Bayesian Networks

IV. Approximate Inference (sections 1–3)

Prof. Dr. Lars Schmidt-Thieme, L. B. Marinho, K. Buza

Information Systems and Machine Learning Lab(ISMLL)
University of Hildesheim

1. **Why exact inference may not be good enough**

2. **Acceptance-Rejection Sampling**

3. **Importance Sampling**

4. **Self and Adaptive Importance Sampling**

5. **Stochastic / Loopy Propagation**

bayesian network	# variables	time for exact inference
studfarm	12	0.18s
Hailfinder	56	0.36s
Pathfinder-23	135	4.04s
Link	742	307.72s[1)]

on a 1.6MHz Pentium-M notebook
([1)] on a 2.5 MHz Pentium-IV)

though

- w/o optimized implementation
- with very simple triangulation heuristics (minimal degree).

1. Why exact inference may not be good enough
2. **Acceptance-Rejection Sampling**
3. Importance Sampling
4. Self and Adaptive Importance Sampling
5. Stochastic / Loopy Propagation

Estimating marginals from data

case	family-out	light-on	bowel-problem	dog-out	hear-bark
1	0	0	0	0	0
2	0	0	0	0	0
3	1	1	1	1	0
4	0	0	1	1	1
5	0	0	0	0	0
6	0	0	0	0	0
7	0	0	0	1	1
8	0	0	0	0	0
9	0	0	1	1	1
10	1	1	0	1	1

Figure 1: Example data for the dog-problem.

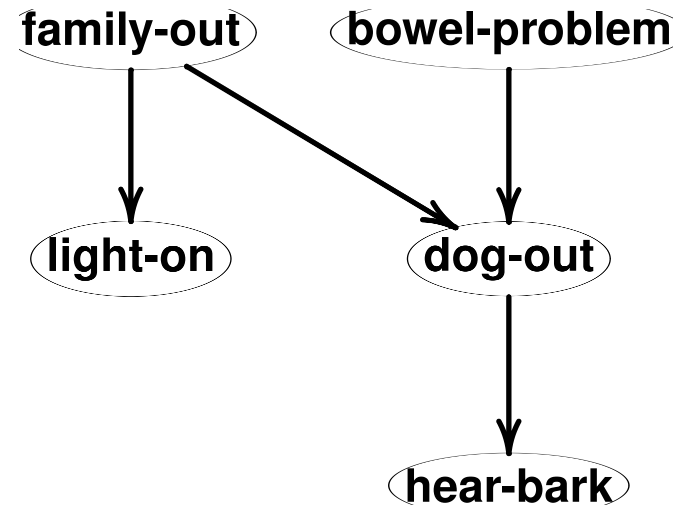


Figure 2: Bayesian network for dog-problem.

family-out	0	8
	1	2

a) counts

family-out	0	0.8
	1	0.2

b) probabilities

Figure 3: Estimating absolute probabilities (root node tables).

Estimating marginals from data

case	family-out	light-on	bowel-problem	dog-out	hear-bark
1	0	0	0	0	0
2	0	0	0	0	0
3	1	1	1	1	0
4	0	0	1	1	1
5	0	0	0	0	0
6	0	0	0	0	0
7	0	0	0	1	1
8	0	0	0	0	0
9	0	0	1	1	1
10	1	1	0	1	1

Figure 1: Example data for the dog-problem.

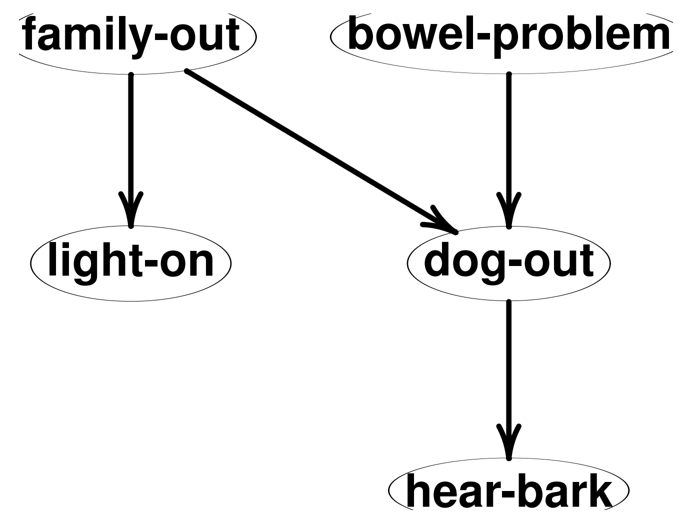


Figure 2: Bayesian network for dog-problem.

family-out	0		1	
bowel	0	1	0	1
dog-out 0	5	0	0	0
1	1	2	1	1

a) counts

family-out	0		1	
bowel	0	1	0	1
dog-out 0	0.5	0	0	0
1	0.1	0.2	0.1	0.1

b) absolute probabilities

family-out	0		1	
bowel	0	1	0	1
dog-out 0	$\frac{5}{6}$	0	0	0
0	$\frac{1}{6}$	1	1	1

c) cond. probabilities

Figure 4: Estimating conditional probabilities (inner node tables).

Estimating marginals from data given evidence

If we want to estimate the probabilities for **family-out** given the evidence that **dog-out** is 1, we have

(i) identify all cases that are **compatible with the given evidence,**

(ii) estimate the target potential $p($familiy-out$)$ from these cases.

case	family-out	light-on	bowel-problem	dog-out	hear-bark	
1	0	0	0	0	0	rejected
2	0	0	0	0	0	rejected
3	1	1	1	1	0	accepted
4	0	0	1	1	1	accepted
5	0	0	0	0	0	rejected
6	0	0	0	0	0	rejected
7	0	0	0	1	1	accepted
8	0	0	0	0	0	rejected
9	0	0	1	1	1	accepted
10	1	1	0	1	1	accepted

Figure 5: Accepted and rejected cases for evidence dog-out = 1.

family-out 0	3
1	2

a) counts

family-out 0	0.6
1	0.4

b) probabilities

Figure 6: Estimating target potentials given evidence, here $p($family-out$|$dog-out $= 1)$.

Learning and inferencing

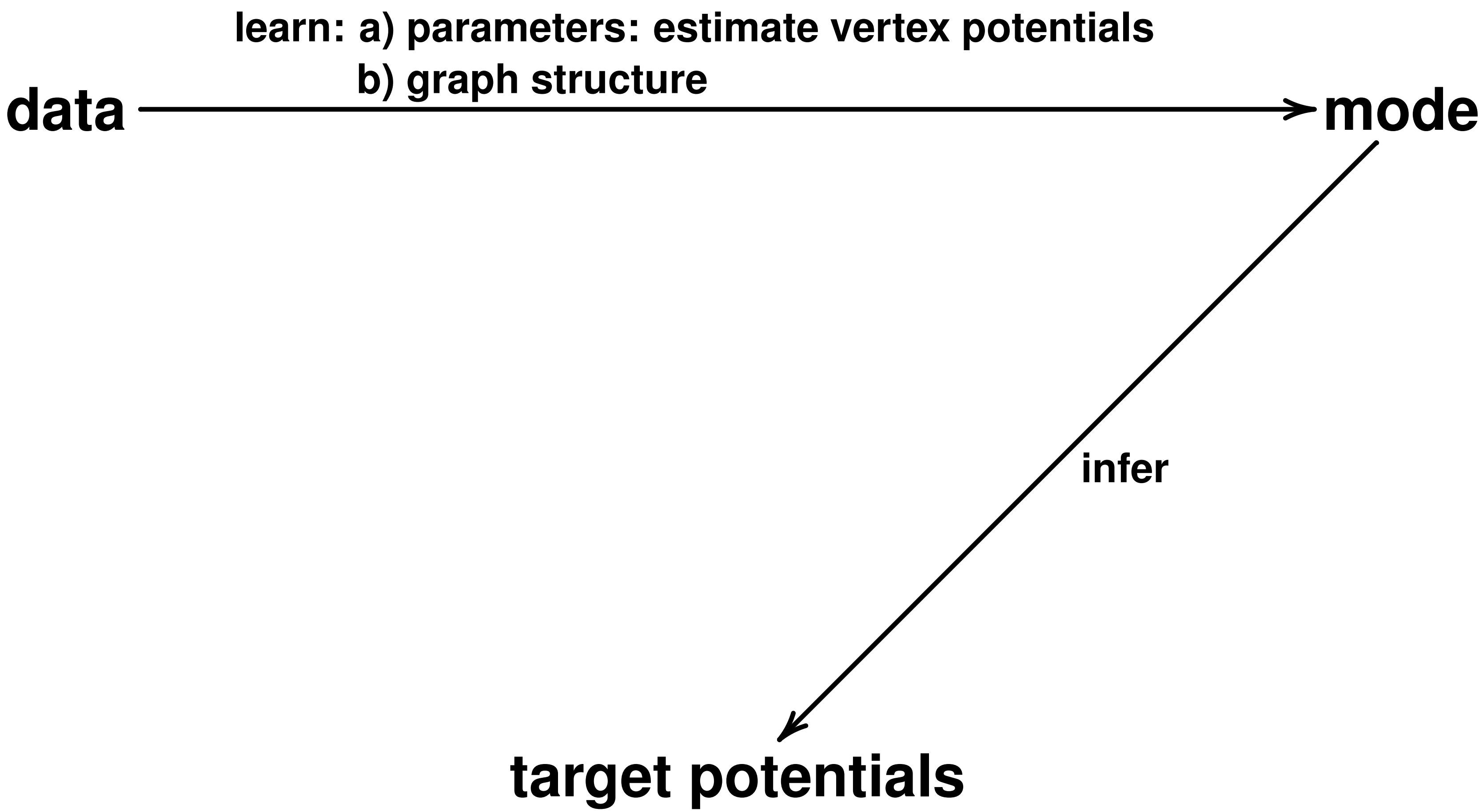


Figure 7: Learing models from data for inferencing.

Sampling and estimating

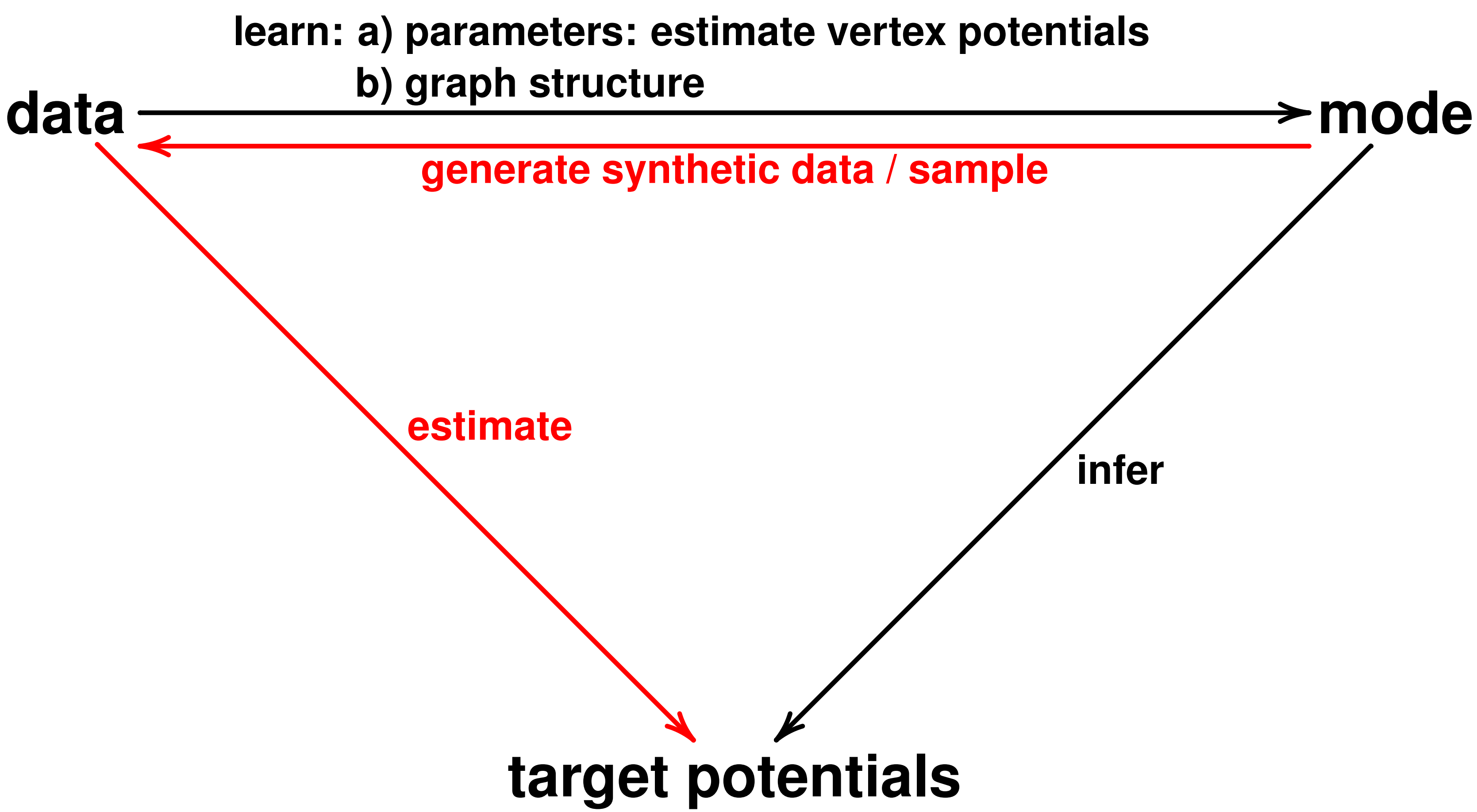


Figure 7: Learing models from data for inferencing vs. sampling from models and estimating.

Sampling a discrete distribution

Given a discrete distribution, e.g.,

Pain	Y				N			
Weightloss	Y		N		Y		N	
Vomiting	Y	N	Y	N	Y	N	Y	N
Adeno Y	.169	.210	.048	.049	.119	.112	.009	.005
N	.003	.009	.010	.024	.039	.090	.044	.062

Figure 8: Example for a discrete distribution.

How do we draw samples from this distribution?
= generate synthetic data that is distributed according to it?

Sampling a discrete distribution

(i) Fix an enumeration of all states of the distribution p, i.e.,

$$\sigma : \{1, \ldots, |\Omega|\} \to \Omega \text{ bijective}$$

with Ω the set of all states,

(ii) compute the cumulative distribution function in the state index, i.e.,

$$\begin{aligned} \text{cum}_{p,\sigma} : \{1, \ldots, |\Omega|\} &\to [0, 1] \\ i &\mapsto \sum_{j \leq i} p(\sigma(j)) \end{aligned}$$

(iii) draw a random real value r uniformly from [0,1],

(iv) search the state ω with

$$\text{cum}_{p,\sigma}(\omega) \leq r$$

and maximal $\text{cum}_{p,\sigma}(\omega)$.

Pain	Y				N			
Weightloss	Y		N		Y		N	
Vomiting	Y	N	Y	N	Y	N	Y	N
Adeno Y	.169	.210	.048	.049	.119	.112	.009	.005
N	.003	.009	.010	.024	.039	.090	.044	.060

Figure 8: Example for a discrete distribution.

Adeno	Y								N							
Pain	Y				N				Y				N			
Weightloss	Y		N		Y		N		Y		N		Y		N	
Vomiting	Y	N	Y	N	Y	N	Y	N	Y	N	Y	N	Y	N	Y	N
$\text{cum}_{p,\sigma}(i)$	.169	.379	.427	.476	.595	.707	.716	.721	.724	.733	.743	.767	.806	.896	.940	1.000
index i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

Figure 9: Cumulative distribution function.

Sampling a Bayesian Network / naive approach

As a bayesian network encodes a discrete distribution, we can use the method from the former slide to draw samples from a bayesian network:

(i) Compute the full JPD table from the bayesian network,

(ii) draw a sample from the table as on the slide before.

This approach is not sensible though, as we actually used bayesian networks s.t. we **not** have to compute the full JPD (as it normally is way to large to handle).

How can we make use of the independencies encdoded in the bayesian network structure?

Sampling a Bayesian Network

Idea: sample variables separately, one at a time.

If we have sampled

$$X_1, \ldots, X_k$$

already and X_{k+1} is a vertex s.t.

$$\mathrm{desc}(X_{k+1}) \cap \{X_1, \ldots, X_k\} = \emptyset$$

then

$$p(X_{k+1}|X_1, \ldots, X_k) = p(X_{k+1}|\,\mathrm{pa}(X_{k+1}))$$

i.e., we can sample X_{k+1} from its vertex potential given the evidence of its parents (as sampled before).

$\text{sample-forward}(B := (G := (V, E), (p_v)_{v \in V})) :$
$\sigma :=$ *topological-ordering*(G)
$x := 0_V$
for $i = 1, \ldots, |\sigma|$ **do**
  $v := \sigma(i)$
  $q := p_v|_{x|_{\mathrm{pa}(v)}}$
  draw $x_v \sim q$
od
return x

Figure 10: Algorithm for sampling a bayesian network.

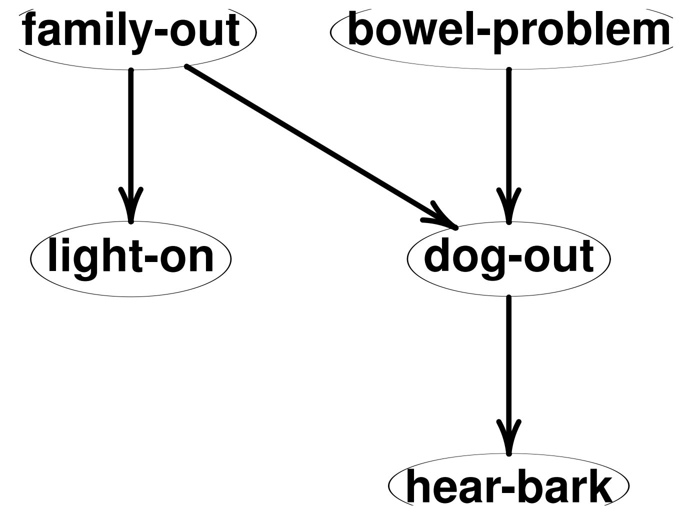
Sampling a Bayesian Network / example

Let $\sigma := (F, B, L, D, H)$.

1. $x_F \sim p_F = (0.85, 0.15)$
   say with outcome 0.
2. $x_B \sim p_B = (0.8, 0.2)$
   say with outcome 1.
3. $x_L \sim p_L(F = 0) = (0.95, 0.05)$
   say with outcome 0.
4. $x_D \sim p_D(F = 0, B = 1) = (0.03, 0.97)$
   say with outcome 1.
5. $x_H \sim p_H(D = 1) = (0.3, 0.7)$
   say with outcome 1.

The result is

$$x = (0, 1, 0, 1, 1)$$



F = 0	0.85
1	0.15

B = 0	0.8
1	0.2

F	0	1
L = 0	0.95	0.4
1	0.05	0.6

D	0	1
H = 0	0.99	0.3
1	0.01	0.7

B	0		1	
F	0	1	0	1
D = 0	0.7	0.1	0.03	0.01
1	0.3	0.9	0.97	0.99

Figure 11: Bayesian network for dog-problem.

Acceptance-rejection sampling

Inferencing by **acceptance-rejection sampling** means:

(i) draw a sample from the bayesian network (w/o evidence entered),

(ii) drop all data from the sample that are not conformant with the evidence,

(iii) estimate target potentials from the remaining data.

For bayesian networks sampling is done by forward-sampling. — Forward sampling is stopped as soon as an evidence variable has been instantiated that contradicts the evidence.

Acceptance-rejection sampling for bayesian networks is also called **logic sampling** [Hen88].

infer-acceptance-rejection(B : *bayesian network*,
        W : *target domain*, E : *evidence*, n : *sample size*) :
$D := ($*sample-forward*$(B) \mid i = 1, \ldots, n)$
return *estimate*(D, W, E)

sample-forward$(B := (G := (V, E), (p_v)_{v \in V}))$:
$\sigma :=$ *topological-ordering*(G)
$x := 0_V$
for $i = 1, \ldots, |\sigma|$ **do**
    $v := \sigma(i)$
    $q := p_v|_{x|_{\mathrm{pa}(v)}}$
    draw $x_v \sim q$
od
return x

estimate(D : *data*, W : *target domain*, E : *evidence*) :
$D' := (d \in D \mid d|_{\mathrm{dom}(E)} = \mathrm{val}(E))$
return *estimate*(D', W)

estimate(D : *data*, W : *target domain*) :
$q :=$ *zero-potential on* W
for $d \in D$ **do**
    $q(d)$++
od
$q := q/|\mathit{data}|$
return q

Figure 12: Algorithm for acceptance-rejection sampling.

1. Why exact inference may not be good enough

2. Acceptance-Rejection Sampling

3. Importance Sampling

4. Self and Adaptive Importance Sampling

5. Stochastic / Loopy Propagation

Acceptance rate of acceptance-rejection sampling

How efficient acceptance-rejection sampling is depends on the **acceptance rate**.

Let E be evidence. Then the acceptance rate, i.e., the fraction of samples conformant with E, is

$$p(E)$$

the marginal probability of the evidence.

Thus, acceptance-rejection sampling performs poorly if the probability of evidence is small. In the studfarm example

$$p(J = aa) = 0.00043$$

i.e., from 2326 sampled cases 2325 are rejected.

Idea of importance sampling

Idea: do not sample the evidence variables, but instantiate them to the values of the evidence.
Instantiating the evidence variables first, means, we have to sample the other variables from

$$p(X_{k+1}| \ X_1 = x_1, \ldots, X_k = x_k, \\ E_1 = e_1, \ldots, E_m = e_m)$$

even for a topological ordering of non-evidential variables.
Problem: if there is an evidence variable that is a descendant of a non-evidential variable X_{k+1} that has to be sampled, then

- it does neither occur among its parents nor is independent from X_{k+1}, and
- it may open dependency chains to other variables !

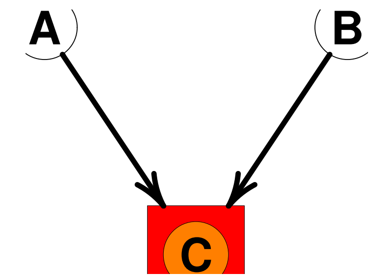


Figure 13: If C is evidential and already instantiated, say $C = c$, then A is dependent on C, so we would have to sample A from $p(A|C = c)$. Even worse, B is dependant on C and A (d-separation), so we would have to sample B from $p(B|A = a, C = c)$. But neither of these cpdfs is known in advance.

Inference from a stochastic point of view

Let V be a set of variables and p a pdf on $\prod \mathrm{dom}(V)$. Infering the marginal on a given set of variables $W \subseteq V$ and given evidence E means to compute

$$(p_E)^{\downarrow W}$$

i.e., for all $x \in \prod \mathrm{dom}(W)$

$$(p_E)^{\downarrow W}(x) = \sum_{y \in \prod \mathrm{dom}(V \setminus W \setminus \mathrm{dom}(E))} p(x, y, e)$$
$$= \sum_{y \in \prod \mathrm{dom}(V)} I_{x,e}(y) \cdot p(y)$$

with the indicator function

$$I_x : \prod \mathrm{dom}(V) \rightarrow \{0, 1\}$$
$$y \mapsto \begin{cases} 1, & \text{if } y|_{\mathrm{dom}(x)} = x \\ 0, & \text{else} \end{cases}$$

So we can reformulate the inference problem as the problem of **averaging a given random variable** f **(here:** $f := I_{x,e}$**) over a given pdf** p, i.e., to compute / estimate the mean

$$\mathbb{E}_p(f) := \sum_{x \in \mathrm{dom}(p)} f(x) \cdot p(x)$$

Theorem 1 (strong law of large numbers). *Let* $p : \Omega \rightarrow [0, 1]$ *be a pdf,* $f : \Omega \rightarrow \mathbb{R}$ *be a random variable with* $\mathbb{E}_p(|f|) < \infty$, *and* $X_i \sim f, i \in \mathbb{N}$ *independently. Then*

$$\frac{1}{n} \sum_{i=1}^{n} X_i \rightarrow_{a.s.} \mathbb{E}_p(f)$$

Proof. See, e.g., [Sha03, p. 62] □

Sampling from the wrong distribution

Inference by sampling applies the SLLN:

$$\sum_{x \in \mathrm{dom}(p)} f(x) \cdot p(x) =: \mathbb{E}_p(f)$$

$$\approx \frac{1}{n} \sum_{\substack{x \sim p \\ (n \text{ draws})}} f(x)$$

Now let q be any other pdf with

$$p(x) > 0 \Longrightarrow q(x) > 0, \quad \forall x \in \mathrm{dom}(p) = \mathrm{dom}(q)$$

Due to

$$\sum_{x \in \mathrm{dom}(p)} f(x) \cdot p(x) = \sum_{x \in \mathrm{dom}(p)} f(x) \cdot \frac{p(x)}{q(x)} \cdot q(x)$$

$$=: \mathbb{E}_q(f \cdot \frac{p}{q})$$

$$\approx \frac{1}{n} \sum_{\substack{x \sim q \\ (n \text{ draws})}} f(x) \cdot \frac{p(x)}{q(x)}$$

we can sample from q instead from p if we adjust the function values of f accordingly.

The pdf q is called **importance function**, the function $w := p/q$ is called **score** or **case weight**.

Often we know the case weight only up to a multiplicative constant, i.e., $w' := c \cdot w \propto p/q$ with unknown constant c.
For a sample $x_1, \ldots, x_n \sim q$, we then can approximate $\mathbb{E}_p(f)$ by

$$\mathbb{E}_p(f) \approx \frac{1}{n} \sum_{i=1}^{n} f(x_i) \cdot w(x_i)$$

$$\approx \frac{1}{\sum_i w'(x_i)} \sum_{i=1}^{n} f(x_i) \cdot w'(x_i)$$

Case weight

Back to

sampling from the true distribution p_E
vs.
sampling from the bayesian network with pre-instantiated evidence variables (the wrong distribution)

The probability for a sample x from a Bayesian network among samples conformant with a given evidence E is

$$p(x|E) = \frac{p(x)}{p(E)} = \frac{\prod_{v \in V} p_v(x_v \mid x|_{\mathrm{pa}(v)})}{p(E)}$$

The probability for a sample x from a Bayesian network with pre-instantiated evidence variables is

$$q_E(x) = \prod_{v \in V \setminus \mathrm{dom}(E)} p_v(x_v \mid x|_{\mathrm{pa}(v)})$$

Thus, the case weight is

$$w(x) := \frac{p(x|E)}{q_E(x)} = \frac{\prod_{v \in \mathrm{dom}(E)} p_v(x_v \mid x|_{\mathrm{pa}(v)})}{p(E)}$$

Likelihood weighting sampling

Inferencing by **importance sampling** means:

(i) choose a sampling distribution q,

(ii) draw a weighted sample from q,

(iii) estimate target potentials from these sample data.

For bayesian networks using *sampling from bayesian networks with pre-instantiated evidence variables* and the case weight

$$w(x) := \prod_{v \in \mathrm{dom}(E)} p_v(x_v \,|\, x|_{\mathrm{pa}(v)})$$

is called **likelihood weighting sampling** [FC90, SP90]

```
infer-likelihood-weighting(B : bayesian network,
                W : target domain, E : evidence, n : sample size) :
(D, w) := (sample-likelihood-weighting(B, E) | i = 1, ..., n)
return estimate(D, w, W)
```

```
sample-likelihood-weighting(B := (G, (p_v)_{v∈V_G}), E : evidence) :
σ := topological-ordering(G \ dom(E))
x := 0_{V_G}
x|dom(E) := val(E)
for i = 1, ..., |σ| do
    v := σ(i)
    q := p_v|x|pa(v)
    draw x_v ~ q
od
w(x) := ∏_{v∈dom(E)} p_v(x_v | x|pa(v))
return (x, w(x))
```

```
estimate(D : data, w : case weight, W : target domain) :
q := zero-potential on W
w_tot := 0
for d ∈ D do
    q(d) := q(d) + w(d)
    w_tot := w_tot + w(d)
od
q := q/w_tot
return q
```

Figure 14: Algorithm for inference by likelihood weighting sampling.

Likelihood weighting sampling / example

Let the evidence be $D = 1$. Fix $\sigma := (F, B, L, H)$.

1. $x_F \sim p_F = (0.85, 0.15)$
   say with outcome 0.

2. $x_B \sim p_B = (0.8, 0.2)$
   say with outcome 1.

3. $x_L \sim p_L(F = 0) = (0.95, 0.05)$
   say with outcome 0.

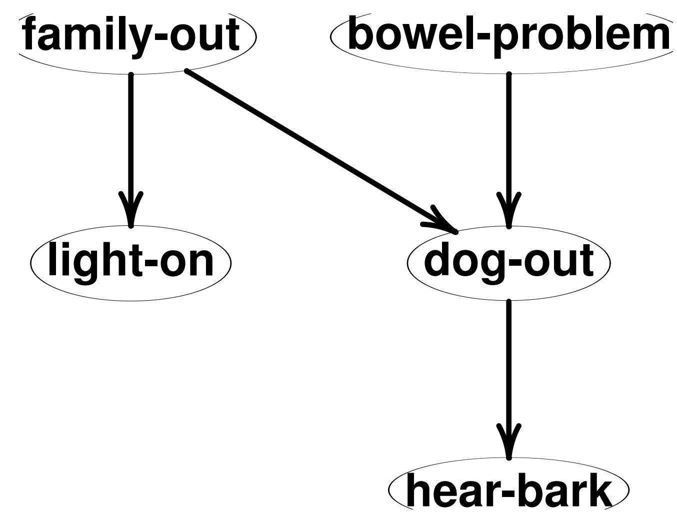
4. $x_H \sim p_H(D = 1) = (0.3, 0.7)$
   say with outcome 1.

The result is

$$x = (0, 1, 0, 1, 1)$$

and the case weight

$$w(x) = p_D(D = 1|F = 0, B = 1) = 0.97$$



F = 0	0.85
1	0.15

B = 0	0.8
1	0.2

F	0	1
L = 0	0.95	0.4
1	0.05	0.6

D	0	1
H = 0	0.99	0.3
1	0.01	0.7

B	0		1	
F	0	1	0	1
D = 0	0.7	0.1	0.03	0.01
1	0.3	0.9	0.97	0.99

Figure 15: Bayesian network for dog-problem.

Acceptance-rejection sampling

Acceptance-rejection sampling can be viewed as another instance of importance sampling. Here, the sampling distribution is $q := p$ (i.e., the distribution without evidence entered; the target distribtion is p_E !) and the case weight

$$w(x) := I_e(x) := \begin{cases} 1, & \text{if } x|_{\mathrm{dom}(E)} = \mathsf{val}(E) \\ 0, & \text{otherwise} \end{cases}$$

References

[CD00] Jian Cheng and Marek J. Druzdzel. Ais-bn: An adaptive importance sampling algorithm for evidential reasoning in large bayesian networks. *Journal on Artificial Intelligence*, 13:155–188, 2000.

[FC90] R. Fung and K. Chang. Weighting and integrating evidence for stochastic simulation in bayesian networks. In M. Henrion, R.D. Shachter, L. N. Kanal, and J. F. Lemmer, editors, *Uncertainty in Artificial Intelligence 5*, pages 209–219. North Holland, Amsterdam, 1990.

[Hen88] M. Henrion. Propagation of unvertainty by logic sampling in bayes' networks. In J. F. Lemmer and L. N. Kanal, editors, *Uncertainty in Artificial Intelligence 2*, pages 149–164. North Holland, Amsterdam, 1988.

[Sha03] Jun Shao. *Mathematical Statistics*. Springer, 2003.

[SP90] R. D. Shachter and M. Peot. Simulation approaches to general probabilistic inference on belief networks. In M. Henrion, R. D. Shachter, L. N. Kanal, and J. F. Lemmer, editors, *Uncertainty in Artificial Intelligence 5*, pages 221–231. North Holland, Amsterdam, 1990.