Supplementary material for:

Predictive coding and the Bayesian brain: Intractability hurdles that are yet to overcome

by Johan Kwisthout and Iris van Rooij

In this supplementary material we provide proofs of the complexity-theoretic results we claim in our text. First we provide an overview of some needed concepts and definitions from Bayesian networks and complexity theory (Section 1). We then formally define the PREDICTION, ERROR-COMPUTATION, and HYPOTHESIS-UPDATING problems we introduced (informally) in the main text and prove NP-hardness of these problems (Section 2). In Section 3, we discuss some parameters that, when constrained, render the above problems tractable.

1 Preliminaries

For readers unfamiliar with basic notations from Bayesian networks and complexity theory we review some of the basics relevant for our purpose. For details and more background we refer the reader to textbooks like (Pearl, 1988; Jensen & Nielsen, 2007; Koller & Friedman, 2009; Garey & Johnson, 1979; Downey & Fellows, 1999) or to the references cited in the respective subsections.

1.1 Bayesian networks

A Bayesian or probabilistic network $\mathcal{B}$ is a graphical structure that models a set of stochastic variables, the conditional independencies among these variables, and a joint probability distribution over these variables. $\mathcal{B}$ includes a directed acyclic graph $\mathbf{G}_\mathcal{B} = (\mathbf{V}, \mathbf{A})$, modeling the variables and conditional independencies in the network, and a set of parameter probabilities Γ in the form of conditional probability tables (CPTs), capturing the strengths of the relationships between the variables. The network models a joint probability distribution $\Pr(\mathbf{V}) = \prod_{i=1}^{n} \Pr(V_i \mid \pi(V_i))$ over its variables, where $\pi(V_i)$ denotes the parents of V_i in $\mathbf{G}_\mathcal{B}$. We will use upper case letters to denote individual nodes in the network, upper case bold letters to denote sets of nodes, lower case letters to denote value assignments to nodes, and lower case bold letters to denote joint value assignments to sets of nodes. We will sometimes write $\Pr(\mathbf{x})$ as a shorthand for $\Pr(\mathbf{X} = \mathbf{x})$ if no ambiguity can occur. We denote with $\Omega(X)$ the set of all values that X can take; $\Omega(\mathbf{X})$ is defined analogously for sets of variables.

Arguably the most important computational problems in Bayesian networks are INFERENCE and MAP. In the INFERENCE problem we update the posterior probability distribution of some set of variables $\mathbf{H}$ using new observations $\mathbf{e}$, i.e., we compute $\Pr(\mathbf{H} \mid \mathbf{e})$. In the MAP problem we seek to find the most probable value assignment $\mathbf{h}$ for $\mathbf{H}$, i.e., we compute $\text{argmax}_\mathbf{h} \Pr(\mathbf{H} = \mathbf{h} \mid \mathbf{e})$; this is also referred to as to find the *mode* of the probability distribution $\Pr(\mathbf{H} \mid \mathbf{e})$. In both cases, we may need to marginalize over other variables in the network, as $\Pr(\mathbf{H}) = \sum_{\mathbf{g} \in \Omega(\mathbf{G})} \Pr(\mathbf{H} \wedge \mathbf{g})$. Thus, while the probability distribution of any variable (and therefore also the value with the highest probability) can in essence be calculated using well-known laws in probability theory, e.g., the chain rule, marginalisation, and conditioning, this calculation can take a time which is exponential in the size of the network. We formally define the INFERENCE and MAP problems below.

INFERENCE
Instance: Let $\mathcal{B} = (\mathbf{G}_\mathcal{B}, \Pr)$ be a probabilistic network, with $\mathbf{G}_\mathcal{B} = (\mathbf{V}, \mathbf{A})$ denoting the dependencies between the variables and Pr denoting the joint probability distribution. Let $\mathbf{H} \subseteq \mathbf{V}$ denote a set of variables, and let $\mathbf{E} \subseteq \mathbf{V}$ denote a (possibly empty) set of evidence variables with joint value assignment $\mathbf{e}$.
Output: The probability distribution $\Pr(\mathbf{H} \mid \mathbf{e})$.

MAP
Instance: As in INFERENCE.
Output: The mode of the probability distribution $\Pr(\mathbf{H} \mid \mathbf{e})$.

An important parameter of a Bayesian network is its *treewidth*. Treewidth has been introduced informally in the main text as a measure on the localness of the dependencies in the network. We now formally define the treewidth (Robertson & Seymour, 1986) of a Bayesian network $\mathcal{B}$ as the treewidth of a triangulation of the moralization $\mathbf{G}_\mathcal{B}^\mathbf{M}$ of its graph $\mathbf{G}_\mathcal{B}$. This moralization is the undirected graph that is obtained from $\mathbf{G}_\mathcal{B}$ by adding arcs so as to connect all pairs of parents of a variable, and then dropping all directions; we will use the phrase 'moralized graph' to refer to the moralization of the graph of a network. A triangulation of the moralized graph $\mathbf{G}_\mathcal{B}^\mathbf{M}$ is any graph $\mathbf{G_T}$ that embeds $\mathbf{G}_\mathcal{B}^\mathbf{M}$ as a subgraph and in addition is chordal, that is, it does not include loops of more than three variables without any pair being adjacent in $\mathbf{G_T}$. A tree-decomposition of a triangulation $\mathbf{G_T}$ is a tree $\mathbf{T_G}$ such that

- each node $\mathbf{X}_i$ in $\mathbf{T_G}$ is a bag of nodes which constitute a clique in $\mathbf{G_T}$;
- for every i, j, k, if $\mathbf{X}_j$ lies on the path from $\mathbf{X}_i$ to $\mathbf{X}_k$ in $\mathbf{T_G}$, then $\mathbf{X}_i \cap \mathbf{X}_k \subseteq \mathbf{X}_j$.

The width of the tree-decomposition $\mathbf{T_G}$ of the graph $\mathbf{G_T}$ equals $\max_i(|\mathbf{X}_i| - 1)$, that is, it equals the size of the largest clique in $\mathbf{G_T}$, minus 1. The treewidth of a Bayesian network $\mathcal{B}$ now is defined as the minimum width over all possible tree-decompositions of triangulations of $\mathbf{G}_\mathcal{B}^\mathbf{M}$.

To compare two probability distributions (e.g., the actual and an estimated distribution) of a Bayesian network, typically the Kullback-Leibler divergence is used (Kullback & Leibler, 1951). The KL-divergence D_{KL} between a predicted or estimated distribution $\Pr_\mathrm{P}$ and the observed or actual distribution $\Pr_\mathrm{O}$ is defined as follows:

$$D_{KL}(\Pr_\mathrm{P}, \Pr_\mathrm{O}) \stackrel{def}{=} \sum_\omega \begin{cases} 0 & \text{if } \Pr_\mathrm{P}(\omega) = \Pr_\mathrm{O}(\omega) = 0 \\ \Pr_\mathrm{O}(\omega) \ln \frac{\Pr_\mathrm{O}(\omega)}{\Pr_\mathrm{P}(\omega)} & \text{otherwise} \end{cases}$$

This measure is undefined if $\Pr_\mathrm{P}(\omega) = 0$ and $\Pr_\mathrm{O}(\omega) \neq 0$ for any ω. Note that the KL-divergence is not a metric, as it is not symmetrical: typically $D_{KL}(\Pr_\mathrm{P}, \Pr_\mathrm{O}) \neq D_{KL}(\Pr_\mathrm{O}, \Pr_\mathrm{P})$.

To compare two joint value assignments, one can use the Hamming distance (Hamming, 1950), where $D_H(\mathbf{p}, \mathbf{o})$ equals the number of variables where the corresponding value assignments are different. The Hamming distance is symmetric and is a metric.

1.2 Parameterized complexity theory

In the remainder, we assume that the reader is familiar with basic concepts of computational complexity theory, such as the complexity classes P and NP, and NP-completeness proofs using polynomial-time many-one (or *Karp*) reductions. In addition to these basic concepts we discuss some aspects from parameterized complexity theory, in particular the W-hierarchy and the class FPT.

Sometimes problems are intractable (i.e., NP-hard) in general, but become tractable if some *parameter* of the problem can be assumed to be small. A *parameterized problem* is a pair (Π, κ) of a decision problem Π and a polynomial time computable *parameterization* $\kappa : \{0,1\}^* \to \mathbb{N}$ mapping strings to natural numbers. The parameterized problem (Π, κ) is *fixed-parameter tractable* if there exists an algorithm deciding every instance (x, k) of (Π, κ) with running time $\mathcal{O}(f(\kappa(x,k)) \cdot \ |x|^c)$ for an arbitrary computable function f and a constant c, independent of $|x|$ (Flum & Grohe, 2006; Downey & Fellows, 1999). The class of all fixed-parameter tractable decision problems is denoted as FPT. To improve readability, if the parameterization is clear from the context (e.g., $\kappa(x,k) = k$), we just mention the parameter k or parameter set $\{k_1, \ldots, k_n\}$.

Informally, a problem is called fixed-parameter tractable for a parameter l if it can be solved in time, exponential *only* in $\{k_1, \ldots, k_n\}$ and polynomial in the input size $|x|$. In practice, this means that problem instances can be solved efficiently, even when the problem is NP-hard in general, if the parameters $\{k_1, \ldots, k_n\}$ are known to be small. In contrast, the W-hierachy of complexity classes consist of problems that are assumed to be fixed-parameter *intractable* for parameters $\{k_1, \ldots, k_n\}$. FPT and the lowest level of the W-hierachy, viz. W[1], play a similar role in parameterized complexity theory as P and NP in traditional complexity theory; for instance, it is widely assumed that FPT is a strict subset of W[1]. To prove completeness for W[1] or some other level of the W-hiearchy, so-called FPT-reduction are used. An FPT-reduction is a many-one reduction from (Π_1, κ_1) to (Π_2, κ_2) that is allowed to run in $\mathcal{O}(f(\{k_1, \ldots, k_n\}) \cdot \ |x|^c)$, and where the parameters are carried over, i.e., $\kappa_2 = g(\kappa_1)$ for some function g.

2 Complexity proofs

In the remainder, to obtain continuity between the different problems, we denote with $\mathbf{H}$ the variables constituting the top-down signal, i.e., the hypothesised causes of the observation, and with $\mathbf{O}$ the variables constituting the bottom-up signal, i.e., the (predicted or observed) observation. In the predictive coding framework, based on so-called Empirical Bayes, the posteriors of each level (the hypothesis) define the priors on the level below (the prediction). Each level L of the hierarchy can thus be defined as a separate Bayesian network $\mathcal{B}_L$, whose variables are partitioned into hypothesis variables $\mathbf{H_L}$ (carrying the top-down signal), observation variables $\mathbf{O_L}$ (carrying the signal to be predicted), and intermediate variables $\mathbf{I_L}$, where we assume without loss of generality that all hypothesis variables are roots and all observation variables are leafs (see Figure 1). Note that the posteriors on the observations at level L define the priors at level $L-1$, i.e., the observations at level $\mathbf{O_L}$ and the hypothesis variables at level $\mathbf{H_{L-1}}$ are identified.

In the literature, two distinct ways of thinking about predictions and hypotheses can be seen, based on whether one sees the inference steps as *fixating* ones belief (Friston, 2002) or as *updating* ones

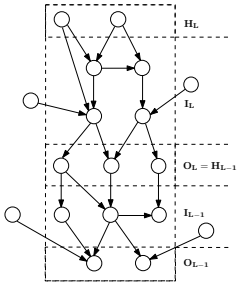

Figure 1: Two levels of the predictive coding hierarchy. Note that the hypothesis variables of level $L+1$ are identified with the observation variables at level L. The intermediate variables at each level are those variables that are neither hypothesis nor observation. They may also be contextual variables that influence the computations on particular levels, but are no part of the knowledge structure of that level themselves

---

distribution over beliefs (Knill & Pouget, 2004) The former is associated with finding MAPs, or 'MAX-propagation', the latter with updating probability distributions, or 'SUM-propagation'. We will use MAX, respectively SUM to denote the two variants of this inference steps[1]. In the generative (backward) process, either a joint value assignment $\mathbf{h}$ for $\mathbf{H}$, or a probability distribution $\Pr_H$ over $\mathbf{H}$, is assumed, and a joint value assignment $\mathbf{o}$ for $\mathbf{O}$ (MAX), or a probability distribution $\Pr_O$ over $\mathbf{O}$ (SUM), is computed. In the inference (forward) process, either a prediction $\mathbf{o}$ or a distribution $\Pr_O$ is assumed and the most probable hypothesis $\mathbf{h}$ (MAX) or the posterior distribution $\Pr_H$ (SUM) is computed.

[1]Note that SUM and MAX refer to the *output* of the inference step, rather than the *input*. It is indifferent (from a computational point of view) whether the input is a value assignment or a probability distribution, but we will assume that the input matches the output, i.e., input and output are either both joint value assignments (MAX) or both probability distributions (SUM).

Note that in predictive coding $\mathbf{h}$ (respectively $\Pr_H$) is not inferred directly from $\mathbf{o}$ (respectively $\Pr_O$), but a candidate hypothesis $\mathbf{h_c}$ (respectively candidate distribution $\Pr_{H_c}$) is updated, given the *error* between the predicted value (or distribution) of $\mathbf{O}$ and the actual observation $\mathbf{o_a}$ to $\mathbf{O}$. We operationalize this error-computation as computing the *distance* between prediction and observation. If the prediction is a joint value assignment $\mathbf{o_p}$, then it is natural to compute the *structural distance* (e.g., the Hamming distance or edit distance) between $\mathbf{o_p}$ and $\mathbf{o_a}$. If the prediction is a probability distribution $\Pr_{O_p}$, then it is natural to compute the *divergence* (e.g., the Kullback-Leibler divergence) between $\Pr_{O_p}$ and the (deterministic) distribution $\Pr_{O_a}$ associated with $\mathbf{o_a}$.

Furthermore, one can look at the updating stap as finding an updated hypothesis that minimizes prediction error (Kilner, Friston, & Frith, 2007) or as finding an updated hypothesis that best explains the observation (Friston, 2002). We will denote the former problem as Hypothesis-Updating (PRED) and the latter as Hypothesis-Updating (EXPL). Observe that if prediction error is zero, then the hypothesis is the most likely one, given the observation. From Bayes' theorem it follows that both minimzing prediction error and finding the best explaining hypothesis are related (when the prediction error is zero):

$$\Pr(\text{hypothesis} \mid \text{observation}) \propto \Pr(\text{observation} \mid \text{hypothesis}) \cdot \Pr(\text{hypothesis})$$

That is, finding the best explaining updated hypothesis corresponds to finding a hypothesis that yields zero prediction error if the prior distribution over the candidate hypotheses is uniform. Note, however, that *approximating* the best explaining hypothesis can not be *identified* with lowering prediction error, although one approach may well be a reasonable heuristic for the other, and vice versa.

2.1 Prediction

In the main paper, we defined Prediction as the problem of determining the posterior probability distribution (respectively the mode of that distribution), given the priors of (respectively a joint value assignment to) the hypothesis nodes. We now formally define these problems as follows.

Prediction (SUM)
Instance: A Bayesian network $\mathcal{B} = (\mathbf{G}_\mathcal{B}, \Gamma)$, where $\mathbf{V}(\mathbf{G}_\mathcal{B})$ is partitioned into a set of observed nodes $\mathbf{H}$, a set of prediction nodes $\mathbf{O}$, and a set of intermediate nodes $\mathbf{I}$.
Output: $\Pr_{O_p}$, i.e., the probability distribution over the prediction nodes $\mathbf{O}$.

Prediction (MAX)
Instance: A Bayesian network $\mathcal{B} = (\mathbf{G}_\mathcal{B}, \Gamma)$, where $\mathbf{V}(\mathbf{G}_\mathcal{B})$ is partitioned into a set of observed nodes $\mathbf{H}$ with observation $\mathbf{h}$, a set of prediction nodes $\mathbf{O}$, and a set of intermediate nodes $\mathbf{I}$.
Output: $\text{argmax}_\mathbf{o} \Pr(\mathbf{o} \mid \mathbf{h})$, i.e., the most probable joint value assignment $\mathbf{o}$ to the prediction nodes $\mathbf{O}$ and the hypothesis $\mathbf{h}$, or $\perp$ if $\Pr(\mathbf{o} \mid \mathbf{h}) = 0$ for every joint value assignment $\mathbf{o}$ to $\mathbf{O}$.

Corollary 1. Prediction (SUM) *and* Prediction (MAX) *are* NP*-hard, both to compute exactly and to approximate.*

NP-hardness of PREDICTION follows directly from the NP-hardness of INFERENCE (which remains NP-hard if there is no evidence) (Cooper, 1990; Kwisthout, 2009) and MAP (Park & Darwiche, 2004; De Campos, 2011); from these results it follows that PREDICTION remains NP-hard if all variables are binary and if **O** and/or **H** consists of a singleton variable. Furthermore, PREDICTION inherits the inapproximability results of INFERENCE (Dagum & Luby, 1993) and MAP (Park & Darwiche, 2004).

2.2 Error-Computation

In the main paper, ERROR-COMPUTATION was defined as computing the Kullback-Leibler divergence (SUM), respectively Hamming distance (MAX), between the prediction and the observation. We formalize these problems as follows.

ERROR-COMPUTATION (SUM)
Instance: A Bayesian network $\mathcal{B} = (\mathbf{G}_\mathcal{B}, \Gamma)$, where $\mathbf{V}(\mathbf{G}_\mathcal{B})$ is partitioned into a set of observed nodes **H**, a set of prediction nodes **O**, and a set of intermediate nodes **I**; the predicted probability distribution $\Pr_{\mathrm{O_p}}$ over the prediction nodes, and the (deterministic) distribution $\Pr_{\mathrm{O_a}}$ corresponding to an observation $\mathbf{o_a}$.
Output: $D_{KL}(\Pr_{\mathrm{O_p}}, \Pr_{\mathrm{O_a}})$.

ERROR-COMPUTATION (MAX)
Instance: A Bayesian network $\mathcal{B} = (\mathbf{G}_\mathcal{B}, \Gamma)$, where $\mathbf{V}(\mathbf{G}_\mathcal{B})$ is partitioned into a set of observed nodes **H**, a set of prediction nodes **O**, and a set of intermediate nodes **I**; the predicted joint value assignment $\mathbf{o_p}$ to the prediction nodes, and the observed joint value assignment $\mathbf{o_a}$.
Output: $D_H(\mathbf{o_p}, \mathbf{o_a})$.

Note that computing the Kullback-Leibler distance between two arbitrary probability distributions is NP-hard (Kwisthout, 2009). We will show that ERROR-COMPUTATION (SUM) is NP-hard by a modification of the proof in Kwisthout (2009) to reflect that we compare an arbitrary probability distribution (i.e., the prediction) with an observation (i.e., a deterministic distribution where the probability of the observed value is set to 1 and all other values to 0).

We reduce from SATISFIABILITY, defined as follows:

SATISFIABILITY
Instance: A Boolean formula ϕ with n variables.
Question: Is ϕ satisfiable?

We construct a Bayesian network $\mathcal{B}_\phi$ from ϕ as follows. For each propositional variable x_i in ϕ, a binary stochastic variable X_i is added to $\mathcal{B}_\phi$, with possible values TRUE and FALSE and a uniform probability distribution. For each logical operator in ϕ, an additional binary variable in $\mathcal{B}_\phi$ is introduced, whose parents are the variables that correspond to the input of the operator, and whose conditional probability table is equal to the truth table of that operator. For example, the value TRUE of a stochastic variable mimicking the *and*-operator would have a conditional probability of

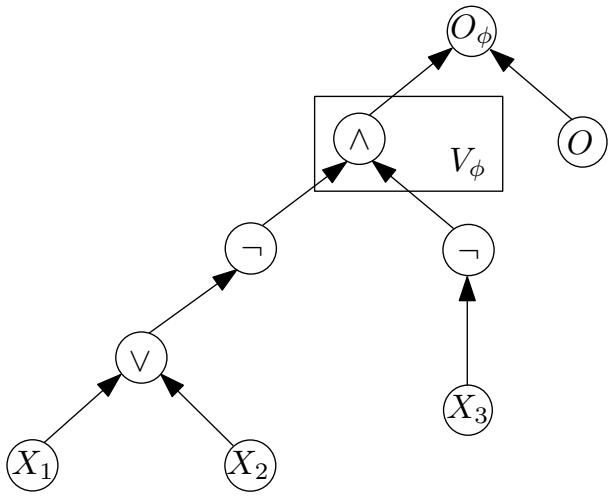

Figure 2: The Bayesian network $\mathcal{B}_\phi$ corresponding to $\phi = \neg(x_1 \vee x_2) \wedge \neg x_3$

1 if and only if both its parents have the value TRUE, and 0 otherwise. The top-level operator in ϕ is denoted as V_ϕ. In addition, we add binary variables O and O_ϕ to $\mathcal{B}_\phi$, where O is uniformly distributed and O_ϕ acts as an *and*-operator connecting O and V_ϕ. In Figure 2 the network $\mathcal{B}_\phi$ is shown for the formula $\neg(x_1 \vee x_2) \wedge \neg x_3$. Note that the above network $\mathcal{B}_\phi$ can be constructed from ϕ in polynomial time.

Now, for any particular truth assignment $\mathbf{x}$ to the set of all propositional variables $\mathbf{X}$ in the formula ϕ we have that the probability of the value TRUE of V_ϕ, given the joint value assignment to the stochastic variables matching that truth assignment, equals 1 if $\mathbf{x}$ satisfies ϕ, and 0 if $\mathbf{x}$ does not satisfy ϕ. Likewise, $\Pr(O_\phi = \text{TRUE} \mid \mathbf{x}, O)$ equals 1 if $\mathbf{x}$ satisfies ϕ *and* $O = \text{TRUE}$, and 0 otherwise. We define $\Pr_{\mathrm{O_p}} = \{O, O_\phi\}$ and $\Pr_{\mathrm{O_a}} = \{\Pr(O = \text{FALSE}) = 1, \Pr(O_\phi = \text{FALSE}) = 1\}$. Thus, the probability distribution $\Pr_{\mathrm{O_a}}$ is deterministic and corresponds to the observation $O = \text{FALSE}$ and $O_\phi = \text{TRUE}$.

Theorem 2. ERROR-COMPUTATION (SUM) *is* NP*-hard*

Proof. We show that if we can compute $D_{KL}(\Pr_{\mathrm{O_p}}, \Pr_{\mathrm{O_a}})$ in polynomial time, we can decide SATISFIABILITY. Observe that the prior probability of $V_\phi = \text{TRUE}$ is $\frac{\#_\phi}{2^n}$, where $\#_\phi$ is the number of satisfying truth assignments of the set of propositional variables $\mathbf{X}$. If ϕ is not satisfiable, then $\Pr(V_\phi = \text{TRUE}) = 0$, and thus also $\Pr(O_\phi = \text{TRUE}) = 0$. We then have that $D_{KL}(\Pr_{\mathrm{O_p}}, \Pr_{\mathrm{O_a}}) = 0$ as $\Pr_{\mathrm{O_p}}$ and $\Pr_{\mathrm{O_a}}$ are identical distributions. If, on the other hand, ϕ *is* satisfiable, then $\Pr(V_\phi = \text{TRUE}) > 0$, hence $\Pr(O_\phi = \text{TRUE}) > 0$ and thus $D_{KL}(\Pr_{\mathrm{O_p}}, \Pr_{\mathrm{O_a}}) > 0$. Thus, if we can compute $D_{KL}(\Pr_{\mathrm{O_p}}, \Pr_{\mathrm{O_a}})$ in polynomial time, we are able to decide SATISFIABILITY, hence ERROR-COMPUTATION (SUM) is NP-hard. □

However, computing the Hamming distance D_H between two joint value assignments can be done in polynomial time by the following simple (and trivially polynomial-time) algorithm, where we

assume that $\mathbf{O} = \{O_1, \ldots, O_n\}$:

COMPUTE-HAMMING($\mathbf{o_p}, \mathbf{o_a}$)

```
1: d := 0
2: for i = 1 to n do
3:    if o_{p_i} ≠ o_{a_i} then
4:       d := d + 1
5:    end if
6: end for
7: return d
```

2.3 Hypothesis-Updating

For hypothesis updating, apart from the distinction between the SUM and MAX variants, we also make a distinction between the error-minimization (PRED) and observation-explaining (EXPL) variants. We thus have four variants of the HYPOTHESIS-UPDATING problem. Note that the actual observation is not included in the input, and that–in general–neither $(\Pr_O, \epsilon_p)$ nor $(\mathbf{o_p}, \epsilon_p)$ uniquely identifies $\mathbf{o_a}$ (as multiple actual observations may yield the same Kullback-Leibler or Hamming distance from the predicted observations).

HYPOTHESIS-UPDATING (SUM, PRED)
Instance: A Bayesian network $\mathcal{B} = (\mathbf{G}_\mathcal{B}, \Gamma)$, where $\mathbf{V}(\mathbf{G}_\mathcal{B})$ is partitioned into a set of observed nodes $\mathbf{H}$, a set of prediction nodes $\mathbf{O}$, and a set of intermediate nodes $\mathbf{I}$; a probability distribution $\Pr_{H_c}$ over the hypothesis variables $\mathbf{H}$, denoting the candidate hypothesis; a probability distribution $\Pr_{O_p}$ over the observation variables $\mathbf{O}$, denoting the prediction; and the Kullback-Leibler distance $D_{KL}(\Pr_{O_p}, \mathbf{o_a}) = \epsilon_p$ between the predicted observation $\Pr_O$ and the actual observation $\mathbf{o_a}$, denoting the prediction error.
Output: An updated probability distribution $\Pr_{H_u}$ such that $D_{KL}(\text{PREDICTION}(\Pr_{Hu}), \mathbf{o_a}) < \epsilon_p$.

HYPOTHESIS-UPDATING (MAX, PRED)
Instance: A Bayesian network $\mathcal{B} = (\mathbf{G}_\mathcal{B}, \Gamma)$, where $\mathbf{V}(\mathbf{G}_\mathcal{B})$ is partitioned into a set of observed nodes $\mathbf{H}$, a set of prediction nodes $\mathbf{O}$, and a set of intermediate nodes $\mathbf{I}$; a joint value assignment $\mathbf{h_c}$ to $\mathbf{H}$, denoting the candidate hypothesis, a joint value assignment $\mathbf{o_p}$ to the observation variables $\mathbf{O}$, denoting the prediction; and the Hamming distance $D_H(\mathbf{o_p}, \mathbf{o_a}) = \epsilon_p$ between the predicted observation $\mathbf{o_p}$ and the actual observation $\mathbf{o_a}$, denoting the prediction error.
Output: An updated hypothesis $\mathbf{h_u}$ such that $D_H(\text{PREDICTION}(\mathbf{h_u}), \mathbf{o_a}) < \epsilon_p$.

HYPOTHESIS-UPDATING (SUM, EXPL)
Instance: A Bayesian network $\mathcal{B} = (\mathbf{G}_\mathcal{B}, \Gamma)$, where $\mathbf{V}(\mathbf{G}_\mathcal{B})$ is partitioned into a set of observed nodes $\mathbf{H}$, a set of prediction nodes $\mathbf{O}$, and a set of intermediate nodes $\mathbf{I}$; a probability distribution $\Pr_{H_c}$ over the hypothesis variables $\mathbf{H}$, denoting the candidate hypothesis; a probability distribution $\Pr_{O_p}$ over the observation variables $\mathbf{O}$, denoting the prediction; and the

Kullback-Leibler distance $D_{KL}(\Pr_{O_p}, \mathbf{o_a}) = \epsilon_p$ between the predicted observation $\Pr_O$ and the actual observation $\mathbf{o_a}$, denoting the prediction error.
Output: $\Pr_{H_u}$, i.e., the probability distribution over the hypothesis nodes $\mathbf{H}$.

HYPOTHESIS-UPDATING (MAX, EXPL)
Instance: A Bayesian network $\mathcal{B} = (\mathbf{G}_\mathcal{B}, \Gamma)$, where $\mathbf{V}(\mathbf{G}_\mathcal{B})$ is partitioned into a set of observed nodes $\mathbf{H}$, a set of prediction nodes $\mathbf{O}$, and a set of intermediate nodes $\mathbf{I}$; a joint value assignment $\mathbf{h_c}$ to $\mathbf{H}$, denoting the candidate hypothesis, a joint value assignment $\mathbf{o_p}$ to the observation variables $\mathbf{O}$, denoting the prediction; and the Hamming distance $D_H(\mathbf{o_p}, \mathbf{o_a}) = \epsilon_p$ between the predicted observation $\mathbf{o_p}$ and the actual observation $\mathbf{o_a}$, denoting the prediction error.
Output: $\text{argmax}_{\mathbf{h_u}} \Pr(\mathbf{h_u} \mid \mathbf{o_a})$, i.e., the most probable joint value assignment $\mathbf{h_u}$ to the prediction nodes $\mathbf{H}$ given the observation $\mathbf{o_a}$, or $\perp$ if $\Pr(\mathbf{h_u} \mid \mathbf{o_a}) = 0$ for every joint value assignment $\mathbf{h_u}$ to $\mathbf{H}$.

2.3.1 Error-minimization variants

First, we will prove that HYPOTHESIS-UPDATING (MAX, PRED) is NP-hard, and then proceed to show that HYPOTHESIS-UPDATING (SUM, PRED) is NP-hard using a slight modification of the proof construct. We reduce HYPOTHESIS-UPDATING (MAX, PRED) from MAJSAT, defined as follows:

MAJSAT
Instance: A Boolean formula ϕ with n variables.
Question: Does the majority of truth assignments to ϕ satisfy ϕ?

MAJSAT is NP-hard, in fact, is strictly harder than any problem in NP under common complexity-theoretic assumptions (Littman, Goldsmith, & Mundhenk, 1998).

First of all, we define the Boolean formula $\psi = \phi \wedge x_{n+1}$, introducing an additional variable x_{n+1}. We construct a Bayesian network $\mathcal{B}_\psi$ from ψ in a similar fashion as above. For each propositional variable x_i in ψ, a binary stochastic variable X_i is added to $\mathcal{B}_\psi$, with possible values TRUE and FALSE and a uniform probability distribution. For each logical operator in ψ, an additional binary variable in $\mathcal{B}_\psi$ is introduced, whose parents are the variables that correspond to the input of the operator, and whose conditional probability table is equal to the truth table of that operator. The top-level operator in ψ is denoted as V_ψ; note that V_ψ connects both the top-level operator in ϕ and x_{n+1}, mimicking an *and*-operator. In Figure 3 the network $\mathcal{B}_\psi$ is shown for $\psi = (\neg(x_1 \vee x_2) \wedge \neg x_3) \wedge x_4$ for the formula $\phi = \neg(x_1 \vee x_2) \wedge \neg x_3$.

Now, for any particular truth assignment $\mathbf{x}$ to the set of all propositional variables $\mathbf{X}$ in the formula ϕ we have that the probability of the value TRUE of V_ψ, given the joint value assignment to the stochastic variables matching that truth assignment, equals 1 if $\mathbf{x}$ satisfies ϕ *and* X_{n+1} is set to TRUE, and 0 if $\mathbf{x}$ does not satisfy ϕ *or* X_{n+1} is set to FALSE. Without any given joint value assignment, the prior probability of V_ψ is $\frac{\#_\phi}{2^{n+1}}$, where $\#_\phi$ is the number of satisfying truth assignments to ϕ. Note that the above network $\mathcal{B}_\psi$ can be constructed from ϕ in polynomial time.

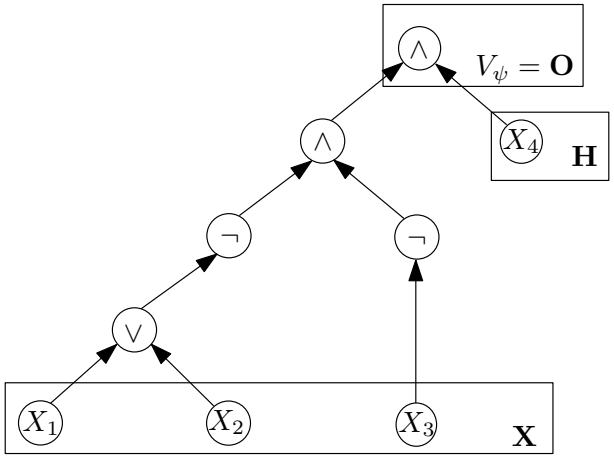

Figure 3: The Bayesian network $\mathcal{B}_\psi$ corresponding to $\psi = (\neg(x_1 \vee x_2) \wedge \neg x_3)\neg x_4$

We define $\mathbf{H} = X_{n+1}$ and $\mathbf{O} = V_\psi$. Furthermore, we define $\mathbf{h_c} = \text{FALSE}$ and $\mathbf{o_a} = \text{TRUE}$. Note that $\mathbf{o_p} = \text{PREDICTION}(\mathbf{h_c}) = \text{argmax}_{\mathbf{O}} \Pr(\mathbf{O} \mid \mathbf{h_c}) = \text{FALSE}$ and thus $D_H(\mathbf{o_p}, \mathbf{o_a}) = 1$.

Theorem 3. HYPOTHESIS-UPDATING (MAX, PRED) *is* NP*-hard*

Proof. We will prove that we can find an updated hypothesis $\mathbf{h_u}$ such that $D_H(\text{PREDICTION}(\mathbf{h_u}), \mathbf{o_a}) < 1$ if and only if ϕ is a YES-instance of MAJSAT; given that MAJSAT is NP-hard, so follows NP-hardness of HYPOTHESIS-UPDATING (MAX, PRED). Assume that ϕ is such a YES-instance, then the majority of truth assignments to ϕ satisfy ϕ. We set $\mathbf{h_u} = \text{TRUE}$. As a result, we have that $\Pr(V_\psi = \text{TRUE} \mid X_{n+1} = \text{TRUE}) > \frac{1}{2}$ and thus that $\mathbf{o_p} = \text{PREDICTION}(\mathbf{h_u}) = \text{argmax}_{\mathbf{O}} \Pr(\mathbf{O} \mid \mathbf{h_u}) = \text{TRUE}$ and thus $D_H(\mathbf{o_p}, \mathbf{o_a}) = 0$. Now assume that $D_H(\mathbf{o_p}, \mathbf{o_a}) < 1$ (implying that $D_H(\mathbf{o_p}, \mathbf{o_a}) = 0$) and thus that $\mathbf{o_p} = \text{PREDICTION}(\mathbf{h_u}) = \text{argmax}_{\mathbf{O}} \Pr(\mathbf{O} \mid \mathbf{h_u}) = \text{TRUE}$. This can only be the case if $\mathbf{h_u} = \text{TRUE}$ and the majority of truth assignments to ϕ satisfy ϕ. This proves NP-hardness of HYPOTHESIS-UPDATING (MAX, PRED). □

Note that HYPOTHESIS-UPDATING (MAX, PRED) is NP-hard, even if *both* the hypothesis and the prediction consist of a *single, binary* variable.

We can augment this proof construct to obtain NP-hardness of HYPOTHESIS-UPDATING (SUM, PRED) as well. To this end, we add an additional *scale* variable S_δ, with V_ψ as its only parent, and with the following conditional probability distribution:

$$\Pr(S_\delta = \text{TRUE} \mid V_\psi) = \begin{cases} 1 & \text{if } V_\psi = \text{TRUE} \\ 1 - \delta & \text{if } V_\psi = \text{FALSE} \end{cases}$$

Here, δ is an arbitrary small (yet computationable in polynomial time in the size of $\mathcal{B}_\psi$) number. We set $\Pr_{H_c}$ to $\Pr(H_c = \text{FALSE}) = 1$, and define $\mathbf{O} = S_\delta$ with $\Pr_{O_a}$ as $\Pr(O_a = \text{TRUE}) = 1$. Note that

$\Pr_{O_p} = \textsc{Prediction}(\Pr_{H_c}) = \Pr(O_p = \textsc{false}) = 1$ and therefore that $\epsilon_p = D_{KL}(\Pr_{O_p}, \Pr_{O_a}) = 1 \cdot \ln \frac{1}{1-\delta} + 0 \cdot \ln \frac{0}{\delta} = \ln \frac{1}{1-\delta}$.

Furthermore, we reduce Hypothesis-Updating (SUM, PRED) from Satisfiability, rather than Majsat.

Theorem 4. Hypothesis-Updating (SUM, PRED) *is* NP*-hard*

Proof. Observe that we can bring $\epsilon_p = D_{KL}(\Pr_{O_p}, \Pr_{O_a})$ arbitrarily close to 0 by making δ sufficiently small. We will show that if we can find an updated probability distribution $\Pr_{H_u}$ such that $D_{KL}(\textsc{Prediction}(\Pr_{H_u}), \Pr_{O_a}) < \ln \frac{1}{1-\delta}$, then we can decide Satisfiability. Note that if ϕ is unsatisfiable, then $\Pr(V_\psi = \textsc{true}) = 0$ and thus $\Pr(S_\delta = \textsc{true}) = 1-\delta$ and $D_{KL}(\textsc{Prediction}(\Pr_{H_u}), \Pr_{O_a}) = \ln \frac{1}{1-\delta}$ for an arbitrary choice of probability distribution $\Pr_{H_u}$. However, if ϕ is satisfiable, then we can lower the Kullback-Leibler divergence by setting $\Pr(x_{n+1} = \textsc{true}) > 0$. In that case, we have that $\Pr(V_\psi = \textsc{true}) > 0$ and thus $D_{KL}(\Pr_{O_p}, \Pr_{O_a}) < \ln \frac{1}{1-\delta}$.

Likewise, if $D_{KL}(\textsc{Prediction}(\Pr_{H_u}), \Pr_{O_a}) < \ln \frac{1}{1-\delta}$, then $\Pr(V_\psi = \textsc{true}) > 0$ and thus there is a satisfying truth assignment to ϕ. This proves NP-hardness of Hypothesis-Updating (SUM, PRED). □

Note that this NP-hardness proof holds for an *arbitrarily small* improvement of the prediction error, i.e., there cannot exist a polynomial-time algorithm that can guarantee to update the hypothesis such that the prediction error decreases with a factor ϵ for *any* value of $\epsilon > 0$, unless P = NP.

2.3.2 Observation-explaining variants

Allthough $\mathbf{o_a}$ is not available in the input, we will show that Hypothesis-Updating (MAX, EXPL) is NP-hard, even *if it were*. Thus, the intractability can not (only) be ascribed to the recovery of $\mathbf{o_a}$ from the prediction error and the candidate hypothesis. Furthermore, we show that having the candidate hypothesis and prediction error readily available does not render the inference problem tractable.

The construction of the reduction of these variants is identical to these of the error-minimization variants, but now we observe $\mathbf{o_a} = \textsc{false}$ rather than $\mathbf{o_a} = \textsc{true}$. Furthermore, we redefine the prior probability of the variable X_{n+1} as $\Pr(X_{n+1} = \textsc{true}) = \frac{1}{2} + \frac{1}{2^n}$, and we reduce from Satisfiability.

Theorem 5. Hypothesis-Updating *(MAX, EXPL) and* Hypothesis-Updating *(SUM, EXPL) are* NP*-hard*

Proof. For Hypothesis-Updating (MAX, EXPL) we have that $\mathbf{h_c} = \textsc{false}$ and $\mathbf{o_p} = \textsc{Prediction}(\mathbf{h_c}) = \text{argmax}_{\mathbf{O}} \Pr(\mathbf{O} \mid \mathbf{h_c}) = \textsc{false}$, and observe $\mathbf{o_a} = \textsc{false}$. Correspondingly, for Hypothesis-Updating (SUM, EXPL) we have that $\Pr_{H_c}(H_c = \textsc{false}) = 1$, $\Pr_{O_p}(O_p = \textsc{false}) = 1$, and observe $\Pr_{O_p}(O_p = \textsc{false}) = 1$.

Now, $\mathbf{h_u} = \text{argmax}_{\mathbf{H}} \Pr(\mathbf{H} \mid \mathbf{o_a})$. If ϕ is unsatisfiable, then $\Pr(V_\psi = \textsc{true}) = 0$ and $\Pr(\mathbf{h_u} = \textsc{true} \mid \mathbf{o_a}) = \frac{1}{2} + \frac{1}{2^n}$ and thus $\text{argmax}_{\mathbf{H}} \Pr(\mathbf{H} \mid \mathbf{o_a}) = \textsc{true}$ because of the slight bias of the prior

distribution of $\Pr(X_{n+1})$ towards TRUE. However, if ϕ is satisfiable, then $\Pr(\mathbf{h_u} = \text{TRUE} \mid \mathbf{o_a}) < \frac{1}{2}$ and thus $\text{argmax}_{\mathbf{H}} \Pr(\mathbf{H} \mid \mathbf{o_a}) = \text{FALSE}$. Thus, if we can solve HYPOTHESIS-UPDATING (MAX, EXPL) or HYPOTHESIS-UPDATING (SUM, EXPL) we can also decide SATISFIABILITY. □

Note that this result holds *even if there is* **no** *prediction error* at all.

3 Fixed parameter tractability results

The fixed parameter tractability results here correspond to the known fpt results for INFERENCE and MAP. It is known that INFERENCE is fixed parameter tractable for the parameters tw (the *treewidth* of the network) and c (the maximal *cardinality* of the variables in the network), resulting in a $\mathcal{O}(c^{\text{tw}} \cdot n)$ running time (Lauritzen & Spiegelhalter, 1988). Thus, if we compute posterior distributions over at most $d = c^{|\mathbf{O}|}$ possible observations, respectively $c^{|\mathbf{H}|}$ possible hypotheses, we obtain a $\mathcal{O}(d \cdot c^{\text{tw}} \cdot n)$ algorithm for solving the SUM problem variants.

If we can solve the SUM variants tractably, we can also solve the MAX variants tractably: just select the joint value assignment with the highest probability, which can be done in $\mathcal{O}(d)$ time. In addition, we have that MAP is fixed parameter tractable for the parameters tw (the *treewidth* of the network), c (the maximal *cardinality* of the variables in the network), and $1-p$ (the probability of the most probable explanation), resulting in a $\mathcal{O}(c^{\frac{\log(p)}{\log(1-p)}} \cdot n)$ running time (Kwisthout, 2011). This yields the following fpt results.

Result 6. PREDICTION *(SUM) is fixed parameter tractable for* $\{c, \text{tw}, |\mathbf{O}|\}$

Result 7. PREDICTION *(MAX) is fixed parameter tractable for* $\{c, \text{tw}, 1-p\}$ *and for* $\{c, \text{tw}, |\mathbf{O}|\}$

Result 8. ERROR-COMPUTATION *(SUM) is fixed parameter tractable for* $\{c, \text{tw}, |\mathbf{O}|\}$

Result 9. HYPOTHESIS-UPDATING *(SUM) is fixed parameter tractable for* $\{c, \text{tw}, |\mathbf{H}|\}$

Result 10. HYPOTHESIS-UPDATING *(MAX) is fixed parameter tractable for* $\{c, \text{tw}, 1-p\}$ *and for* $\{c, \text{tw}, |\mathbf{H}|\}$

References

Cooper, G. F. (1990). The computational complexity of probabilistic inference using Bayesian belief networks. *Artificial Intelligence*, *42*(2), 393–405.

Dagum, P., & Luby, M. (1993). Approximating probabilistic inference in Bayesian belief networks is NP-hard. *Artificial Intelligence*, *60*(1), 141–153.

De Campos, C. P. (2011). New complexity results for MAP in Bayesian networks. In T. Walsh (Ed.), *Proceedings of the twenty-second international joint conference on artificial intelligence* (p. 2100-2106).

Downey, R. G., & Fellows, M. R. (1999). *Parameterized complexity.* Springer Verlag, Berlin.

Flum, G., & Grohe, M. (2006). *Parameterized Complexity Theory.* Berlin: Springer.

Friston, K. (2002). Functional integration and inference in the brain. *Progress in Neurobiology*, *590*, 1-31.

Garey, M. R., & Johnson, D. S. (1979). *Computers and intractability. a guide to the theory of NP-completeness*. W. H. Freeman and Co., San Francisco, CA.

Hamming, R. (1950). Error detecting and error correcting codes. *Bell System Technical Journal*, *29*(2), 147160.

Jensen, F. V., & Nielsen, T. D. (2007). *Bayesian networks and decision graphs* (Second ed.). Springer Verlag, New York, NY.

Kilner, J. M., Friston, K. J., & Frith, C. D. (2007). The mirror-neuron system: a Bayesian perspective. *Neuroreport*, *18*, 619-623.

Knill, D., & Pouget, A. (2004). The bayesian brain: the role of uncertainty in neural coding and computation. *Trends in Neuroscience*, *27*(12), 712-719.

Koller, D., & Friedman, N. (2009). *Probabilistic graphical models: principles and techniques*. MIT press, Cambridge, MA.

Kullback, S., & Leibler, R. A. (1951). On information and sufficiency. *The Annals of Mathematical Statistics*, *22*, 79–86.

Kwisthout, J. (2009). *The computational complexity of probabilistic networks*. Unpublished doctoral dissertation, Faculty of Science, Utrecht University, The Netherlands.

Kwisthout, J. (2011). Most probable explanations in Bayesian networks: Complexity and tractability. *International Journal of Approximate Reasoning*, *52*(9), 1452 - 1469.

Lauritzen, S. L., & Spiegelhalter, D. J. (1988). Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society*, *50*(2), 157–224.

Littman, M. L., Goldsmith, J., & Mundhenk, M. (1998). The computational complexity of probabilistic planning. *Journal of Artificial Intelligence Research*, *9*, 1–36.

Park, J. D., & Darwiche, A. (2004). Complexity results and approximation settings for MAP explanations. *Journal of Artificial Intelligence Research*, *21*, 101–133.

Pearl, J. (1988). *Probabilistic reasoning in intelligent systems: Networks of plausible inference*. Morgan Kaufmann, Palo Alto, CA.

Robertson, N., & Seymour, P. D. (1986). Graph minors II: Algorithmic aspects of tree-width. *Journal of Algorithms*, *7*, 309–322.