

Ausarbeitung

zum Seminar

Imperfektion und Datenbanken
Thema: Rough Sets

von Christian Köllner
Betreuer: Philipp Bender

WS 03/04, Universität Karlsruhe

Inhaltsverzeichnis

Einführung	3
Informations- und Entscheidungssysteme.....	3
Ununterscheidbarkeit	4
Approximation von Mengen.....	4
Redukte	6
B-positive Region	7
Allgemeine Entscheidung	8
Entscheidungsregeln.....	8
Umgang mit Regelkonflikten	11
Bildung optimaler Entscheidungsregeln.....	11
Literaturverweise.....	12

Einführung

Die Rough Set-Theorie wurde in den frühen 80er Jahren von Zdzislaw Pawlak entwickelt. Sie beschäftigt sich mit der Klassifizierung und Analyse von Datentabellen. Die Daten können von Messungen oder menschlichen Experten stammen. Obwohl sie prinzipiell diskret vorliegen müssen, existieren Methoden, die die Verarbeitung von kontinuierlichen Werten erlauben. Das Hauptziel der Rough Set-Analyse ist die Gewinnung einer Annäherung, die die den Daten zugrunde liegenden Konzepte beschreiben soll.

Es gibt zweierlei Gründe, solche Annäherungen zu entwickeln. In manchen Fällen könnte das Ziel sein, Einblick in ein Problem zu bekommen, indem man das konstruierte Modell manuell analysiert. In anderen Anwendungen sind das Modell selbst und die in ihm enthaltenen Konzepte von zweitrangiger Bedeutung. Vielmehr geht es darum, eine Klassifizierung zu erhalten, die bisher unbekannte Objekte richtig einordnet.

Der Modellierungsprozess besteht typischerweise aus einigen Stufen, die eine geeignete Parametrisierung und Feineinstellung erfordern. Um diese Schritte ausführen zu können, ist eine Umgebung notwendig, die eine interaktive Verarbeitung und Verwaltung der Daten zulässt. Methoden der Rough Set-Theorie sind in diversen Toolkits implementiert, die eine interaktive Modellentwicklung erlauben. Es gibt eine Reihe von Softwarepaketen, die auf Rough Sets basieren.

Informations- und Entscheidungssysteme

Eine Datenmenge wird durch eine Tabelle repräsentiert, bei der jede Zeile für einen Fall, ein Ereignis, einen Patienten oder einfach nur ein Objekt steht. Jede Spalte steht für ein Attribut (eine Variable, eine Beobachtung, eine Eigenschaft, usw.), das für jedes Objekt gemessen werden kann. Es kann aber auch von einem menschlichen Experten oder Benutzer stammen. Eine solche Tabelle wird Informationssystem genannt. Formal handelt es sich um ein Paar $\mathbf{A} = (U, A)$, wobei U eine Menge von Objekten - das Universum - ist und A eine nichtleere, endliche Menge von Attributen, so dass $a:U \rightarrow V_a$ für jedes $a \in A$. Die Menge V_a heißt Wertemenge von a .

Möchte man in einem Informationssystem das Universum anhand einer zu treffenden Entscheidung klassifizieren, ist es sinnvoll, das Entscheidungsattribut d gesondert zu betrachten. Ein Paar $\mathbf{A} = (U, A \cup \{d\})$ mit Entscheidungsattribut $\{d\}$ heißt dann Entscheidungssystem (die restlichen Bezeichnungen wie oben).

	<i>Attribute</i>			<i>Entscheidung Grippe</i>
	<i>Kopfschmerzen</i>	<i>Muskelschmerzen</i>	<i>Temperatur</i>	
e1	ja	ja	normal	nein
e2	ja	ja	hoch	ja
e3	ja	ja	sehr hoch	ja
e4	nein	ja	normal	nein
e5	nein	nein	hoch	nein
e6	nein	ja	sehr hoch	ja

Abb. 1: *Beispiel eines Entscheidungssystems*

Ununterscheidbarkeit

In einem Informationssystem $\mathbf{A} = (U, A)$ kann man für eine Teilmenge $B \subseteq A$ eine Äquivalenzrelation $IND_A(B) = \{ (x, x') \in U^2 \mid \forall a \in B \ a(x) = a(x') \}$ definieren. Sie heißt B-Ununterscheidbarkeitsrelation. Der Name rührt daher, dass all die Elemente in U als äquivalent erklärt werden, die sich bezüglich der Attributmeng B nicht unterscheiden.

In *Abb. 1* induziert $IND(\{ Grippe \})$ z.B. die Partitionierung $\{ \{ e1, e4, e5 \}, \{ e2, e3, e6 \} \}$.

Approximation von Mengen

Eine Äquivalenzrelation induziert eine Partitionierung des Universums. Die Partitionen können benutzt werden, um neue Untermengen des Universums zu bilden. Untermengen, die dabei am ehesten interessieren, sind die, die im Entscheidungsattribut den gleichen Wert haben. Es kann jedoch passieren, dass ein Konzept wie z.B. „Grippe“ nicht scharf definiert werden kann.

	<i>Attribute</i>			<i>Entscheidung Grippe</i>
	<i>Kopfschmerzen</i>	<i>Muskelschmerzen</i>	<i>Temperatur</i>	
e1	ja	ja	hoch	Ja
e2	ja	ja	hoch	Ja
e3	ja	ja	sehr hoch	Ja
e4	ja	ja	sehr hoch	Nein
e5	nein	nein	hoch	nein
e6	nein	nein	normal	ja

Abb. 2: *„Grippe“ lässt sich aus den gegebenen Parametern nicht scharf definieren.*

Zum Beispiel kann die Menge der Patienten mit Grippe nicht scharf mit den in *Abb. 2* definierten Attributen eingegrenzt werden. Die „problemati-

schen“ Patienten sind $e3$ und $e4$. Mit anderen Worten ist es nicht möglich, eine scharfe Definition solcher Patienten über die gegebene Tabelle zu induzieren. Hier tritt der Begriff der Rough Sets zum Vorschein. Obwohl man diese Patienten nicht scharf definieren kann, ist es möglich, Patienten mit einem sicherlich positiven Ergebnis von Patienten mit sicher negativem Ergebnis und Patienten im Grenzfall zu unterscheiden. Die Grenze zwischen beiden Fällen ist nicht leer, die Menge ist unscharf. Im Folgenden werden die Begriffe formalisiert. Sei $\mathbf{A} = (U, A)$ ein Informationssystem und sei $B \subseteq A$ und $X \subseteq U$. Man kann X annähern, indem man die untere und obere B -Approximation von X bildet. Man bezeichnet diese Annäherungen mit $\underline{B}X$ bzw. $\overline{B}X$, wobei $\underline{B}X = \{x \mid [x]_B \subseteq X\}$ und $\overline{B}X = \{x \mid [x]_B \cap X \neq \emptyset\}$.

Die Objekte in $\underline{B}X$ können unter Kenntnis von B mit Sicherheit als Elemente von X klassifiziert werden, während die Elemente von $\overline{B}X$ unter Kenntnis von B nur als mögliche Elemente von X klassifiziert werden können. Die Menge $BN_B(X) = \overline{B}X - \underline{B}X$ bezeichnet man als B -Grenzregion von X , sie besteht also aus den Objekten, die man unter Kenntnis von B nicht sicher X zurechnen kann. Die Menge $U - \overline{B}X$ bezeichnet man mit B -außen-Region von X ; sie besteht aus denjenigen Objekten, die als mit Sicherheit nicht zu X gehörig klassifiziert werden können. A heißt unscharf bzw. scharf, wenn die Grenzregion nichtleer bzw. leer ist.

Man kann folgende Eigenschaften dieser Approximationen aufzeigen:

1. $\underline{B}(X) \subseteq X \subseteq \overline{B}(X)$,
2. $\underline{B}(\emptyset) = \overline{B}(\emptyset) = \emptyset, \underline{B}(U) = \overline{B}(U) = U$,
3. $\overline{B}(X \cup Y) = \overline{B}(X) \cup \overline{B}(Y)$,
4. $\underline{B}(X \cap Y) = \underline{B}(X) \cap \underline{B}(Y)$,
5. $X \subseteq Y$ impliziert, dass $\underline{B}(X) \subseteq \underline{B}(Y)$ und $\overline{B}(X) \subseteq \overline{B}(Y)$,
6. $\underline{B}(X \cup Y) \supseteq \underline{B}(X) \cup \underline{B}(Y)$,
7. $\overline{B}(X \cap Y) \subseteq \overline{B}(X) \cap \overline{B}(Y)$,
8. $\underline{B}(-X) = -\underline{B}(X)$,
9. $\overline{B}(-X) = -\overline{B}(X)$,
10. $\underline{B}(\underline{B}(X)) = \overline{B}(\underline{B}(X)) = \underline{B}(X)$,
11. $\overline{B}(\overline{B}(X)) = \underline{B}(\overline{B}(X)) = \overline{B}(X)$,

wobei \bar{X} definiert ist als $U - X$.

Die obere und untere B-Approximation sollen anhand eines Beispiels visualisiert werden. Sei ein Entscheidungssystem durch Abb. 3 gegeben.

Die zu klassifizierende Menge X sei die Menge aller gesunden Patienten, also $X := \{x \mid krank(x) = Nein\} = \{x1, x3\}$.

Anhand des Attributs „Temperatur“ soll klassifiziert werden, also $B := \{Temperatur\}$. Dann lassen die

oben definierten Approximationsmengen durch nachfolgende Grafik voranschaulichen:

x	Temperatur	krank
x1	37 - 38	Nein
x2	38 - 39	Ja
x3	38 - 39	Nein
x4	> 39	Ja

Abb. 3: Ein einfaches Entscheidungssystem

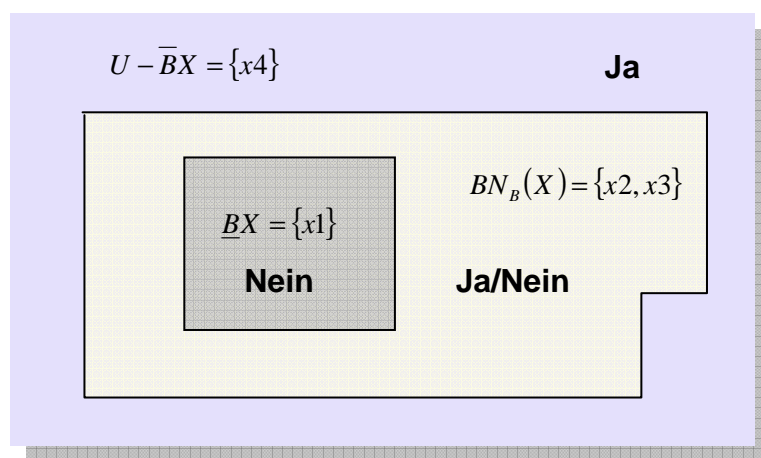


Abb. 4: Visualisierung der unteren B-Approximation, der B-Grenzregion und der B-Außenregion.

Redukte

Bei der Betrachtung von Informationssystemen kann sich herausstellen, dass bestimmte Attribute überflüssig sind. Auch zur kompakteren und stabileren Analyse eines Informationssystems kann es sinnvoll sein, auf einige Attribute zu verzichten.

Im folgenden Entscheidungssystem reichen die Attribute „Abschluss“ und „Französisch“ vollkommen aus, um das immer noch Universum anhand des Attributs „Entscheidung“ zu klassifizieren:

	Abschluss	Erfahrung	Französisch	Referenz	Entscheidung
x1	MBA	mittel	ja	exzellent	einstellen
x2	MBA	wenig	ja	neutral	ablehnen
x3	MCE	wenig	ja	gut	Ablehnen

x4	MSc	hoch	ja	neutral	einstellen
x5	MSc	mittel	ja	neutral	ablehnen
x6	MSc	hoch	ja	exzellent	einstellen
x7	MBA	hoch	nein	gut	einstellen
x8	MCE	wenig	nein	exzellent	ablehnen

Abb. 5: Entscheidungssystem mit verzichtbaren Attributen

Eine nichtleere Teilmenge B der Attributmenge A heißt Redukt. Redukste, die die Partitionierung des Universums nicht ändern, sind von besonderem Interesse und heißen optimal. Die Ermittlung optimaler Redukste kann algorithmisch durch Reduktion auf einen Algorithmus zur Minimierung boolescher Funktionen geschehen. Dieses Problem ist NP-hart, es existieren allerdings gute Heuristiken. Im Folgenden soll ein Algorithmus zur Bestimmung optimaler Redukste vorgestellt werden.

Gegeben sei ein Informationssystem $\mathbf{A} = (U, A)$. Ausgangspunkt ist die Unterscheidbarkeitsmatrix $((c_{ij}))$ mit $c_{ij} = \{a \in A \mid a(x_i) \neq a(x_j)\}$ $i, j = 1, \dots, n$.

Interpretieren lässt sich diese Matrix als $c_{ij} =$ „Menge der Attribute, in denen sich x_i von x_j unterscheidet“. Als Nächstes wird zu jedem Attribut a_i eine boolesche Variable a_i^* definiert. Mit Hilfe dieser neuen Variablen bildet man die Matrix $((c_{ij}^*))$ mit $c_{ij}^* = \{a^* \mid a \in c_{ij}\}$. Hieraus konstruiert man die Funktion $f_A(a_1^*, \dots, a_m^*) = \bigwedge \{ \bigvee c_{ij}^* \mid 1 \leq j \leq i \leq n, c_{ij} \neq \emptyset \}$. Die Menge der Primimplikanten von f_A bestimmt die Menge aller optimalen Redukste dergestalt, dass die Attribute, die mit den im Primimplikanten vorkommenden booleschen Variablen korrespondieren, ein mögliches optimales Redukt bilden.

B-positive Region

In einem Entscheidungssystem induziert das Entscheidungsattribut eine Partition des Universums über $CLASS_A(d) = \{X_A^1, \dots, X_A^{r(d)}\}$, wobei $r(d)$ die Anzahl der Werte ist, die das Entscheidungsattribut d annehmen kann. Die Menge X_A^i heißt i -te Entscheidungsklasse von A . Sind $X_1 \dots X_{r(d)}$ die Entscheidungsklassen

von A, so heißt die Menge $POS_B(d) = \underline{B}X_1 \cup \dots \cup \underline{B}X_{r(d)}$ B-positive Region von A. Sie lässt sich interpretieren als die Menge der Objekte, die sich eindeutig einer Entscheidung zuordnen lassen.

Im Beispiel besteht die $\{Temperatur\}$ -positive Region des Entscheidungssystems aus der Menge $\{x1, x4\}$, da sich nur diese eindeutig klassifizieren lassen. Die Elemente x2 und x3 besitzen für das Attribut „Temperatur“ unterschiedliche Entscheidungswerte und gehören somit nicht in die B-positive Region.

x	Temperatur	krank
x1	37 - 38	Nein
x2	38 - 39	Ja
x3	38 - 39	Nein
x4	> 39	Ja

$$POS_{\{Temperatur\}}(d) = \{x1, x4\}$$

Abb. 6: $\{Temperatur\}$ -positive Region des gegebenen Entscheidungssystems

Allgemeine Entscheidung

Die allgemeine Entscheidung in einem Entscheidungssystem A ist gegeben durch die Abbildung

$$\begin{aligned} \partial_A : U &\rightarrow 2^{V_d} \\ x &\mapsto \{i \mid \exists x' \in U : x' IND(A) x \wedge d(x') = i\} \end{aligned}$$

Sie lässt sich interpretieren als die Menge aller Entscheidungswerte, die x und die zu x äquivalenten Objekte annehmen. Ein Entscheidungssystem heißt konsistent (deterministisch), falls $|\partial_A(x)| = 1 \forall x \in U$. In diesem Falls gilt $POS_A(d) = U$. Ein Entscheidungssystem heißt inkonsistent (indeterministisch), falls $\neg(|\partial_A(x)| = 1 \forall x \in U) \Leftrightarrow \exists x \in U : |\partial_A(x)| \neq 1$. In diesem Fall gilt $POS_A(d) \neq U$.

Entscheidungsregeln

Ziel der Analyse eines Entscheidungssystems ist meist die Konstruktion allgemeiner, einfacher Entscheidungsregeln, die die Klassifizierung neuer, nicht im Entscheidungssystem vorkommender Objekte erlauben. Es soll ein Algorithmus vorgestellt werden, der aus einem gegebenen Entscheidungssystem Entscheidungsregeln konstruieren kann.

Ausgangspunkt ist die entscheidungs-relative Unterscheidbarkeitsmatrix $M^d(A)$ des Entscheidungssystems. Sie ist gegeben durch $M^d(A) = ((c_{ij}^d))$ mit

$$c_{ij}^d = \begin{cases} \phi, & \text{falls } d(x_i) = d(x_j) \\ \{a \in A \mid a(x_i) \neq a(x_j)\}, & \text{sonst} \end{cases}$$

Für jedes k im Bereich $1 \dots |U|$ wird die (k, d) -relative Unterscheidungsfunktion $f_A^{d,k}$ konstruiert. Dies ist eine boolesche Funktion, gegeben durch

$$f_A^{d,k}(a_1^*, \dots, a_m^*) = \wedge \{ \vee c_{ik}^* \mid 1 \leq i \leq n, c_{ik} \neq \phi \}$$

Aus den Primimplikanten dieser Funktionen lassen sich Entscheidungsregeln konstruieren.

Die Funktionsweise des Algorithmus soll am folgenden Beispiel erläutert werden. Gegeben sei folgendes Entscheidungssystem:

	Kopf-schmerzen [k]	Temperatur [t]	Grippe
e1	ja	normal	nein
e2	ja	hoch	ja
e3	ja	sehr hoch	ja
e4	nein	normal	nein
e5	nein	hoch	nein
e6	nein	sehr hoch	ja
e7	nein	hoch	ja
e8	nein	sehr hoch	nein

Abb. 7: Ein Entscheidungssystem

Im ersten Schritt wird die entscheidungs-relative Unterscheidbarkeitsmatrix gebildet:

	e1	e2	e3	e4	e5	e6	e7	e8
e1	∅	{t}	{t}	∅	∅	{k,t}	{k,t}	∅
e2	{t}	∅	∅	{k,t}	{k}	∅	∅	{k,t}
e3	{t}	∅	∅	{k,t}	{k,t}	∅	∅	{k}
e4	∅	{k,t}	{k,t}	∅	∅	{t}	{t}	∅
e5	∅	{k}	{k,t}	∅	∅	{t}	∅	∅
e6	{k,t}	∅	∅	{t}	{t}	∅	∅	∅
e7	{k,t}	∅	∅	{t}	∅	∅	∅	{t}
e8	∅	{k,t}	{k}	∅	∅	∅	{t}	∅

Abb. 8: entscheidungs-relative Unterscheidbarkeitsmatrix

Der zweite Schritt besteht darin, die (k,d) -relativen Unterscheidungsfunktionen zu konstruieren:

$$f_A^{d,1} = t \wedge t \wedge (k \vee t) \wedge (k \vee t) = t$$

$$f_A^{d,2} = \dots = k \wedge t$$

$$f_A^{d,3} = k \wedge t$$

$$f_A^{d,4} = t$$

$$f_A^{d,5} = k \wedge t$$

$$f_A^{d,6} = t$$

$$f_A^{d,7} = t$$

$$f_A^{d,8} = k \wedge t$$

Ein Primimplikant der i -ten Unterscheidungsfunktion in konjunktiver Form enthält dann genau die Variablen, deren zugehörige Attribute zur Konstruktion einer Entscheidungsregel für die i -te Zeile des Entscheidungssystems herangezogen werden müssen.

Im gegebenen Beispiel sind die Primimplikanten der Unterscheidungsfunktionen identisch mit den Funktionen selbst.

Mit Hilfe des Entscheidungssystems und den Unterscheidungsfunktionen lassen sich für dieses Beispiel folgende Entscheidungsregeln gewinnen:

R1: Wenn Temperatur=normal, folgt Grippe=nein

R2: Wenn Kopfschmerzen=ja und Temperatur=hoch, folgt Grippe=ja

R3: Wenn Kopfschmerzen=ja und Temperatur=sehr hoch, folgt Grippe=ja

R4: Wenn Temperatur=normal, folgt Grippe=nein

R5: Wenn **Kopfschmerzen=nein** und **Temperatur=hoch**, folgt **Grippe=nein**
R6: Wenn **Temperatur=sehr hoch**, folgt **Grippe=ja**
R7: Wenn **Temperatur=hoch**, folgt **Grippe=ja**
R8: Wenn **Kopfschmerzen=nein** und **Temperatur=sehr hoch**, folgt **Grippe=nein**

Aus der Betrachtung der gewonnenen Regeln kann man feststellen, dass diese Regeln noch einige Unregelmäßigkeiten enthalten. So ist z.B. R3 ein Implikat von R6, was bedeutet, dass R3 überflüssig ist. In einer Nachbearbeitung der Entscheidungsregeln können überflüssige Regeln erkannt und eliminiert werden. Gravierender ist der Widerspruch zwischen R6 und R8. Für die Attributkombination „**Kopfschmerzen=nein**“ und „**Temperatur=sehr hoch**“ liefern sie unterschiedliche Entscheidungen. Hier spiegeln sich die Inkonsistenzen des gegebenen Entscheidungssystems direkt in der Regelbildung wider. Abschließend ist man jedoch an einer einzigen, eindeutigen Entscheidungsfindung interessiert; man muss also einen Weg finden, mit Entscheidungskonflikten umzugehen.

Umgang mit Regelkonflikten

Der Support einer Regel R ist die Anzahl der Objekte des Universums, die mit R klassifiziert werden. Ein hoher Support spricht somit für eine wichtige und (wahrscheinlich) korrekte Regel.

Tritt bei der Entscheidungsfindung ein Konflikt auf, so wird die Entscheidung über einen Voting-Mechanismus getroffen. Hierbei wird zu jeder zutreffenden Regel der Support bestimmt. Die optimale Entscheidung wird über eine Gewichtung der möglichen Entscheidungen anhand der zugehörigen Regel-Supporte gefällt.

Ebenfalls denkbar ist der Fall, dass für eine bestimmte Attributkombination überhaupt keine zutreffende Regel existiert. Soll ein solches Objekt klassifiziert werden, nimmt man im Allgemeinen die im Entscheidungssystem am häufigsten vorkommende Entscheidung.

Bildung optimaler Entscheidungsregeln

In Entscheidungsregeln können weitere nicht wünschenswerte Effekte auftreten. Enthält das zugrunde liegende Entscheidungssystem sehr viele Attribute, können die auftretenden Regeln sehr komplex und „verrauscht“ werden. Das bedeutet, dass die vielen Konjunktionen eine solche Regel nur künstlich präzise machen. Bei der Klassifikation neuer Objekte treten dann erhebliche Probleme auf. Um kompaktere, einfachere Regeln zu erhalten, betrachtet man häufig statt der ganzen Attributmenge nur ein Redukt der Attribute und bildet die Entscheidungsregeln nur über dieses Redukt. Man erhofft sich so sogar präzisere (obwohl kürzere) Regeln. Die Grundidee ist, dass das im Entscheidungssystem implizit enthaltene Konzept in den meisten Fällen nicht sonderlich kompliziert ist – also durch einfache Regeln nachgebildet werden kann. Längliche, komplexe Regeln sprechen also eher für eine Verrauschung. Kurze Regeln sind vorzuziehen. Es bleibt die Frage, wie man ein Redukt finden kann, das zu „guten“ Entscheidungsregeln führt. Vorgestellt wurde bereits ein Algorithmus, der

optimale Redukte eines Entscheidungssystems bilden kann, d.h. Redukte, die die Klassifizierung aufrecht erhalten. In der Praxis ist die Forderung nach einer konstant bleibenden Klassifizierung jedoch oft zu stark. Lässt man sie fallen, liefern die folgenden Ansätze brauchbare Ergebnisse.

- **Dynamische Redukte**

Man bildet klassenerhaltende Redukte über zufällige Teilmengen des Universums. In das endgültige Redukt werden die am häufigsten vorkommenden Attribute übernommen [2].

- **Berücksichtigung von Abhängigkeiten zwischen Redukten**

Wenn ein bestimmter Attributwert eines Attributs A sehr oft einen bestimmten Attributwert eines Attributs B nach sich zieht, spricht man von einer Implikation $A \Rightarrow B$. Man kann solche Abhängigkeiten herausfinden, analysieren und ihre Stärke bestimmen [2]. Hängt ein Attribut B nun hochgradig von einem anderen Attribut A ab, so macht es evtl. Sinn, A nicht in das endgültige Redukt mit aufzunehmen.

- **Redukte durch relevante Attribute**

Zu jedem Attribut eines Entscheidungssystems lässt sich eine Maßzahl für dessen Relevanz bestimmen, indem man berechnet, wie stark sich die Klassifizierung des Universums unter Weglassen dieses Attributs ändern würde [2]. Das Redukt besteht dann nur aus den relevantesten Attributen.

Diskretisierung kontinuierlicher Attributwerte

Die Wertemengen von Attributen müssen bei Rough Set – Verfahren stets diskret vorliegen. In der Praxis ist dies jedoch oft nicht der Fall. Darum existieren Diskretisierungsalgorithmen, die eine gegebene Wertemenge unter Berücksichtigung der Attributwerte auf eine Menge von Intervallen abbilden und dabei die Klassifizierung des Universums erhalten [2].

Literaturverweise

[1] Z. Pawlak, J. Grzymala-Busse, R. Slowinski, W. Ziarko: Rough Sets

[2] J. Komorowski, Z. Pawlak, L. Polkowski, A. Skowron: Rough Sets: A Tutorial