

Statistische Verfahren in der Künstlichen Intelligenz, Bayesische Netze

Erich Schubert

6. Juli 2003

Zitat von R. P. Feynman

Richard P. Feynman (Nobelpreisträger in Physik):

“Chance” is a word which is in common use in everyday living. By chance, we mean something like a guess. Why do we make guesses? We make guesses when we wish to make a judgment but have incomplete information or uncertain knowledge... Sometimes we make guesses because we wish, with our limited knowledge, to say *as much as we can* about some situation... There are good guesses and there are bad guesses. *The theory of probability is a system for making better guesses.* The language of probability allows us to speak quantitatively about some situation which may be highly variable, but which does have some consistent average behavior.

Statistische Verfahren

Der Begriff „statistische Verfahren“ wird gerne intuitiv falsch interpretiert: diese Verfahren machen nicht „automatisch“ mehr Fehler als andere.

Statistik zu verwenden heisst hier vor allem:

1. unvollständige Datenspeicherung (um die Komplexität zu beschränken)
2. „uncertain reasoning“ (Schließen unter Ungewissheit)
3. Fehlerabschätzung
4. Optimalitätsuntersuchung

Formel von Bayes

- $P(B|A) = \frac{P(A \cap B)}{P(A)}$ ist die Formel für die bedingte Wahrscheinlichkeit von „ B tritt ein, wenn A bereits eingetreten ist“.
- Die Produktregel besagt $P(A \cap B) = P(A) \cdot P(B|A) = P(B) \cdot P(A|B)$
- Für paarweise disjunkte Ereignisse A_i mit $B \subset \bigcup_i A_i$ gilt:
 $P(B) = \sum_i P(A_i) \cdot P(B|A_i)$ – man kann B „zerlegen in die Fälle A_i “.

Nun betrachten wir $P(A_j|B)$ (j beliebig) – wir vertauschen die Rollen:

$$P(A_j|B) = \frac{P(B \cap A_j)}{P(B)} = \frac{P(A_j) \cdot P(B|A_j)}{P(B)} = \frac{P(A_j) \cdot P(B|A_j)}{\sum_i P(A_i) \cdot P(B|A_i)}$$

Dies ist die bekannte „BAYES’SCHES FORMEL“.

Bedeutung der Formel von Bayes

Um die Bedeutung der Formel von Bayes zu verstehen kann man sich A_j und B folgendermaßen vorstellen:

- B ist ein Effekt (Wirkung, Symptom – z.B. „Fieber“),
- A_j sind verschiedene Ursachen (z.B. „Schnupfen“, „Grippe“, „Malaria“)

Dann erlaubt uns die Regel von Bayes bei

- bekannten Abhängigkeiten der Wirkung von den Ursachen – $P(B|A_j)$
- bekannten Wahrscheinlichkeiten der Ursachen – $P(A_j)$

auf die Wahrscheinlichkeit zu schließen, *dass der Effekt von einer bestimmten Ursache ausgelöst wird*. (Hier also z.B. abschätzen, ob das Fieber von einem Schnupfen oder einer Grippe kommt.)

Beispiel für die Bayes'sche Formel

Wir haben in einer E-Mail das Wort „free“ gefunden („Ereignis B “).

- Das Wort „free“ kommt in $P(B|A_s) = 80\%$ aller unerwünschten und $P(B|A_h) = 5\%$ der erwünschten E-Mails vor,
- insgesamt in $P(B) = 35\%$ der E-Mails.
- $P(A_s) = 40\%$ sind unerwünschte E-Mails, $P(A_h) = 60\%$ sind erwünscht.

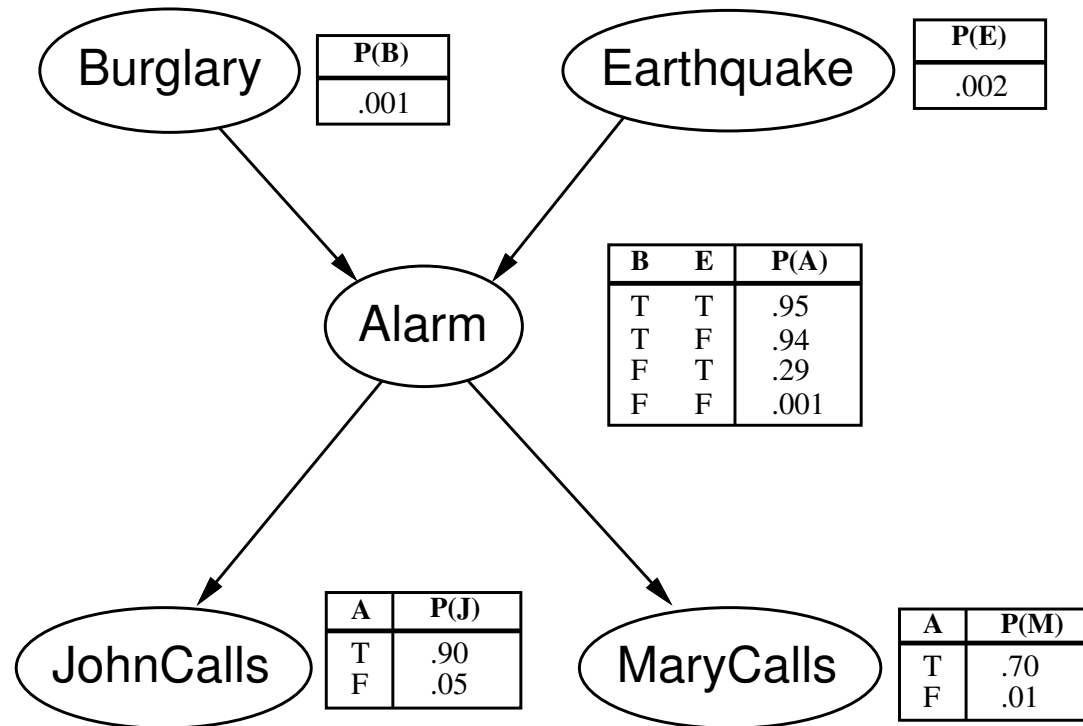
Nach der Formel von Bayes gilt:

$$P(A_s|B) = \frac{0.40 \cdot 0.80}{0.40 \cdot 0.80 + 0.60 \cdot 0.05} \approx 91.4\%$$

$$P(A_h|B) = \frac{0.60 \cdot 0.05}{0.40 \cdot 0.80 + 0.60 \cdot 0.05} \approx 8.6\%$$

Bayes'sche Netze

Man bezeichnet als „Bayes'sches Netz“ (auch: belief network, causal network, probabilistic network, knowledge map) einen gerichteten, zyklensfreien Graphen, wobei für jeden Knoten eine Wahrscheinlichkeitsverteilung in Abhängigkeit seiner Elternknoten gegeben ist.



Anwendungen für Bayes-Netze:

Typische Anwendungen für solche Bayes-Netze:

1. *Entscheidungsfindung*: Ermitteln der Wahrscheinlichkeiten bestimmter Ereignisse für die Wertungsfunktion eines Agenten
2. *Gezielte weitere Abfragen*: Ermitteln, welche zusätzlichen Informationen notwendig sind, eine bestimmte Entscheidung „sicher“ treffen zu können
3. *Sensitivitätsanalyse*: Welche Elemente im Modell haben den größten Einfluss (sollten also möglichst genau sein)
4. *Erklärung von Entscheidungen*: Was ist die wahrscheinlichste Erklärung für das beobachtete Ereignis

Komplexität der Anfragen:

- Ist der Graph ein sogenannter Polytree, so ist die *exakte Inferenz* in linearer Zeit möglich.
- Im Allgemeinen ist das Problem aber *nur exponentiell lösbar*. Es ist nicht schwer zu sehen dass sich das SAT-Problem hier abbilden lässt; aber das Problem ist sogar *#P-hard*, also echt schwerer als NP-vollständig („number-P hard“: so schwer wie die Berechnung der *Anzahl* der Lösungen von SAT).
- In der Praxis spielen deswegen approximative Verfahren (Monte-Carlo-Methoden) eine immer größere Rolle, bei denen das Netz sozusagen „simuliert“ wird.

Naive Bayes

„Naive Bayes“-Netze sind sehr einfache Bayes-Netze mit

- nur einer „Ursache“ und (bei festem Wert der Ursache) voneinander unabhängigen „Wirkungen“.
- linearen Speicherbedarf, also ideal für große Probleme
- gilt als einer der besten Allzweck-Lern-Algorithmen
- schnell zu berechnen, schnell zu trainieren
- beispielsweise in Spam-Filtern (Mozilla, Spamassassin, ...) verwendet
- überraschend gute Ergebnisse auch bei ignorierter Abhängigkeit

Leistungsfähigkeit Bayes'scher Netze:

Bayes'sche Netze dienen derzeit vor allem als Expertensysteme.

Das bekannteste wohl ist das PATHFINDER-System zur Diagnose von Lymphknotenerkrankungen. Dieses umfasst 100 Symptome und 14.000 Wahrscheinlichkeiten um 100 Krankheiten zu diagnostizieren. Dieses System ist angeblich mittlerweile besser in der Diagnose als die besten Experten.

Aber: dieses Netz wurde von Experten erstellt und von diesen mit Wahrscheinlichkeitstabellen gefüllt.

— Geht so etwas nicht auch automatisch?

Lernen von Bayes'schen Netzen

Beim Lernen von Bayes'schen Netzen muss man im wesentlichen vier Fälle unterscheiden:

Struktur	Beobachtbarkeit	Verfahren
vorgegeben	vollständig	Maximum-Likelihood-Schätzer
vorgegeben	unvollständig	Expectation-Maximization
unbekannt	vollständig	Suchen im Modellraum
unbekannt	unvollständig	EM+Suchen im Modellraum

Bei vorgegebener Struktur müssen nur die CPTs berechnet werden.

- Vollständig beobachtbar: direkte Berechnung
- Unvollständig beobachtbar: CPTs für versteckte Variablen müssen komplizierter berechnet werden

Das EM-Verfahren

Der EM-Algorithmus ist ein allgemeines Verfahren für Lernprobleme.

- Startparameter wählen (zufällig oder fest)
- *Expectation step*: Berechnen, wie „typisch“ diese Beobachtungen sind
- *Maximization step*: Parameter optimieren für diese „Beobachtungen“

Seien \vec{x} die Beobachtungen, \vec{Z} die versteckten Variablen und $\vec{\theta}$ die Parameter. Dann lässt sich der EM-Algorithmus abstrakt beschreiben als

$$\vec{\theta}_{i+1} = \operatorname{argmax}_{\vec{\theta}} \sum_{\vec{z}} \underbrace{P(\vec{Z} = \vec{z} | \vec{x}, \vec{\theta}_i)}_{\text{Gewichtung}} \underbrace{L(\vec{x}, \vec{Z} = \vec{z} | \vec{\theta})}_{\text{„Erklärbarkeit“}}$$

Lernen von Strukturen

- Aufwendig: Abhängigkeit und Zusammenhänge untersuchen

Wie stark sich die fehlerhafte Feststellung der Ursache-Wirkungs-Zusammenhänge auf die Struktur des Netzes auswirken kann sieht man in der folgenden Abbildung:

