Decision Tree Learning

Caroline Sporleder

Computational Linguistics
Universität des Saarlandes

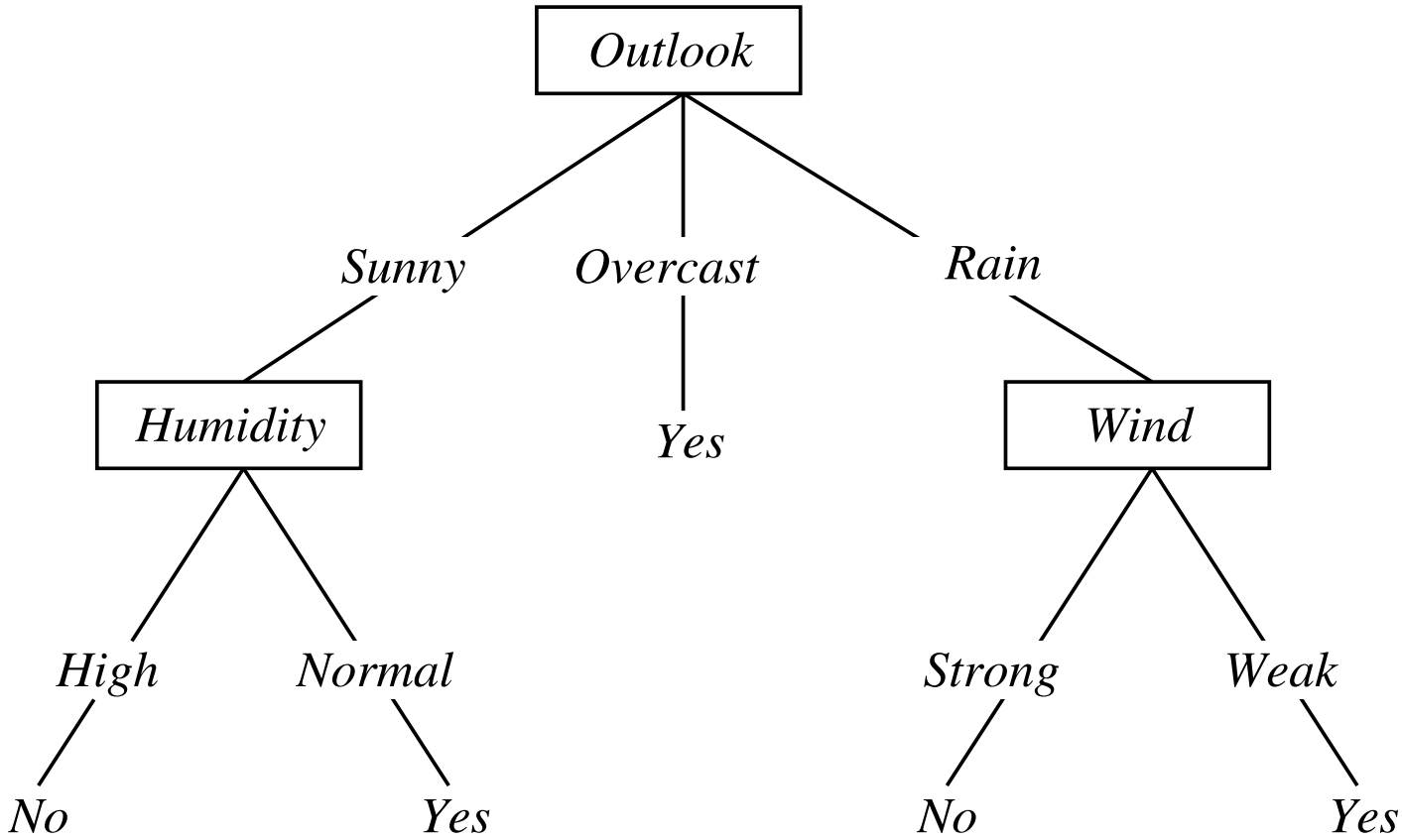
Sommersemester 2011

28.04.2011

Entscheidungsbäume

Repräsentation von Regeln als Entscheidungsbaum (1)

Wann spielt Max Tennis?



Repräsentation von Regeln als Entscheidungsbaum (2)

Entscheidungsbäume repräsentieren Disjunktionen von Konjunktionen von Attribut-Wert-Paaren.

Zum Beispiel:

$Outlook = overcast \vee (Outlook = sunny \wedge Humidity = normal) \vee (Outlook = rain \wedge Wind = weak)$

Entscheidungsbäume

Wann Entscheidungsbäume?

- Instanzen können durch Attribut-Wert-Paare repräsentiert werden
- Targetfunktion ist diskret
- disjunktive Hypothesen sind notwendig
- fehlerhafte oder inkonsistente Trainingsdaten (Noise)

Lernen von Entscheidungsbäumen

Entscheidungsbäume Lernen

Was wäre ein guter Entscheidungsbaum für die folgenden Daten?

Num	Doz	The	Wo	We	Max da?
1	B	A	D	S	N
2	A	D	F	R	N
3	B	D	D	R	J
4	A	A	F	S	N
5	B	A	D	R	J

Einfacher Algorithmus

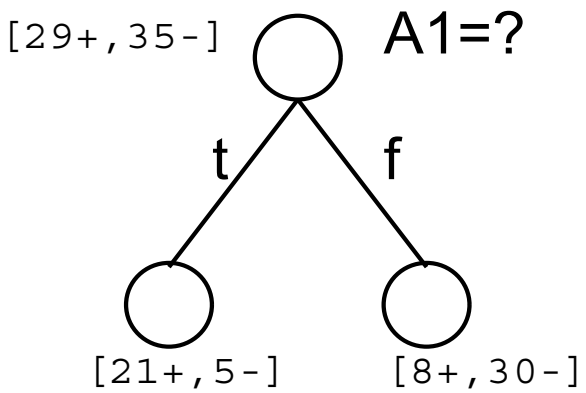
1. wähle ein Attribut A für den nächsten Knoten n
2. für jeden Wert von A erstelle einen Tochterknoten
3. teile die Trainingsbeispiele unter den Tochterknoten auf (je nachdem, welchen Wert sie für A haben)
4. wenn positive und negative Beispiele perfekt separiert sind STOP, ansonsten iteriere über Tochterknoten

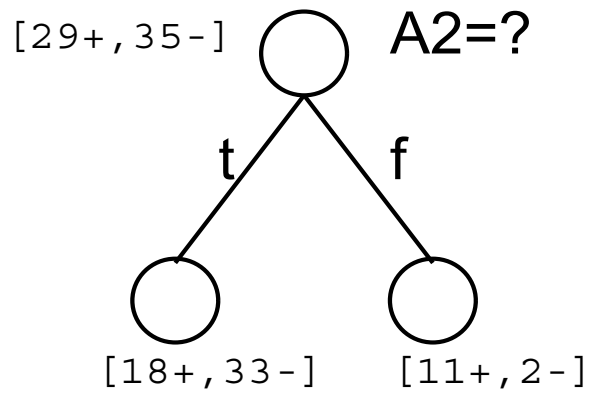
Einfacher Algorithmus

1. wähle ein Attribut A für den nächsten Knoten n
2. für jeden Wert von A erstelle einen Tochterknoten
3. teile die Trainingsbeispiele unter den Tochterknoten auf (je nachdem, welchen Wert sie für A haben)
4. wenn positive und negative Beispiele perfekt separiert sind STOP, ansonsten iteriere über Tochterknoten

Welches Attribut ist das beste?

Welches Attribut ist das beste? (1)



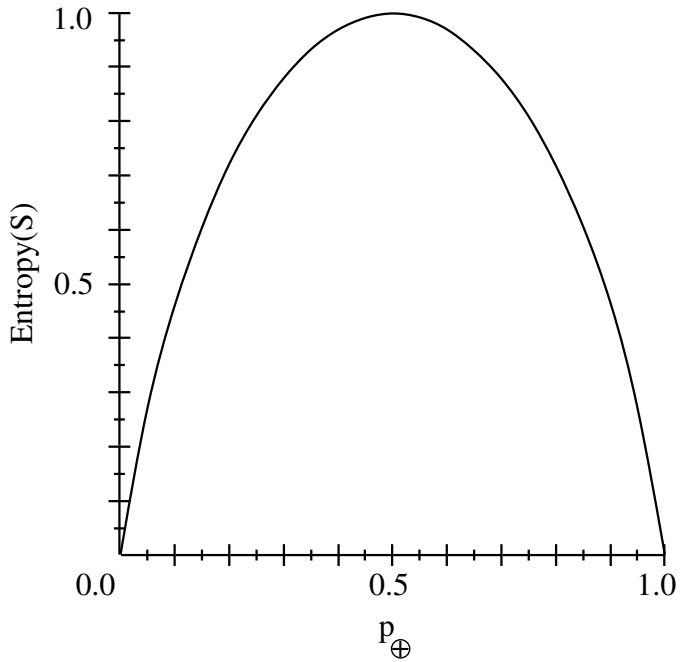


Welches Attribut ist das beste? (2)

- ein Attribut mit n Werten teilt die Menge der Instanzen S in eine Partition aus n Teilmengen S_1 bis S_n
- ideal ist ein Partition, deren Teilmengen möglichst 'rein' sind, d.h. möglichst viele Instanzen einer Ausgabeklasse (z.B., positiv oder negativ)
- je 'reiner' die Partition, die durch ein Attribut entsteht, desto früher sollte das Attribut ausgewählt werden

$\Rightarrow$ Wir brauchen ein 'Reinheitsmaß' für Mengen von Instanzen

Entropie (1)



- S ist eine Menge von Trainingsinstanzen
- $p_\oplus$ ist der Anteil der positiven Beispiele in S
- $p_\ominus$ ist der Anteil der negativen Beispiele in S
- Entropy mißt die 'Verunreinigung' (impurity) von S

$$Entropy(S) \equiv -p_\oplus \log_2 p_\oplus - p_\ominus \log_2 p_\ominus$$

Entropie (2)

Entropie(S) = erwartete Anzahl von Bits, die benötigt wird um die Ausgabeklasse ($\oplus$ oder $\ominus$) einer zufällig gewählten Instanz aus S zu kodieren (mit optimaler, kürzester Kodierung)

Warum?

Informationstheorie: der Code mit optimaler Länge benötigt $-\log_2 p$ Bits für Nachrichten der Wahrscheinlichkeit p.

D.h. die erwartete Anzahl von Bits, die benötigt werden um $\oplus$ oder $\ominus$ zufällig gewählte Instanzen aus S zu kodieren ist:

$$p_{\oplus}(-\log_2 p_{\oplus}) + p_{\ominus}(-\log_2 p_{\ominus})$$

$$Entropy(S) \equiv -p_{\oplus}\log_2 p_{\oplus} - p_{\ominus}\log_2 p_{\ominus}$$

Information Gain (IG) (1)

$Gain(S, A)$ = erwartete Reduktion in Entropie, wenn S anhand von A geteilt wird.

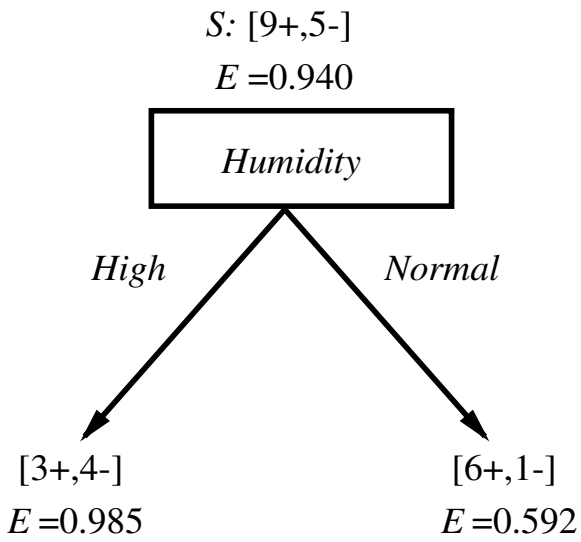
$$Gain(S, A) \equiv Entropy(S) \; - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v)$$

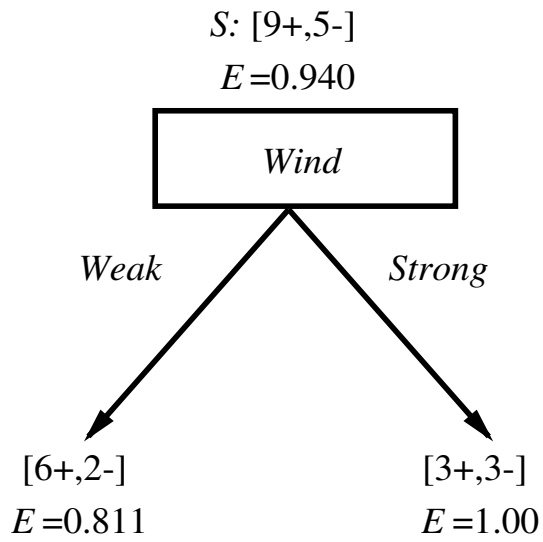
Beispiel (1)

Trainingsinstanzen

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Beispiel (2)





Gain (S, Humidity)

= .940 - (7/14).985 - (7/14).592
= .151

Gain (S, Wind)

= .940 - (8/14).811 - (6/14)1.0
= .048

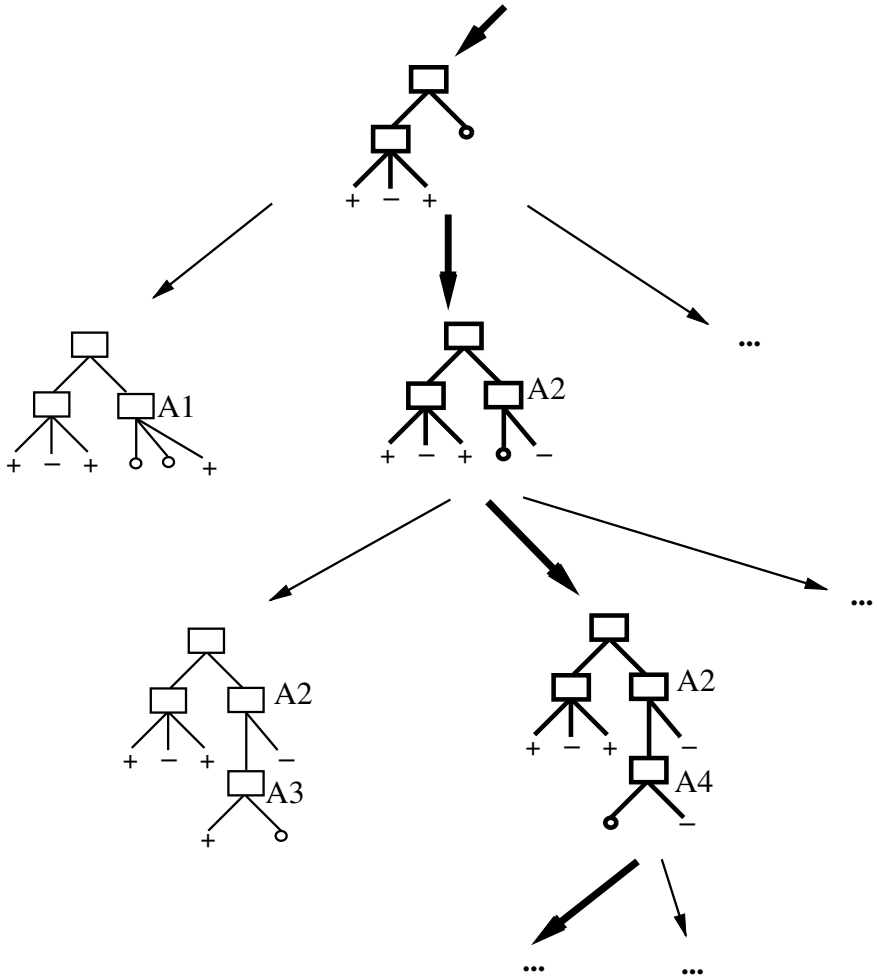
Hypothesenraum und induktiver Bias

Hypothesenraum

Was ist der Hypothesenraum?

Hypothesenraum

Was ist der Hypothesenraum?



Suche im Hypothesenraum

- Hypothesenraum ist vollständig
- der Algorithmus gibt genau eine Hypothese zurück
- Suche ist ‘greedy’
- Attributauswahl ist statistisch (robust gegenüber verrauschten Daten)

Induktiver Bias

Hypothesenraum, H, ist die Potenzmenge der Instanzen in X

Was ist der induktive Bias?

Induktiver Bias

Hypothesenraum, H, ist die Potenzmenge der Instanzen in X

Was ist der induktive Bias?

- Präferenz für kurze Bäume, mit Attributen, die einen hohen IG haben, in der Nähe der Wurzel
- Bias ist eine Hypothesen-**Präferenz**, keine **Restriktion** des Hypothesenraums
- **Occam's razor**: Präferenz für die kürzeste Hypothese, die die Daten erklärt

Occam's Razor

Warum eine Präferenz für kurze Hypothesen?

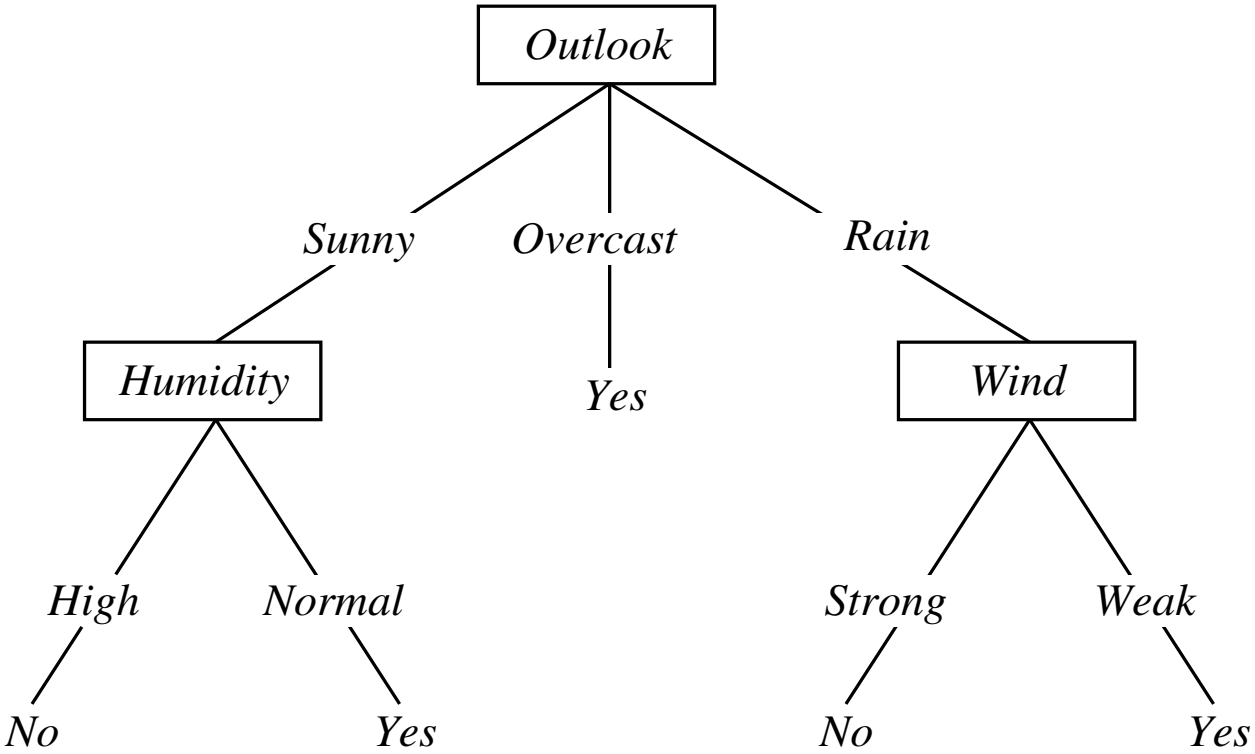
- es gibt weniger kurze als lange Hypothesen
  $\longrightarrow$ es ist unwahrscheinlich, dass eine kurze Hypothese **zufällig** die Daten erklärt
  $\longrightarrow$ eine lange Hypothese, die die Daten erkärt, kann dagegen Zufall sein

Overfitting

Overfitting (1)

Was passiert, wenn ein verrauschtes Beispiel dazukommt?

$< Sunny, Hot, Normal, Strong, No >$



Overfitting (2)

Der Fehler der Hypothese h über:

- Trainingsdaten: $error_{train}(h)$
- die gesamte Distribution $\mathcal{D}$ der Daten: $error_{\mathcal{D}}(h)$

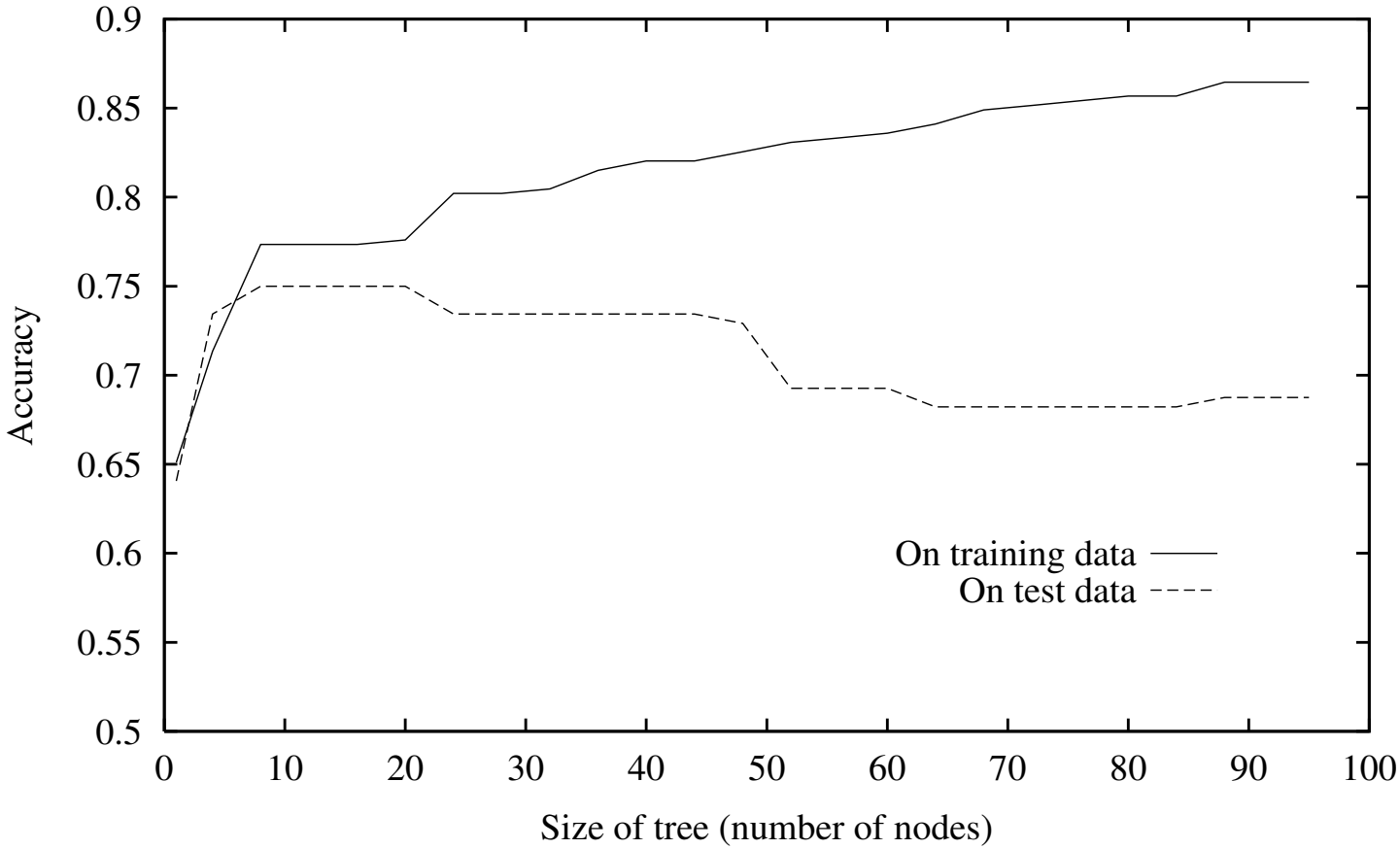
Hypothese $h \in H$ **overfits** die Trainingsdaten, wenn es eine alternative Hypothese $h' \in H$ gibt, so dass

$$error_{train}(h) < error_{train}(h')$$

und

$$error_{\mathcal{D}}(h) > error_{\mathcal{D}}(h')$$

Overfitting (3)



Overfitting (4)

Wie kann Overfitting vermieden werden?

Overfitting (4)

Wie kann Overfitting vermieden werden?

- Stoppe Attributselektion, wenn der Datensplit nicht mehr signifikant ist
- Baue vollständigen Baum, dann schneide zurück (**pruning**)

Overfitting (4)

Wie kann Overfitting vermieden werden?

- Stoppe Attributselektion, wenn der Datensplit nicht mehr signifikant ist
- Baue vollständigen Baum, dann schneide zurück (**pruning**)

Wie wird der 'beste' Baum ausgewählt?

- Messe die Performanz des Baums auf den Trainingsdaten
- Messe die Performanz auf einem separaten Validierungsset
- Minimum Description Length (MDL): minimize
  $size(tree) + size(misclassifications(tree))$

Reduced Error Pruning

Teile die Daten in *Trainings*− (T) und *Validierungsset* (V)

Wiederhole solange die Performanz nicht sinkt:

1. Für jeden Knoten, n, messe den Effekt den das prunen von n auf V hätte
2. Lösche den Knoten, n, der die Performanz auf V am stärksten erhöht.

Exkurs: Precision und Recall

Evaluationsmaße

Fehlertypen

	korrekte Ausgabeklasse	
vorhergesagte Ausgabeklasse	+	-
+	true positive (TP)	false positive (FP)
-	false negative (FN)	true negative (TN)

Performanzmaße

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}$$

$$\text{Precision} = \frac{TP}{TP+FP}$$

$$\text{Recall} = \frac{TP}{TP+FN}$$

$$\text{F-Score} = \frac{2*Prec*Rec}{Prec+Rec}$$

Zusammenfassung

Zusammenfassung

Was ihr gelernt haben solltet:

- Was ist ein Entscheidungsbaum und wie können durch Entscheidungsbäume Regeln repräsentiert werden?
- Wie können Entscheidungsbäume gelernt werden?
- Unter welchen Umständen ist Entscheidungsbaumlernen sinnvoll (vgl. mit Candidate Elimination)?
- Was ist der Hypothesenraum?
- Was ist der induktive Bias?
- Was ist Overfitting und wie kann es vermieden werden?