Memory-Based Learning

Caroline Sporleder

Computational Linguistics
Universität des Saarlandes

Sommersemester 2011

09.06.2011

Lazy vs. Eager Learner (1)

Generalisierung	Abstraktion	
	+	-
+	DTs, NB, Candidate Elimination	Memory-Based Learning
-	???	Table Look-Up

Lazy vs. Eager Learner (2)

Eager Learners

abstrahieren schon in der Trainingsphase von den Trainingsdaten, d.h. sie lernen Regeln, Bäume etc.
Der Lernaufwand liegt größtenteil in der Trainingsphase.

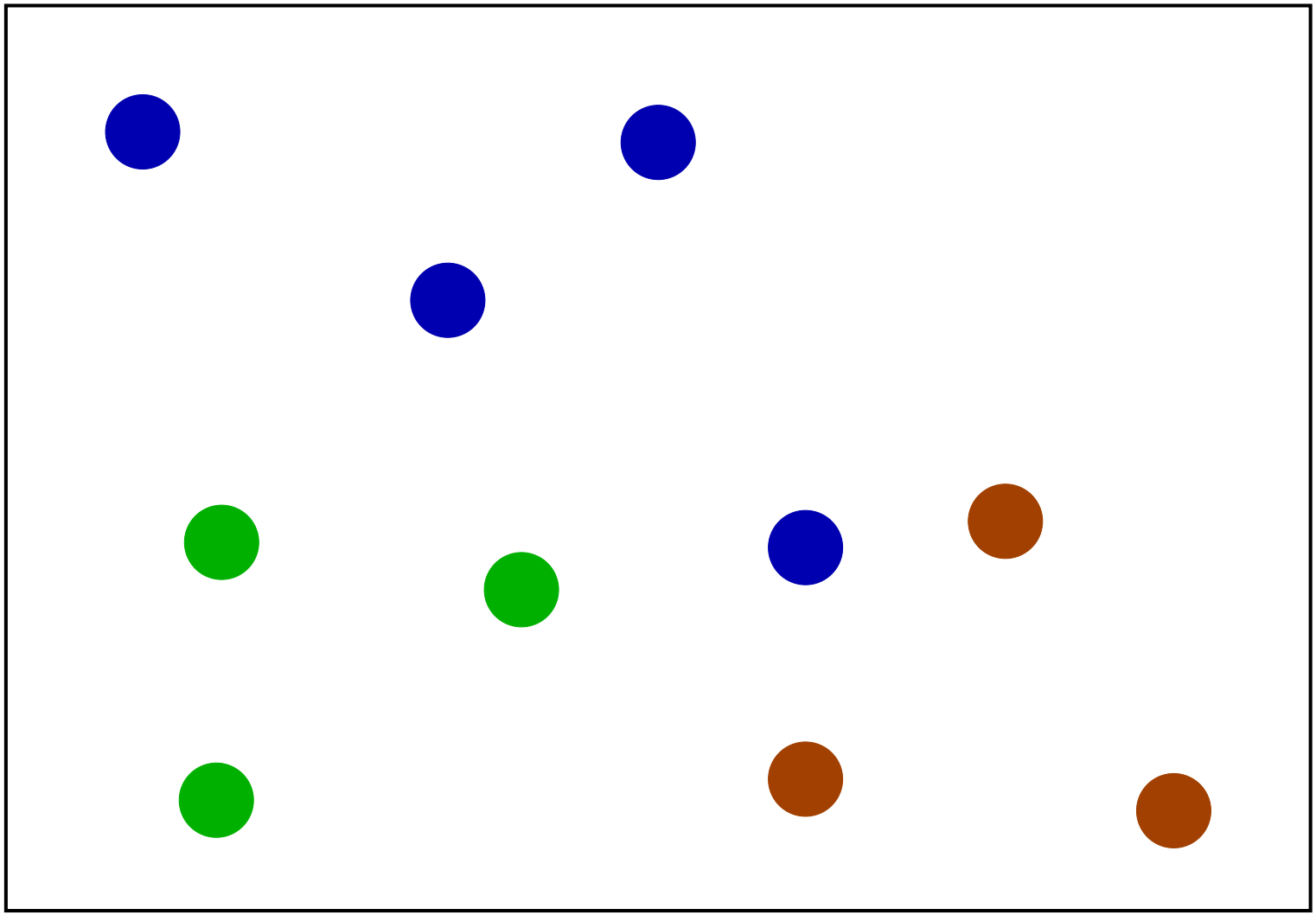
Lazy Learner

tun in der Trainingsphase nichts weiter als sich die Trainingsdaten zu merken. Während der Testphase vergleichen Lazy Learners die Testinstanzen dann mit den Trainingsinstanzen, um die korrekte Ausgabeklasse zu finden.
Der Lernaufwand liegt größtenteil in der Testphase.

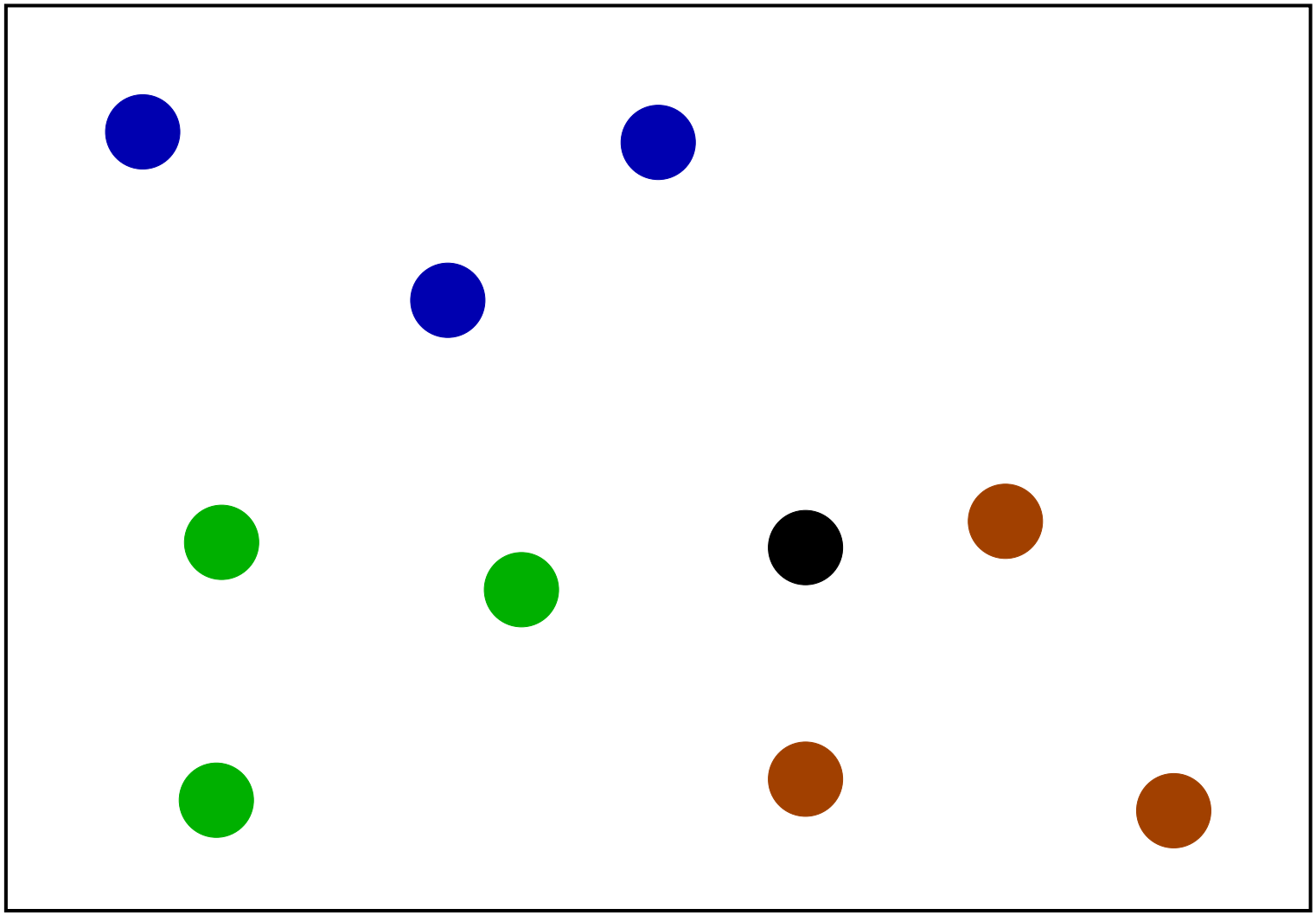
Abstraktion

. . . bedeutet meist, nur die groben Muster zu behalten (d.h. z.B. in Regeln zu kodieren) und Idiosynkrasien (outlier) zu ignorieren (vgl. Occam's Razor).

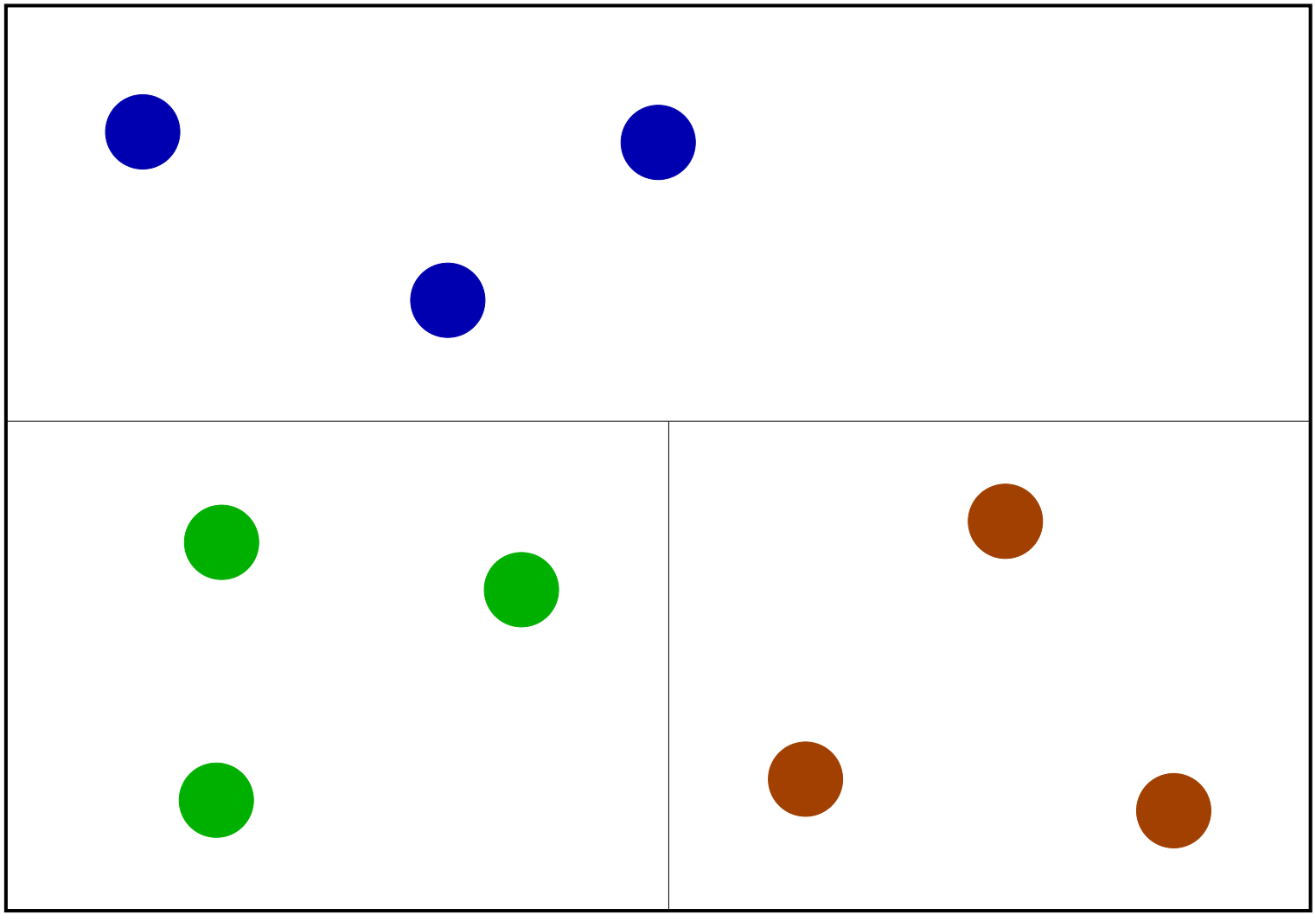
Abstraktion beim Eager Learning



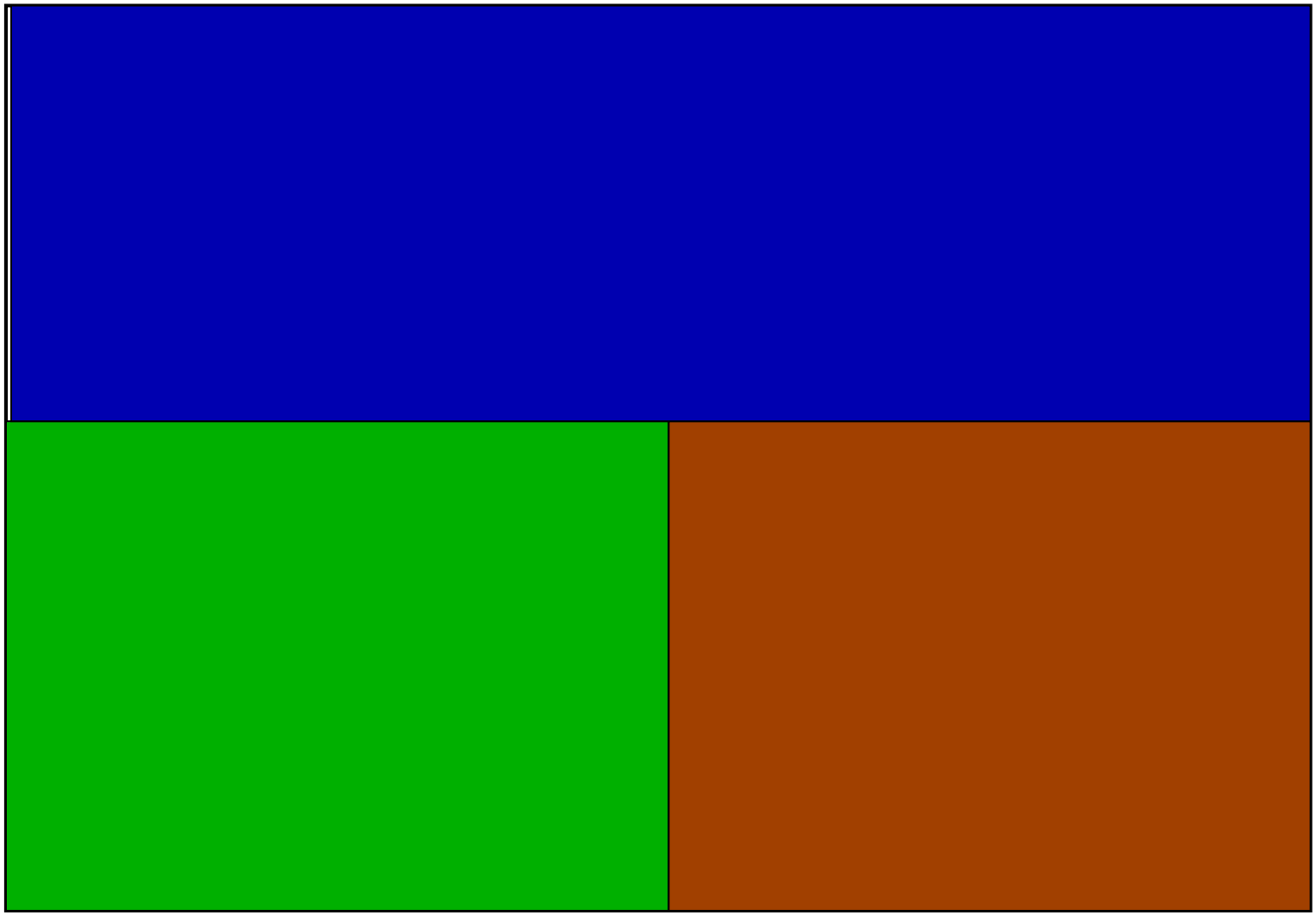
Abstraktion beim Eager Learning



Abstraktion beim Eager Learning



Abstraktion beim Eager Learning



Memory-Based Learning

- eine Art von lazy learning
- neue Instanzen werden anhand ihrer **Ähnlichkeit** mit den Trainingsinstanzen klassifiziert
  $\Rightarrow$ der Wahl eines guten Ähnlichkeitsmaßes kommt zentrale Bedeutung zu
- in den 50ern für die Mustererkennung entwickelt (K-NN Algorithmus)
- später in der (Computer)Linguistik aufgegriffen (Royal Skousen, Analogical Modeling of Language)
  $\Rightarrow$ **Sprache enthält viele Idiosynkrasien und Subregularitäten, die für eine korrekte Klassifizierung oft wichtig sind**

K-Nearest Neighbour Algorithmus (1)

K-Nearest Neighbour (k-nn, KNN)

- aus der Mustererkennung
- ursprünglich nur für numerische Attribute

Grundidee

- identifiziere die Menge N der k-nächsten Nachbarn einer Testinstanz t
- weise t die Ausgabeklasse zu, die in N am häufigsten vertreten ist

K-Nearest Neighbour Algorithmus (2)

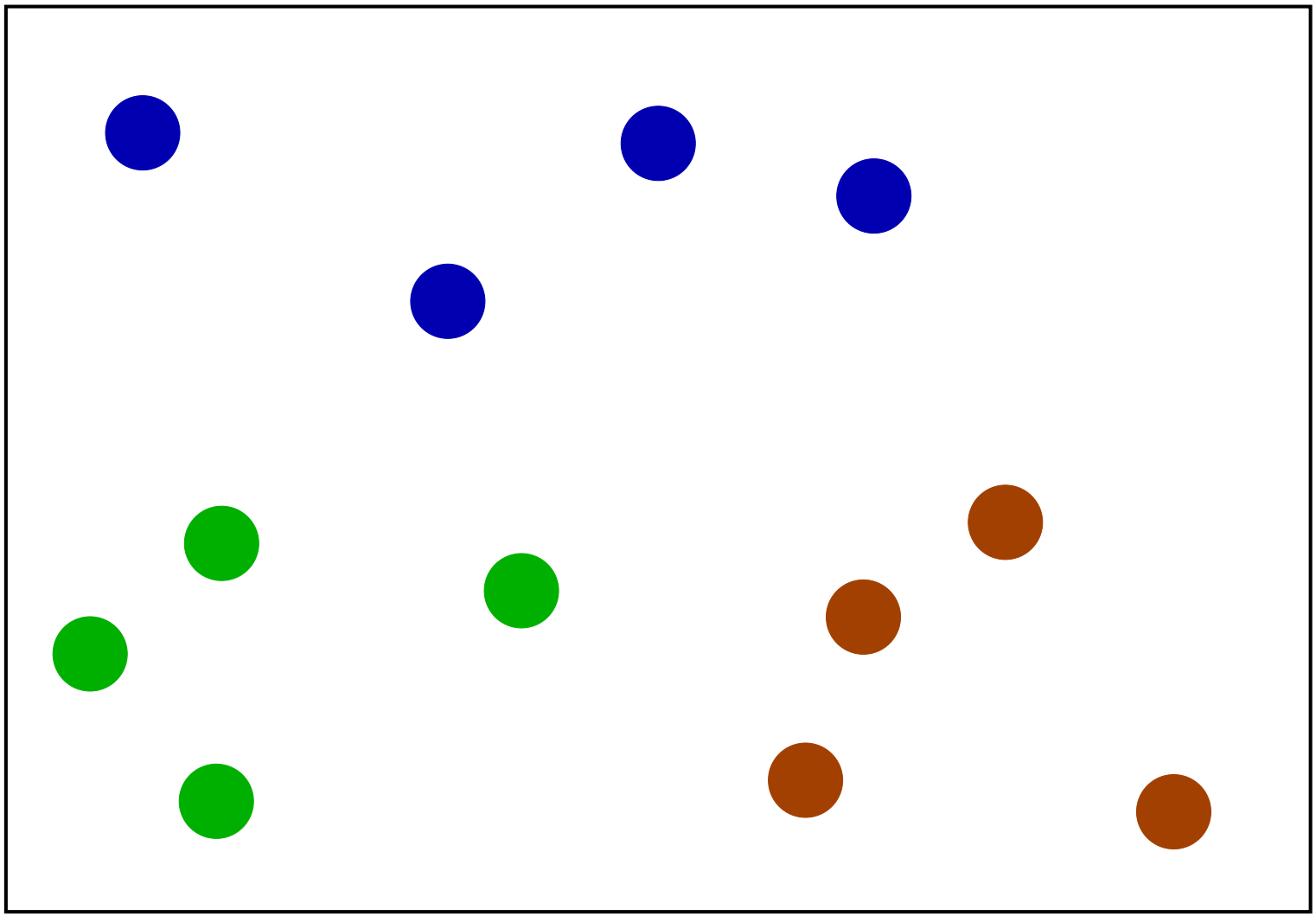
Ähnlichkeitsmaß

Euklidischer Abstand (numerische Attribute)

$\Delta(X, Y) = \sqrt{\Sigma_{i=1}^{n}(x_i - y_i)^2}$

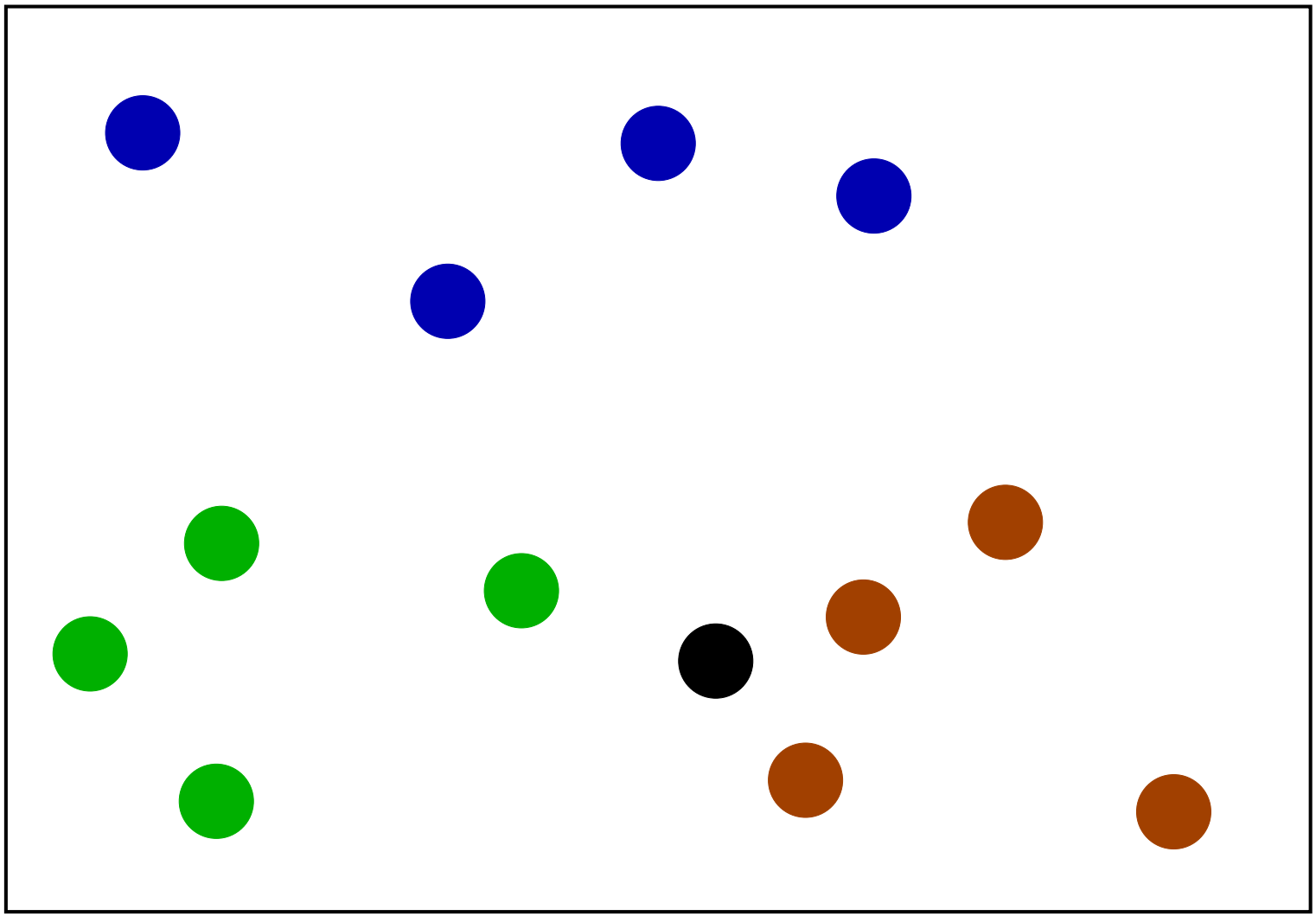
Beispiel 1: KNN Algorithmus

Für $k = 3$



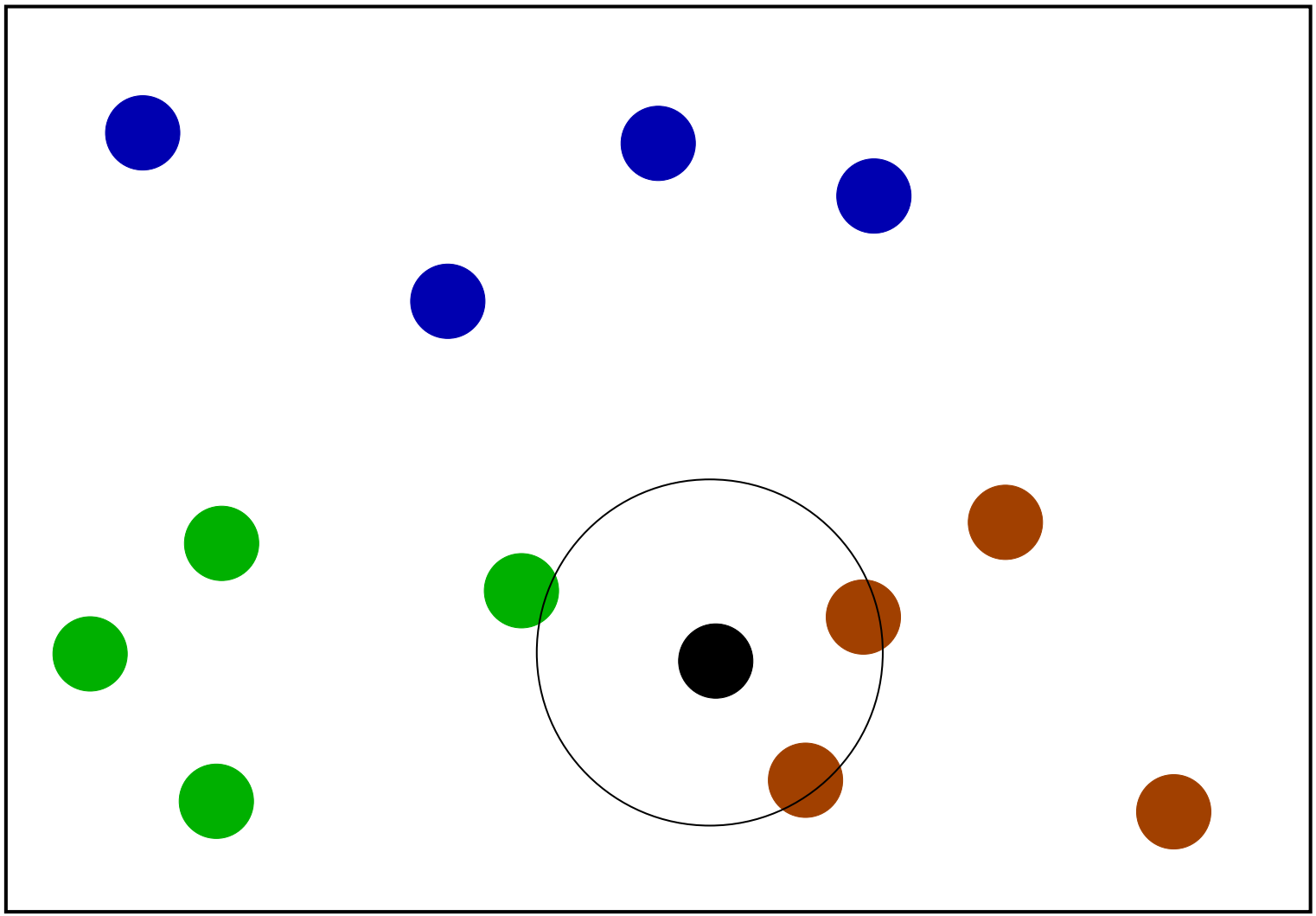
Beispiel 1: KNN Algorithmus

Für $k = 3$



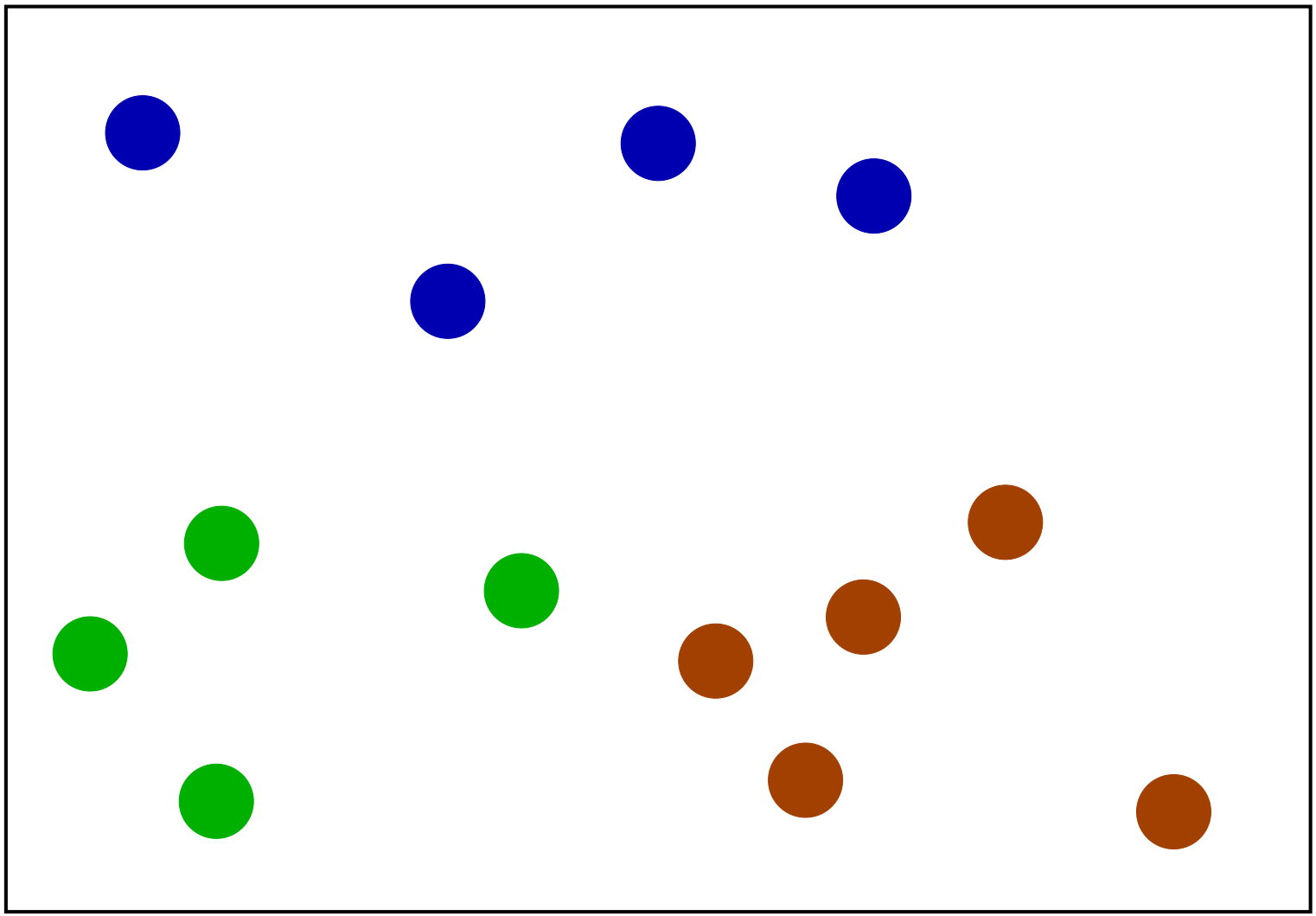
Beispiel 1: KNN Algorithmus

Für $k = 3$



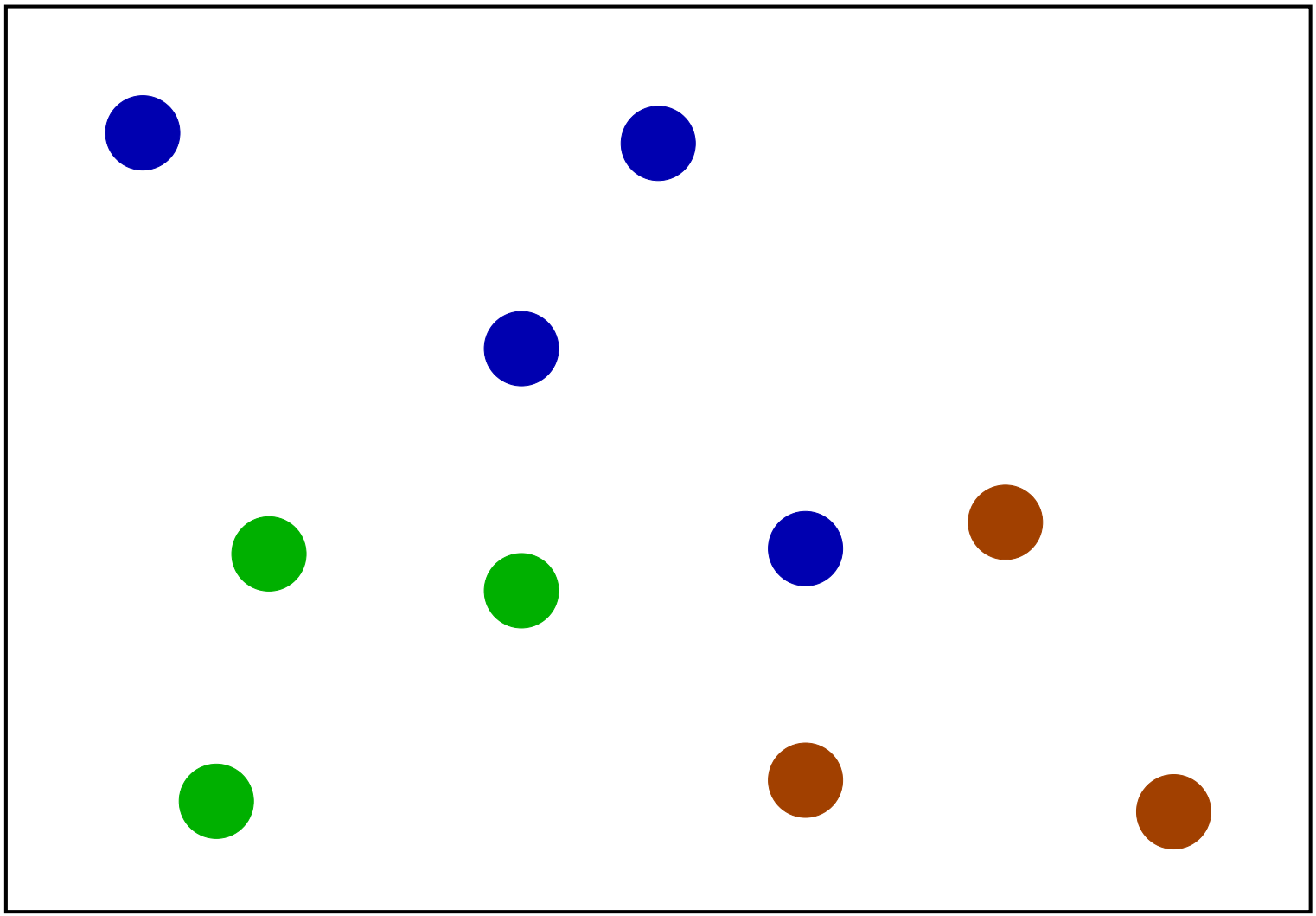
Beispiel 1: KNN Algorithmus

Für $k = 3$



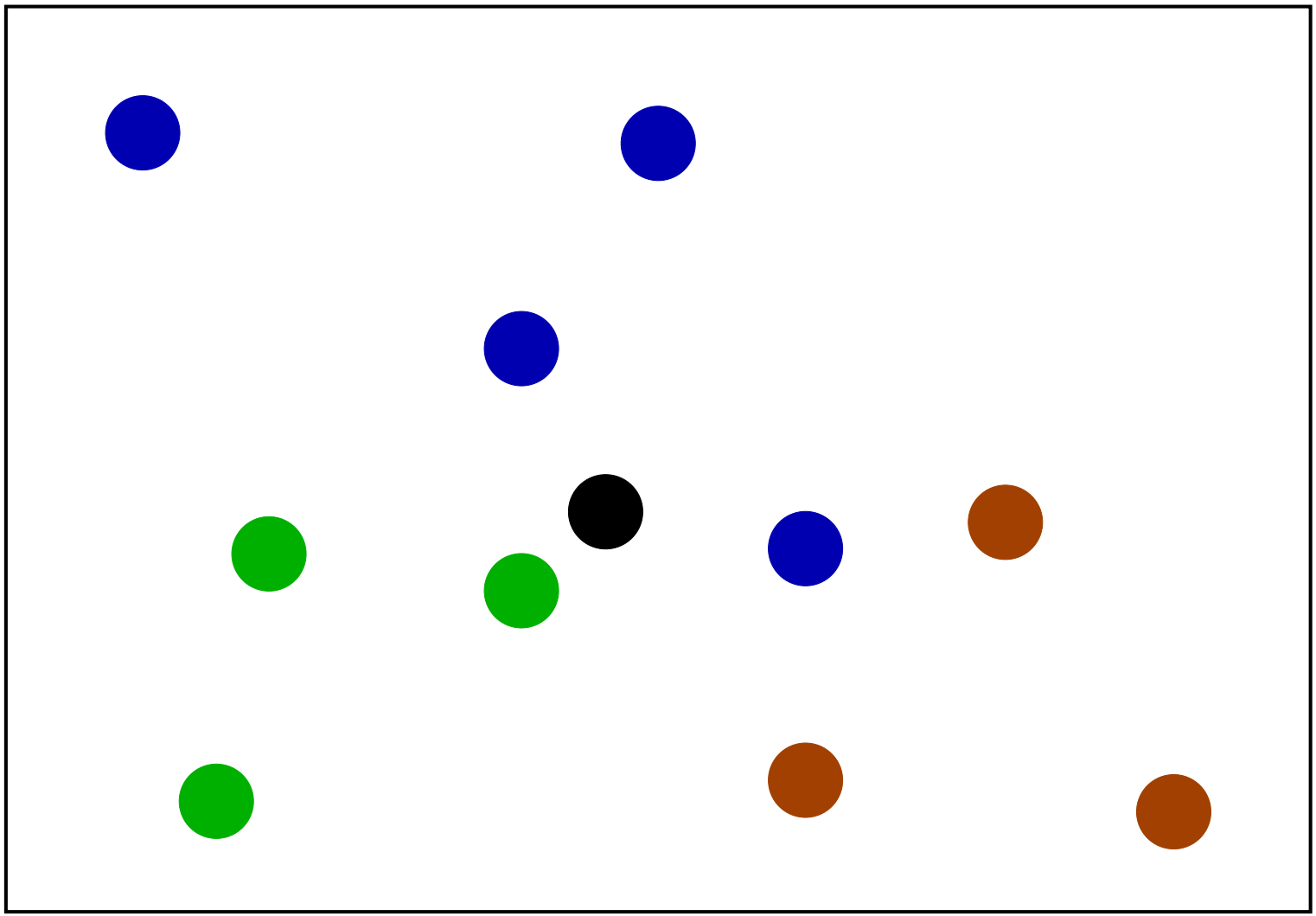
Beispiel 2: KNN Algorithmus mit Outliern

Für $k = 3$



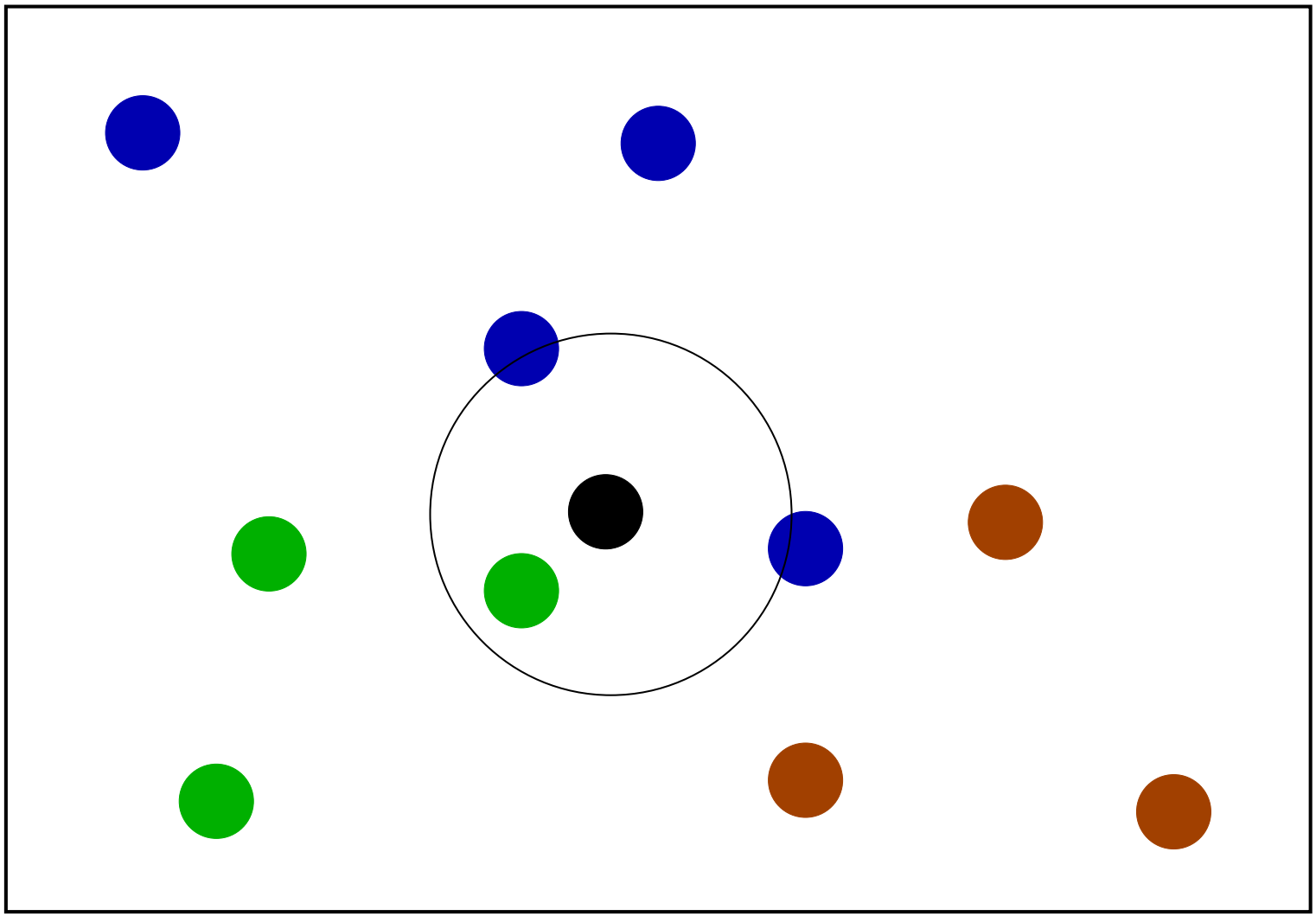
Beispiel 2: KNN Algorithmus mit Outliern

Für $k = 3$



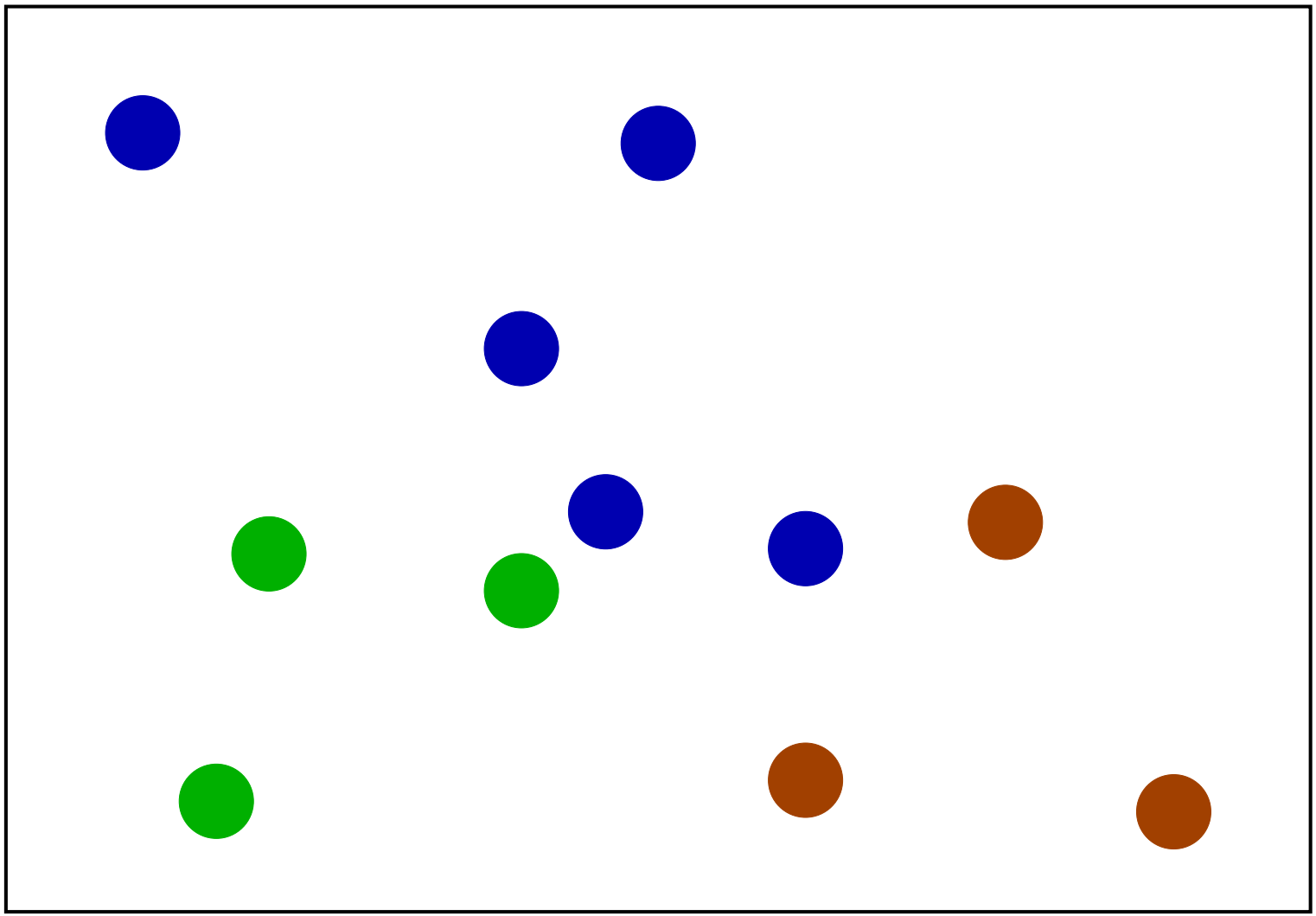
Beispiel 2: KNN Algorithmus mit Outliern

Für $k = 3$

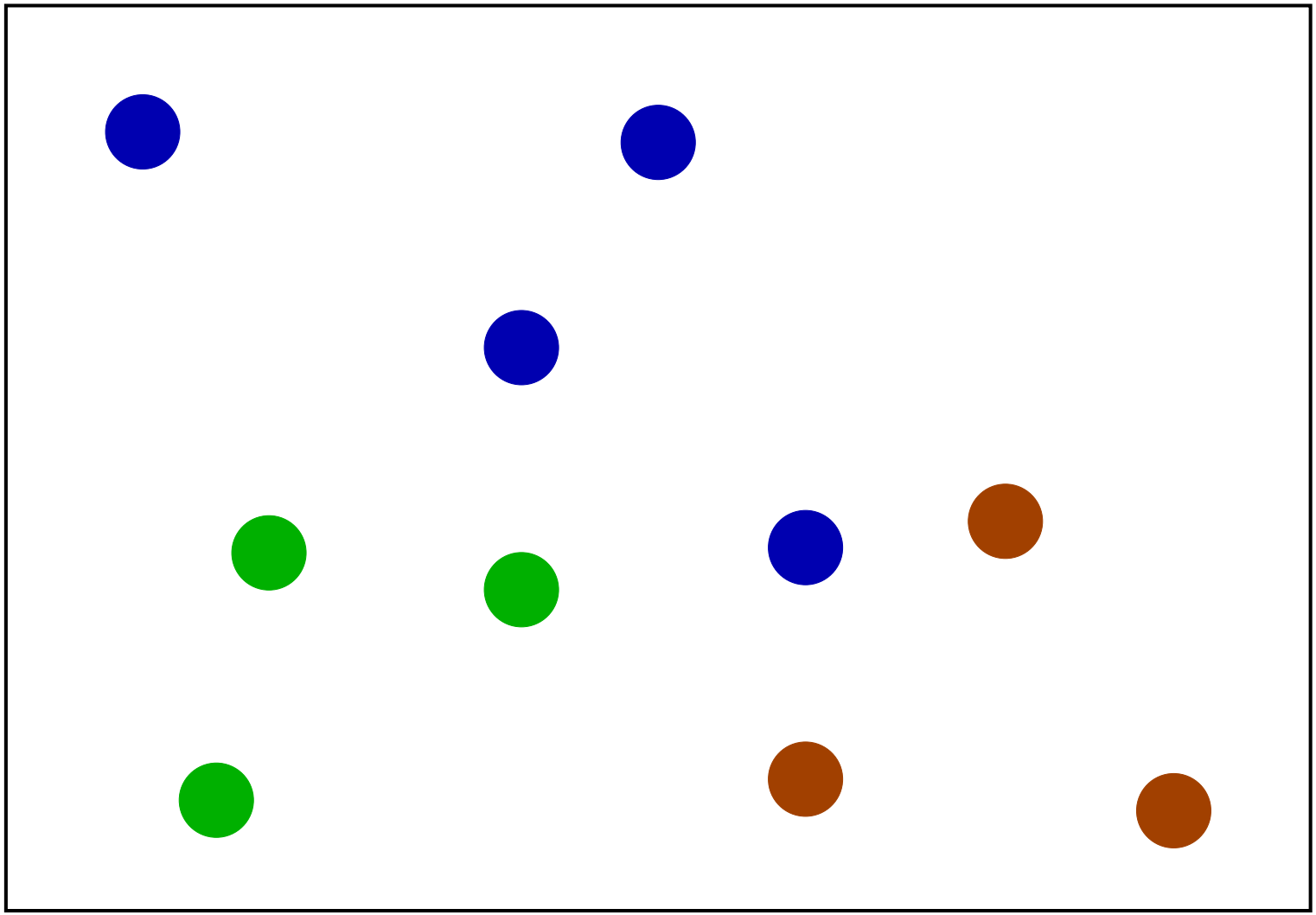


Beispiel 2: KNN Algorithmus mit Outliern

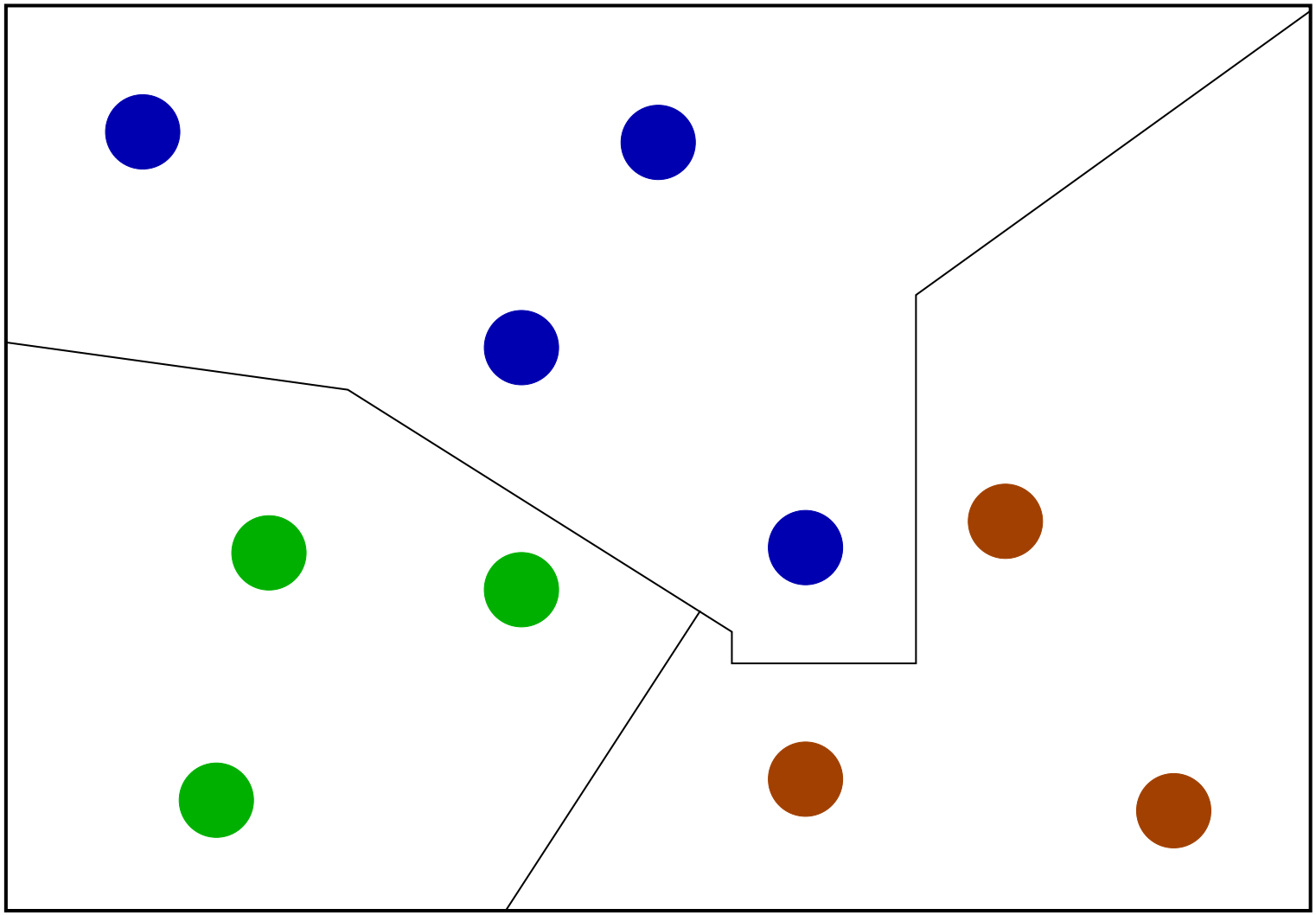
Für $k = 3$



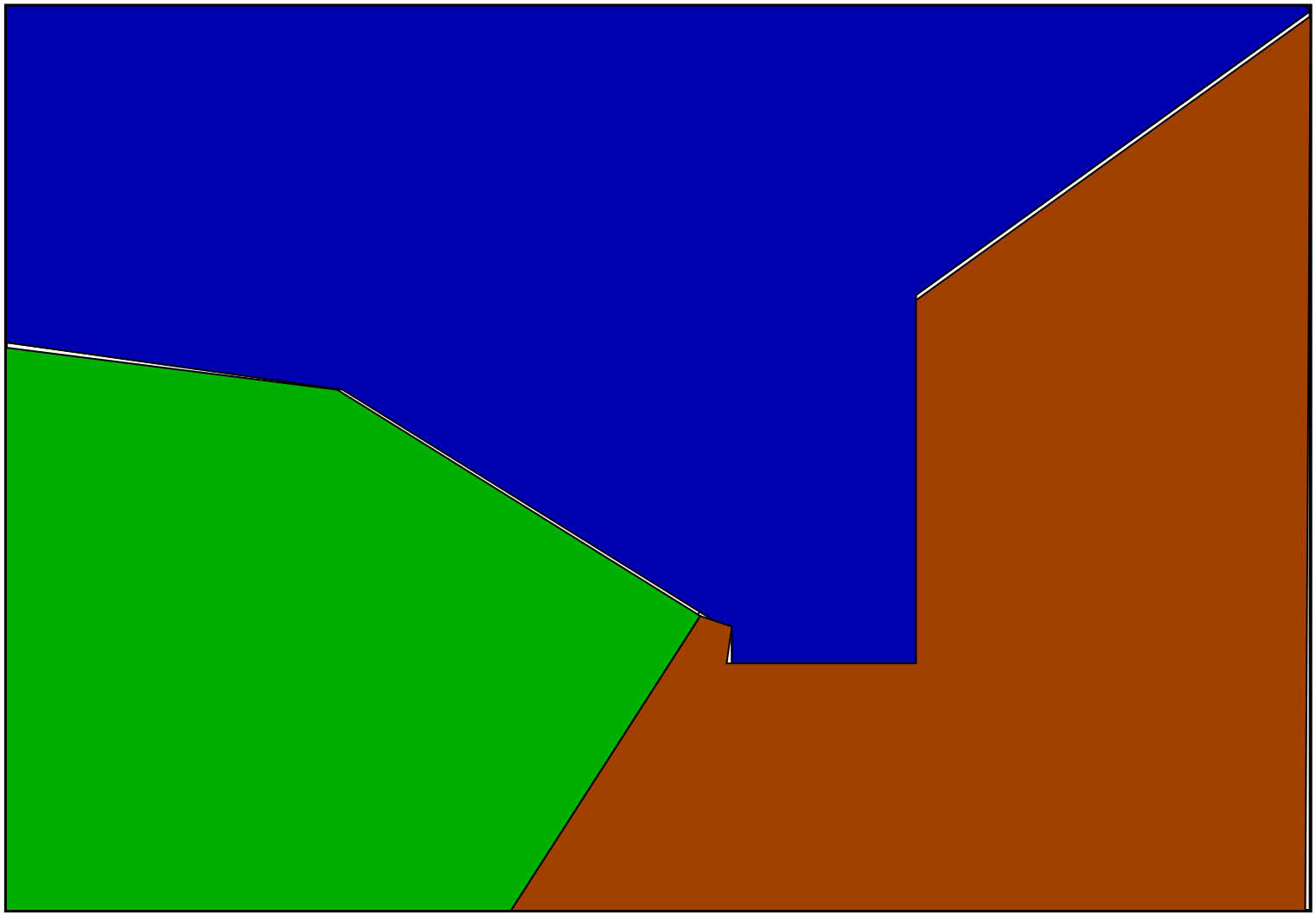
Beispiel: Entscheidungsgrenzen für MBL



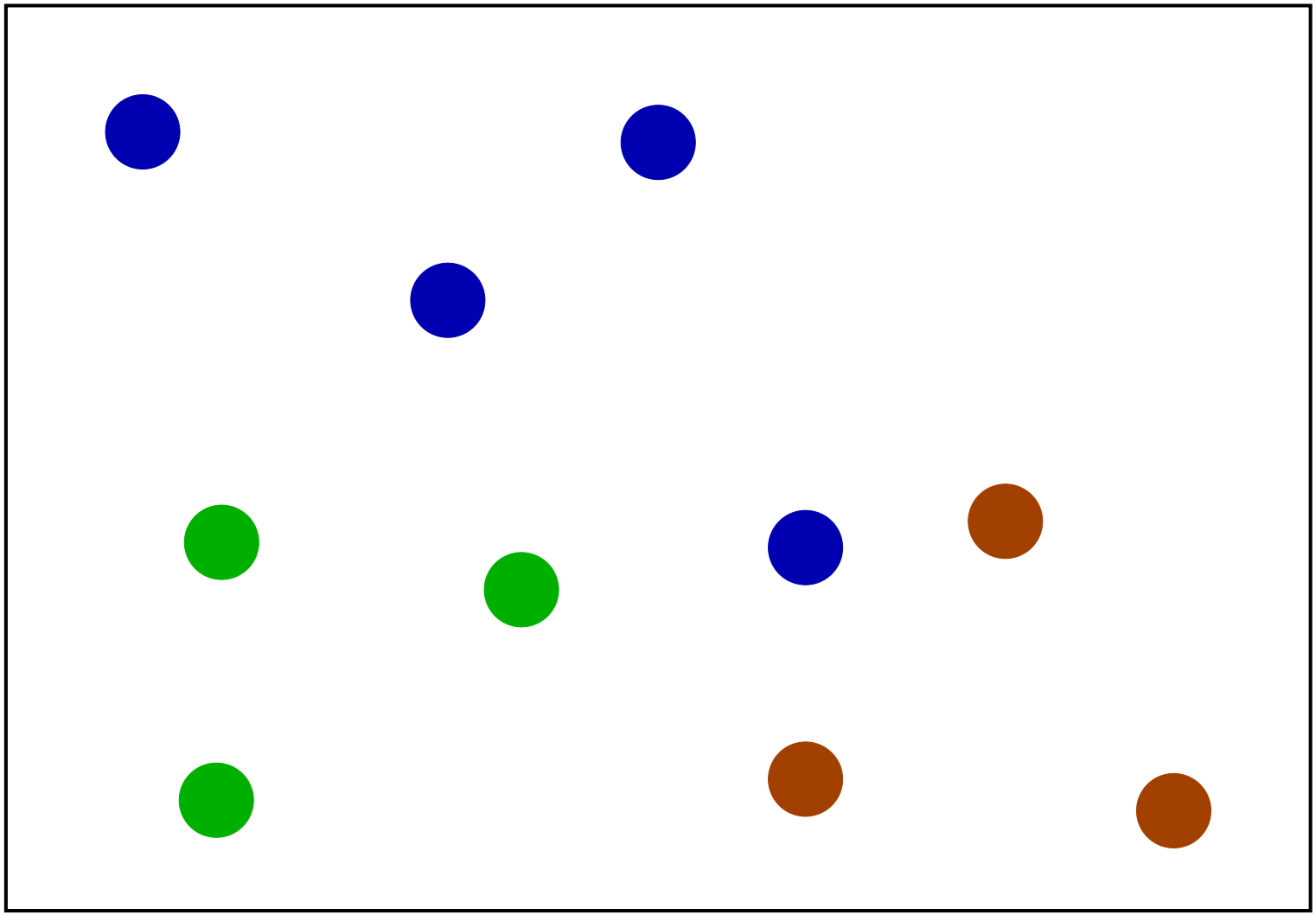
Beispiel: Entscheidungsgrenzen für MBL



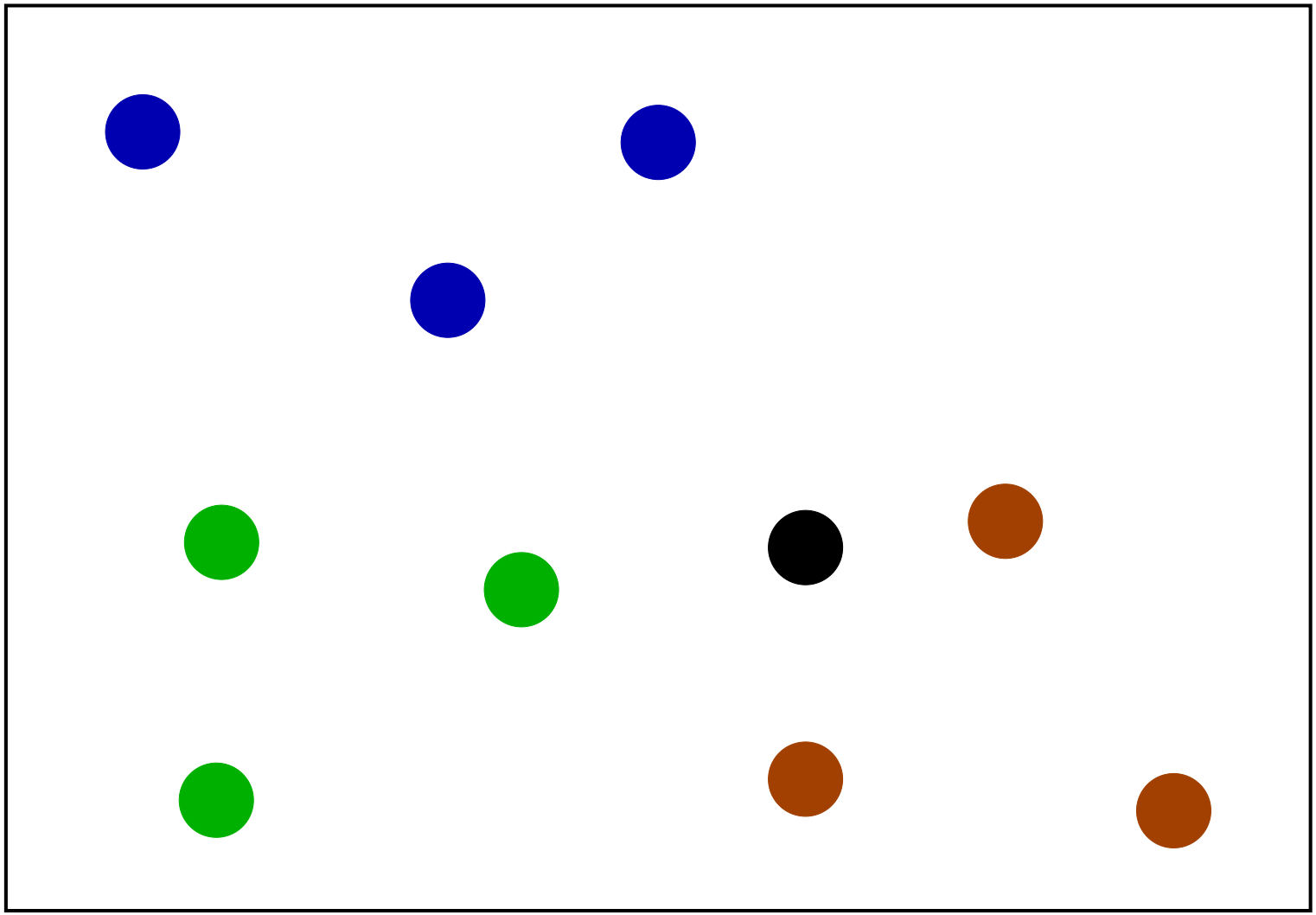
Beispiel: Entscheidungsgrenzen für MBL



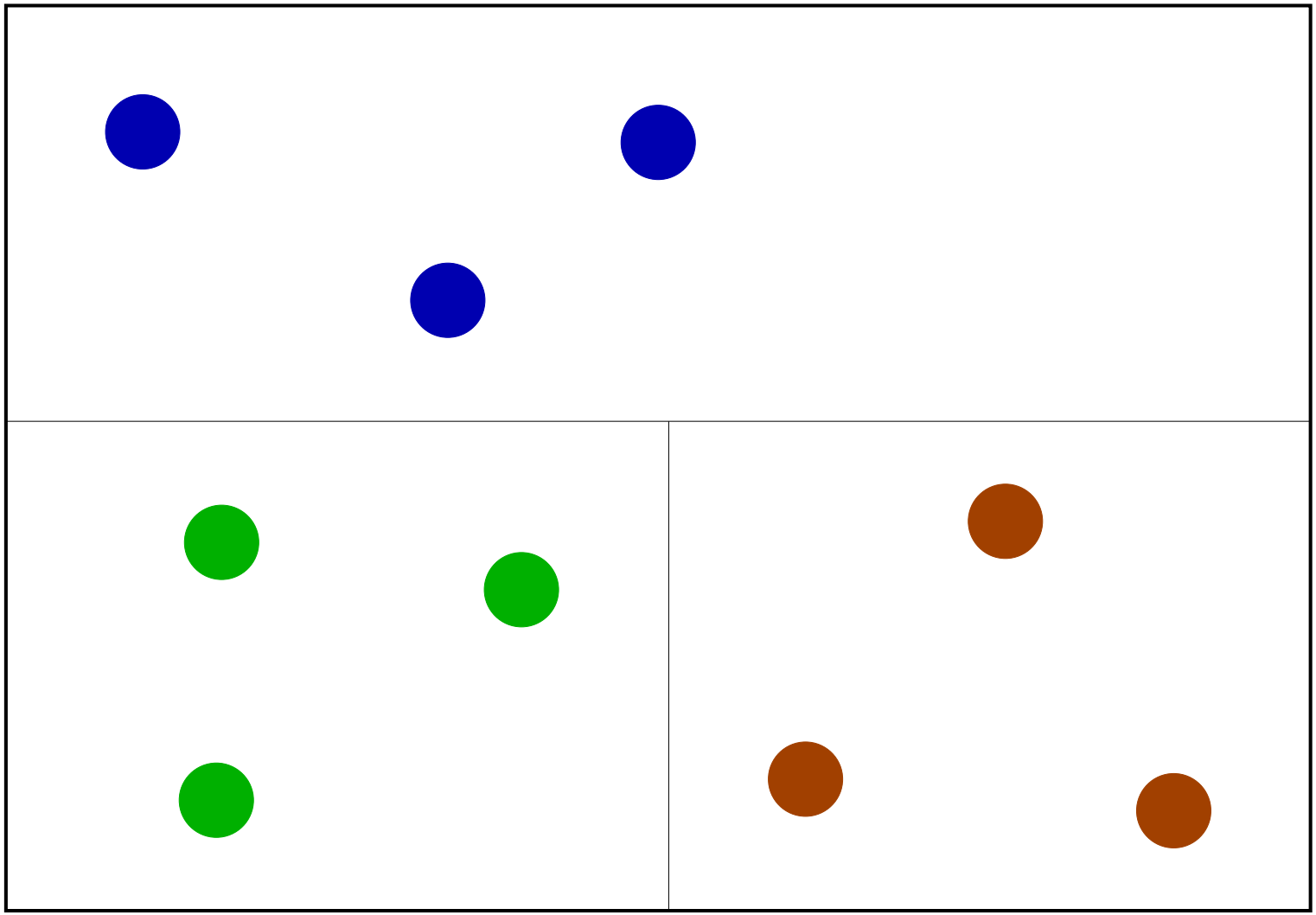
Beispiel: Entscheidungsgrenzen für Eager Learners



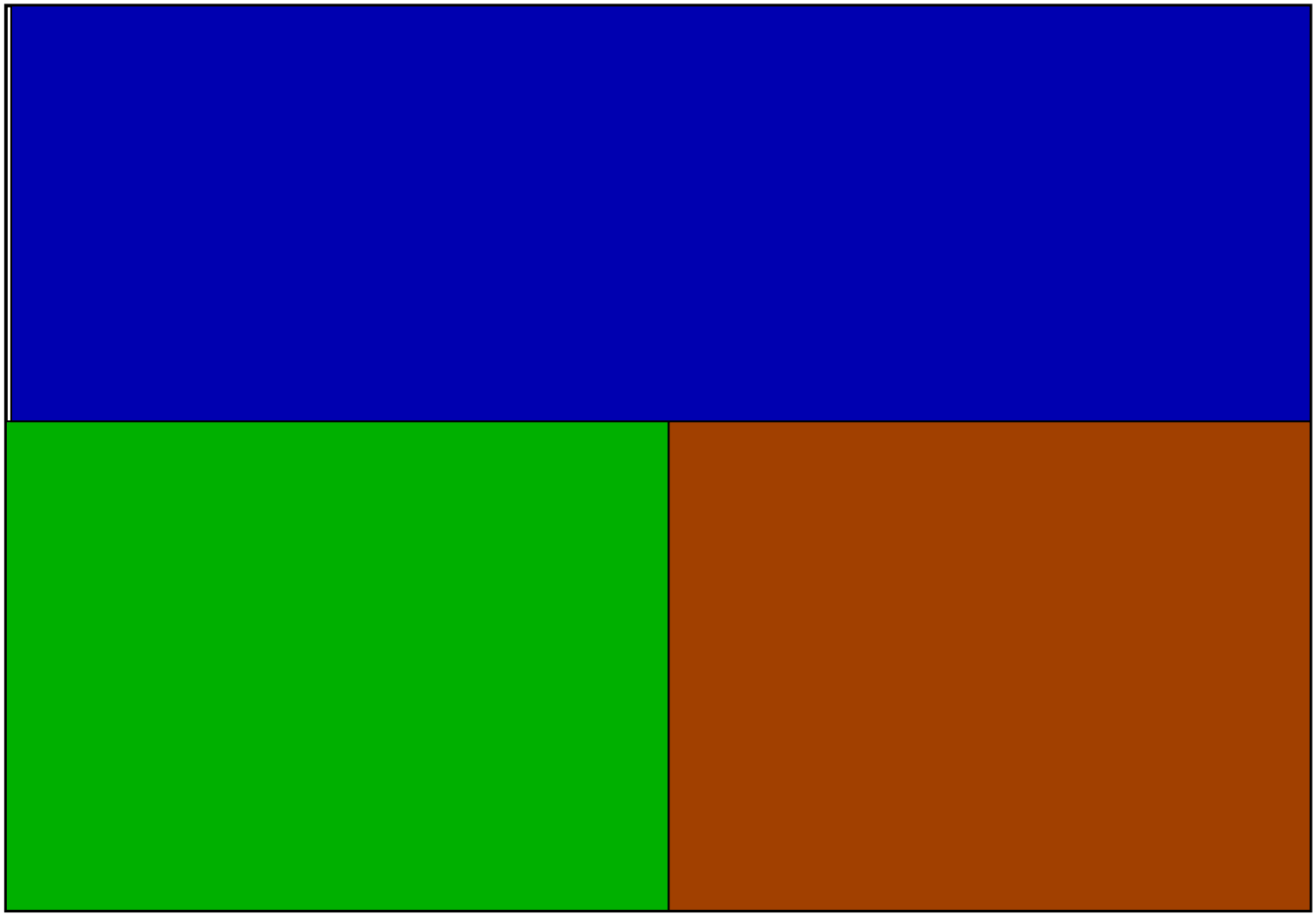
Beispiel: Entscheidungsgrenzen für Eager Learners



Beispiel: Entscheidungsgrenzen für Eager Learners



Beispiel: Entscheidungsgrenzen für Eager Learners



Entscheidungsgrenzen

Die Entscheidungsgrenzen (engl. decision boundaries) beim Memory-Based Learning sind oft komplexer als bei einem Eager Learner. Dies kann Vor- und Nachteile haben.

Welches k?

- idealerweise sollte k so gewählt werden, dass eine Mehrheitsentscheidung möglich ist, z.B. $k = 3$ bei zwei Ausgabeklassen
- ein größeres k reduziert den Effekt von Outliern
  $\Rightarrow$ Smoothing
- statt k nearest neighbours kann man auch die Gesamtheit der Trainingsinstanzen betrachten, die k weit weg sind (d.h. k bezieht sich dann auf die Distanz innerhalb derer man schaut nicht auf die Anzahl der Vergleichsinstanzen)

Pruning

Eine Reihe von verschiedenen Pruningverfahren können angewendet werden:

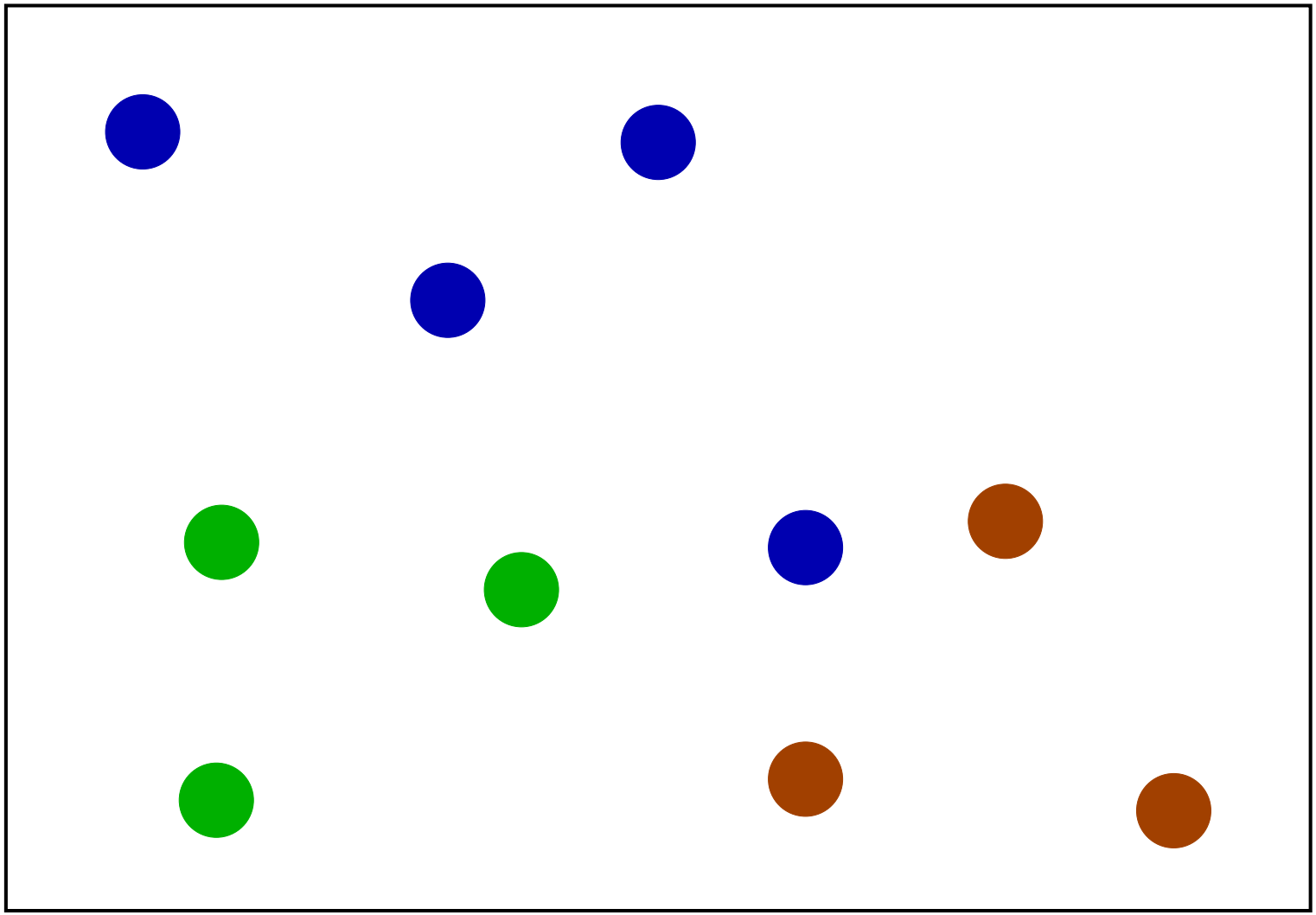
Entfernen von Outliern

Wilson Editing entfernt Outliers. Dies ist eine Abstraktion und hat einen Effekt auf die Generalisierungsperformanz. MBL wird dadurch dem Eager Learning etwas ähnlicher.

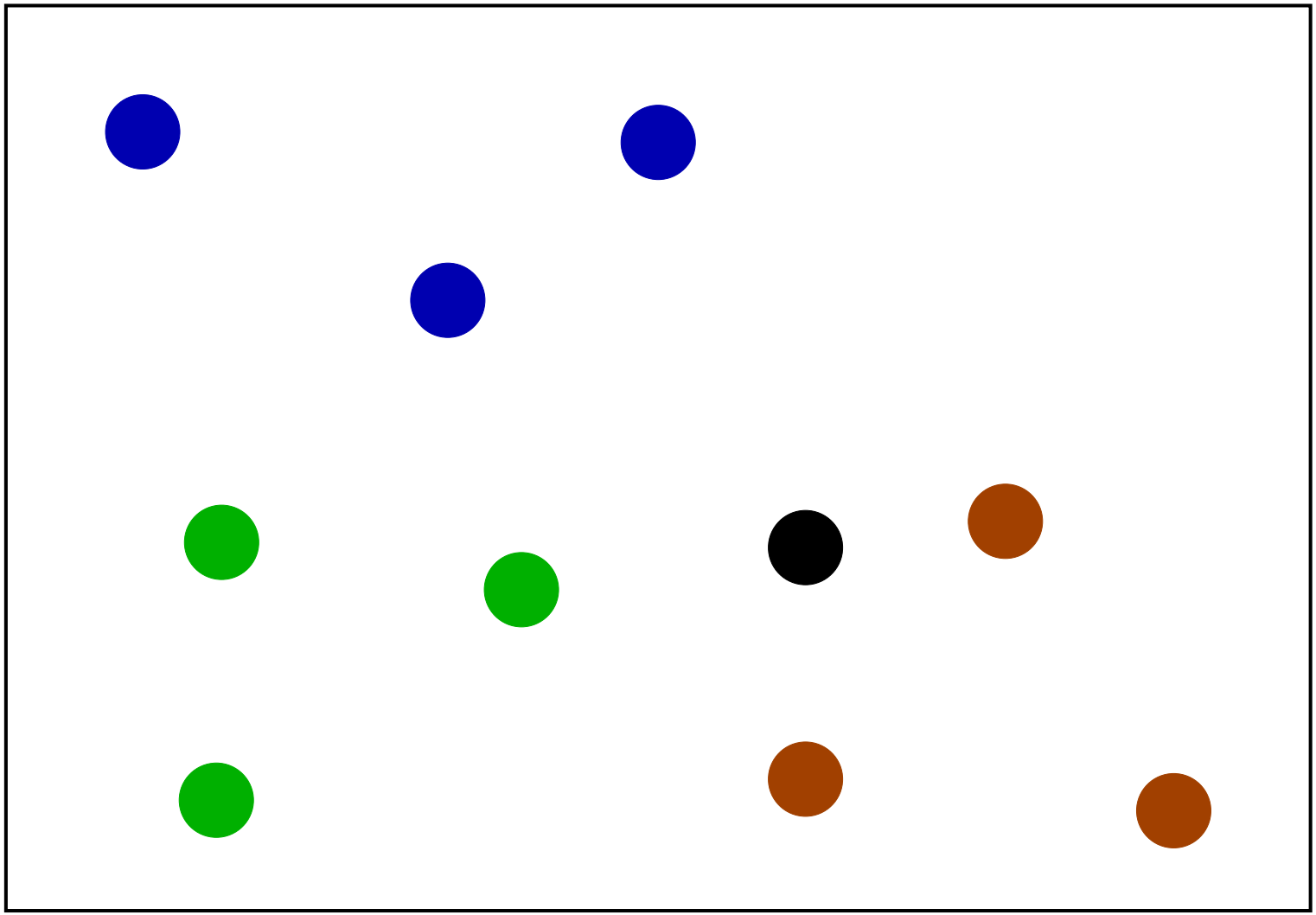
Entfernen von Redundanz

Werden nur Trainingsinstanzen entfernt, die keinen Einfluß auf die Entscheidungsgrenzen haben (Hart's Editing), hat dies keinen Effekt auf die Generalisierungsperformanz. Diese Art von Pruning kann aber deutlich den Speicherplatz und auch die Klassifikationszeit reduzieren.

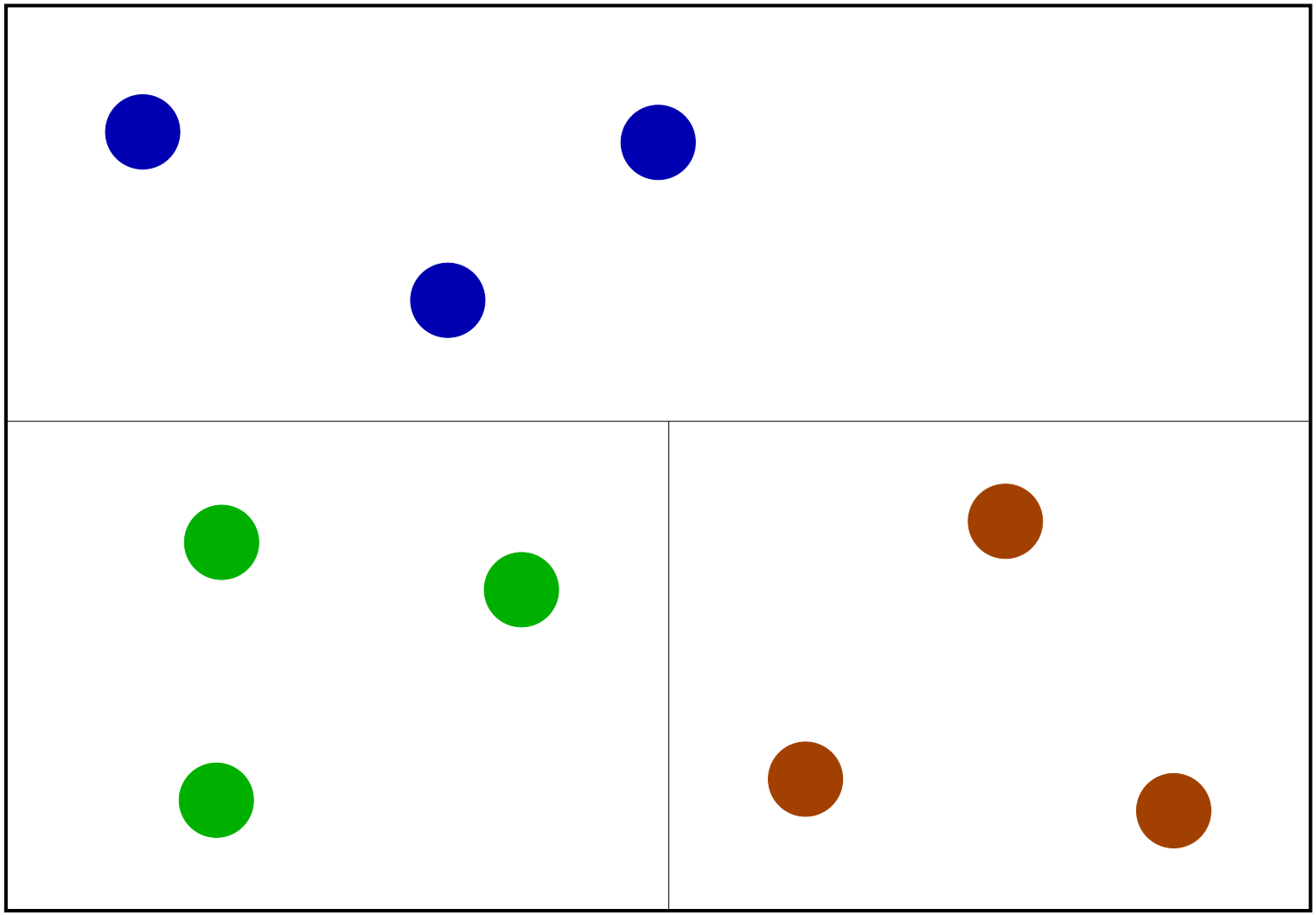
Beispiel: Entfernen von Outliern



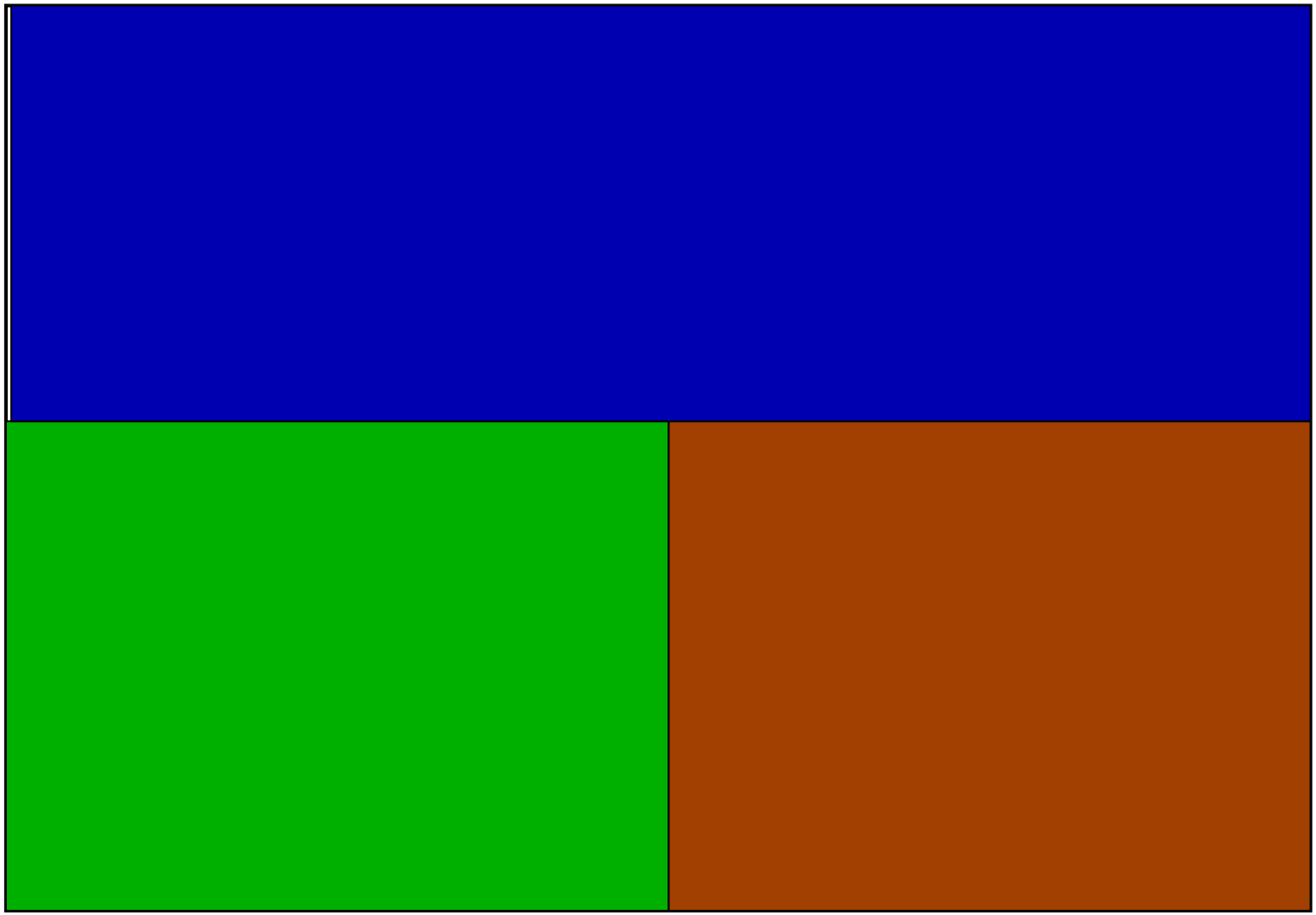
Beispiel: Entfernen von Outliern



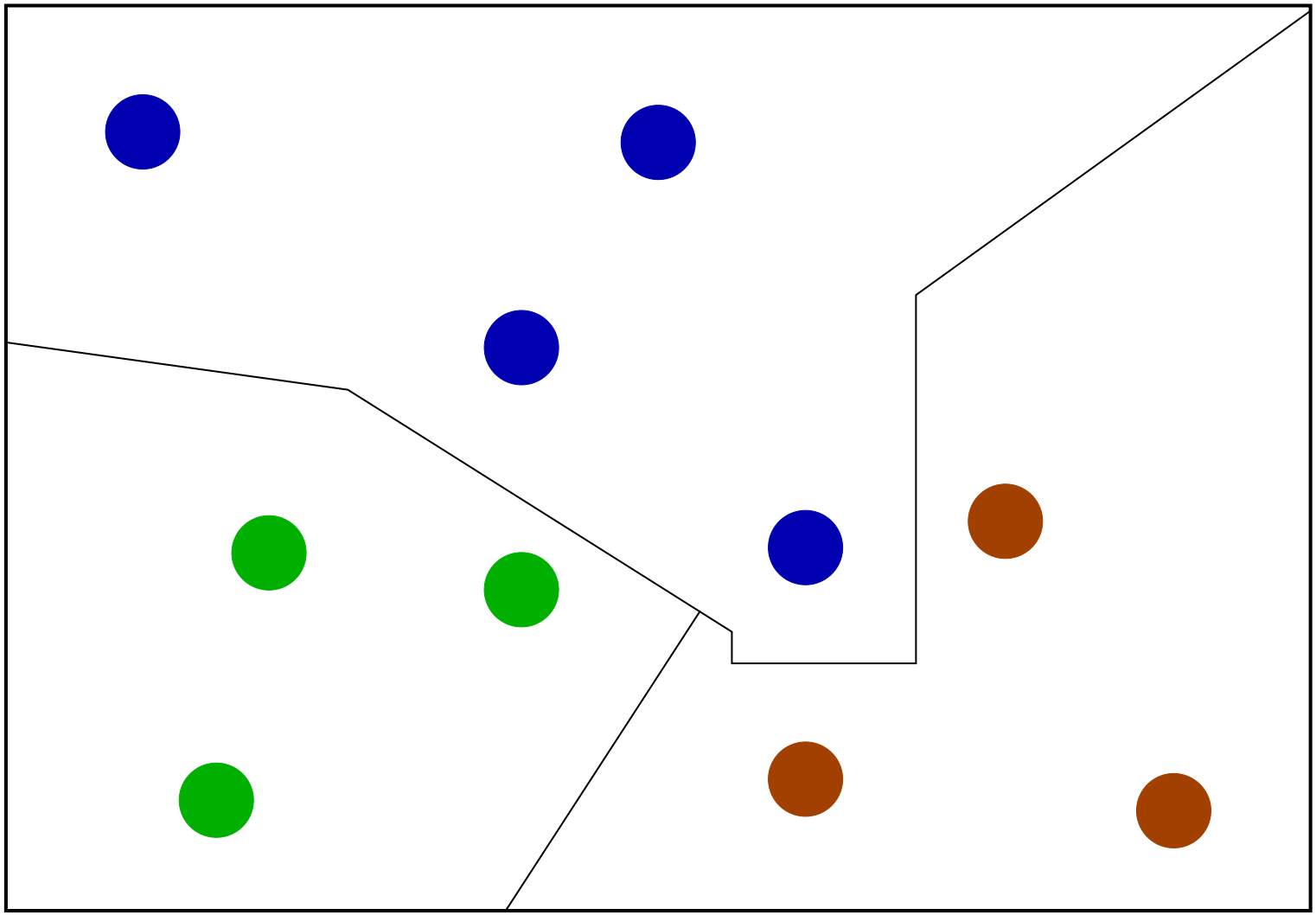
Beispiel: Entfernen von Outliern



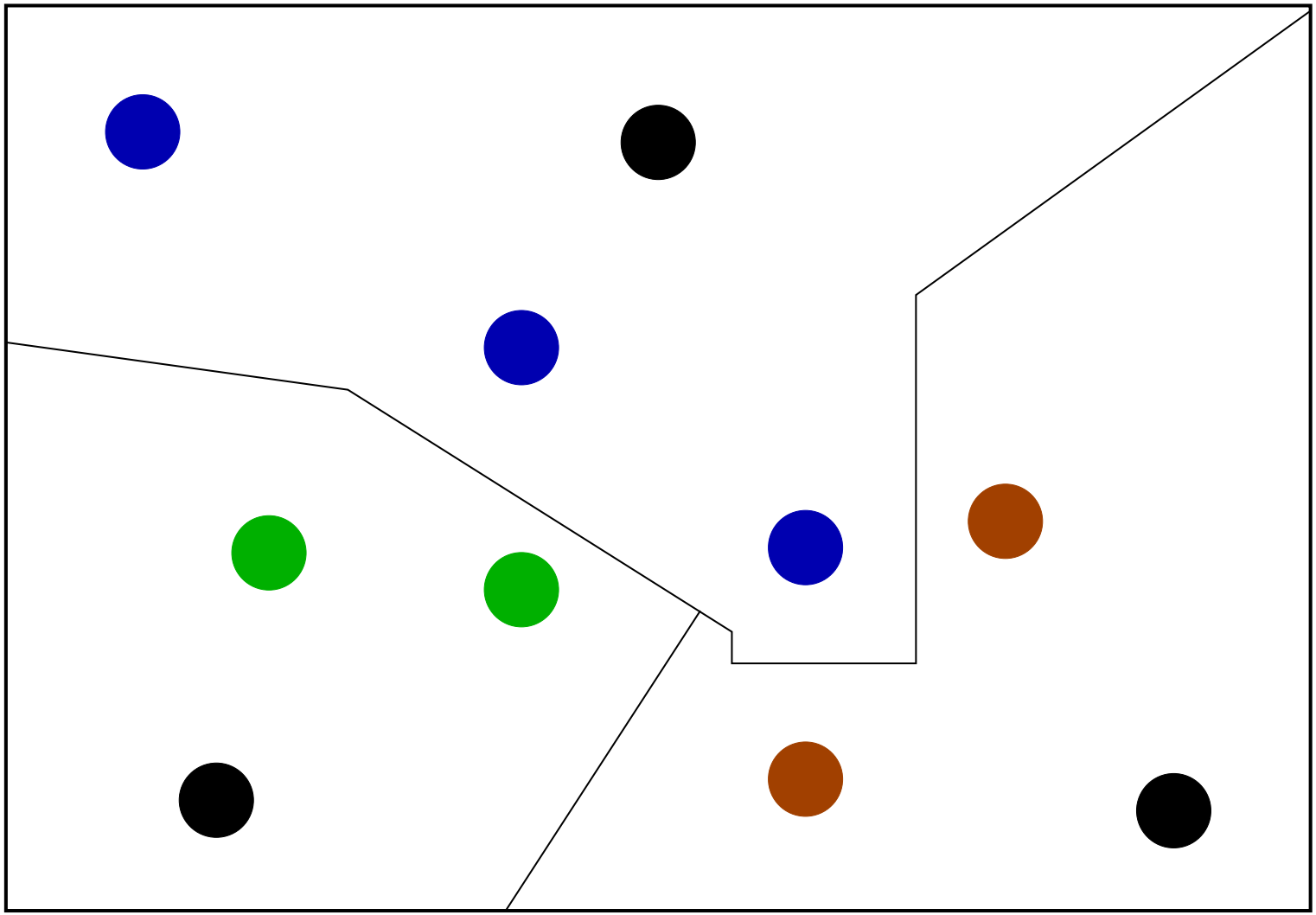
Beispiel: Entfernen von Outliern



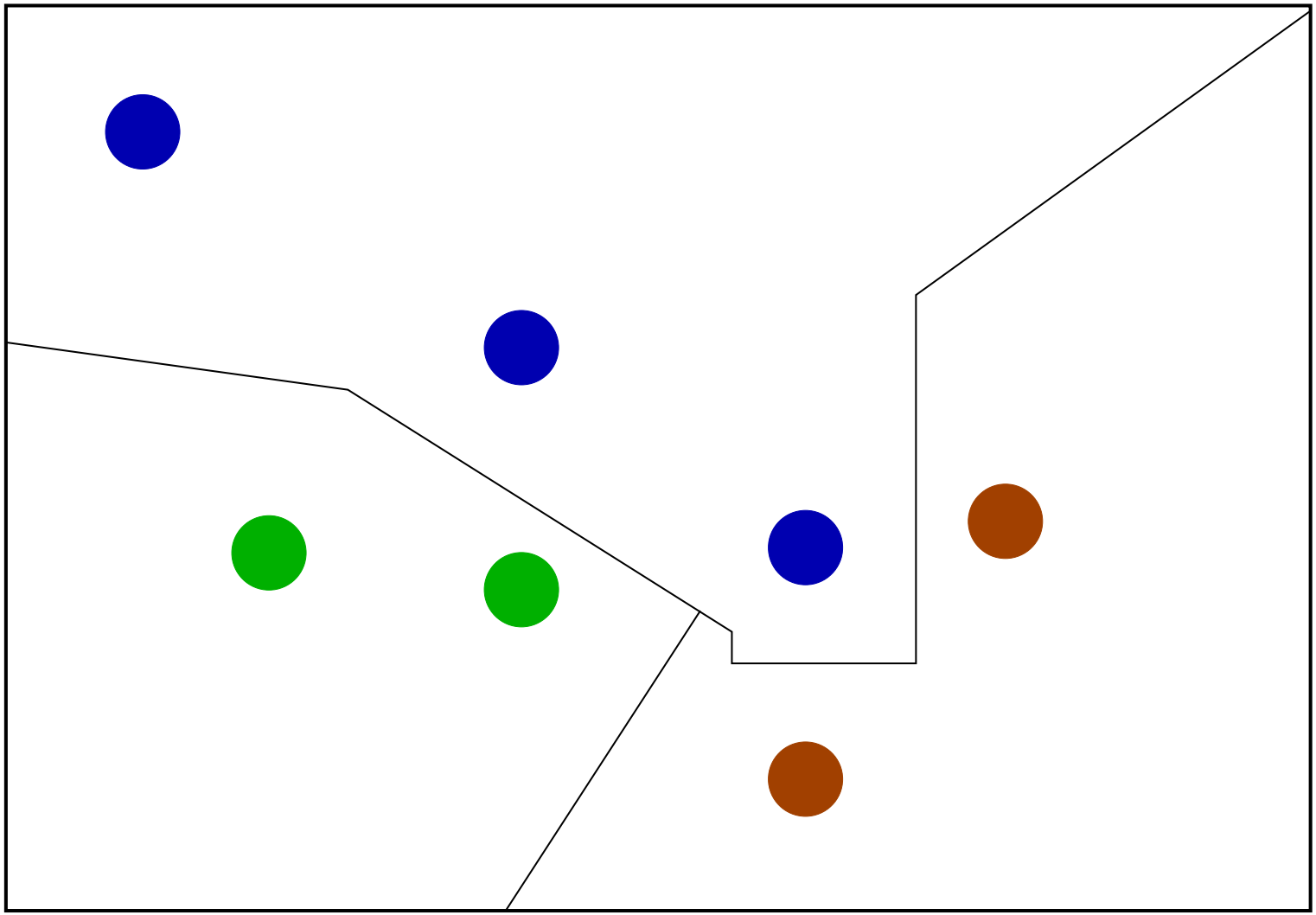
Beispiel: Entfernen von Redundanz



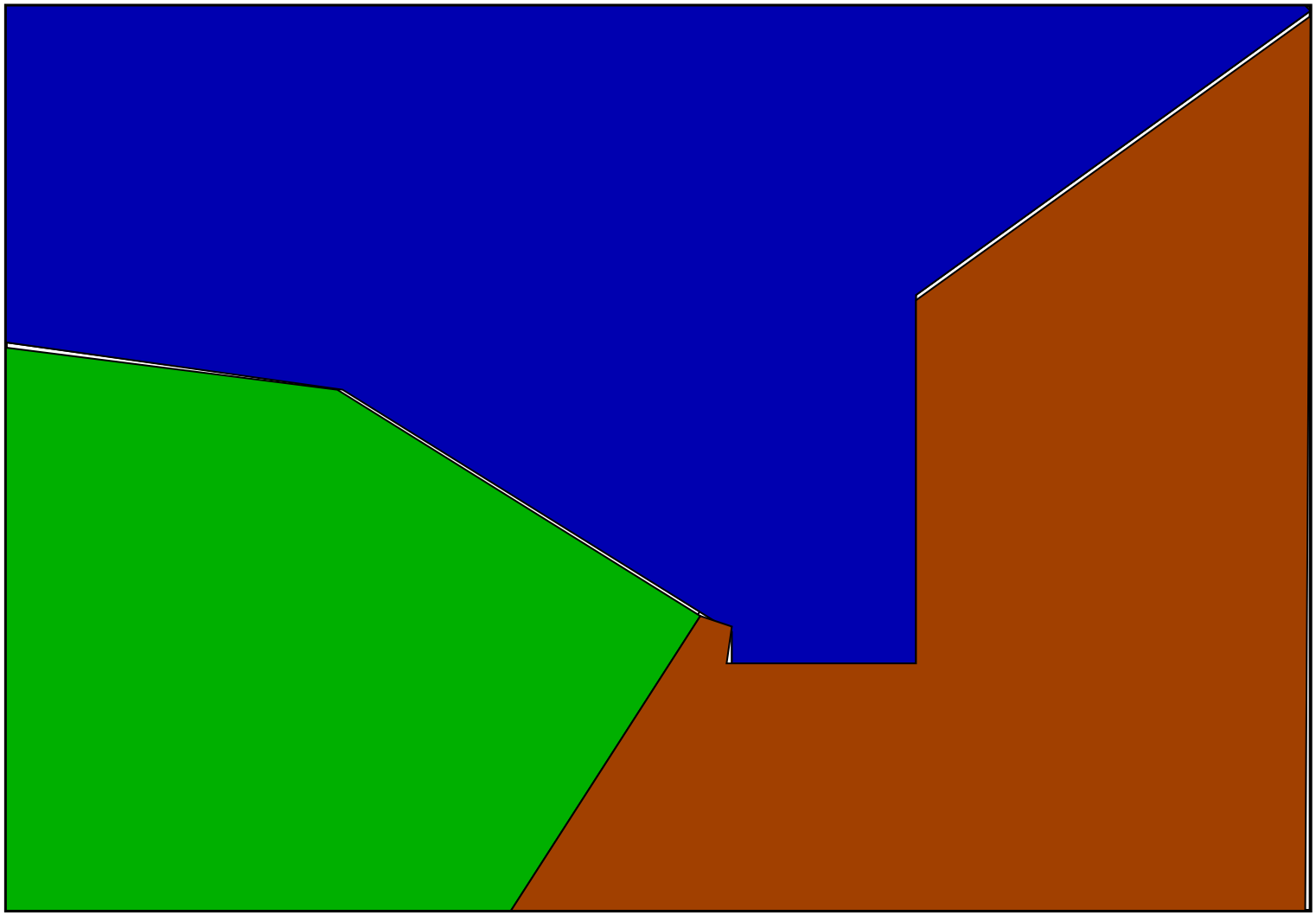
Beispiel: Entfernen von Redundanz



Beispiel: Entfernen von Redundanz



Beispiel: Entfernen von Redundanz



IB1

Das Euclidische Distanzmaß läßt sich nur auf numerische Attribute anwenden. IB1 ist eine Generalisierung, die sich auch auf nominale Attribute anwenden läßt.

$\Delta(X, Y) = \Sigma_{i=1}^{n} \delta(x_i, y_i)$ wobei

$$\delta(x_i, y_i) = \left\{ \begin{array}{ll} \frac{x_i - y_i}{max_i - min_i} & \textit{falls numerisch} \\ 0 & \textit{falls } x_i = y_i \\ 1 & \textit{falls } x_i \neq y_i \end{array} \right\}$$

Manchmal ist ein einfacher Test auf Gleichheit bei nominalen Attributen zu simplistische (wann?). Dann kann man auch komplexere Ähnlichkeitsmaße definieren.

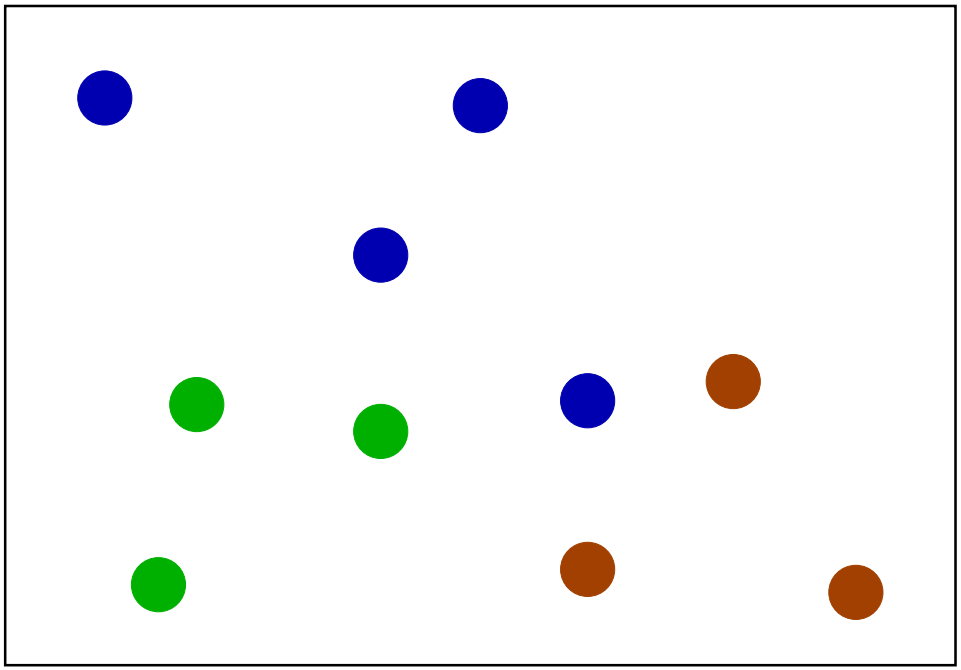
Gewichtung von Attributen

Nicht immer sind alle Attribute gleich nützlich um die Ausgabeklasse zu inferieren. In diesem Fall können Attribute gewichtet werden, z.B. mit Information Gain:

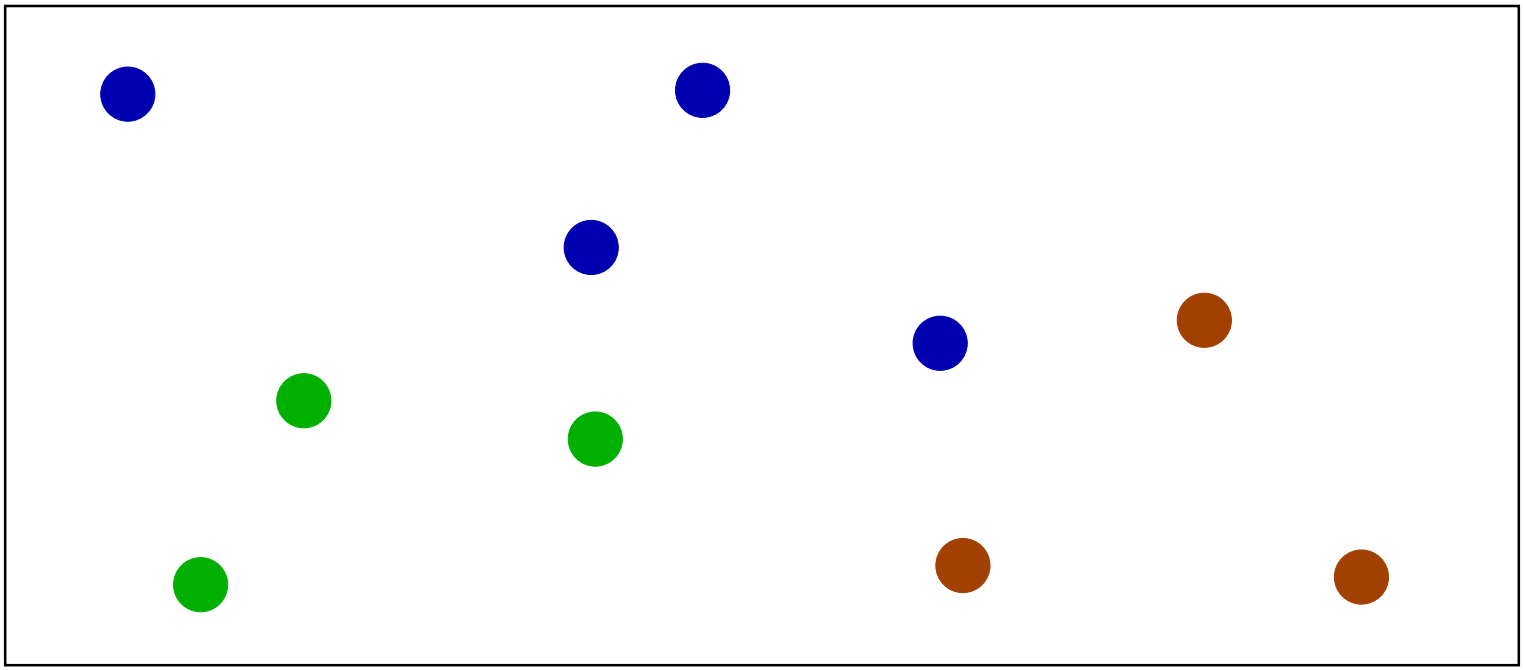
$\Delta(X, Y) = \Sigma_{i=1}^{n} \mathbf{w_i} \delta(x_i, y_i)$ wobei w_i der Information Gain von Attribut i ist.

Durch unterschiedliche Gewichtung der Attribute, dehnt oder schrumpft man die entsprechenden Achsen im Attributraum.

Beispiel: Attribut-Gewichtung



Beispiel: Attribut-Gewichtung



Induktiver Bias

Was ist der induktive Bias von Memory-Based Learning?

Induktiver Bias

Was ist der induktive Bias von Memory-Based Learning?

die Definition der Ähnlichkeitsfunktion

Zusammenfassung

Was ihr gelernt haben solltet:

- eager vs. lazy learners (Abstraktion und Generalisierung)
- Umgang mit Outliern, Idiosynkrasien, Subregularitäten
- Entscheidungsgrenzen
- Memory-Based Learning und warum es für linguistische Anwendungen Sinn machen kann
- Pruning
- Ähnlichkeitsmaße für numerische und nominale Attribute
- Attribut-Gewichtung