Take Care when Tuning Bayesian Networks

Janneke H. Bolt[1,2], Arjen Hommersom[2], and Silja Renooij[1]

[1] Department of Information and Computing Sciences, Utrecht University, The Netherlands {j.h.bolt,s.renooij}@uu.nl
[2] Open University of the Netherlands, Department of Computer Science, Heerlen, The Netherlands
{janneke.bolt,arjen.hommersom}@ou.nl

Abstract. [3] Parameter tuning in Bayesian networks aims to fine-tune certain output probabilities to ensure that they match the expected outcome based upon domain expertise. Various tuning heuristics exist that can change different sets of parameters to arrive at the requested change in output. Previous research has evaluated the performance of these heuristics by comparing networks before and after tuning. In this paper, we study the performance of tuning heuristics by qualitatively inspecting the changes they incur compared to a ground truth distribution. We show for example networks with a fixed structure that tuning an output probability towards its ground truth value does not necessarily mean that a network as a whole more closely fits the ground truth. In fact, this fit can become worse; tuning of Bayesian networks should therefore be done with care.

Keywords: Bayesian networks, parameter tuning, ground truth comparison

1 Introduction

A Bayesian network is a concise model of a joint probability distribution [13] consisting of an acyclic directed graph and a set of (conditional) probabilities. The graphical representation of the modelled distribution makes these networks highly intuitive and therefore very suitable for use in for example clinical settings. In the final stages of Bayesian network construction, when the model and its probability parameters have been assessed or learned, the network can be fine-tuned to ensure that the outcome of certain probability queries match the expected outcome based on domain expertise [8]. This is, for the acceptance of a decision support system by its end users, a crucial property.

Network tuning consists of applying a change to one or more parameters from the network's conditional probability tables (CPTs) to effectuate the desired change in an output query of interest. In fine-tuning a Bayesian network, we can opt to change only a single parameter (one-way tuning) or a combination of parameters (n-way tuning). Typically there are multiple choices of parameter adaptations to obtain the desired result. Previous research has considered different interpretations of the optimal choice, focusing on the total absolute change in parameters [2,3,8], the relative change in parameters [8], the log-odds change in parameters [4,6,8], and the uncertainty-weighted change in parameters [2]. Tuning heuristics employing these different interpretations have been evaluated by comparing (joint) distributions before and after tuning, using measures such as the CD-distance [7] and the KL-divergence [11]. All those evaluations, however, lack a comparison with the underlying ground truth distribution.

In this paper we study the change in the quality of the network due to parameter tuning, where we define the quality of the network first of all in terms of how well the network's joint distribution fits the underlying ground truth distribution. To the best of our knowledge, there is no research that has investigated whether parameter tuning actually improves the network as a whole. For real-life problems, the fit to a ground truth distribution cannot be studied, since the ground truth is unknown. In this paper, we therefore simulate a ground truth distribution from which we generate data. We subsequently learn a Bayesian network from this data and tune parameters in the network to enforce an output query of interest to match the associated probability from the ground truth distribution[4]. We then compare the KL-divergence between the network's joint distribution and the ground truth distribution before and after tuning to assess to what extent tuning improves the quality of the network as a whole. In addition, we consider the effect of tuning on the quality of individual queries. We do this by evaluating whether tuning one output query of interest towards the ground truth ensures that other output queries also improve their fit to the ground truth distribution. We observe that an improvement of a network by tuning one of its outputs with the investigated tuning heuristics, judged by these criteria, often fails.

After some preliminaries in Section 2, we describe the details of our experimental study in Section 3. The results of our study are presented and discussed in Section 4. The paper is concluded with Section 5.

2 Preliminaries

In this section we provide the necessary background on Bayesian networks, tuning heuristics and KL-divergence. The notation used in this work is also introduced.

[4] Note that in earlier research we employed a similar approach to compare tuning heuristics involving different levels of uncertainty [2], however, in that study we were concerned with the likeliness of the proposed changes and we did not compare networks with ground truth distributions.

2.1 Bayesian Networks

A Bayesian network is a concise representation of a joint probability distribution Pr over a set of discrete stochastic variables that consists of an acyclic directed graph and a set of (conditional) probability tables (CPTs) [10,13]. In the following we will indicate single variables by uppercase letters (V); lowercase letters (v_i) indicate a value assignment to one of V's values in $\mathbf{\Omega}(V)$. In this paper we restrict ourselves to binary variables, writing v and $\overline{v}$ to denote the two possible instantiations of a variable V. For sets of variables and sets of value assignments we will use bold face letters. The nodes of the graph represent the variables of the modelled distribution[5] and the structure of the graph captures, as far as possible, the independences of this distribution according to the d-separation criterion. The joint distribution now factorizes over the CPTs associated with the nodes in the graph. Each CPT-row consists of the conditional probabilities $\Pr(v_i|\pi_j^V)$, $v_i \in \mathbf{\Omega}(V)$, of a variable V given a specific value assignment π_j^V to its parents in the graph. Note that the parameter values of a CPT-row sum to 1. The probability of a joint value assignment $\mathbf{v}$ to $\mathbf{V}$ now equals

$$\Pr(\mathbf{v}) = \prod_{V \in \mathbf{V}} \Pr(v_i|\pi_j^V) \tag{1}$$

where v_i and π_j^V are compatible with $\mathbf{v}$. Using this formula, any probability of interest from the distribution can be computed.

The CPT-entries required for a Bayesian network are more generally referred to as network *parameters*. To quantify the network, estimates for these parameters are obtained from data and/or through expert elicitation.

2.2 Network Tuning

In tuning, our goal is to adapt network parameters in order to achieve that some output query probability, say $\Pr(h|\mathbf{e})$, attains a desired value, say q_t. To this end, choices have to be made about which parameters to adapt and how much change to apply to them. In order to decide on parameter changes that achieve our goal different tuning heuristics exist [1–3,6]. Definition 1 below summarises the different heuristics that we consider in this paper. The most common heuristic, gradient tuning, aims to *minimise* the parameter changes and can always be applied. The other three heuristics assume that the higher-order uncertainty in the parameter estimates is explicitly captured by a known distribution; tuning then involves this parameter uncertainty aiming to find the *most likely* parameter changes. A common assumption is that the uncertainty in the estimated parameter of a binary variable follows a beta-distribution [9], which is fully characterized by its shape parameters α and β and has a variance of $\mathrm{Var}[x] = \frac{\alpha \cdot \beta}{(\alpha+\beta)^2 \cdot (\alpha+\beta+1)}$.

In all heuristics, the choices of the parameters and their relative adaptations are based on the partial derivatives of the function $f_{\Pr(h|\mathbf{e})}(\mathbf{x})$ that expresses

[5] We will use the terms nodes and variables interchangeably.

probability $\Pr(h \mid \mathbf{e})$ in terms of the network parameters $\mathbf{x}$ involved in its computation [5], either or not weighted in varying degrees with the variance in the distribution that the parameters are assumed to follow [2].

Definition 1 (gradient/$\sqrt[3]{\textsc{var}}$-weighted/SD-weighted/VAR-weighted n-way tuning). *Let* $\Pr(h \mid \mathbf{e})$ *be a query response of a Bayesian network that is to be tuned to* q_t*. Let* $\nabla_i f(\boldsymbol{x}^o) = \partial / \partial x_i \; f_{\Pr(h \mid \mathbf{e})}(\boldsymbol{x}^o)$ *be the partial derivative of* $f_{\Pr(h \mid \mathbf{e})}(\mathbf{x})$ *with respect to* $x_i \in \mathbf{x}$ *at* $\mathbf{x} = \boldsymbol{x}^o$*, where* $\boldsymbol{x}^o$ *represents the original values of the parameters. Then in n-way (weighted) tuning*

- *we select the set* $\mathbf{x_s} = \{x_1, \ldots, x_n\}$ *(not necessarily ordered) of* $n \geq 1$ *parameters in* $\mathbf{x}$ *with top-n highest values* $|\nabla_i f(\boldsymbol{x}^o) \cdot w_i|$*, where* w_i *is a weight as defined below;*
- *for* $n > 1$*, we express all* $x_i \in \{x_2 \ldots x_n\}$ *of* $f_{\Pr(h \mid \mathbf{e})}(\mathbf{x_s})$ *in* x_1*:*

$$x_i = \frac{\nabla_i f(\boldsymbol{x}^o) \cdot w_i}{\nabla_1 f(\boldsymbol{x}^o) \cdot w_1} \cdot (x_1 - x_1^o) + x_i^o$$

  where x_i^o *is the original value of the parameter that is replaced by* x_i*;*
- *find the tuned parameter values* $\mathbf{x_s^*}$ *by solving for* x_1

$$f_{\Pr(h \mid \mathbf{e})}(x_1) = q_t$$

The different heuristics now differ only in the weights used:

- $w_i = 1$ *in gradient tuning*
- $w_i = \sqrt[3]{\mathrm{Var}[x_i]}$ *in* $\sqrt[3]{\textsc{var}}$*-weighted tuning*
- $w_i = \sqrt[2]{\mathrm{Var}[x_i]}$ *in* SD*-weighted tuning*
- $w_i = \mathrm{Var}[x_i]$ *in* VAR*-weighted tuning*

With $w_i = 1$, tuning is based on the gradient alone; the other heuristics include uncertainty by assigning, respectively, an increasing weight to the variance in the parameters. Note that both before and after tuning, the actual network parameters are point probabilities.

2.3 Kullback-Leibler Divergence

A common measure for comparing two probability distributions Pr and Pr′ over the same set of variables $\mathbf{V}$ is the Kullback-Leibler (KL) divergence [11].

Definition 2 (Kullback-Leibler divergence). *Let* Pr *and* Pr′ *be probability distributions over the same set of variables* $\mathbf{V}$*, then the Kullback-Leibler divergence between* Pr *and* Pr′ *equals*

$$\mathrm{KL}(\Pr, \Pr') = \sum_{\mathbf{v} \in \mathbf{V}} \Pr(\mathbf{v}) \cdot \ln \frac{\Pr(\mathbf{v})}{\Pr'(\mathbf{v})}$$

The KL-divergence is positive and equals zero if and only if $\Pr = \Pr'$; it is often considered to capture how much an approximating probability distribution Pr′ is different from a true distribution Pr. Note, however, that since the measure is asymmetric, it is not a proper distance measure.

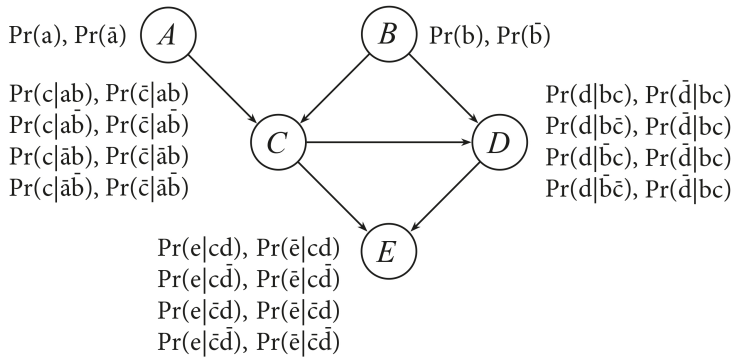

Fig. 1. The graph of the Bayesian networks of the experiments with the associated probabilities required for the CPT-entries.

3 Experimental Study

Existing research on Bayesian network tuning has focused on comparing properties of the network before and after tuning. Such comparisons, however, cannot reveal whether tuning improves a network compared to the distribution that is modelled. In this paper we use synthetic data to create Bayesian networks, thereby having an underlying ground truth distribution available. As such we can evaluate whether tuning an output probability of interest in the learned network towards its ground truth value increases the quality of the tuned network compared to the ground truth distribution. More specifically, we aim to answer the following research questions:

- Is the joint probability distribution defined by a Bayesian network closer to the ground truth distribution after tuning compared to before tuning?
- Are output query probabilities, other than the tuned one, closer to their ground truth values after tuning compared to before tuning?

Note that, since a network is mostly used for inferring query outputs this second question is an important one. To answer the first question we evaluate for all networks in our experiments whether or not $\mathrm{KL}(\Pr_G, \Pr^t) < \mathrm{KL}(\Pr_G, \Pr)$, that is, if the distribution $\Pr^t$ of the tuned network is closer to the ground truth distribution $\Pr_G$ than the distribution $\Pr$ prior to tuning[6] . To answer the second question we similarly evaluate qualitatively if query probabilities computed from the tuned networks are closer to their ground truth value than the same probabilities computed prior to tuning. We thereby investigate the performances of four different tuning heuristics, all of which assign a different weight to the uncertainty in the parameter estimates of the learned networks.

[6] Because of KL's asymmetry, we also computed whether or not $\mathrm{KL}(\Pr^t, \Pr_G) < \mathrm{KL}(\Pr, \Pr_G)$. This yielded comparable results, leading to the same conclusions.

3.1 Networks and Data

To create synthetic data that represents the ground truth, and Bayesian networks with parameters learned from those data, we follow the same procedure as used in earlier research [2].

The ground truth distribution is defined by a Bayesian network with the same structure as the network that will be subjected to tuning. This network has five binary variables and a graph as depicted in Fig. 1. This figure in addition shows which parameters are required for the network's specification.

1. We construct a ground truth distribution by generating random values for the parameters of the network in Fig. 1[7]. We construct two types of *ground truth network*: one where the parameters are generated from a uniform distribution, and one where the parameters are biased towards 0 and 1. The latter is done by drawing from the distribution with a probability density function defined by $(x - 0.5)^2$.
2. From each ground truth network, we generate 75 data points, i.e. joint value combinations of the variables, by means of forward sampling.
3. These 75 generated data points are then used to estimate the parameters of a *learned* network that will be subjected to tuning. The low number of data points ensures sufficient higher-order uncertainty about the estimates. We apply standard Laplace smoothing to avoid zero probabilities.

In total we generate 1000 networks in which the ground truth parameters are drawn from a uniform distribution and 1000 networks in which the ground truth parameters are biased towards 0 and 1.

Some of the tuning heuristics that we will use exploit the uncertainty in the parameter estimates. We model that uncertainty in each parameter estimate by using a Beta distribution with parameters α and β based on counts in the data. The α of a parameter $\Pr(x|\boldsymbol{y})$ is the number of data points matching $x\boldsymbol{y}$ plus 1 and the β is the number of data points matching $\bar{x}\boldsymbol{y}$ plus 1. Similar to Laplace smoothing, a count of one is added to avoid zero probabilities.

3.2 Tuning and Evaluation

The query output probability that we will tune is the probability $\Pr(d|e)$ that is computed from the network as follows (slightly abusing notation):

$$\Pr(d|e) = \frac{\sum_{ABC} \Pr(A) \cdot \Pr(B) \cdot \Pr(C|A,B) \cdot \Pr(d|B,C) \cdot \Pr(e|C,d)}{\sum_{ABCD} \Pr(A) \cdot \Pr(B) \cdot \Pr(C|A,B) \cdot \Pr(D|B,C) \cdot \Pr(e|C,D)}$$

We note that to compute this probability, parameters of all 14 CPT-rows specified in the network are used. Tuning aims to achieve the output $\Pr(d|e) = q_t$, with q_t the value of $\Pr_G(d|e)$ as computed from the ground truth distribution.

[7] We generate values only for the 14 free parameters of the network, that is, one per CPT-row; the complementary parameters are defined by the fact that each CPT-row has to sum to 1.

We consider the additional set of query probabilities $Q = \{\Pr(d|\overline{e}), \Pr(d|a), \Pr(c|e)\}$ with their ground truth values $\Pr_G(d|\overline{e})$, $\Pr_G(d|a)$ and $\Pr_G(c|e)$. This set—with the output $\Pr(d|\overline{e})$ of the target variable that excludes $E = e$, the output $\Pr(d|a)$ of the target variable that does not exclude $E = e$, and the output $\Pr(c|e)$ of another variable conditioned on the same observation $E = e$ as the tuned output—gives a diverse impression of the effect of tuning on other output probabilities of the network.

Note that, since the effect of parameters on an output variable in general decreases with their distance to the output variable [12], also a small graph is apt for answering the research questions.

In our experiments we now perform the following steps for each of the 2000 learned networks:

1. Compute $\mathrm{KL}(\Pr_G, \Pr)$.
2. Compute the query probabilities from set Q, and assess the differences with their ground truth values.
3. Construct 20 tuned networks, by using all 4 types of n-way tuning from Definition 1 with the choice to vary the number of tuning parameters n from 1 up to 5 for each type.
4. For each tuned network, compute $\mathrm{KL}(\Pr_G, \Pr^t)$, and decide if $\mathrm{KL}(\Pr_G, \Pr^t) < \mathrm{KL}(\Pr_G, \Pr)$.
5. For each tuned network, compute the query probabilities from set Q, assess the difference with their ground truth values, and decide if this difference is smaller in the tuned network compared to the learned network.

4 Results and Discussion

Below we present and discuss the results of our experiments that can be found in Tables 1-5. Each table includes results for the 4 types of heuristics (abbreviated to grad, $\sqrt[3]{\text{VAR}}$-W, SD-W, VAR-W) for n-way tuning, $n = 1, \ldots, 5$, both for networks learned from parameters generated from uniform ground truth distributions and from biased ground truth distributions. In the following we will refer to the experiments with these two types of network as the uniform case and the biased case, respectively. We will report the percentages of networks in which we see the different qualitative changes discussed above. Note that in some cases, with the given heuristic and number of tuning parameters, it will be impossible to tune the network since parameter values outside the interval $[0, 1]$ would be needed. We will exclude those networks from the analyses. To give an impression of how many networks are excluded we first provide an overview of these numbers.

Number of (Un)Tuned Networks Table 1 shows the number of networks for which parameter values outside the interval $[0, 1]$ would be needed to achieve $\Pr(d|e) = q_t$; these are the networks that could *not* be tuned with the specified number of tuning parameters and the selected tuning heuristic.

Table 1. Number out of 1000 networks that could not be tuned.

	uniform				biased			
	grad	$\sqrt[3]{\text{VAR}}$-W	SD-W	VAR-W	grad	$\sqrt[3]{\text{VAR}}$-W	SD-W	VAR-W
1-way	73	41	36	121	195	143	134	305
2-way	61	19	9	25	184	133	91	142
3-way	50	22	10	10	177	123	83	74
4-way	45	23	14	1	166	116	87	46
5-way	41	22	15	0	167	110	83	30

Table 2. Percentage of networks with a decreased KL-divergence from the ground truth networks after tuning.

	uniform				biased			
	grad	$\sqrt[3]{\text{VAR}}$-W	SD-W	VAR-W	grad	$\sqrt[3]{\text{VAR}}$-W	SD-W	VAR-W
1-way	35.3	38.4	39.6	37.4	42.6	44.3	46.9	43.3
2-way	45.5	48.6	50.3	47.1	49.3	53.7	55.7	51.4
3-way	52.6	54.8	57.5	56.8	51.9	57.8	58.9	61.0
4-way	54.6	60.5	62.4	62.6	52.3	60.0	63.3	66.6
5-way	55.6	63.0	68.0	68.7	53.3	61.0	64.6	70.3

We observe that at worst, in the uniform case, 121 out of 1000 networks, and in the biased case, 305 out of 1000 networks could not be tuned. Both numbers were found for VAR-weighted weighted tuning with just one tuning parameter.

For the rest of the experiments we provide numbers for just the networks that could be tuned below.

Effect on the KL-divergence Table 2 shows the percentages of networks in which $\text{KL}(\Pr_G, \Pr^t) < \text{KL}(\Pr_G, \Pr)$. In these networks, tuning decreased the KL-divergence from the ground truth networks. Note that after changing one or more of its parameters, a network will model a different distribution. The KL-divergence therefore will most probably change as well. In our experiments the KL-divergences indeed either increased or decreased and never remained the same.

We observe, for all heuristics, that as we tune more parameters, we find an increasing percentage of networks in which the KL-divergence decreases. We moreover observe that all uncertainty weighted heuristics perform better than the gradient heuristic, in the sense that higher percentages of tuned networks better fit the ground truth distribution.

With around 70%, the best result is found given 5-way tuning with the VAR-weighted heuristic, for both the uniform and the biased case. Note that even given this best result, in still roughly 30% of the networks, their KL-divergence from the ground truth network is increased rather than decreased as a result of tuning. In roughly 30% of the networks, tuning thus results in a poorer fit to the ground truth distribution.

Table 3. Percentages of networks in which the distance between the output $\Pr(d|\bar{e})$ and $\Pr_G(d|\bar{e})$ decreased after tuning.

	uniform				biased			
	grad	$\sqrt[3]{\text{VAR-W}}$	SD-W	VAR-W	grad	$\sqrt[3]{\text{VAR-W}}$	SD-W	VAR-W
1-way	42.3	43.8	43.7	41.2	41.6	43.4	44.6	43.9
2-way	41.4	44.1	45.1	44.3	42.8	43.6	43.2	45.2
3-way	45.2	44.8	44.4	45.2	44.7	44.4	46.6	47.3
4-way	45.4	46.8	48.5	44.9	44.2	47.1	47.3	47.9
5-way	46.8	47.3	49.1	46.7	45.6	46.0	47.2	47.3

Effect on $\Pr(d|\bar{e})$ In Table 3 the percentages of networks in which $|\Pr_G(d|\bar{e}) - \Pr^t(d|\bar{e})| < |\Pr_G(d|\bar{e}) - \Pr(d|\bar{e})|$ are given. In these networks, tuning of output probability $\Pr(d|e)$ also improved the fit of output probability $\Pr(d|\bar{e})$ to its ground truth value $\Pr_G(d|\bar{e})$.

Note that the computation of $\Pr(d|\bar{e})$ needs parameters from all CPT-rows. Most probably therefore, the distance between the output $\Pr(d|\bar{e})$ and $\Pr_G(d|\bar{e})$ will change when the network is tuned. In our experiments this distance indeed either increased or decreased and never remained the same.

A striking observation is that all percentages in the table are below 50%. So, for all tuning heuristics and all number of tuning parameters, in more than 50% of the tuned networks the distance between the output $\Pr(d|\bar{e})$ and its ground truth value is *increased* by tuning the network, indicating a poorer fit. With an outcome just below 50% the best result for the uniform case is found given SD-weighted tuning with five variables and for the biased case given VAR-weighted tuning with four variables.

There is a slight trend that, for a given heuristic, the more tuning parameters the better the result and that, at least in the biased case, for a given number of tuning parameters, the weighted heuristics give a better result than the gradient heuristic, but these trends are neither outspoken nor consistent.

Effect on $\Pr(d|a)$ In some networks, $\Pr(d|e)$ is tuned by changing parameters in the CPT of variable E, and/or parameters that condition on $\bar{a}$ in the CPT of variable C. Since these parameters are not used in the computation of $\Pr(d|a)$, in these networks the distance between the output $\Pr(d|a)$ and $\Pr_G(d|a)$ remains equal after tuning. Table 4 shows the percentages of these networks. For the networks in which the distance does change, Table 5 shows the percentages of networks for which this distance decreased.

In contrast to the output $\Pr(d|\bar{e})$ now all percentages are above 50%. Again there is a slight trend that involving parameter uncertainty improves the results. This is a bit more prominent in the biased case. However, in all heuristics, there seems to be a limit to the benefit of adding tuning parameters in tuning $\Pr(d|e)$ for the quality of the output $\Pr(d|a)$. The best result for both the uniform and for the biased case is found given 3-way tuning with the VAR-weighted heuristic, with respectively around 62% and 65% of networks in which $\Pr(d|a)$ improves.

Table 4. Percentages of the networks with an equal distance between the output $\Pr(d|a)$ and $\Pr_G(d|a)$ before and after tuning.

	uniform				biased			
	grad	$\sqrt[3]{\text{VAR-W}}$	SD-W	VAR-W	grad	$\sqrt[3]{\text{VAR-W}}$	SD-W	VAR-W
1-way	22.2	27.4	29.7	35.8	24.1	27.1	29.0	35.7
2-way	5.8	10.1	13.1	20.9	5.9	10.1	14.2	25.1
3-way	1.0	2.2	2.6	4.9	1.2	2.6	4.3	4.6
4-way	0.2	0.4	0.5	1.4	0.4	0.7	1.1	1.1
5-way	0	0	0	0.3	0	0	0.1	0.1

Table 5. Percentage of networks in which the distance between the output $\Pr(d|a)$ and $\Pr_G(d|a)$ decreased after tuning considering the networks in which this distance changed.

	uniform				biased			
	grad	$\sqrt[3]{\text{VAR-W}}$	SD-W	VAR-W	grad	$\sqrt[3]{\text{VAR-W}}$	SD-W	VAR-W
1-way	53.0	56.8	57.1	56.7	53.7	53.5	54.1	53.7
2-way	57.9	59.0	58.8	58.1	58.5	60.0	60.0	58.2
3-way	59.4	60.0	60.0	62.2	59.3	60.0	61.1	64.7
4-way	60.0	60.6	61.1	62.0	58.8	58.8	60.8	64.5
5-way	59.3	60.4	60.6	61.5	59.0	59.2	60.3	63.6

Effect on $\Pr(c|e)$ Table 6, to conclude, shows the percentages of networks for which the distance between the outputs $\Pr(c|e)$ and $\Pr_G(c|e)$ decreased after tuning of output $\Pr(d|e)$.

Note that the computation of $\Pr(c|e)$ needs parameters from all CPT-rows. Most probably therefore, the distance between the outputs $\Pr(c|e)$ and $\Pr_G(c|e)$ will change when the network is tuned. In our experiments this distance indeed either increased or decreased and never remained the same.

We observe that all percentages, except for 1-way and 2-way tuning with the gradient and the variance weighted heuristics in the uniform case, are above 50%. There is a slight trend that for a given heuristic, the more tuning parameters the better the result, and that, for a given number of tuning parameters, the weighted heuristics give a better result than the gradient heuristic, but again these trends are neither outspoken nor consistent. The best results are found with the variance weighted heuristic with 5-way tuning (uniform case) and 4-way tuning (biased case), with around 56% and 63% of networks for which this output improves.

Effect of available data We repeated all the experiments using just 25 out of the 75 data points. These experiments gave comparable results for the percentages of networks in which improvement was observed for the considered criteria. The number of networks that could not be tuned, however, increased. Since these results do not affect our conclusions, we do not present them in more detail.

Table 6. Percentage of networks in which the distance between the output $\Pr(c|e)$ and $\Pr_G(c|e)$ decreased after tuning.

	uniform				biased			
	grad	$\sqrt[3]{\text{VAR-W}}$	SD-W	VAR-W	grad	$\sqrt[3]{\text{VAR-W}}$	SD-W	VAR-W
1-way	48.5	50.9	51.3	48.9	51.2	52.4	52.5	50.5
2-way	48.9	52.0	52.3	49.4	56.9	59.2	57.8	59.4
3-way	52.6	53.5	53.7	53.9	58.9	61.1	61.4	59.3
4-way	53.3	53.9	55.3	55.1	58.8	59.9	59.9	63.4
5-way	53.0	54.4	54.7	56.0	58.6	60.1	60.2	63.0

5 Conclusions and Future Research

In this paper we explored the qualitative effect of parameter tuning on the quality of example Bayesian networks with a given structure, by comparing the networks before and after tuning with the ground truth distribution that is modelled. More specifically, we studied whether the joint probability distribution defined by the example networks is closer to the ground truth distribution after tuning compared to before tuning, with closeness measured in terms of the KL-divergence. In addition, we studied whether output query probabilities, other than the one that was the focus of tuning, are closer to their ground truth values after tuning compared to before tuning with closeness measured as the absolute difference in probability. For tuning we used four different heuristics all assigning a different weight to the parameter uncertainty and we tuned with 1 up to 5 parameters.

The most striking observation is that for one of the considered outputs, its fit to its ground truth value becomes worse after tuning in more than half of the networks. This observation holds given all number of tuning parameters and for all heuristics. For the other two considered outputs results are better, but even for these outputs, the best results do not exceed 65% of networks with a better fit. With respect to the KL-divergence, we observed that in the best case, in around 70% of networks the distance to the ground truth distribution decreased by tuning. Thus even in the best case, in roughly 30% of the networks tuning resulted in a poorer fit to the ground truth distribution. Our main conclusion therefore is that it might not be taken for granted that outputs of a Bayesian network other than the tuned output will be closer to their ground truth values, nor that the distribution represented by the network is closer to the modelled ground truth distribution after tuning one output of the network to its ground truth value. Tuning of Bayesian networks should therefore be done with care.

With respect to the KL-divergence we moreover observed that, using more tuning parameters improved the results and that assigning a higher weight to the variance of the parameters improved the results. These trend were far less prominent or even absent for the considered output probabilities. In future research we want to investigate these trends in more detail. We further want to investigate quantitative effects as well, since if changes are just small, a deterioration of an output may be acceptable. Further research may reveal in which cases tuning may be unproblematic. It will also be interesting to investigate the effects

of tuning in more realistic distributions, for example by using real life Bayesian networks, considering the modelled distribution as ground truth distribution and further following the procedure used in this paper. Note that then just a single ground truth distribution for a network structure is available, which limits the research options. Finally, we would like to investigate whether it is possible to predict under what conditions tuning will bring about an improvement of other outputs and of the KL-divergence. This knowledge may then be used to develop tuning heuristics that are ensured to enhance a network as a whole.

Acknowledgements

This publication is part of the Personalized Care in Oncology program with file number P21-03 of the research program NWO Perspectief which is (partly) financed by the Dutch Research Council (NWO).

References

1. Bolt, J.H.: Bayesian networks: a combined tuning heuristic. In: Antonucci, A., Corani, G., Campos, C.P. (eds.) Proceedings of the 8th International Conference on Probabilistic Graphical Models. Proceedings of Machine Learning Research, vol. 52, pp. 37–49. PMLR, Lugano, Switzerland (2016)
2. Bolt, J.H., Hommersom, A., Renooij, S.: Involving uncertainty in Bayesian network tuning. In: Sauerwald, K., Thimm, M. (eds.) Symbolic and Quantitative Approaches to Reasoning with Uncertainty. pp. 61–74. Springer Nature Switzerland, Cham (2025)
3. Bolt, J.H., Renooij, S.: Local sensitivity of Bayesian networks to multiple simultaneous parameter shifts. In: van der Gaag, L.C., Feelders, A.J. (eds.) Proceedings of the 7th International Conference on Probabilistic Graphical Models. pp. 65–80. Springer International Publishing, Cham (2014)
4. Bolt, J.H., van der Gaag, L.C.: Balanced sensitivity functions for tuning multi-dimensional Bayesian network classifiers. International Journal of Approximate Reasoning **80**, 361–376 (2017). https://doi.org/10.1016/j.ijar.2016.07.011
5. Castillo, E., Gutiérrez, J.M., Hadi, A.S.: Parametric structure of probabilities in Bayesian networks. In: Froidevaux, C., Kohlas, J. (eds.) Proceedings of the European Conference on Symbolic and Quantitative Approaches to Reasoning and Uncertainty. pp. 89–98. Springer Berlin Heidelberg, Berlin, Heidelberg (1995)
6. Chan, H., Darwiche, A.: Sensitivity analysis in Bayesian networks: From single to multiple parameters. In: Halpern, J., Meek, C. (eds.) Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence. pp. 67–75 (2004)
7. Chan, H., Darwiche, A.: A distance measure for bounding probabilistic belief change. International Journal of Approximate Reasoning **38**, 149–174 (2005)
8. Chan, H., Darwiche, A.: When do numbers really matter? Journal of Artificial Intelligence Research **17**, 265–287 (2002). https://doi.org/10.1613/JAIR.967
9. Heckerman, D.: A tutorial on learning with Bayesian networks. In: Jordan, M.I. (ed.) Learning in Graphical Models, pp. 301–354. Springer Netherlands, Dordrecht (1998). https://doi.org/10.1007/978-94-011-5014-9_11
10. Jensen, F.V., Nielsen, T.D.: Bayesian Networks and Decision Graphs. Springer Science & Business Media, 2nd edn. (2007)

11. Kullback, S., Leibler, R.: On information and sufficiency. Annals of Mathematical Statistics **22**, 79–86 (1951)
12. Leonelli, M., Smith, J.Q.: The diameter of a stochastic matrix: A new measure for sensitivity analysis in Bayesian networks. International Journal of Approximate Reasoning **185**, 109470 (2025)
13. Pearl, J.: Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann (1988)