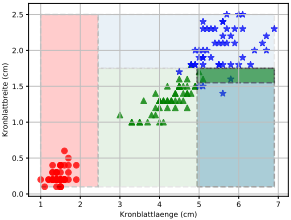
Der CART Algorithmus für die Klassifikation

Prof. Dr. Jörg Frochte

Maschinelles Lernen

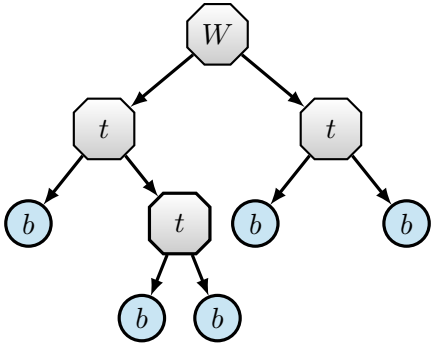


$$G = 1 - \sum_{i=1}^{c} N(i)^2$$

CART-Algorithmus

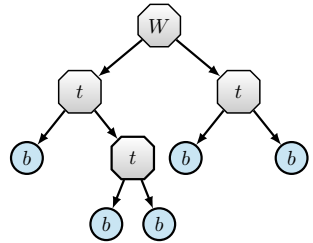
- Der CART (Classification and Regression Trees) wurde 1984 von Leo Breiman et al. publiziert.
- Wie der Name des Algorithmus' schon erahnen lässt, kann er sowohl zur Klassifizierung als auch zur Regression eingesetzt werden.
- Wir beginnen mit der Klassifikation und lernen dafür die **Gini Impurity** als Maß für die Verunreinigung kennen.
- Das Ziel ist, die *Verunreinigung* (engl. impurity) zu reduzieren, und dies über ein geeignetes Maß zu messen.
- Man will versuchen, die Trainingsmenge bei jeder Entscheidung homogener zu bekommen.
- Wir gehen im Folgenden davon aus, dass wir $c \in \mathbb{N}$ verschiedene Klassen bestimmen wollen.

- Als Beispiel nehmen wir das bekannte *Fisher's Iris data set*.
- Hier wollen wir die drei Arten ($c = 3$) von Schwertlilien bestimmen.
- Diese werden mit Integerwerten von 1 bis 3 kodiert, also z. B. $1 :=$Iris setosa, $2 :=$Iris virginica und $3 :=$Iris versicolor.
- An jedem Knoten wird nun die Trainingsmenge bzgl. ihrer *Impurity* betrachtet.



- Nehmen wir an, dass m Datensätze an einem Knoten in einer Trainingsmenge existieren, dann gibt $0 \leq N(i) \leq 1$ den Anteil davon an, der zur Klasse gehört.
- Um diesen zu berechnen, zählen wir diese Fälle und dividieren durch m.
- Wenn für alle außer einem $N(i)$ null ist und dieses eben eins, dann wird der Knoten als *pure* bezeichnet.

- Zur Gini Impurity als Maß kommt man nun, indem man sich vorstellt, an einem Knoten würde die Klassifikation gewürfelt.
- Die Chancen beim Würfel wären dabei so verteilt wie Verhältnisse in der Trainingsmenge am Knoten.
- Liegen z. B. ein Beispiel Iris setosa, 2 Beispiele Iris virginica und 3 Beispiele Iris versicolor vor, so würde in $1/6 = N(1)$ der Fälle die Klassifikation 1 erfolgen, in $1/3 = N(2)$ die Klassifikation 2 und in $1/2 = N(3)$ die Klassifikation 3.



- **Frage**: Wie sehen dann die Wahrscheinlichkeiten aus, dass Beispiele der Trainingsmenge falsch klassifiziert würden?

- Um das auszurechnen, bildet man das Produkt aus $N(i)$ der Klassifikation i, die gewählt wurde, und $\sum_{j \neq i} N(j)$, die nun fehl-klassifiziert werden, d. h.
  $N(i) \cdot \sum_{j \neq i} N(j) = \sum_{j \neq i} N(i)N(j)$.

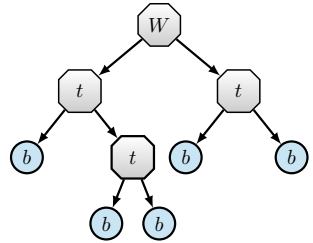
Beispiel:

Mit einer Wahrscheinlichkeit von $1/6$ wird die Klassifikation 1 vergeben. Für die Fälle der Klasse 2, d. h. $\frac{1}{3}$ aller Fälle, und die Fälle der Klasse 3, d. h. $\frac{1}{2}$ aller Fälle, wäre das falsch:

1. $\frac{1}{6} \cdot \frac{1}{3} + \frac{1}{6} \cdot \frac{1}{2} = \frac{1}{6} \cdot \frac{5}{6} = \frac{5}{36}$

Dies ist der Anteil der Fehlklassifikationen der anderen Klassen als Klasse 1. Analog erhält man Klassen 2 und 3:

2. $\frac{1}{2} \cdot \frac{1}{6} + \frac{1}{2} \cdot \frac{1}{3} = \frac{1}{4}$
3. $\frac{1}{3} \cdot \frac{1}{6} + \frac{1}{3} \cdot \frac{1}{2} = \frac{2}{9}$



- Addiert man dies nun für alle Fälle auf, ergibt sich:

$$G = \sum_{i=1}^{c} \sum_{j \neq i} N(i)N(j)$$

- In unserem Beispiel bedeutet das:

$$\begin{aligned} G &= \left(\frac{1}{6} \cdot \frac{1}{3} + \frac{1}{6} \cdot \frac{1}{2}\right) + \left(\frac{1}{3} \cdot \frac{1}{6} + \frac{1}{3} \cdot \frac{1}{2}\right) + \left(\frac{1}{2} \cdot \frac{1}{6} + \frac{1}{2} \cdot \frac{1}{3}\right) \\ &= \frac{1}{6} \cdot \frac{5}{6} + \frac{1}{3} \cdot \frac{2}{3} + \frac{1}{2} \cdot \frac{1}{2} = \frac{22}{36} = \frac{11}{18} \end{aligned}$$

- Die Doppelsumme ist aufwendig zu berechnen, daher formen wir weiter um.
- Wir nutzen dazu, dass alle $N(i)$ addiert eins ergeben müssen. Es gilt also:

$$1 = \sum_{j=1}^{c} N(j) = N(1) + N(2) + \cdots + N(c) \Leftrightarrow 1 - N(i) = \sum_{j=1, j \neq i}^{c} N(j)$$

$$1 = \sum_{j=1}^{c} N(j) = N(1) + N(2) + \cdots + N(c) \Leftrightarrow 1 - N(i) = \sum_{j=1, j\neq i}^{c} N(j)$$

- Das können wir nun für die Umformung an der mit $(*)$ markierten Stelle verwenden, da $N(i)$ in der Formel vor das Summenzeichen gezogen werden darf:

$$G = \sum_{i=1}^{c} \sum_{j\neq i} N(i)N(j) = \sum_{i=1}^{c} N(i) \underbrace{\sum_{j\neq i} N(j)}_{(*)} = \sum_{i=1}^{c} N(i)\left(1 - N(i)\right)$$

$$= \sum_{i=1}^{c} N(i) - N(i)^2 = \sum_{i=1}^{c} N(i) - \sum_{i=1}^{c} N(i)^2 = 1 - \sum_{i=1}^{c} N(i)^2$$

- Diese Größe ist nun leichter zu berechnen, wie sich schon an unserem Beispiel zeigt:

$$G = 1 - \left(\frac{1}{6}\right)^2 - \left(\frac{1}{3}\right)^2 - \left(\frac{1}{2}\right)^2 = \frac{11}{18} \approx 0.61$$

- Unser Ziel ist es, die Gini Impurity in jedem Schritt zu verringern.
- In einem reinen Knoten wird $\sum_{i=1}^{c} N(i)^2 = 1$ und wir erhalten 0 für die Gini Impurity.
- Je stärker die Menge durchmischt ist, desto mehr nähert sie sich 1 an.
- Um die Gini Impurity von zwei Teilmengen X_1, X_2 zu ermitteln, bilden wir das gewichtete Mittel der entsprechenden Gini Impurities G_1, G_2.

$$\bar{G} = \frac{G_1 \cdot |X_1| + G_2 \cdot |X_2|}{|X_1| + |X_2|}$$

- Um uns klarzumachen, wie der CART hier vorgeht, nehmen wir tatsächlich das Schwertlilien-Beispiel, tun jedoch so, als wenn uns nur zwei Merkmale, nämlich die Kronblattlänge und -breite zur Verfügung stünden.
- Der CART nutzt nun die Merkmale, um das jeweils verbleibende Merkmalsgebiet mit Schnitten parallel zu den Achsen aufzuteilen.
- Dies reduziert deutlich die Komplexität, eine gute Aufteilung zu finden.

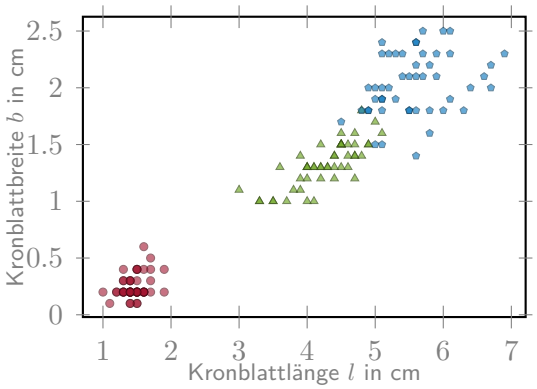
- Es gibt in $I \subset \mathbb{R}$ unendlich viele Schnittpunkte, die man testen könnte. Wir nutzen daher die Beispiele der Trainingsmenge, um mögliche Schnittpunkte zu definieren.
- Nehmen wir an, mit T_i wären die n Werte des Merkmales i in aufsteigender Reihenfolge aus der Trainingsmenge bezeichnet. Dann bilden wir die Menge der Testpunkte durch:

$$A_i = \left(T_i[j] + T_i[j+1]\right)/2 \qquad j = 0, \ldots, n-1$$

- Nun testen wir für jeden dieser Punkte, ob sich die Gini Impurity verringert, wenn wir durch diesen Wert eine Aufteilung im Merkmalsraum vornehmen.
- Wenn wir uns dies aufgrund der Datenmenge leisten können, betrachten wir alle so berechneten Testpunkte für alle Merkmale und wählen denjenigen Schnitt aus, welche die Verunreinigung am meisten reduziert.
- Wird dies zu kostspielig, kann dem Algorithmus auch ein Teil der Merkmale nicht zur Auswahl gegeben und so der Suchraum eingeschränkt werden.
- Natürlich verringert dies in der Regel die Qualität des Ergebnisses.

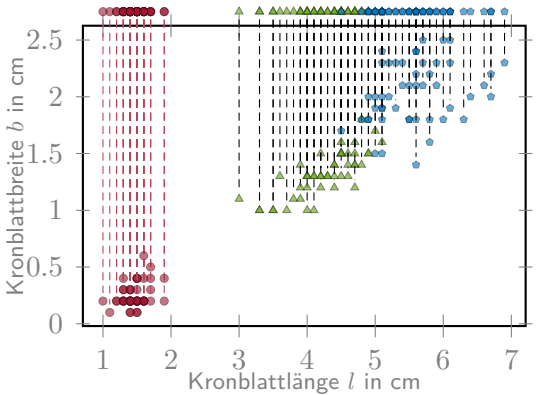
CART mit Gini Impurity

- Beispiel: Schwertlinien Datenbank mit zwei Merkmalen.



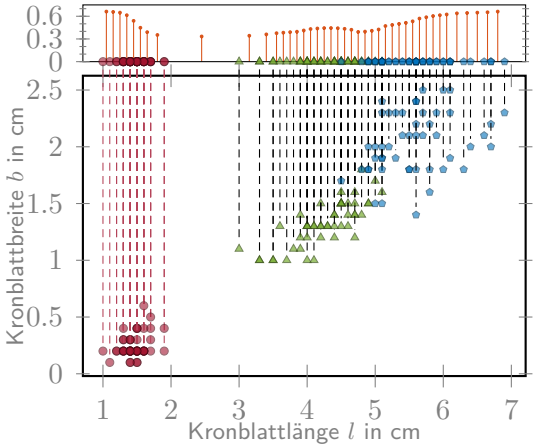
CART mit Gini Impurity

- Beispiel: Schwertlinien Datenbank mit zwei Merkmalen.
- Wir beginnen mit der Kronblattlänge.



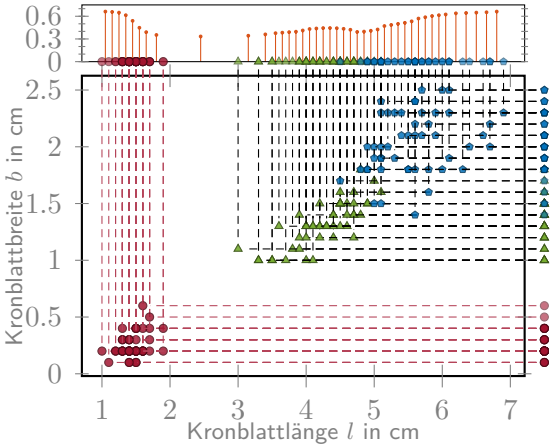
CART mit Gini Impurity

- Beispiel: Schwertlinien Datenbank mit zwei Merkmalen.
- Wir beginnen mit der Kronblattlänge.
- Die Gini Impurity wird für jede Schnittmöglichkeit A_i berechnet.



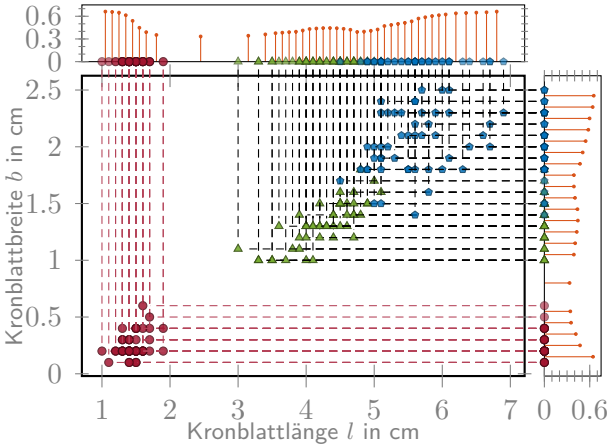
CART mit Gini Impurity

- Beispiel: Schwertlinien Datenbank mit zwei Merkmalen.
- Wir beginnen mit der Kronblattlänge.
- Die Gini Impurity wird für jede Schnittmöglichkeit A_i berechnet.
- Dies wird für alle Merkmale gemacht.



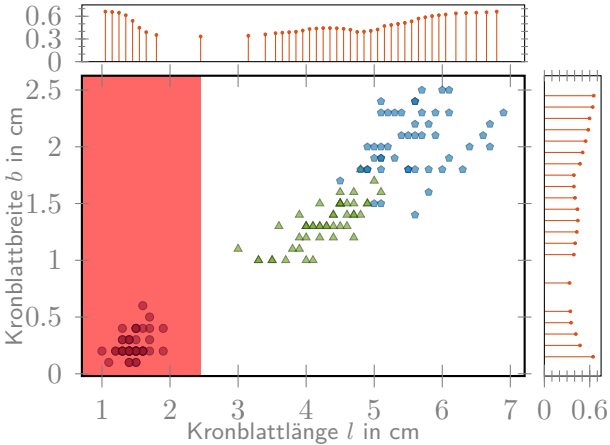
CART mit Gini Impurity

- Beispiel: Schwertlinien Datenbank mit zwei Merkmalen.
- Wir beginnen mit der Kronblattlänge.
- Die Gini Impurity wird für jede Schnittmöglichkeit A_i berechnet.
- Dies wird für alle Merkmale gemacht.
- Der Schnitt wird bei der kleinsten Gini Impurity gemacht.



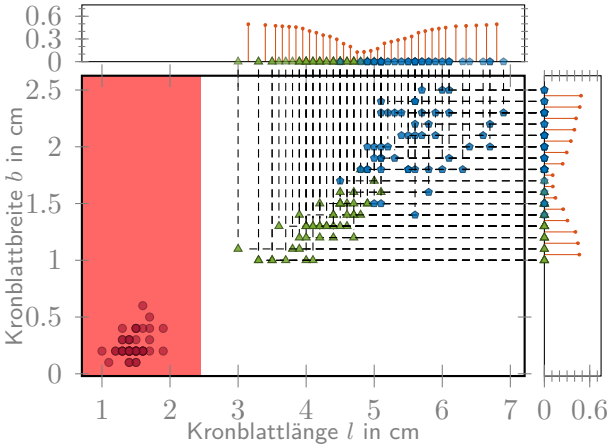
CART mit Gini Impurity

- Beispiel: Schwertlinien Datenbank mit zwei Merkmalen.
- Wir beginnen mit der Kronblattlänge.
- Die Gini Impurity wird für jede Schnittmöglichkeit A_i berechnet.
- Dies wird für alle Merkmale gemacht.
- Der Schnitt wird bei der kleinsten Gini Impurity gemacht.
- Anschließend werden unreine Bereiche weiter unterteilt.



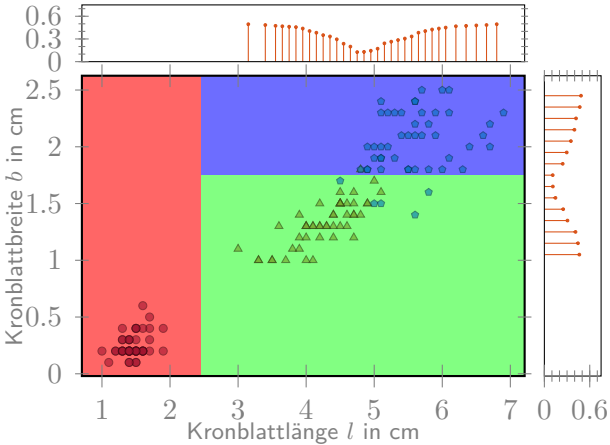
CART mit Gini Impurity

- Beispiel: Schwertlinien Datenbank mit zwei Merkmalen.
- Wir beginnen mit der Kronblattlänge.
- Die Gini Impurity wird für jede Schnittmöglichkeit A_i berechnet.
- Dies wird für alle Merkmale gemacht.
- Der Schnitt wird bei der kleinsten Gini Impurity gemacht.
- Anschließend werden unreine Bereiche weiter unterteilt.



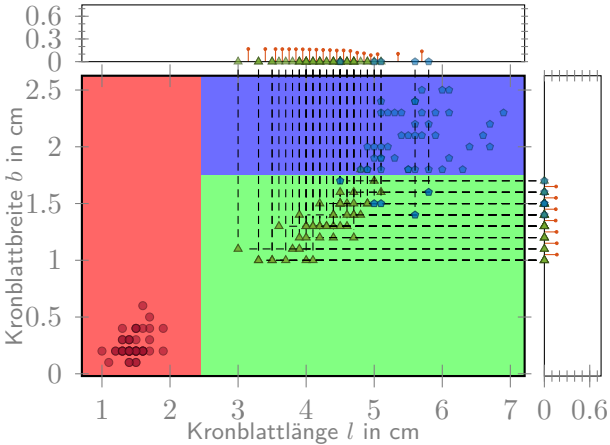
CART mit Gini Impurity

- Beispiel: Schwertlinien Datenbank mit zwei Merkmalen.
- Wir beginnen mit der Kronblattlänge.
- Die Gini Impurity wird für jede Schnittmöglichkeit A_i berechnet.
- Dies wird für alle Merkmale gemacht.
- Der Schnitt wird bei der kleinsten Gini Impurity gemacht.
- Anschließend werden unreine Bereiche weiter unterteilt.



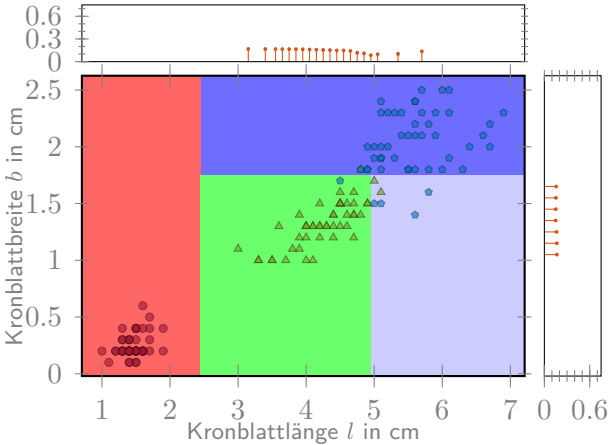
CART mit Gini Impurity

- Beispiel: Schwertlinien Datenbank mit zwei Merkmalen.
- Wir beginnen mit der Kronblattlänge.
- Die Gini Impurity wird für jede Schnittmöglichkeit A_i berechnet.
- Dies wird für alle Merkmale gemacht.
- Der Schnitt wird bei der kleinsten Gini Impurity gemacht.
- Anschließend werden unreine Bereiche weiter unterteilt.



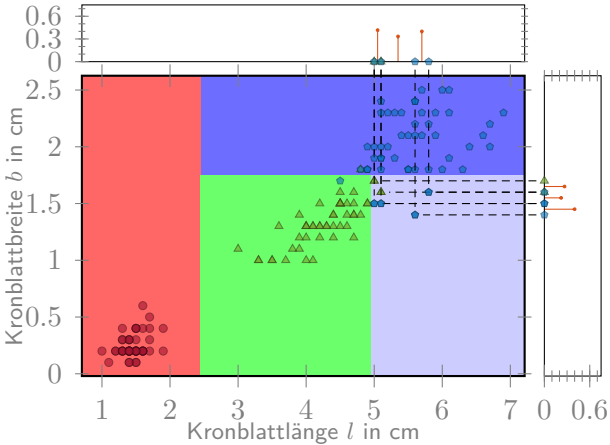
CART mit Gini Impurity

- Beispiel: Schwertlinien Datenbank mit zwei Merkmalen.
- Wir beginnen mit der Kronblattlänge.
- Die Gini Impurity wird für jede Schnittmöglichkeit A_i berechnet.
- Dies wird für alle Merkmale gemacht.
- Der Schnitt wird bei der kleinsten Gini Impurity gemacht.
- Anschließend werden unreine Bereiche weiter unterteilt.



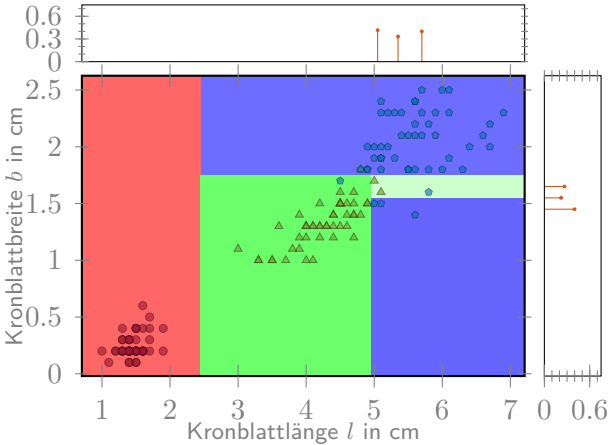
CART mit Gini Impurity

- Beispiel: Schwertlinien Datenbank mit zwei Merkmalen.
- Wir beginnen mit der Kronblattlänge.
- Die Gini Impurity wird für jede Schnittmöglichkeit A_i berechnet.
- Dies wird für alle Merkmale gemacht.
- Der Schnitt wird bei der kleinsten Gini Impurity gemacht.
- Anschließend werden unreine Bereiche weiter unterteilt.

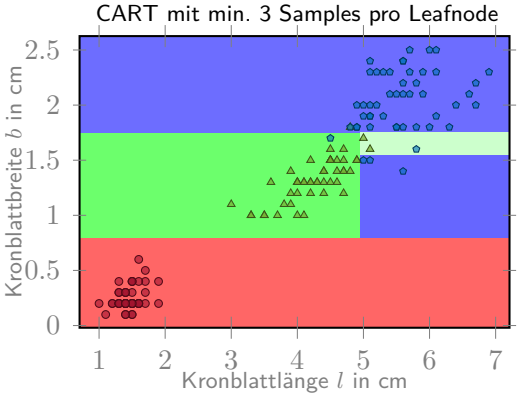


CART mit Gini Impurity

- Beispiel: Schwertlinien Datenbank mit zwei Merkmalen.
- Wir beginnen mit der Kronblattlänge.
- Die Gini Impurity wird für jede Schnittmöglichkeit A_i berechnet.
- Dies wird für alle Merkmale gemacht.
- Der Schnitt wird bei der kleinsten Gini Impurity gemacht.
- Anschließend werden unreine Bereiche weiter unterteilt.

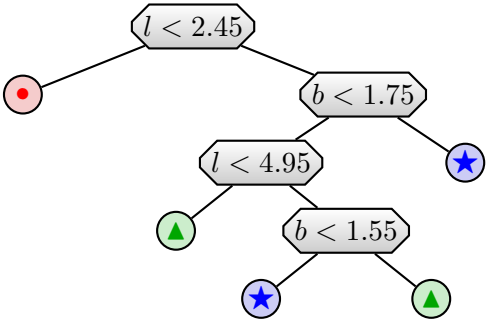


- Allerdings ist auch bei Ausschöpfung aller Merkmale nicht sichergestellt, dass das Ergebnis bzgl. der Klassifizierung fehlerfrei ist.
- CART-Bäume reagieren bzgl. ihrer Gestalt sehr empfindlich auf Änderungen in der Trainingsmenge und nehmen deutlich andere Schnitte vor.



- Die Auswirkung bzgl. der Qualität auf der Trainingsmenge sind gering, jedoch kann es bei der Testmenge zu größeren Verschiebungen kommen.
- Der erste Schnitt erfolgte auf der Basis des nullten Merkmals – wie wir später sehen werden, liegt das an der Implementierung – hätte jedoch genauso gut etwa bei 0.75 auf der Basis des ersten Merkmals erfolgen können. etc.

- Es reicht für den gelernten Baum, sich auf einen Vergleichsoperator, also $<$ oder $>$, festzulegen. Wir testen immer auf `kleiner`.



Entscheidungsbaum für Iris Dataset beschränkt auf zwei Merkmale