Data Mining und Maschinelles Lernen

Lösungsvorschlag für das 10. Übungsblatt

TECHNISCHE UNIVERSITÄT DARMSTADT

Aufgabe 1: AdaBoost (1)

TECHNISCHE UNIVERSITÄT DARMSTADT

Rechnen Sie das AdaBoost-Beispiel aus der Vorlesung (Ensemble-Methoden, Folien 14ff) nach. Verwenden Sie für die einzelnen Datenpunkte die folgenden Koordinaten:

x	y	Klasse	x	y	Klasse
1	5	+	3	1	-
2	2	+	4	6	-
5	8	+	7	4	-
6	10	+	9	3	-
8	7	+	10	9	-

Als Basis-Lerner sollen Decision Stumps (also waagrechte bzw. senkrechte Splits, z.B. $x > 4 \rightarrow +$) verwendet werden.
Der Basislerner wählt unter allen möglichen Splits jenen aus, bei dem die Gesamtsumme der Gewichte der falsch klassifizierten Beispiele minimiert wird.
Wählen Sie bei Gleichstand den zuerst gefundenen Test beginnend mit vertikalen Splits mit aufsteigenden Thresholds.

Aufgabe 1: AdaBoost (2)

TECHNISCHE UNIVERSITÄT DARMSTADT

1. initialize example weights $w_i = 1/N \quad (i = 1..N)$
2. for $m = 1$ to t // t ... number of iterations
   a) learn a classifier C_m using the current example weights
   b) compute a weighted error estimate $err_m = \frac{\sum w_i \, of \, all \, incorrectly \, classified \, e_i}{\sum_{i=1}^{N} w_i}$
   c) compute a classifier weight $\alpha_m = \frac{1}{2} \ln\left(\frac{1 - err_m}{err_m}\right)$
   d) for all correctly classified examples e_i: $w_i \leftarrow w_i e^{-\alpha_m}$
   e) for all incorrectly classified examples e_i: $w_i \leftarrow w_i e^{\alpha_m}$
   f) normalize the weights w_i so that they sum to 1
3. for each test example
   a) try all classifiers C_m
   b) predict the class that receives the highest sum of weights α_m

= 1 because weights are normalized

update weights so that sum of correctly classified examples equals sum of incorrectly classified examples

Aufgabe 1: AdaBoost (3)

TECHNISCHE UNIVERSITÄT DARMSTADT

a) Berechnen Sie die ersten 3 AdaBoost-Iterationen.
Generelles Vorgehen In jeder Iteration sind jeweils 10 horizontale (y $\leq$ Wert) und 10 vertikale Splits möglich. Jeder Split ermöglicht 2 Vergleiche, z.B. y $\leq$ 4 und y $>$ 4.
Für jeden dieser Vergleiche addieren wir die Gewichte der Beispiele auf, die durch den Vergleich falsch klassifiziert werden. Anschließend wählen wir denjenigen Vergleich aus, der die geringste Summe aufweist.
Wir berechnen danach die Gewichtung α_m des aus dem Vergleich resultierenden Klassifizierer (Decision Stump). Die Gewichte der Beispiele werden unter Verwendung der Gewichtung α_m erhöht bzw. gesenkt, falls sie falsch bzw. richtig klassifiziert werden.
Am Ende jeder Iteration werden die Gewichte der Beispiele so normiert, daß ihre Summe eins ergibt.
Die Folien 11-15 illustrieren diese Vorgehensweise.
Hinweis: *Bei den berechneten Fehlern können abhängig von der verwendeten Methode (Berechnung einer oder beider Tabellenspalte(n)) Abweichungen auftreten, da die Gesamtsumme der Beispielsgewichte bedingt durch die Rundung der Werte selten 1 beträgt.*

AdaBoost
Erste Iteration (1)

TECHNISCHE UNIVERSITÄT DARMSTADT

Beginnen wir nun mit den Berechnungen. Am Anfang haben alle Beispiele das Gewicht $1/10$ (10 Beispiele). Betrachten wir nun alle vertikalen Splits (links), sowie horizontalen Splits (rechts).

	Fehler	
Wert	$x \leq$ Wert $\Rightarrow +$	$x >$ Wert $\Rightarrow +$
1	$4/10$	$6/10$
2	$3/10$	$7/10$
3	$4/10$	$6/10$
4	$5/10$	$5/10$
5	$4/10$	$6/10$
6	$3/10$	$7/10$
7	$4/10$	$6/10$
8	$3/10$	$7/10$
9	$4/10$	$6/10$
10	$5/10$	$5/10$

	Fehler	
Wert	$y \leq$ Wert $\Rightarrow +$	$y >$ Wert $\Rightarrow +$
1	$6/10$	$4/10$
2	$5/10$	$5/10$
3	$6/10$	$4/10$
4	$7/10$	$3/10$
5	$6/10$	$4/10$
6	$7/10$	$3/10$
7	$6/10$	$4/10$
8	$5/10$	$5/10$
9	$6/10$	$4/10$
10	$5/10$	$5/10$

Betrachten wir beide Tabellen, sehen wir, daß es 5 Splits mit minimalen Fehler gibt (rot markiert). Wir entscheiden uns für den zuerst gefundenen Split ($x \leq 2 \Rightarrow +$).

AdaBoost
Erste Iteration (2)

TECHNISCHE UNIVERSITÄT DARMSTADT

Berechnen wir nun das Gewicht des resultierenden Klassifizierers. Hierfür benötigen wir zunächst den Fehler err_1:

$$err_1 = \frac{3}{10}$$

Hiermit können wir nun das Gewicht α_1 des Klassifizierers berechnen:

$$\alpha_1 = \frac{1}{2} \log\left(\frac{1 - err_1}{err_1}\right) = \frac{1}{2} \log\left(\frac{7}{3}\right) \approx 0,424$$

Damit ergeben sich die folgenden Faktoren, mit denen die einzelnen Gewichte multipliziert werden:

$$w_i \leftarrow \begin{cases} w_i \cdot e^{-\alpha_1} \approx 0,0654, & \text{falls } w_i \text{ korrekt klassifiziert wird} \\ w_i \cdot e^{\alpha_1} \approx 0,1528, & \text{falls } w_i \text{ falsch klassifiziert wird} \end{cases}$$

Da in der ersten Iteration alle Anfangsgewichte gleich waren, sind dies auch die möglichen neuen Gewichte (mit Faktor 10).
Normalisierung: Da sieben Beispiele korrekt und drei falsch klassifiziert wurden, erhalten wir:

$$3 \cdot 0,1528 + 7 \cdot 0,0654 = 0,9162$$

als Gesamtsumme der Gewichte. Diese wollen wir auf eins normieren, aus diesem Grund teilen wir alle Gewichte durch 0,9162.

AdaBoost
Erste Iteration (3)

TECHNISCHE UNIVERSITÄT DARMSTADT

Damit erhalten wir folgende Gewichte:

$$w_i = \begin{cases} 0,071, & \text{falls } w_i \text{ korrekt klassifiziert wird} \\ 0,167, & \text{falls } w_i \text{ falsch klassifiziert wird} \end{cases}$$

und folgende Tabelle:

x	y	Gewicht	Klasse
1	5	0,071	+
2	2	0,071	+
3	1	0,071	-
4	6	0,071	-
5	8	0,167	+
6	10	0,167	+
7	4	0,071	-
8	7	0,167	+
9	3	0,071	-
10	9	0,071	-

Rundungsbedingt ergibt die Gesamtsumme der Gewichte den Wert 0,998.

AdaBoost
Zweite Iteration (1)

TECHNISCHE UNIVERSITÄT DARMSTADT

Suchen wir nun den nächsten Split.
Zur Veranschaulichung, die Berechnung des Fehlers für den ersten möglichen Stump $x \leq 1 \Rightarrow +$:

x	y	Klasse	(Fehler)	x	y	Klasse	(Fehler)
1	5	+	(0)	3	1	-	(0)
2	2	+	(0,071)	4	6	-	(0)
5	8	+	(0,167)	7	4	-	(0)
6	10	+	(0,167)	9	3	-	(0)
8	7	+	(0,167)	10	9	-	(0)

Summe der Fehlerterme: 0,572

AdaBoost
Zweite Iteration (2)

TECHNISCHE UNIVERSITÄT DARMSTADT

In Tabellenform:

Wert	Fehler	
	$x \leq$ Wert $\Rightarrow +$	$x >$ Wert $\Rightarrow +$
1	0,572	0,426
2	0,501	0,497
3	0,572	0,426
4	0,643	0,355
5	0,476	0,522
6	0,309	0,689
7	0,38	0,618
8	0,213	0,785
9	0,284	0,714
10	0,355	0,643

Wert	Fehler	
	$y \leq$ Wert $\Rightarrow +$	$y >$ Wert $\Rightarrow +$
1	0,714	0,284
2	0,643	0,355
3	0,714	0,284
4	0,785	0,213
5	0,714	0,284
6	0,785	0,213
7	0,618	0,380
8	0,451	0,547
9	0,522	0,476
10	0,355	0,643

Den Tabellen entnehmen wir, daß 3 mögliche Vergleiche optimal sind. Wiederum entschieden wir uns für den zuerst gefundenen Split, $x \leq 8 \Rightarrow +$. Punkte (3, 1), (4, 6) und (7, 4) werden falsch und alle anderen richtig klassifiziert.

AdaBoost
Zweite Iteration (3)

TECHNISCHE UNIVERSITÄT DARMSTADT

Demnach hat err_2 den folgenden Wert:

$$err_2 \approx 0,213$$

Mit err_2 können wir α_2 berechnen:

$$\alpha_2 = \frac{1}{2} \log \left(\frac{0,787}{0,213} \right) \approx 0,652$$

Berechnen wir nun die Faktoren e^{α_2} bzw. $e^{-\alpha_2}$, mit denen wir die Gewichte multiplizieren:

$$e^{-\alpha_2} = 0,521$$

$$e^{\alpha_2} = 1,919$$

Damit ergibt sich die folgende Rechenhilfe-Tabelle:

	Neues Gewicht	
Altes Gewicht	Korrekt klassifiziert	falsch klassifiziert
0,071	0,037	0,136
0,167	0,087	0,32

AdaBoost
Zweite Iteration (4)

TECHNISCHE UNIVERSITÄT DARMSTADT

Multiplizieren wir die Gewichte der korrekt (blau) und falsch (rot) klassifizierten Beispiele mit den entsprechenden Faktoren und normieren diese, erhalten wir folgende Gewichte:

x	y	Altes Gewicht	Neues Gewicht		Klasse
			Nicht normiert	Normiert	
1	5	0,071	0,037	0,045	+
2	2	0,071	0,037	0,045	+
3	1	0,071	0,136	0,166	-
4	6	0,071	0,136	0,166	-
5	8	0,167	0,087	0,106	+
6	10	0,167	0,087	0,106	+
7	4	0,071	0,136	0,166	-
8	7	0,167	0,087	0,106	+
9	3	0,071	0,037	0,045	-
10	9	0,071	0,037	0,045	-

Erneut ist zu beachten, daß aufgrund der Rundung die Gesamtsumme den Wert 0,996 hat.

AdaBoost
Dritte Iteration (1)

TECHNISCHE UNIVERSITÄT DARMSTADT

Für den letzten Klassifizierer nun wiederum die Splits:

Wert	Fehler	
	$x \leq$ Wert $\Rightarrow +$	$x >$ Wert $\Rightarrow +$
1	0,363	0,633
2	0,318	0,682
3	0,484	0,512
4	0,650	0,346
5	0,544	0,452
6	0,438	0,558
7	0,604	0,392
8	0,498	0,498
9	0,543	0,453
10	0,588	0,408

Wert	Fehler	
	$y \leq$ Wert $\Rightarrow +$	$y >$ Wert $\Rightarrow +$
1	0,574	0,422
2	0,529	0,467
3	0,574	0,422
4	0,740	0,256
5	0,695	0,301
6	0,861	0,135
7	0,755	0,241
8	0,649	0,347
9	0,694	0,302
10	0,588	0,408

Der beste Split ist $y > 6 \Rightarrow +$.

AdaBoost
Dritte Iteration (2)

TECHNISCHE UNIVERSITÄT DARMSTADT

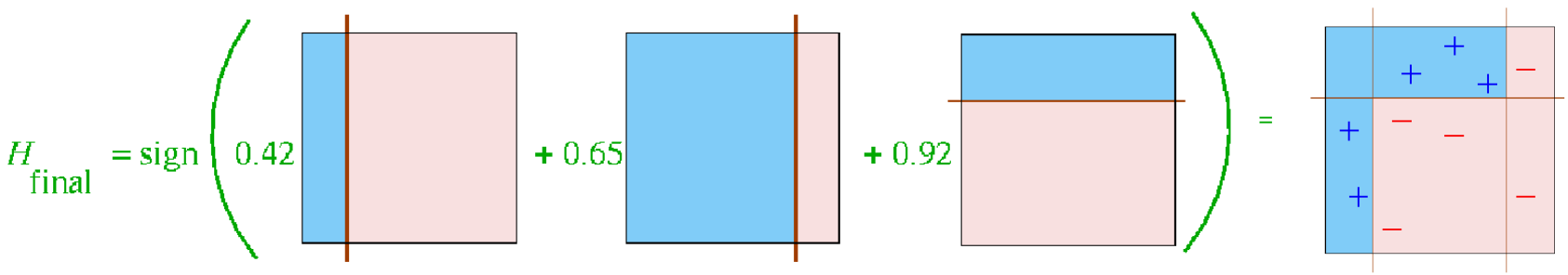
Berechnen wir nun das Gewicht des Klassifizierers. Es gilt

$$err_3 = 0,135$$

und damit

$$\alpha_3 = 0,929.$$

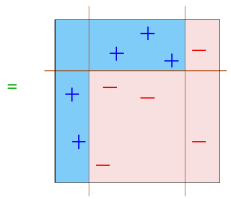
Finales Modell: Damit haben wir die drei Klassifizierer (und deren Gewichte) des Beispiels aus der Vorlesung berechnet! Eine Illustration des resultierenden Klassifizierers befindet sich auf Folie "*Final Hypothesis*".



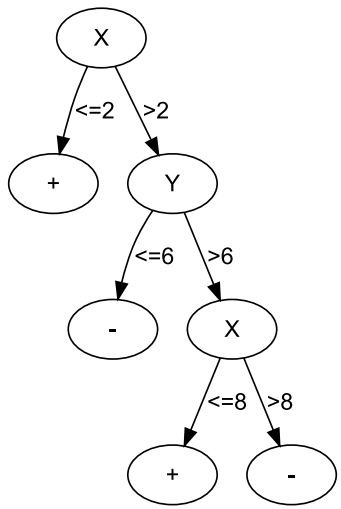
AdaBoost
Darstellung (1)

TECHNISCHE UNIVERSITÄT DARMSTADT

b) Generieren Sie aus den eben berechneten Decision Stumps einen Entscheidungsbaum.
Lösung: Basierend auf der finalen Hypothese



ist eine von mehreren möglichen Lösungen folgende:



Aufgabe 2: Stacking (1)

TECHNISCHE UNIVERSITÄT DARMSTADT

In dieser Aufgabe sollen Sie unter Verwendung mehrerer Basislerner und der Ensemble-Methode Stacking einen Entscheidungsbaum lernen. Verwenden Sie hierfür den Datensatz und die drei Decision Stumps aus der vorherigen Aufgabe. a) Konvertieren Sie diesen Datensatz, d.h. ersetzen sie die Attribute durch eine neue Attributsmenge, die jeweils ein Attribut für jeden Decision Stump beinhaltet. Als Attributwerte werden die Vorhersagen des entsprechenden Klassifizierers verwendet.

Aufgabe 2: Stacking (2)

TECHNISCHE UNIVERSITÄT DARMSTADT

- Basic Idea:
  - learn a function that combines the predictions of the individual classifiers
- Algorithm:
  - train n different classifiers $C_1 ... C_n$ (the *base classifiers*)
  - obtain predictions of the classifiers for the training examples
  - form a new data set (the *meta data*)
    - **classes**
      - the same as the original dataset
    - **attributes**
      - one attribute for each base classifier
      - value is the prediction of this classifier on the example
  - train a separate classifier M (the *meta classifier*)

This is better done with cross-validation!

Stacking
Datensatz (1)

TECHNISCHE UNIVERSITÄT DARMSTADT

In der vorherigen Aufgabe haben wir drei Klassifizierer generiert, diese verwenden wir nun als Basisklassifizierer für die Ensemble-Methode Stacking. Hierfür müssen wir zunächst den Datensatz konvertieren.
Jeder Basisklassifizierer wird durch ein Attribut repräsentiert und dessen Vorhersage für eine Instanz stellt deren Attributwert dar. Das heißt, wir bestimmen für jede Instanz einen Vektor, der aus den Vorhersagen des ersten, zweiten und dritten Klassifizierers und der ursprünglichen Klasse besteht. Wenden wir dies auf unseren Datensatz an, erhalten wir folgendes:

		C1	C2	C3	
X	Y	$x \leq 2$	$x \leq 8$	$y > 6$	Klasse
1	5	+	+	-	+
2	2	+	+	-	+
3	1	-	+	-	-
4	6	-	+	-	-
5	8	-	+	+	+
6	10	-	+	+	+
7	4	-	+	-	-
8	7	-	+	+	+
9	3	-	-	-	-
10	9	-	-	+	-

Stacking
Datensatz (2)

TECHNISCHE UNIVERSITÄT DARMSTADT

Die ersten beiden Zeilen stellen die ursprünglichen Attributwerte dar und können (bzw., im üblichen Szenario, sollen!) entfernt werden. Damit erhalten wir den folgenden konvertierten Datensatz:

C1	C2	C3	
$x \leq 2$	$x \leq 8$	$y > 6$	Klasse
+	+	-	+
+	+	-	+
-	+	-	-
-	+	-	-
-	+	+	+
-	+	+	+
-	+	-	-
-	+	+	+
-	-	-	-
-	-	+	-

Stacking
Klassifizierer (1)

TECHNISCHE UNIVERSITÄT DARMSTADT

b) Bestimmen Sie nun auf dem konvertierten Datensatz einen Entscheidungsbaum mittels des Verfahrens ID3 und Maß Information Gains. Entscheiden Sie sich bei Gleichstand z.B. für den als erstes gefundenen Test.

Lösung: Auf ID3 im speziellen gehen wir nicht mehr ein. Für die Berechnung des maximalen Information Gains bestimmen wir nur die gewichtete Summe der Entropien und minimieren diese:

		+	-	P+	P–	$\lvert S_i\rvert/\lvert S\rvert$	$E(S_i)$	$\lvert S_i\rvert/\lvert S\rvert \cdot E(S_i)$	Total Gain
C1	+	2	0	1,00	0,00	0,2	0,00	0,00	0,24
	-	3	5	0,38	0,63	0,8	0,95	0,76	
C2	+	5	3	0,63	0,38	0,8	0,95	0,76	0,24
	-	0	2	0,00	1,00	0,2	0,00	0,00	
C3	+	3	1	0,75	0,25	0,4	0,81	0,32	0,13
	-	2	4	0,33	0,67	0,6	0,92	0,55	

Die Attribute C1 und C2 sind beide gleichwertig, wir entscheiden uns für den ersten Test, also für C1.

Stacking
Klassifizierer (2)

TECHNISCHE UNIVERSITÄT DARMSTADT

Wir müssen nur die Beispiele für C1=- betrachten (die Beispiele von C1=+ gehören alle zur Klasse +):

		C1	C2	C3	
X	Y	$x \leq 2$	$x \leq 8$	$y > 6$	Klasse
3	1	-	+	-	-
4	6	-	+	-	-
5	8	-	+	+	+
6	10	-	+	+	+
7	4	-	+	-	-
8	7	-	+	+	+
9	3	-	-	-	-
10	9	-	-	+	-

Bestimmen wir also den nächsten Test:

		+	-	P+	P–	$\lvert S_i\rvert/\lvert S\rvert$	$E(S_i)$	$\lvert S_i\rvert/\lvert S\rvert \cdot E(S_i)$	Total Gain
S		3	5	0,38	0,63	-	0,95	-	-
C2	+	3	3	0,5	0,5	0,75	1	0,75	0,2
	-	0	2	0	1	0,25	0	0	
C3	+	3	1	0,75	0,25	0,5	0,81	0,41	0,54
	-	0	4	0	1	0,5	0	0	

Stacking
Klassifizierer (3)

TECHNISCHE UNIVERSITÄT DARMSTADT

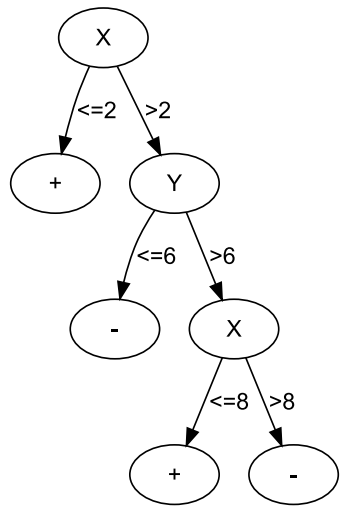
		+	-	P+	P–	$\lvert S_i\rvert/\lvert S\rvert$	$E(S_i)$	$\lvert S_i\rvert/\lvert S\rvert \cdot E(S_i)$	Total Gain
S		3	5	0,38	0,63	-	0,95	-	-
C2	+	3	3	0,5	0,5	0,75	1	0,75	0,2
	-	0	2	0	1	0,25	0	0	
C3	+	3	1	0,75	0,25	0,5	0,81	0,41	0,54
	-	0	4	0	1	0,5	0	0	

Diesmal ist C3 der beste Test.
Für den Ast C3=+ wird noch ein Test auf C2 angehängt, der die verbleibenden Instanzen perfekt auf die Klassen + und – aufteilt.

Stacking
Klassifizierer (4)

TECHNISCHE UNIVERSITÄT DARMSTADT

c) Zeichnen Sie diese Baum und vergleichen Sie ihn mit dem Entscheidungsbaum aus Aufgabe 1.
Lösung:

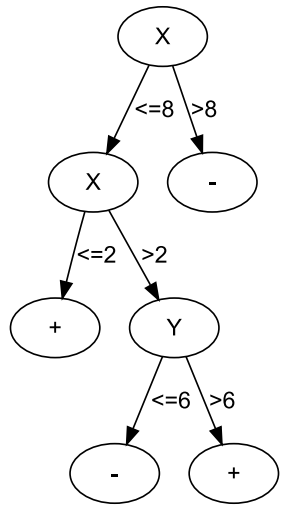


Dieser Baum entspricht dem Baum aus Aufgabe 1b).

Stacking
Klassifizierer (5)

TECHNISCHE UNIVERSITÄT DARMSTADT

Natürlich kann sich durch eine andere Wahl im Wurzelknoten ein anderer Baum ergeben, z.B:



Aufgabe 3: `RelieF` (1)

TECHNISCHE UNIVERSITÄT DARMSTADT

Gegeben sind folgende 12 Beispiele der Wetter-Daten:

ID	outlook	temperature	humidity	windy	play
1	sunny	hot	high	FALSE	no
2	rainy	mild	high	FALSE	yes
3	rainy	cool	normal	FALSE	yes
4	rainy	cool	normal	TRUE	no
5	overcast	cool	normal	TRUE	yes
6	sunny	mild	high	FALSE	no
7	sunny	cool	normal	FALSE	yes
8	rainy	mild	normal	FALSE	yes
9	sunny	mild	normal	TRUE	yes
10	overcast	mild	high	TRUE	yes
11	overcast	hot	normal	FALSE	yes
12	rainy	mild	high	TRUE	no

Berechnen Sie die `RelieF` Feature-Gewichte (Foliensatz "*Instanzenbasiertes Lernen*", Folie "*RELIEF (2)*") für alle 4 Attribute (die ID ist nur zur leichten Identifizierung eines Beispiels vorhanden). Berechnen Sie den Nearest Hit und Nearest Miss für jedes Beispiel ($r = 12$). Gehen Sie davon aus, dass jedes Beispiel genau einmal gewählt wird (Es kann aber natürlich mehrfach als Nearest-Neighbor auftauchen.). Als Distanz-Funktion nehmen Sie einfach die Anzahl der verschiedenen Attribute.

Aufgabe 3: `RelieF` (2)

TECHNISCHE UNIVERSITÄT DARMSTADT

1. set all attribute weights $w_A = 0.0$
2. for $i = 1$ to r (← user-settable parameter)
   - select a random example x
   - find
     - h: nearest neighbor of same class (*near hit*)
     - m: nearest neighbor of different class (*near miss*)
   - for each attribute A

$$w_A \leftarrow w_A + \frac{1}{r} \cdot (d_A(m, x) - d_A(h, x))$$

where $d_A(x,y)$ is the distance in attribute A between examples x and y (normalized to [0,1]-range).

Note: when used for feature weighting, all $w_A < 0.0$ are set to 0 in the end.

Aufgabe 3: `RelieF` (3)

TECHNISCHE UNIVERSITÄT DARMSTADT

Der Algorithmus geht folgendermaßen vor:
1. Zuerst setzt der Benutzer den Wert für r (in der Aufgabe gilt $r = 12$).
2. Als nächstes wählt der Algorithmus ein Beispiel zufällig aus (hier wird **nicht** sichergestellt, dass das Beispiel nicht schon einmal verwendet wurde).
3. Dann findet er den *nearest hit* und den *nearest miss* und macht ein Update auf den Gewichtswert jedes Attributs.

In der Aufgabe gehen wir davon aus, dass ein bereits verwendetes Beispiel nicht noch einmal genommen wird.
Wir machen uns ausserdem die Eigenschaft zu Nutze, dass man auch eine Tabelle der Distanzen für die *Hits* und die *Misses* erstellen kann und dann einfach direkt für jedes Beispiel den *nearest hit* und *nearest miss* ausrechnen kann (indem man die Werte für jedes Attribut einfach addiert, was dem Update-Schritt des Algorithmus entspricht):
Aus

$$w_A \leftarrow w_A + \frac{1}{r}(d_A(m_i, x_i) - d_A(h_i, x_i))$$

mit m_i und h_i als near miss und near hit von dem i.-ten Beispiel x_i, wird

$$0 + \sum_{1 \leq i \leq r} \frac{1}{r}(d_A(m_i, x_i) - d_A(h_i, x_i)) = \frac{1}{r}\left(\sum_{1 \leq i \leq r} d_A(m_i, x_i) - d_A(h_i, x_i)\right)$$

Aufgabe 3: `RelieF` (4)

TECHNISCHE UNIVERSITÄT DARMSTADT

Wir erstellen also 2 Tabellen für die *Hits* und eine für die *Misses* (zu beachten ist, dass die Tabellen an der Diagonalen gespiegelt sind):

Hits für +								
ID	2	3	5	7	8	9	10	11
2	0	2	4	3	1	3	2	3
3	2	0	2	1	1	3	4	2
5	4	2	0	2	3	2	2	2
7	3	1	2	0	2	2	4	2
8	1	1	3	2	0	2	3	2
9	3	3	2	2	2	0	2	3
10	2	4	2	4	3	2	0	3
11	3	2	2	2	2	3	3	0

Hits für -				
ID	1	4	6	12
1	0	4	1	3
4	4	0	4	2
6	1	4	0	2
12	3	2	2	0

Misses				
ID	1	4	6	12
2	2	3	1	1
3	3	1	3	3
5	4	1	4	3
7	2	2	2	4
8	3	2	2	2
9	3	2	2	2
10	3	3	2	1
11	2	3	3	4

In den Tabellen sind die *nearest neighbors* der gleichen Klasse rot markiert (wobei bei gleicher Distanz immer das erste Beispiel in der Tabelle ausgewählt wurde).

Aufgabe 3: `RelieF` (5)

TECHNISCHE UNIVERSITÄT DARMSTADT

Nun erstellt man eine große Tabelle in der man für jedes Attribut die Unterscheidungen festhält (gleicher Wert = Distanz 0, unterschiedlicher Wert = Distanz 1).
In dieser Tabelle repräsentiert eine Zeile jeweils einen Durchlauf des Algorithmus (wobei wir die Gewichte erst am Ende updaten, bzw. berechnen).

	Hits					Misses				
ID	ID	outl.	temp.	hum.	wind	ID	outl.	temp.	hum.	wind
2	8	0	0	1	0	6	1	0	0	0
3	7	1	0	0	0	4	0	0	0	1
5	3	1	0	0	1	4	1	0	0	0
7	3	1	0	0	0	1	0	1	1	0
8	2	0	0	1	0	4	0	1	0	1
9	5	1	1	0	0	4	1	1	0	0
10	2	1	0	0	1	12	1	0	0	0
11	3	1	1	0	0	1	1	0	1	0
1	6	0	1	0	0	2	1	1	0	0
4	12	0	1	1	0	3	0	0	0	1
6	1	0	1	0	0	2	1	0	0	0
12	4	0	1	1	0	2	0	0	0	1
		6	6	4	2		7	4	2	4

Aufgabe 3: `RelieF` (6)

TECHNISCHE UNIVERSITÄT DARMSTADT

Da man nun alle Werte zu Verfügung hat, kann man die `RelieF` Feature-Gewichte errechnen:

$$
\begin{aligned}
W(outlook) &= \frac{1}{r}\left(\sum_{1\leq i\leq r} d_A(m_i, x_i) - d_A(h_i, x_i)\right) \\
W(outlook) &= \frac{1}{r}\left(d_{outlook}(m_1, x_1) - d_{outlook}(h_1, x_1) + d_{outlook}(m_2, x_2) - \ldots\right) \\
&= \frac{1}{r}(-6 + 7) = {}^{1}/_{12}
\end{aligned}
$$

$W(temperature) = -{}^{6}/_{12} + {}^{4}/_{12} = -{}^{1}/_{6}$ ($\rightarrow W(temperature) = 0$)
$W(humidity) = -{}^{4}/_{12} + {}^{2}/_{12} = -{}^{1}/_{6}$ ($\rightarrow W(humidity) = 0$)
$W(wind) = -{}^{2}/_{12} + {}^{4}/_{12} = {}^{1}/_{6}$

Aufgabe 4: Diskretisierungsmethoden (1)

TECHNISCHE UNIVERSITÄT DARMSTADT

Gegeben sei folgende Version der Wetter-Daten mit 12 Trainings-Beispielen und 2 numerischen Attributen.

ID	outlook	temperature	humidity	windy	play
1	sunny	85	85	FALSE	no
2	rainy	70	96	FALSE	yes
3	rainy	68	80	FALSE	yes
4	rainy	65	70	TRUE	no
5	overcast	64	65	TRUE	yes
6	sunny	72	95	FALSE	no
7	sunny	69	70	FALSE	yes
8	rainy	75	80	FALSE	yes
9	sunny	75	70	TRUE	yes
10	overcast	72	90	TRUE	yes
11	overcast	81	75	FALSE	yes
12	rainy	71	91	TRUE	no

Diskretisieren Sie die beiden numerischen Attribute mit den Verfahren, die Sie in der Vorlesung kennen gelernt haben. Wählen Sie die Anzahl der Intervalle so, daß Sie die bekannten Daten erhalten könnten (drei Werte für Temperature, zwei für Humidity). Vergleichen Sie die Resultate miteinander und mit den bekannten Daten (die aus der vorherigen Aufgabe).

Aufgabe 4: Diskretisierungsmethoden
equal-width (1)

TECHNISCHE UNIVERSITÄT DARMSTADT

Bei equal-width wird versucht gleich große Intervalle zu bilden:

- Attribut “temperature”: $(max - min)/3 = (85 - 64)/3 = 21/3 = 7$

  $\rightarrow$ Die Intervalle sollten also die Größe 7 haben. Es resultieren folgende Einteilungen:

  $[64, 70], [71, 77], [78, 85]$

  ($[64, 71], [72, 78], [79, 85]$ wäre auch zulässig, man muss sich hier eine geeignete Einteilung überlegen)

  Dies ergibt dann die Intervalle $(-\infty, 70], (70, 77], (77, \infty)$

- Attribut “humidity”: $(96 - 65)/2 = \lfloor 31/2 \rfloor = 15$

  $\rightarrow$ Die Intervalle sollten also die Größe 15 haben: $[65, 80], [81, 96]$

  Dies ergibt dann die Intervalle $(-\infty, 80], (80, \infty)$.

Aufgabe 4: Diskretisierungsmethoden equal-frequency (1)

TECHNISCHE UNIVERSITÄT DARMSTADT

Bei equal-frequency wird versucht möglichst gleich viele Beispiele pro Intervall zu haben:

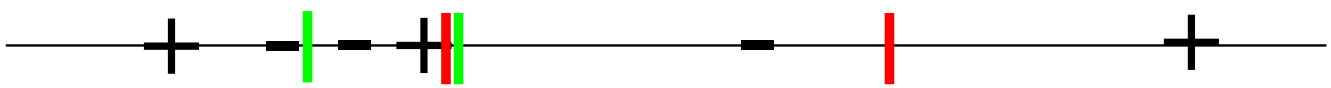
- Attribut “temperature”:$12/3 = 4$ Es sollten also jeweils 4 Beispiele in einem Intervall liegen.
  Sortieren wir die Werte, ergibt sich: 64,65,68,69 < 70,71,72,72 < 75, 75, 81 85
  Daher liegen die Beispiele
  Nr.5 temp.=64, Nr.4 temp.=65, Nr.3 temp.=68, Nr.7 temp.=69
  im ersten Intervall, die Beispiele
  Nr.2 temp.=70, Nr.12 temp.=71, Nr.6 temp.=72, Nr.10 temp.=72
  im zweiten und die Beispiele
  Nr.8 temp.=75, Nr.9 temp.=75, Nr.11 temp.=81, Nr.1 temp.=85
  im dritten Intervall.

Aufgabe 4: Diskretisierungsmethoden equal-frequency (2)

TECHNISCHE UNIVERSITÄT DARMSTADT

- Attribut “humidity”: $12/2 = 6$ Es sollten jeweils 6 Beispiele in einem Intervall liegen.
  Die Beispiele
  Nr.5 hum.=65, Nr.4 hum.=70, Nr.7 hum.=70, Nr.9 hum.=70, Nr.11 hum.=75, Nr.3 hum.=80, Nr.8 hum.=80 (die beiden Beispiele mit humidity = 80 könnten auch im zweiten Intervall liegen)
  im ersten und die Beispiele
  Nr.1 hum.=85, Nr.10 hum.=90, Nr.12 hum.=91, Nr.6 hum.=95, Nr.2 hum.=96
  im zweiten Intervall.

Die folgende Grafik erklärt den Unterschied beider Methoden nochmals anschaulich, wobei es sich hier um ein allgemeines Beispiel handelt, das nicht den vorherigen Aufgaben entspricht (die roten Striche sind für equal-width und die grünen für equal-frequency):



Aufgabe 4: Diskretisierungsmethoden
chi-merge (1)

TECHNISCHE UNIVERSITÄT DARMSTADT

Wir werden nicht auf die Berechnung jedes einzelnen χ^2-Wertes eingehen, wir werden jedoch zwei Beispiele exemplarisch berechnen und anschließend nur noch die zur Berechnung benötigten Werte und das Ergebnis auflisten.

Aufgabe 4: Diskretisierungsmethoden
chi-merge (2)

TECHNISCHE UNIVERSITÄT DARMSTADT

Zur Erinnerung der χ^2-Wert zweier Intervalle I_1 und I_2 berechnet sich wie folgt:

$$\chi^s = \sum_{i=1}^{2} \sum_{j=1}^{c} \frac{(A_{ij} - E_{ij})^2}{E_{ij}},$$

mit

- A_{ij}: Anzahl der Beispiele in I_i, die zur Klasse j gehören
- E_{ij}: zu erwartende Anzahl von Beispiele in I_i, die zur Klasse j gehören

$$E_{ij} = N_i \cdot \frac{C_j}{N_1 + N_2}$$

- N_i: Anzahl der Beispiele in I_i

$$N_i = \sum_{j=1}^{c} A_{ij}$$

- C_j: Anzahl der Beispiele in I_1 und I_2, die zur Klasse j gehören. Sollte dieser Wert 0 sein, wird diese Klasse bei der Berechnung des χ^2-Wertes nicht berücksichtigt (da Division durch 0), d.h. alle entsprechenden Terme werden bei der Berechnung auf 0 gesetzt.

$$C_j = A_{1j} + A_{2j}$$

Aufgabe 4: Diskretisierungsmethoden
chi-merge (3)

TECHNISCHE UNIVERSITÄT DARMSTADT

Anmerkungen: Der χ^2-Wert zweier Intervalle, die exakt die gleiche Klassenverteilung aufweisen, ist minimal bzw. 0.
Diese Intervalle werden somit verbunden (falls noch kein Stopkriterium, wie z.B. minimale Intervallanzahl, erfüllt ist).
D.h. natürlich auch, daß in der Initialphase reine Intervalle (also Intervalle mit nur einer auftretenden Klasse) sofort verbunden werden können.

Aufgabe 4: Diskretisierungsmethoden chi-merge, Beispiel 1 (2 Klassen) (1)

TECHNISCHE UNIVERSITÄT DARMSTADT

Intervall I_1 hat 3 positive Beispiele und 1 negatives, Intervall I_2 hat 4 positive und 2 negative Beispiele. Also haben wir folgende Werte

$$A_{1+} = 3 \qquad A_{1-} = 1 \qquad A_{2+} = 4 \qquad A_{2-} = 2$$

Hieraus berechnen wir zunächst

$$C_+ = 7 \qquad C_- = 3 \qquad N_1 = 4 \qquad N_2 = 6$$

und anschließend

$$E_{1+} = 4\frac{7}{4+6} = 2,8 \qquad E_{1-} = 4\frac{3}{4+6} = 1,2 \qquad E_{2+} = 6\frac{7}{4+6} = 4,2 \qquad E_{2-} = E_{2+} = 6\frac{3}{4+6} = 1,8$$

Mit diesen Werten können wir nun den χ^2-Wert berechnen:

$$\begin{aligned}
\chi^2 &= \frac{(3-2,8)^2}{2,8} + \frac{(1-1,2)^2}{1,2} + \frac{(4-4,2)^2}{4,2} + \frac{(2-1,8)^2}{1,8} \\
&= \frac{1}{70} + \frac{1}{30} + \frac{1}{105} + \frac{1}{45} \\
&= \frac{9+21+6+14}{630} = \frac{5}{63}
\end{aligned}$$

Aufgabe 4: Diskretisierungsmethoden chi-merge, Beispiel 2 (3 Klassen) (1)

TECHNISCHE UNIVERSITÄT DARMSTADT

Die Beispiele der 3 Klassen *a*,*b* und *c* verteilen sich auf das Intervall I_1 bzw. auf I_2 wie folgt

$$A_{1a} = 5 \qquad A_{1b} = 3 \qquad A_{1c} = 4 \qquad A_{2a} = 4 \qquad A_{2b} = 3 \qquad A_{2c} = 1$$

Hieraus berechnen wir

$$C_a = 9 \qquad C_b = 6 \qquad C_c = 5 \qquad N_1 = 12 \qquad N_2 = 8$$

und

$$E_{1a} = 12\frac{9}{12+8} = 5,4 \qquad E_{1b} = 12\frac{6}{12+8} = 3,6 \qquad E_{1c} = 12\frac{5}{12+8} = 3$$

$$E_{2a} = 8\frac{9}{12+8} = 3,6 \qquad E_{2b} = 8\frac{6}{12+8} = 2,4 \qquad E_{2c} = 8\frac{5}{12+8} = 2$$

Daraus berechnen wir

$$\begin{aligned} \chi^2 &= \frac{(5-5,4)^2}{5,4} + \frac{(3-3,6)^2}{3,6} + \frac{(4-3)^2}{3} + \frac{(4-3,6)^2}{3,6} + \frac{(3-2,4)^2}{2,4} + \frac{(1-2)^2}{2} \\ &\approx 1,16 \end{aligned}$$

Aufgabe 4: Diskretisierungsmethoden
chi-merge: Temperature (1)

TECHNISCHE UNIVERSITÄT DARMSTADT

Wir möchten mittels χ-Merge Temperature in 3 diskrete Werte aufteilen.
Zuerst sortieren wir die Werte des Attributs *Temperature*. Je nach auf- bzw. absteigender Sortierung erhält man unterschiedliche Ergebnisse. Wir entscheiden uns für eine aufsteigende Sortierung.
Reine Intervalle sind in der folgenden Tabelle bereits verbunden (siehe Anmerkungen).

I_1	I_2	A_{1y}	A_{1n}	A_{2y}	A_{2n}	N_1	N_2	C_y	C_n	E_{1y}	E_{1n}	E_{2y}	E_{2n}	χ^2
64	65	1	0	0	1	1	1	1	1	0,50	0,50	0,50	0,50	2,00
65	68-70	0	1	3	0	1	3	3	1	0,75	0,25	2,25	0,75	4,00
68-70	71	3	0	0	1	3	1	3	1	2,25	0,75	0,75	0,25	4,00
71	**72**	**0**	**1**	**1**	**1**	**1**	**2**	**1**	**2**	**0,33**	**0,67**	**0,67**	**1,33**	**0,75**
72	75-81	1	1	3	0	2	3	4	1	1,60	0,40	2,40	0,60	1,88
75-81	85	3	0	0	1	3	1	3	1	2,25	0,75	0,75	0,25	4,00

Wir wählen die Intervalle mit geringstem χ^2-Wert aus, also die Intervalle $[71, 71]$ und $[72, 72]$, und verbinden diese zu $[71, 72]$.

Aufgabe 4: Diskretisierungsmethoden
chi-merge: Temperature (2)

TECHNISCHE UNIVERSITÄT DARMSTADT

Anschließend berechnen wir die χ^2-Werte von [71, 72] und seinen benachbarten Intervalle neu. Alle anderen Werte können übernommen werden.

I_1	I_2	A_{1y}	A_{1n}	A_{2y}	A_{2n}	N_1	N_2	C_y	C_n	E_{1y}	E_{1n}	E_{2y}	E_{2n}	χ^2
64	**65**	**1**	**0**	**0**	**1**	**1**	**1**	**1**	**1**	**0,50**	**0,50**	**0,50**	**0,50**	**2**
65	68-70	0	1	3	0	1	3	3	1	0,75	0,25	2,25	0,75	4
68-70	71-72	3	0	1	2	3	3	4	2	2,00	1,00	2,00	1,00	3
71-72	75-81	1	2	3	0	3	3	4	2	2,00	1,00	2,00	1,00	3
75-81	85	3	0	0	1	3	1	3	1	2,25	0,75	0,75	0,25	4

Diesmal verschmelzen wir [64, 64] und [65, 65] zu [64, 65]und berechnen anhand der veränderten Werte die benötigten Werte (s.o.) neu.

I_1	I_2	A_{1y}	A_{1n}	A_{2y}	A_{2n}	N_1	N_2	C_y	C_n	E_{1y}	E_{1n}	E_{2y}	E_{2n}	χ^2
64-65	**68-70**	**1**	**1**	**3**	**0**	**2**	**3**	**4**	**1**	**1,6**	**0,4**	**2,4**	**0,6**	**1,88**
68-70	71-72	3	0	1	2	3	3	4	2	2	1	2	1	3
71-72	75-81	1	2	3	0	3	3	4	2	2	1	2	1	3
75-81	85	3	0	0	1	3	1	3	1	2,25	0,75	0,75	0,25	4

Der χ^2-Wert von [64, 65] und [68, 70] ist minimal, deshalb verschmelzen wir diese Intervalle zu [64, 70] und berechnen neu.

Aufgabe 4: Diskretisierungsmethoden
chi-merge: Temperature (3)

TECHNISCHE UNIVERSITÄT DARMSTADT

I_1	I_2	A_{1y}	A_{1n}	A_{2y}	A_{2n}	N_1	N_2	C_y	C_n	E_{1y}	E_{1n}	E_{2y}	E_{2n}	χ^2
64-70	**71-72**	**4**	**1**	**1**	**2**	**5**	**3**	**5**	**3**	**3,13**	**1,88**	**1,88**	**1,13**	**1,74**
71-72	75-81	1	2	3	0	3	3	4	2	2	1	2	1	3
75-81	85	3	0	0	1	3	1	3	1	2,25	0,75	0,75	0,25	4

Wir verschmelzen [64, 70] und [71, 72] und erhalten damit die folgenden Intervalle: [64, 72], [75, 81] und [85, 85].
Offensichtlich können nicht alle möglichen Temperaturen einem dieser Intervalle zugeordnet werden (siehe auch Zusatzaufgabe davor). Aus diesem Grund müssen die entstandenen Intervalle angepaßt werden. Hierfür gibt es verschiedene Möglichkeiten diese Intervalle zu bearbeiten, u.a.

- $(-\infty, 72]$, $(72, 81]$ und $(81, \infty)$ oder
- $(-\infty, 75)$, $[75, 85)$ und $[85, \infty)$ oder
- $(-\infty, 73.5)$, $[73.5, 83)$ und $[83, \infty)$ usw.

Aufgabe 4: Diskretisierungsmethoden
chi-merge: Humidity (1)

TECHNISCHE UNIVERSITÄT DARMSTADT

Für Humidity gehen wir analog vor, wobei diesmal die Diskretisierung nur 2 Werte ergeben soll.

I_1	I_2	A_{1y}	A_{1n}	A_{2y}	A_{2n}	N_1	N_2	C_y	C_n	E_{1y}	E_{1n}	E_{2y}	E_{2n}	χ^2
65	**70**	**1**	**0**	**2**	**1**	**1**	**3**	**3**	**1**	**0,75**	**0,25**	**2,25**	**0,75**	**0,44**
70	75-80	2	1	3	0	3	3	5	1	2,50	0,50	2,50	0,50	1,20
75-80	85	3	0	0	1	3	1	3	1	2,25	0,75	0,75	0,25	4,00
85	90	0	1	1	0	1	1	1	1	0,50	0,50	0,50	0,50	2,00
90	91-95	1	0	0	2	1	2	1	2	0,33	0,67	0,67	1,33	3,00
91-95	96	0	2	1	0	2	1	1	2	0,67	1,33	0,33	0,67	3,00

Es werden $[65, 65]$ und $[70, 70]$ zu $[65, 70]$ verbunden, anschließend wird neu berechnet.

I_1	I_2	A_{1y}	A_{1n}	A_{2y}	A_{2n}	N_1	N_2	C_y	C_n	E_{1y}	E_{1n}	E_{2y}	E_{2n}	χ^2
65-70	**75-80**	**3**	**1**	**3**	**0**	**4**	**3**	**6**	**1**	**3,43**	**0,57**	**2,57**	**0,43**	**0,88**
75-80	85	3	0	0	1	3	1	3	1	2,25	0,75	0,75	0,25	4,00
85	90	0	1	1	0	1	1	1	1	0,50	0,50	0,50	0,50	2,00
90	91-95	1	0	0	2	1	2	1	2	0,33	0,67	0,67	1,33	3,00
91-95	96	0	2	1	0	2	1	1	2	0,67	1,33	0,33	0,67	3,00

Aufgabe 4: Diskretisierungsmethoden
chi-merge: Humidity (2)

TECHNISCHE UNIVERSITÄT DARMSTADT

Diesmal verschmelzen wir [65, 70] und [75, 80] zu [65, 80] und berechnen neu.

I_1	I_2	A_{1y}	A_{1n}	A_{2y}	A_{2n}	N_1	N_2	C_y	C_n	E_{1y}	E_{1n}	E_{2y}	E_{2n}	χ^2
65-80	85	6	1	0	1	7	1	6	2	5,25	1,75	0,75	0,25	3,43
85	**90**	**0**	**1**	**1**	**0**	**1**	**1**	**1**	**1**	**0,50**	**0,50**	**0,50**	**0,50**	**2,00**
90	91-95	1	0	0	2	1	2	1	2	0,33	0,67	0,67	1,33	3,00
91-95	96	0	2	1	0	2	1	1	2	0,67	1,33	0,33	0,67	3,00

Wir wählen [85, 85] und [90, 90] aus und verschmelzen sie zu [85, 90]. Anschließend berechnen wir die fehlenden Werte.

I_1	I_2	A_{1y}	A_{1n}	A_{2y}	A_{2n}	N_1	N_2	C_y	C_n	E_{1y}	E_{1n}	E_{2y}	E_{2n}	χ^2
65-80	**85-90**	**6**	**1**	**1**	**1**	**7**	**2**	**7**	**2**	**5,44**	**1,56**	**1,56**	**0,44**	**1,15**
85-90	91-95	1	1	0	2	2	2	1	3	0,5	1,5	0,5	1,5	1,33
91-95	96	0	2	1	0	2	1	1	2	0,67	1,33	0,33	0,67	3

Es werden [65, 80] und [85, 90] zu [65, 90] verschmolzen und die benötigten Werte berechnet.

Aufgabe 4: Diskretisierungsmethoden
chi-merge: Humidity (3)

TECHNISCHE UNIVERSITÄT DARMSTADT

I_1	I_2	A_{1y}	A_{1n}	A_{2y}	A_{2n}	N_1	N_2	C_y	C_n	E_{1y}	E_{1n}	E_{2y}	E_{2n}	χ^2
65-90	91-95	7	2	0	2	9	2	7	4	5,73	3,27	1,27	0,73	4,28
91-95	**96**	**0**	**2**	**1**	**0**	**2**	**1**	**1**	**2**	**0,67**	**1,33**	**0,33**	**0,67**	**3**

Als letztes verschmelzen wir $[91, 95]$ und $[96, 96]$ und erhalten als Ergebnis die Intervalle $[65, 90]$ und $[91, 96]$.

Aufgabe 4: Diskretisierungsmethoden
Entropy-split (1)

TECHNISCHE UNIVERSITÄT DARMSTADT

Bei Entropy-split zerlegen wir eines der bestehenden Intervalle in jeweils zwei Teilintervalle, falls das zu zerlegende Intervall den niedrigsten E-Score (Berechnung von Entropy-split ist äquivalent zu diesem Maß) aufweist.
E-Score ist wiederum äquivalent zu Information-Gain. Beide Maße und deren Berechnung wurden bereits ausführlich in den Übungen über Entscheidungsbäume besprochen. Aus diesem Grund werden wir nicht weiter auf deren Berechnung eingehen.

Aufgabe 4: Diskretisierungsmethoden
Entropy-split: Temperature (1)

TECHNISCHE UNIVERSITÄT DARMSTADT

Wir sortieren zunächst die Attributwerte (aufsteigend).

	A<Wert		A≥Wert		
Wert	positiv	negativ	positiv	negativ	E-Score
64	0	0	8	4	0,918
65	1	0	7	4	0,867
68	1	1	7	3	0,901
69	2	1	6	3	0,918
70	3	1	5	3	0,907
71	4	1	4	3	0,876
72	4	2	4	2	0,918
75	5	3	3	1	0,907
81	7	3	1	1	0,901
85	8	3	0	1	**0,775**

Der Teilungspunkt 85 hat den besten (niedrigsten) E-Score. Wir teilen das Intervall $(-\infty, \infty)$ in $(-\infty, 85)$ und $[85, \infty)$ auf.

Aufgabe 4: Diskretisierungsmethoden
Entropy-split: Temperature (2)

TECHNISCHE UNIVERSITÄT DARMSTADT

	A<Wert		A≥Wert		
Wert	positiv	negativ	positiv	negativ	E-Score
64	0	0	8	3	0,845
65	1	0	7	3	0,801
68	1	1	7	3	0,807
69	2	1	6	2	0,840
70	3	1	5	2	0,844
71	4	1	4	2	0,829
72	4	2	4	1	0,829
75	5	3	3	0	**0,694**
81	7	3	1	0	0,801
85	0	0	0	1	-

Jetzt ist 75 der beste Teilungspunkt.
Wir zerlegen das Intervall $(-\infty, 85)$ in $(-\infty, 75)$ und $[75, 85)$. Demnach sieht die Diskretisierung von Temperature wie folgt aus: $(-\infty, 75)$, $[75, 85)$ und $[85, \infty)$.

Aufgabe 4: Diskretisierungsmethoden
Entropy-split: Humidity (1)

TECHNISCHE UNIVERSITÄT DARMSTADT

	A<Wert		A≥Wert		
Wert	positiv	negativ	positiv	negativ	E-Score
65	0	0	8	3	0,918
70	1	0	7	4	0,867
75	3	1	5	3	0,907
80	4	1	4	3	0,876
85	6	1	2	3	**0,750**
90	6	2	2	2	0,874
91	7	2	1	2	0,803
95	7	3	1	1	0,867
96	7	4	1	0	0,801

Wir teilen das Intervall $(-\infty, \infty)$ in $(-\infty, 85)$ und $[85, \infty)$ auf.

Aufgabe 4: Diskretisierungsmethoden
Vergleich (1)

TECHNISCHE UNIVERSITÄT DARMSTADT

Nun gehen wir noch kurz auf den Vergleich der Ergebnisse ein:
Da man nicht weiß, wie die numerischen Daten des originalen Datensatzes diskretisiert wurden, kann man hier keine genaue Aussage treffen. Daher kann man sich nur die Unterschiede zwischen den einzelnen Methoden und den Originaldaten anschauen (diese sind rot markiert):

ID	outlook	temperature	equal-width	equal-frequency	chi-merge	entropy-split
1	sunny	hot	hot	hot	hot	hot
2	rainy	mild	cool	mild	cool	cool
3	rainy	cool	cool	cool	cool	cool
4	rainy	cool	cool	cool	cool	cool
5	overcast	cool	cool	cool	cool	cool
6	sunny	mild	mild	mild	cool	cool
7	sunny	cool	cool	cool	cool	cool
8	rainy	mild	mild	hot	mild	mild
9	sunny	mild	mild	hot	mild	mild
10	overcast	mild	mild	mild	cool	cool
11	overcast	hot	hot	hot	mild	mild
12	rainy	mild	mild	mild	cool	cool

Aufgabe 4: Diskretisierungsmethoden
Vergleich (2)

TECHNISCHE UNIVERSITÄT DARMSTADT

ID	outlook	humidity	equal-width	equal-frequency	chi-merge	entropy-split
1	sunny	high	high	high	normal	high
2	rainy	high	high	high	high	high
3	rainy	normal	normal	normal	normal	normal
4	rainy	normal	normal	normal	normal	normal
5	overcast	normal	normal	normal	normal	normal
6	sunny	high	high	high	high	high
7	sunny	normal	normal	normal	normal	normal
8	rainy	normal	normal	normal	normal	normal
9	sunny	normal	normal	normal	normal	normal
10	overcast	high	high	high	normal	high
11	overcast	normal	normal	normal	normal	normal
12	rainy	high	high	high	high	high