Ein Beispiel für maschinelles Lernen

von Ulla Mentel

1 Überblick:

1. Einleitung

2. Informelle Beschreibung

3. Algorithmus

4. Bewertungsregeln

5. pruning (Reduzierung), warum?

6. Und wann?

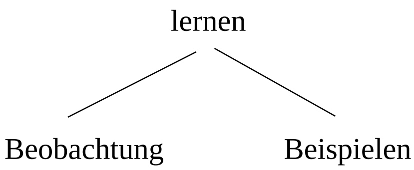
7. Error- based- pruning

1.1 Quellen:

"Einführung in die künstliche Intelligenz" (Skript) von Katharina Morik
"C 4. 5, programs for machine learning" von J. Ross Quinlan

2 Wir reden über intelligente Agenten

Was ist Intelligenz ??!
Ein Merkmal von Intelligenz ist Lernfähigkeit. Also sollen unsere Agenten lernfähig sein.



- gegeben: eine Menge von Beispielen, alle Informationen sind als Terme, bestehend aus einer Sammlung von Eigenschaften und Attributen, dargestellt

- gesucht: eine Klassifizierung der Beispielmenge (Trainingsmenge), um ein neues Beispiel einordnen zu können.

2.1 verschiedene Methoden zur Klassifizierung

Wir wollen uns hier mit aussagelogischen Modellen beschäftigen, also dem Begriffslernen aus Beispielen. Begriffslernen ist eine klassische Aufgabe des maschinellen Lernens. Der Ansatz ist ein "top- down" Lernen aus Beispielen mit statistischer Merkmalselektion.
Programme, die Entscheidungsbäume konstruieren

- an den Kanten stehen Attributwerte

- Knoten sind Verzweigungspunkte

- Blätter stellen Begriffsnamen (Klassen) dar

für ein neues Beispiel soll entschieden werden, zu welchem Begriff (welcher Klasse) es gehört:
folge den Kanten, deren Beschriftung dem Attributwert des Beispiels entspricht, bis ein Blatt erreicht ist. Das Beispiel wird der Klasse zugeordnet, die an dem Blatt angegeben ist.

Die bekanntesten Realisierenden sind C 4.5 , wie auch ID3, und haben ihren Ursprung in Hunts Concept Learning Systems.

Die im Algorithmus erwähnten Aufteilungstests werden im Anschluss eingehend behandelt.

3 Divide-and-conquer Algorithmus (Hunt, 50-er Jahre)

Menge von Beispielen T, Klassen $\{C_1, C_2, ..., C_n\}$, es gibt 3 Möglichkeiten:

- T beinhaltet eine oder mehrere Fälle (Beispiele) und alle gehören zu einer einzelnen Klasse C

- T ist leer. Der Entscheidungsbaum ist dann ein Blatt, aber die Klasse,die mit diesem Blatt assoziiert wird, muß abgegrenzt sein gegenüber der Information anderer Mengen als T. [C4.5 benutzt die meisten frequentierte Klasse am Elternknoten dieses Knotens]

- T beinhaltet Fälle, die zu einer Mischung von Klassen gehört: T wird zu Untermengen verfeinert. Ein Test "Test$X$" wird ausgewählt, basierend auf einem einzelnen Attributs, das ein oder mehrere, sich gegenseitig ausschließende, Ergebnisse $\{O_1, O_2, .., O_n\}$ hat. T wird aufgeteilt in die Unterklassen $T_1, T_2, .., T_n$, wobei T_i alle Fälle aus T beinhaltet, die das Ergebnis O_i bzgl. des ausgewählten Tests haben. Der Entscheidungsbaum besteht aus einem, den Test identifizierenden Entscheidungsknoten und einem Zweig für jedes mögliche Ergebnis.
  Der selbe Baum-Bildungs-Mechanismus wird rekursiv aufgerufen.

4 Bewertungsregeln

Forderung:
Der Baum soll aussagekräftig und möglichst klein sein.

Naiver Ansatz:
Alle möglichen Bäume werden erzeugt und der minimale wird auswählt.

aber
Hyafil und Rivest [1976] haben gezeigt, dass das Problem den kleinsten Entscheidungsbaum zu finden NP- vollständig ist.

In unserem kleinen Beispiel gibt es schon 4* 10 mögliche Entscheidungsbäume.

deshalb
ist die Auswahl des Tests entscheidend

die meisten Methoden, Entscheidungsbäume zu konstruieren, sind "nonbacktracking" und " greedy" Algorithmen

4.1 (Informations-)Gewinn- Kriterium [ID3]

Wie ermittelt man das Attribut mit dem größten Informationsgewinn für die Klassifizierung eines Objekts?

Notation:
für eine Liste S von Beispielen (Trainingsfällen) bedeutet
freq (C_i, S) := Anzahl der Beispiele aus S, die zur Klasse C_i gehören
$|S|$:= Anzahl der Beispiele in S

Stichprobe aus S und Annahme, daß die Stichprobe zur Klasse C_i gehört, hat die Wahrscheinlichkeit

$$\frac{freq(C_i, S)}{|S|}$$

der Informationsgehalt eines Attributwerts ist abhängig von seiner Wahrscheinlichkeit und berechnet sich durch

$$-log_2 \frac{freq(C_i, S)}{|S|}$$

Um den Informationsgehalt eines Attributs zu ermitteln, summiert man über die Klassen im Verhältnis zu ihrer Häufigkeit in S (*Entropie der Menge S*)

$$info(S) = -\sum_{i=1}^{n} \frac{freq(C_i, S)}{|S|} * log_2 \frac{freq(C_i, S)}{|S|}$$

angewendet auf die Menge der Trainingsfälle T, mißt info(T) den Durchschnitt von Informationen, die benötigt werden um die Klasse eines Falls aus T zu identifizieren

T wird aufgeteilt gemäß den n Ergebnissen von Test$X$. Der Informationsgehalt ist dann die gewichtete Summe über die Untermengen T_i

$$info_x(T) = \sum_{i=1}^{n} \frac{|T_i|}{|T|} * info(T_i)$$

der Informationsgewinn durch die Teilung von T gemäß Test$X$:

$$gain(X) = info(T) - info_x(T)$$

und soll maximal sein

danach wird das Attribut aus der Liste entfernt und der Alogrithmus rekursiv wieder aufgerufen

Nachteil:
das Gewinn- Kriterium favorisiert Tests mit vielen Ergebnissen, das kann zu unsinnigen Anwendungen führen

4.2 Beispiel

Aussicht	TempC	Luftfeucht.%	windig?	Klasse
sonnig	24	70	ja	Spiel
sonnig	27	90	ja	kein Spiel
sonnig	30	85	nein	kein Spiel
sonnig	21	95	nein	keinSpiel
sonnig	18	70	nein	Spiel
bewölkt	21	90	ja	Spiel
bewölkt	29	78	nein	Spiel
bewölkt	15	65	ja	Spiel
bewölkt	28	75	nein	Spiel
regnet	20	80	ja	kein Spiel
regnet	16	70	ja	kein Spiel
regnet	24	80	nein	Spiel
regnet	18	80	nein	Spiel
regnet	19	96	nein	Spiel

es gibt 2 Klassen, **Spiel** und **kein Spiel**, 9 Fälle gehören zu **Spiel** und 5 zu **kein Spiel**

$$info(T) = -\frac{9}{14} * log_2\left(\frac{9}{14}\right) - \frac{5}{14} * log_2\left(\frac{5}{14}\right) = 0.940$$

wählt man das Attribut **Aussicht**, dann ist

$$\begin{aligned} info_x(T) = & \frac{5}{14} * \left(-\frac{2}{5} * log_2\left(\frac{2}{5}\right) - \frac{3}{5} * log_2\left(\frac{3}{5}\right)\right) \\ & +\frac{4}{14} * \left(-\frac{4}{4} * log_2\left(\frac{4}{4}\right) - \frac{0}{4} * log_2\left(\frac{0}{4}\right)\right) \\ & +\frac{5}{14} * \left(-\frac{3}{5} * log_2\left(\frac{3}{5}\right) - \frac{2}{5} * log_2\left(\frac{2}{5}\right)\right) \quad = \quad 0.694 \end{aligned}$$

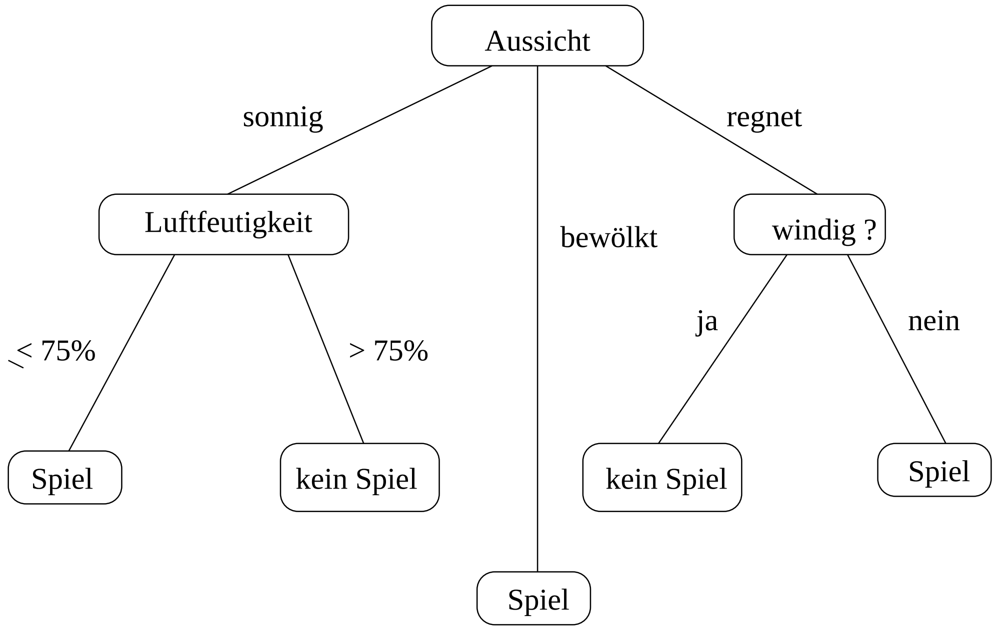
$$grain(X) = 0.940 - 0.694 = 0.246$$

würde man das Attribut **windig** wählen, dann müßte man zwei Untermengen untersuchen

$$\begin{aligned} info_x(T) = & \frac{6}{14} * \left(-\frac{3}{6} * log_2\left(\frac{3}{5}\right) - \frac{3}{6} * log_2\left(\frac{3}{6}\right)\right) \\ & +\frac{8}{14} * \left(-\frac{6}{8} * log_2\left(\frac{6}{8}\right) - \frac{2}{8} * log_2\left(\frac{2}{8}\right)\right) \\ & = \quad 0.892 \end{aligned}$$

$$grain(X) = 0.940 - 0.892 = 0.048$$

der reslultierende Baum bei der Wahl des Attributs **Aussicht**:



4.3 Gewinn- Zuweisungs- Kriterium [C 4.5]

Eine Art Normalisierung wird eingeführt, durch die der vermeintliche Gewinn zurechtgerückt wird. Betrachte man den Informationsgewinn eines Attributs einen Fall betreffend, läßt sich nicht die Klasse zu der der Fall gehört, sondern das Ergebnis des Tests erkennen.

Analog zu info (S) definieren wir

$$split.info(X) = -\sum_{i=1}^{n} \frac{|T_i|}{|T|} * log_2 \frac{|T_i|}{|T|}$$

Dies repräsentiert den Informationsgehalt erzeugt durch die Teilung gemäß T in n Untermengen. Der Informationsgewinn wird durch die, für die Klassifizierung relevante Information bemessen, die sich aus derselben Teilung ergibt. So ist

$$gaininfo(X) = \frac{gain(X)}{splitinfo(X)}$$

Wenn die Teilung fast trivial ist, wird die split- Information klein und die Zuweisung unstabil sein.

Um dies zu vermeiden, wählt das Gewinn- Zuweisungs- Kriterium einen Test aus, der die Zuweisung maximiert.

Grund für die Einschränkung ist, daß der Informationsgewinn groß sein soll - mindestens so groß wie der durchschnittliche Gewinn über alle Testergebnisse.

zurück zu unserem Beispiel 4.2 der Test auf Aussicht produziert 3 Untermengen mit 5, bzw 4, bzw 5 Fällen. Die split- info berechnet sich als

$$-\frac{5}{14} * log_2\left(\frac{5}{14}\right) - \frac{4}{14} * log_2\left(\frac{4}{14}\right) - \frac{5}{15} * log_2\left(\frac{5}{14}\right) = 1.577$$

der Gewinn ist nach wie vor 0.246

die Gewinn- Zuweisung ist somit $0.246 : 1.577 = 0.156$

Das Gewinn- Zuweisungs- Kriterium ist robust und ergibt eine bessere Testauswahl als das Gewinn- Kriterium (sagt Quinlan).

5 Pruning (Reduzieren) von Entscheidungsbäumen, warum?

Die rekursive Aufteilungsmethode führt zu sehr komplexen Bäumen, weil die Daten " überbewertet" werden. Es wird mehr Struktur gefolgert, als die Trainingsmenge rechtfertigt.

Ein Beispiel:
Es wird eine Menge von Zufallsdaten generiert, bei denen die Klassen von jedem Fall recht zusammenhangslos zu ihren Attributwerten sind.

- gegeben: künstliche Datenmenge mit 10 Attributen, jedes kann mit der gleichen Wahrscheinlichkeit den Wert 0 oder 1 annehmen 1000 zufällig generierte Fälle werden aufgeteilt in 500 Trainingsflle und 500 Testfälle

- Ergebnis: C 4.5 erzeugt einen Baum mit 119 Knoten. Die Fehlerrate bei de Testfällen liegt über 35%.

Die Beschäftigung mit Zufallsklassifizierern mag realitätsfremd erscheinen. Ist sie aber nicht, denn Beispiele (Fälle) sind häufig unklar, weil die Attribute nicht alle Informationen aufbringen, die zur Klassifizierung relevant sind
Also hofft man, dass einfachere Bäume auch eine geringere Fehlerrate haben.

6 Wann soll vereinfacht werden?

Zwei Mglichkeiten:

1. *stopping* oder auch *prepruning*
   die beste Art der Aufteilung zu suchen und unter dem Blickwinkel der statistischen Bedeutung, des Informationsgewinns, der Fehlerreduzierung usw. beurteilt. Ein Schwellwert wird definiert, fällt die Beurteilung unter diese Schwelle, wird die Teilung zurückgewiesen und der Baum für die Untermenge ist das meist zugewiesene Blatt.

   - Vorteil: es werden keine Strukturen aufgebaut, die im endgültigen, vereinfachten Baum nicht gebraucht werden.
   - Nachteil: Es ist nicht leicht, solche stopping- Regeln zu finden. Ist die Schwelle zu hoch, wird die Teilung gestoppt, bevor der Nutzen späterer Teilungen offensichtlich wird. Ist dagegen die Schwelle zu niedrig, ist die Vereinfachung gering.

2. **Error- based- pruning:**
   Der Baum kann " ungehindert wachsen " und wird erst danach reduziert,. C 4.5 erlaubt die Ersetzung von Unterbäumen durch Blätter oder durch einen seiner Zweige.

Angenommen es wäre möglich, die Fehlerrate eines Baumes, mitsamt seinen Unterbäumen und Blättern, vorherzusagen, dann wäre das Vorgehen:

- beginne unten am Ende des Baumes
- würde die Ersetzung eines Unterbaumes zu Reduzierung der Fehlerrate führen, dann reduziere entsprechend
- rufe den Prozeß rekursiv wieder auf, bis der Baum minimal ist

Wie kann die Fehlerrate vorhergesagt werden?

Die Trainingsmenge, aus der der Baum aufgebaut wird, sieht keine solche Abschätzung vor, die Reduzierung führt im Gegenteil bei der Trainingsmenge zu einer größeren Fehlerrate.

Wieder gibt es 2 Mglichkeiten:

- *cost- complexity- pruning* [Breimann, 1984]
  die vorhergesagte Fehlerrate wird als gewichtete Summe ihrer Komplexität und ihrer Fehler in der Trainingsmenge modelliert.
- *redueced- error- pruning* [Quinlan, 1987]
  die Fehlerrate und die Komplexität wird durch eine Menge von Extrafällen direkt eingeschätzt.

Um zu verhindern, dass die Datenmenge, aus der der Baum aufgebaut wird, zu klein wird durch die Reservierung für die Extramenge, benutzt man den Ansatz des *Kreuzbeweises.*

Die Wahrscheinlichkeit von Fehlern kann nicht genau festgestellt werden, hat aber selber eine Wahrscheinlich- keitsverteilung, die zusammengefat wird in einem Paar aus Sicherheisgrenzen.

Gegeben: ein Sicherheitslevel CF

> $\Rightarrow$ die untere Grenze wird berechnet durch die Binominalverteilung und wird geschrieben als U_{CF} (E,N) ,wobei N die Anzahl der Trainingsfälle ist, die durch ein Blatt überdeckt werden, E davon fälschlicherweise.

C 4.5 setzt diese untere Grenze mit der erwarten Fehlerrate gleich. Ein Blatt, das N Trainingsfälle überdeckt, mit einer angenommenen Fehlerrate von U_{CF}(E,N), würde einen Fehleranstieg von $N * U_{CF}(E, N)$ ergeben. Die Anzahl der vorausgesagten Fehler bezogen auf einen Baum ist genau die Summe der vorausgesagten Fehler an seinen Zweigen.

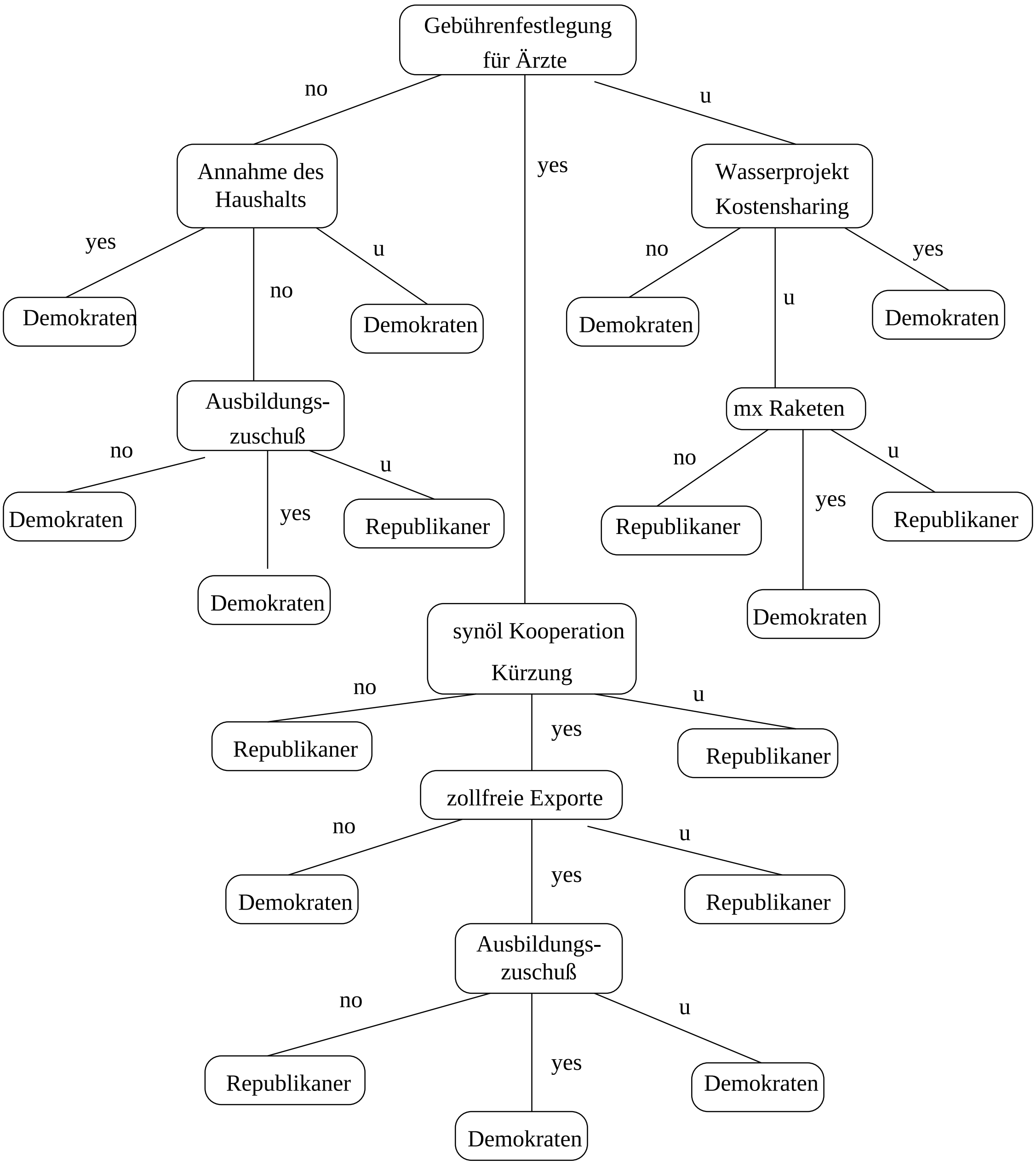
Wie viele Heuristiken mit fragwürdiger Untermauerung, scheint die Abschätzung häufig akzeptable Ergebnisse hervorzubringen.

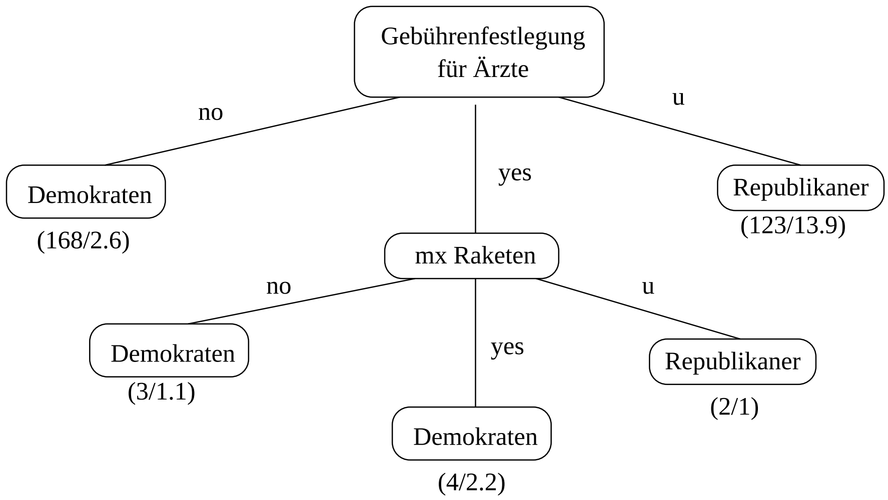
Die Summe der vorausgesagten Fehlern wird zur direkten Abschätzung der Fehlerrate im reduzierten Baum für die nicht betrachteten Fälle zur Verfügung gestellt.

Es ist zu hoffen, dass der reduzierte Baum eine niedrigere Fehlerrate hat als der Orgilalbaum in bezug auf die nichtbetrachteten Fälle, insbesondere dann, wenn die Menge der nichtbetrachteten Fälle wesentlich größer ist als die Menge der Trainingsfälle.

6.1 ein letztes Beispiel

zeigt das Abstimmungsverhalten der amerikanischen Kongressabgeordneten
vor dem Error- based- pruning und danach
siehe nächste Seite


Gebührenfestlegung
für Ärzte
no
yes
u
Annahme des
Haushalts
Wasserprojekt
Kostensharing
yes
no
u
Demokraten
Demokraten
no
u
yes
Demokraten
Demokraten
Ausbildungs-
zuschuß
mx Raketen
no
yes
u
Demokraten
Demokraten
Republikaner
no
yes
u
Republikaner
Republikaner
Demokraten
synöl Kooperation
Kürzung
no
yes
u
Republikaner
Republikaner
zollfreie Exporte
no
yes
u
Demokraten
Republikaner
Ausbildungs-
zuschuß
no
yes
u
Republikaner
Demokraten
Demokraten


Gebührenfestlegung
für Ärzte
no
yes
u
Demokraten
(168/2.6)
Republikaner
(123/13.9)
mx Raketen
no
yes
u
Demokraten
(3/1.1)
Demokraten
(4/2.2)
Republikaner
(2/1)