

Variable Elimination for Forgetting in Probabilistic Graphical Models

A Work in Progress

Tanya Braun, Marcel Gehrke, Kai Sauerwald, Jan Speller



Probabilistic Graphical Models (PGMs)

- Set of factors F over subsets of random variables based on (conditional) independences among them
- Represent a full joint probability distribution P_F as a normalised product of the factors

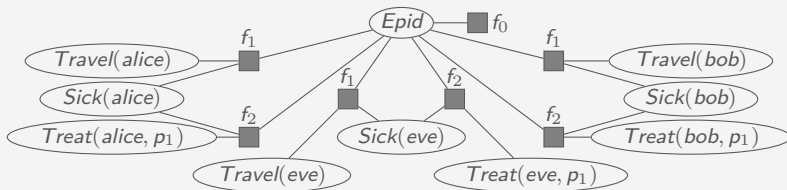


Figure: Example factor graph

Forgetting in PGMs: Objectives & Use Cases

Objectives

- ▶ Exploration of forgetting concept in marginalisation
- ▶ Long-term goal: Complexity reduction in temporal case

Use cases

- ▶ Privacy: Forgetting variables that should no longer be stored without relearning a model
- ▶ Mental models: Emulate human forgetting
- ▶ Text analysis / digital humanities: Forgetting of characters over time / long texts
- ▶ Explanations: Components that are somehow central should not be forgotten

Forgetting in PGMs: One Possible Interpretation

- ▶ Idea: Forgetting as deleting parts of a model F
 - ▶ $\text{forget}(\delta, F)$ returns a model F' , in which δ is “somehow” forgotten
 - ▶ Components to forget (δ): Random variable, factorisation (factor), edge

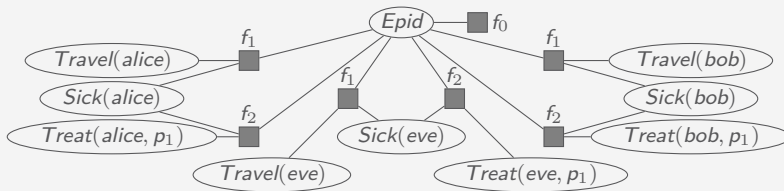


Figure: Example factor graph

Possible Properties

- ▶ *Semantics preserving*: Queries return the same result as before, i.e.,

$$P_F(\mathbf{R} \mid \mathbf{e}) = P_{\text{forget}(\delta, F)}(\mathbf{R} \mid \mathbf{e})$$

- ▶ *Reversible*: Original model can be restored from the one after forgetting, i.e.,

$$\text{remember}(\delta', \text{forget}(\delta, F)) = F$$

- ▶ *Order-independent*: The order of forgetting does not matter, i.e.,

$$\text{forget}(\delta_2, \text{forget}(\delta_1, F)) = \text{forget}(\delta_1, \text{forget}(\delta_2, F))$$

- ▶ *Idempotent*: Forgetting something twice does not change the model further, i.e.,

$$\text{forget}(\delta, F) = \text{forget}(\delta, \text{forget}(\delta, F))$$

Marginalisation as Forgetting

- ▶ Use variable elimination (VE) operators to implement forgetting in a model F
 - ▶ *Multiply* two factors $f' = \phi'(\mathbf{R}')$, $f'' = \phi''(\mathbf{R}'')$: Multiply the values for all possible values $\mathbf{r} \in \text{ran}(\mathbf{R}' \cup \mathbf{R}'')$ projected onto $\mathbf{R}', \mathbf{R}''$, i.e., $\text{multiply}(f', f'') = f$ with

$$\forall \mathbf{r} \in \text{ran}(\mathbf{R}' \cup \mathbf{R}'') : \phi(\mathbf{r}) = \phi'(\pi_{\mathbf{R}'}(\mathbf{r})) \cdot \phi''(\pi_{\mathbf{R}''}(\mathbf{r}))$$

- ▶ Postcondition: $P_{F \setminus \{f', f''\} \cup \{\text{multiply}(f', f'')\}} = P_F$
- ▶ *Sum out* a variable R from a factor $f = \phi(R, \mathbf{R})$: Sum over the range values of R for all possible values $\mathbf{r}$ of the remaining variables $\mathbf{R}$, i.e., $\text{sum-out}(R, f) = f' = \phi'(\mathbf{R})$ with

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

- ▶ Precondition: f is the only factor containing R , i.e., $\forall f' \in F, f' \neq f, R \notin \text{rv}(f')$ (otherwise, multiply all factors containing R first)
 - ▶ Postcondition: $P_{F \setminus \{f\} \cup \{\text{sum-out}(R, f)\}} = \sum_{r \in \text{ran}(R)} P_F$

Forgetting Random Variables and Factors

Forget factor f'

$\triangleq$

Multiply f' with another factor f''

1. $f \leftarrow$ Multiply f', f'' using multiply

How to choose target f'' ?

- ▶ Factor with largest overlap
- ▶ Smallest size of resulting factor

Note that $\text{forget}(f', F) = \text{forget}(f'', F)$

Forget random variable R

$\triangleq$

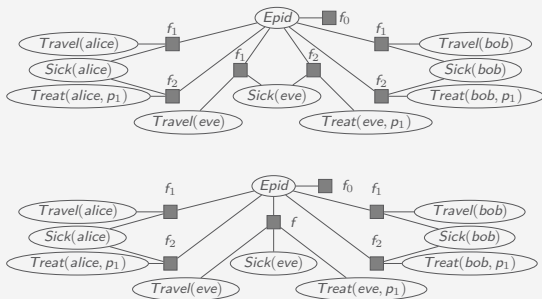
Sum out R

1. $f \leftarrow$ Multiply factors f_i that contain R using multiply
2. Sum out R in f using sum-out

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

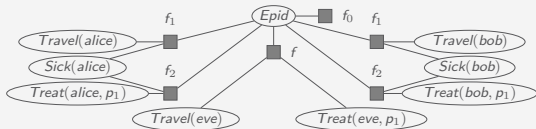
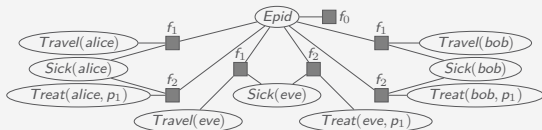
Forgetting $f_1 = \phi_1(E, Tl.e, S.e)$: Multiply f_1 with $f_2 = \phi_2(E, Tt.e, S.e)$

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

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

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

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



Effects on Runtime Complexity

- ▶ Runtime complexity for an operation in a model F

$O(r^w)$ with $r = \max_{R \in rv(F)} |ran(R)|$ and w the largest interim result

- ▶ w bounded from below by the maximum number of inputs in any factor: $\max_{f \in F} |rv(f)|$
 - ▶ w bounded above by the number of variables in F : $n = |rv(F)|$
 - ▶ Optimal elimination order leads to a minimum w (tree width)
- ▶ Forgetting a factor in F can increase w : increasing the bound from below
 - ▶ Forgetting a variable in F can increase w : forcing a non-optimal elimination order

Properties

- ▶ *Semantics preserving*: ✓ [by pre / postconditions of operators]
- ▶ *Reversible*: only with additional information
 - ▶ Graph: Inputs of original factors need to be stored (linear in w)
 - ▶ Potentials: Actual numbers necessary (exponential in w)
- ▶ *Order-independent*: ✓ [by commutativity of multiplication / addition; but complexity might change!]
- ▶ *Idempotent*: ✓ [not really possible to forget δ twice as it is no longer accessible in F]

What about Edges?

- ▶ Edges encode direct dependences
- ▶ Forgetting an edge means eliminating a variable R from a single factor f
 - Not possible with above operators
- ▶ For each remaining r , there is a set of potentials that the values r of R map to
 - [How do we arrive at a single value?](#)
 - ▶ Sum? Max? Average?
 - Only semantics preserving if using sum and R only occurs in f



Forget edge $(B, f) \rightarrow$ Remove B from f

A	B	ϕ	ϕ'
<i>true</i>	<i>true</i>	φ_1	$\varphi_1 \circ \varphi_2$
<i>true</i>	<i>false</i>	φ_2	
<i>false</i>	<i>true</i>	φ_3	$\varphi_3 \circ \varphi_4$
<i>false</i>	<i>false</i>	φ_4	

Other Dimensions

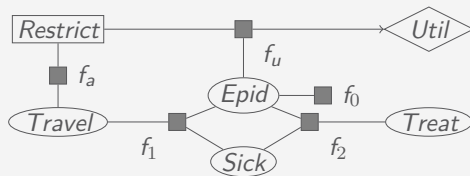
► First-order PGMs

- Logical variables encode sets of indistinguishable objects
- Can be solved using lifted VE
- Forgetting indistinguishable objects



► Decision-theoretic PGMs

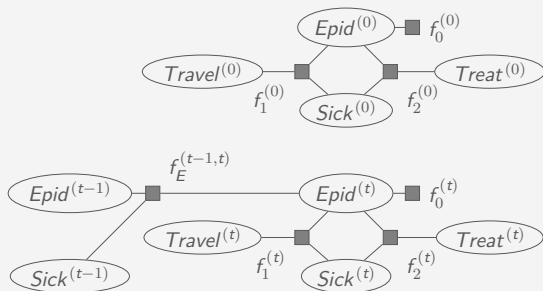
- PGMs extended with decision and utility nodes
- Inference tasks: MEU problem
- Can be solved using VE operations
- Handling of decisions and utilities



Other Dimensions

► Temporal PGMs

- Factors to describe inter- and intra-slice behaviour
- Inference tasks: filtering, prediction, hindsight
- Can be solved using VE operations
- Forgetting evidence over time (using TAME by Gehrke et al., 2020)
- Forgetting factors between time steps
 - Markov- k to Markov- $k - 1$ assumption $\rightarrow$ decreases complexity



Other Dimensions

- ▶ Combinations of the above that can (*probably*) be handled similarly
 - ▶ First-order decision-theoretic PGMs
 - ▶ First-order temporal PGMs
 - ▶ First-order temporal decision-theoretic PGMs
- ▶ Further variants of interest:
 - ▶ Continuous models, e.g., Gaussian Bayesian networks
 - ▶ Causal models: directed models encoding cause-effect relationships
 - ▶ Partially directed models: allow for encoding causality and correlation
 - ▶ Combinations with first-order or temporal constructs

Other Interpretations of Forgetting in PGMs

- Potentials (certainty about them) $\rightarrow$ Variant of *Credal nets*
 - Use intervals to denote forgetting specific potentials

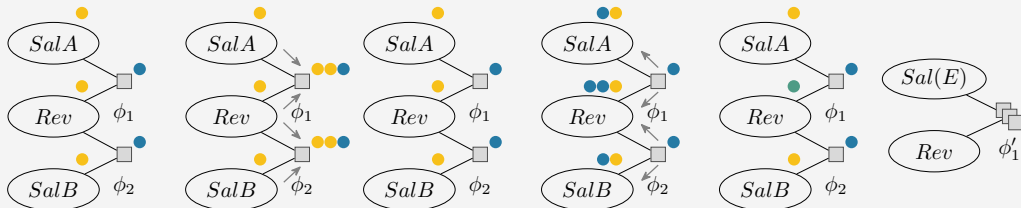


A	B	ϕ	ϕ'
true	true	φ_1	$\varphi_1 \pm \varepsilon$
true	false	φ_2	$\varphi_2 \pm \varepsilon$
false	true	φ_3	$\varphi_3 \pm \varepsilon$
false	false	φ_4	$\varphi_4 \pm \varepsilon$

Other Interpretations of Forgetting in PGMs

- Abstraction $\rightarrow \epsilon$ -Advanced Colour Passing by Luttermann et al. (2024)

- Combine ϵ -equivalent factors to abstract away from details



- Moves from propositional model to first-order model

Summary

- ▶ Use cases in privacy, mental modelling, text analysis
- ▶ Properties: semantics-preserving, reversible, order-independent, idempotent
- ▶ Marginalisation as forgetting using VE
 - ▶ Sum-out, multiply operators to realise forgetting of variables, factors
 - ▶ Forgetting can negatively impact complexity
 - ▶ Forgetting edges not obvious how to handle
- ▶ Extensions: first-order, decision-theoretic, temporal, combinations thereof
- ▶ Other interpretations: forgetting potentials, abstraction

*This work was partially funded by
the Ministry of Culture and Science of the German State of North Rhine-Westphalia.*