

Variable Elimination as a Form of Forgetting in (Lifted) Probabilistic Graphical Models

FILMRISS

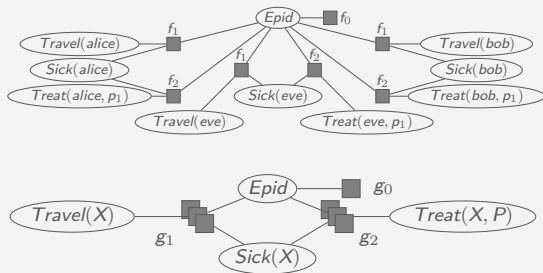
Forgetting In Lifted Models and Reasoning In Search of Solutions
(Version 1.1)

Tanya Braun



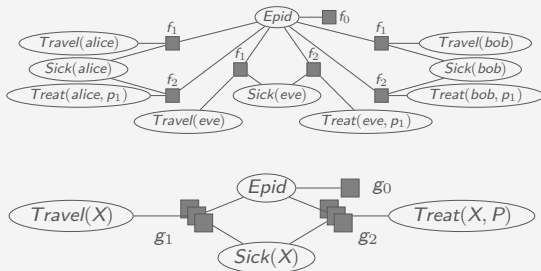
(Lifted) Probabilistic Graphical Models (PGMs)

- ▶ Represent a full joint probability distribution over a set of random variables
- ▶ Set of factors over subsets of random variables based on (conditional) independences among them
- ▶ *Lifted*: indistinguishable factors represented by a single factor parameterised using logical variables (parfactor)



An Intuitive Idea What Forgetting Can Mean in PGMs

- ▶ Keep whatever you want to forget implicitly in the (factorised) full joint but forget its explicit representation
- ▶ Full joint remains “the same” through valid mathematical operations on the (factorised) full joint
 - ▶ “the same”: queries for the remaining variables return the same result as before, i.e., $P_F(\mathbf{R} \mid \mathbf{e}) = P_{F'}(\mathbf{R} \mid \mathbf{e})$, where F refers to the model before forgetting and F' to the model afterwards



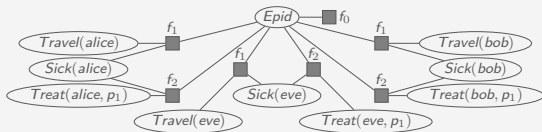
What Makes Sense to Forget under these Circumstances?

Propositional PGM

- ▶ Random variables
- ▶ Factors

Lifted PGM

- ▶ Constants (analogous to forgetting a random variable or a factor)
- ▶ (Parameterised) random variables
- ▶ Parfactors



Forgetting Random Variables and Factors

Forgetting random variable R :

Sum out R

- ▶ $f \leftarrow$ Multiply factors f_i that contain R
- ▶ Sum out R in $f = \phi(R, \mathbf{R})$:
for each $\mathbf{r} \in \text{ran}(\mathbf{R})$

$$\phi(\mathbf{r}) = \sum_{r \in \text{ran}(R)} \phi(R = r, \mathbf{r})$$

Forgetting factor f :

Multiply f with another factor f'

- ▶ $f'' \leftarrow$ Multiply $f = \phi(\mathbf{R})$ and $f' = \phi'(\mathbf{S})$:
for each $\mathbf{t} \in \text{ran}(\mathbf{T})$, $\mathbf{T} = \mathbf{R} \bowtie \mathbf{S}$,
 $\mathbf{r} = \pi_{\mathbf{R}}(\mathbf{t})$, $\mathbf{s} = \pi_{\mathbf{S}}(\mathbf{t})$

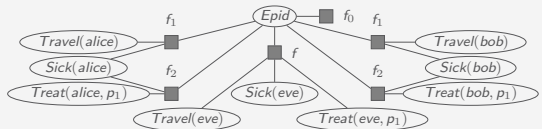
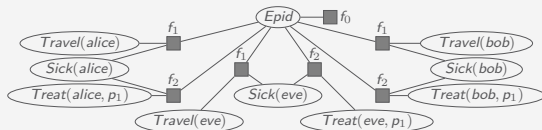
$$\phi''(\mathbf{t}) = \phi(\mathbf{r}) \cdot \phi'(\mathbf{s})$$

- ▶ How to choose f' ? Ideas:
 - ▶ Factor with largest overlap
 - ▶ Smallest size of resulting factor

Forgetting Random Variables and Factors: Example

Forgetting $f_1 = \phi_1(E, Tl.e, S.e)$: Multiply f_1
with $f_2 = \phi_2(E, Tt.e, S.e)$

$\phi''(E, S.e, Tl.e, Tt.e) \leftarrow \phi_1(E, S.e, Tl.e) \cdot \phi_2(E, S.e, Tt.e)$



Forgetting Random Variables and Factors: Example

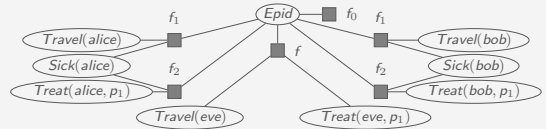
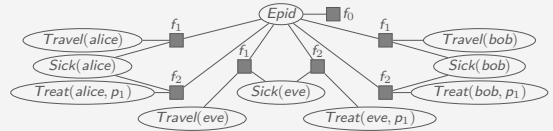
Forgetting $f_1 = \phi_1(E, Tl.e, S.e)$: Multiply f_1 with $f_2 = \phi_2(E, Tt.e, S.e)$

$$\phi''(E, S.e, Tl.e, Tt.e) \leftarrow \phi_1(E, S.e, Tl.e) \cdot \phi_2(E, S.e, Tt.e)$$

Forgetting $Sick.eve$: Sum out $Sick.eve$

$$\phi(E, S.e, Tl.e, Tt.e) \leftarrow \phi_1(E, S.e, Tl.e) \cdot \phi_2(E, S.e, Tt.e)$$

$$\phi(E, Tl.e, Tt.e) \leftarrow \sum_{s \in \text{ran}(S.e)} \phi(E, S.e = s, Tl.e, Tt.e)$$



Forgetting Random Variables and Factors: Example

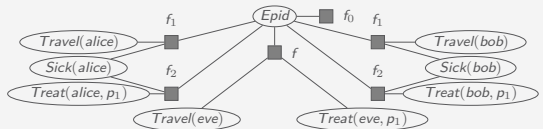
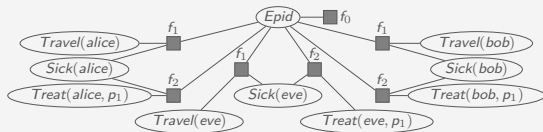
Forgetting $f_1 = \phi_1(E, Tl.e, S.e)$: Multiply f_1 with $f_2 = \phi_2(E, Tt.e, S.e)$

$$\phi''(E, S.e, Tl.e, Tt.e) \leftarrow \phi_1(E, S.e, Tl.e) \cdot \phi_2(E, S.e, Tt.e)$$

Forgetting $Sick.eve$: Sum out $Sick.eve$

$$\phi(E, S.e, Tl.e, Tt.e) \leftarrow \phi_1(E, S.e, Tl.e) \cdot \phi_2(E, S.e, Tt.e)$$

$$\phi(E, Tl.e, Tt.e) \leftarrow \sum_{s \in \text{ran}(S.e)} \phi(E, S.e = s, Tl.e, Tt.e)$$



Forgetting in PGMs

= Standard variable elimination operations

Forgetting in lifted PGMs: Parfactor

Forgetting parfactor g :

Multiply g with another parfactor g' using lifted variable elimination

- ▶ $g'' \leftarrow$ Multiply g and g' (lifted)
 - ▶ Multiplication is scaled to account for different number of groundings represented by g and g'



Forgetting in lifted PGMs: Parfactor

Forgetting parfactor g :

Multiply g with another parfactor g' using lifted variable elimination

- ▶ $g'' \leftarrow$ Multiply g and g' (lifted)
 - ▶ Multiplication is scaled to account for different number of groundings represented by g and g'

Example: Forgetting parfactor g_0 :

Multiply g_0 with g_1

- ▶ Scale for g_0 : $\sqrt{3}$, because g_1 stands for $|dom(X)| = 3$ factors



Forgetting in lifted PGMs: Parameterised Random Variable

Forgetting parameterised variable $A = R(\mathbf{X})$:
Sum out A (lifted)

- ▶ $g \leftarrow$ Multiply parfactors g_i containing A
- ▶ $g \leftarrow$ Sum out A from g



Forgetting in lifted PGMs: Parameterised Random Variable

Forgetting parameterised variable $A = R(\mathbf{X})$:
Sum out A (lifted)

- ▶ $g \leftarrow$ Multiply parfactors g_i containing A
- ▶ $g \leftarrow$ Sum out A from g

Example: Forget $Treat(X, P)$

- ▶ $g \leftarrow g_2$
- ▶ $g \leftarrow$ Sum out $Treat(X, P)$ from g
 - ▶ Sum over range values of $Treat(X, P)$
 - ▶ Exponentiate result with $|dom(P)|$



Forgetting in lifted PGMs: Parameterised Random Variable

Forgetting parameterised variable $A = R(\mathbf{X})$:
Sum out A (lifted)

- ▶ $g \leftarrow$ Multiply parfactors g_i containing A
- ▶ $g \leftarrow$ Sum out A from g

Example: Forget $Treat(X, P)$

- ▶ $g \leftarrow g_2$
- ▶ $g \leftarrow$ Sum out $Treat(X, P)$ from g
 - ▶ Sum over range values of $Treat(X, P)$
 - ▶ Exponentiate result with $|dom(P)|$



Forgetting in lifted PGMs

= Lifted variable elimination operations

Forgetting in lifted PGMs: Parameterised Random Variable

Forgetting parameterised variable $A = R(\mathbf{X})$:
Sum out A (lifted)

- ▶ Problem: Preconditions
 - ▶ A only contained in one parfactor (same as with VE)
 - ▶ A contains all logical variables occurring in g
 - ▶ Logical variables of A *count-normalised* w.r.t. remaining logical variables
- ▶ Effect: Can lead to groundings



Forgetting in lifted PGMs: Parameterised Random Variable

Forgetting parameterised variable $A = R(\mathbf{X})$:
Sum out A (lifted)

► Problem: Preconditions

- A only contained in one parfactor (same as with VE)
- A contains all logical variables occurring in g
- Logical variables of A count-normalised w.r.t. remaining logical variables

► Effect: Can lead to groundings



- Forget $Treat(X, P)$: ✓
- Forget $Travel(X)$: ✓
- Forget $Sick(X)$: Counting P necessary
- Forget $Epid$: Grounding X necessary

Forgetting in lifted PGMs: Constants

Forgetting constant $x \in \text{dom}(X)$:

Sum out all variables $R(x, \mathbf{X})$ whose PRV $R(X, \mathbf{X})$ contains x in its constraint for X :

- ▶ Shatter G on x (grounds model w.r.t. x)
- ▶ Sum out all $R(x, \mathbf{X})$ using (lifted) variable elimination



Forgetting in lifted PGMs: Constants

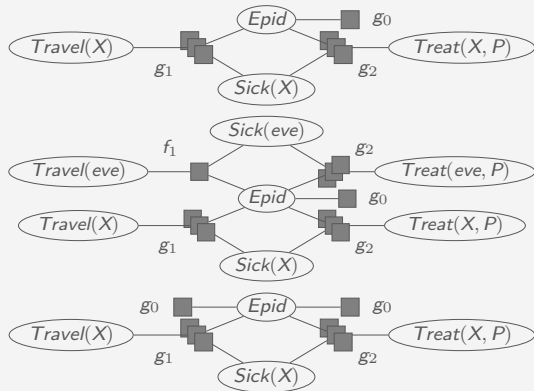
Forgetting constant $x \in \text{dom}(X)$:

Sum out all variables $R(x, \mathbf{X})$ whose PRV $R(X, \mathbf{X})$ contains x in its constraint for X :

- ▶ Shatter G on x (grounds model w.r.t. x)
- ▶ Sum out all $R(x, \mathbf{X})$ using (lifted) variable elimination

Example: Forget eve

- ▶ Shatter G on eve (split off eve)
- ▶ Sum out $S(\text{eve})$, $Tl(\text{eve})$, $Tt(\text{eve}, P)$



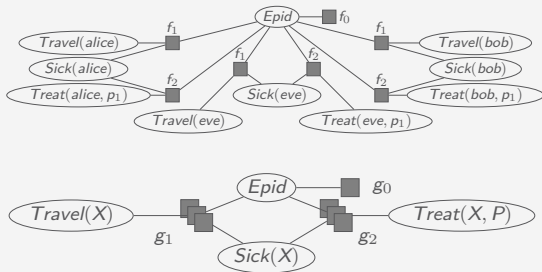
Forgetting in (lifted) PGMs

Forgetting in (lifted) PGMs

(Lifted) variable elimination operations to forget (parameterised) random variables and (par)factors (, and constants)

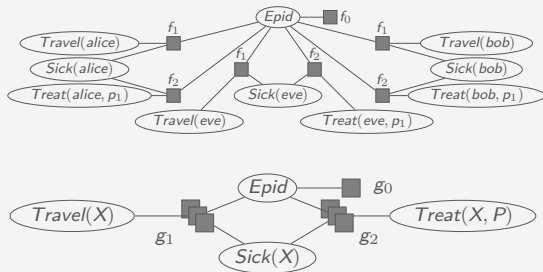
Tractability & Completeness

Results from lifted variable elimination carry over to forgetting.



Other Things to Forget?

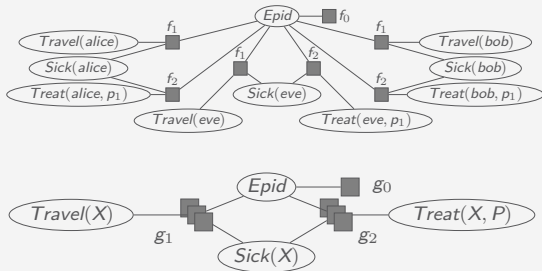
- Influences, i.e., edges (direct / indirect)
 → *Some form of front door / back door criterion to cut all connections between them?*
- Probabilities (certainty about them) →
Higher-order probabilities? Interval probabilities?



What about Deletion as a Way of Forgetting?

The easy ones

- ▶ Delete (par)factor from model $\rightarrow$ Removes information (no longer implicitly in the full joint); can change other influences in the model as well
- ▶ Delete constants $\rightarrow$ Corresponds to deleting propositional factors



What about Deletion as a Way of Forgetting?

The difficult ones

- ▶ Delete variables → *Delete all factors that contain variable?*
- ▶ Delete edges → Delete variables from (par)factors (*How do you even do this? Sum out locally?*)
- ▶ Delete influence between variables? → *Might correspond to a form of deleting edges*

