Dr. Andreas Hotho, Robert Jäschke
Fachgebiet Wissensverarbeitung

16.1.2009

# 9. Übung „Knowledge Discovery“

Wintersemester 2008/2009

## 1 Naiver Bayes-Klassifikator

| Day | Outlook | Temperature | Humidity | Wind | PlayTennis |
|---|---|---|---|---|---|
| D1 | Sunny | Hot | High | Weak | No |
| D2 | Sunny | Hot | High | Strong | No |
| D3 | Overcast | Hot | High | Weak | Yes |
| D4 | Rain | Mild | High | Weak | Yes |
| D5 | Rain | Cool | Normal | Weak | Yes |
| D6 | Rain | Cool | Normal | Strong | No |
| D7 | Overcast | Cool | Normal | Strong | Yes |
| D8 | Sunny | Mild | High | Weak | No |
| D9 | Sunny | Cool | Normal | Weak | Yes |
| D10 | Rain | Mild | Normal | Weak | Yes |
| D11 | Sunny | Mild | Normal | Strong | Yes |
| D12 | Overcast | Mild | High | Strong | Yes |
| D13 | Overcast | Hot | Normal | Weak | Yes |
| D14 | Rain | Mild | High | Strong | No |

1. Klassifizieren Sie mit Hilfe des naiven Bayes-Klassifikators den Datensatz *D15 = (O=Overcast, T=Cool, H=High, W=Strong)*.

$$c_{NB} = \underset{c \in \{yes,no\}}{\operatorname{argmax}} P(c) \prod_{i=1}^{4} P(o_i|c)$$

$$c_{NB} = \underset{c \in \{yes,no\}}{\operatorname{argmax}} P(c)P(O = Over|c)P(T = Cool|c)P(H = High|c)P(W = Stro|c)$$

für $c = yes$ ergibt sich:

$$\frac{9}{14} \cdot \frac{4}{9} \cdot \frac{3}{9} \cdot \frac{3}{9} \cdot \frac{3}{9} = \frac{6}{567} \approx 0.0106$$

für $c = no$ ergibt sich:

$$\frac{5}{14} \cdot \frac{0}{5} \cdot \frac{1}{5} \cdot \frac{4}{5} \cdot \frac{3}{5} = \frac{0}{875} = 0.0$$

und damit wird *D15* mit $c_{NB} = yes$ klassifiziert.

2. Klassifizieren Sie mit Hilfe des naiven Bayes-Klassifikators den Datensatz
*D16 = (T=Cool, H=High, W=Strong)* und vergleichen Sie ihr Ergebnis mit dem aus Teilaufgabe a).

$$c_{NB} = \underset{c \in \{yes,no\}}{\operatorname{argmax}} P(c)P(T = Cool|c)P(H = High|c)P(W = Stro|c)$$

für $c = yes$ ergibt sich:

$$\frac{9}{14} \cdot \frac{3}{9} \cdot \frac{3}{9} \cdot \frac{3}{9} = \frac{1}{42} \approx 0.0238$$

für $c = no$ ergibt sich:

$$\frac{5}{14} \cdot \frac{1}{5} \cdot \frac{4}{5} \cdot \frac{3}{5} = \frac{6}{175} \approx 0.0343$$

und damit wird *D16* mit $c_{NB} = no$ klassifiziert.
Durch Ignorieren des Attributs *Outlook* ändert sich die Klassifikation von *yes* auf *no*. Die Wahrscheinlickeit $P(O = Overcast|no) = 0$ dominiert in Teilaufgabe 1 den gesamten Term und dadurch haben die übrigen Attribute keinen Einfluß auf das Ergebnis der Klassifikation.

3. Diskutieren Sie die praktischen Auswirkungen z.B. auf Spamfilter.

Ist die Menge der Trainingsbeispiele gering, so gibt es möglicherweise mehrere Wahrscheinlichkeiten $P(o_i|c) = 0$, welche den gesamten Bayesschen Klassifikator beeinflussen. Dies gilt aber generell: ist für ein Attribut kein Wert vorhanden, der eine bestimmte Klassifikation stützt, so ist die Wahrscheinlichkeit dieser Klassifikation bei diesem Attributwert gleich Null - unabhängig von den anderen Attributen.
Ein Spamfilter in dessen gelernter Spamliste sich beispielsweise nur Mails mit der Absenderdomain xyz.de befinden, würde mittels des naiven Bayesschen Klassifikators keine Mail, die nicht aus der Domain xyz.de stammt, als Spam identifizieren. Und zwar unabhängig davon, ob andere Kriterien (Betreff, Inhalt, ...) für die Hypothese *Spam* sprechen, oder nicht.

4. Was bewirkt eine Änderung der Berechnung der geschätzten Wahrscheinlichkeit von $P(a|c) = \frac{n_c}{n}$ zu $P(a|c) = \frac{n_c+mp}{n+m}$ (wobei $n$ die Gesamtzahl der Trainingsbeispiele mit Klassifikation $c$ und $n_c$ die Anzahl der Trainingsbeispiele mit Klassifikation $c$ und dem Attributwert $a$ darstellt; $m$ ist eine Konstante und $p$ der geschätzte Wert für $P(a|c)$ – ist dieser unbekannt, wird Gleichverteilung der Attributwerte angenommen)? Vergleichen Sie diesen Ansatz unter dem Gesichtspunkt der in 3. diskutierten Probleme.

Es werden $m$ *virtuelle* (gleichverteilte) Trainingsbeispiele hinzugefügt. Diese verhindern Null-Werte und damit die Dominanz eines einzigen Attributes.
Damit läßt sich verhindern, daß beispielsweise alleine die Absender-Domain einer Mail die Klassifikation als *Spam* oder *nicht Spam* entscheidet.

## 2 k-nächste Nachbarn Klassifikator

1. Wie könnte ein Klassifikationsverfahren aussehen, welches nur die $k$ nächsten Nachbarn einer zu klassifizierenden Instanz in Betracht zieht? Geben Sie die prinzipiellen Schritte eines solchen Verfahrens.

   Prinzipielle Schritte:

   - Bestimmung der $k$-nächsten Nachbarn aus der Menge der Trainingsbeispiele zum Anfragewert
   - Auswahl des häufigsten (oder distanz-gewichtet, oder nach Verteilung der Klasse gewichtet) Funktions-/Klassifikationswertes als Funktions-/Klassifikationswert für den Anfragewert

2. Nennen Sie jeweils mindestens zwei Abstandsmaße für numerische und kategorische Werte.

   - numerisch:
     - euklidische Metrik: $d(x_i, x_j) = \sqrt{\sum_{r=1}^{n} \left(a_r(x_i) - a_r(x_j)\right)^2}$
     - Manhattan-Metrik: $d(x_i, x_j) = \sum_{r=1}^{n} |a_r(x_i) - a_r(x_j)|$
   - kategorisch:
     - ungewichtet: $d(x_i, x_j) = \sum_{r=1}^{n} \delta\left(a_r(x_i), a_r(x_j)\right)$ mit $\delta\left(a_r(x_i), a_r(x_j)\right) = 0$ gdw. $a_r(x_i) = a_r(x_j)$
     - gewichtet: $d(x_i, x_j) = \sum_{r=1}^{n} g_r \delta\left(a_r(x_i), a_r(x_j)\right)$

3. Berechnen Sie für $k = 4$ den Abstand zum Beispiel (sunny, cool, high, strong) für den Datensatz der Aufgabe 1.

   Vier nächste Nachbarn:

| Day | Outlook | Temperature | Humidity | Wind | PlayTennis | Distance |
|---|---|---|---|---|---|---|
| D2 | sunny | hot | high | strong | No | 1 |
| D1 | sunny | hot | high | weak | No | 2 |
| D6 | rain | cool | normal | strong | No | 2 |
| D7 | overcast | cool | normal | strong | Yes | 2 |

   $3\times$ „No“, $1\times$ „Yes“ $\Longrightarrow$ Klassifikation „No“