Maschinelles Lernen 07

Prof. Dr. David Spieler – david.spieler@hm.edu

Hochschule München

8. Januar 2020

Entscheidungsbäume

Entscheidungsbäume

Entscheidungsbäume

Bei den bisherigen Klassifikationsmethoden hatten wir angenommen, dass unsere Features reellwertig oder im Spezialfall diskret waren, also dass $\mathcal{X} \subseteq \mathbb{R}^d$ bzw. $\mathcal{X} \subseteq \mathbb{N}^d$.
In beiden Fällen gibt es natürliche Distanzmetriken, wie z.B. die euklidische Norm $||\mathbf{x} - \mathbf{x}'||_2$. Mit Hilfe einer solchen Metrik kann z.B. bei KNN der bzw. die nächsten Nachbarn bestimmt werden.

Entscheidungsbäume

Wir müssen jedoch auch den Fall betrachten, wenn $\mathcal{X}$ nominal, das bedeutet diskret, jedoch ohne natürliche Distanzmetrik ist. Zum Beispiel wollen wir anhand der Features

- Farbe $\mathcal{X}_{\text{Farbe}} = \{\text{rot}, \text{grün}, \text{gelb}\}$
- Form $\mathcal{X}_{\text{Form}} = \{\text{rund}, \text{dünn}\}$
- Größe $\mathcal{X}_{\text{Größe}} = \{\text{groß}, \text{mittel}, \text{klein}\}$
- Geschmack $\mathcal{X}_{\text{Geschmack}} = \{\text{süß}, \text{sauer}\}$

verschiedene Obstsorten beschreiben.

Entscheidungsbäume

Ein Objekt $\mathbf{x} \in \mathcal{X}$ wird demnach durch ein d-Tupel, in diesem Beispiel durch ein 4-Tupel

$$\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\mathbf{x}_4) \in \mathcal{X}$$

mit

$$\mathcal{X} = \mathcal{X}_{\text{Farbe}} \times \mathcal{X}_{\text{Form}} \times \mathcal{X}_{\text{Größe}} \times \mathcal{X}_{\text{Geschmack}}.$$

So wäre ein Apfel etwa beschrieben durch

$$(\text{rot}, \text{rund}, \text{mittel}, \text{süß}).$$

Entscheidungsbäume

Bei der Klassifikation mit Entscheidungsbäumen befinden wir uns demnach in einem Szenario, bei welchen wir eine Funktion

$$f : \mathcal{X} \to \mathcal{Y}$$

lernen wollen, wobei $\mathcal{X}$ reell, diskret oder auch nominal ist und

$$\mathcal{Y} = \{y_1, \ldots, y_m\}$$

eine diskrete Menge von Klassen.

Entscheidungsbäume

Bei Entscheidungsbäumen klassifiziert man nun Objekte anhand nominaler Kriterien mit Hilfe von Fragensequenzen, wobei die nächste Frage abhängt von der Antwort auf die aktuelle Frage. Wichtig dabei ist, dass die Antwort auf jede Frage nominal ist wie z.B. generell $\{\text{ja}, \text{nein}\}$ oder speziell $\{\text{rot}, \text{grün}, \text{gelb}\}$. Solche Sequenzen von Fragen werden systematisch in einem Entscheidungsbaum repräsentiert.

Entscheidungsbäume

Die Knoten des Baums sind systematische Fragen und die Kanten die jeweiligen Antwortmöglichkeiten. Die Blätter des Baums sind die Klassen. Die Klassifikation beginnt mit der Frage in der Wurzel des Baums und endet in einem Blatt, also einer Klasse.
Die Kanten, die einen Knoten verlassen müssen

- eindeutig und
- erschöpfend

sein, sodass immer genau einer Kante gefolgt wird.

Entscheidungsbäume

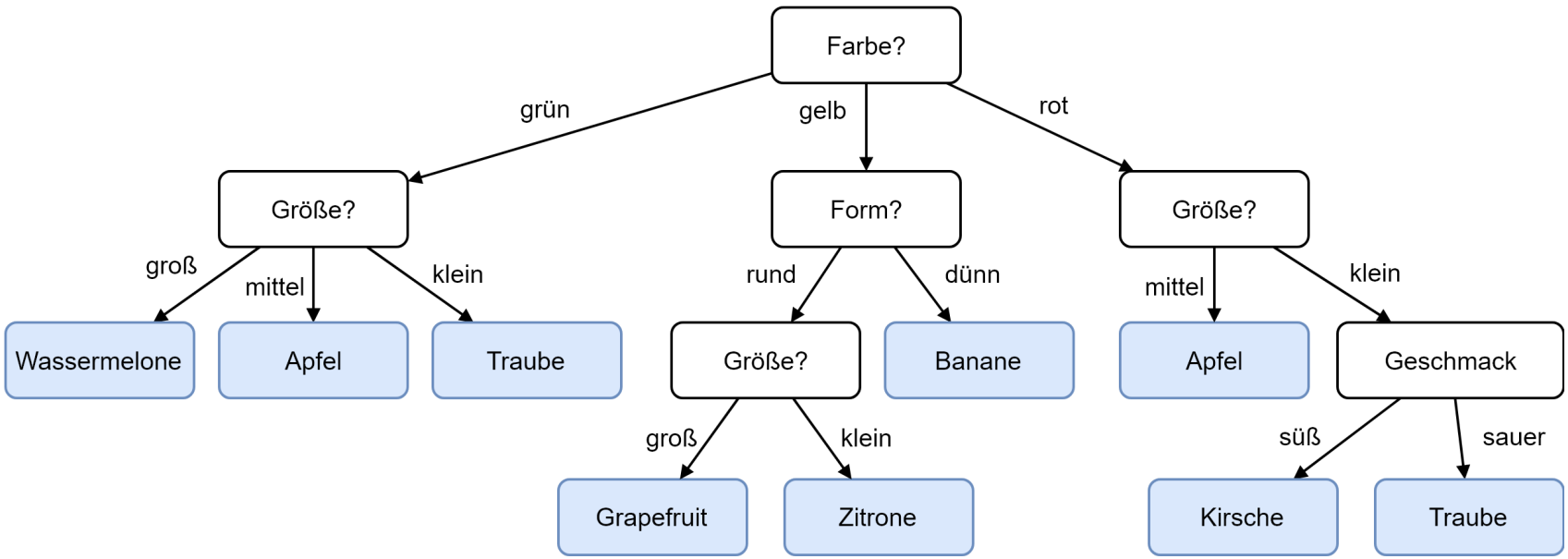


Abbildung 1: Entscheidungsbaum zur Klassifikation von Eigenschaften nach Obstsorten. Die Blätter des Baums sind die Klassen (Obstsorten) und sind blau markiert. Abbildung adaptiert von [DHS00].

Entscheidungsbäume

Eine Eigenschaft von Entscheidungsbäumen ist, dass sie sehr gut interpretierbar sind:

- Die Klassifikation einzelner Datenpunkte $\mathbf{x} \in \mathcal{X}$ kann vom Menschen nachvollzogen werden.
- Die Klassen $y \in \mathcal{Y}$ selbst erhalten eine Beschreibung anhand von logischen Kriterien. Zum Beispiel:

$$\begin{aligned} \text{Apfel} &= (\text{Farbe} = \text{grün} \wedge \text{Größe} = \text{mittel}) \\ &\quad \vee (\text{Farbe} = \text{rot} \wedge \text{Größe} = \text{mittel}) \\ &\overset{!}{=} (\text{Farbe} = \text{grün} \vee \text{Farbe} = \text{rot}) \wedge \text{Größe} = \text{mittel} \end{aligned}$$

- Entscheidungsbäume können daher auch durch explizites Vorwissen ergänzt werden.

Entscheidungsbäume
CART

Wie wird nun ausgehend von

- einer Menge von Trainingsdaten $\mathcal{D} \subseteq \mathcal{X} \times \mathcal{Y}$ und
- einer Menge von Entscheidungsfragen

ein Entscheidungsbaum gelernt? Ein Entscheidungsbaum teilt sukzessive die Menge $\mathcal{D}$ in immer kleinere Teilmengen. Idealerweise endet jeder Pfad in einer reinen Menge, d.h. einer Menge $\mathcal{F} \subseteq \mathcal{D}$ für die gilt, dass alle Labels y mit $(\mathbf{x}, y) \in \mathcal{F}$ *gleich* sind. Dies ist üblicherweise nicht der Fall und es muss geregelt werden, ob in solchen weitere Aufteilungen erfolgen sollen.

Entscheidungsbäume
CART

Allgemeiner Lernalgorithmus für Entscheidungsbäumen mit Trainingsdaten $\mathcal{D}$ und Menge an Fragen $\mathcal{Q}$:

Algorithm 1 dtree($\mathcal{D}$, $\mathcal{Q}$)

1: **if** stop_criteria($\mathcal{D}$) **then**
2: **return** LEAF(compute_class($\mathcal{D}$))
3: **else**
4: **for** each $q \in \mathcal{Q}$ **do**
5: $S^q = \text{split}(\mathcal{D}, q)$
6: $\Delta i_q = \text{compute\_improvement}(\mathcal{D}, S)$
7: **end for**
8: $b = \arg\max_{\{q \mid \Delta i_q > 0\}} \Delta i_q$
9: **return** NODE(b,dtree($S_1^{(b)}$, $\mathcal{Q} \setminus b$), dtree($S_2^{(b)}$, $\mathcal{Q} \setminus b$), ...)
10: **end if**

Entscheidungsbäume
CART

Das CART (Classification And Regression Tree) Framework bietet eine allgemeine Methodik um verschiedenste Arten von Entscheidungsbäumen zu generieren – anhand von sechs grundlegenden Fragestellungen:

1. Wieviele Entscheidungsmöglichkeiten und damit Aufteilungen gibt es pro Knoten?
2. Welche Eigenschaften werden in einem Knoten getestet?
3. Wann soll ein Knoten zu einem Blatt werden?
4. Wie wird ein zu großer Baum gestutzt?
5. Wie soll einem unreinen Blatt eine Klasse zugeordnet werden?
6. Wie wird mit unvollständigen Daten umgegangen?

Entscheidungsbäume

Aufteilungen

Jede Entscheidung ist mit einem Split (Aufteilung) der Trainingsdaten verbunden. Die Anzahl der Splits kann frei gewählt werden und auch innerhalb eines Baums variieren. Bereits zwei Splits reichen im Allgemeinen aus, d.h. binäre Entscheidungsbäume sind universell. Die Entscheidung beinflußt potentiell die Performance der Methodik und auch die Wahl der Eigenschaften.

Entscheidungsbäume

Aufteilungen

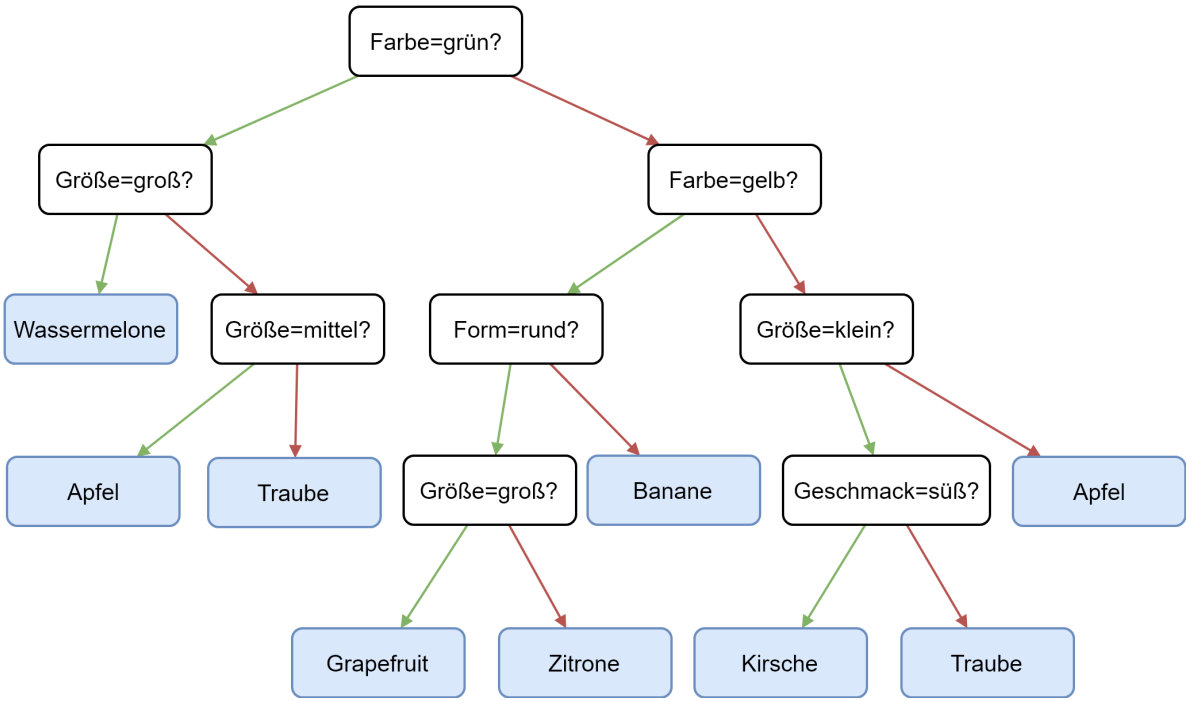


Abbildung 2: Ein binärer Entscheidungsbaum zur Klassifikation von Eigenschaften nach Obstsorten. Die Blätter des Baums sind die Klassen (Obstsorten) und sind blau markiert. Positive Entscheidungen ("ja") sind grün und negative Entscheidungen ("nein") sind rot markiert. Abbildung adaptiert von [DHS00].

Entscheidungsbäume

Fragenauswahl und Knotenunreinheit

Hauptziel der Erstellung eines Entscheidungsbaums ist die Einfachkeit, d.h. ein Baum mit möglichst wenig Kanten und Knoten. Wir suchen daher für jeden Knoten die Frage, welche die resultierenden Datenmengen so rein wie möglich macht. Wir nähern uns formal dem Konzept der Reinheit durch das Gegenteil, der Unreinheit (Impurity). Wir bezeichnen mit $i(N)$ die Unreinheit in Knoten N und es sollte gelten, dass

- $i(N) = 0$, falls alle Daten in Knoten N die gleiche Klasse haben
- $i(N)$ groß, wenn alle Kategorien gleich häufig vertreten sind

Entscheidungsbäume

Fragenauswahl und Knotenunreinheit

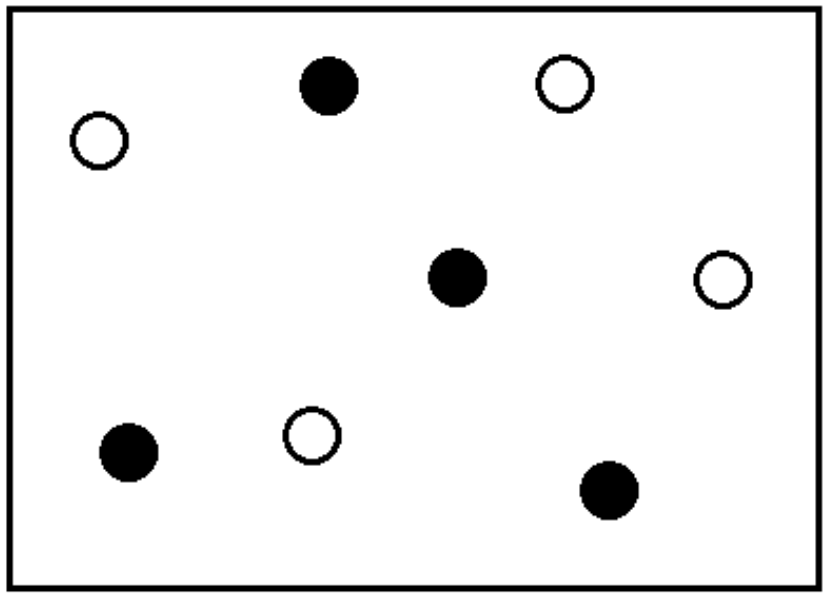
Das bekannteste Maß für Unreinheit oder auch Unordnung ist die Entropie definiert als

$$i(N) = -\sum_{j=1}^{m} P(y_j) \log_2 P(y_j)$$

wobei $P(y_j)$ die relative Häufigkeit der Klasse y_j innerhalb der Trainingsdaten an Knoten N bezeichnet.

Entscheidungsbäume

Fragenauswahl und Knotenunreinheit

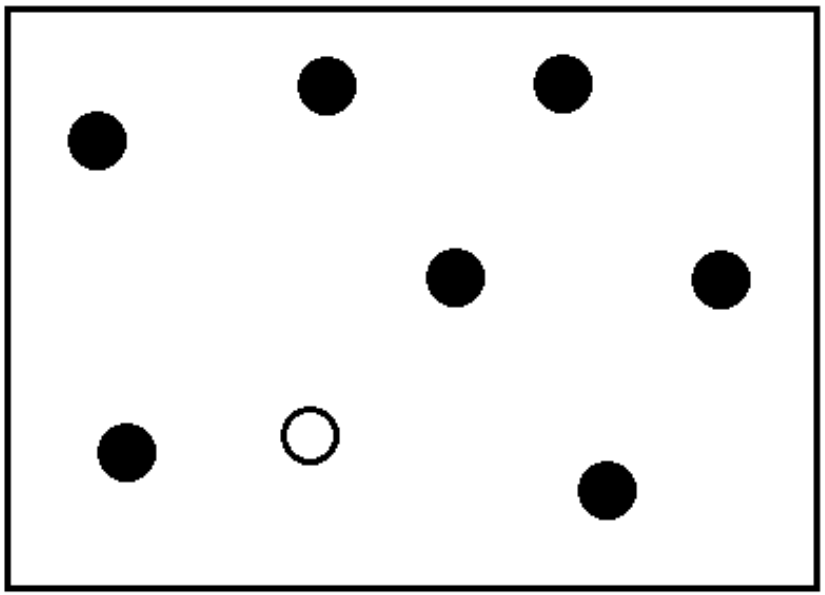


Schwarze und weiße Kugeln - Unrein

$$p_\circ = p_\bullet = \frac{4}{8} \Rightarrow i(N) = -2 \cdot 0.5 \log_2 0.5 = 1$$

Entscheidungsbäume

Fragenauswahl und Knotenunreinheit

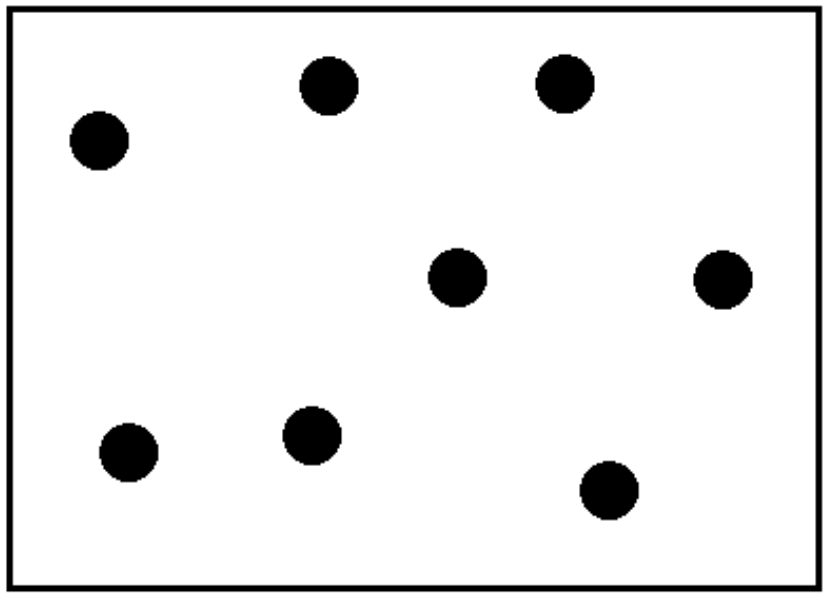


Schwarze und weiße Kugeln - Reiner

$$p_\circ = \frac{1}{8}, p_\bullet = \frac{7}{8} \Rightarrow i(N) = -\frac{1}{8}\log_2\frac{1}{8} - \frac{7}{8}\log_2\frac{7}{8} \approx 0.54$$

Entscheidungsbäume

Fragenauswahl und Knotenunreinheit

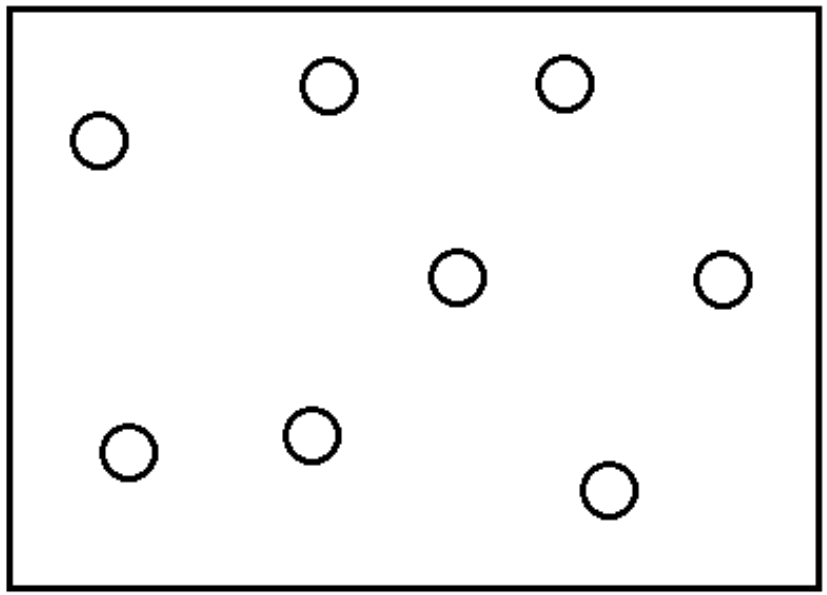


Schwarze und weiße Kugeln - Rein

$$p_\circ = 0, p_\bullet = 1 \Rightarrow i(N) = -1 \log_2 1 = 0$$

Entscheidungsbäume

Fragenauswahl und Knotenunreinheit



Schwarze und weiße Kugeln - Rein

$$p_\circ = 1, p_\bullet = 0 \Rightarrow i(N) = -1 \log_2 1 = 0$$

Entscheidungsbäume

Fragenauswahl und Knotenunreinheit

Ein weiteres häufiges Maß ist die Gini Unreinheit definiert als

$$i(N) = \sum_{i \neq j} P(y_i)P(y_j) = \frac{1}{2}\left(1 - \sum_{j=1}^{m} P^2(y_j)\right)$$

und die Missclassification Impurity definiert als

$$i(N) = 1 - \max_j P(y_j).$$

Entscheidungsbäume

Fragenauswahl und Knotenunreinheit

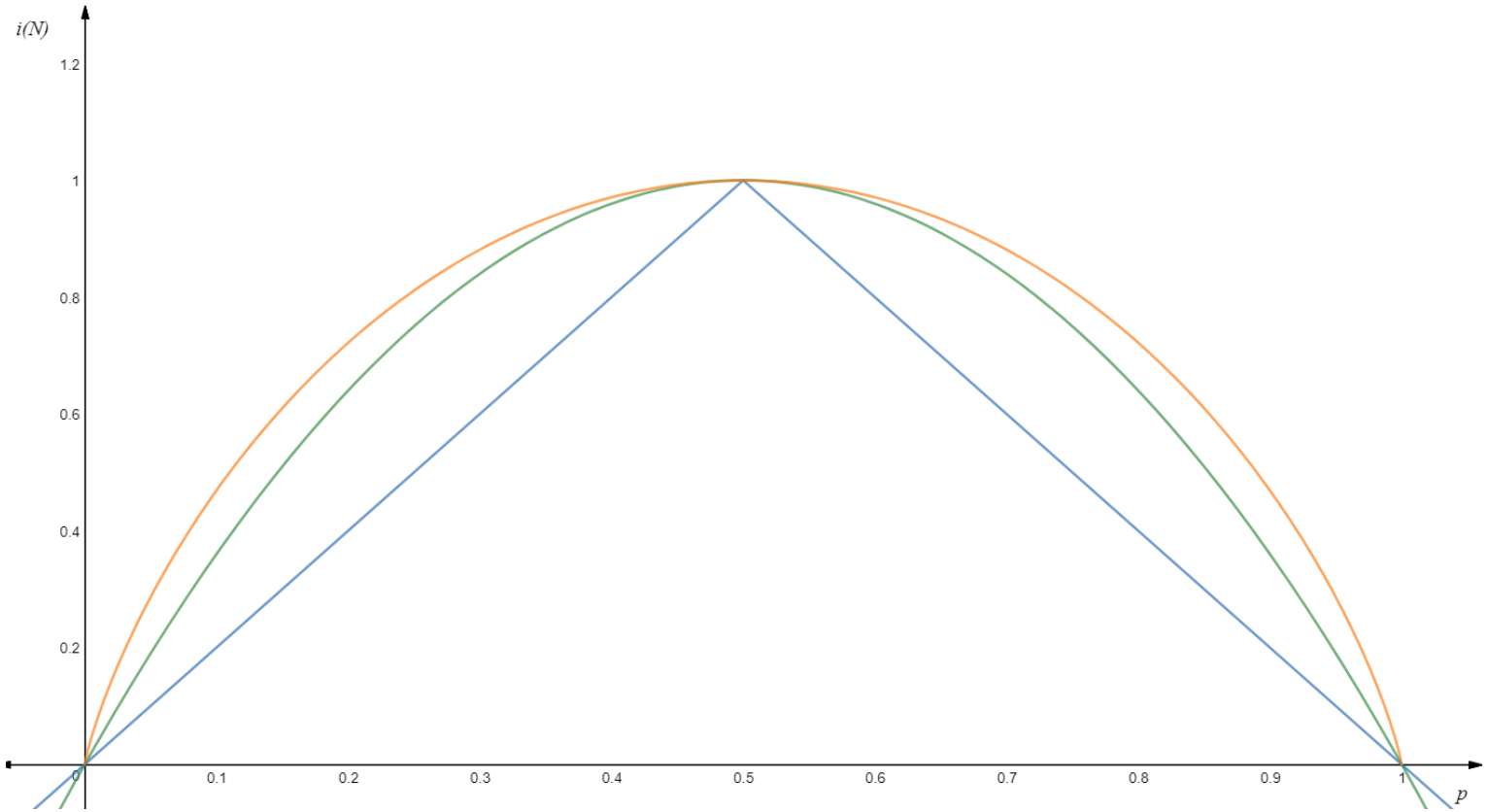


Abbildung 3: Plots der Unreinheitsmaße Entropie (orange), Gini Impurity (grün) und Missclassification Impurity (blau) in Abhängigkeit von der relativen (binären) Klassenzugehörigkeit $p = P(y_1) = 1 - P(y_2)$.

Entscheidungsbäume

Fragenauswahl und Knotenunreinheit

Wir betrachten der Einfachheit halber nur den Fall eines (bisher partiell bis zu einem Knoten N erstellten) binären Baums und wollen wissen, welche Frage an diesem Knoten an die übrigen Testdaten gestellt werden sollte. Eine Heuristic in diesem Fall ist die Frage, welche den Rückgang bzgl. der Unreinheit definiert als

$$\Delta i(N) = i(N) - P_P i(N_P) - P_N i(N_N)$$

minimiert. N_P und N_N sind die positiven bzw. negativen Nachfolgeknoten von N und $P_P = 1 - P_N$ ist der Anteil der Datenpunkte, die dem positiven Knoten zugeordnet werden.

Entscheidungsbäume

Fragenauswahl und Knotenunreinheit

Hinweise:

- Bei nominalen Features muss meist ein vollständiger Vergleich aller möglichen Fragen pro Knoten in allen Dimensionen durchgeführt werden. Im Beispiel der Obstklassifikation etwa Größe $=$ klein, $\ldots$, Größe $=$ groß, Farbe $=$ rot, $\ldots$, Farbe $=$ gelb, Geschmack $=$ sauer, $\ldots$
- Bei diskreten und reellen Features werden oft Vergleiche der Art $\mathbf{x}_i <= c$ mit $c \in \mathbb{R}$ verwendet. Generell beschränkt sich der Suchraum für die Konstanten c meist auf tatsächlich in den Trainingsdaten vorkommende Werte von $\mathbf{x}_i$ oder gewichtete Mittel.

Entscheidungsbäume

Fragenauswahl und Knotenunreinheit

Wir wollen nun einen Entscheidungsbaum mit Hilfe der bisherigen Ideen (Entropie als Unreinheitsmaß) aufbauen.

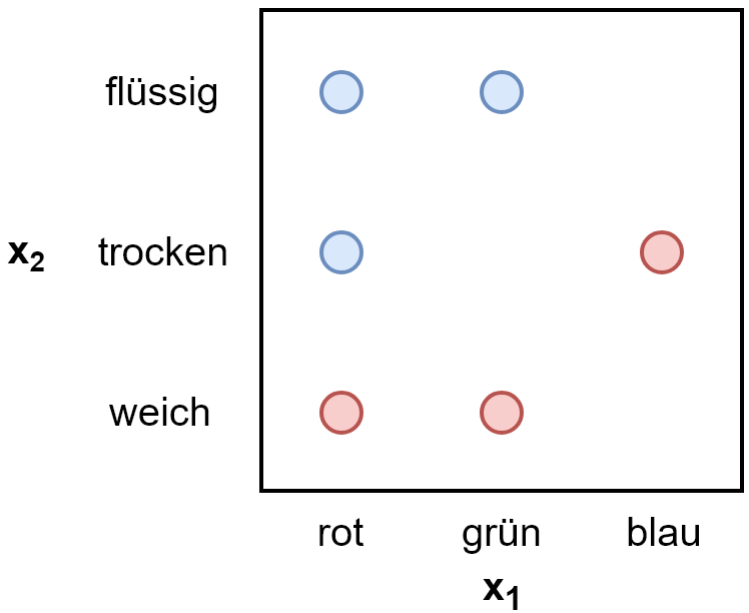


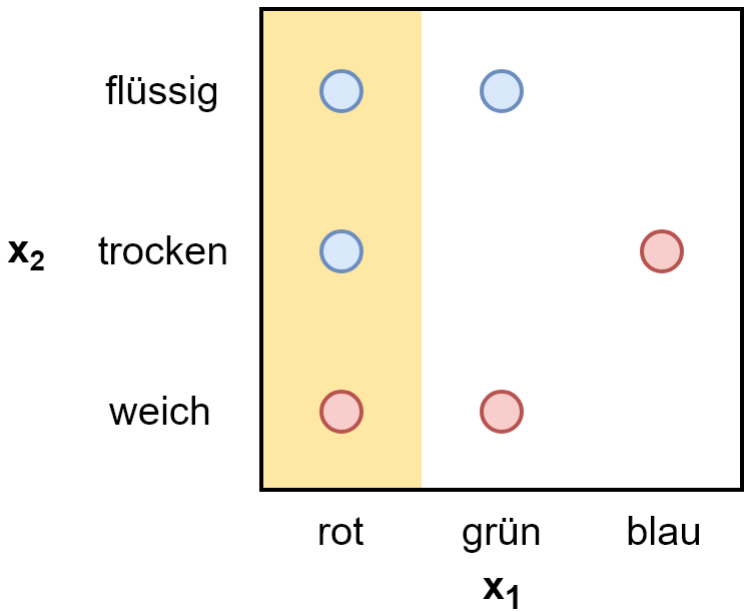
Abbildung 4: Beispiel: Klassifikation von Süßigkeiten.

$$i(N) = -\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2} = 1$$

Entscheidungsbäume

Fragenauswahl und Knotenunreinheit

Frage an der Wurzel $\mathbf{x}_1$ =rot:



$$i(L) = -\frac{1}{3}\log_2\frac{1}{3} - \frac{2}{3}\log_2\frac{2}{3} \approx 0.9183$$

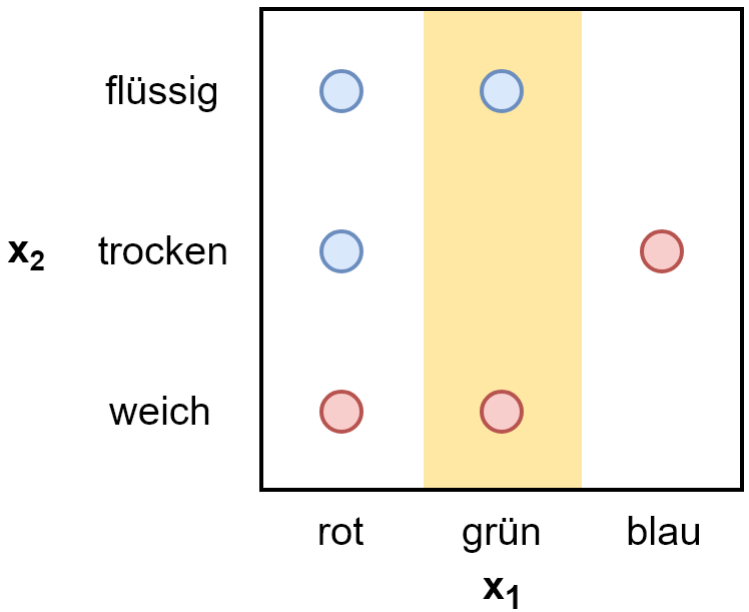
$$i(R) = -\frac{2}{3}\log_2\frac{2}{3} - \frac{1}{3}\log_2\frac{1}{3} \approx 0.9183$$

$$\Delta i(N) \approx 1 - \frac{1}{2}\cdot 0.9183 - \frac{1}{2}\cdot 0.9183 \approx \mathbf{0.0817}$$

Entscheidungsbäume

Fragenauswahl und Knotenunreinheit

Frage an der Wurzel $\mathbf{x}_1$ =grün:



$$i(L) = -\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2} = 1$$

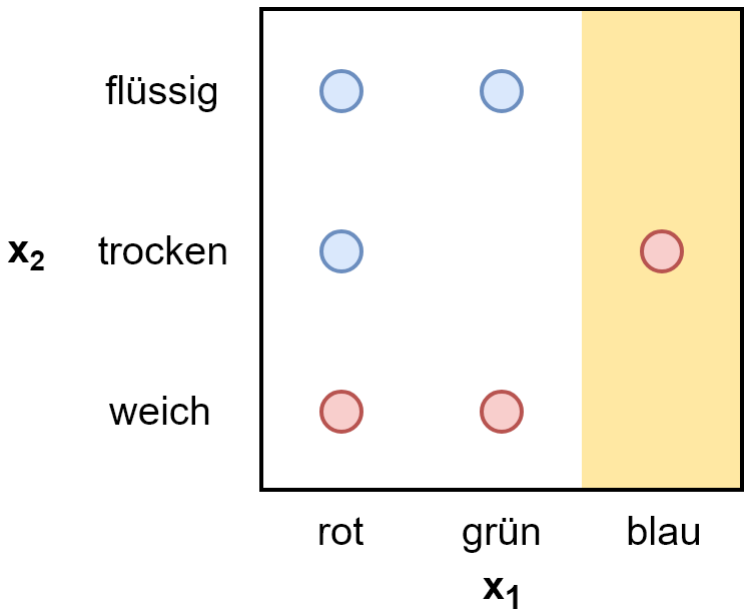
$$i(R) = -\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2} = 1$$

$$\Delta i(N) = 1 - \frac{2}{6}\cdot 1 - \frac{4}{6}\cdot 1 = \mathbf{0}$$

Entscheidungsbäume

Fragenauswahl und Knotenunreinheit

Frage an der Wurzel $\mathbf{x}_1$ =blau:



$$i(L) = -1 \log_2 1 - \text{“}0 \log_2 0\text{“} = 0$$

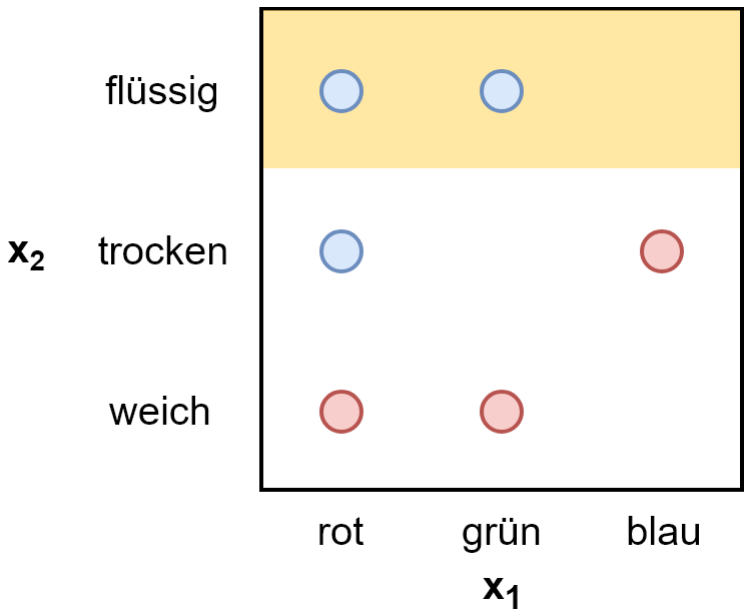
$$i(R) = -\frac{2}{5} \log_2 \frac{2}{5} - \frac{3}{5} \log_2 \frac{3}{5} \approx 0.971$$

$$\Delta i(N) \approx 1 - \frac{1}{6} \cdot 0 - \frac{5}{6} \cdot 0.971 \approx \mathbf{0.1908}$$

Entscheidungsbäume

Fragenauswahl und Knotenunreinheit

Frage an der Wurzel $\mathbf{x}_2$ =flüssig:



$$i(L) = -\text{“}0 \log_2 0\text{“} - 1 \log_2 1 = 0$$

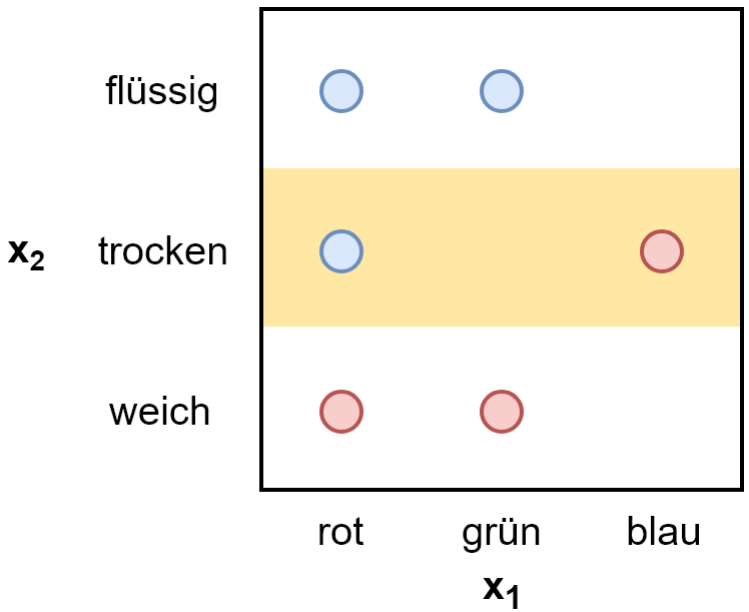
$$i(R) = -\frac{3}{4} \log_2 \frac{3}{4} - \frac{1}{4} \log_2 \frac{1}{4} \approx 0.8113$$

$$\Delta i(N) \approx 1 - \frac{1}{3} \cdot 0 - \frac{2}{3} \cdot 0.8113 \approx \mathbf{0.4591}$$

Entscheidungsbäume

Fragenauswahl und Knotenunreinheit

Frage an der Wurzel $\mathbf{x}_2$ =trocken:

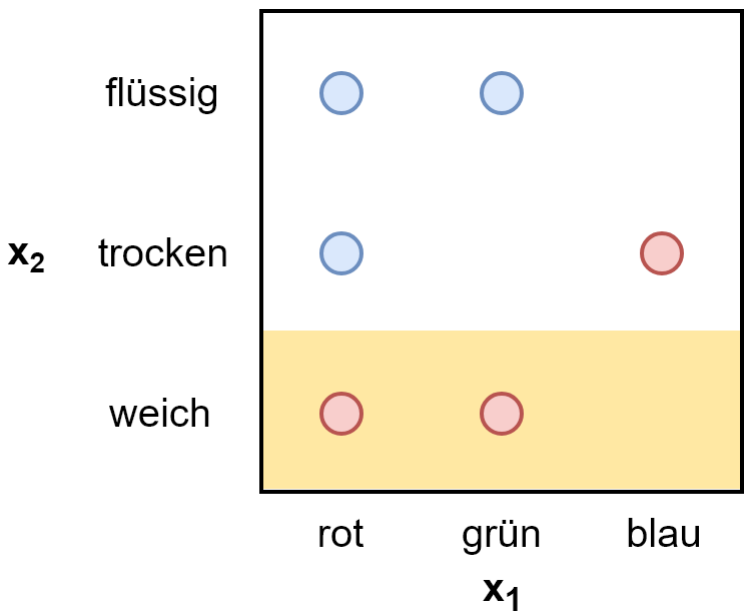


$$i(L) = 1$$
$$i(R) = 1$$
$$\Delta i(N) = \mathbf{0}$$

Entscheidungsbäume

Fragenauswahl und Knotenunreinheit

Frage an der Wurzel $\mathbf{x}_2$ =weich:

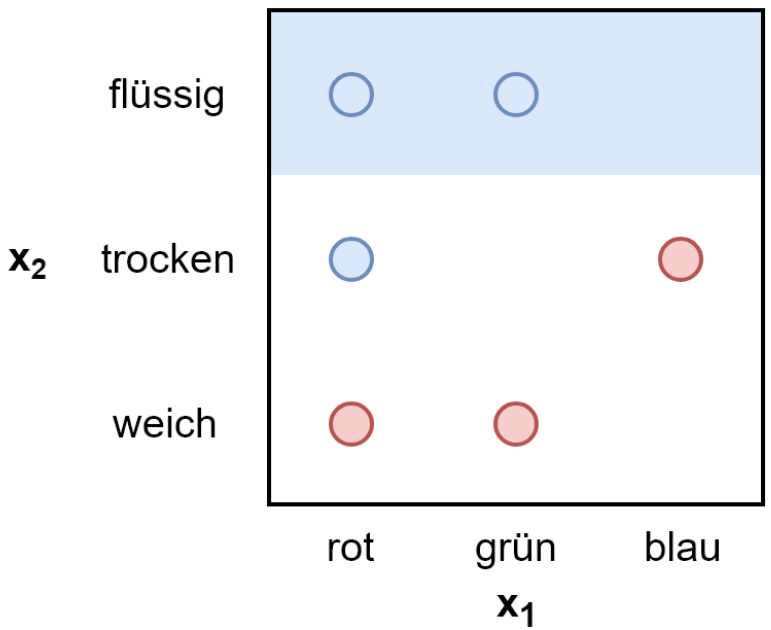


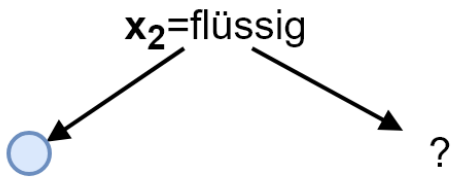
Analog zu $\mathbf{x}_2$ =flüssig?, also $\Delta i(N) = \mathbf{0.4591}$.

Entscheidungsbäume

Fragenauswahl und Knotenunreinheit

Wir wählen also als erste Frage $\mathbf{x}_2$ =flüssig, da sie neben $\mathbf{x}_2$ =weich den maximalen $\Delta i(N)$-Wert besitzt. Außerdem stellen wir fest, dass die Daten im linken Knoten rein sind und wir können damit direkt die Klasse positiv (blau) bestimmen.



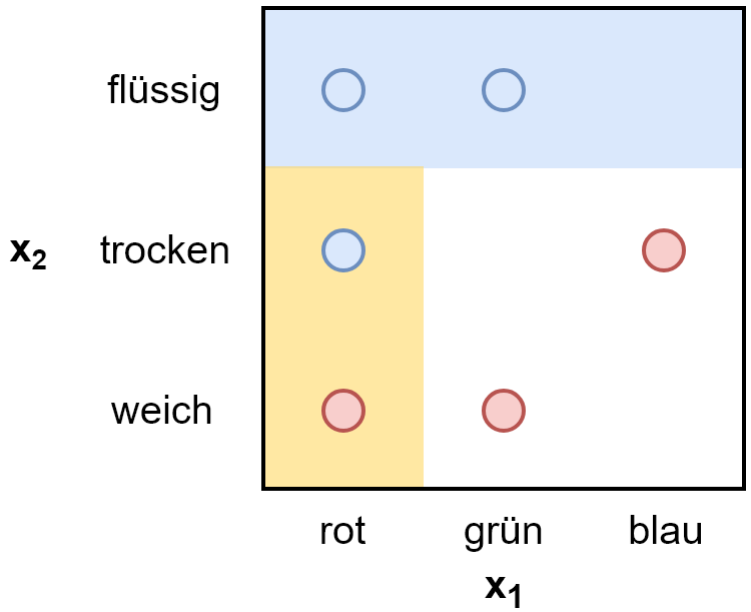


Die Unreinheit des rechten Knoten beträgt
$i(R) = -\frac{3}{4}\log_2\frac{3}{4} - \frac{1}{4}\log_2\frac{1}{4} \approx 0.8113.$

Entscheidungsbäume

Fragenauswahl und Knotenunreinheit

Nächste Frage $\mathbf{x}_1$ =rot:



$$i(L) = 1$$

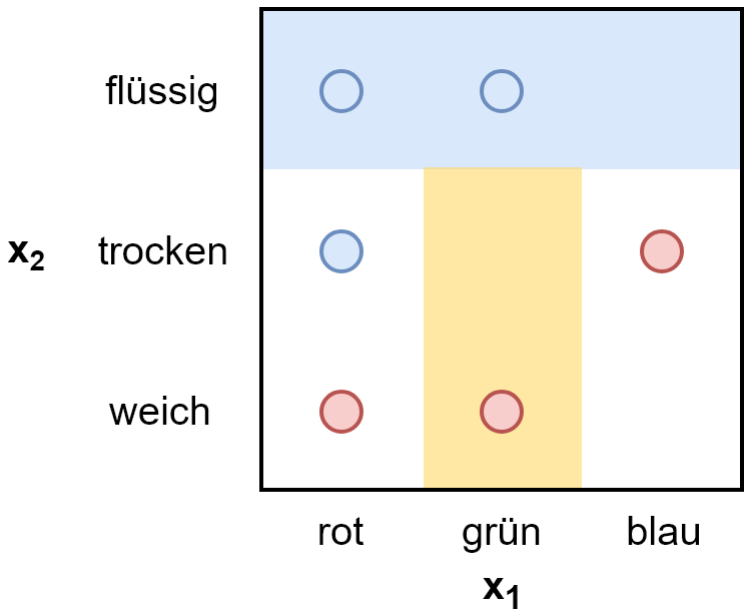
$$i(R) = 0$$

$$\Delta i(N) \approx 0.8113 - \frac{1}{2} \cdot 1 - \frac{1}{2} \cdot 0 \approx \mathbf{0.3113}$$

Entscheidungsbäume

Fragenauswahl und Knotenunreinheit

Nächste Frage $\mathbf{x}_1$ =grün:



$$i(L) = 0$$

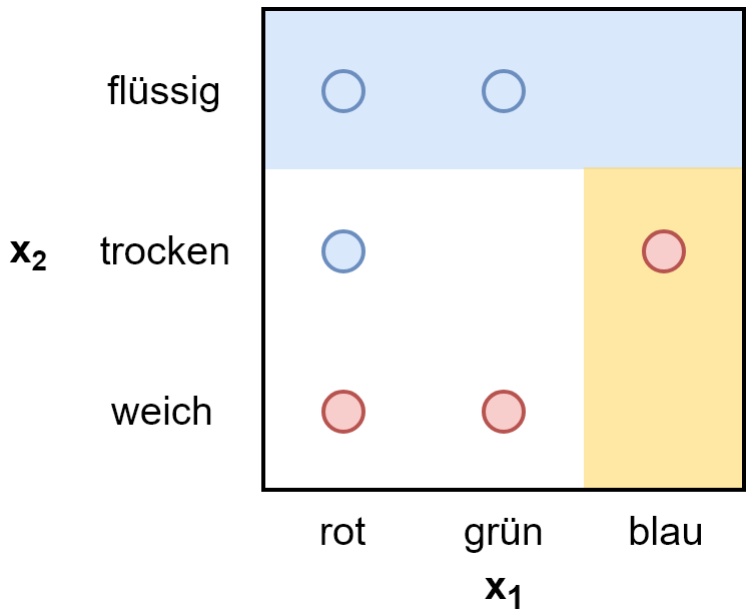
$$i(R) = -\frac{2}{3}\log_2\frac{2}{3} - \frac{1}{3}\log_2\frac{1}{3} \approx 0.9183$$

$$\Delta i(N) \approx 0.8113 - \frac{1}{4}\cdot 0 - \frac{3}{4}\cdot 0.9183 \approx \mathbf{0.1226}$$

Entscheidungsbäume

Fragenauswahl und Knotenunreinheit

Nächste Frage $\mathbf{x}_1$ =blau:



$$i(L) = 0$$

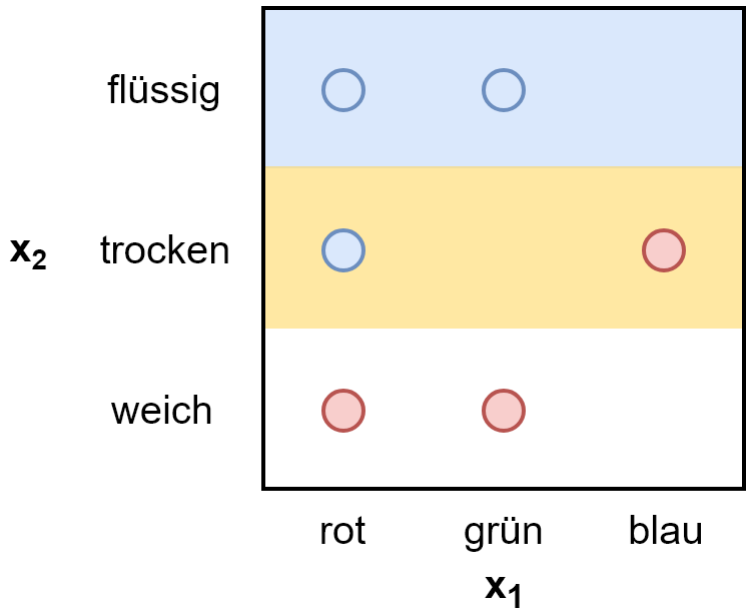
$$i(R) = -\frac{2}{3}\log_2\frac{2}{3} - \frac{1}{3}\log_2\frac{1}{3} \approx 0.9183$$

$$\Delta i(N) \approx 0.8113 - \frac{1}{4}\cdot 0 - \frac{3}{4}\cdot 0.9183 \approx \mathbf{0.1226}$$

Entscheidungsbäume

Fragenauswahl und Knotenunreinheit

Nächste Frage $\mathbf{x}_2$ =trocken:



$$i(L) = 1$$

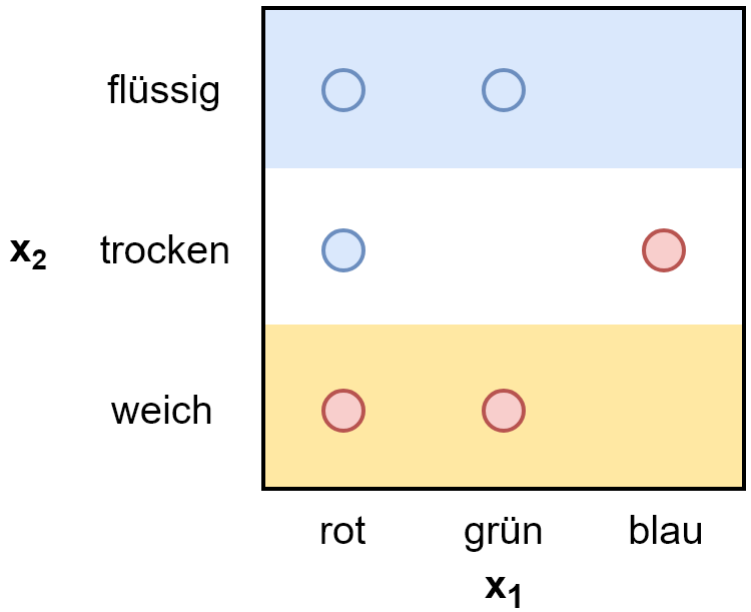
$$i(R) = 0$$

$$\Delta i(N) \approx 0.8113 - \frac{1}{2} \cdot 1 - \frac{1}{2} \cdot 0 \approx \mathbf{0.3113}$$

Entscheidungsbäume

Fragenauswahl und Knotenunreinheit

Nächste Frage $\mathbf{x}_2$ =weich:



$$i(L) = 0$$

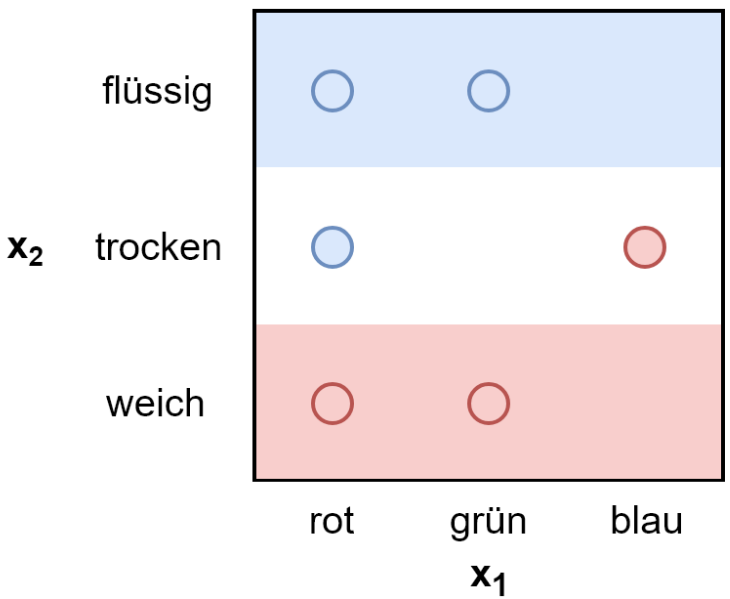
$$i(R) = 1$$

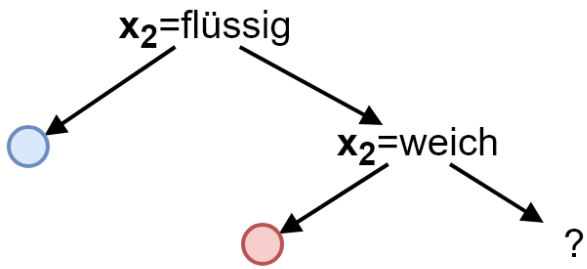
$$\Delta i(N) \approx 0.8113 - \frac{1}{2} \cdot 0 - \frac{1}{2} \cdot 1 \approx \mathbf{0.3113}$$

Entscheidungsbäume

Fragenauswahl und Knotenunreinheit

Wir wählen also als nächstes Frage $\mathbf{x}_2$ =weich, da sie neben $\mathbf{x}_2$ =trocken den maximalen $\Delta i(N)$-Wert besitzt. Außerdem stellen wir fest, dass die Daten im linken Knoten rein sind und wir können damit direkt die Klasse negativ (rot) bestimmen.



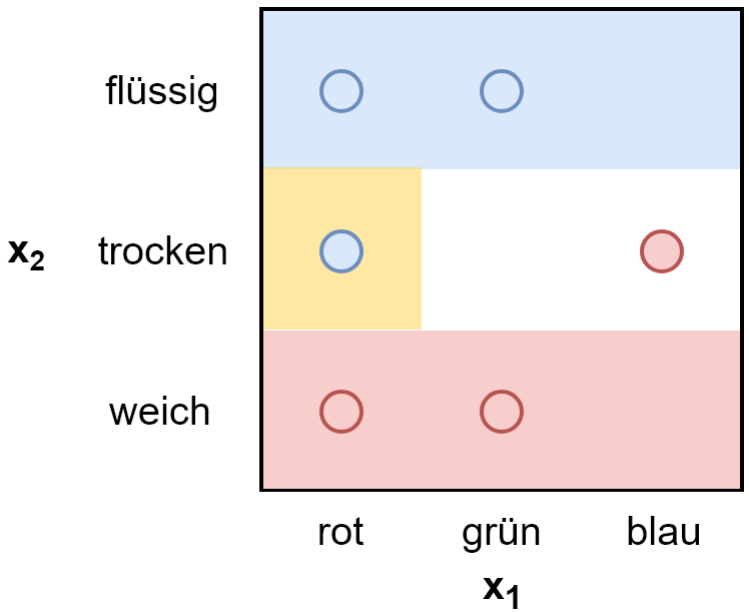


Die Unreinheit des rechten Knoten beträgt $i(R) = 1$.

Entscheidungsbäume

Fragenauswahl und Knotenunreinheit

Nächste Frage $\mathbf{x}_1$ =rot:

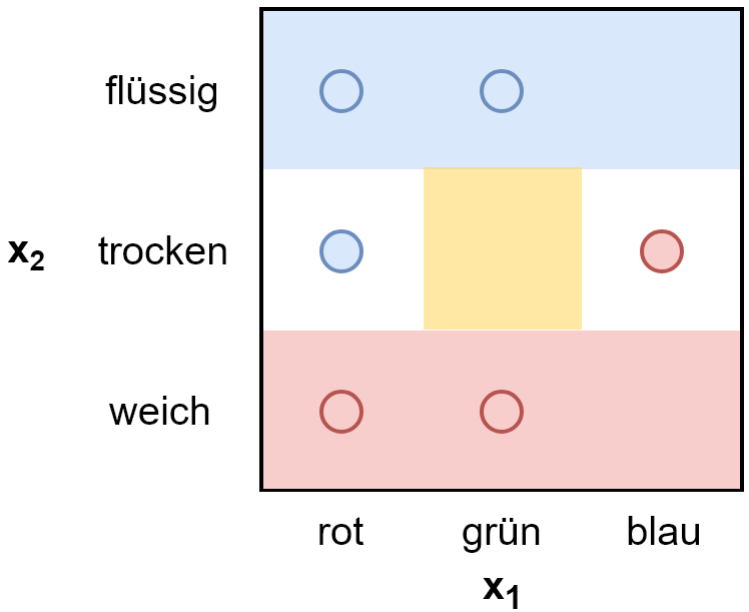


$$i(L) = 0$$
$$i(R) = 0$$
$$\Delta i(N) = 1 - 0 - 0 = \mathbf{1}$$

Entscheidungsbäume

Fragenauswahl und Knotenunreinheit

Nächste Frage $\mathbf{x}_1$ =grün:

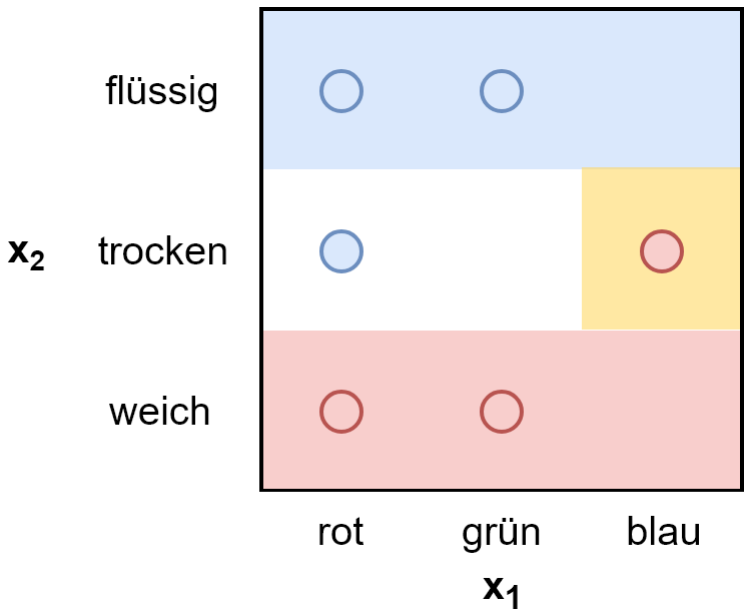


$$i(L) = 0$$
$$i(R) = 1$$
$$\Delta i(N) = 1 - 0 - 1 = \mathbf{0}$$

Entscheidungsbäume

Fragenauswahl und Knotenunreinheit

Nächste Frage $\mathbf{x}_1$ =blau:

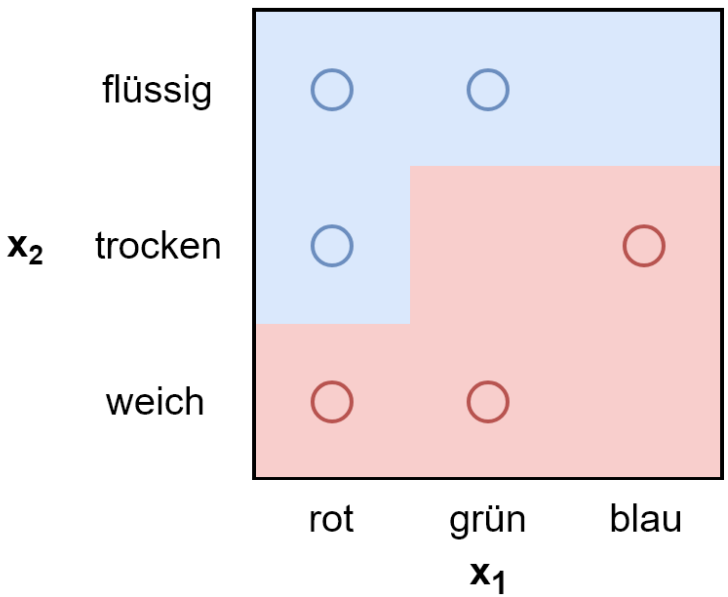


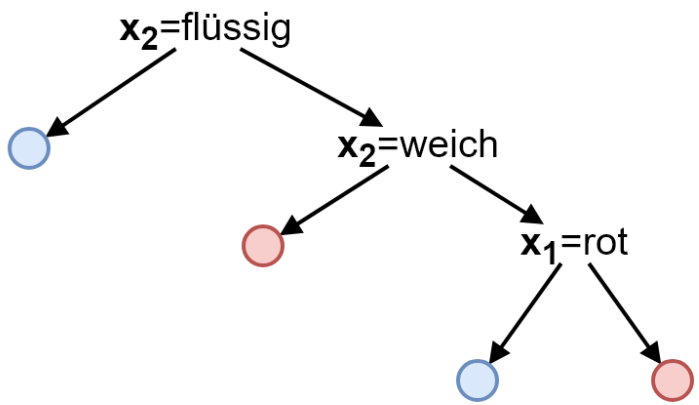
$$i(L) = 0$$
$$i(R) = 0$$
$$\Delta i(N) = 1 - 0 - 0 = \mathbf{1}$$

Entscheidungsbäume

Fragenauswahl und Knotenunreinheit

Wir wählen also als nächstes Frage $\mathbf{x}_1$ =rot, da sie neben $\mathbf{x}_1$ =blau den maximalen $\Delta i(N)$-Wert besitzt. Außerdem stellen wir fest, dass die Daten im linken Knoten rein sind und wir können damit direkt die Klasse positiv (blau) bestimmen. Auch der rechte Knoten ist nun rein negativ (rot) und wir sind fertig.





Entscheidungsbäume

Fragenauswahl und Knotenunreinheit

Im Falle von reellwertigen Features, etwa $\mathcal{X} = \mathbb{R}^d$ können und sollten die entsprechenden Fragen passend gewählt werden. Oftmals werden auch Linearkombinationen erlaubt, was die Baumkomplexität deutlich verringert.

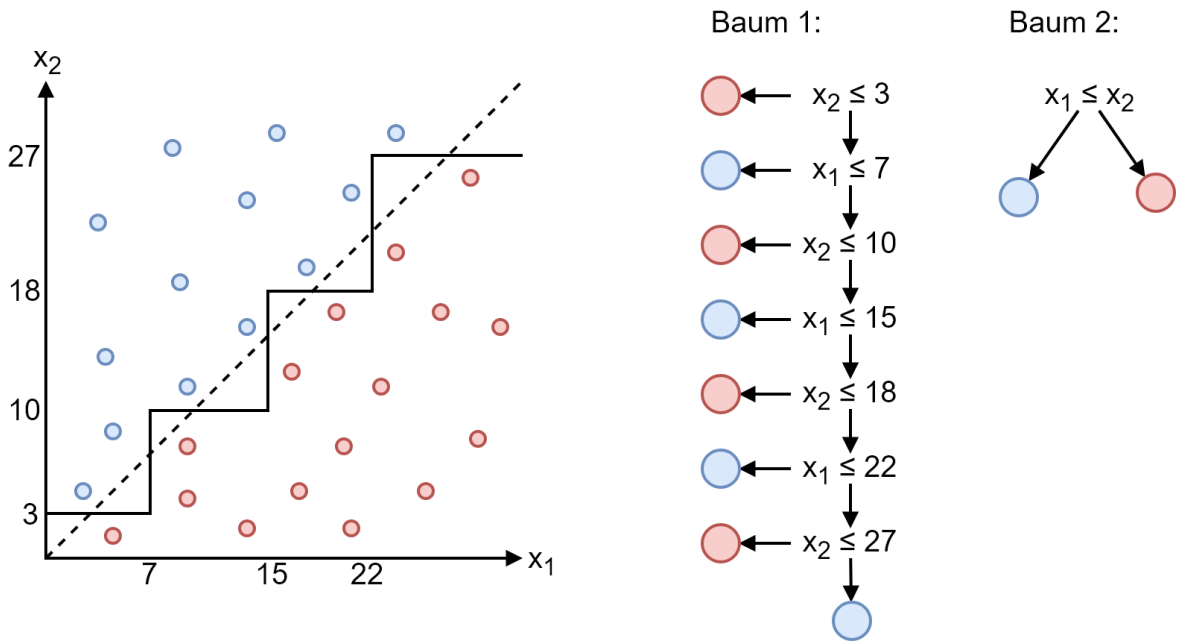


Abbildung 5: Baum 2 erlaubt Linearkombinationen der Features $\mathbf{x}_1$ und $\mathbf{x}_2$ und wird dadurch einfacher als Baum 1.

Entscheidungsbäume

Fragenauswahl und Knotenunreinheit

- Der vorgestellte Algorithmus zur Selektion von Fragen durch Maximierung des Unreinheitsrückgangs ist lokal und damit greedy. Das bedeutet, es gibt keine Garantie, dass die Folge von lokalen Festlegungen zu einem globalen Optimums bzgl der Baumgröße führt.
- Obwohl es eigentlich gilt die Missclassification Impurity zu reduzieren ist oft die Gini Unreinheit vorzuziehen, da sie vorausschauender ist.
- In der Praxis ist jedoch meist die Wahl des Maßes für Unreinheit nebensächlich. Wichtiger sind Kriterien, wann der Baumaufbau gestoppt oder wann gestutzt werden soll.

Entscheidungsbäume

Stopkriterien

Problem: Baut man den Baum an allen Stellen immer bis zur minimalen Unreinheit wird man mit sich mit sehr großer Wahrscheinlichkeit im Regime der Überanpassung befinden. Im Extremfall z.B. besteht jedes Blatt aus einem einzigen Datenpunkt, der Baum hat also die Trainingsdaten auswendig gelernt.

Entscheidungsbäume

Stopkriterien

Testfehlerkriterium: Eine allgemeine Möglichkeit ist es, einen Split der Daten $\mathcal{D} = \mathcal{T} \dot{\cup} \mathcal{V}$ in Trainings- und Testdaten durchzuführen und nach jedem Schritt im Baumaufbau den Fehler auf den Testdaten zu protokollieren. Schließlich verwendet man die Splittingtiefe pro Baumteil, welcher zum minimalen Testfehler geführt hatte.

Entscheidungsbäume

Stopkriterien

Mindestreduktion: Eine weitere direkte Möglichkeit ist es, von weiteren Splits abzusehen, wenn der beste Split nicht zu einer Reduktion der Unreinheit größer als ein zu wählender Hyperparameter $\beta \in \mathbb{R}_{\geq 0}$ führt, also wenn

$$\max \Delta i(N) < \beta.$$

- Hier können alle Daten für das Lernen verwendet werden und Blätter können auf allen Ebenen liegen (vorteilhaft bei hoher Diversität in den Feature-Dimensionen).
- Jedoch ist die optimale Wahl des Hyperparameters β meist nicht ersichtlich.

Entscheidungsbäume

Stopkriterien

Mindestgröße: Hier wird das Splitting beendet, sobald eine Mindestgröße der Datenmenge entweder absolut (Anzahl der Datenpunkte, z.B. 50) oder relativ (Anteil an der Trainingsdatenmenge, z.B. 3%) unterschritten wird.

- Diese Methode hat den Vorteil, dass sich die Partitionsgrößen bei ungleich verteilten Daten anpassen, d.h. man erhält kleine Partitionen bei dichten Daten und große Partitionen bei spärlichen Daten.

Entscheidungsbäume

Stopkriterien

Globale Kriterien: Der Baum wird gesplittet solange, bis ein Minimum bzgl. eines globalen Kriteriums

$$\alpha \cdot \text{Größe} + \sum_{\text{Blätter}} i(N)$$

eingenommen wird. Als Maß für die Größe kann z.B. die Anzahl der Knoten und/oder Kanten des Baums verwendet werden. Der Hyperparameter $\alpha \in \mathbb{R}_{\geq}$ wägt zwischen der Größe und der Unreinheit der Blätter ab.

- Im Falle der Entrope als Unreinheitsmaß hat der Term $\sum_{\text{Blätter}} i(N)$ eine intuitive Bedeutung als Maß für die Gesamtunsicherheit (in Bits) der Trainingsdaten innerhalb der Baumrepräsentation.
- Auch hier ist die optimale Wahl des Hyperparameters α meist nicht ersichtlich.

Entscheidungsbäume

Stopkriterien

Hypothesentest: Hier wird getestet, ob ein Split sich signifikant von einem zufälligen Split unterscheidet. Angenommen n Datenpunkte befinden sich in einem Knoten wovon tatsächlich n_1 Punkte zu Klasse y_1 und n_2 Punkte zu Klasse y_2 gehören und der zu untersuchende Split packt den Bruchteil pn der Punkte in den linken Ast und $(1-p)n$ in den rechten Ast. Ein zufälliger Split würde pn_1 der Klasse y_1 Punkte und pn_2 der Klasse y_2 Punkte nach links packen. Man kann nun mit der Chi-Quadrat-Statistik

$$\mathcal{X}^2 = \sum_{i=1}^{2} \frac{(n_{iL} - pn_i)^2}{pn_i}$$

wobei n_{iL} die Anzahl der vom Split nach links gepackten Punkte ist, bei vorher gewählten Konfidenz mit Hilfe der $\mathcal{X}^2$-Wertetabelle überprüfen, ob man einen zufälligen Split ausschließen kann ($\mathcal{X}^2 >$ Tabellenwert).

Entscheidungsbäume

Stutzen

Stopkriterien leiden unter dem Horizonteffekt, d.h. bei einem zu früh entschiedenen Stop werden evtl. spätere, global gesehen jedoch gute Split nicht erkannt.
Beim Stutzen wird der Baum komplett aufgebaut (Blätter haben minimale Unreinheit) und anschließend werden Teile des Baums entfernt.

- Die Vorteile sind, dass der Horizonteffekt vermieden wird, alle Trainingsdaten verwendet werden und der Baum potentiell interpretierbarer wird.
- Der Nachteil ist, dass die Berechnungskosten höher sind, als im Falle von Stopkriterien.

Entscheidungsbäume

Stutzen

Verschmelzen: Bei dieser Strategie werden sukzessive Splits rückgängig gemacht. Dies geschieht durch das Verschmelzen von Knoten oft angefangen von den Blättern, die nur einen kleinen Beitrag zur Unreinheit leisten.
Regeln: Jedes Blatt kann durch eine korrespondierende aussagenlogische Formel bzgl. der Features beschrieben werden. Durch Betrachtung dieser Liste können potentiell auf der Formelebene Vereinfachungen gefunden werden.

Entscheidungsbäume

Klassenzuordnung

Wenn keine Unreinheit in einem Blatt herrscht, so ist die Klassenzuordnung klar. Durch Stopkriterien oder Stutzen können sich in einem Blatt Datenpunkte verschiedener Klassen befinden. In diesem Fall wird meist die Klasse mit den meisten Datenpunkten gewählt.

Entscheidungsbäume

Unvollständige Daten

Manchmal kommt es vor, dass nicht für alle Trainings- bzw. Testdaten alle Features bekannt sind. Im Obstbeispiel z.B. könnte für ein Exemplar die Farbe und die Größe unbekannt sein, also $\mathbf{x} = (?, \text{rund}, ?, \text{süß}) \in \mathcal{D}$. In solchen Fällen

- kann man diese Daten aus den Trainingsdaten entfernen (nur sinnvoll wenn $|\mathcal{D}|$ groß)
- oder (besser) man passt die Berechnung der Unreinheit an die vorhandenen Daten je Feature/Frage an.
- In jedem Fall muss der Entscheidungsbaum an jedem Knoten Alternativen bereit stellen, falls ein Feature nicht vorhanden ist.

Entscheidungsbäume

References

R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern classification (2nd edition)*, Wiley-Interscience, New York, NY, USA, 2000.