Grundlagen der Künstlichen Intelligenz

11 Bayessche Netze

Struktur, Semantik, Konstruktion, Inferenz

Volker Steinhage

Inhalt

- Motivation
- Bayessche Netze
- Bayessche Netze und Verbundwahrscheinlichkeitsverteilungen
- Bayessche Netze und bedingte Unabhängigkeiten
- Konstruktion von Bayesschen Netzen
- Inferenz in Bayesschen Netzen
- Exakte und approximative Inferenz

Motivation (1)

Die Verteilung der *Verbundwahrscheinlichkeiten*

	zahnschmerzen		¬ zahnschmerzen	
	verfangen	¬verfangen	verfangen	¬verfangen
loch	0.108	0.012	0.072	0.008
¬loch	0.016	0.064	0.144	0.576

- erlaubt die Beantwortung aller Anfragen an die Domäne bzgl. unbedingter und bedingter Wahrscheinlichkeiten,
  - indem die Anfragen als Disjunktion über den atomaren Ereignissen formuliert und die entspr. Wahrscheinlichkeiten aufaddiert werden.
- wächst aber exponentiell in der Zahl der Zufallsvariablen und ist bzgl. der Erfassung der Verbundwahrscheinlichkeiten nicht naheliegend und einfach.

Motivation (2)

- Die Bayessche Regel zusammen mit der Annahme von absoluten und bedingten Unabhängigkeiten zwischen den Zufallsvariablen erlaubte dagegen eine effiziente Reduzierung der zur Anfragebeantwortung erforderlichen unbedingten und bedingten Wahrscheinlichkeiten.

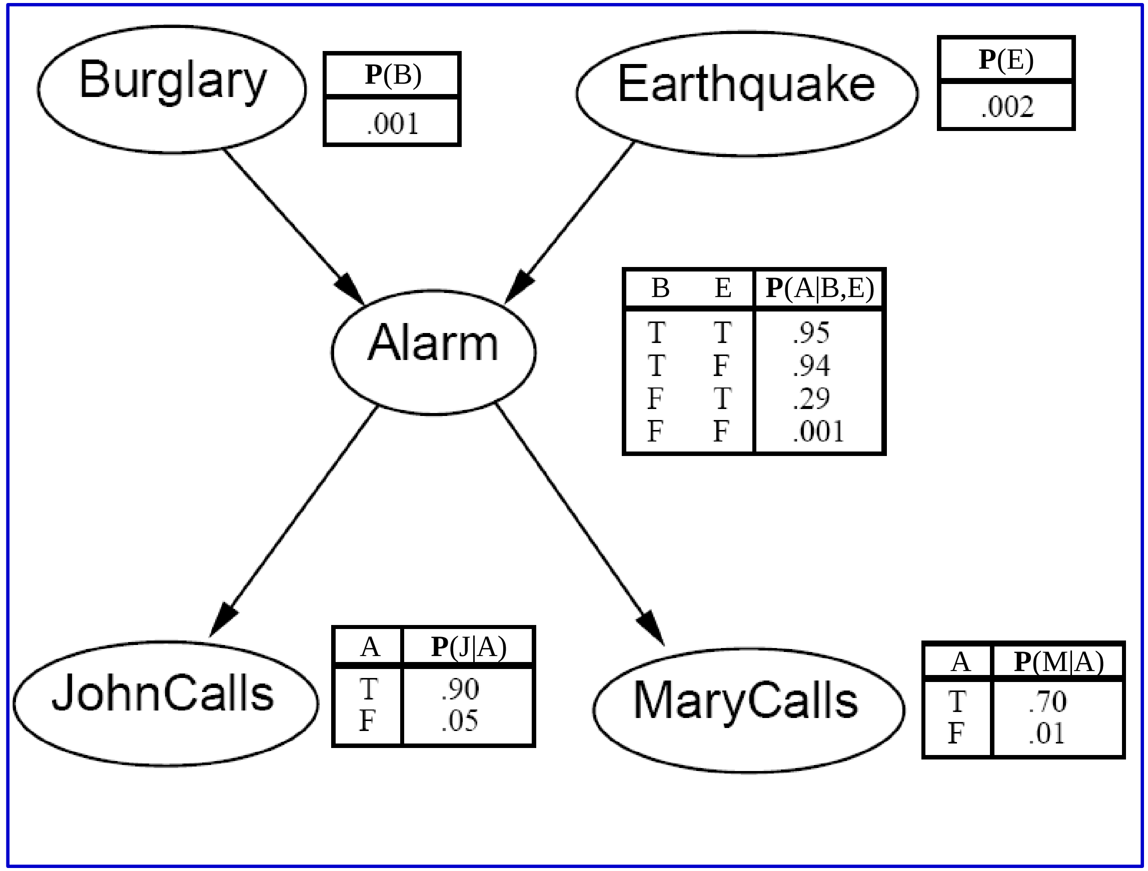
- Bayessche Netze sind eine Repräsentationsform für die Darstellung und Verarbeitung von Abhängigkeiten zwischen Zufallsvariablen und der Verteilung der Verbundwahrscheinlichkeiten.

Struktur von Bayesschen Netzen*

1) Knoten: *Zufallsvariablen*

2) Gerichtete Kanten zwischen Knoten: *direkte* Einflüsse

3) Jeder Knoten hat eine Tabelle von bedingten W‘keiten (*Conditional Probability Table, CPT*),



die den Effekt der Elternknoten auf den Knoten quantifiziert.

4) Der Graph ist azyklisch; also ein DAG.

---

* auch *belief networks, probabilistic networks, causal networks*)

Semantik von Bayesschen Netzen

Zwei Zugänge zum Verständnis von Bayesschen Netzen (BN):

1. BN repräsentiert die Verbundwahrscheinlichkeitsverteilung der Zufallsvariablen.

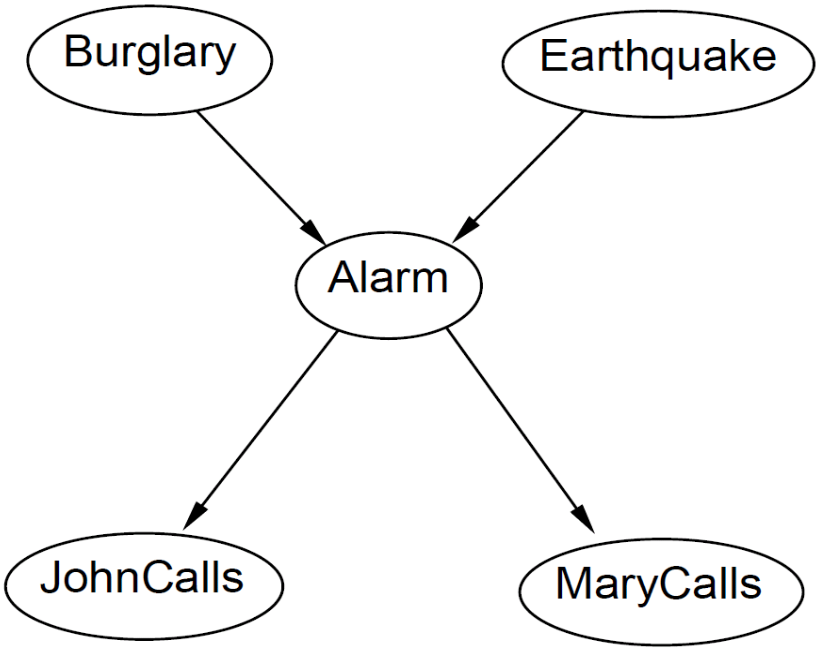
   $\rightsquigarrow$ Geeignete Sichtweise für die Konstruktion des BNs.

2. BN kodiert eine Menge von Unabhängigkeitsannahmen.

   $\rightsquigarrow$ Geeignete Sichtweise zur Konstruktion von Inferenzen.

Beispiel von Juda Pearl

- Im Haus ist eine Einbruchsicherung (*Alarm*) installiert, die auf Einbrüche (*Burglary*), aber auch auf leichte Erdbeben (*Earthquake*) reagiert.

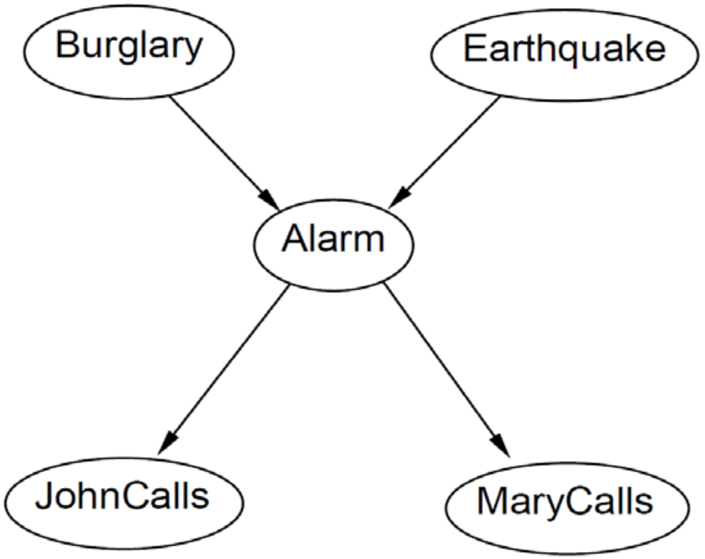


- Ihre Nachbarn John und Mary sagen zu, Sie bei Alarm im Büro anzurufen (*JohnCalls* bzw. *MaryCalls*).

- John verwechselt manchmal das Telefonläuten mit Alarm und ruft dann auch an.

- Mary hört manchmal laute Musik und überhört dann den Alarm.

Kodierung von Unabhängigkeitsannahmen in Bayesschen Netzen



- Allgemein: BNs repräsentieren *vollständig* die Abhängigkeiten von direkten Elternknoten.
- am Beispiel:
  - *Alarm* hängt *nur* von *Burglary* und *Earthquake* ab.
  - *MaryCalls* hängt *nur* von *Alarm* ab.

    ⇝ *MaryCalls* ist also unabhängig von *JohnCalls*, *Earthquake* und *Burglary*:

    $\mathbf{P}(\mathit{MarryCalls}|\mathit{JohnCalls}, \mathit{Alarm}, \mathit{Earthquake}, \mathit{Burglary}) = \mathbf{P}(\mathit{MaryCalls}|\mathit{Alarm})$

⇝ Bayessche Netze kodieren damit also auch Unabhängigkeitsannahmen.

Bayessche Netze und Verbundwahrscheinlichkeit

Ein BN sind als kompakte Repräsentation einer Verbundwahr'keitsverteilung interpretierbar.

Jedes atomare Ereignis der Verteilung ist eine Konjunktion einer bestimmten Wertebelegung $P(X_1 = x_1 \wedge \ldots \wedge X_n = x_n)$, abgekürzt: $P(x_1, \ldots, x_n)$. Für jedes atomare Ereignis gilt nach Produktregel:

$$
\begin{aligned}
\boxed{P(x_1, \ldots, x_n) \quad = \quad P(x_n \mid x_{n-1}, \ldots, x_1) \cdot \ldots \cdot P(x_2 \mid x_1) P(x_1)} \\
= \quad \prod_{i=1}^{n} P(x_i \mid x_{i-1}, \ldots, x_1).
\end{aligned}
$$

Wegen der Unabhängigkeitsannahmen ist dies äquivalent zu:

$$
P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i \mid parents(X_i)).
$$

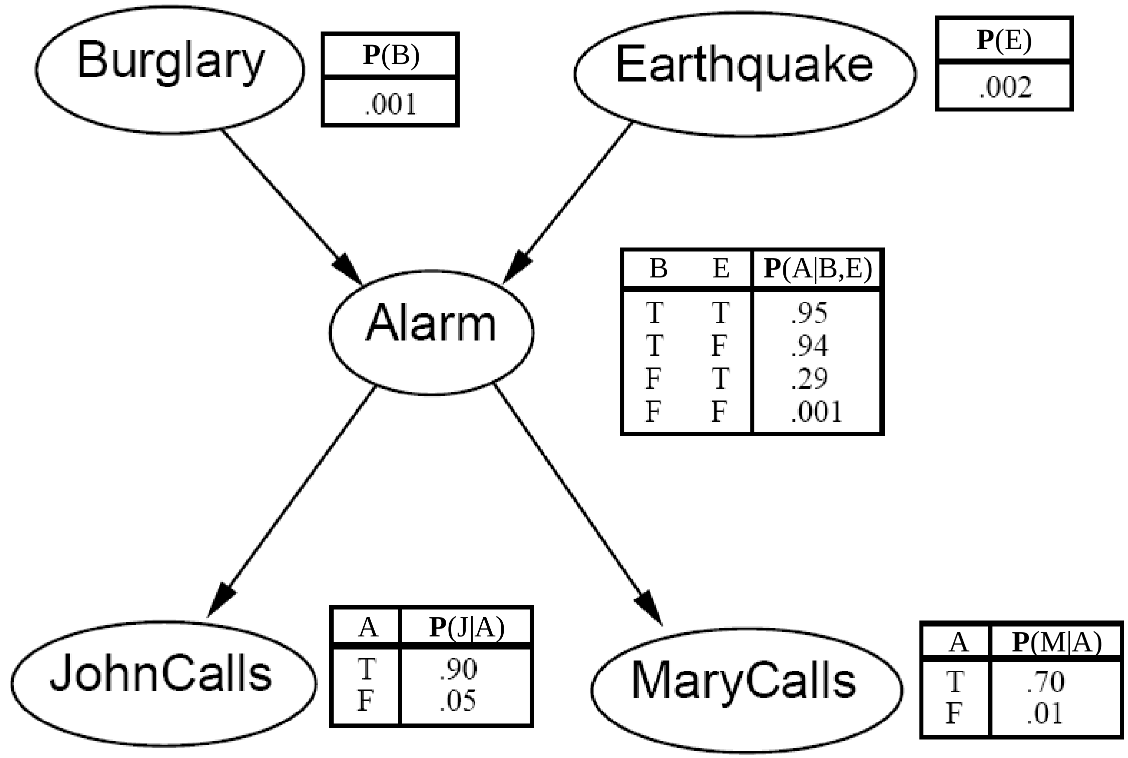
D.h. mit der Netztopologie und den CPT können wir die Verbundw'keit berechnen!

Beispiel für Berechnung eines atomaren Ereignisses

- W‘keiten für negative Ereignisse ergeben sich als

$$P(\neg x) = 1 - P(x).$$

- W‘keiten für atomare Ereignisse ergeben sich durch Faktorisierung über die Produktregel:



$$
\begin{aligned}
P(j, m, a, \neg b, \neg e) &= P(j \mid a) \cdot P(m \mid a) \cdot P(a \mid \neg b, \neg e) \cdot P(\neg b) \cdot \mathrm{P}(\neg e) \\
&= 0.9 \cdot 0.7 \cdot 0.001 \cdot 0.999 \cdot 0.998 \\
&= 0.00062
\end{aligned}
$$

Kompaktheit Bayesscher Netze

- Zur expliziten Repräsentation der Verbundwahrscheinlichkeitsverteilung brauchen wir eine Tabelle der Größe 2^n bei n Booleschen Variablen.

- Falls in einem BN mit n Knoten jeder Knoten maximal k Eltern hat, brauchen wir nur n Tabellen der Größe 2^k bei booleschen Variablen.

Beispiel: $n = 20$ und $k = 5$

⇝ $2^n = 2^{20} = 1.048.576$ vs. $n \cdot 2^k = 20 \cdot 2^5 = 640$

→ Im ungünstigsten Fall kann natürlich auch ein BN exponentiell groß werden (wenn jede Variable von jeder anderen direkt beeinflusst wird).

→ Abhängig von der *Strukturiertheit* der Anwendungsdomäne (lokale vs. globale Interaktion) und dem Geschick des Designers.

Entwurf eines Bayesschen Netzes (1)

1. Wähle Menge von relevanten Variablen, welche die Domäne beschreiben.

2. Ordne alle Variablen.

3. Nimm erste Variable in der Liste.

4. Gib alle direkten Einflüsse von Knoten, die schon im Netz sind, auf den Knoten an: Kanten + CPT.

5. Streiche die Variable aus der Liste.

6. Solange Liste nicht leer: gehe zu Schritt 3.

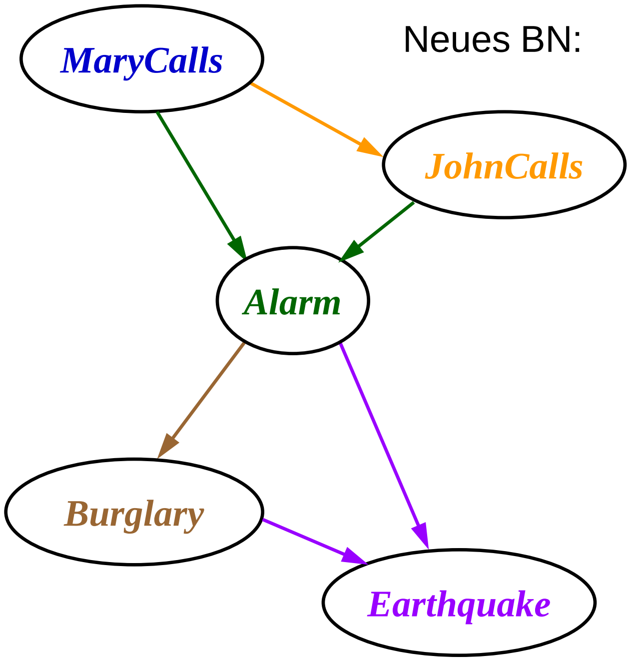
Frage: Welche Ordnung der Liste ist geeignet?

Beispiel (1)

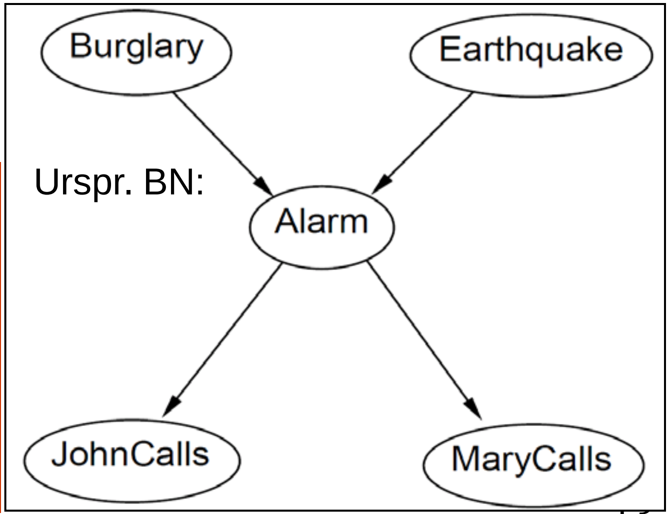
Ordnung: *MaryCalls, JohnCalls, Alarm, Burglary, Earthquake*

1) Wähle *MaryCalls*: ↝ Keine Elternknoten.
2) Wähle *JohnCalls*: Bei Evidenz *MaryCalls* ist *Alarm* und damit *JohnCalls* wahrscheinlicher.
   - ↝ **P**(*JohnCalls* | *MaryCalls*) ≠ **P**(*JohnCalls*)
   - ↝ *MaryCalls* wird Elternknoten von *JohnCalls*

- Wähle *Alarm*: Bei Evidenzen *MaryCalls* und *JohnCalls* ist *Alarm* wahrscheinlicher.
  - ↝ *MaryCalls* und *JohnCalls* werden Elternknoten von *Alarm*
- Wähle *Burglary*: Hierfür ist Evidenz *Alarm* alleine hinreichend.
  - ↝ *Alarm* wird Elternknoten von *Burglary*
- Wähle *Earthquake*: Bei alleiniger Evidenz *Alarm* ist *Earthquake* wahrscheinlich; bei gemeinsamer Evidenz *Alarm und Burglary* ist *Earthquake* weniger wahrscheinlich.
  - ↝ *Burglary* und *Alarm* werden Elternknoten von *Earthquake*

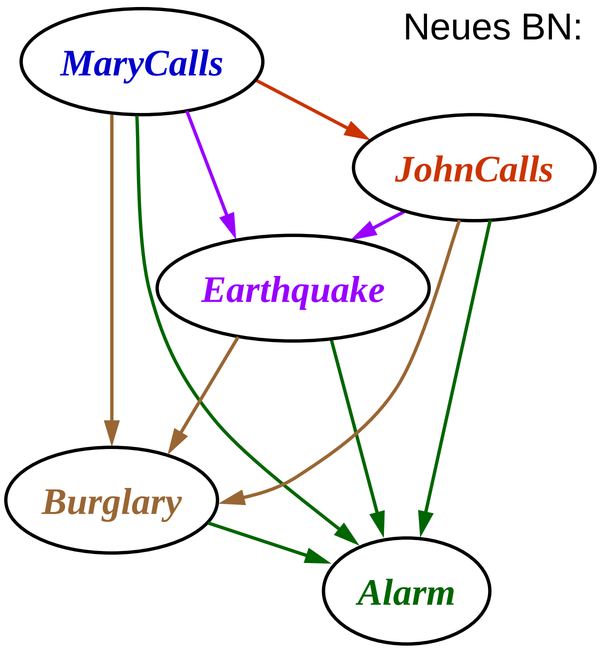


- ➢ Die Zahl der Abhängigkeiten (Kanten) ist gegenüber dem urspr. Netzentwurf von vier auf sechs um zwei gestiegen.
- ➢ Die Zahl der zu ermittelnden Wahrscheinlichkeiten ist um drei gestiegen.
- ➢ Gravierend ist die Qualität der neuen Abhängigkeiten: Wie soll z.B. **P**(*Earthquake* | *Alarm*, *Burglary*) erfasst werden?

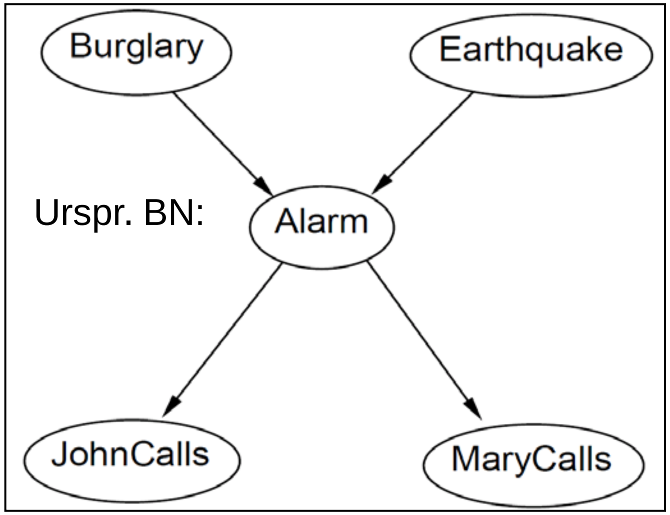


Beispiel (2)

- Ordnung: *MaryCalls, JohnCalls, Eathquake, Burglary, Alarm*



- Die Zahl der Abhängigkeiten (Kanten) hat sich gegenüber dem ursprünglichen Netzentwurf mehr als verdoppelt.
- Die Zahl der zu ermittelnden Wahrscheinlichkeiten ist auf 31 gestiegen und entspricht somit der vollständigen Verbundwahrscheinlichkeitsverteilung.
- Wiederum nur schwer zu erfassende neue Abhängigkeiten.



Entwurf eines Bayesschen Netzes (2)

- Der Aufbau eines „*diagnostischen Netzes*" führt zu Bedingungen von Symptomen zu Ursachen und damit zu neuen Abhängigkeiten zwischen ansonsten unabhängigen Ursachen – und oft auch zwischen unabhängigen Symptomen.

- Besser ist der Aufbau eines „kausalen Netzes":
  - starte mit den grundlegenden Ursachen (*root causes*)
  - erweitere schrittweise jeweils um die direkten Auswirkungen
  - bis zu den Blattknoten, die keine Auswirkungen auf andere Variable haben.

- Bemerkung: alle drei Netze des Beispiels repräsentieren dieselbe Verteilung der Verbundw'keiten, berücksichtigen jedoch in unterschiedlichem Maße die Unabhängigkeiten!

Bayessche Netze und Graphische Modelle

Bayessche Netze zählen zu den sogenannten *Graphischen Modellen*:

- Graphische Modelle werden als Kombination von Wahr‘keitstheorie und Graphentheorie betrachtet.
- Andere Formen von *Graphischen Modellen* sind Markov Random Fields, Conditional Random Fields, Faktorgraphen, u.a.

- Bisher: Betrachtung von BNs zur effizienten Kodierung der Wahr‘keiten.
- Jetzt: Betrachtung des operationellen Teils, nämlich die Inferenz in BNs.

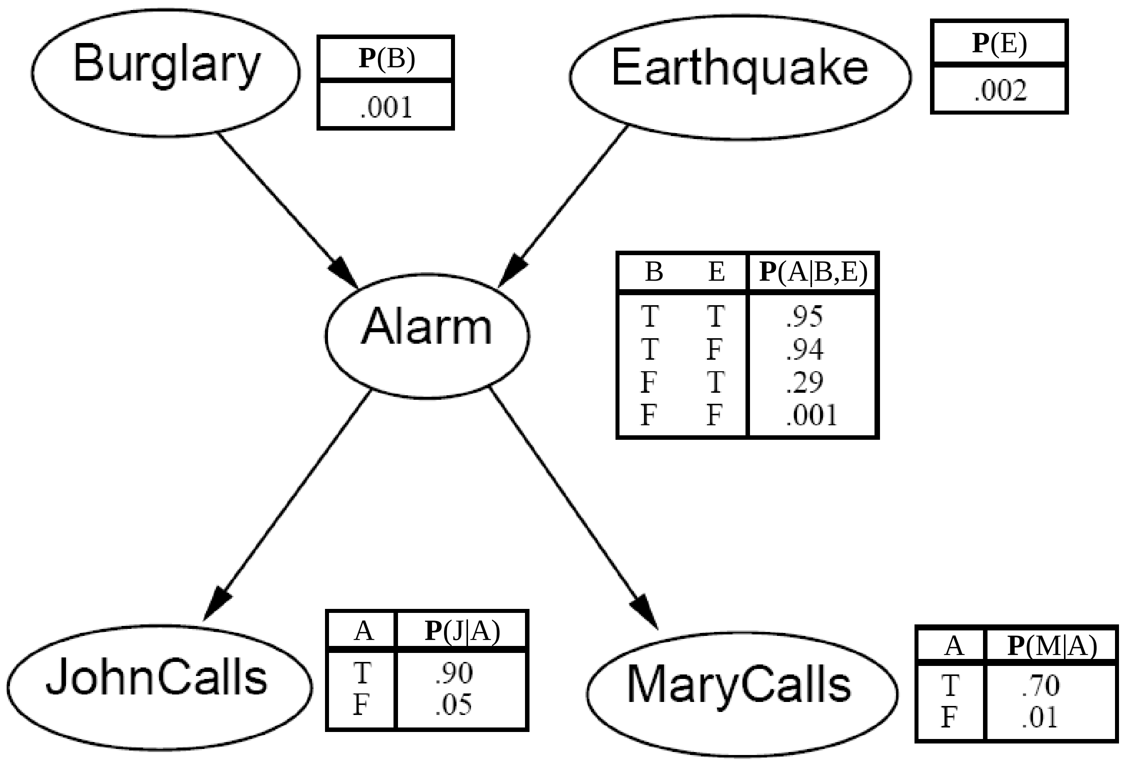
Inferenz in Bayesschen Netzen

Probabilistische Inferenz:

Gegeben: Instanziierte Evidenzvariablen

Gesucht: Wahrscheinlichkeitsverteilung von Anfragevariablen:

$$\mathbf{P}(\mathit{Query} \mid \mathit{Evidence})$$



Exakte Inferenz durch Aufzählen (1)

- Auswertung von Anfrage **P**(X|**e**) bei Belegung **e** der Evidenzvariablenmenge **E**?

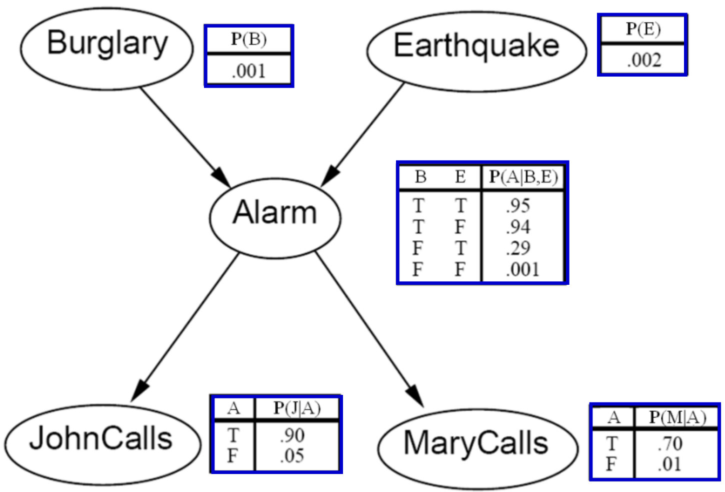
  ↝ Summation der Verbundw‘keiten über der Menge **Y** aller unbeobachteten Variablen:

$$\mathbf{P}(X \mid \mathbf{e}) = \alpha \cdot \mathbf{P}(X, \mathbf{e}) = \alpha \cdot \textstyle\sum_{\mathbf{y}} \mathbf{P}(X, \mathbf{e}, \mathbf{y}).$$

- Wegen Unabhängigkeitsannahmen der BNs ist die Anfrage durch die Summierung über Produkten von bedingten Wahr‘keiten im Netz zu beantworten:

$$\boxed{P(x_1, \ldots, x_n) = \textstyle\prod_{i=1}^{n} P(x_i \mid parents(X_i)).}$$

- Die bedingten W‘keiten sind in den CPTs des BNs notiert!

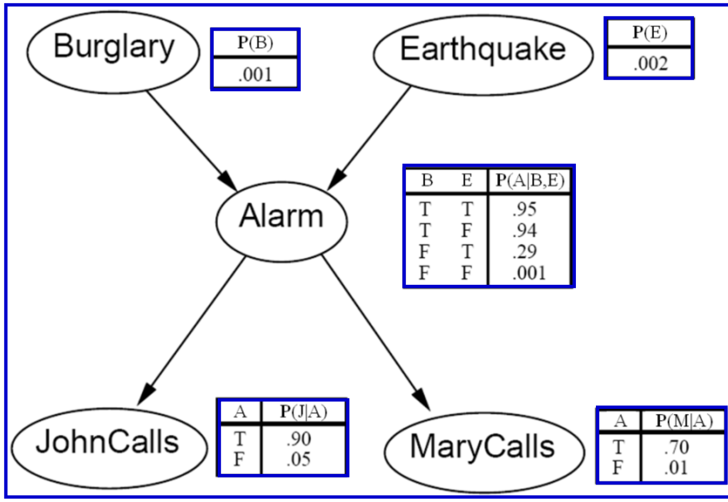


Exakte Inferenz durch Aufzählen (2)

- Eine systematische Methode, unbeobachtete Variablen („hidden variables“) aus der Verbundverteilung heraus zu marginalisieren.

- Einfache Anfrage im „Burglary“-Netzwerk sei:

  P(Burglary | JohnCalls = true, MaryCalls = true)?



- Abgekürzt: $\mathbf{P}(B|j,m) = \mathbf{P}(B,j,m) / \mathbf{P}(j,m)$ // per Def.

  $= \alpha \cdot \mathbf{P}(B,j,m)$

  $= \alpha \cdot \Sigma_e \, \Sigma_a \, \mathbf{P}(B,e,a,j,m)$

- Faktorisiere die Verbundverteilung als Produkt von CPT-Einträgen:

$$\mathbf{P}(B|j,m) = \alpha \cdot \Sigma_e \, \Sigma_a \, \mathbf{P}(B) \cdot \mathbf{P}(e) \cdot \mathbf{P}(a|B,e) \cdot \mathbf{P}(j|a) \cdot \mathbf{P}(m|a)$$

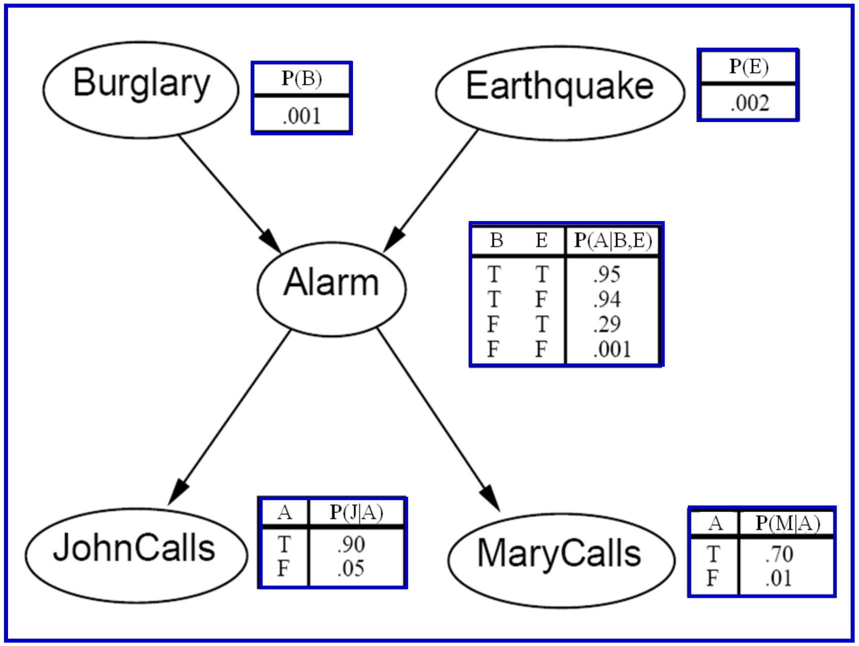
$$= \alpha \cdot \mathbf{P}(B) \cdot \Sigma_e \, \mathbf{P}(e) \cdot \Sigma_a \, \mathbf{P}(a|B,e) \cdot \mathbf{P}(j|a) \cdot \mathbf{P}(m|a)$$

- Rekursive *depth-first*-Aufzählung der Faktoren für *n* Boolesche Variable in $O(n)$ Platz- und $O(2^n)$ Zeitkomplexität.

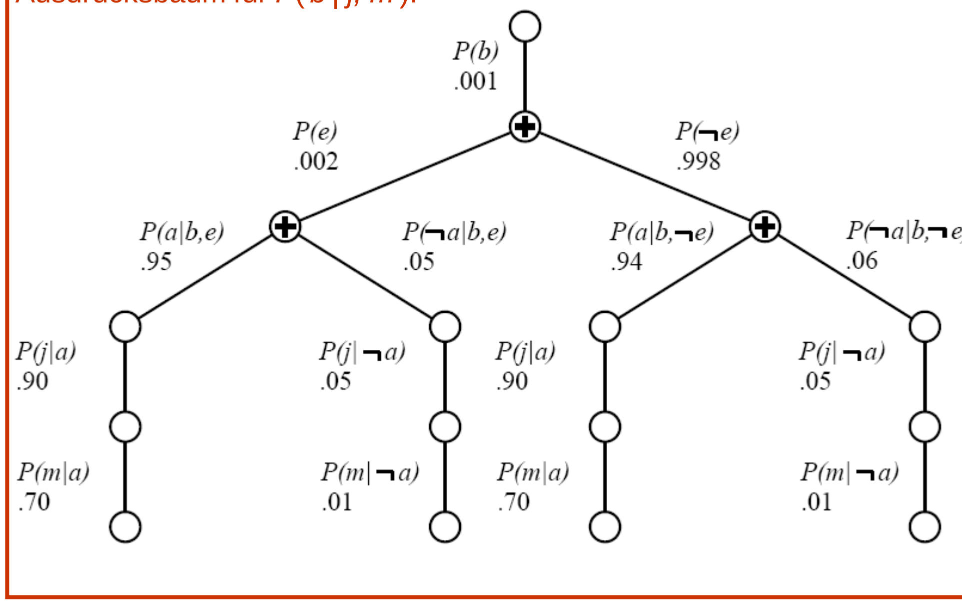
Exakte Inferenz durch Aufzählen (3)

Beispiel: **P**(Burglary | JohnCalls = true, MaryCalls = true)?

↝ $\mathbf{P}(B|j,m) = \alpha \cdot \mathbf{P}(B) \cdot \Sigma_e \mathbf{P}(e) \cdot \Sigma_a \mathbf{P}(a|B,e) \cdot \mathbf{P}(j|a) \cdot \mathbf{P}(m|a)$



Ausdrucksbaum für $P(b \mid j, m)$:



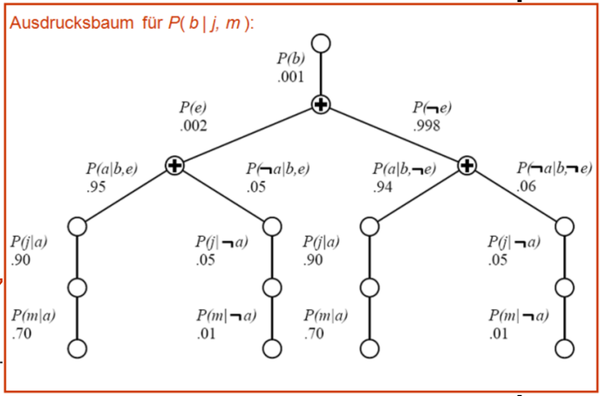
- Führt zu $P(b \mid j, m) = \alpha \cdot 0.00059224$ und <u>analog</u> $P(\neg b \mid j, m) = \alpha \cdot 0.0014919$.
- Normierung: $\mathbf{P}(B \mid j, m) = \alpha \cdot \langle 0.00059224, 0.0014919 \rangle = \langle 0.284, 0.716 \rangle$

↝ Bei beiden Anrufen besteht die Wahrscheinlichkeit von 28% für einen Einbruch!

Aufzählungsalgorithmus

function ENUMERATION-ASK(X, **e**, bn) **returns** a distribution over X
  inputs: X, the query variable
    e, observed values for variables $\mathbf{E}$
    bn, a Bayesian network with variables $\{X\} \cup \mathbf{E} \cup \mathbf{Y}$

  $\mathbf{Q}(X) \leftarrow$ a distribution over X, initially empty
  for each value x_i of X **do**
    extend **e** with value x_i for X
    $\mathbf{Q}(x_i) \leftarrow$ ENUMERATE-ALL(VARS[bn], **e**)
  return NORMALIZE($\mathbf{Q}(X)$)



---

function ENUMERATE-ALL($vars$, **e**) **returns** a real number
  if EMPTY?($vars$) **then return** 1.0
  $Y \leftarrow$ FIRST($vars$)
  if Y has value y in **e**
    then return $P(y \mid Pa(Y)) \times$ ENUMERATE-ALL(REST($vars$), **e**)
    else return $\Sigma_y\ P(y \mid Pa(Y)) \times$ ENUMERATE-ALL(REST($vars$), $\mathbf{e}_y$)
      where $\mathbf{e}_y$ is **e** extended with $Y = y$

Pa(Y) für Eltern(Y)

then für Evidenzvariable

else für unbeob. Variable

Redundante Berechnungen und Variablenelimination

Erhebliche Beschleunigung durch Vermeidung wiederholter oder unnötiger Berechnungen möglich:

1) Redundante Teilberechnungen sind nur einmal auszuführen und die entsprechenden Zwischenergebnisse zu speichern. Dazu sind die Ausdrücke von rechts nach links (bzw. im Baum von unten nach oben) auszuwerten.

2) Irrelevant heißt eine Variable V bzgl. einer Anfragevariablen X, wenn gilt: V ist weder Vorfahre der Anfragevariablen X noch Vorfahre einer Evidenzvariablen.

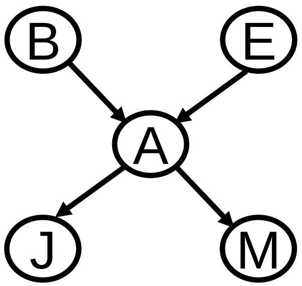
Beispiel: $\mathbf{P}(\mathit{JohnCalls} \mid \mathit{Burglary} = \mathit{true})$

$\mathbf{P}(J \mid b) = \alpha \cdot P(b) \cdot \sum_e P(e) \cdot \sum_a P(a \mid b, e) \cdot P(J \mid a) \cdot \sum_m P(m \mid a)$

$X = \mathit{JohnCalls}$, $\mathbf{E} = \{\mathit{Burglary}\}$

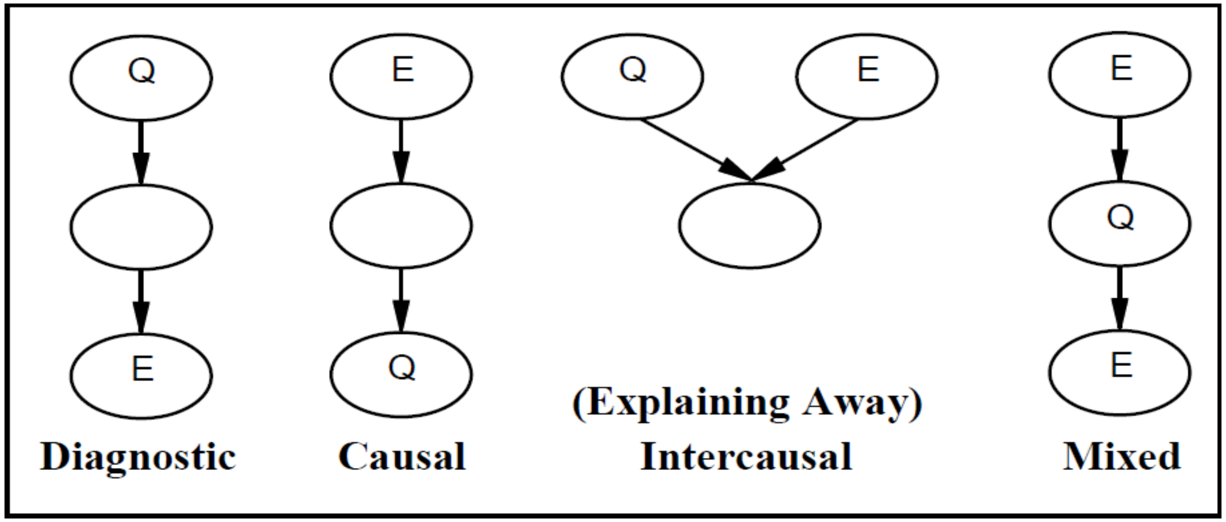
$\text{Ancestors}(\{X\} \cup \mathbf{E}) = \{\mathit{Alarm}, \mathit{Earthquake}, \mathit{Burglary}\}$

also ist *MaryCalls* irrelevant.



So sind alle Blattknoten im Ausdruckbaum, die für irrelevante Variablen stehen, eliminierbar. Nach deren Entfernung gibt es neue Blattknoten, die möglicherweise auch irrelevant sind, …

Typen von Inferenzen



(1) Diagnostisch: Von Effekten zu Ursachen

Bspl.: *P*(*burglary* | *johnCalls*) = 0.016

(2) Kausal: Von Ursachen zu Effekten

P(*johnCalls* | *burglary*) = 0.86

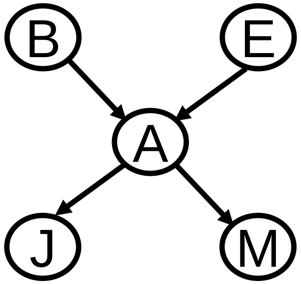
(3) Interkausal: Zwischen Ursachen eines gemeinsamen Effektes

P(*burglary* | *alarm*) = 0.376, aber

P(burglary | *alarm*, *earthquake*) = 0.003 (Explaining Away).

(4) Gemischt: Kombination von (1) – (3)

P(*alarm* | *johnCalls*, ¬*earthquake*) = 0.03



Komplexität der exakten Inferenz

Einfach verbundene Netzwerke (sog. „*Polytrees*"):

- Def.[Polytree]: Unter Vernachlässigung der Kantenrichtung sind jeweils zwei beliebige Knoten des Netzes durch *maximal* einen Pfad verbunden.
- Zeit- und Platzkomplexität *linear* in der Größe des Netzes:
  - mit Größe = Zahl der CPT-Einträge: $O(d^k \cdot n)$ mit n Knoten mit max. d Variablenwerten und max. k Elternknoten
  - Zeit- und Platzkomplexität linear in der Knotenzahl des Netzes
    - bei max. k Elternknoten

Mehrfach verbundene Netze:

- mindestens NP-hart (man kann z.B. 3-SAT leicht nachbilden)
- daher Betrachtung *approximativer* Inferenz

Approximative Inferenz

Idee: *Betrachte BN als Beschreibung eines Zufallsprozesses und führe eine stochast. Simulation des Prozesses durch, um gewünschte W'keiten zu schätzen.*

Stochastische Simulation:

1. Ziehe N Beispiele von einer Stichprobenverteilung S.
2. Berechne eine *approximative* Wahr'keit $\dot{P}$ auf der Grundlage der Verteilung der relativen Häufigkeiten.
3. Zeige, dass diese gegen die wahre gesuchte Wahr'keit P konvergiert.

Verfahren, die Wahr'keiten durch eine stochastische Simulation, nämlich das Ziehen von zufällig angeordneten *Stichproben* (*Beispielen, Samples*), abzuschätzen, werden auch als *Monte-Carlo-Methoden* bezeichnet.

Stichprobe von einem leeren Netzwerk (1)

Einfachste Sampling-Methode: Erzeugung von Ereignissen, *denen* *keine* *Evidenz zugeordnet ist*:

⇝ Sampling aller Variablen in ihrer topologischen Reihenfolge.

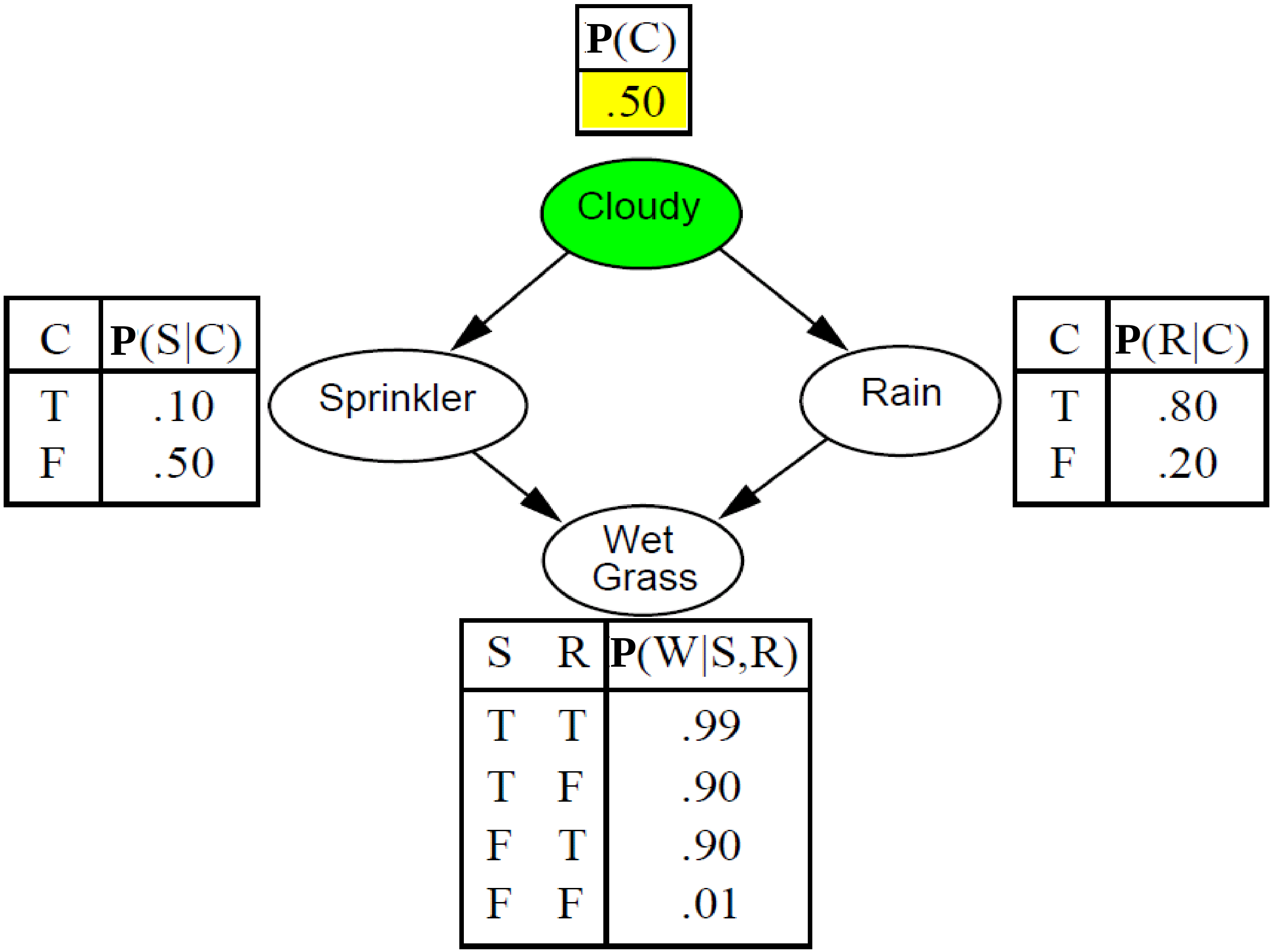
Keine
Evidenz-
variablen

function PRIOR-SAMPLE(bn) **returns** an event sampled from bn
  inputs: $\boxed{bn\text{, a belief network specifying joint distribution } \mathbf{P}(X_1, \ldots, X_n)}$

  $\mathbf{x} \leftarrow$ an event with n elements
  for $i = 1$ **to** n **do**
    $x_i \leftarrow$ a random sample from $\mathbf{P}(X_i \mid parents(X_i))$
      given the values of $parents(X_i)$ in $\mathbf{x}$
  return $\mathbf{x}$

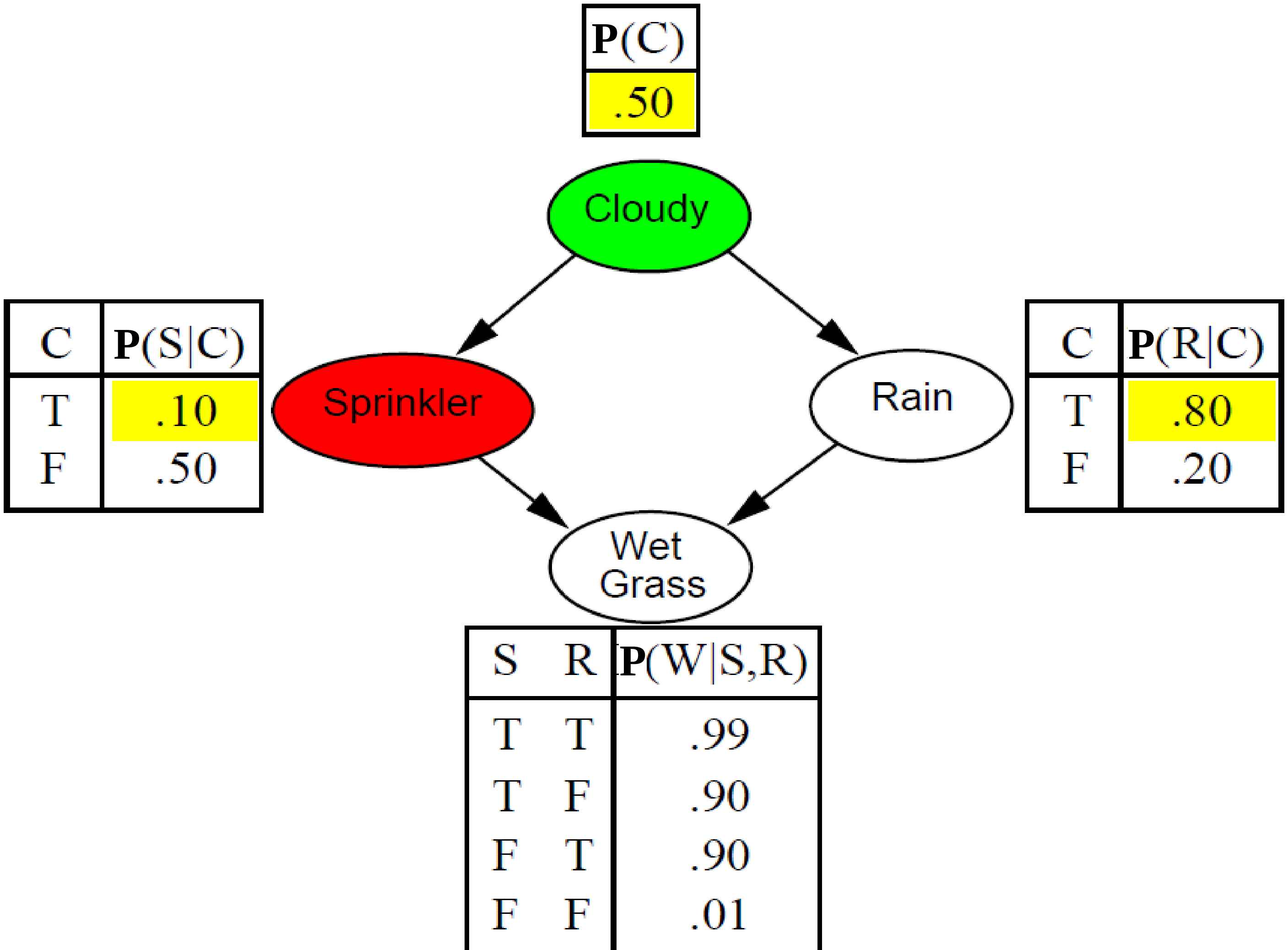
Beispiel zu Prior-Sample (1)

1) Stichprobe aus **P**(Cloudy) = ⟨0.5,0.5⟩ liefere *true* :



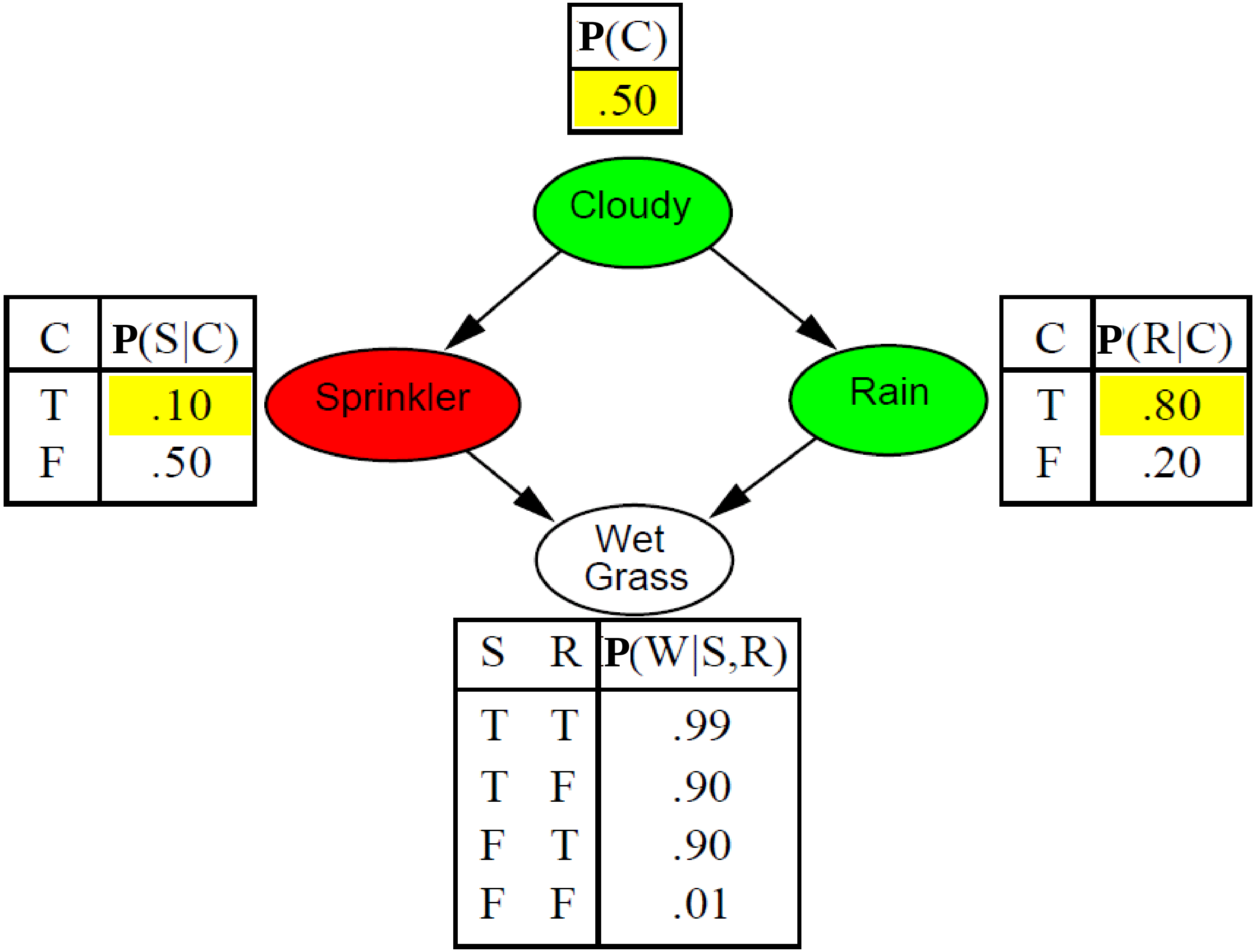
Beispiel zu Prior-Sample (2)

2) Stichprobe aus **P**(Sprinkler|cloudy) = ⟨0.1,0.9⟩ liefere *false* :



Beispiel zu Prior-Sample (3)

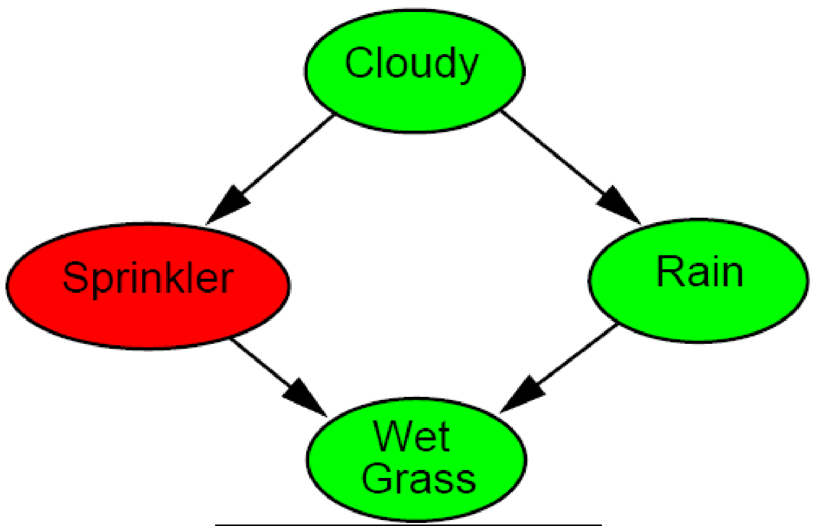
3) Stichprobe aus **P**(Rain|cloudy) = ⟨0.8,0.2⟩ liefere *true* :



Beispiel zu Prior-Sample (4)

4) Stichprobe aus **P**(WetGrass|¬sprinkler,rain) = ⟨0.9,0.1⟩ liefere *true* :

P(C)
.50



C	**P**(S|C)
T	.10
F	.50

C	**P**(R|C)
T	.80
F	.20

S	R	**P**(W|S,R)
T	T	.99
T	F	.90
F	T	.90
F	F	.01

Prior-Sample liefert

also hier das Ereignis:

[cloudy, ¬sprinkler,

rain, wetGrass]

bzw. [*t*, *f*, *t* ,*t*]

Stichprobe von einem leeren Netzwerk (2)

Die Wahrscheinlichkeit, dass *Prior-Sample* ein bestimmtes atomares Ereignis erzeugt ist

$$\boxed{S_{PS}(x_1,\ldots,x_n) =}\boxed{\prod_{i=1}^{n} P(x_i \mid parents(X_i))}\boxed{= P(x_1,\ldots,x_n),}$$

Sampling nach Bayes-Netz

Def. Verbundwk. nach Bayes-Netz

also die korrekte Verbundwahrscheinlichkeit des atomaren Ereignisses.

Sei *N* die Anzahl aller gezogenen Stichproben durch *Prior-Sample* und sei $N_{PS}(x_1, \ldots, x_n)$ die Anzahl der durch *Prior-Sample* gezogenen Samples des atomaren Ereignisses $x_1, \ldots, x_n$.

Wir erwarten also, dass die relative Häufigkeit $N_{PS}(x_1, \ldots, x_n)/N$ nach folg. Stichprobenwahrscheinlichkeit konvergiert:

$$\lim_{N\to\infty} \dot{P}(x_1, \ldots, x_n) = \lim_{N\to\infty} N_{PS}(x_1, \ldots, x_n)/N = S_{PS}(x_1, \ldots, x_n) = P(x_1, \ldots, x_n).$$

Eine Schätzung mit dieser Grenzwerteigenschaft heißt konsistent.

Schreibweise: $\dot{P}(x_1, \ldots, x_n) \approx P(x_1, \ldots, x_n)$

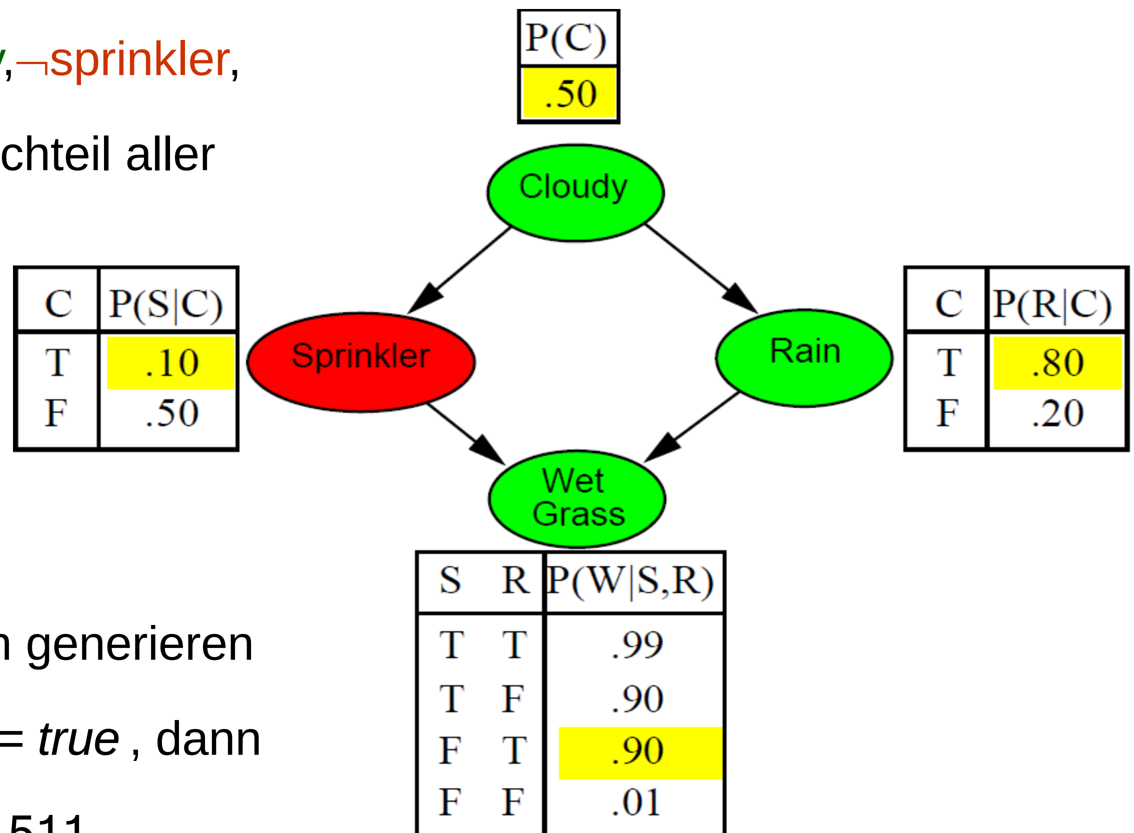
Stichprobe von einem leeren Netzwerk (3)

Am Beispiel: die W'keit für das Sampling von (cloudy,¬sprinkler,rain,wetGrass)

$$S_{PS}(cloudy, \neg sprinkler, rain, wetGrass) = 0.5 \cdot 0.9 \cdot 0.8 \cdot 0.9 = 0.324.$$

Im Grenzwert großer N erwarten wir, dass ca. 32.4% aller Stichproben diesem Ereignis entsprechen: $N_{PS}(cloudy, \neg sprinkler, rain, wetGrass)/N \approx 0.324$.

Die Wahrscheinlichkeit P(cloudy,¬sprinkler, rain,wetGrass) wird also als Bruchteil aller von dem Sampling-Prozess gezogenen atom. Ereignisse geschätzt, die mit diesem Ereignis übereinstimmen.



Wenn wir also 1000 Stichproben generieren und 511 davon ergeben $Rain = true$, dann schätzen wir $\hat{P}(Rain = \text{true}) = 0.511$.

Rejection Sampling

Berechne $\hat{P}(X \mid \mathbf{e})$ durch Samples, die zu **e** passen.

N = # samples

function REJECTION-SAMPLING(X, $\mathbf{e}$, bn, N) **returns** an estimate of $P(X|\mathbf{e})$
  local variables: $\mathbf{N}$, a vector of counts over X, initially zero

  for $j = 1$ to N **do**
    $\mathbf{x} \leftarrow$ PRIOR-SAMPLE(bn)
    if $\mathbf{x}$ is consistent with $\mathbf{e}$ **then**
      $\mathbf{N}[x] \leftarrow \mathbf{N}[x]+1$ where x is the value of X in $\mathbf{x}$
  return NORMALIZE($\mathbf{N}[X]$)

Beispiel: Schätze $\mathbf{P}(Rain \mid Sprinkler = true)$ unter Verwendung von 100 Samples

Ergebnis sei: 27 Stichproben mit *Sprinkler = true*, von diesen

– 8 mit *Rain = true*

– 19 mit *Rain = false*.

$\leadsto \hat{P}(Rain \mid Sprinkler = true) = Normalize(\langle 8,19 \rangle) = \langle 0.296, 0.704 \rangle$

Bewertung von Rejection Sampling

Rejection Sampling erzeugt also konsistente A-Posteriori-Schätzungen.

Problem: Hoffnungslos teuer bei hoher Ablehnungsrate von Stichproben.

Im Beispiel: 73 von 100 abgelehnten Stichproben.

Der Bruchteil der mit der Evidenz **e** inkonsistenten Stichproben

- ist hoch bei kleinem P(**e**) und
- wächst exponentiell mit steigender Zahl der Evidenzvariablen.

Besser ist die folgende Wahrscheinlichkeitsgewichtung bzw. Likelihood-Gewichtung, die nur Ereignisse erzeugt, die konsistent zur Evidenz **e** sind.

Der folgende Algorithmus wird mit **P**(*Rain* | *Sprinkler* = *true*, *WetGrass* = *true*) als Abfrage illustriert.

Likelihood: eine nicht normalisierte bedingte Wahrscheinlichkeit

Likelihood-Gewichtung (1)

Idee:

1) Halte die Werte von Evidenzvariablen **E** fest und sample nur Nichtevidenz-variablen X und **Y**.

2) Gewichte jedes Sample mit der Likelihood, die ihm nach der Evidenz zukommt.

- Damit ist jedes erzeugte Ereignis mit der Beobachtung **E** konsistent.
- Nicht alle Ereignisse sind gleich wichtig.
  - Solche Ereignisse, deren Evidenz wahrscheinlich erscheint (gemessen über dem Produkt der bedingt W-keiten jeder Evidenzvariablen bei bekannten Eltern), gehen mit höherem Gewicht ein.
  - Ereignisse, deren tatsächliche Evidenz unwahrscheinlich erscheint, erhalten weniger Gewicht.

Likelihood-Gewichtung (2)

function LIKELIHOOD-WEIGHTING(X, $\mathbf{e}$, bn, N) **returns** an estimate of $P(X|\mathbf{e})$
  local variables: $\mathbf{W}$, a vector of weighted counts over X, initially zero

  for $j = 1$ to N **do**
    $\mathbf{x}, w \leftarrow$ WEIGHTED-SAMPLE(bn, $\mathbf{e}$)
    $\mathbf{W}[x] \leftarrow \mathbf{W}[x] + w$ where x is the value of X in $\mathbf{x}$
  return NORMALIZE($\mathbf{W}[X]$)

Ereignis **x** mit Gewicht w

---

function WEIGHTED-SAMPLE(bn, $\mathbf{e}$) **returns** an event and a weight

  $\mathbf{x} \leftarrow$ an event with n elements; $w \leftarrow 1$
  for $i = 1$ **to** n **do**
    if X_i has a value x_i in $\mathbf{e}$
      then $w \leftarrow w \times P(X_i = x_i \mid parents(X_i))$
      else $x_i \leftarrow$ a random sample from $\mathbf{P}(X_i \mid parents(X_i))$
  return $\mathbf{x}$, w

Bewertung der Kompatibilität von Ereignis mit Evidenz

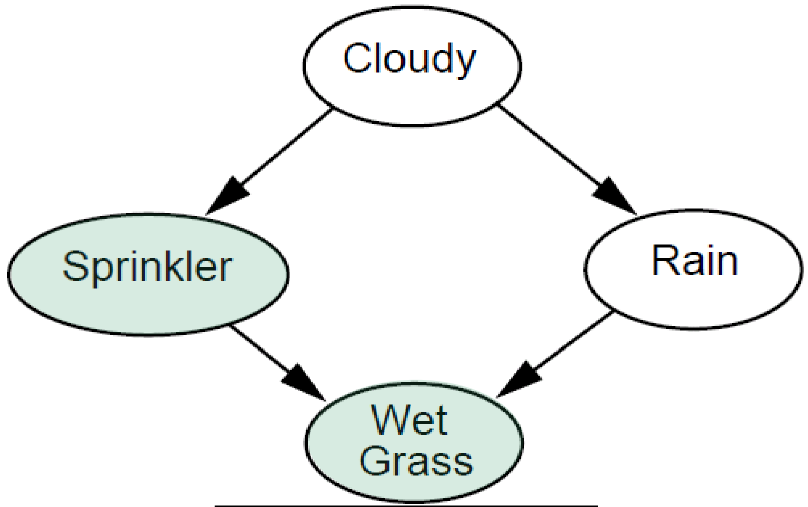
Sampling von Anfrage- und unbeobachteten Variablen

Beispiel: Likelihood-Gewichtung

Beispiel: Abfrage **P**(*Rain* | *Sprinkler = true, WetGrass = true*)

Sampling nur für Anfragevariable *Rain* und unbeobachtete Variable *Cloudy*

P(C)
.50



C	**P**(S|C)
T	.10
F	.50

C	**P**(R|C)
T	.80
F	.20

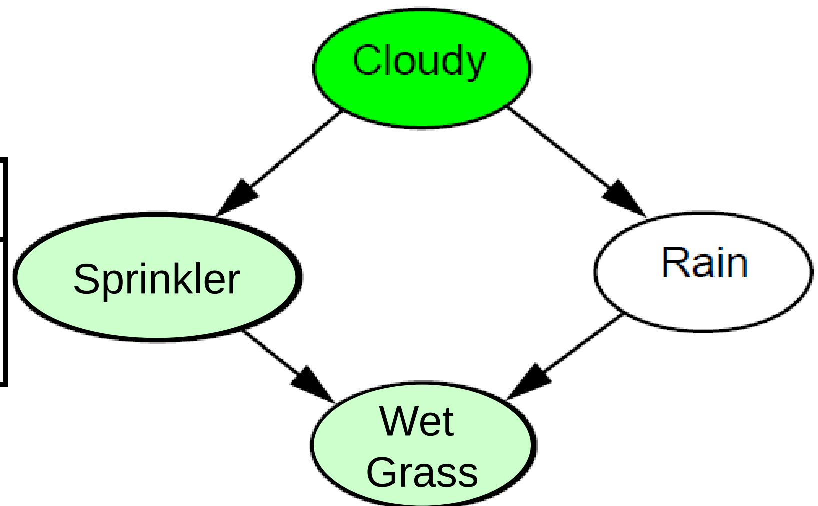
S	R	**P**(W|S,R)
T	T	.99
T	F	.90
F	T	.90
F	F	.01

Beispiel: Likelihood-Gewichtung (Forts. 1)

1) Stichprobe aus **P**(*Cloudy*) = ⟨0.5,0.5⟩ liefere *true* :

   w ← 1.0

P(C)
.50



C	**P**(S|C)
T	.10
F	.50

C	**P**(R|C)
T	.80
F	.20

S	R	**P**(W|S,R)
T	T	.99
T	F	.90
F	T	.90
F	F	.01

w = 1.0

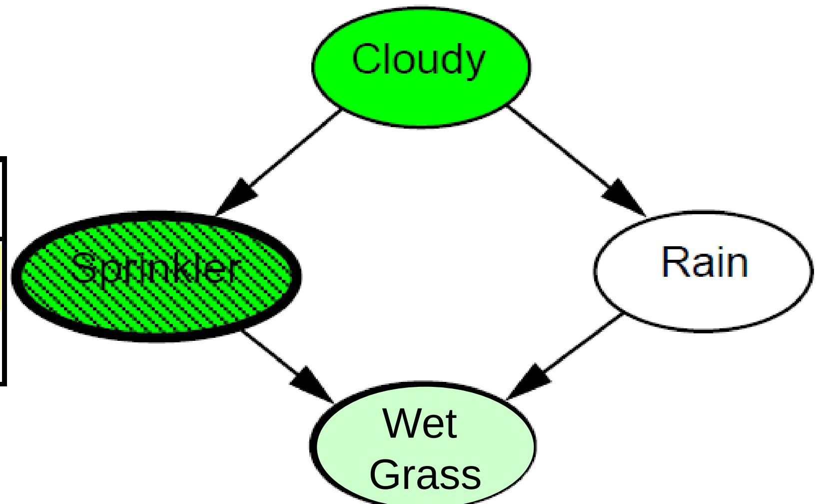
Beispiel: Likelihood-Gewichtung (Forts. 2)

2) *Sprinkler* ist Evidenzvariable mit Wert *true* (was bei *cloudy* unwahrscheinlich ist):

$w \leftarrow w \cdot P(\textit{sprinkler} \mid \textit{cloudy}) = 1.0 \cdot 0.1 = 0.1$

Gewichtsreduktion: Die Evidenzvariable *Sprinkler = true* zeigt hier, dass dieses Sample nicht gut passt

P(C)
.50



C	**P**(S|C)
T	.10
F	.50

C	**P**(R|C)
T	.80
F	.20

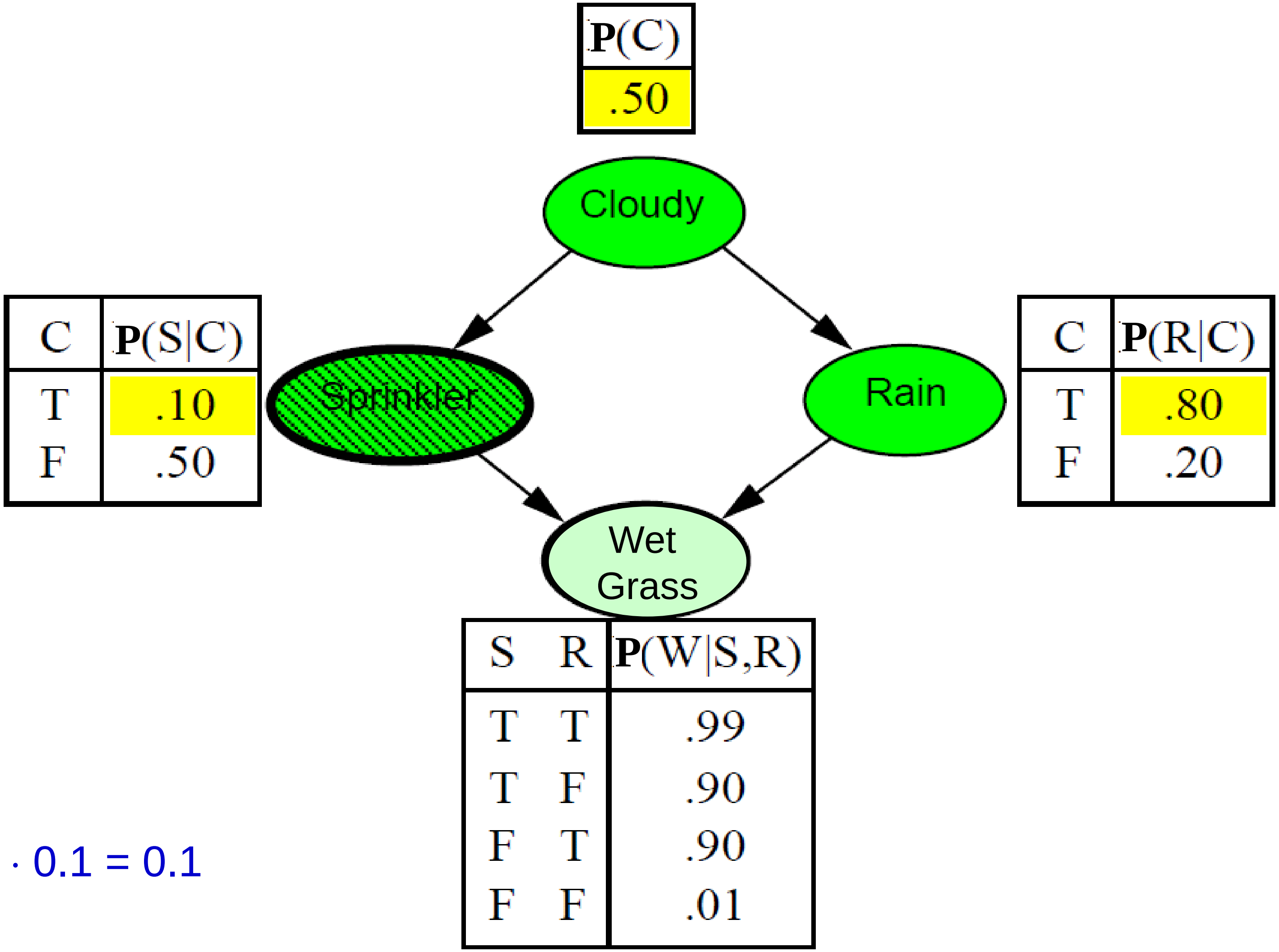
S	R	**P**(W|S,R)
T	T	.99
T	F	.90
F	T	.90
F	F	.01

$w = 1.0 \cdot 0.1 = 0.1$

Beispiel: Likelihood-Gewichtung (Forts. 3)

3) Stichprobe aus **P**(*Rain*) = ⟨0.8,0.2⟩ liefere *true* :

w = 0.1



w = 1.0 · 0.1 = 0.1

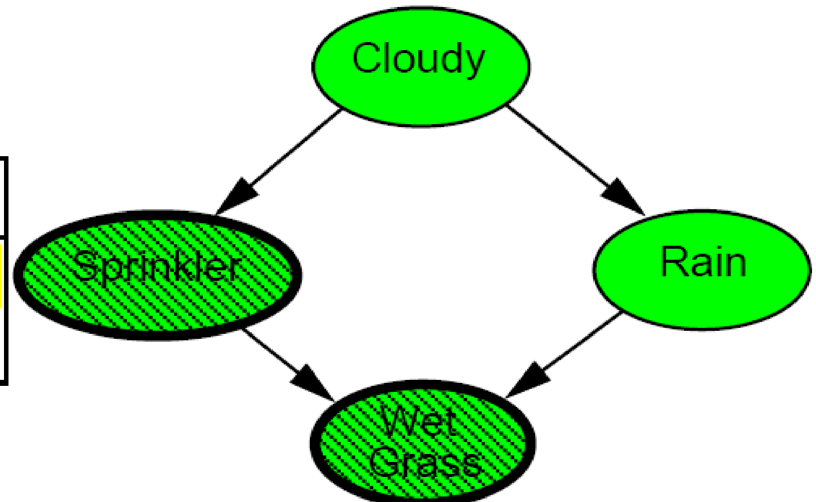
Beispiel: Likelihood-Gewichtung (Forts. 4)

4) *WetGrass* ist Evidenzvariable mit Wert *true* :

w ← w · P(wetgrass | sprinkler, rain) = 0.1 · 0.99 = 0.099

P(C)
.50

C	P(S|C)
T	.10
F	.50



C	P(R|C)
T	.80
F	.20

S	R	P(W|S,R)
T	T	.99
T	F	.90
F	T	.90
F	F	.01

Weighted-Sample liefert Ereignis (cloudy,sprinkler, rain,wetgrass) mit Gewicht w = 0.099, was als *Rain = true* gezählt wird.

Das Gewicht ist gering, da das Ereignis einen wolkige Tag zeigt, bei dem der Sprinkler wahrscheinlich nicht läuft.

Betrachtung der Likelihood-Gewichtung (1)

Seien Evidenzvariable **E** mit Werten **e** feststehend, Anfragevariable sei X und **Z** seien die unbeobachteten Variablen **Y** *und* Anfrage X, also: **Z** = {X} ∪**Y**.

Der Algorithmus *Weighted-Sample* sampelt jede Variable aus **Z** für bekannte Elternwerte:

$$S_{WS}(\boldsymbol{z},\boldsymbol{e}) = \prod_i P(z_i \mid parents(Z_i)).$$

Dabei kann *parents(*Z_i*)* sowohl verborgene Variable als auch Evidenzvariable enthalten. Anders als die unbedingte Verteilung **P**(**z**) berücksichtigt die Verteilung S_{WS} die Evidenz der Vorfahren von Z_i.

Das Gewicht für ein gegebenes Sample (**z**,**e**) ist das Produkt der Wahr‘keiten für jede Evidenzvariable mit bekannten Eltern:

$$w(\boldsymbol{z},\boldsymbol{e}) = \prod_j P(e_j \mid parents(E_j)).$$

Betrachtung der Likelihood-Gewichtung (2)

Das Gewicht für ein gegebenes Sample (**z**,**e**) ist das Produkt der Wahrscheinlichkeiten für jede Evidenzvariable mit bekannten Eltern:

$$w(\boldsymbol{z},\boldsymbol{e}) = \prod_j P(e_j \mid parents(E_j)).$$

Die gewichtete Sampling-Wahrscheinlichkeit ist dann

$$S_{WS}(\mathbf{z},\mathbf{e}) \cdot w(\mathbf{z},\mathbf{e}) = \Pi_i P(z_i \mid parents(Z_i)) \cdot \Pi_i P(e_i \mid parents(E_i)).$$

Beide Produkte decken also alle Variablen des Netzes ab.

Es lässt sich zeigen, dass $\acute{P}(x|\mathbf{e})$ für große N über die Sampling-Verteilung $\Sigma_{\mathbf{y}} S_{WS}(x,\mathbf{y},\mathbf{e}) \cdot w(x,\mathbf{y},\mathbf{e})$ gegen $P(x|\mathbf{e})$ konvergiert: $\acute{P}(x|\mathbf{e}) \approx P(x|\mathbf{e})$.

Die Likelihood-Gewichtung erzeugt also konsistente Schätzungen. Sie kann deutlich effizienter sein als Rejection Sampling.

Systeme

- Das bekannteste medizinische Expertensystem, das Bayessche Netze einsetzt, ist PATHFINDER IV.
  - Deckt ca. 60 Lymphknotenkrankheiten und 100 Symptome und Test-ergebnisse ab.
  - Es waren 14000 Schätzungen von Wahrscheinlichkeiten erforderlich, die in 40 Stunden Arbeit erstellt wurden.
  - Besser als Weltklasse-Experten.
- Weiteste Verbreitung: Diagnose- und Reparaturassistenten in MS Windows, z.B. Printer-Wizard aber (leider) auch Clippy, die Büroklammer.

→ Viele kommerzielle und PD-Tools für Bayessche Netze und Erweiterungen erhältlich, z.B. unter http://code.google.com/p/bnt/ sowie Tutorials, z. B. unter: http://www.cs.ubc.ca/~murphyk/Bayes/bnintro.html (beide Links: 20.05.2015).

Zusammenfassung Bayessche Netze

- Bayessche Netze erlauben eine kompakte Repräsentation der Verbund-w‘keit.
- Dies wird erreicht durch Unabhängigkeitsannahmen.
- Sie unterstützen verschiedene Formen des Schließens bei gegebenen Evidenzen: kausal, diagnostisch, interkausal, gemischt.
- Inferenz bedeutet dabei die Berechnung der Wahr‘keitsverteilung für die Belegungen einer Menge von Variablen bei gegebenen Evidenzen.
- Die Komplexität der Inferenz in Bayesschen Netzen hängt von der Struktur des Netzwerkes ab.
- Für Polybäume ist die Komplexität polynomiell in der Größe des Netzwerks.
- Im Allgemeinen ist die Inferenz in Bayesschen Netzen NP-hart.
- Es gibt Approximationstechniken, um diesem Problem zu begegnen.