Handout zum Vortrag: Naive Bayes für Regressionsprobleme

Referent: Nils Knappmeier

Datum: 14.12.2005

1 Einleitung

1.1 Traditioneller Naive Bayes

Der Naive Bayes Algorithmus

- $p(K|A_1, A_2, \ldots, A_n) = \frac{p(A_1,A_2,\ldots,A_n|K)\cdot p(K)}{p(A_1,A_2,\ldots,A_n)}$
- Annahme: A_k sind unabhängig voneinander.
- Daher: Wähle k mit maximalem

$$\hat{P}(K|A_1, A_2, \ldots, A_n) = p(K) \cdot \prod_{i=1}^{n} p(A_i|k)$$

- $p(K)$ und $p(A_i|K)$ kann anhand der Trainingsdaten berechnet werden.

1.2 Naive Bayes und Regression

- Angepasste Bayes-Formel

$$p(Z|A) = \frac{p(A|Z) \cdot p(Z)}{\int p(A|Z) \cdot p(Z) dZ}$$

Bayes Formel

- Nach Anwendung der Unabhängigkeitsannahme:

$$p(Z|A) = \frac{p(Z) \cdot \prod_{i=1}^{n} p(A_i|Z)}{\int p(Z) \cdot \prod_{i=1}^{n} p(A_i|Z) dZ}$$

- Zu ermitteln: Annäherungen an $p(A_i|Z)$ und $p(Z)$

Zunächst wird eine Annäherung an die Bayes-Formel berechnet, indem die einzelnen Komponenten $p(A_i|Z)$ und $p(Z)$ berechnet werden. Diese Annäherung wird dann für jedes z in einem gegebenen Zielintervall (z.B. $[10, 20]$ mit Schrittweite 0.1) berechnet. Der Median oder der Erwartungswert der Annäherung ist dann der vorhergesagte Zielwert z.

2 Annäherung durch Interpolation mit Gauss-Kurven

Um eine kontinuierliche Dichtefunktion zu erhalten, werden die Beispielwerte mit Gauss-Glocken interpoliert.

$$\frac{1}{h}K\left(\frac{b-b_i}{h}\right) \quad \text{mit} \quad K(x) = (2\pi)^{-\frac{1}{2}} \cdot e^{-\frac{x^2}{2}}$$

Die b_i sind dabei jeweils die Zentren der Gauss-Glocken. Die Standardabweichung h wird per Leave-One-Out-Cross-Validation bestimmt.

Leave-One-Out-Cross-Validation bedeutet die Maximierung einer Pseudwahrscheinlichkeitsfunktion

$$f_i^*(b_i) = \frac{1}{(n-1)h}\sum_{j=1;i\neq j}^{n} K\left(\frac{b_i - b_j}{h}\right)$$

über alle Beispielwerte, also:

$$h_{CV} = \arg\max_h \left\{\frac{1}{n}\sum_{i=1}^{n}\log f^*(b_i)\right\}$$

Die Maximierung wird durch Ausprobieren aller Werte für h in einem diskretisierten Intervall durchgeführt.

3 Algorithmus: Naive Bayes für Regression

3.1 Ermittlung der Teilfunktionen $p(A_i|Z)$ und $p(Z)$

3.1.1 Berechnung von $p(A_i|Z)$ für numerische Attribute

$$p(A_i|Z) =: p(B|Z) = \frac{p(B,Z)}{p(Z)}$$

Berechnung des Zähler (z.B. über das Interval $k \in [10, 20]$ mit Schrittweite 0.1):

$$\hat{p}(B = b.Z = z) = \frac{1}{nh_bh_z}\sum_{i=1}^{n} K\left(\frac{b-b_i}{h_b}\right)K\left(\frac{z-z_i}{h_z}\right)$$

Dabei ist b_i Attributwert der Trainingsinstanz i.

$$\hat{p}(Z = z) = \frac{1}{nh_z}\sum_{i=1}^{n} K\left(\frac{k-k_i}{h_z}\right)$$

Die Berechnung des h_z und h_b funktioniert analog zu Kapitel 2

3.1.2 Berechnung von $p(A_i|Z)$ für nominale Attribute

$$p(A_i|Z) =: p(B = b|Z = z) = \frac{p(B=b)\,p(Z=z|B=b)}{\sum_{b\in\mathbf{Kat_B}} p(B=b)\,p(Z=z|B=b)}$$

Die einzelnen Kompopnenten werden jeweils berechnet durch

$$\hat{p}(Z = z|B = b) = \frac{1}{n_k}\sum_{i\in T, b=b_i} K\left(\frac{z-z_i}{h_z}\right)$$

und

$$\hat{p}(B = b) = \frac{\text{Anzahl Beispiel mit } A_i = b}{\text{Anzahl Beispiele insgesamt}}$$

Dies wird wieder für alle z des Zielintervalls (z.B. $[10, 20]$ Schrittweite 0.1) getan.

3.1.3 Berechnung von $p(z)$

$p(z)$ wird als Interpolation aller z_i (Zielwerte der Trainingsdaten) berechnet, für das Interval (z.B. $[10, 20]$ mit Schrittweite 0.1)

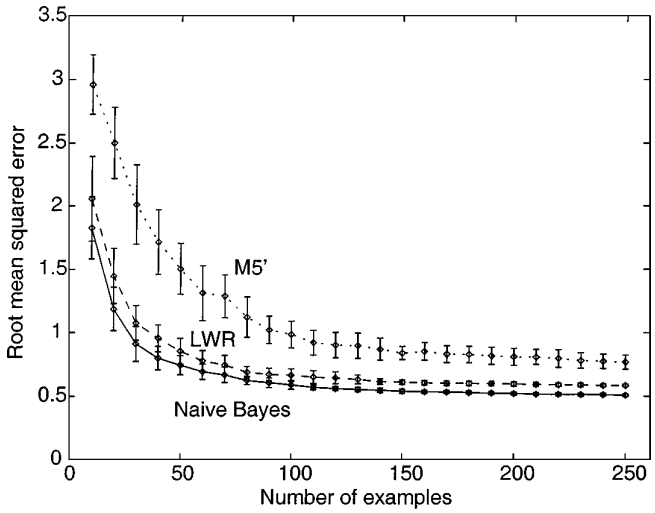
3.2 Berechnung des Zielwerts

Man kann beweisen, dass der quadratische Fehler des vorhergesagten Wertes minimiert wird, denn der **Durchschnitt** der Zielfunktion berechnet wird und dass der absolute Fehler minimiert wird, denn der **Median** berechnet wird. Die Berechnung kann einfach numerisch durchgeführt werden, da jetzt die Funktionswerte von $p(Z = z|A)$ für alle im Zielintervall vorliegen.

4 Evaluation

4.1 Probleme mit unabhängigen Attributen

Als Beispiel für ein Problem im unabhängigen Attributen wurde eine 3D-Spirale mathematisch erzeugt durch $x_1 = y \cdot \sin(y) + N(0, 1)$ und $x_2 = y \cdot \cos(y) + N(0, 1)$. Dann wurde auf der Basis von 1000 Beispielen mit verschienen Algorithmen (LWR, M5' und Naive Bayes) die Werte von y in Abhängigkeit von $x1$ und $x2$ berechnet. Im Bild kann man erkennen, dass Naive Bayes hier geringe Fehler macht, als die anderen Algorithmen.



4.2 Standard-Datensätze

Die folgenden Tabellen zeigen, dass der Naive Bayes Algorithmus gegen LR, LWR und M5' im Direktvergleich eher schlechter abschneidet. Gezählt wurden die Zahl der Datensätze, bei denen der Spalten-Algorithmus eine signifikant geringerer Fehlerrate aufweißt, als der Zeilen-Algorithmus.

	Gewinner			
Verlierer ⇓	Naive Bayes	LR	LWR	M5’
Naive Bayes		18	20	23
LR	8		13	15
LWR	6	10		15
M5’	3	4	6	

Durchschnittlicher Quadratischer Fehler

	Gewinner			
Verlierer ⇓	Naive Bayes	LR	LWR	M5’
Naive Bayes		13	19	22
LR	13		17	16
LWR	6	9		19
M5’	5	5	8	

Durschnittlicher Absoluter Fehler

5 Fazit

Für Regressionsprobleme gilt: Unabhängigkeit der Attribute

- erfüllt: Naive Bayes funktioniert gut
- nicht-erfüllt: Andere Algorithmnen schneiden besser ab

Auch wenn der Naive Bayes für Klassifikationsprobleme immer gut funktioniert.

6 Quellen

- Technical Notes: Naive Bayes for Regression; E. Frank, L.Trigg, G.Holmes, I.H.Witten; Machine Learning 41, 5-25, 2000
- Retrofitting Decision Tree Classifiers Using Kernel Density Estimation; P.Smyth, A.Gray, U.M.Fayyad (Appendix: Univariate Bandwidth Selection for Kernel Density Estimation)
- Naive Bayes zur Klassifikation: http://www.ke.informatik.tu-darmstadt.de/lehre/ws05/mldm/bayes.pdf