UNIVERSITÄT
LEIPZIG

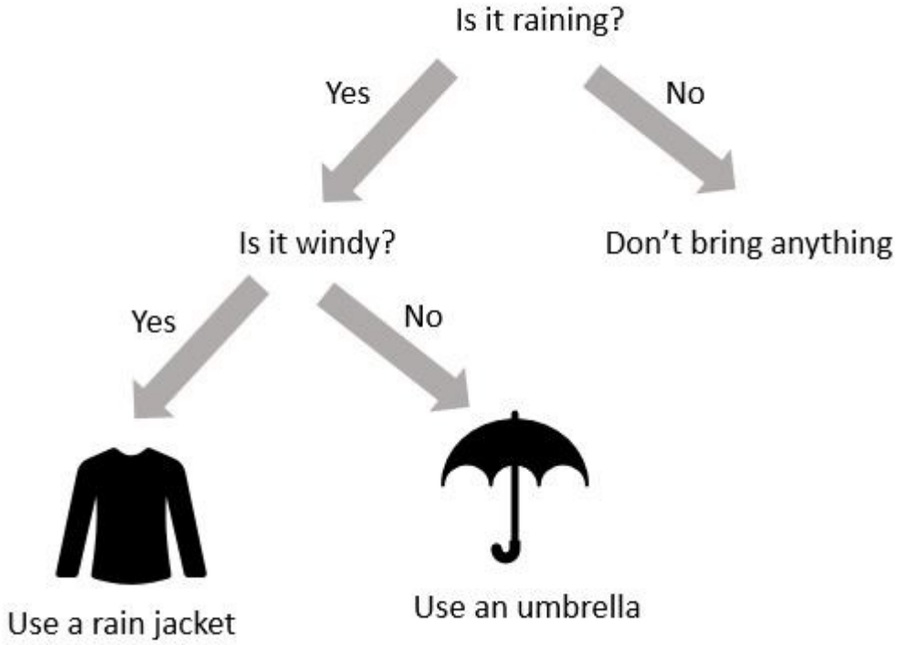
Data Mining

INTERACTIVE VISUAL DATA MINING

Decision Trees

Decision Trees

Decision Trees



Decision Trees

Was sind Entscheidungsbäume?

- Instanzen mit Attributen werden als Baum dargestellt
- Baum ist gerichtet
- Ordnung der “Aussagekraft”

- Baum t sagt die Werte c1, … , cn eines Attributes für die Instanz X voraus
  - t: X ist Element von {c1, … , cn}
- Regeln sind in Form von if … , then … aufgestellt
  - Pfade des Baums kodieren die Regeln

Decision Trees

Anwendungsbereiche

- Klassifikation und Regression
- Beispiele
  - Bewertung von Kreditwürdigkeit
  - Medizinische Diagnosen

- Bedingungen:
  - Instanzen werden durch Attribute-Werte Paare beschrieben
  - Attribute der Daten werden zugeordnet
  - Attribute müssen diskret sein

Decision Trees

Warum Entscheidungsbäume?

- Regeln sind einfach ablesbar
- Sind gut darstellbar
- Whitebox Verfahren
- Relativ schnell berechenbar

Decision Trees

Aussichten	Luftfeuchtigkeit	Wind	Entscheidung
Sonnig	Hoch	stark	nein
Regen	Hoch	stark	nein
Regen	normal	stark	nein
Sonnig	normal	stark	ja
Bedeckt	hoch	stark	ja
Bedeckt	normal	stark	ja
Sonnig	hoch	schwach	nein
Sonnig	normal	schwach	ja
Regen	hoch	schwach	ja
Regen	normal	schwach	ja
Bedeckt	hoch	schwach	ja
Bedeckt	normal	schwach	ja

Decision Trees

Aussichten	Luftfeuchtigkeit	Wind	Entscheidung
Bedeckt	hoch	stark	ja
Bedeckt	normal	stark	ja
Bedeckt	hoch	schwach	ja
Bedeckt	normal	schwach	ja
Regen	hoch	stark	nein
Regen	normal	stark	nein
Regen	hoch	schwach	ja
Regen	normal	schwach	ja
Sonnig	hoch	stark	nein
Sonnig	normal	stark	ja
Sonnig	hoch	schwach	nein
Sonnig	normal	schwach	ja

Decision Trees

Aussichten	Luftfeuchtigkeit	Wind	Entscheidung
Bedeckt	hoch	stark	ja
Bedeckt	normal	stark	ja
Bedeckt	hoch	schwach	ja
Bedeckt	normal	schwach	ja
Regen	hoch	stark	nein
Regen	normal	stark	nein
Regen	hoch	schwach	ja
Regen	normal	schwach	ja
Sonnig	hoch	stark	nein
Sonnig	normal	stark	ja
Sonnig	hoch	schwach	nein
Sonnig	normal	schwach	ja

Decision Trees

Aussichten	Luftfeuchtigkeit	Wind	Entscheidung
Bedeckt	hoch	stark	ja
Bedeckt	normal	stark	ja
Bedeckt	hoch	schwach	ja
Bedeckt	normal	schwach	ja
Regen	hoch	stark	nein
Regen	normal	stark	nein
Regen	hoch	schwach	ja
Regen	normal	schwach	ja
Sonnig	hoch	stark	nein
Sonnig	normal	stark	ja
Sonnig	hoch	schwach	nein
Sonnig	normal	schwach	ja

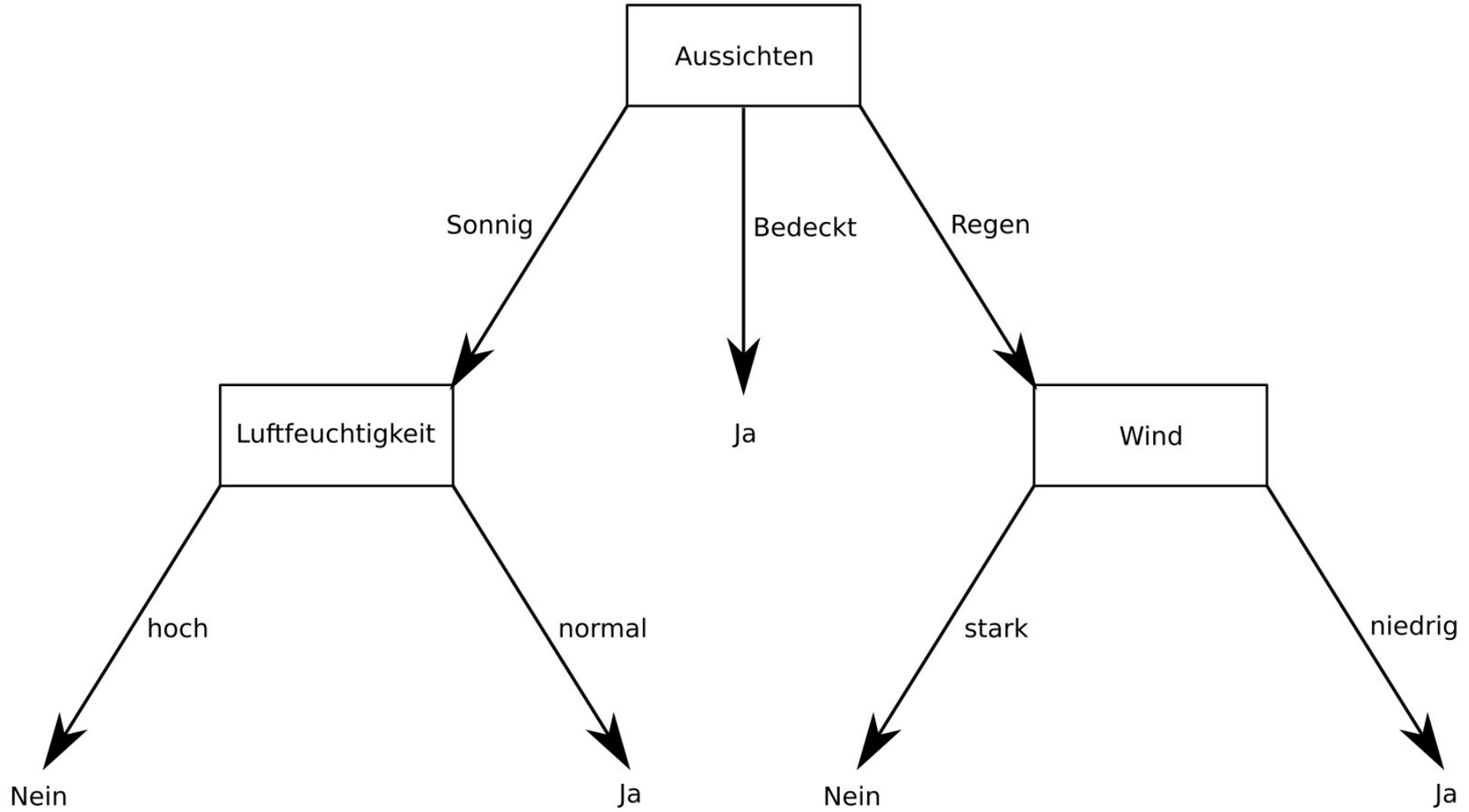
Decision Trees

Aussichten	Luftfeuchtigkeit	Wind	Entscheidung
Bedeckt	hoch	stark	ja
Bedeckt	normal	stark	ja
Bedeckt	hoch	schwach	ja
Bedeckt	normal	schwach	ja
Regen	hoch	stark	nein
Regen	normal	stark	nein
Regen	hoch	schwach	ja
Regen	normal	schwach	ja
Sonnig	hoch	stark	nein
Sonnig	normal	stark	ja
Sonnig	hoch	schwach	nein
Sonnig	normal	schwach	ja

Decision Trees

Aussichten	Luftfeuchtigkeit	Wind	Entscheidung
Bedeckt	hoch	stark	ja
Bedeckt	normal	stark	ja
Bedeckt	hoch	schwach	ja
Bedeckt	normal	schwach	ja
Regen	hoch	stark	nein
Regen	normal	stark	nein
Regen	hoch	schwach	ja
Regen	normal	schwach	ja
Sonnig	hoch	stark	nein
Sonnig	normal	stark	ja
Sonnig	hoch	schwach	nein
Sonnig	normal	schwach	ja

Decision Trees



Decision Trees

Herausforderungen

- Kann man die Reihenfolge der Attribute so wählen, dass der Baum möglichst klein wird?
  - Priorisierung
- Kontinuierliche Werte?
- Fehler und fehlende Werte?

Decision Trees

ID3

- Lernt den Baum von oben nach unten (top down)
- Wählt das beste Attribut zum spalten aus (greedy)
  - “Beste” im Sinne der Informationsentropie
- Rekursion auf den Kinderknoten

Pseudocode:

1. X $\sqcup$ das “beste” Attribut aus den Attributen
2. X = “Spalt” Attribut vom derzeitigen Knoten
3. Erstelle für alle Werte von X Kindsknoten mit den zugehörigen Werten
4. Gehe zu den Kindsknoten und starte bei 1., bis keine Attribute mehr übrig sind

Decision Trees

Probleme

- Bleibt oft in lokalen Optima stecken
- Nur diskrete Daten

- Overfitting
  - Wachsen bei bestimmter Tiefe stoppen
    - Wann?
  - Pruning des Baums
    - Post-pruning
    - Reduced-error pruning

Decision Trees

Bewertungsfunktion Entropie

- Entropie ist ein Maß über den Informationsgehalt in den Daten
- Boolsche Klassifikation
  - $Entropie(S) = -p_x log_2(p_x) - p_y log_2(p_y)$
- Für mehrere Klassen
  - $Entropie(S) = \sum_{i=1}^{c} -p_i log_2(p_i)$

Decision Trees

Information Gain

- Deskriptor, wie gut ein Merkmal die Menge klassifiziert
- Vergleicht Entropie vor und nach dem Split mit Attribut A
- Gibt Auskunft über das Sinken der Entropie nach dem Split
- $Gain\,(S,A) = Entropie\,(S) - \sum_{v \in Werte(A)} \frac{|S_v|}{|S|}\, Entropy(S_v)$
- S = Trainingsbeispiele
- A = gewähltes Attribut
- Werte(A) = Alle Werte die A annehmen kann
- S_v = Untermenge von S, in der alle Beispiele Attribut = v haben

Decision Trees

Am Beispiel

Tag	Aussichten	Temperatur	Luftfeuchtigkeit	Wind	Entscheidung
1	Sonne	heiß	hoch	schwach	nein
2	Sonne	heiß	hoch	hoch	nein
3	Bedeckt	heiß	hoch	schwach	ja
4	Regen	angenehm	hoch	schwach	ja
5	Regen	kalt	normal	schwach	ja
6	Regen	kalt	normal	hoch	nein
7	Bedeckt	kalt	normal	hoch	ja
8	Sonne	angenehm	hoch	schwach	nein
9	Sonne	kalt	normal	schwach	ja
10	Regen	angenehm	normal	schwach	ja
11	Sonne	angenehm	normal	hoch	ja
12	Bedeckt	angenehm	hoch	hoch	ja
13	Bedeckt	heiß	normal	schwach	ja
14	Regen	angenehm	hoch	hoch	nein

Decision Trees

- Entropie über den Datensatz:

- Entropy (S)= -9/14*log2(9/14) -5/14*log2(5/14) = 0.94

- Information Gain von Aussichten:

- Gain(S, Aussichten) = Entropie(S) - 5/14*Entropie(Sonne)
  - 4/14*Entropie(Regen)
  - 5/14*Entropie(Bedeckt)
  = 0.94 - 5/14*0.971 - 4/14*0 - 5/14*0,971

  Gain(S, Aussichten) = 0,246

Decision Trees

- Entropie über den Datensatz:

- Entropy (S)= -9/14*log2(9/14) -5/14*log2(5/14) = 0.94

  Gain(S, Aussichten) = 0,246

  Gain(S, Luftfeuchtigkeit) = 0,151

  Gain(S, Wind) = 0,048

  Gain(S, Temperatur) = 0,029

Decision Trees

- Ausblick ist eindeutig der beste Split
- Information Gain muss für Teilbäume neuberechnet werden

Tag	Temperatur	Luftfeuchtigkeit	Wind	Entscheidung
1	heiß	hoch	schwach	nein
2	heiß	hoch	hoch	nein
8	angenehm	hoch	schwach	nein
9	kalt	normal	schwach	ja
11	angenehm	normal	hoch	ja

Sonne

Tag	Temperatur	Luftfeuchtigkeit	Wind	Entscheidung
4	angenehm	hoch	schwach	ja
5	kalt	normal	schwach	ja
6	kalt	normal	hoch	nein
10	angenehm	normal	schwach	ja
14	angenehm	hoch	hoch	nein

Regen

Tag	Temperatur	Luftfeuchtigkeit	Wind	Entscheidung
3	heiß	hoch	schwach	ja
7	kalt	normal	hoch	ja
12	angenehm	hoch	hoch	ja
13	heiß	normal	schwach	ja

Decision Trees

- Ausblick ist eindeutig der beste Split
- Information Gain muss für Teilbäume neuberechnet werden

Tag	Temperatur	Luftfeuchtigkeit	Wind	Entscheidung
1	heiß	hoch	schwach	nein
2	heiß	hoch	hoch	nein
8	angenehm	hoch	schwach	nein
9	kalt	normal	schwach	ja
11	angenehm	normal	hoch	ja

Sonne

- Beispiel Teilbaum Sonne
  - Gain(Sonne, Luftfeuchtigkeit) = 0.97 -(⅗) * 0 - (⅖) * 0 = 0.97
  - Gain(Sonne, Temperatur) = 0.97 - (⅖) * 0 - (⅖) * 1 - (⅕) * 0 = 0.57
  - Gain(Sonne,Wind) = 0.97 - (⅖) * 1 - (⅗) * 0.981 = 0.019

Decision Trees

C4.5 Algorithmus

- Quinlan, 1993
- Erweiterung vom ID3
- Kann fehlende Attribute in Trainingsdaten verarbeiten
- Kontinuierliche Daten werden unterstützt
- Lösung für Overfittingprobleme

- Unbekannte Attributwerte:
  - Gewichtung und Wahrscheinlichkeiten
- Kontinuierliche Werte:
  - Alle Splits berechnen und den Besten wählen
- Post-Pruning
  - Ast durch Blatt ersetzen
  - Ast durch Teilast ersetzen

Decision Trees

Zusammenfassung

Pro:

- Regeln können einfach abgeleitet werden
- Robust beim Training
  - Klassifikationsfehler
  - Fehler in den Attributen
- Relativ einfach erweiterbar zu Random Forest

Kontra:

- Immenser Rechenaufwand bei kontinuierlichen Werten
- Baum kann sehr groß werden
  - Pruning
- Schnelles Overfitting