Bayessche Netzwerke

von Steffen Otto

Proseminar: Machine Learning

Inhaltsverzeichnis

Proseminar: Machine Learning · SS 2006 27. Juni 2006

Bayessche Netzwerke

Steffen Otto

Zusammenfassung: *Der folgende Artikel beschreibt, wie mit Hilfe von Wahrscheinlichkeiten ein Bayessches Netzwerk aufgebaut wird und damit effizient Wahrscheinlichkeits-Vorhersagen getroffen werden können. Außerdem wird das Bayessche Lernen mittels Parametrisierung näher spezifiziert.*

1 Einleitung

Bayessche Netze kommen in modernen Expertensystemen zur Wissensrepräsentation zum Einsatz. Das Wissen wird über Zufallsvariablen und deren Beziehungen untereinander repräsentiert. Zur effektiven Erfassung des Wissens ist es nötig, mit möglichst wenigen Angaben eine hinreichende Beschreibung der charakteristischen Merkmale eines Systems zu erhalten. Bayessche Netzwerke sind weniger auf das Optimum an Genauigkeit, als viel mehr auf Effizienz für Speicher und Rechenzeit ausgelegt. Im Folgenden geht es nun darum, wie ein solches Netzwerk aufbaut wird und damit Vorhersagen getroffen werden können.

2 Bayessche Netzwerke

Ein Bayessches Netz ist ein gerichteter, azyklischer Graph, der effizient gemeinsame Wahrscheinlichkeits-Verteilungen und Aussagen zur bedingten Unabhängigkeit von Zufallsvariablen beschreibt. Er bildet den Zusammenhang von verschiedenen Eigenschaften nicht 1:1 ab, sondern stellt eine vereinfachte Form dar, mit der **Wahrscheinlichkeiten** (im Folgenden mit **W.** abgekürzt) effizient berechnet werden können.

2.1 Allgemeine Struktur

Im Bayesgraph entspricht jeder Knoten einer diskreten Zufallsvariablen mit Wahrscheinlichkeits-Tabelle. Die Wurzelknoten sind a priori W. und über Kanten mit den übrigen Knoten verbunden.

Die Kanten beschreiben die direkten Abhängigkeiten zwischen den Variablen. Außer den Wurzelknoten hat jeder Knoten X mindestens einen Elternknoten: par(X). Die W. eines Knotens ist immer vom Eintreten der Eigenschaft des Elternknotens abhängig und wird mit ***P(X|par(X))*** beschrieben.

Ein Bayesgraph darf keine zyklischen Strukturen enthalten. D.h. ein Knoten darf über keinen gerichteten Pfad mit sich selbst verbunden sein. Sonst hinge seine W. ja von sich selbst ab. Um auch zyklische Strukturen beschreiben zu können, wird eine Momentaufnahme eines Systems gemacht und dort, wo Zyklen auftreten, eine Kante entfernt, so dass

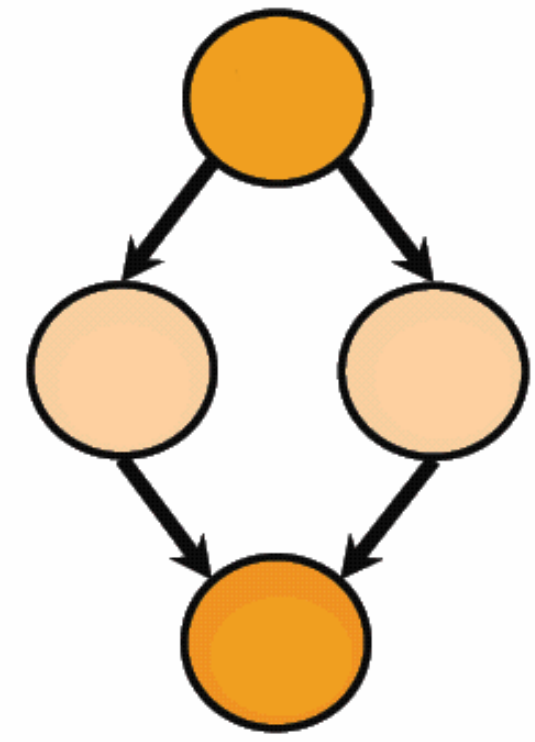

Abbildung 1: Allgemeine Struktur eines Bayesgraphen

wieder ein azyklischer Bayesgraph entsteht. Die fehlende Kante wird nun reihum in unterschiedlichen Bayesgraphen beschrieben und getrennt behandelt.

2.2 Datenerhebung zum Aufstellen eines Bayesgraphen

Zunächst wird eine genügend große Anzahl von Daten benötigt. Diese sind aus umfangreichen statistischen Ermittlungen zu erhalten. In der Praxis werden Daten häufig geschätzt, da deren Erhebung oft mit einem hohen finanziellen Aufwand verbunden ist.

Um später eine Vorhersage machen zu können, müssen die Daten alle relevanten Einflussgrößen mit den entsprechenden Ausprägungen enthalten. Des Weiteren muss die Abhängigkeit zwischen den Einflussgrößen bekannt sein.

Für Bayessche Netze ist es wichtig, möglichst wenig Kanten zu haben. Bei einem durchschnittlichen Bayesgraphen mit n Knoten beträgt die Laufzeit zur Berechnung einer W. $O(n*2^n)$. Mit dem Speicherbedarf verhält es sich ähnlich. Kommt eine Kante dazu, so erhöht sich die Rechenzeit unter Umständen um das Doppelte. Nun liegt es nahe, warum ein Bayesgraph in der Praxis mit möglichst wenig Kanten und Knoten auskommen sollte. Für die Erzeugung werden also nur die wichtigen Abhängigkeiten zwischen den Merkmalen betrachtet.

2.3 Erzeugung eines Bayesgraphen

Die gesammelten Daten werden nun mit der vorhandenen Erfahrung zu einem möglichst einfachen Bayesgraphen zusammengebaut. Unabhängige Zufallsvariablen ergeben hier die Wurzelknoten. Beim Aufbau ist auf Zyklenfreiheit zu achten. Zum Schluss werden noch unnötige Pfeile (gerichtete Kanten) entfernt und die Knoten indiziert. Dazu werden zunächst alle Wurzelknoten, dann alle ihre Kinder, dann alle Kindeskinder, usw. durchnummeriert.

2.4 Berechnung der Wahrscheinlichkeit mit Hilfe eines Bayesgraphen

Die W., dass ein Merkmal eintritt, wird jeweils durch die Elternknoten bestimmt. Daher ist die Wahrscheinlichkeit für das Auftreten einer Zufallsvariablen B und deren Elternknoten A wie folgt zu bestimmen:

$$par(B) = Elternknoten\ von\ B = A$$

$$P(B|\ par(B)) = P(B|A)$$

Wenn $P(B|A)$ nicht bekannt ist, so kann mit Hilfe des Bayestheorems aus $P(A|B)$, $P(A)$ und $P(B)$ die gewünschte W. berechnet werden:

$$P(B \mid A) = \frac{P(A \mid B) * P(B)}{P(A)}$$

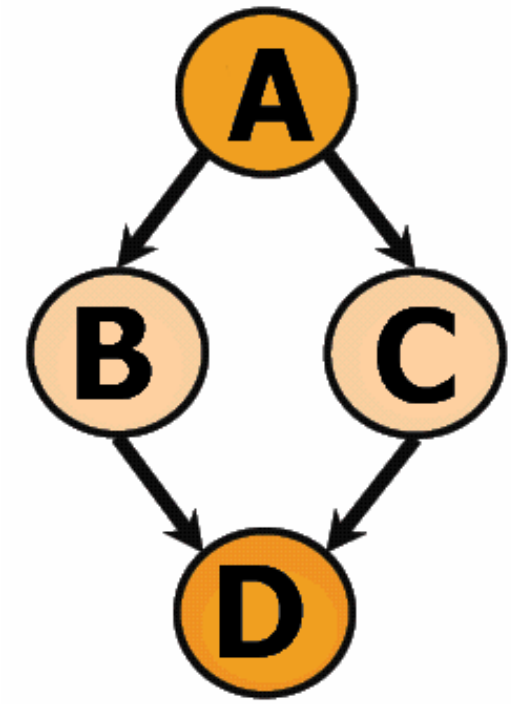

Abbildung 2: Bayesgraph mit 4 Knoten

P(B) ist hier die allgemeine W., dass *B* auftritt. Dazu werden alle W. aufsummiert, für die Merkmal *B* eintritt. Analog dazu erfolgt die Berechnung von *P(A).*

Die Verbund-W. berechnet sich mit:

$$P_{verbund}(B) = \sum_{\alpha} P(B) * P(B \mid par(B))$$

α steht hier für alle Elternknoten von *B*, sodass die gemeinsame Summe berechnet wird.

Für die gemeinsame W., also das gleichzeitige Eintreffen von mehreren Merkmalen wird die Kettenregel angewandt:

$$\prod P(V_j)$$

Somit ist also:

$$\boldsymbol{P(A,B) = P(A)*P(B)}$$

Für das Bsp. aus Abb. 2 gilt:

$$\boldsymbol{P(D|A,B,C) = P(D|par(B)) = P(D|B,C)}$$

Da *B* und *C* jeweils von *A* abhängen, reicht es, die gemeinsame W. der Elternknoten von *D*, also *B* und *C*, zu berechnen.

2.5 Beispiel Lungendiagnose

Im folgenden Bsp. wird nun ein Bayesgraph optimiert und die gemeinsame W. zweier Merkmale bestimmt. Der Graph beschreibt die Häufigkeit, mit der die Einflüsse Asienbesuch/Raucher auf verschiedene Krankheitsbilder und deren Diagnose/Folgekrankheiten wirken.

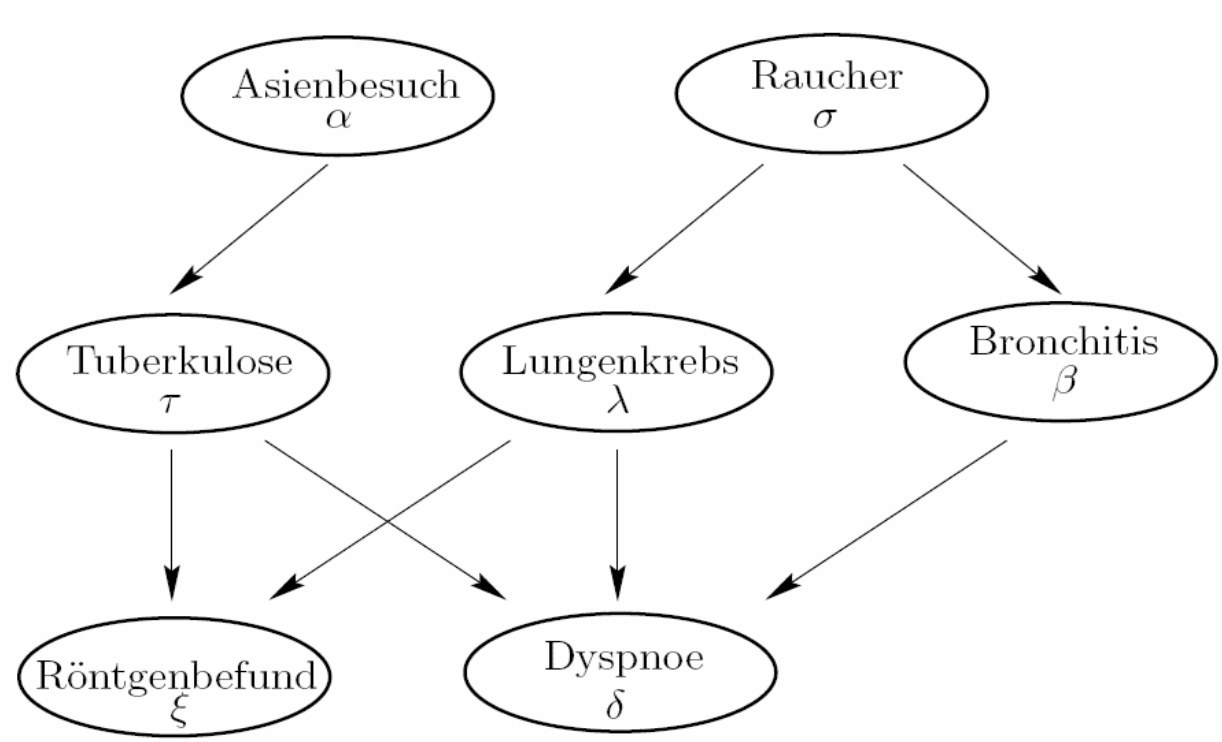

Abbildung 3: Erster Ansatz für einen Bayesgraphen der Lungendiagnose

2.5.1 Optimierung des Graphen

Da Tuberkulose und Lungenkrebs beide jeweils mit Röntgenbefund und Dyspnoe (Atemnot) zusammenhängen, lässt sich hier mit einem zusätzlichen „Indikatorzustand“ die Dimension des Netzes reduzieren und somit Rechenzeit sowie Speicher sparen. Dazu wird ein neuer Knoten „Tuberkulose oder Lungenkrebs“ erzeugt, dessen Wahrscheinlichkeits-Tafel nur aus den Wahrscheinlichkeiten *0* und *1* besteht.

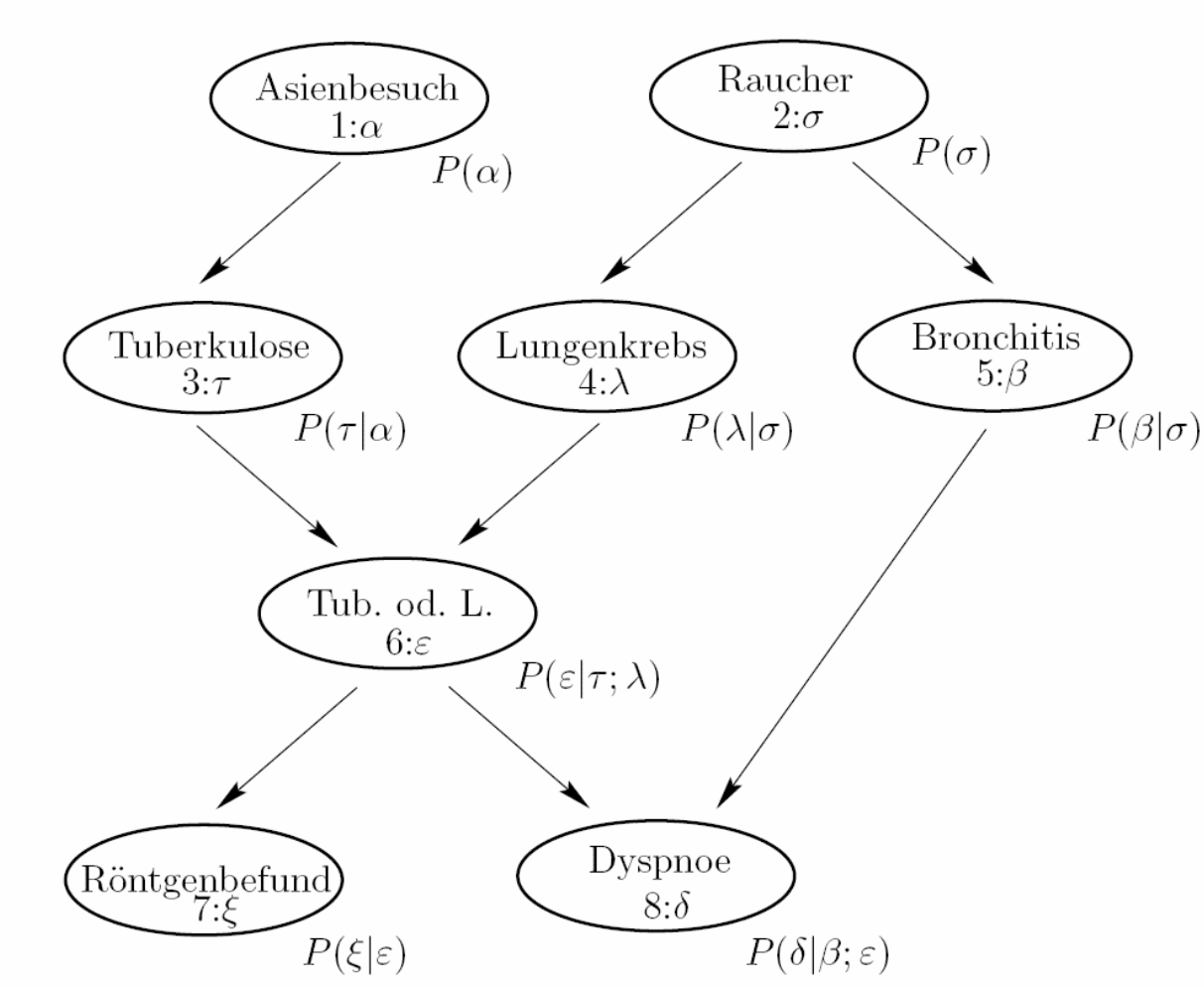

Abbildung 4: Optimierter Graph der Lungendiagnose mit indizierten Knoten und zu definierenden W.

2.5.2 Beispiel-Berechnung für gleichzeitiges Erkranken an Tuberkulose und Bronchitis

Nun soll die W., mit der ein Patient gleichzeitig an Tuberkulose und Bronchitis erkrankt, berechnet werden. Dazu wird die Wahrscheinlichkeits-Tabelle zum Bayesgraphen gebraucht, welche im gegebenen Fall durch statistische Erhebungen vorliegt.

In der nebenstehenden Tabelle sind die komplementären Merkmalsausprägungen der Übersichtlichkeit halber nicht aufgeführt. Diese lassen sich einfach durch *(1 – Wahrscheinlichkeitswert)* berechnen.

Tabelle 1: Bedingte W. für den Lungendiagnose-Graphen

Knoten	$P(\dots)$	Wert
$1{:}\alpha$	α	0,01
$2{:}\sigma$	σ	0,50
$3{:}\tau$	$\tau \mid \alpha$	0,05
	$\tau \mid \neg\alpha$	0,01
$4{:}\lambda$	$\lambda \mid \sigma$	0,10
	$\lambda \mid \neg\sigma$	0,01
$5{:}\beta$	$\beta \mid \sigma$	0,60
	$\beta \mid \neg\sigma$	0,30
$6{:}\varepsilon$	$\varepsilon \mid \lambda; \tau$	1,00
	$\varepsilon \mid \lambda; \neg\tau$	1,00
	$\varepsilon \mid \neg\lambda; \tau$	1,00
	$\varepsilon \mid \neg\lambda; \neg\tau$	0,00
$7{:}\xi$	$\xi \mid \varepsilon$	0,98
	$\xi \mid \neg\varepsilon$	0,05
$8{:}\delta$	$\delta \mid \beta; \varepsilon$	0,90
	$\delta \mid \beta; \neg\varepsilon$	0,80
	$\delta \mid \neg\beta; \varepsilon$	0,70
	$\delta \mid \neg\beta; \neg\varepsilon$	0,10

Aus den in 2.4 angegebenen Regeln ergibt sich für die Berechnung folgende Formel:

$$
\begin{aligned}
P(\tau,\beta) &= P_{verbund}(\tau) * P_{verbund}(\beta) \\
&= \left(\sum_{\alpha} P(\alpha) P(\tau \mid \alpha) \right) \\
&\quad * \left(\sum_{\sigma} P(\sigma) P(\beta \mid \sigma) \right) \\
&= (0{,}01 * 0{,}05 + 0{,}99 * 0{,}01) \\
&\quad * (0{,}5 * 0{,}6 + 0{,}5 * 0{,}3) \\
&= 0{,}00468
\end{aligned}
$$

Somit erkrankt ein Patient in unserem Modell mit 0,468 % W. gleichzeitig an Tuberkulose und Bronchitis.

3 Lernen mit Bayesschen Netzwerken

Zur Beschreibung eines Bayesnetzes werden zwei Dinge benötigt:

- die **Netztopologie** und
- die **Parameter** für jeden Knoten.

Es ist möglich, beides aus gegebenen Daten zu erlernen, wobei das Strukturlernen deutlich aufwändiger ist. Wenn die Daten unvollständig sind, oder es versteckte Knoten gibt, kann das Lernen noch schwieriger werden.

Zunächst wird auf das Parameterlernen bei bekannter Graphenstruktur eingegangen.

3.1 Lernen der Parameter

Zur Anwendung kommt der Maximum-Likelihood-Schätzer ***L***.

Die Ausgangsdaten sind:

- Trainingsdaten ***D*** mit ***N*** unabhängigen Datensätzen
- Topologische Struktur des Bayesgraphen mit indizierten Zufallsvariablen X_i

Zur Verdeutlichung wird die Theorie anhand eines Beispiels vermittelt:

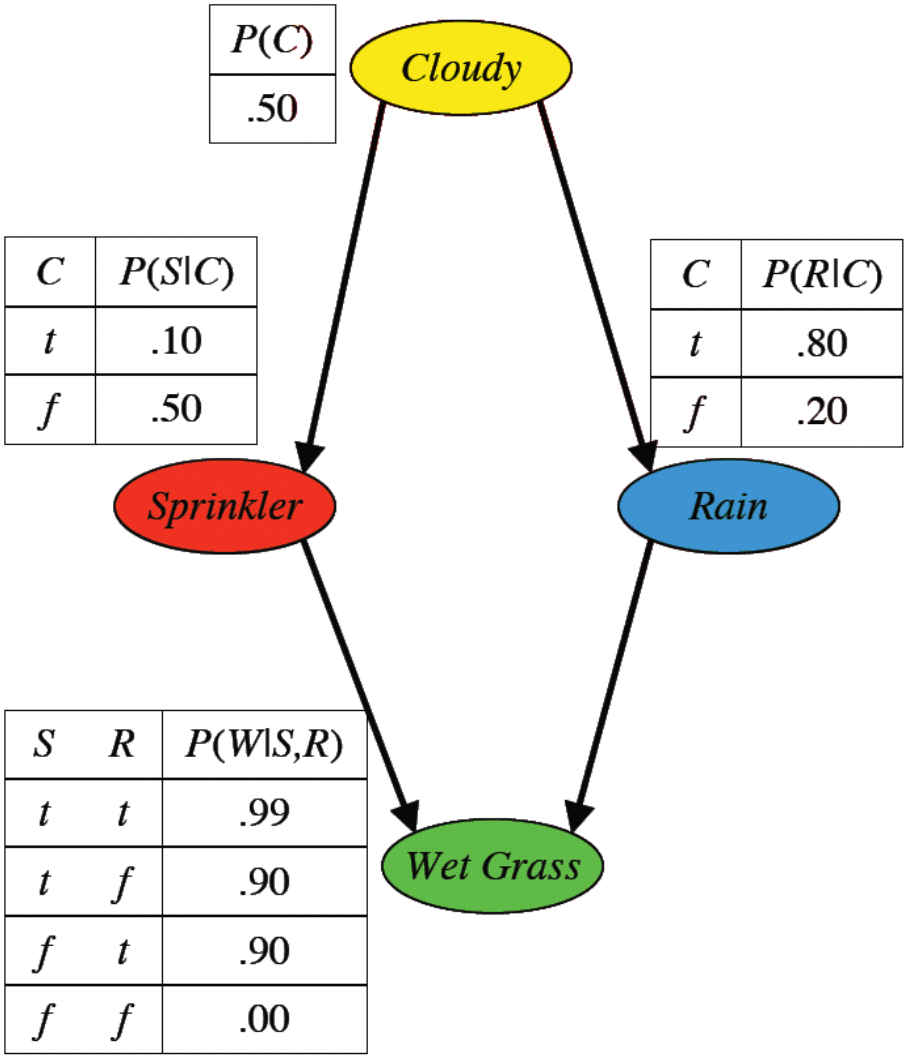

Abbildung 5: Topologische Struktur zur Erzeugung des Bayesgraphen (t = 1 = true; f = 0 = false)

Unter der Vorraussetzung, dass nur Trainingsdaten *D* und ein indizierter Graph bekannt sind, gilt es, für jeden Knoten die Wahrscheinlichkeits-Verteilung zu finden. Dadurch wird der Likelihood-Schätzer *L* der Trainingsdaten maximiert. Die normalisierte log-L-Funktion von *D* bildet die Summe über alle Knoten:

$$L = \frac{1}{N} \sum_{i=1}^{m} \sum_{k=1}^{S} \log P\left(X_i \mid par\left(X_i\right), D_k\right)$$

Zu erkennen ist, dass sich die Summe entsprechend der Struktur des Graphen aufteilt. Folglich kann der Beitrag jedes Knotens zur log-L-Funktion unabhängig maximiert werden (angenommen, die Parameter jedes Knotens sind unabhängig).

Nun soll die bedingte-Wahrscheinlichkeits-Tabelle für den „Wet Grass"-Knoten mit Hilfe einiger Trainingsdaten geschätzt werden. Dazu wird gezählt, wie oft das Grass nass war, als es regnete und der Sprinkler an war → *N(W=1, S=1, R=1)*, oder es wird gezählt, wie oft das Grass bei Regen nass war und der Sprinkler aus war → *N(W=1, S=0, R=1)*, etc.

Mit diesen Zählungen (welche statistisch genug sind) kann der Maximum-Likelihood-Schätzer der bedingten Wahrscheinlichkeits-Verteilung wie folgt gefunden werden:

$$P(W = w \mid S = s, R = r) \approx \frac{N(W = w, S = s, R = r)}{N(S = s, R = r)} = \frac{N(W = w, S = s, R = r)}{N(W = 0, S = s, R = r) + N(W = 1, S = s, R = r)}$$

"Das Lernen" beschränkt sich also nur auf das Auszählen der Trainingsdaten (im Falle von multinomialer Verteilung). Für Gauß'sche Parameter kann der Erwartungswert und die Varianz berechnet, zur Bestimmung der Gewichtungsmatrix lineare Regression benutzt werden. Für andere Arten von Verteilungen bedarf es komplexerer Verfahren.

Bei spärlicher Trainingsdatenmenge führen Maximum-Likelihood-Schätzungen zur Bestimmung von Wahrscheinlichkeits-Verteilungen leicht zu stark verfälschten Ergebnissen. Dies lässt sich mit Dirichlet-Verteilungen (Pseudozählungen) unter Anwendung des Bayes-Satzes lösen, woraus eine Maximum a posteriori (MAP)-Schätzung hervorgeht. Für Gauß'sche Verteilungen eignet sich eine Wishart-Verteilung (multivariate Entsprechung der χ^2-Verteilung).

3.2 Lernen der Netztopologie

Im Folgenden wird angenommen, dass Vorwissen über die Parameter θ und entsprechende Daten ***D*** bekannt sind, nicht aber die Topologie des zugehörigen Bayesnetzes. Die Graphenstruktur legt fest, welche Zufallsvariablen in der jeweiligen Parametrisierung (bedingt) unabhängig voneinander sind. Umgekehrt bedingt die Menge der Unabhängigkeiten aber nicht die Topologie des Graphen.

Um nun eine möglichst gute Netzstruktur zu bekommen wird von einer „beliebigen" (hypothetischen) Netzstruktur $\boldsymbol{S^h}$ ausgegangen. Diese wird schrittweise durch Hinzufügen und Entfernen von Kanten verbessert, bis keine Verbesserungen mehr möglich sind. Dieses Suchverfahren nennt sich *Greedy-Search* und wir weiter unten beschrieben. Um Netzstrukturen vergleichen zu können bedarf es einer Bewertungsfunktion für ein vorgegebenes System aus Daten D und Topologie S^h. Gesucht ist also $p(S^h|D)$. Folgendes liefert der Bayessatz:

$$p(S^h \mid D) = \frac{p(S^h) * p(D \mid S^h)}{p(D)}$$

$P(D)$ ist eine Normalisierungskonstante, da sie unabhängig von der Netzstruktur ist. $p(S^h)$ spielt zunächst keine Rolle, da zu Beginn alle Topologien als gleich wahrscheinlich angenommen werden. Daher lässt sich $p(D|S^h)$ als alleiniges Bewertungskriterium für S^h heranziehen. $p(D|S^h)$ lässt sich folgendermaßen bestimmen:

$$p(D \mid S^h) = \int p(D, \theta \mid S^h) d\theta$$

wobei θ der Satz an Parametern ist. Um die Grundvoraussetzung eines Bayesnetzes zu erfüllen muss die Gleichung noch auf eine multinomiale und multivarite Variable erweitert werden und es ergibt sich die von Cooper & Herskovits erstmals 1992 vorgestellte Gleichung:

$$p(D \mid S^h) = \prod_{i=1}^{n} \prod_{j=1}^{q_i} \frac{\Gamma(\alpha_{ij})}{\Gamma(\alpha_{ij} + N_{ij})} * \prod_{k} \frac{\Gamma(\alpha_{ijk} + N_{ijk})}{\Gamma(\alpha_{ijk})}$$

N_{ijk} ➔ Anzahl der Beobachtungen in D in denen $X_i = k$ und $par(i) = j$

N_{ij} ➔ Werte der marginalen Tafel $\sum_k N_{ijk}$

α_{ijk} ➔ Vorwissen über die Häufigkeit des Auftretens von $X_i = k$ und $par(i) = j$

α_{ij} ➔ Werte der marginalen Tafel $\sum_k \alpha_{ijk}$

Ändert sich bei Greedy-Search nun die Netzstruktur durch Hinzufügen und Entfernen einer Kante, so muss nun auch das Vorwissen α neu berechnet werden → α'. Dies lässt sich folgendermaßen bewerkstelligen:

$$p_{neu}(X_i, par_{neu}(i) \mid S^h_{neu}) = \sum_{X/X_i, par^{neu}(i)} p(X \mid S^h_{orig})$$

$$\alpha_i^{'} = UserSampleSize * p_{neu}(X_i, par_{neu}(i) \mid S^h_{neu})$$

$par_{neu}(i)$ ➔ Veränderte Elternmenge von X_i

UserSampleSize ➔ Vorwissen wird mit UserSampleSize multipliziert, um evt. fehlendes Vorwissen durch Imaginärwissen zu besetzen

Beim Hinzufügen / Entfernen hat dies jeweils nur Auswirkungen auf die Elternknoten. Also müssen nur Diese neu berechnet werden.

3.2.1 Entfernen einer Kante

Bei Entfernen einer Kante wird folgendes berechnet:

$$p_{neu}(X_i, parn_{neu}(i) \mid S^h_{neu}) = \sum_{par_{del}(i)} p_{akt}(X_i, par_{akt}(i) \mid S^h_{akt})$$

$$\left[= \sum_{X/\{X_i, par_{neu}(i)\}} p(X \mid S^h_{orig}) \right]$$

$par_{akt}(i)$ ➔ Elternmenge von X_i in S^h_{akt}

$par_{del}(i)$ ➔ Menge der aus $par_{akt}(i)$ entfernten Knoten ($par_{del}(i) = par_{akt}(i) / par_{neu}(i)$)

3.2.2 Hinzufügen einer Kante

Beim Hinzufügen einer Kante das folgende:

$$p_{neu}(X_i, par_{neu}(i) \mid S^h_{neu}) = p_{neu}(X_i, par_{akt}(i) \cup par_{add}(i) \mid S^h_{neu})$$
$$= p_{akt}(X_i, par_{akt}(i) \mid S^h_{akt}) * pv_{orw}(par_{add}(i) \mid S^h)$$

$$\left[= \sum_{X/\{X_i, par_{neu}(i)\}} pv_{orw}(X \mid S^h) \right]$$

$par_{add}(i)$ ➔ neuer Elternknoten von X_i in S^h_{neu}

$pv_{orw}(par_{add}(i) \mid S^h)$ ➔ a-priori-Wahrscheinlichkeitstafel zu $par_{add}(i)$,

sie lässt sich vorab zu jedem Knoten X_i berechnen mit:

$$pv_{orw}(X_i \mid S^h) = \sum_{X/\{X_i} p_{orig}(X_i, par_{orig}(i) \mid S^h_{orig})$$

Bei Greedy-Search wird nun jeweils die neu berechnete Topologie (nach dem Hinzufügen bzw. Entfernen) bewertet und mit Der der Alten verglichen. So verbessert sich die Topologie schrittweise. Als Ausgangstopologie dient ein einfacher Knoten.

3.3 Anwendungen

Bayessche Netzwerke finden in den verschiedensten Bereichen Anwendung, so z. B. in der Bioinformatik, Medizin, in den Ingenieurwissenschaften und in der Epidemiologie. Ein großer Schwerpunkt liegt bei Systemen mit Künstlicher Intelligenz.

Ein Schachcomputer beispielsweise analysiert die Spielweise des Gegners mit Hilfe eines Bayesnetzes um dessen nächsten Zug vorhersagen zu können. Aus der gewonnenen W. für mögliche gegnerische Züge kann der Computer dann einen möglichst vorteilhaften eigenen Zug machen.

Bei Office von Microsoft gibt es seit Version 97 einen Hilfe-Agenten, der Fragen zur Bedienung des Programms beantworten kann. Dahinter steckt ein Bayesnetz, welches auf die Analyse von Struktur und Inhalt der Eingabe trainiert wurde. Mit dessen Hilfe kann der Frage-Kontext der enthaltenen und bekannten Stichwörter erkannt werden und die entsprechende Hilfeseite aufgerufen werden.

4 Zusammenfassung

Mit Bayesschen Netzwerken können hochdimensionale Wissensmengen in einfachen Graphen mit Wahrscheinlichkeitsverteilungen dargestellt werden. Aus dem Erfahrungswissen heraus lassen sich durch verschiedene Methoden auch Verteilungsfunktionen erstellen. Ist das Bayesnetz erst einmal bekannt, können die verschiedensten W. damit berechnen werden.

Trotz der vielen Möglichkeiten von Bayesnetzen, weist dieser Wissensspeicher-Ansatz dennoch einige Schwächen auf. Bei einer bekannten Reihenfolge von Ursache und Wirkung ist die Aufstellung eines Bayesnetzes noch relativ einfach. Wenn jedoch zu wenig Abhängigkeiten bekannt sind, kann es bei ungünstiger Indizierung zu dicht besetzten Bayesnetzen kommen, was wiederum bei der Berechnung von W. viel Rechenleistung und Speicher in An-

spruch nimmt. Die benötigten Wahrscheinlichkeiten müssen also alle zumindest ansatzweise bekannt sein.

Ein weiterer Nachteil ist die gerichtete Struktur eines Bayesnetzes. So können die bedingten Unabhängigkeiten jeweils nur bezüglich aller ihrer Vorgänger dargestellt werden. Außerdem können zyklische Abhängigkeiten nicht beschrieben werden, da diese im Bayesgraph nicht auftauchen dürfen. Somit ist es im Allgemeinen nicht ohne Hilfsmodelle möglich, alle vorhandenen Unabhängigkeiten der Ursprungsverteilung in Bayesnetzen zu nutzen/darzustellen.

Trotz dieser Nachteile nutzen heutzutage viele Expertensysteme das Bayessche Netzwerk-Modell als flexible, lernfähige Wissensbasis.

Literatur

[DuHaSt00] R. Duda, P. Hart, D. Stork, Pattern Classification, Wiley Interscience, 2000

[Hecker95] D. Heckerman, A Tutorial on Learning With Bayesian Networks, Techical Report MSR-TR-95-06, 1995

[Mitche97] T. Mitchell, Machine Learning, McGraw Hill, 1997

[Rullhu01] D. Rullhusen: Probabilistische Wissensrepräsentation mittels Bayes-Netzen, 2001

[Murphy98] Kevin Murphy, A Brief Introduction to Graphical Models and Bayesian Networks, www.cs.ubc.ca/~murphyk/Bayes/bayes.html, 1998

[Tresch06] Achim Tresch, Maschinelles Lernen: Bayes-Netze, 2006