



Seminar aus maschinellem Lernen

Combining with Nearest Neighbor



Inhalt

- Klassifikation nicht abgedeckter Beispiele mithilfe von Rule Stretching
 - Motivation
 - Definition
 - Algorithmus
 - Experimentelle Evaluation
- Hybride Algorithmen mit instanzbasierter Klassifikation
 - Motivation
 - k -NN, Maximum-entropy Modeling
 - Hybride Algorithmen
 - Experimentelle Evaluation
 - Analyse
- Vergleich der beiden Ansätze



Rule Stretching

Motivation

2

- bei einer geordneten Regelmengemenge (decision list):
 - keine Klassifikationskonflikte aufzulösen
 - erste feuerrnde Regel führt zur Klassifikation
 - letzte zu prüfende Regel fungiert als Default-Regel
- Ein Beispiel wird immer abgedeckt
- bei einer ungeordneten Regelmengemenge:
 - Strategien zur Lösung von Klassifikationskonflikten erforderlich
 - falls keine der Regeln feuert ist eine Strategie zur Klassifikation notwendig (zumeist wird die am häufigsten vorkommende Klasse gewählt)
- Wird ein Beispiel von keiner der Regeln abgedeckt, so kommt es mithilfe einer Strategie zur Zuweisung einer Klasse

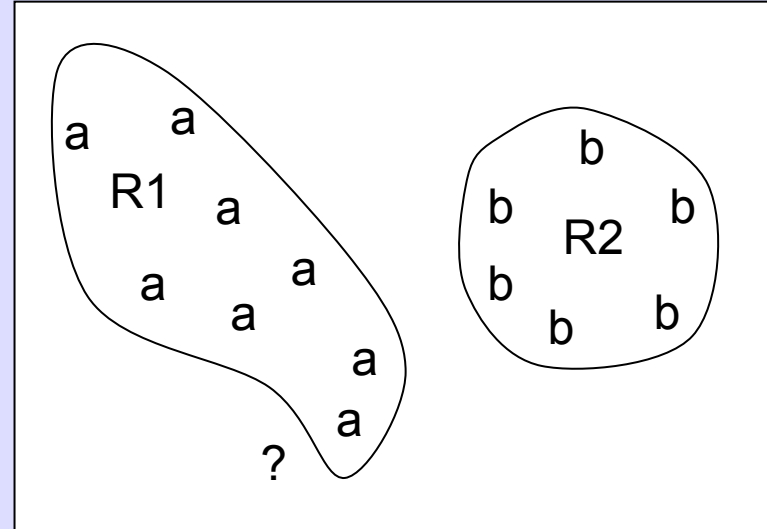


Rule Stretching

Motivation

Szenario:

Sei $H = \{R1, R2\}$ eine ungeordnete Regelmengung. Nach der Anwendung der Regeln R1 und R2 zur Klassifikation der Beispiele bleiben teilweise Beispiele unabgedeckt (hier, '?').



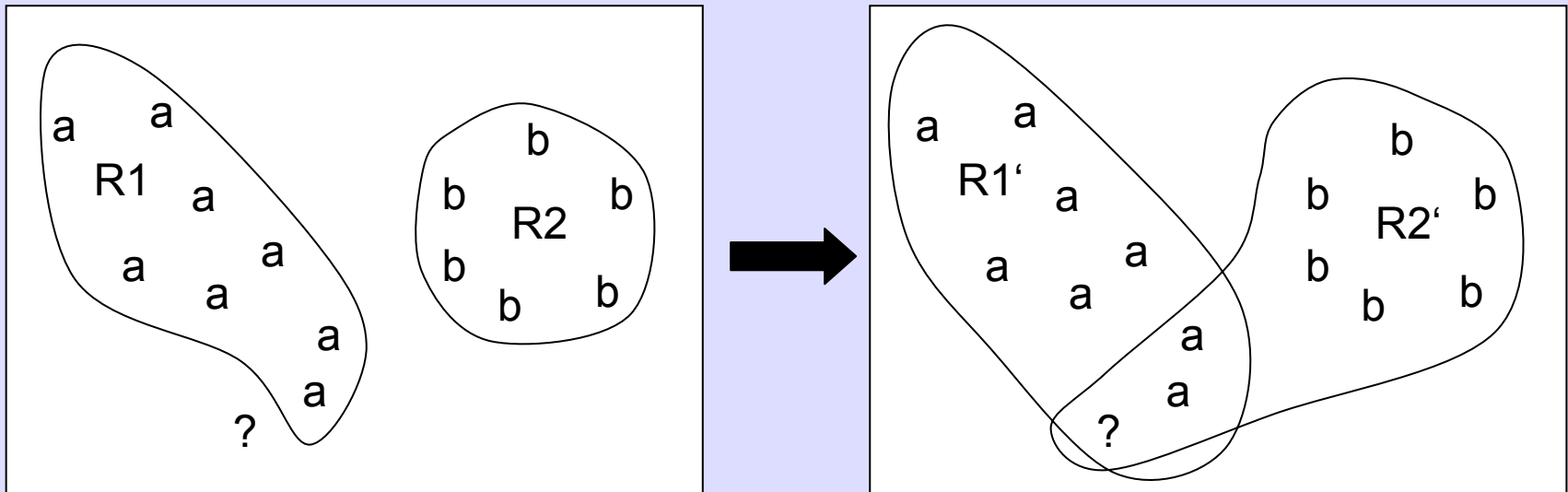
Wie wird nun Beispiel, '?' klassifiziert?
Neuer Ansatz: Rule Stretching



Rule Stretching

Motivation

Die ungeordneten Regeln der ermittelten Hypothese können so „auseinandergedehnt“ werden, dass bislang nicht abgedeckte Beispiele klassifiziert werden. Dies geschieht durch Generalisierung der Regeln.



Durch welche Regel wird ‚?‘ nun klassifiziert? Vergleich der Accuracy → R1‘



Rule Stretching

Prinzipieller Ablauf

5

1. Ermitteln einer ungeordneten Regelmenge
2. Klassifikation neuer Beispiele durch die ermittelte Regelmenge
3. Beispiele die von keiner der Regeln abgedeckt werden an das Rule Stretching System übergeben



Rule Stretching

Algorithmus

6

Generelle Version des Rule Stretching Algorithmus:

Input: Hypothese H , Hintergrundwissen B , Beispielmenge E und ein nicht abgedecktes Beispiel e

Output: Klassenlabel c

1. $H' = \{r' \mid r \in H \wedge r' = \text{minimal_generalisation}(r, e)\}$
2. $V = \{(r, v) \mid r \in H' \wedge v = \text{coverage}(r, E)\}$
3. $c = \text{classify}(V)$



Rule Stretching

Definition

7

Definition 1. Ein Atom c ist eine Generalisierung der Atome a und b , falls Substitutionen θ_1 und θ_2 existieren, sodass $c\theta_1 = a$ und $c\theta_2 = b$.

Definition 2. Eine Generalisierung c zweier Atome a und b ist eine „least general generalisation“ (*lgg*), falls für jede andere Generalisierung c_i von a und b Substitutionen θ_i existieren, sodass $c = c_i\theta_i$.



Rule Stretching

Algorithmus

Ein spezieller Rule Stretching Algorithmus:

Input: Hypothese H , Hintergrundwissen B , Beispielmenge E und ein nicht abgedecktes Beispiel e

Output: Klassenlabel c

1. $H' = \{r' \mid r \in H \wedge r' = lgg(r, e)\}$
2. $V = \{(r, v) \mid r \in H' \wedge v = coverage(r, E)\}$
3. $c = use_best_rule(V)$



Rule Stretching

Experiment

9

Die Experimente wurden mithilfe von „Virtual Predict“, einem *inductive logic programming system*, durchgeführt. Die für die Studie relevanten Parameter und ihre Werte sind der folgenden Tabelle zu entnehmen:

PARAMETER	VALUE
STRATEGY	Separate and Conquer
OPTIMISATION CRITERION	Information Gain
PROBABILITY ESTIMATE	M Estimate, with $M=2$
MEASURE	Information Gain
INCREMENTAL REDUCED ERROR PRUNING	Most Accurate on Training Set
INFERENCE METHOD	Naive Bayes
EXPERIMENT TYPE	10-Fold Cross Validation



Rule Stretching

Experiment

10

Im Fall, dass zwei oder mehr Regeln ein Beispiel abdecken wird die wahrscheinlichste Klasse mithilfe des naive Bayes durch Maximierung des folgenden Ausdrucks ermittelt:

$$P'(C | R_1 \wedge \dots \wedge R_n) = P(C) P(R_1 | C) \dots P(R_n | C)$$

Dabei stellt C eine Klasse und R_1 bis R_n die Regeln, die das betrachtete Beispiel abdecken, dar.

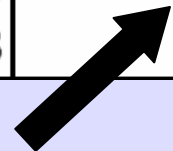


Rule Stretching

Experiment

Rule Stretching und Zuweisung der „majority class“ wurde an sieben Problemen getestet:

DOMAIN	CLASSES	CLASS DISTRIBUTION (%)	UNCOVERED EXAMPLES (%)	TOTAL EXAMPLES
BALANCE	3	7.84; 46.08; 46.08	19.04	625
CAR	4	3.76; 3.99 22.22; 70.02	3.36	1728
HOUSE VOTES	2	38.62; 61.38	2.53	435
KRKI	2	34.2; 65.8	2.6	1000
ALZHEIMERS TOX.	2	50; 50	6.54	886
SECONDARY PROTEIN STRUCTURE	2	43.4; 56.6	27.61	1014
STUDENT LOAN	2	35.7; 64.3	5.4	1000



Anteil der nicht durch die Basishypothese abgedeckten Beispiele



Rule Stretching

Experiment

12

Es ergeben sich folgende Resultate (Maß: accuracy):

DOMAIN	MAJORITY CLASS	RULE STRETCHING
BALANCE	76.32%	84.64%
CAR	93.98%	94.16%
HOUSE VOTES	94.25%	95.4%
KRKI	97.1%	99.4%
ALZHEIMERS TOX.	88.26%	90.18%
SECONDARY PROTEIN STRUCTURE	57.99%	61.14%
STUDENT LOAN	92.2%	92.7%



Rule Stretching

Experiment

13

Test:

Sei $accRS_i$ die durch Rule Stretching erreichte accuracy bei Problem i , sowie entsprechend $accMC_i$ die accuracy bei Zuweisung der majority class

Nullhypothese: $P(accRS_i \leq accMC_i) = P(accRS_i > accMC_i) = 0,5$

Bei allen sieben durchgeführten Experimenten zeigt sich $accRS_i > accMC_i$. Unter der Annahme, dass der „Erfolg“ beider Algorithmen a priori gleichwahrscheinlich ist, ergibt sich bei sieben Durchläufen eine Wahrscheinlichkeit von 0,0078, dass einer der Algorithmen in jedem der Durchläufe besser abschneidet als der andere. Dies erlaubt eine Ablehnung der Nullhypothese zum Niveau von 1%.



Rule Stretching

Fazit

14

Die Experimente haben gezeigt:

- Rule Stretching zeigt eine signifikant bessere Performance bei der Zuweisung einer Klasse zu einem durch die Basishypothese nicht klassifizierten Beispiel als die Zuweisung der „majority class“

Möglichkeiten zur Modifikation des Algorithmus:

- *use_best_rule* durch Naive Bayes ersetzen
- oder durch Klassifizierungskomponente von CN2



Hybride Algorithmen mit instanzbasierter Klassifikation

Motivation (lazy learner vs. eager learner)

15

Instanzbasiertes Lernen (lazy learner)

- Abstrahiert nicht von den Trainingsbeispielen sondern hält diese im Speicher vor
- Um eine neue Instanz zu klassifizieren werden im Speicher die „ähnlichsten“ Instanzen in der lokalen Nachbarschaft gesucht und der neuen Instanz die am häufigsten auftretende Klasse zugewiesen

Eager learner

- Hoher Aufwand für die Abstraktion der Trainingsinstanzen
 - Ermitteln einer geeigneten Repräsentation wie Entscheidungsbäume oder Regelmengen
- Klassifikationsphase mit recht geringem Aufwand



Hybride Algorithmen mit instanzbasierter Klassifikation

Motivation (lazy learner vs. eager learner)

16

Neuer Ansatz:

Kombination aus rechenaufwändigem *eager learning* in der Lernphase mit rechenintensiver instanzbasierter Klassifikation.

Die Standard-Klassifikationskomponente des *eager lerners* wird dabei durch eine instanzbasierte Klassifikation (hier: *k*-NN *classifier*) ersetzt.

Es werden hierzu drei hybride Algorithmen konstruiert:

- maximum-entropy modeling mit *k*-NN
- Regellerner mit *k*-NN (in zwei Versionen)



Hybride Algorithmen mit instanzbasierter Klassifikation

Instanzbasiertes Lernen

17

relevante Aspekte instanzbasierter Lerner :

- Abstandsberechnung:
 - MVDM (Modified Value Distance Metric)
 - feature weighting
- Klassifikation mithilfe k -NN



Abstandsberechnung

Berechnung mittels MVDM:

MVDM ermittelt anhand der Trainingsdaten den Abstand zweier Featurewerte v_1 und v_2 :

$$\delta(v_1, v_2) = \sum_{i=1}^j \left| P(C_i | v_1) - P(C_i | v_2) \right|$$

wobei j die Anzahl der Klassen darstellt.

Mit dieser Gleichung wird für jedes Feature das in den Inputdaten vorkommt eine Matrix mit den Distanzen der Werte des betrachteten Features erstellt.



Hybride Algorithmen mit instanzbasierter Klassifikation

Instanzbasiertes Lernen

Annahme: Es gibt 2 Features, wobei das erste Feature in den Trainingsdaten 4 Werte annimmt, das zweite 3.

Die durch MVDM berechneten Matrizen haben die folgende Form:

$$\begin{pmatrix} \delta(v_1, v_1) & \delta(v_1, v_2) & \delta(v_1, v_3) & \delta(v_1, v_4) \\ \delta(v_2, v_1) & \delta(v_2, v_2) & \delta(v_2, v_3) & \delta(v_2, v_4) \\ \delta(v_3, v_1) & \delta(v_3, v_2) & \delta(v_3, v_3) & \delta(v_3, v_4) \\ \delta(v_4, v_1) & \delta(v_4, v_2) & \delta(v_4, v_3) & \delta(v_4, v_4) \end{pmatrix} \text{ für Feature 1}$$

$$\begin{pmatrix} \delta(v_1, v_1) & \delta(v_1, v_2) & \delta(v_1, v_3) \\ \delta(v_2, v_1) & \delta(v_2, v_2) & \delta(v_2, v_3) \\ \delta(v_3, v_1) & \delta(v_3, v_2) & \delta(v_3, v_3) \end{pmatrix} \text{ für Feature 2}$$



Maximum-entropy Modeling (MAXENT)

Statistisches Modellieren: konstruieren eines stochastischen Modells (eine Verteilung), um das Verhalten eines zufälligen Prozesses vorherzusagen

Prinzip des Maximum-entropy Modelings:

- Modelliere alles was bekannt ist
 - Treffe keinerlei Annahmen über unbekannte Zusammenhänge
- Es ist ein Modell zu suchen, das alle bekannten Fakten enthält, ansonsten jedoch alle nicht weiter spezifizierten Ereignisse als gleichwahrscheinlich betrachtet



Hybride Algorithmen mit instanzbasierter Klassifikation

Maximum-entropy Modeling

21

Beispiel:

Ermitteln eines Modells zur Übersetzung des Englischen Wortes *in* ins Französische. $p(f)$ stellt hierbei die Wahrscheinlichkeit dar, dass ein Experte die Phrase f als Übersetzung für *in* wählen würde.

Wir finden heraus, dass ein Übersetzer immer eine der folgenden französischen Phrasen wählt: $\{dans, en, \grave{a}, au \text{ cours } de, pendant\}$.

Daraus ergibt sich die erste Bedingung für das Modell:

$$p(dans) + p(en) + p(\grave{a}) + p(au \text{ cours } de) + p(pendant) = 1$$

Es könnten nun verschiedenste Annahmen getroffen werden, bei denen obige Gleichung gilt. Beispielsweise $p(\grave{a}) = p(pendant) = 1/2$ oder $p(dans) = 1$. Dies würde jedoch das Prinzip des MAXENT verletzen, da wir lediglich wissen, dass der Experte eine der obigen Phrasen wählt.



Hybride Algorithmen mit instanzbasierter Klassifikation

Maximum-entropy Modeling

22

Beschränkt man sich auf die Fakten und nimmt sonst eine Gleichverteilung der Ereignisse an, so ergibt sich folgendes Modell:

$$p(\textit{dans}) = 1/5$$

$$p(\textit{en}) = 1/5$$

$$p(\grave{\textit{a}}) = 1/5$$

$$p(\textit{au _ cours _ de}) = 1/5$$

$$p(\textit{pendant}) = 1/5$$



Hybride Algorithmen mit instanzbasierter Klassifikation

Maximum-entropy Modeling

23

Würde man ferner feststellen, dass der Experte in 30% der Fälle entweder *dans* oder *en* als Übersetzung für *in* wählt, ergeben sich folgende Bedingungen:

$$p(dans) + p(en) = 3/10$$

$$p(dans) + p(en) + p(\grave{a}) + p(au_cours_de) + p(pendant) = 1$$

Und somit folgendes Modell:

$$p(dans) = 3/20$$

$$p(en) = 3/20$$

$$p(\grave{a}) = 7/30$$

$$p(au_cours_de) = 7/30$$

$$p(pendant) = 7/30$$



Hybride Algorithmen mit instanzbasierter Klassifikation

Maximum-entropy Modeling

24

Es können stets weitere Fakten in die Constraints aufgenommen werden. Beispielsweise wird beobachtet, dass der Experte in der Hälfte der Fälle *dans* oder *à* wählt:

$$p(dans) + p(en) = 3/10$$

$$p(dans) + p(en) + p(\grave{a}) + p(au_cours_de) + p(pendant) = 1$$

$$p(dans) + p(\grave{a}) = 1/2$$

Auch für diese Bedingungen kann wieder ein p ermittelt werden, welches sowohl die Constraints, als auch die Prinzipien des Maximum-entropy Modelings erfüllt. Je mehr Bedingungen aufgenommen werden, desto komplexer wird auch die Wahl von p .



k -NN und Maximum-entropy Modeling

Beobachtung: Die durch MAXENT erzeugte Matrix und die der MVDM des k -NN sind strukturgleich (Matrix bedingter Wahrscheinlichkeiten $P(C_i|v_j)$)

Prinzipieller Ablauf:

1. Training MAXENT
2. Ersetzen der Matrix der MVDM durch die von MAXENT erzeugte Matrix
3. Klassifikation mit k -NN



Hybride Algorithmen mit instanzbasierter Klassifikation

Hybride Algorithmen: MAXENT-H

26

Durch diese Konstruktion kann eine Schwachstelle der MVDM des k -NN behoben werden:

Empfindlichkeit hinsichtlich selten auftretender Daten:

- kommen v_1, v_2 mit $v_1 \neq v_2$ bei der gleichen Klasse vor
berechnet MVDM eine Distanz von 0
- kommen v_1, v_2 mit $v_1 \neq v_2$ bei verschiedenen Klassen vor
wird ihr Abstand hoch geschätzt

Die durch MAXENT durchgeführte Schätzung der Wahrscheinlichkeiten können MVDM eventuell „glätten“.



Hybride Algorithmen mit instanzbasierter Klassifikation

Hybride Algorithmen: k -NN und RULES

27

k -NN und Rule Learning

- Zwei verschiedene Versionen eines hybriden Algorithmus, der Regellernalgorithmen mit k -NN kombiniert
- Regelmenge ermittelt durch RIPPER

Ansatz für Realisierung: „Konvertierung“ der Regeln

Für jede Instanz, d.h. sowohl die Instanzen aus den Trainings-, als auch die aus den Testdaten, werden die Regeln in binäre Features „konvertiert“, die angeben, ob eine Regel feuert oder nicht.



Hybride Algorithmen mit instanzbasierter Klassifikation

Hybride Algorithmen: k -NN und RULES

28

Beispiel:

- R1: if (füße = 4) and (material = holz) and (lehne = true) then STUHL
- R2: if (füße = 4) and (material = holz) and (lehne = false) then TISCH
- R3: if (füße = 4) and (material = holz) then SCHRANK

Betrachtete Instanz: <füße=4,material=holz,lehne=true,farbe=rot>

Die Regeln R1 und R3 feuern $\rightarrow (1, 0, 1)$



RULES-R-H

- erste Version des hybriden Algorithmus (mittleres R steht für „replace“)
- ursprüngliche Features der Instanzen werden durch die binären Regel-Features ersetzt (vgl. Bsp.: $\langle R1 = 1, R2 = 0, R3 = 1 \rangle$)
- der original Feature space wird in einen neuen transformiert

Prinzipieller Ablauf:

1. Alle Trainings- und Testinstanzen in dieses binäre Format konvertieren und dem instanzbasierten Lerner übergeben
2. hybrider Algorithmus verwendet die k -NN Klassifikation um neue Beispiele zu klassifizieren



Hybride Algorithmen mit instanzbasierter Klassifikation

Hybride Algorithmen: RULES-R-H

30

Aus Sicht des k -NN kann die Ersetzung der original Features der Instanzen durch die „Regel-Features“ als ein Kompressions- und Filterungsschritt betrachtet werden:

- Entfernen von Noise und „irrelevanten Informationen“
- Gruppieren interagierender Feature-Werte

In den Hybriden werden Regeln als binäre (aktiv/inaktiv) Features repräsentiert, wobei mehr als eine Regel für eine Instanz aktiv sein kann. Verschiedene Regeln anstatt nur einer können an der Klassifikation beteiligt sein.



RULES-A-H

- zweite Version des hybriden Algorithmus (mittleres A steht für „adding“)
- die binären Regel-Features werden den ursprünglichen Features der Instanzen hinzugefügt (vgl. Bsp.: $\langle \text{füsse} = 4, \dots, R2 = 0, R3 = 1 \rangle$)
- k -NN Klassifizierer mit zusätzlichen Features

Prinzipieller Ablauf:

1. Alle Trainings- und Testinstanzen werden um die binären „feuert-Muster“ ergänzt und dem instanzbasierten Lerner übergeben
2. hybrider Algorithmus verwendet die k -NN Klassifikation um neue Beispiele zu klassifizieren



Hybride Algorithmen mit instanzbasierter Klassifikation

Hybride Algorithmen: RULES-A-H

32

- beim RULES-A-H Algorithmus können die Regel-Features nicht als Kompressions- und Filterungsschritt betrachtet werden
 - jedoch werden durch Hinzufügen der zusätzlichen Features die Abstandsberechnungen des k -NN modifiziert
- Gewichtung der zusätzlichen Features um deren Einfluss auf die Berechnungen zu vergrößern



Hybride Algorithmen mit instanzbasierter Klassifikation

Experiment

33

1. Die drei „parent-Algorithmen“ und die drei hybriden Algorithmen werden auf 29 Datensätze angewendet
2. Es werden 10-fold cross validation Experimente durchgeführt und die mittlere accuracy und Standardabweichung gemessen
3. Zur Überprüfung der Performanceunterschiede auf Signifikanz werden *paarweise t-tests* zwischen den Ergebnissen vorgenommen

k -NN und RULES bieten eine Reihe von Parametern. Um diese zu setzen wird eine Wrapper-basierende Methode zur automatischen Ermittlung eingesetzt.



Hybride Algorithmen mit instanzbasierter Klassifikation

Experiment

34

task	# inst.	<i>k</i> -NN	MAXENT	RULES	MAXENT-H	RULES-R-H	RULES-A-H
abalone	4177	24.6 ± 2.8	23.6 ± 1.7	18.1 ± 1.7	22.8 ± 2.2	18.0 ± 1.7	25.0 ± 2.5
audiology	226	80.5 ± 6.3	80.9 ± 5.0	76.5 ± 7.7	81.3 ± 5.8	61.0 ± 9.4	81.8 ± 5.2
bridges	104	54.7 ± 10.6	61.6 ± 9.1	53.8 ± 14.3	55.7 ± 13.1	52.9 ± 17.2	55.7 ± 13.1
car	1728	96.5 ± 1.3	90.9 ± 2.2	97.6 ± 1.1	96.5 ± 1.5	94.0 ± 4.0	98.4 ± 0.9
cl-h-disease	303	55.7 ± 5.4	55.1 ± 5.0	58.4 ± 5.9	54.8 ± 5.8	58.4 ± 5.9	58.4 ± 5.2
connect4	67557	77.7 ± 1.8	75.7 ± 0.5	76.3 ± 1.7	78.1 ± 1.9	75.2 ± 1.3	78.6 ± 2.5
ecoli	336	79.5 ± 4.9	76.5 ± 7.8	69.7 ± 10.9	78.6 ± 2.8	72.6 ± 12.1	78.0 ± 6.3
flag	194	66.9 ± 11.4	69.8 ± 13.4	61.8 ± 8.8	68.9 ± 14.9	61.8 ± 7.8	65.8 ± 10.9
glass	214	67.7 ± 8.4	70.1 ± 11.6	60.7 ± 6.5	61.5 ± 9.9	60.8 ± 8.0	66.9 ± 10.1
kr-vs-kp	3196	96.8 ± 1.2	96.8 ± 0.6	99.2 ± 0.5	99.1 ± 0.4	99.2 ± 0.5	99.2 ± 0.5
letter	20000	95.6 ± 0.5	85.0 ± 0.7	73.8 ± 1.5	95.9 ± 0.7	74.6 ± 1.4	95.6 ± 0.4
lung-cancer	32	33.3 ± 12.9	39.2 ± 24.7	31.7 ± 24.1	43.3 ± 13.3	25.0 ± 12.9	34.2 ± 16.0
monks1	432	100.0 ± 0.0	75.0 ± 4.0	99.3 ± 2.0	93.7 ± 10.5	100.0 ± 0.0	100.0 ± 0.0
monks2	432	94.0 ± 11.8	65.1 ± 5.5	72.0 ± 8.1	96.3 ± 8.3	75.5 ± 12.2	97.0 ± 4.2
monks3	432	97.2 ± 2.5	97.2 ± 2.5	97.2 ± 2.5	97.2 ± 2.5	96.5 ± 2.6	97.2 ± 2.5
mushroom	8124	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0	98.7 ± 2.6	100.0 ± 0.0
nursery	12960	99.4 ± 0.5	92.4 ± 0.4	97.7 ± 0.8	97.9 ± 1.1	97.9 ± 0.7	99.2 ± 0.4
optdigits	5620	98.0 ± 0.7	95.8 ± 0.8	89.3 ± 1.0	97.3 ± 0.6	89.9 ± 1.0	97.9 ± 0.6
pendigits	10992	93.4 ± 0.9	86.0 ± 1.2	82.6 ± 1.7	92.0 ± 1.4	81.7 ± 3.5	92.5 ± 1.5
promoters	106	87.0 ± 7.1	92.5 ± 9.0	79.4 ± 7.9	93.5 ± 8.2	80.3 ± 8.7	83.5 ± 11.2
segment	2310	95.7 ± 0.9	92.1 ± 2.8	90.5 ± 3.6	95.9 ± 1.1	90.6 ± 3.6	95.8 ± 1.3
solar-flare	1389	94.2 ± 2.2	94.7 ± 1.8	94.6 ± 1.5	94.2 ± 1.8	94.6 ± 1.5	94.2 ± 1.9
soybean-l	683	92.8 ± 4.2	92.2 ± 2.8	91.1 ± 3.0	93.1 ± 3.2	91.8 ± 3.2	92.8 ± 3.5
splice	3190	95.3 ± 1.0	94.6 ± 0.8	94.1 ± 1.6	94.8 ± 1.5	94.2 ± 1.1	95.8 ± 0.7
tictactoe	958	95.8 ± 3.8	98.3 ± 0.7	99.7 ± 0.7	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0
vehicle	846	67.6 ± 4.5	63.5 ± 5.2	55.7 ± 5.0	66.3 ± 5.4	56.0 ± 5.2	64.2 ± 5.7
votes	435	95.2 ± 2.4	96.5 ± 2.1	94.2 ± 1.6	95.4 ± 2.3	94.2 ± 1.6	94.9 ± 1.3
wine	178	96.1 ± 2.6	94.9 ± 5.3	93.2 ± 6.5	96.6 ± 2.7	92.7 ± 7.0	95.5 ± 4.9
yeast	1484	53.3 ± 2.9	49.3 ± 3.9	42.2 ± 3.2	55.4 ± 3.0	40.2 ± 2.1	52.0 ± 4.7

Mittlere accuracy und Standardabweichung der 10-fold CV Experimente



Hybride Algorithmen mit instanzbasierter Klassifikation

Analyse

35

Vergleich der Performance der „parent-Algorithmen“ anhand eines paarweisen t-tests (mittlere accuracy, Standardabweichung):

	k -NN	MAXENT	RULES
k -NN		10/17/2	14/12/3
MAXENT	2/17/10		10/13/6
RULES	3/12/14	6/13/10	

- k -NN weist hier meist eine bessere Performance auf als die anderen beiden Algorithmen
- MAXENT schneidet leicht besser ab als RULES



Hybride Algorithmen mit instanzbasierter Klassifikation

Analyse

36

Vergleich der drei hybriden Algorithmen mit ihren „parent-Algorithmen“ anhand eines paarweisen t-tests (mittlere accuracy, Standardabweichung):

	<i>k</i> -NN	MAXENT
MAXENT-H	4/21/4	12/17/0
	<i>k</i> -NN	RULES
RULES-R-H	2/14/13	3/23/3
RULES-A-H	4/24/1	16/13/0

- bei einer mit *k*-NN vergleichbaren „Leistung“ übertrifft MAXENT-H den MAXENT-Algorithmus auf 12 Datensätzen
- RULES-R-H weist eine deutlich schlechtere Performance als *k*-NN auf, jedoch eine ähnliche wie RULES
- RULES-A-H hat bei 4 Datensätzen eine signifikant höhere accuracy als *k*-NN und bei 16 verglichen mit RULES



Hybride Algorithmen mit instanzbasierter Klassifikation

Analyse

37

	k -NN	MAXENT
MAXENT-H	4/21/4	12/17/0
	k -NN	RULES
RULES-R-H	2/14/13	3/23/3
RULES-A-H	4/24/1	16/13/0

Weshalb schneidet RULES-R-H hier so viel schlechter ab als RULES-A-H?

Der Grund liegt evtl. in der automatischen Ermittlung der Parameter:

- in 70% der Experimente wurde für k ein Wert von 1 gewählt, was gleichbedeutend damit ist, dass der potentielle Vorteil der Verwendung der k -NN Klassifikation nicht zum tragen kommt
- bei RULES-A-H wurde $k = 1$ in lediglich 23% der Experimente gewählt



Hybride Algorithmen mit instanzbasierter Klassifikation

Analyse

38

Wie verändert sich bei der Bildung der hybriden Algorithmen die Zusammensetzung des Fehlers?

Algorithm	Bias	Algorithm	Bias
<i>k</i> -NN	57.29	MAXENT-H	56.92
MAXENT	65.56	RULES-R-H	51.63
RULES	55.50	RULES-A-H	54.99

- Die drei hybriden Algorithmen machen einen geringeren systematischen Fehler als ihre „parent-Algorithmen“



Die Experimente haben gezeigt:

→ MAXENT-H und RULES-A-H:

- Verbesserung der Klassifikations-Performance hinsichtlich der mittleren accuracy und der Standardabweichung

→ Bei allen hybriden Algorithmen:

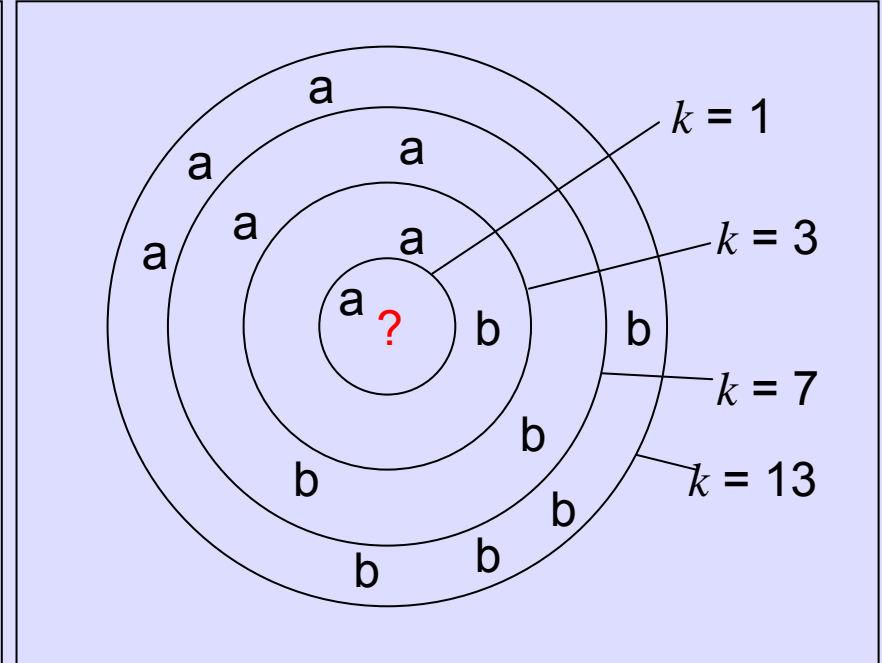
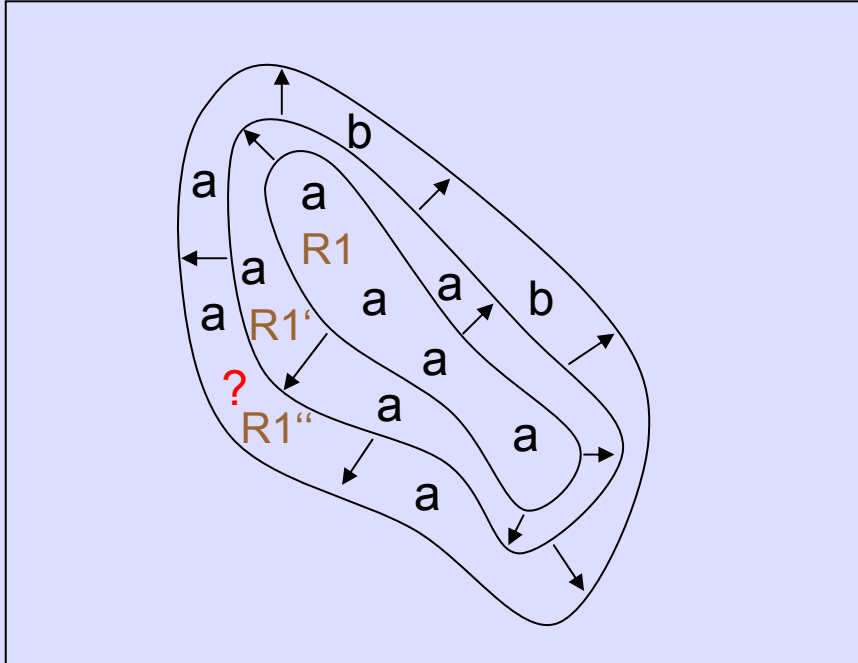
- Verringerung des systematischen Fehlers



Combining with Nearest Neighbor

Rule Stretching / Hybride Algorithmen mit instanzbasierter Klassifikation

40



Generalisierung:

- Vergrößerung des „Einflussbereichs“ der Regel
- zusätzliche Beispiele werden abgedeckt

Vergrößerung von k :

- zusätzliche Beispiele werden bei der Klassifikation berücksichtigt
- weiter entfernte Beispiele fließen ein (gemessen an der Distanz)



Vielen Dank für Ihre Aufmerksamkeit!