Discretizing Continuous Variables

Seminar Maschinelles Lernen

Darius Gasiorek

07. Dezember 2005

Inhalt

- Einleitung
  - Naive Bayes
  - Naive Bayes und Diskretisierung
- Diskretisierungsmethoden
  - unsupervised
  - supervised
- Evaluation

Einführung

- nominale vs. numerische Attributwerte
  - nominale Attribute haben eine festgelegte Wertemenge (z.B. Bundesland)
  - numerische Attribute haben keine explizit festgelegte Wertemenge (z.B. Größe, Einkommen)
- Naive Bayes klassifiziert in der Regel besser, wenn numerische Attribute diskretisiert werden
- im Folgenden
  - kurze Beschreibung von Naive Bayes
  - Motivation für Diskretisierung
  - Vorstellung verschiedener Methoden
  - Vergleich der Methoden im Hinblick auf Naive Bayes

Naive Bayes

- Jede Instanz besteht aus
  - Vektor von Attributwerten x $<x_1, x_2, \ldots, x_k>$
  - Klasse c

- Klassifizierung erfolgt durch
  - Berechnung der Wahrscheinlichkeit für jede Klasse anhand der relativen Häufigkeiten der Trainingsdaten für jeden Wert der Attribute
  - Vorhersagen der Klasse mit der höchsten Wahrscheinlichkeit

Naive Bayes

$$p(C = c \mid X = x) = \frac{p(C = c)\, p(X = x \mid C = c)}{p(X = x)}$$

$p(X = x)$ unabhängig von der Klasse

- Aufgrund der Unabhägigkeitsannahme gilt

$$p(X = x \mid C = c) = \prod p(X_i = x_i \mid C = c)$$

Naive Bayes und Diskretisierung

- Bei numerischen Attributen ist eine unendliche Anzahl von Werten möglich

  – nur wenige oder gar keine Trainingsinstanzen mit entsprechendem Wert vorhanden

  – $p(X_i = x_i \mid C = c)$ nimmt dann sehr kleine Werte an

  – falls keine Trainingsinstanzen mit entsprechenden Werten vorhanden, sind die Wahrscheinlichkeiten für alle Klassen gleich

  → Keine Entscheidung durch Naive Bayes möglich

→Diskretisieren

  –Zusammenfassen einzelner Werte zu einem Intervall

  –Behandlung der Intervalle als nominale Werte X_i^*

Naive Bayes und Diskretisierung

$$p(X_i = x_i \mid C = c) \approx p(a < X_i \leq b \mid C = c)$$

$$\approx p(X_i^* = x_i^* \mid C = c)$$

- Durch die Annäherung der Wahrscheinlichkeit entsteht jedoch ein Informationsverlust (Bias) der einzelnen Trainingsinstanzen

- Aber: bessere Naive Bayes Ergebnisse nach Diskretisierung

Diskretisierungsmethoden

Equal Width Discretization (EWD)

- EWD ist unsupervised
- Gegeben *n* Trainingsinstanzen
  - $v_{\min}$ kleinster Wert eines Attributs
  - $v_{\max}$ größter Wert eines Attributs
- Aufteilung der Werte in *k* Intervalle
  - Intervallbreite $w = (v_{\max} - v_{\min}) / k$
  - *k* ist ein vom User festzulegender Parameter

Equal Frequency Discretization (EFD)

- EFD ist ebenfalls unsupervised
- Gegeben *n* Trainingsinstanzen
- Aufteilung in *k* Intervalle
  - annähernd gleiche Anzahl an Trainingsinstanzen in jedem Intervall
  - also *n* / *k* Instanzen
  - *k* ist vom User festzulegen

EWD und EFD

- unsupervised Methoden

→relativ hoher Informationsverlust, da keine Berücksichtigung der Klasse

- Keine Berücksichtigung des Trade-Offs zwischen Bias und Varianz
- Nachteil ist die fehlende Robustheit
  - Ausreißer nach oben oder unten ziehen die Intervalle auseinander

- Dennoch
  - oft genutzt
  - relativ gute Ergebnisse mit Naive Bayes

Fuzzy Discretization (FD)

- Idee: ähnliche Werte sollten ähnlichen Einfluss auf die Wahrscheinlichkeiten der Intervalle haben

- Ausgangsbasis ist Diskretisierung in *k* Intervalle mittels EWD
- Klassenwahrscheinlichkeiten der Intervalle werden nicht nur aus den Instanzen im Intervall, sondern aus allen Trainingsinstanzen berechnet
- Annahme
  - „Einfluss“ einer Instanz auf die Wahrscheinlichkeit ist normal-verteilt
  - mit Erwartungswert *v*
  - Standardabweichung σ („fuzziness-Parameter“)

Fuzzy Discretization (FD)

- Einfluss einer Trainingsinstanz *j* auf Intervall *i*

$$P(v,\sigma,i)=\int_{a_i}^{b_i}\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{1}{2}\left(\frac{x-v}{\sigma}\right)^2}dx$$

- Klassenwahrscheinlichkeit des Intervalls

$$\frac{\sum_{j=1}^{n_c}P(v_j,\sigma,i)}{n}$$

Entropy Minimization Discretization (EMD)

- Entstanden in Anlehnung an Top Down Induction of Decision Trees
- Idee: Bilden von möglichst gleichgroßen Intervallen mit der jeweils geringsten Entropie (dem größten Informationsgewinn)
- Ablauf
  - Zusammenfassen aller Werte zu einem Intervall
  - Rekursive Teilung eines Intervalls
  - Kandidaten für Trennpunkte sind die Mittelwerte zwischen jeweils zwei Attributwerten
  - Wahl des Schnittpunkts mit der kleinsten gewichteten Summe der Entropien der beiden neu entstandenen Intervalle
  - Rekursive Weiterteilung bis Abbruchkriterium erreicht

Iterative Discretization (ID)

- Diskretisierung durch „Ausprobieren“

- Ausgangsbasis
  - *k* Intervalle
  - z.B. mittels EWD erstellt
- In jeder Iteration
  - Verschmelzen zweier Intervalle oder Aufspalten eines Intervalls
  - Evaluierung des Lerners mittels Leave-One-Out Cross-Validation
- Diskretisierung mit der geringsten Fehlerrate wird gewählt

→sehr ineffizient!

Proportional k-Interval Discretization (PKID)

- Versuch, den Trade-Off zwischen Varianz und Bias der Diskretisierung zu berücksichtigen

- Varianz: Stabilität der Wahrscheinlichkeit eines Intervalls
- Bias: Informationsverlust einer Instanz durch die Diskretisierung

- Allgemein
  - je größer das Intervall, desto größer der Bias und desto kleiner die Varianz
  - je kleiner das Intervall, desto kleiner der Bias und desto größer die Varianz

Proportional k-Interval Discretization (PKID)

- Sei
  - *n* die Anzahl aller Trainingsinstanzen
  - *s* die Anzahl der Instanzen in einem Intervall
  - *t* die Anzahl der Intervalle

→PKID berechnet *s* und *t* so, dass gilt

$$s * t = n$$

$$s = t \approx \sqrt{n}$$

Weighted PKID (WPKID)

- Erkenntnis bei PKID
  - bei kleinen Datensets sind die Intervalle zu klein, und die Varianz ist daher relativ hoch
  - hier bringt eine Verbesserung der Varianz mehr als eine Verbesserung des Bias
- WPKID setzt ein Minimum für die Intervallgröße

$$s * t = n$$

$$s - m = t$$

$$m = 30$$

Überlegung:

Bei sehr großen Datensets Beschränkung nach oben?

WPKID - Rechenbeispiel

- sei $n = 1800$
- PKID würde ca. 42 Intervalle mit jeweils ca. 42 Instanzen erstellen

- WPKID berechnet

  $s * (s - 30) = 1800$

  $s = 60$

  $t = 30$

  → also 30 Intervalle mit jeweils 60 Instanzen

- Die Erhöhung der Intervallgröße von 42 auf 60 Instanzen reduziert die Varianz

Lazy Discretization (LD)

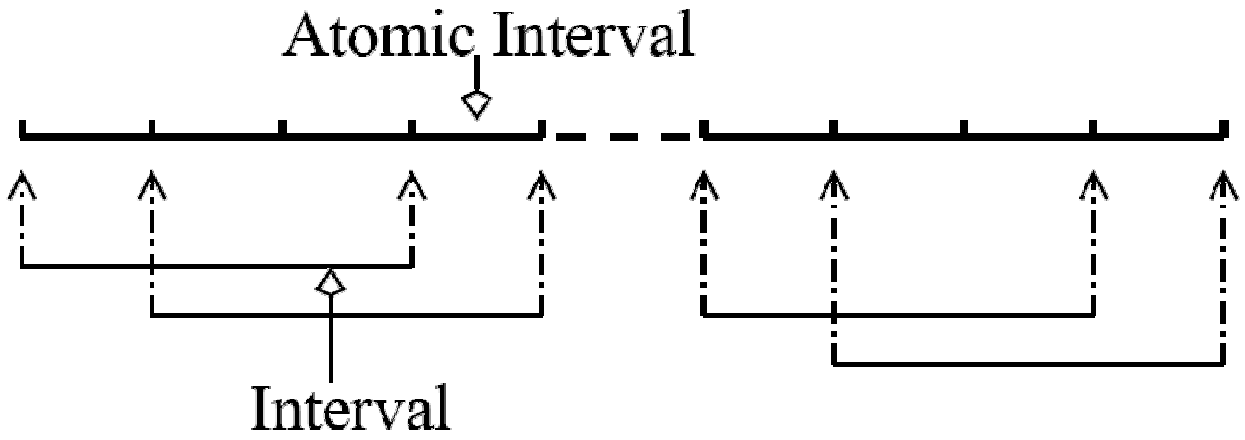
- Diskretisierung wird für jede zu klassifizierende Testinstanz neu durchgeführt
  - keine Berechnung von Wahrscheinlichkeiten vor der Klassifizierung
  - daher „lazy“
- LD bildet nur ein Intervall, um den neuen Attributwert *v*
  - *v* bildet dabei den mittleren Wert des Intervalls
  - Größe des Intervalls wird mittels EFD mit *k*=10 ermittelt
- Problem
  - EFD mit *k*=10 nicht unbedingt die beste Wahl
  - lange Laufzeit

Non-Disjoint Discretization (NDD)

- Idee: Vorteile von LD und PKID kombinieren
  - Informationsverlust geringer, wenn Wert in der Mitte liegt
  - Trade-Off zwischen Bias und Varianz berücksichtigen
  - Wahrscheinlichkeiten vor der Klassifizierung berechnen

- Ablauf
  - Bilden von Intervallen nach PKID
  - Unterteilen der Intervalle in jeweils drei gleich große Intervalle
  - Zusammenfassen von jeweils drei kleinen Intervallen zu einem neuen Intervall

Non-Disjoint Discretization (NDD)



- Berechnen der Wahrscheinlichkeiten für alle sich überlappenden Intervalle
- Für jede Testinstanz wird jenes große Intervall gewählt, in dessen mittlerem kleinen Intervall der Wert *v* liegt
  - Ausnahme bilden Werte im ersten und letzten kleinen Intervall

Weighted Non-Disjoint Discretization (WNDD)

- Weiterentwicklung von NDD
- Basis bilden Intervalle nach WPKID nicht nach PKID
- Signifikant bessere Ergebnisse als die meisten anderen Methoden

Evaluierung

Evaluierung

- 35 Datensets
  - Unterschiedlich viele Trainingsinstanzen
  - Unterschiedliches Verhältnis von numerischen und nominalen Attributen
  - Unterschiedlich viele Klassen
- 3-fold Cross-Validation
- Leider kein Test ohne Diskretisierung

Evaluation

Datensatz	Größe	num.	nom.	Klassen	Fehlerrate (%)			
					EWD	EFD	FD	PKID
Pittsburgh	106	3	8	3	12,9	12,1	10,5	13,0
Sonar	208	60	0	2	26,9	25,2	26,8	25,7
Vehicle	846	18	0	4	38,7	40,5	42,4	38,2
Annealing	898	6	32	6	3,5	2,3	3,9	2,2
Forest-Covertype	581012	10	44	7	32,4	32,9	32,2	31,7
Mittlerer Fehler					20,1	19,9	20,9	19,1

Evaluation

Datensatz	Größe	num.	nom.	Klassen	Fehlerrate (%)			
					LD	NDD	WPKID	WNDD
Pittsburgh	106	3	8	3	12,3	13,1	11,9	10,8
Sonar	208	60	0	2	27,3	26,9	23,7	22,8
Vehicle	846	18	0	4	38,7	38,5	38,2	38,8
Annealing	898	6	32	6	1,6	1,8	2,2	2,3
Forest-Covertype	581012	10	44	7		31,4	31,7	31,4
Mittlerer Fehler					18,6	19,1	18,7	18,2

Evaluation - WNDD

	EWD	EFD	FD	EMD	PKID	WPKID	LD	NDD
Win	8	3	9	4	4	8	11	11
Lose	26	31	25	28	25	22	19	18
Tie	1	1	1	3	6	5	4	6
Sign Test	< 0.01	< 0.01	< 0.01	< 0.01	< 0.01	< 0.01	0.10	0.13

Evaluation

Datensatz	Größe	num.	nom.	Klassen	Fehlerrate (%)			
					EWD	EFD	FD	PKID
Pittsburgh	106	3	8	3	12,9	12,1	10,5	13,0
Sonar	208	60	0	2	26,9	25,2	26,8	25,7
Vehicle	846	18	0	4	38,7	40,5	42,4	38,2
Annealing	898	6	32	6	3,5	2,3	3,9	2,2
Forest-Covertype	581012	10	44	7	32,4	32,9	32,2	31,7
Mittlerer Fehler					20,1	19,9	20,9	19,1

Evaluation

Datensatz	Größe	num.	nom.	Klassen	Fehlerrate (%)			
					LD	NDD	WPKID	WNDD
Pittsburgh	106	3	8	3	12,3	13,1	11,9	10,8
Sonar	208	60	0	2	27,3	26,9	23,7	22,8
Vehicle	846	18	0	4	38,7	38,5	38,2	38,8
Annealing	898	6	32	6	1,6	1,8	2,2	2,3
Forest-Covertype	581012	10	44	7		31,4	31,7	31,4
Mittlerer Fehler					18,6	19,1	18,7	18,2

Evaluation

Datensatz	Größe	num.	nom.	Klassen	Fehlerrate (%)			
					EWD	EFD	FD	PKID
Pittsburgh	106	3	8	3	12,9	12,1	10,5	13,0
Sonar	208	60	0	2	26,9	25,2	26,8	25,7
Vehicle	846	18	0	4	38,7	40,5	42,4	38,2
Annealing	898	6	32	6	3,5	2,3	3,9	2,2
Forest-Covertype	581012	10	44	7	32,4	32,9	32,2	31,7
Mittlerer Fehler					20,1	19,9	20,9	19,1

Evaluation

Datensatz	Größe	num.	nom.	Klassen	Fehlerrate (%)			
					LD	NDD	WPKID	WNDD
Pittsburgh	106	3	8	3	12,3	13,1	11,9	10,8
Sonar	208	60	0	2	27,3	26,9	23,7	22,8
Vehicle	846	18	0	4	38,7	38,5	38,2	38,8
Annealing	898	6	32	6	1,6	1,8	2,2	2,3
Forest-Covertype	581012	10	44	7		31,4	31,7	31,4
Mittlerer Fehler					18,6	19,1	18,7	18,2

Evaluation

Datensatz	Größe	num.	nom.	Klassen	Fehlerrate (%)			
					LD	NDD	WPKID	WNDD
Pittsburgh	106	3	8	3	12,3	13,1	11,9	10,8
Sonar	208	60	0	2	27,3	26,9	23,7	22,8
Vehicle	846	18	0	4	38,7	38,5	38,2	38,8
Annealing	898	6	32	6	1,6	1,8	2,2	2,3
Forest-Covertype	581012	10	44	7		31,4	31,7	31,4
Mittlerer Fehler					18,6	19,1	18,7	18,2

Evaluation

Datensatz	Größe	num.	nom.	Klassen	Fehlerrate (%)			
					LD	NDD	WPKID	WNDD
Pittsburgh	106	3	8	3	12,3	13,1	11,9	10,8
Sonar	208	60	0	2	27,3	26,9	23,7	22,8
Vehicle	846	18	0	4	38,7	38,5	38,2	38,8
Annealing	898	6	32	6	1,6	1,8	2,2	2,3
Forest-Covertype	581012	10	44	7		31,4	31,7	31,4
Mittlerer Fehler					18,6	19,1	18,7	18,2

Fazit

- Diskretisierung ist eine gute Möglichkeit, um mit numerischen Attributen umzugehen
- Alternativen zur Diskretisierung siehe nächster Vortrag

- Kombination der einzelnen Ideen in neuen Verfahren möglich und sinnvoll
  - WNDD als Kombination sehr erfolgreich
  - Weitere Kombinations- / Verbesserungsmöglichkeiten (LD mit WPKID, FD mit WPKID etc.)

Vielen Dank für die Aufmerksamkeit!!!