Moderne Methoden der KI: Maschinelles Lernen

Prof. Dr.Hans-Dieter Burkhard
Vorlesung Sommer-Semester 2007

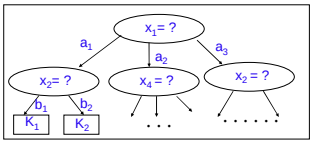
Entscheidungsbäume

Darstellung durch Regeln
ID3 / C4.5
Bevorzugung kleiner Hypothesen
Overfitting

Entscheidungsbäume

Entscheidungs-/Klassifizierungsverfahren
- Funktionen f über Merkmalsvektoren $[x_1,...,x_n]$
- Diskreter Wertebereich $\{K_1,...,K_k\}$ (Klassifikation, Entscheidung).

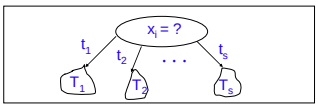
Entscheidungsbaum:
- Gerichteter Baum mit markierten Knoten:
  - Blätter enthalten Ergebnis (Klasse)
  - innere Knoten enthalten Tests für Merkmalswerte



Entscheidungsbäume

Rekursive Beschreibung:
- Blattknoten: $T = \{ label(T) \}$ (Beschriftung: Wert von f an diesem Knoten)
- Innere Knoten: $T= \{ test\_i , [t_1:T_1,...,t_s:T_s] \}$
  - Test für Merkmal i :
    - Verzweigung in Unterbaum T_r , falls x_i den Test t_r erfüllt

Verwendete Notation: $T= \{ test\_i , [t_1:T_1,...,t_s:T_s] \}$
- Baum mit Test des Merkmals M_i im Wurzelknoten
- und mit t_r beschrifteten Verzweigungen in Unterbäume T_r



Entscheidungsbäume

Entscheidungsverfahren:
- Für gegebenen Merkmalsvektor in jedem Knoten (Beginn: Wurzel) gemäß Test verzweigen. Erreichtes Blatt liefert Ergebnis.

Rekursive Beschreibung des Entscheidungsverfahrens:

1. Falls Blatt erreicht: Resultat ausgeben. Stop.
2. Für die Wurzel des aktuellen Baumes Test ausführen und gemäß Ergebnis verzweigen in den betreffenden Nachfolgerbaum. Weiter bei 1.

$decision([x_1,...,x_n], T) := label(T)$, falls $T = \{ label(T) \}$ (Blattknoten)

$decision([x_1,...,x_n], T) := decision(\{ [x_1,...,x_n], T_r \})$,
falls $T = \{ test\_i , [t_1:T_1,...,t_s:T_s] \}$ und x_i erfüllt t_r

Entscheidungsbäume und Regeln

Test t_i (z.B. $x_i = a_r$?) entspricht Constraint des i-ten Merkmals
Jeder Weg im Baum entspricht einer Konjunktion von Constraints
Klasse (bzw. Konzept) kann durch mehrere Blätter erkannt werden

Beschreibung einer Klasse durch Alternative von Konjunktionen
- d.h. beliebiger AK-Ausdruck über Tests bzw. Attribut-Wert-Paaren
- (Test kann auch Zugehörigkeit zu einem Intervall prüfen)

Repräsentation durch Regeln (leichter kommunizierbar: Data Mining)
- Logischer Ausdruck als Bedingung, Klasse als Ergebnis

Kompaktifizierung von Regeln: Geeignete Zusammenfassung gemäß AK

Konstruktion von Entscheidungsbäumen

Ausgangspunkt:
X : Menge von Objekten x mit Merkmalen $M_1,...,M_n$
- Objekte als Merkmalsvektoren $x = [x_1,...,x_n]$

K : Klasseneinteilung von X mit K(x) = Klasse von x

Prinzip für Verfahren zur Konstruktion eines Entscheidungsbaums T(D) aus einer Menge D von Trainingsbeispielen $d = [x,K(x)]$:

Falls $K(x) = K_i$ für alle $d = [x,K(x)] \in D$ gilt: (*als Schreibweise:* „ $D \subseteq K_i$ “)
$T(D) := \{K_i\}$ (Blattknoten mit Markierung K_i)

Sonst: **Auswahl** eines Testmerkmals M_i und von Tests $t_1,...,t_s$
$T(D) := \{ test\_i: [t_1:T(D_1),...,t_s:T(D_s)] \}$
mit $D_r := \{ [[x_1,...,x_n],K(x)] \in D \mid x_i$ erfüllt $t_r \}$

Konstruktion von Entscheidungsbäumen

Verfahren muss noch präzisiert werden.

Vorgaben bzgl.

- Abbruch bei inkonsistenter Trainingsmenge?
- Auswahlkriterien für Testmerkmal M_i und Tests $t_1,...,t_s$
  - Tests jeweils disjunkt und vollständig
    (genau einen Weg für jedes Beispiel x auswählen)
  - gleiches Merkmal evtl. mehrmals mit unterschiedlichen Tests
    (z.B. binäre Bäume) - hier nicht betrachtet
- Auswahl eines jeweils „besten“ Merkmals
  - führt auf Suchverfahren „Bergsteigen“
  - „bestes“ Merkmal im Sinne geringer Komplexität des Baumes

Konstruktion von Entscheidungsbäumen

Auswahl eines „*besten*“ Testmerkmals M_i

Informationstheoretischer Ansatz: Informationsgewinn (Entropie-Abnahme)

Entropie für Beispielmenge D

$$H(D) := \sum_{i=1}^{k} -p_i \log_2 p_i$$

wobei $p_i = p(K_i \mid D) := |D \cap K_i| : |D|$

Wahrscheinlichkeit von K_i relativ zu D,

d.h. Wahrscheinlichkeit für $d \in D$, zu K_i zu gehören.

Maß für Informationsgehalt/Unsicherheit:

Sicherheit: $H(D) = 0$.

Maximale Unsicherheit (Gleichverteilung): $H(D) = \log_2 |D|$.

Konstruktion von Entscheidungsbäumen

Informationsgewinn (bedingte Entropie)

bei Test test_i von M_i mit Ausgängen $t_1,...,t_s$ (disjunkt, vollständig)

$$H(D,test\_i) := H(D) - \sum_{r=1}^{s} p(D_r|D)\, H(D_r)$$

mit $D_r := \{ [[x_1,...,x_n],K(x)] \in D \mid x_i \text{ erfüllt } t_r \}$

und $p(D_r|D) = |D_r| : |D|$

Wahrscheinlichkeit von D_r relativ zu D,

d.h. Wahrscheinlichkeit für $d \in D$, zu D_r zu gehören.

M_i mit maximalem Gewinn auswählen.

Maximaler Gewinn für geringe $H(D_r)$ (geringe Unsicherheit in D_r).

Basis-Algorithmus ID3

ID3 („inductive learning of decision trees“, Quinlan1986),
Erweiterung: C4.5 (Quinlan1993),

für Merkmale M_i

mit endlichen Wertebereichen $[a^i_1,...,a^i_{s(i)}]$,

Auswahl gemäß Informationsgewinn

Konstruktion eines Entscheidungsbaumes ID3(D,M1) aus Trainings-Menge D unter Verwendung der Merkmale aus M1

Basis-Algorithmus ID3

Falls $D \subseteq K_i$: $ID3(D,M) := \{K_i\}$

Falls $M=\emptyset$: $ID3(D,M) := \{K_l\}$ mit K_l = häufigste Klasse in D

Sonst: Auswahl des besten Testmerkmals $M_i \in M$

$$ID3(D,M) := \{ test\_i , [t_1: T_1 ,..., t_{s(i)}: T_{s(i)}] \}$$

mit $T_r := ID3(D_r, M-\{M_i\})$,

falls $D_r := \{ [[x_1,...,x_n],K(x)] \in D \mid x_i = a^i_r \} \neq \emptyset$

$T_r := \{K_l\}$ mit K_l = häufigste Klasse in D , sonst

Eigenschaften von ID3

Suchverfahren „Bergsteigen“ im Raum aller(!) Entscheidungsbäume

Suche nach „effizientestem“ (kleinstem?) Baum

Problem lokaler Optimierungsschritte:
global optimale Lösung kann verfehlt werden

Suchrichtung: einfach → komplex, Entropie als Auswahlkriterium

Suche unter Betrachtung jeweils aller relevanten Beispiele

Suchraum vollständig: Entscheidungsbäume = diskrete Funktionen

Suchverfahren unvollständig:

Jeweils genau eine aktuelle Hypothese

kein Überblick über weitere konsistente Hypothesen

Ergebnis ergibt sich aus Reihenfolge gemäß Auswahlkriterium

Eigenschaften von ID3

Inductive bias:

Komplexes Auswahlkriterium bevorzugt (im wesentlichen)

kürzere Bäume

Attribute mit hohem Informationsgewinn nah an der Wurzel

ist Konsequenz des Suchverfahrens:

Bergsteigen = Präferenzen im Hypothesenraum

„Preference bias“: spezielle Hypothesen bevorzugen

Restriction bias vs. Preference bias

Restriction bias:

Beschränkung des Hypotheseraums, z.B.

konjunktiv definierte Konzepte

lineare Funktionen

Stetigkeit/Ähnlichkeit

Preference bias:

bestimmte Hypothesen bevorzugen

einfache Bäume

kurze Erklärungen

ähnlicheres Verhalten

Prinzip: Bevorzugung einfacher Hypothesen

Ockham´s Razor:

Pluralitas non est ponenda sine necessitate.

Wilhelm von Ockham, 1320

(Eine Vielfachheit ist ohne Notwendigkeit nicht zu setzen.)

Als philosophisches Ökonomie-Prinzip:

Einfache Erklärungen bevorzugen.

Prinzip: Bevorzugung einfacher Hypothesen

Ökonomie-Prinzip:

Einfache Erklärungen bevorzugen.

Beispiel: Modell des Sonnensystems

geringere Wahrscheinlichkeit für zufällige Übereinstimung bei geringerer Anzahl von Hypthesen:

weniger „kurze“ Bäume

einfache Beschreibung

Evolutions-Argument: einfache Hypothesen effizienter bzgl. interner Repräsentation des Lernenden

Overfitting

Gegeben Hypothesenraum H.

Fehler(h, X´) := Fehler der Hypthese h auf einer Menge X´ ⊆ X

Hypothese h∈H ist überangepasst („overfitting“), falls gilt:
∃ h´∈ H : Fehler(h,D) < Fehler(h´,D) ∧ Fehler(h,X) > Fehler(h´,X)

Ursachen:

Fehlerhafte (verrauschte) Trainingsdaten in D

Anpassung an sehr spezielle (korrekte) Beispiele in D

Zu langes Lernen ergibt zu starke Spezialisierung

Erfahrung: Overfitting als häufige Ursache von Fehlern im ML

Strategien gegen Overfitting

Ziel: Entscheidungsbaum mit angemessener Größe durch

Frühzeitiges Beenden der Erweiterung des Baumes

Nachträgliches Verkleinern des Baumes

Benötigt Kriterien für Abbruch bzw. für Verkleinerung

Möglichkeiten:

Beispielmenge aufteilen in

Trainingsmenge D (ca. 2/3 der verfügbaren Daten) und

Evaluierungsdaten E (ca. 1/3)

Annahme: Fehlerursache nicht gleichzeitig in D und E

Statistische Güte-Schätzungen anhand aller Beispieldaten

Minimale Beschreibungskomplexität bzgl. Beispieldaten

Strategien gegen Overfitting

Reduced Error pruning (Quinlan 1987):

Aufteilung der Beispielmenge in D und E

T/n := durch „Pruning“ im Knoten n aus T entstandener Baum
enthält anstelle des Teilbaumes mit der Wurzel n einen Blattknoten mit Bewertung K_i (gemäß der häufigsten Klassifizierung von mit n assoziierten Beispielen aus D)

Iteratives Verfahren
Für alle Knoten n:
Vergleichen von T und T/n durch Evaluierung auf Basis E
Für Knoten n mit maximaler Verbesserung wird T durch T/n ersetzt

Iterieren, solange eine Verbesserung möglich ist.

Strategien gegen Overfitting

Rule Post-Pruning (verwandte Methode in C4.5, Quinlan 1993)

Transformation des Baumes in Regelmenge (je Pfad eine Regel)
Für jede Regel:
Pruning von Vorbedingungen, soweit sich Verbesserung ergibt
(erwartete Genauigkeiten mit/ohne Vorbedingung vergleichen)
Anordnung der Regeln gemäß erwarteter Genauigkeit
(bei Klassifizierung dann in dieser Reihenfolge abarbeiten)

Evaluierung anhand einer Evaluierungsmenge E oder mittels statistischer Verfahren für erwartete Genauigkeit (C4.5)

Übergang zu Regelmenge vorteilhaft
(Auflösung der Baum-Strukturen: Abhängigkeiten, Anordnung)

Alternative Auswahlkriterien für Merkmale

Informationsgewinn H(D,test_i)
favorisiert Merkmale M_i mit vielen Werten
Problematisch bei Merkmalen, die für jedes $x \in X$ unterschiedlichen Wert ergeben ($M_i(x) = M_i(x') \rightarrow x = x'$)
$\Rightarrow$ keine Generalisierungsmöglichkeit

Alternative Maße: siehe Literatur

Einbeziehung von Kosten für Attribute:
modifiziertes Maß: $H^2(D,test\_i) : Kosten(M_i)$

Weitere Probleme

Behandlung kontinuierlicher Merkmalswerte:
Definition neuer diskreter Merkmale
z.B. $a_r < x_i < b_r$, $r = 1,...,s$
Optimal: Wahl der Grenzen a_r , b_r in der Mitte zwischen zwei Klassengrenzen der Beispiele aus D

Weitere Probleme

Behandlung fehlender Werte („missing values“)
Annahmen über Verteilung möglicher Werte anhand der Traingsmenge D
Benutzung für Berechnung des Informationsgewinns
Benutzung für Klassifizierung neuer Objekte
(Verfahren in C4.5)

Zusammenfassung

Entscheidungsbäume sind ein effizientes Klassifzierungsverfahren (z.B. für Data Mining).
Alternativ kann eine Regelmenge erzeugt werden.
Das Lernverfahren verwendet vollständige Hypothesenräume und ein unvollständiges Suchverfahren (prediction bias).
Die Suche erfolgt von einfachen zu komplexeren Hypothesen.
Verwendet wird ein Bergsteigen-Suchverfahren, optimiert wird bei ID3 bezüglich Einfachheit und Informationsgewinn. Die Modifikation des Auswahlkriteriums ist durch alternative Maße möglich.
Das Overfitting-Problem kann durch Pruning gemildert werden.
Fehler (Rauschen) in der Trainingsmenge werden toleriert.
Fehlende Werte können geschätzt werden.
ID3 ist in zahlreichen Varianten erweitert/modifiziert worden (insbesondere: C4.5).