information Gain* abhängig von Anzahl & Größe der *k* Partitionen: $\mathrm{IG}_{Y,\{D_1,\ldots,D_k\}} = \mathrm{H}_{Y\sim D} - \mathrm{E}\left[\mathrm{H}_{Y\sim D_i}\right]$

- Beispiel: ID der *n* Beispiele als Attribut
  $\Rightarrow$ Jedes Beispiel ist eine Partition ($k = n$).
  $\Rightarrow \mathrm{E}\left[\mathrm{H}_{Y\sim D_i}\right] = 0 \Rightarrow \mathrm{IG}_{Y,\{D_1,\ldots,D_k\}}$ maximal.

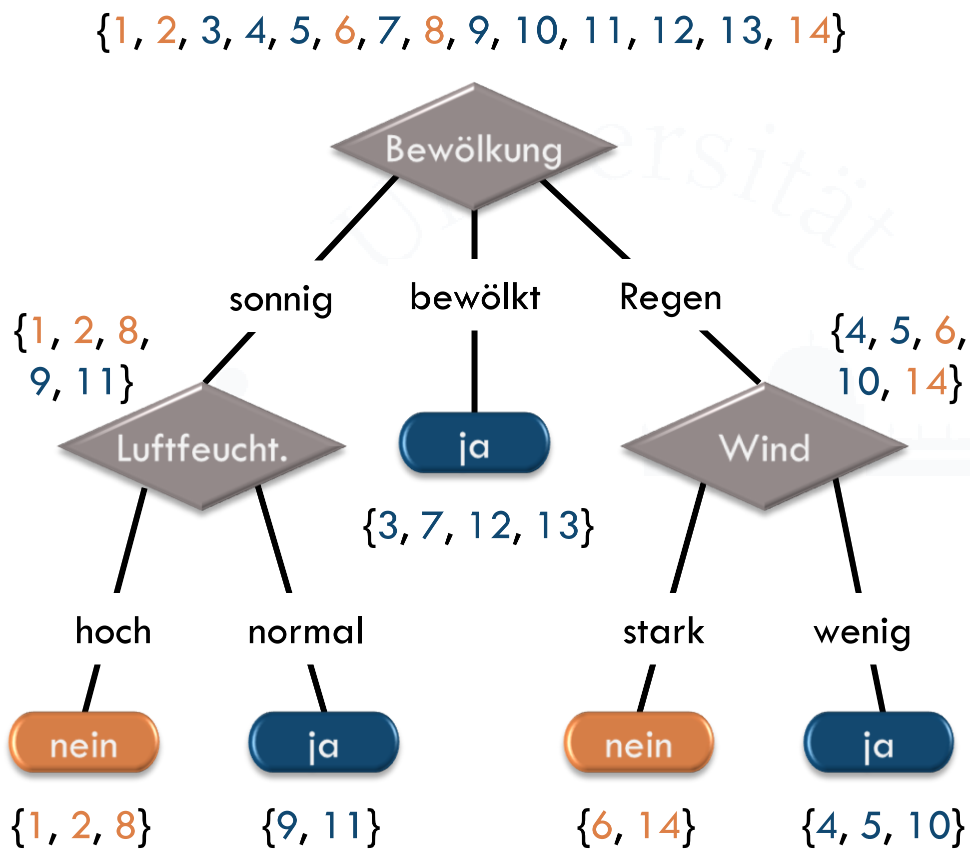
- Information *Gain Ratio* korrigiert diesen Bias:

$$\mathrm{GR}_{Y,\{D_1,\ldots,D_k\}} = \frac{\mathrm{IG}_{Y,\{D_1,\ldots,D_k\}}}{\mathrm{H}_{\{D_1,\ldots,D_k\}}} \qquad \mathrm{H}_{\{D_1,\ldots,D_k\}} = -\sum_{i=1}^{k} \frac{|D_i|}{|D|} \log \frac{|D_i|}{|D|}$$

Entscheidungsbäume

Lernen von Entscheidungsbäumen (nominale Attribute)

Tag	Bewölkung	Temp.	Luftfeucht.	Wind	Spielen?
1	sonnig	warm	hoch	wenig	nein
2	sonnig	warm	hoch	stark	nein
3	bedeckt	warm	hoch	wenig	ja
4	Regen	mild	hoch	wenig	ja
5	Regen	kühl	normal	wenig	ja
6	Regen	kühl	normal	stark	nein
7	bedeckt	kühl	normal	stark	ja
8	sonnig	mild	hoch	wenig	nein
9	sonnig	kühl	normal	wenig	ja
10	Regen	mild	normal	wenig	ja
11	sonnig	mild	normal	stark	ja
12	bedeckt	mild	hoch	stark	ja
13	bedeckt	warm	normal	wenig	ja
14	Regen	mild	hoch	stark	nein

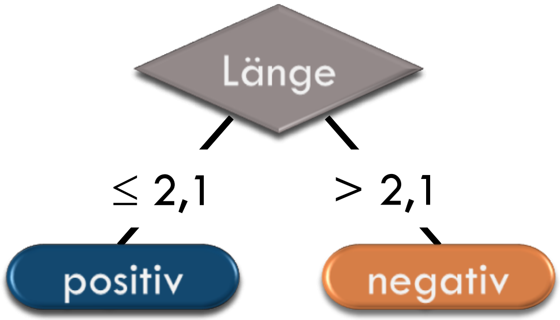


Entscheidungsbäume

Lernen von Entscheidungsbäumen (numerische & ordinale Attribute)

- Partitionierung bzgl. Attribut mit unendlichem Wertebereich?
- Idee: Endliche Menge an Instanzen
  $\Rightarrow$ Endliche Menge an binären Partitionierungen.
- Beispiel:

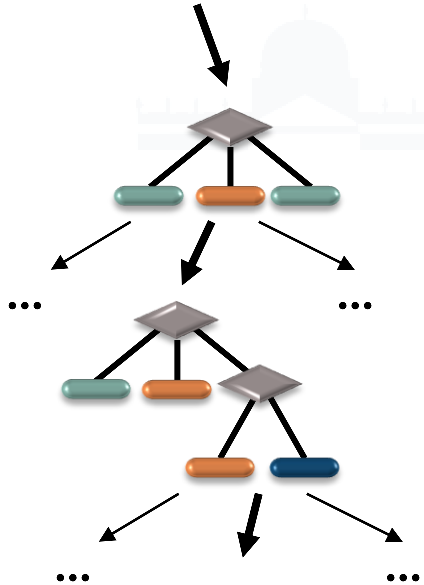
ID	Länge	…	Label
1	2,1	…	-
2	3,7	…	-
3	1,5	…	+



Entscheidungsbäume

Erweiterungen

- Pruning: Knoten nicht weiterentwickeln bzw. zusammen-fassen falls zu wenig Beispiele.
- Backtracking: Testattribut ändern falls Entropie in den Blättern zu hoch.
- Anderes Qualitätsmaß für Partitionierung, z.B. Gini-Index, Reduktion des MSE.
- Komplexere Partitionierungs-kriterien, z.B. Linearkombination von Attributen.



Entscheidungsbäume

Erweiterungen

- Modellbaum: komplexere Vorhersagemodelle (z.B. lineare Regression) in den Blättern.
- Random Forrest: Pool von Entscheidungsbäumen.
  - Lernen der Bäume auf zufällig ausgewählten, unterschiedlichen Instanz-/Attributteilmengen der Trainingsdaten.
  - Vorhersage nach dem Mehrheitsprinzip.
  - Nachteile:
    - Interpretierbarkeit geht (nahezu) verloren.
    - Modellkomplexität und Laufzeit zur Trainings- und Anwendungszeit erhöht sich.

Entscheidungsbäume
Algorithmen

- Klassifikation, d.h. Vorhersage nominaler Zielgröße:
  - Nominale Attribute: ID3
  - Numerische Attribute & Pruning: C4.5
  - Skalierbar für große Datenbanken: SLIQ, C5.0

- Regression, d.h. Vorhersage numerischer Zielgröße:
  - Regressionsbäume (konstante Werte in Blättern): CART
  - Modellbäume (Modelle in den Blättern): M5

Entscheidungsbäume

Vor- und Nachteile

- Vorteile:
  - Einfaches, leicht verständliches Verfahren.
  - Hohe Interpretierbarkeit der Ergebnisse.
- Nachteile:
  - Gefundene Lösung <u>lokal</u> aber i.A. <u>nicht global</u> optimal.
  - Generalisierungsfähigkeit oft geringer im Vergleich zu komplexeren Modellen (z.B. Kernel-Verfahren).
  - Instabile Lösung, d.h. leichte Veränderung der Daten kann starke Änderung des Modells/der Vorhersage bewirken.

Zusammenfassung

- Entscheidungsbäume geeignet für Klassifikations- & Regressionsprobleme.
  - Geeignet bei wenigen binären/nominalen Attributen.
- Lernen von Entscheidungsbäumen durch Greedy-Suche im Raum aller Entscheidungsbäume.
  - Partitionierungskriterium ($=$, $\leq$, $>$, …).
  - Qualitätsmaß für Partitionierung (IG, GR, Gini, MSE, …).
  - Zahlreiche Erweiterungen: Backtracking, Pruning, …