International Journal of Approximate Reasoning 80 (2017) 361–376

Contents lists available at ScienceDirect

International Journal of Approximate Reasoning

www.elsevier.com/locate/ijar

Balanced sensitivity functions for tuning multi-dimensional Bayesian network classifiers ☆

Janneke H. Bolt *, Linda C. van der Gaag

Department of Information and Computing Sciences, Utrecht University, P.O. Box 80.089, 3508 TB Utrecht, The Netherlands

ARTICLE INFO

Article history:
Received 3 December 2015
Received in revised form 11 May 2016
Accepted 19 July 2016
Available online 2 August 2016

Keywords:
Bayesian networks
Multi-dimensional classifiers
Higher-order sensitivity functions
Balanced sensitivity functions
Network tuning

ABSTRACT

Multi-dimensional Bayesian network classifiers are Bayesian networks of restricted topological structure, which are tailored to classifying data instances into multiple dimensions. Like more traditional classifiers, multi-dimensional classifiers are typically learned from data and may include inaccuracies in their parameter probabilities. We will show that the graphical properties and dedicated use of these classifiers induce higher-order sensitivity functions of a highly constrained functional form in these parameters. We then introduce the concept of balanced sensitivity function in which multiple parameters are functionally related by the odds ratios of their original and new values, and argue that these functions provide for a suitable heuristic for tuning multi-dimensional classifiers with guaranteed bounds on the effects on their output probabilities. We demonstrate the practicability of our heuristic by means of a classifier for a real-world application in the veterinary field.

1. Introduction

The family of multi-dimensional Bayesian network classifiers, or MDCs for short, was introduced to generalise Bayesian network classifiers to application domains that require data instances to be classified into multiple dimensions [13,18]. An MDC includes multiple class variables and multiple feature variables, which are connected by a bipartite graph directed from the class variables to the feature variables. Classifying a data instance by an MDC amounts to computing the joint probability distribution over the class variables given the instance's features, and returning the most likely joint class value combination. Since their introduction, multi-dimensional Bayesian network classifiers enjoy a growing interest as suitable models for multi-dimensional classification [4,5,10,17].

Like more traditional classifiers, MDCs are typically learned from data, for which purpose tailored algorithms have been designed [8,13,14,18]. While real-world data often prove suboptimal already for constructing one-dimensional classifiers, that is, classifiers with just a single class variable, skewness properties of the joint probability distribution to be modelled may prove especially problematic upon learning a multi-dimensional classifier. Expert knowledge can then be instrumental for correcting biases from the data collection by careful adjustment of the parameter probabilities of the learned classifier.

Adjusting the parameters of a multi-dimensional Bayesian network classifier requires detailed insight in the effects of the adjustment on all possible output probabilities. For Bayesian networks in general, the technique of sensitivity analysis has evolved as a practical tool for studying the effects of parameter adjustment. Research so far has focused on one-way sensi-

☆ This research was supported by the Netherlands Organisation for Scientific Research (NWO), grant number 612.066.928.
* Corresponding author.
E-mail addresses: j.h.bolt@uu.nl (J.H. Bolt), l.c.vandergaag@uu.nl (L.C. van der Gaag).

http://dx.doi.org/10.1016/j.ijar.2016.07.011
0888-613X/

tivity analyses of Bayesian networks, that is, on analyses in which the effects of varying a single parameter are investigated [9]. The effects of simultaneous variation of multiple parameters have received far less attention, mostly due to the computational burden of establishing the functions describing these effects. Chan and Darwiche have identified the solution space of multiple parameter changes needed to enforce a specific constraint on a network's output probabilities [6]; the runtime complexity of their method for identifying this space however, grows rapidly with the number of conditional probability tables involved. De Bock et al. presented an efficient algorithm for establishing the effects of varying local distributions within a fixed credal set on a most probable explanation (MPE), that is, on the joint value assignment to all non-observed variables of highest probability [2]; they do not provide an exact functional relationship between the varied parameters and an output probability of interest to be used for further manipulation, however.

Yet, for practical applications, tuning the output probabilities of a Bayesian network by changing multiple parameters may be preferred over a change of a single parameter probability, for example as the use of multiple parameters upon tuning may result in a much more levelled adjustment of the network. In this paper, we focus on multi-dimensional Bayesian network classifiers and present a suitable heuristic for tuning output probabilities by simultaneous adjustment of multiple parameters. We will show that the topological properties and dedicated use of a multi-dimensional classifier induce higher-order sensitivity functions of restricted functional form which can be established efficiently. By employing an appropriate balancing scheme of parameter adjustment, the higher-order sensitivity function at hand is reduced to an insightful single-parameter function which allows ready manipulation. Balancing schemes thereby provide for a suitable heuristic for tuning, which is shown to incur changes within guaranteed bounds in all output probabilities over the class variables.

The paper is organised as follows. In Section 2 we review multi-dimensional Bayesian network classifiers, and sensitivity functions of Bayesian networks in general; we further review a distance measure for comparing probability distributions. In Section 3 we derive the general form of a higher-order sensitivity function for MDCs. In Section 4, the concept of balanced sensitivity function is introduced; we further describe how such a function is used for effective tuning of the output probabilities of a multi-dimensional classifier. In Section 5 we prove bounds on the changes induced in all output probabilities by a balancing scheme of parameter adjustment. In Section 6 the basic idea of our heuristic for tuning is illustrated by means of two examples, among which is an example from a real-world setting. Section 7 concludes the paper.

2. Preliminaries

We briefly review multi-dimensional Bayesian network classifiers and thereby introduce our notational conventions. We further describe higher-order sensitivity functions for Bayesian networks in general, and review the *CD*-distance measure introduced by Chan and Darwiche for comparing probability distributions.

2.1. Bayesian networks and multi-dimensional classifiers

We consider a set of discrete random variables $\mathbf{V} = \{V_1, \ldots, V_m\}$, $m \geq 1$. We will use v_i to denote an arbitrary value of V_i, and will write v and $\bar{v}$ for the two values *true* and *false* respectively, of a binary variable V. A joint value combination for the set of variables $\mathbf{V}$ will be indicated by $\mathbf{v}$. We will use $\mathcal{V}_i$ to indicate the set of possible values of V_i, and $\boldsymbol{\mathcal{V}}$ to indicate the set of all possible value combinations of $\mathbf{V}$.

A Bayesian network in general is a graphical model of a joint probability distribution Pr over a set of random variables $\mathbf{V}$. Each variable in $\mathbf{V}$ is represented by a node in a directed acyclic graph, and vice versa; (in-)dependencies between the variables are, as far as possible, captured by the graph's set of arcs according to the well-known d-separation criterion [15]. The set of parents of a variable V_i in the graph will be indicated by $\mathbf{Pa}(V_i)$. Each variable $V_i \in \mathbf{V}$ further has associated a set of local conditional probability distributions $\Pr(V_i \,|\, \mathbf{Pa}(V_i))$; by $\Pr(V_i \,|\, \mathbf{pa}(V_i))$ we indicate a specific distribution from this set, given the value combination $\mathbf{pa}(V_i)$ for V_i's parents. The set of all distributions specified for a variable is called its conditional probability table, or CPT for short; the separate probabilities in this table are termed the variable's parameters. The joint probability distribution Pr now factorises over the network's graphical structure as

$$\Pr(\mathbf{v}) \;=\; \prod_{V_i \in \mathbf{V}} \Pr(v_i \,|\, \mathbf{pa}(V_i))$$

where v_i and $\mathbf{pa}(V_i)$ are compatible with $\mathbf{v}$. We will use $\sim$ and $\nsim$ to indicate compatibility and incompatibility of value combinations, respectively. For the two variables A and B for example, $ab \sim abc$ and $ab \sim b$ yet $ab \nsim \bar{a}cd$. Value combinations for disjoint sets of variables are always considered compatible.

A multi-dimensional Bayesian network classifier is a Bayesian network of restricted graphical topology. Its set of variables is partitioned into a set $\mathbf{C}$ of class variables and a set $\mathbf{F}$ of feature variables; its graphical structure does not allow feature variables to have class children [1,13,18]. We will write $MDC(\mathbf{C}, \mathbf{F})$ to indicate a multi-dimensional classifier. For each feature variable $F_i \in \mathbf{F}$, we will further use $\mathbf{F}_{F_i} = \mathbf{F} \cap \mathbf{Pa}(F_i)$ to denote its set of feature parents and $\mathbf{C}_{F_i} = \mathbf{C} \cap \mathbf{Pa}(F_i)$ to denote its class parents. A multi-dimensional classifier $MDC(\mathbf{C}, \mathbf{F})$ is used to assign a combination of feature values $\mathbf{f}$ to a most likely class value combination $\mathbf{c}$, that is, it is used to establish $\operatorname{argmax}_{\mathbf{c}} \Pr(\mathbf{c} \,|\, \mathbf{f})$. We note that the most likely combination of class values not necessarily coincides with the combination of most likely classes. Throughout this paper, we will assume full feature information, that is, we assume that values are available for all feature variables involved, unless explicitly stated otherwise.

2.2. Sensitivity functions of Bayesian networks

Upon systematically varying multiple parameters $\mathbf{x} = \{x_1, \ldots, x_n\}$, $n \geq 1$, of a Bayesian network in general, a sensitivity function results which expresses the effect of the variation on an output probability $\Pr(\mathbf{w} \,|\, \mathbf{u})$, with $\mathbf{U} \cap \mathbf{W} = \varnothing$, of interest [9]. More specifically, the result is a higher-order sensitivity function in $\mathbf{x}$ which takes the form of a fractional-multilinear function:

$$\Pr(\mathbf{w} \,|\, \mathbf{u})(\mathbf{x}) = \frac{\sum_{\mathbf{x}_k \in \mathcal{P}(\mathbf{x})} \left(c_k \cdot \prod_{x_i \in \mathbf{x}_k} x_i \right)}{\sum_{\mathbf{x}_k \in \mathcal{P}(\mathbf{x})} \left(d_k \cdot \prod_{x_i \in \mathbf{x}_k} x_i \right)}$$

where $\mathcal{P}(\mathbf{x})$ is the powerset of the set of parameters $\mathbf{x}$, and where the constants c_k, d_k are determined by (a subset of) the non-varied parameters of the network at hand. A two-way sensitivity function in the parameters x and y for example, has the following general form:

$$\Pr(\mathbf{w} \,|\, \mathbf{u})(x, y) = \frac{c_1 + c_2 \cdot x + c_3 \cdot y + c_4 \cdot x \cdot y}{d_1 + d_2 \cdot x + d_3 \cdot y + d_4 \cdot x \cdot y}$$

The maximum number of additive terms in an n-way sensitivity function equals $2^{n+1} - 1$, growing exponentially with the number of parameters being varied, and determining the constants involved is computationally challenging in general.

In the sequel, we will write $\mathbf{x^o} = \{x_1^o, \ldots, x_n^o\}$ to indicate the original values of the parameters $\mathbf{x}$ in the network under study; $\Pr^o$ is used to indicate the original probability distribution, that is, with the original values of all parameters involved, and $O^o = \frac{\Pr^o}{1-\Pr^o}$ is used to denote the original odds defined by the network. Throughout this paper, we will assume that deterministic parameters are not varied and that varied parameters will not adopt the extreme values. We will further assume, for ease of exposition, that just a single parameter per local probability distribution $\Pr(V_i \,|\, \mathbf{pa}(V_i))$ over a variable V_i is varied actively. Upon varying a parameter x from $\Pr(V_i \,|\, \mathbf{pa}(V_i))$ however, at least one of the distribution's other parameters has to be varied as well to let the distribution sum to 1; the parameters $\mathbf{y} = \{y_1, \ldots, y_k\}$, $y_j \neq x$, from the distribution at hand are termed the co-varying parameters for x. While different co-variation schemes have been proposed [16], proportional co-variation is the most commonly used scheme in practice. With this scheme, all parameters $y_j \in \mathbf{y}$ are varied proportionally with the parameter x under study, that is, $y_j = y_j^o \cdot (1 - x)/(1 - x^o)$. With proportional co-variation, we thus have that

$$\frac{y_j}{y_j^o} = \frac{(1 - x)}{(1 - x^o)}$$

for all parameters $y_j \in \mathbf{y}$.

2.3. A distance measure for probability distributions

For comparing an original probability distribution $\Pr^o$ with the distribution $\Pr$ after parameter adjustment, Chan and Darwiche introduced a new distance measure, which we will denote by CD. The measure is defined as:

$$CD(\Pr^o, \Pr) \;=\; \ln \max_{\mathbf{v} \in \boldsymbol{\mathcal{V}}} \left(\frac{\Pr(\mathbf{v})}{\Pr^o(\mathbf{v})} \right) \;-\; \ln \min_{\mathbf{v} \in \boldsymbol{\mathcal{V}}} \left(\frac{\Pr(\mathbf{v})}{\Pr^o(\mathbf{v})} \right)$$

where $0/0$ and ∞/∞ are taken to be equal to 1 [7]. By definition, the CD-distance between any two distributions over the same set of variables is positive and symmetric, and equals zero if and only if the two distributions are the same. Given changes in the parameters from the conditional probability table of a single variable V_i with the parents $\mathbf{Pa}(V_i)$ in a Bayesian network, the distance measure reduces to:

$$CD(\Pr^o, \Pr) \;=\; \ln \max_{z \in \Pr(V_i \,|\, \mathbf{Pa}(V_i))} \left(\frac{z}{z^o} \right) \;-\; \ln \min_{z \in \Pr(V_i \,|\, \mathbf{Pa}(V_i))} \left(\frac{z}{z^o} \right)$$

We note that the CD-distance between the original and adjusted distributions over the full set of variables $\mathbf{V}$ now is determined locally at the variable V_i. For changes in just a single local distribution $\Pr(V_i \,|\, \mathbf{pa}(V_i))$ given $\mathbf{pa}(V_i)$ over the variable V_i, the expression reduces even further, taking the minimum and maximum of the term z/z^o over all parameters from the distribution at hand. Since with proportional co-variation we have that

$$\frac{y_j}{y_j^o} = \frac{(1 - x)}{(1 - x^o)}$$

for all co-varying parameters y_j, with this scheme the CD-distance equals

$$CD(\Pr^o, \Pr) \;=\; \ln \max \left\{ \frac{x}{x^o}, \frac{1 - x}{1 - x^o} \right\} - \ln \min \left\{ \frac{x}{x^o}, \frac{1 - x}{1 - x^o} \right\}$$

for all possible values of the parameter x being adjusted.

Given changes in multiple probability tables, the *CD*-distance between an original distribution $\Pr^o$ and the distribution Pr after parameter adjustment is bounded by the sum of the distances induced by the changes in the individual tables. In general, this sum constitutes just an upper bound on the distance $CD(\Pr^o, \Pr)$ as its summands may pertain to mutually incompatible parameter combinations. The sum does equal the exact common contribution to the *CD*-distance of changes in two or more probability tables, if the variables to which the tables pertain do not have a direct child–parent relationship nor share any common parents.

The *CD*-distance between an original distribution $\Pr^o$ and the distribution Pr resulting from parameter variation, serves to bound the odds ratio of the effect of the variation for a specific output probability $\Pr(\mathbf{w}\,|\,\mathbf{u})$ of interest by:

$$e^{-CD(\Pr^o,\Pr)} \;\leq\; \frac{O(\mathbf{w}\,|\,\mathbf{u})}{O^o(\mathbf{w}\,|\,\mathbf{u})} \;\leq\; e^{CD(\Pr^o,\Pr)}$$

The effect on the output probability itself is then bounded by:

$$\frac{p^o \cdot e^{-CD}}{p^o \cdot (e^{-CD}-1)+1} \;\leq\; \Pr(\mathbf{w}\,|\,\mathbf{u}) \;\leq\; \frac{p^o \cdot e^{CD}}{p^o \cdot (e^{CD}-1)+1}$$

writing *CD* for $CD(\Pr^o, \Pr)$ and p^o for $\Pr^o(\mathbf{w}\,|\,\mathbf{u})$ [7]. We note that these bounds are not tight in general; an example in which the bounds are not met is provided in the Appendix.

3. Higher-order sensitivity functions for multi-dimensional classifiers

Establishing a higher-order sensitivity function for a Bayesian network in general is computationally challenging, as the number of additive terms involved, and hence the number of constants to be computed, can be exponential in the number of parameters being varied. We will now show that, as a consequence of its restricted graphical structure and dedicated use, a multi-dimensional classifier allows more ready calculation of higher-order sensitivity functions for its output probabilities. We will show more specifically, that an output probability $\Pr(\mathbf{c}\,|\,\mathbf{f})$ for a particular combination of class values $\mathbf{c}$ can be expressed in terms of the original probabilities $\Pr^o(\mathbf{C}\,|\,\mathbf{f})$ of all such combinations and the original and adjusted values of all parameters compatible with the feature combination $\mathbf{f}$ to be classified. The general form of the sensitivity function is given in the proposition below.

Proposition 1. *Let MDC*$(\mathbf{C}, \mathbf{F})$ *be a multi-dimensional Bayesian network classifier as defined before, and let* $\mathbf{C}_r \subseteq \mathbf{C}$ *be its set of root class variables. Let* $\mathbf{f} \in \mathcal{F}$ *be a combination of feature values, and let* $\mathbf{x} = \{x_1, \ldots, x_n\}$, $n \geq 1$, *be the set of parameters from the probability tables of* $\mathbf{C}_r \cup \mathbf{F}$ *which are compatible with* $\mathbf{f}$. *Then, for all* $\mathbf{c} \in \mathcal{C}$,

$$\Pr(\mathbf{c}\,|\,\mathbf{f})(\mathbf{x}) \;=\; \frac{\Pr^o(\mathbf{c}\,|\,\mathbf{f}) \cdot \prod\limits_{x_i \sim \mathbf{c}, x_j \nsim \mathbf{c}} x_i \cdot x_j^o}{\sum\limits_{\mathbf{c}^* \in \mathcal{C}} \left(\Pr^o(\mathbf{c}^*\,|\,\mathbf{f}) \cdot \prod\limits_{x_i \sim \mathbf{c}^*, x_j \nsim \mathbf{c}^*} x_i \cdot x_j^o \right)}$$

Proof. We write the output probability $\Pr(\mathbf{c}\,|\,\mathbf{f})$ for the value combinations $\mathbf{c}$ and $\mathbf{f}$ as

$$\Pr(\mathbf{c}\,|\,\mathbf{f}) = \frac{\Pr(\mathbf{f}\,|\,\mathbf{c}) \cdot \Pr(\mathbf{c})}{\sum_{\mathbf{c}^* \in \mathcal{C}} \left(\Pr(\mathbf{f}\,|\,\mathbf{c}^*) \cdot \Pr(\mathbf{c}^*) \right)}$$

Including terms involving the original probability values $\Pr^o(\mathbf{c}\,|\,\mathbf{f})$ and $\Pr^o(\mathbf{c})$ results in

$$\Pr(\mathbf{c}\,|\,\mathbf{f}) = \frac{\left(\frac{\Pr(\mathbf{f}\,|\,\mathbf{c}) \cdot \Pr(\mathbf{c}) \cdot \Pr^o(\mathbf{f}\,|\,\mathbf{c}) \cdot \Pr^o(\mathbf{c})}{\Pr^o(\mathbf{f}) \cdot \Pr^o(\mathbf{f}\,|\,\mathbf{c}) \cdot \Pr^o(\mathbf{c})} \right)}{\sum_{\mathbf{c}^* \in \mathcal{C}} \left(\frac{\Pr(\mathbf{f}\,|\,\mathbf{c}^*) \cdot \Pr(\mathbf{c}^*) \cdot \Pr^o(\mathbf{f}\,|\,\mathbf{c}^*) \cdot \Pr^o(\mathbf{c}^*)}{\Pr^o(\mathbf{f}) \cdot \Pr^o(\mathbf{f}\,|\,\mathbf{c}^*) \cdot \Pr^o(\mathbf{c}^*)} \right)} = \frac{\left(\frac{\Pr^o(\mathbf{c}\,|\,\mathbf{f}) \cdot \Pr(\mathbf{f}\,|\,\mathbf{c}) \cdot \Pr(\mathbf{c})}{\Pr^o(\mathbf{f}\,|\,\mathbf{c}) \cdot \Pr^o(\mathbf{c})} \right)}{\sum_{\mathbf{c}^* \in \mathcal{C}} \left(\frac{\Pr^o(\mathbf{c}^*\,|\,\mathbf{f}) \cdot \Pr(\mathbf{f}\,|\,\mathbf{c}^*) \cdot \Pr(\mathbf{c}^*)}{\Pr^o(\mathbf{f}\,|\,\mathbf{c}^*) \cdot \Pr^o(\mathbf{c}^*)} \right)}$$

Rearranging the summands from the denominator into a single fraction gives

$$\Pr(\mathbf{f}) = \frac{\sum\limits_{\mathbf{c}^* \in \mathcal{C}} \left(\Pr^o(\mathbf{c}^*\,|\,\mathbf{f}) \cdot \Pr(\mathbf{f}\,|\,\mathbf{c}^*) \cdot \Pr(\mathbf{c}^*) \cdot \prod\limits_{\mathbf{c}' \in \mathcal{C} \setminus \{\mathbf{c}^*\}} \Pr^o(\mathbf{f}\,|\,\mathbf{c}') \cdot \Pr^o(\mathbf{c}') \right)}{\prod\limits_{\mathbf{c}^* \in \mathcal{C}} \left(\Pr^o(\mathbf{f}\,|\,\mathbf{c}^*) \cdot \Pr^o(\mathbf{c}^*) \right)}$$

where $\mathcal{C}\setminus\{\mathbf{c}^*\}$ denotes the set of all value combinations for $\mathbf{C}$ with the exclusion of the combination $\mathbf{c}^*$ in the term at hand. Substitution and simplification now gives

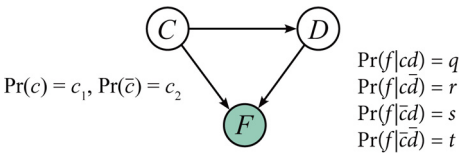

Fig. 1. A small multi-dimensional Bayesian network classifier with its conditional probability tables for C and F.

$$\Pr(\mathbf{c}\,|\,\mathbf{f}) = \frac{\Pr^o(\mathbf{c}\,|\,\mathbf{f}) \cdot \Pr(\mathbf{f}\,|\,\mathbf{c}) \cdot \Pr(\mathbf{c}) \cdot \prod_{\mathbf{c}' \in \mathcal{C}\setminus\{\mathbf{c}\}} \Pr^o(\mathbf{f}\,|\,\mathbf{c}') \cdot \Pr^o(\mathbf{c}')}{\sum_{\mathbf{c}^* \in \mathcal{C}} \left(\Pr^o(\mathbf{c}^*\,|\,\mathbf{f}) \cdot \Pr(\mathbf{f}\,|\,\mathbf{c}^*) \cdot \Pr(\mathbf{c}^*) \cdot \prod_{\mathbf{c}' \in \mathcal{C}\setminus\{\mathbf{c}^*\}} \Pr^o(\mathbf{f}\,|\,\mathbf{c}') \cdot \Pr^o(\mathbf{c}') \right)}$$

$$= \frac{\Pr^o(\mathbf{c}\,|\,\mathbf{f}) \cdot \prod_i \Pr(f_i\,|\,\mathbf{c}, \mathbf{f}_{F_i}) \cdot \Pr(\mathbf{c}) \cdot \prod_{\mathbf{c}' \in \mathcal{C}\setminus\{\mathbf{c}\}} \prod_i \Pr^o(f_i\,|\,\mathbf{c}', \mathbf{f}_{F_i}) \cdot \Pr^o(\mathbf{c}')}{\sum_{\mathbf{c}^* \in \mathcal{C}} \left(\Pr^o(\mathbf{c}^*\,|\,\mathbf{f}) \cdot \prod_i \Pr(f_i\,|\,\mathbf{c}^*, \mathbf{f}_{F_i}) \cdot \Pr(\mathbf{c}^*) \cdot \prod_{\mathbf{c}' \in \mathcal{C}\setminus\{\mathbf{c}^*\}} \prod_i \Pr^o(f_i\,|\,\mathbf{c}', \mathbf{f}_{F_i}) \cdot \Pr^o(\mathbf{c}') \right)}$$

where we exploited the property that $\Pr(\mathbf{f}\,|\,\mathbf{c}) = \prod_i \Pr(f_i\,|\,\mathbf{c}, \mathbf{f}_{F_i})$ for all $\mathbf{f} \in \mathcal{F}$, $\mathbf{c} \in \mathcal{C}$, with $f_i, \mathbf{f}_{F_i} \sim \mathbf{f}$. We note that each term $\Pr(f_i\,|\,\mathbf{c}, \mathbf{f}_{F_i})$ can be reduced to $\Pr(f_i\,|\,\mathbf{c}_{F_i}, \mathbf{f}_{F_i})$, which corresponds to a parameter in the CPT of the variable F_i. We further observe that $\Pr(\mathbf{c}) = \prod_j \Pr(c_j\,|\,\mathbf{pa}(C_j))$ with $c_j, \mathbf{pa}(C_j) \sim \mathbf{c}$. Given that the parameters of the class variables with class parents are not varied and hence are cancelled out from the formula above, we find that

$$\Pr(\mathbf{c}\,|\,\mathbf{f})(\mathbf{x}) = \frac{\Pr^o(\mathbf{c}\,|\,\mathbf{f}) \cdot \prod_{x_i \sim \mathbf{c}, x_j \nsim \mathbf{c}} x_i \cdot x_j^o}{\sum_{\mathbf{c}^* \in \mathcal{C}} \left(\Pr^o(\mathbf{c}^*\,|\,\mathbf{f}) \cdot \prod_{x_i \sim \mathbf{c}^*, x_j \nsim \mathbf{c}^*} x_i \cdot x_j^o \right)}$$

where the parameters x_i, x_j are of the form $\Pr(c_k)$ or of the form $\Pr(f_k\,|\,\mathbf{c}_{F_k}, \mathbf{f}_{F_k})$ with $f_k, \mathbf{f}_{F_k} \sim \mathbf{f}$. □

The higher-order sensitivity function $\Pr(\mathbf{c}\,|\,\mathbf{f})(\mathbf{x})$ stated in the proposition above includes, for each feature variable F_i, for each local distribution over F_i given $\mathbf{pa}(F_i)$ with $\mathbf{f}_{F_i} \sim \mathbf{f}$, all parameters pertaining to the value $f_i \in \mathcal{F}_i$ compatible with $\mathbf{f}$. The parameters $\Pr(f_i'\,|\,\mathbf{pa}(F_i))$ with $f_i' \nsim \mathbf{f}$ of this distribution do not occur in the function, since these parameters are not involved directly in the computation of the output probability: upon varying such a parameter, the output probability is affected only indirectly through co-variation of the parameter $\Pr(f_i\,|\,\mathbf{pa}(F_i))$ with $f_i \sim \mathbf{f}$. Without loss of generality, we can thus include in the sensitivity function just the parameters which are compatible with $\mathbf{f}$. As a consequence, the proposition holds for any co-variation scheme used for the parameters of the feature variables. All parameters $\Pr(c_j)$ of a class variable $C_j \in \mathbf{C}_r$ are also included in the sensitivity function stated above. These parameters cannot be varied independently however, as their sum should remain equal to 1. By assuming a specific co-variation scheme, we could have included the dependent parameters implicitly; by their explicit inclusion, however, the function is independent of the co-variation scheme used for the class parameters.

As an example, we consider the multi-dimensional Bayesian network classifier from Fig. 1, and write the output probability $\Pr(cd\,|\,f)$ as a function of all parameters of the variables C and F which are compatible with f. We find that

$$\Pr(cd\,|\,f)(c_1, c_2, q, r, s, t) = \frac{p_1^o \cdot c_1 \cdot c_2^o \cdot q \cdot r^o \cdot s^o \cdot t^o}{p_1^o \cdot c_1 \cdot c_2^o \cdot q \cdot r^o \cdot s^o \cdot t^o + p_2^o \cdot c_1 \cdot c_2^o \cdot q^o \cdot r \cdot s^o \cdot t^o + p_3^o \cdot c_1^o \cdot c_2 \cdot q^o \cdot r^o \cdot s \cdot t^o + p_4^o \cdot c_1^o \cdot c_2 \cdot q^o \cdot r^o \cdot s^o \cdot t}$$

where $p_1^o = \Pr^o(cd\,|\,f)$, $p_2^o = \Pr^o(c\bar{d}\,|\,f)$, $p_3^o = \Pr^o(\bar{c}d\,|\,f)$ and $p_4^o = \Pr^o(\bar{c}\bar{d}\,|\,f)$ constitute the original joint probability distribution $\Pr^o(CD\,|\,f)$ over the two class variables C and D. We note that, upon using the function for further manipulation, the property $c_1 + c_2 = 1$ needs to be explicitly enforced. The parameters q, r, s and t, on the other hand, can be varied independently.

An important feature of the higher-order sensitivity function from Proposition 1 is that although it includes all parameters from the probability tables of the variables in $\mathbf{C}_r \cup \mathbf{F}$ compatible with the feature value combination to be classified, it is easily adapted to a sensitivity function involving just a subset of these parameters. Since each parameter is included exactly once in each term of the fraction, either by its original value x^o or as a variable x, any non-varied parameter simply cancels out from the function. As an example, the two-way sensitivity function which describes the output probability $\Pr(cd\,|\,f)$ of the classifier from Fig. 1 in terms of just the parameters q and t, is found to be

$$\Pr(cd\,|\,f)(q, t) = \frac{p_1^o \cdot q \cdot t^o}{p_1^o \cdot q \cdot t^o + p_2^o \cdot q^o \cdot t^o + p_3^o \cdot q^o \cdot t^o + p_4^o \cdot q^o \cdot t}$$

The sensitivity function is also readily adapted to output probabilities $\Pr(\mathbf{c}\,|\,\mathbf{g})$ with $\mathbf{G} \subset \mathbf{F}$, provided that the unobserved feature variables have no observed children in the MDC's graphical structure. Since unobserved feature variables do not belong to the sensitivity set of the class variables [9], the parameters of these feature variables are excluded from the sensitivity function. Our result thus holds for particular contexts of partial feature information as well, and therefore generalises to at least some extent beyond our assumption of full evidence.

The sensitivity function stated in Proposition 1 reveals that an output probability of a multi-dimensional Bayesian network classifier is guaranteed to change monotonically with particular schemes of parameter adjustment. The following proposition formalises this observation.

Proposition 2. *Let MDC*$(\mathbf{C}, \mathbf{F})$ *be a multi-dimensional classifier as before, and let* $\mathbf{C}_r \subseteq \mathbf{C}$ *be its set of root class variables. Let* $\Pr(\mathbf{c}\,|\,\mathbf{f})$ *be the output probability of interest of MDC*$(\mathbf{C}, \mathbf{F})$*, and let* $\mathbf{x} = \{x_1, \ldots, x_n\}$, $n \geq 1$, *be the set of parameters from the probability tables of* $\mathbf{C}_r \cup \mathbf{F}$ *which are compatible with* $\mathbf{f}$*. Then, for any value settings* $\mathbf{x}'$, $\mathbf{x}^*$ *for the parameters* $\mathbf{x}$*, it holds that*

- *if* $x'_i \geq x^*_i$ *for all* $x_i \sim \mathbf{c}$ *and* $x'_j \leq x^*_j$ *for all* $x_j \nsim \mathbf{c}$*, then* $\Pr(\mathbf{c}\,|\,\mathbf{f})(\mathbf{x}') \;\geq\; \Pr(\mathbf{c}\,|\,\mathbf{f})(\mathbf{x}^*)$*;*
- *if* $x'_i \leq x^*_i$ *for all* $x_i \sim \mathbf{c}$ *and* $x'_j \geq x^*_j$ *for all* $x_j \nsim \mathbf{c}$*, then* $\Pr(\mathbf{c}\,|\,\mathbf{f})(\mathbf{x}') \;\leq\; \Pr(\mathbf{c}\,|\,\mathbf{f})(\mathbf{x}^*)$.

Proof. We temporarily drop the requirement of co-variation for the class variables. For all parameters $x_i \in \mathbf{x}$ compatible with the value combination $\mathbf{c}$, the function stated in Proposition 1 takes the form $(x_i \cdot r)/(x_i \cdot s + t)$, where the constants r, s, t arise from multiplication and addition of probabilities and hence are non-negative. The first derivative of the function equals $(r \cdot t)/(s \cdot x_i + t)^2$, which is positive for any value of x_i. An increase of the value of x_i will thus result in an increase of the output probability, regardless of the values of r, s and t. Similarly, for a parameter x_j incompatible with $\mathbf{c}$ we find that an increase of the value of x_j will result in a decrease of the output probability. A simultaneous increase of parameters compatible with $\mathbf{c}$ and decrease of parameters incompatible with $\mathbf{c}$ will thus result in an increase of the output probability. We now adopt again the requirement of co-variation for the root class parameters. Since the proposition holds for any simultaneous increase of parameters compatible with $\mathbf{c}$ and decrease of parameters incompatible with $\mathbf{c}$, the proposition also holds if such changes result from co-variation. The proof of the second part of the proposition is analogous. □

From the proposition above we have that by simultaneously increasing the values of the parameters in $\mathbf{x}$ which are compatible with $\mathbf{c}$ and decreasing the parameters incompatible with $\mathbf{c}$, the output probability for the class value combination $\mathbf{c}$ is guaranteed to increase monotonically, and vice versa. An adjustment scheme as described above will be called guaranteed monotone for the output probability $\Pr(\mathbf{c}\,|\,\mathbf{f})$. We note that the conditions for an adjustment scheme to be guaranteed monotone specify the directions in which the separate parameters need to be adjusted to arrive at a desired effect on the output probability at hand. The following corollary now further states that with a guaranteed monotone parameter adjustment, the output probability takes its maximum and minimum at the parameters' extreme values.

Corollary 1. *Let MDC*$(\mathbf{C}, \mathbf{F})$, $\Pr(\mathbf{c}\,|\,\mathbf{f})$ *and* $\mathbf{x}$ *be as before. Then,*

- *the sensitivity function* $\Pr(\mathbf{c}\,|\,\mathbf{f})(\mathbf{x})$ *attains its maximum at* $x_i = 1$ *for all* $x_i \sim \mathbf{c}$ *and* $x_j = 0$ *for all* $x_j \nsim \mathbf{c}$*;*
- *for feature parameters and parameters of binary class variables, the sensitivity function attains its minimum at* $x_i = 0$ *for all* $x_i \sim \mathbf{c}$ *and* $x_j = 1$ *for all* $x_j \nsim \mathbf{c}$*;*
- *for the parameters of non-binary class variables, the sensitivity function attains its minimum at* $x_i = 0$ *for all* $x_i \sim \mathbf{c}$*, irrespective of the values of the parameters* $x_j \nsim \mathbf{c}$.

We note that with a guaranteed monotone scheme of parameter adjustment which involves the parameters of a root class variable, the sensitivity function necessarily includes the root class parameter which is compatible with the value combination $\mathbf{c}$ in the output probability of interest.

4. Balanced tuning of multi-dimensional classifiers

In the previous section we showed that the output probability $\Pr(\mathbf{c}\,|\,\mathbf{f})$ of a multi-dimensional Bayesian network classifier *MDC*$(\mathbf{C}, \mathbf{F})$ changes monotonically with a monotone parameter adjustment. While this property indicates the direction in which parameters are to be adjusted upon tuning, it does not as yet suggest the amount of adjustment required per parameter. For this purpose, we now introduce the concept of a balancing scheme for parameter adjustment. A balancing scheme governs a simultaneous change in all parameters under study by amounts which are defined by the odds ratios of their original values, and thereby further details the approach which was earlier suggested by Chan and Darwiche [6]. Balancing the parameters of a multi-dimensional Bayesian network classifier provides for a simple heuristic for parameter tuning; the heuristic moreover comes with guaranteed bounds on the *CD*-distance between the original probability distribution and the distribution resulting after tuning.

We begin by defining the concept of balancing scheme in general.

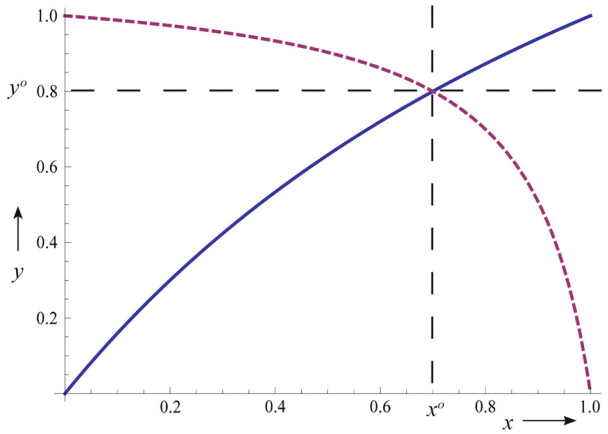

Fig. 2. Positively (solid) and negatively (dashed) balanced parameters x and y, with $x^o = 0.7$ and $y^o = 0.8$.

Definition 4.1. Let $x, y \in \langle 0, 1\rangle$ be two parameters of a Bayesian network in general, and let x^o, y^o be their original values. We say that a scheme for parameter adjustment *balances y positively* with x if

$$\frac{x^o \cdot (1-x)}{(1-x^o) \cdot x} = \frac{y^o \cdot (1-y)}{(1-y^o) \cdot y}$$

The scheme *balances y negatively* with x if

$$\frac{x^o \cdot (1-x)}{(1-x^o) \cdot x} = \frac{(1-y^o) \cdot y}{y^o \cdot (1-y)}$$

We note that, when the value of a parameter x is adjusted from x^o to x^a, the balancing scheme allows ready calculation of the new value y^a for the parameter y: if y is balanced positively with x, we find that

$$y^a = \frac{x^a \cdot (1-x^o) \cdot y^o}{x^a \cdot (y^o - x^o) + x^o \cdot (1-y^o)}$$

and if y is balanced negatively with x, the new value for y is found through

$$y^a = \frac{x^a \cdot x^o \cdot y^o - x^o \cdot y^o}{x^a \cdot (x^o + y^o - 1) - x^o \cdot y^o}$$

The balancing scheme thus prescribes the new value for the parameter y in the closed form $y = \frac{a+b \cdot x}{c+d \cdot x}$ with the constants a, b, c, d built from the original values of the two parameters x and y. An important property of a balancing scheme is that, if a parameter x is varied over the full value range $\langle 0, 1\rangle$, then the parameter y covers the full range $\langle 0, 1\rangle$ as well, that is, the range of possible values for y is not constrained by balancing y with x; this property is illustrated in Fig. 2.

Since a balancing scheme prescribes the unique amount of adjustment for all parameters balanced with a designated parameter x, simultaneous balanced variation of the parameters induces a function which describes the output probability of interest in terms of just the parameter x; this function is termed a balanced sensitivity function.

Definition 4.2. Let $\Pr(\mathbf{w}|\mathbf{u})$ be the output probability of a Bayesian network in general, and let $\mathbf{x} = \{x, \mathbf{x}^+, \mathbf{x}^-\}$ be (a subset of) the network's parameters; let $\Pr(\mathbf{w}|\mathbf{u})(\mathbf{x})$ be the higher-order sensitivity function for $\Pr(\mathbf{w}|\mathbf{u})$ in $\mathbf{x}$. A *balanced sensitivity function* for $\Pr(\mathbf{w}|\mathbf{u})$ is a function $\Pr(\mathbf{w}|\mathbf{u})(x \parallel \mathbf{x}^+, \mathbf{x}^-)$ in x, where the parameters $\mathbf{x}^+$ are balanced positively and the parameters $\mathbf{x}^-$ are negatively balanced with x.

The general form of a balanced sensitivity function $\Pr(\mathbf{w}|\mathbf{u})(x \parallel \mathbf{x}^+, \mathbf{x}^-)$ is found by taking the higher-order sensitivity function $\Pr(\mathbf{w}|\mathbf{u})(\mathbf{x})$ and replacing the parameters from $\mathbf{x} \setminus \{x\}$ by their closed expressions in x given above. A balanced sensitivity function is then found to be of the following functional form:

$$\Pr(\mathbf{w}|\mathbf{u})(x \parallel \mathbf{x}^+, \mathbf{x}^-) = \frac{c_0 + c_1 \cdot x + \ldots + c_m \cdot x^m}{d_0 + d_1 \cdot x + \ldots + d_m \cdot x^m}$$

where the constants c_j, d_j again are determined by the network's non-varied parameters and where each x^k is a multiplicative term of degree k; m is the number of probability tables from which the parameters in $\mathbf{x}$ are taken.

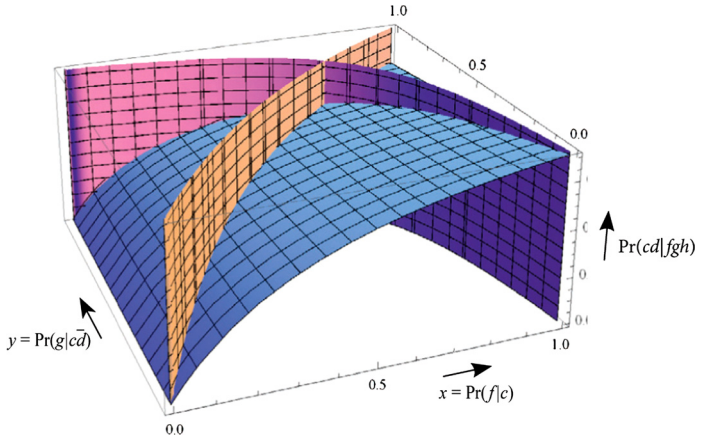

Fig. 3. A two-way sensitivity function in the parameters x and y, and surfaces defining two balanced sensitivity functions for the output probability $\Pr(cd\,|\,fgh)$ of the classifier from Fig. 4, with $x^o = 0.7$ and $y^o = 0.8$.

A balanced sensitivity function $\Pr(\mathbf{w}\,|\,\mathbf{u})(x \parallel \mathbf{x}^+, \mathbf{x}^-)$ in essence is the intersection of the higher-order sensitivity function $\Pr(\mathbf{w}\,|\,\mathbf{u})(\mathbf{x})$ with the surface defined by the balancing scheme employed. As an example, Fig. 3 depicts the two-way sensitivity function $\Pr(cd\,|\,fgh)(x, y)$ of the multi-dimensional classifier from Fig. 4, in the parameters $x = \Pr(f\,|\,c)$ and $y = \Pr(g\,|\,c\bar{d})$; the figure further depicts the two surfaces determining the balanced sensitivity functions that are derived from the two-way function through the balancing schemes $(x \parallel y^+)$ and $(x \parallel y^-)$, respectively.

Building upon the results of the previous section, we can now construct, for the output probability of a multi-dimensional Bayesian network classifier, a guaranteed monotone balancing scheme $(x \parallel \mathbf{x}^+, \mathbf{x}^-)$ and an associated balanced sensitivity function $\Pr(\mathbf{c}\,|\,\mathbf{f})(x \parallel \mathbf{x}^+, \mathbf{x}^-)$. We note that, since a balancing scheme allows all parameters involved to range over the entire interval $\langle 0, 1\rangle$, a guaranteed monotone sensitivity function covers the same value range of the output probability as the underlying higher-order function. For a desired change in the output probability of interest, the amount by which the parameter x is to be adjusted is readily established from the balanced sensitivity function. The balancing scheme of adjustment then serves to enforce the other parameter probabilities to be adjusted accordingly.

5. Bounds on *CD*-distance following a balanced parameter change

Tuning a Bayesian network in general is aimed at adjusting its parameters such that a desired effect on a designated output probability is attained. Parameter adjustment will induce not just the desired effect on the output probability of interest however, but will also affect other probabilities established from the network. We recall from Section 2.3, that the distance measure *CD* can be used for quantifying the differences between two probability distributions over the same set of variables. In this section, we will establish bounds, in terms of this distance measure, on the effects of parameter adjustment on a represented distribution.

We begin by defining the concept of extent of change as a measure of parameter adjustment.

Definition 5.1. Let $x \in \langle 0, 1\rangle$ be a parameter of a Bayesian network in general, and let x^o be its original value. The *extent of change* $\alpha(x)$ of x is defined as

$$\alpha(x) \;=\; \max\left\{\frac{(1-x^o)\cdot x}{x^o\cdot(1-x)}, \frac{x^o\cdot(1-x)}{(1-x^o)\cdot x}\right\}$$

We note that $\alpha(x) \geq 1$, that is, the extent of change of a parameter x equals at least 1 for any value of x; an extent of change equal to 1 is attained if and only if the parameter's adjusted value is equal to its original value. We further note that, by definition, $\alpha(x) = \alpha(y)$ for any balanced change of two parameters x and y.

We now first derive bounds on the separate terms $\frac{x}{x^o}$, $\frac{x^o}{x}$, $\frac{1-x}{1-x^o}$ and $\frac{1-x^o}{1-x}$ involved in an extent of change. We will subsequently use these bounds for establishing an upper bound on the overall effect of parameter adjustment on a represented joint probability distribution.

Lemma 1. *Let $x \in \langle 0, 1\rangle$ be a parameter of a Bayesian network, and let x^o be its original value. Let $\alpha(x)$ be the extent of change of x. Then,*

$$\left\{\frac{x}{x^o}, \frac{x^o}{x}, \frac{1-x}{1-x^o}, \frac{1-x^o}{1-x}\right\} \subseteq \left[\frac{1}{\alpha(x)}, \alpha(x)\right]$$

Proof. By definition we have that $\frac{x}{x^o} \cdot \frac{1-x^o}{1-x} = \alpha(x)$ for all values $x > x^o$ and $\frac{x^o}{x} \cdot \frac{1-x}{1-x^o} = \alpha(x)$ for all values $x < x^o$. We further have that $\alpha(x) = 1$ if and only if $x = x^o$. We now first address the extent of change $\alpha(x)$ for values $x > x^o$. From $x > x^o$ we have that $\frac{x}{x^o} > 1$ and $\frac{1-x^o}{1-x} > 1$. Now suppose that $\frac{x}{x^o} > \alpha(x)$. We would then find that $\frac{1-x^o}{1-x} < 1$, which contradicts our previous finding. We conclude that $\frac{x}{x^o} \leq \alpha(x)$ for all values of x and, by a similar argument, that $\frac{1-x^o}{1-x} \leq \alpha(x)$. Hence, $\{\frac{x}{x^o}, \frac{1-x^o}{1-x}\} \subseteq [1, \alpha(x)]$ and $\{\frac{x^o}{x}, \frac{1-x}{1-x^o}\} \subseteq [\frac{1}{\alpha(x)}, 1]$. For the values $x < x^o$ we analogously find that $\{\frac{x^o}{x}, \frac{1-x}{1-x^o}\} \subseteq [1, \alpha(x)]$ and $\{\frac{x}{x^o}, \frac{1-x^o}{1-x}\} \subseteq [\frac{1}{\alpha(x)}, 1]$. □

We now show that, given a balancing scheme of parameter adjustment with proportional co-variation, the *CD*-distance between the original and adjusted probability distributions is bounded by the extent of change $\alpha(x)$ of the actively varied parameter x.

Proposition 3. *Let $\mathcal{B}$ be a Bayesian network in general. Let $(x \parallel \mathbf{x}^-, \mathbf{x}^+)$ be a balancing scheme of parameter adjustment with the parameters $\mathbf{x} = \{x, \mathbf{x}^-, \mathbf{x}^+\}$ from the conditional probability table of a single variable. Let $\alpha(x)$ be the extent of change of the parameter x. Then, the CD-distance between the original and adjusted probability distributions $\Pr^o$ and $\Pr$ represented by $\mathcal{B}$ is bounded by*

$$CD(\Pr^o, \Pr) \;=\; \ln(\alpha(x))$$

if just a single parameter x is varied actively, and by

$$CD(\Pr^o, \Pr) \;\leq\; 2 \cdot \ln(\alpha(x))$$

if more than one parameter is varied actively.

Proof. Given a change of the value of a single parameter x and proportional variation of its co-varying parameters, we have that the *CD*-distance between the original and adjusted joint distributions equals

$$\ln \max\Big\{\frac{x}{x^o}, \frac{1-x}{1-x^o}\Big\} - \ln \min\Big\{\frac{x}{x^o}, \frac{1-x}{1-x^o}\Big\} \;=\; \ln(\alpha(x))$$

When multiple parameters from the conditional probability table of a single variable V_i are changed, the distance induced by the adjustment equals

$$\ln \max_{z \in \Pr(V_i \,|\, \mathbf{Pa}(V_i))} \Big(\frac{z}{z^o}\Big) - \ln \min_{z \in \Pr(V_i \,|\, \mathbf{Pa}(V_i))} \Big(\frac{z}{z^o}\Big)$$

where z may be an actively varied or a co-varying parameter in a local distribution over V_i. Given proportional co-variation, we now have by Lemma 1 that all elements z/z^o in the expression above are in the interval $\big[\frac{1}{\alpha(x)}, \alpha(x)\big]$. We thus find that $CD(\Pr^o, \Pr) \;\leq\; 2 \cdot \ln(\alpha(x))$. □

Upon adjusting multiple parameters from multiple conditional probability tables, the overall effect on the represented joint probability distribution cannot exceed the sum of the effects induced by the changes in the separate tables. Based on this observation, the following corollary now summarises an upper bound on the *CD*-distance between the original and adjusted joint probability distributions after multiple parameter changes in general.

Corollary 2. *Let $\mathcal{B}$ be a Bayesian network in general, representing the joint probability distribution $\Pr^o$; let $\mathbf{x}$ be (a subset of) the parameters of $\mathcal{B}$, which are varied by an appropriate balancing scheme in the parameter $x \in \mathbf{x}$ with proportional co-variation. Let $\alpha(x)$ be the extent of change of x, and let $\Pr$ be the adjusted probability distribution. Then,*

$$CD(\Pr^o, \Pr) \;\leq\; k \cdot \ln(\alpha(x))$$

where $k = s + 2 \cdot t$, with s the number of probability tables with just a single parameter in $\mathbf{x}$ and t the number of tables with two or more parameters in $\mathbf{x}$.

As stated in the proposition above, if just a single parameter of a CPT is actively varied, the *CD*-distance induced by the change is exactly $\ln(\alpha(x))$. Now recall from Section 2.3 that, upon varying parameters from multiple probability tables, the *CD*-distance between an original distribution $\Pr^o$ and the distribution $\Pr$ after parameter adjustment is equal to this sum if the variables to which the tables pertain do not have a direct child–parent relationship nor share any common parents. Under these specific conditions therefore, the upper bound mentioned in the corollary is tight and, hence, $CD(\Pr^o, \Pr) = k \cdot \ln(\alpha(x))$.

6. Examples

We present two elaborate examples of tuning the output probabilities of a multi-dimensional Bayesian network classifier. Our first example pertains to a small, artificially constructed classifier and serves to illustrate some of the properties discussed in the previous sections. Our second example is taken from a real-world setting and serves to demonstrate the practicability of our approach.

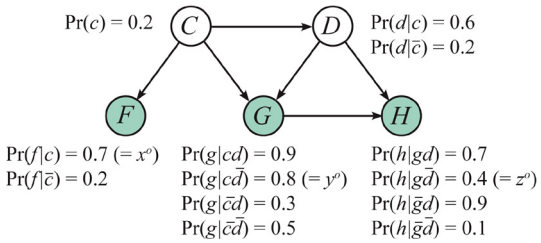

Fig. 4. An example multi-dimensional Bayesian network classifier.

6.1. An artificial example

We consider the multi-dimensional Bayesian network classifier from Fig. 4, and address its output probability $\Pr(cd\,|\,fgh)$. With the original parameter values from the classifier, this output probability is computed to be $\Pr(cd\,|\,fgh) = 0.51$. Now suppose that domain experts indicate that this probability value is too small and should in fact be as large as 0.70. Further suppose that we would like to arrive at this larger value by adjusting the values of the three parameters $x = \Pr(f\,|\,c)$, $y = \Pr(g\,|\,c\bar{d})$ and $z = \Pr(h\,|\,g\bar{d})$ of the feature variables F, G and H, respectively. By Proposition 1, we find that the sensitivity function expressing the output probability in terms of these three parameters equals

$$\begin{aligned}\Pr(cd\,|\,fgh)(x,y,z) &= \\ &= \frac{p_1^o \cdot x \cdot y^o \cdot z^o}{p_1^o \cdot x \cdot y^o \cdot z^o + p_2^o \cdot x \cdot y \cdot z + p_3^o \cdot x^o \cdot y^o \cdot z^o + p_4^o \cdot x^o \cdot y^o \cdot z} \\ &= \frac{1.64 \cdot x}{1.64 \cdot x + 1.74 \cdot x \cdot y \cdot z + 0.15 + 1.39 \cdot z}\end{aligned}$$

where

$$\begin{aligned} p_1^o &= \Pr^o(cd\,|\,fgh) = 0.51 \\ p_2^o &= \Pr^o(c\bar{d}\,|\,fgh) = 0.17 \\ p_3^o &= \Pr^o(\bar{c}d\,|\,fgh) = 0.07 \\ p_4^o &= \Pr^o(\bar{c}\bar{d}\,|\,fgh) = 0.25 \end{aligned}$$

This function attains its maximum value $\Pr(cd\,|\,fgh)(x,y,z) = 0.92$ at $x = 1$, $y = 0$ and $z = 0$.

From the higher-order sensitivity function $\Pr(cd\,|\,fgh)(x,y,z)$, we now derive a guaranteed monotone balanced sensitivity function by appropriately balancing the parameters y and z with the parameter x. Since $x \sim \Pr(cd\,|\,fgh)$ and $y, z \nsim \Pr(cd\,|\,fgh)$, we balance both parameters y and z negatively with x, to guarantee that the output probability retains the same value range as the corresponding higher-order sensitivity function. The balanced sensitivity function $\Pr(cd\,|\,fgh)(x \parallel y^-, z^-)$ thus found, equals

$$\Pr(cd\,|\,fgh)(x \parallel y^-, z^-) = \frac{6.43 \cdot x - 8.04 \cdot x^2 + 2.05 \cdot x^3}{6.02 + 2.22 \cdot x - 16.62 \cdot x^2 + 8.86 \cdot x^3}$$

Fig. 5 depicts the function. The extreme values of the sensitivity function are $\Pr(cd\,|\,fgh)(x \parallel y^-, z^-) = 0$ for $x = 0$ and $\Pr(cd\,|\,fgh)(x \parallel y^-, z^-) = 0.92$ for $x = 1$. We note that the desired value 0.70 for the output probability $\Pr(cd\,|\,fgh)$ can indeed be attained by adjusting the three parameters under study using the balancing scheme $(x \parallel y^-, z^-)$. In fact, the desired output probability is found at $x = 0.86$; the other two parameters then take the values $y = 0.60$ and $z = 0.20$. To illustrate the effect of an inappropriate balancing scheme with parameters related in a way which is not guaranteed monotone, Fig. 5 depicts, in addition to the function $\Pr(cd\,|\,fgh)(x \parallel y^-, z^-)$, also the balanced sensitivity function $\Pr(cd\,|\,fgh)(x \parallel y^+, z^+)$ in which the parameters y and z are inappropriately balanced with x. The figure shows that the sensitivity function for the output probability under study resulting from this latter balancing scheme is not monotone.

The adjustment found upon tuning the output probability $\Pr(cd\,|\,fgh)$ with the balancing scheme $(x \parallel y^-, z^-)$, has an extent of change equal to $\alpha(x = 0.86) = 2.67$. As we adjusted a single parameter from three probability tables each, we find from Proposition 3 that the effect on the joint probability distribution is bounded by $CD(\Pr^o, \Pr) \leq 3 \cdot \ln(2.67) = 2.94$. For this small example network, we could establish the true CD-distance between the original and adjusted distributions to be equal to $CD(\Pr^o, \Pr) = 1.95$, which suggests that the upper bound from Proposition 3 may be somewhat conservative.

For attaining the desired value 0.70 for the output probability of interest, also other combinations of parameters could have been adjusted. Varying other parameter combinations will generally result in another extent of change and hence in another distance between the original probability distribution and the distribution after tuning. For example, the desired value for the output probability is also attained with the values $\Pr(f\,|\,\bar{c}) = 0.09$, $\Pr(g\,|\,\bar{c}\bar{d}) = 0.28$ and $\Pr(h\,|\,gd) = 0.86$ for the three parameters involved. This particular adjustment has an extent of change equal to $\alpha(\Pr(f\,|\,\bar{c}) = 0.09) = 2.59$, from which we find that the effect on the represented probability distribution is bounded by $CD(\Pr^o, \Pr) \leq 3 \cdot \ln(2.59) = 2.85$.

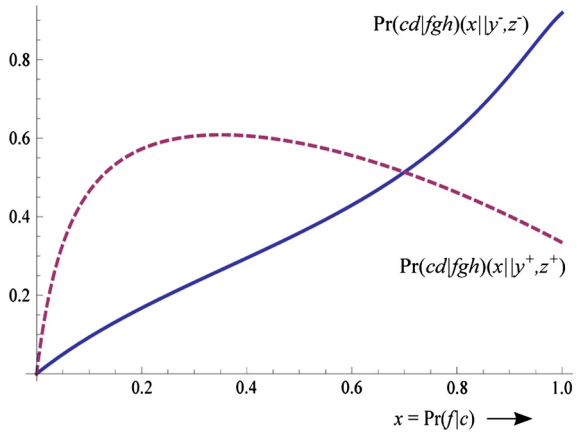

Fig. 5. The balanced sensitivity function $\Pr(cd\,|\,fgh)(x\|y^-, z^-)$ (solid) from the guaranteed monotone balancing scheme $(x\|y^-, z^-)$ and the function $\Pr(cd\,|\,fgh)(x\|y^+, z^+)$ (dashed) from the balancing scheme $(x\|y^+, z^+)$ which is not guaranteed monotone.

Table 1
The proportion of classifiers in which the output probability $\Pr^o(cd\,|\,fgh)$ could be tuned to values with a log-odds ratio of 1 and 2 respectively, through the parameter x, the parameter combination x, y, and the combination x, y, z, given 100 parameter settings for the classifier from Fig. 4.

	log odds ratio	
	1	2
x	40/100	8/100
x, y	59/100	21/100
x, y, z	63/100	25/100

Again, we could establish the true *CD*-distance between the original and adjusted distributions, which was found to be equal to $CD(\Pr^o, \Pr) = 1.87$. Based on the upper bound, and in fact on the true distance of the adjusted distribution from the original one, the second tuning option would be preferred.

By means of our example, we now further illustrate that tuning by adjusting multiple parameters serves to increase the tuning range, that is, the range of values which can be attained for an output probability of interest. We consider again the output probability $\Pr(cd\,|\,fgh)$ of our multi-dimensional classifier. By varying just the parameter $x = \Pr(f\,|\,c)$, we find the one-way sensitivity function

$$\Pr(cd\,|\,fgh)(x) = \frac{1.64 \cdot x}{0.70 + 2.20 \cdot x}$$

which has its maximum $\Pr(cd\,|\,fgh)(x) = 0.57$ at $x = 1$. By tuning through just the parameter x, we can thus attain any desired value from the interval $\langle 0, 0.57\rangle$ for the output probability. By varying x and balancing the parameter $y = \Pr(g\,|\,c\bar{d})$ negatively with x, we find the sensitivity function

$$\Pr(cd\,|\,fgh)(x \parallel y^-) = \frac{4.60 \cdot x - 4.10 \cdot x^2}{1.97 + 4.79 \cdot x - 6.05 \cdot x^2}$$

which has its maximum $\Pr(cd\,|\,fgh)(x\|y^-) = 0.70$ at $x = 1$. Analogously, we find that the value 0.92 can be attained for the output probability by employing the balancing scheme $(x \parallel y^-, z^-)$ with the parameters x, y and $z = \Pr(h\,|\,g\bar{d})$.

To provide an indication of the differences in tuning range between tuning with just x, with x and y, and with x, y and z, we randomly generated a total of 100 parameter settings for our example classifier and established in how many of those we could tune the output probability $\Pr(cd\,|\,fgh)$ to values for which the log-odds ratio

$$\ln\left(\frac{\Pr(cd\,|\,fgh)}{\Pr^o(cd\,|\,fgh)} \cdot \frac{1 - \Pr^o(cd\,|\,fgh)}{1 - \Pr(cd\,|\,fgh)}\right)$$

equals 1 and 2, respectively; the results are reported in Table 1. We observe that increasing the number of parameters for tuning indeed serves to increase the tuning range for the output probability of interest.

To investigate whether tuning through multiple parameters would result in a smaller distance between the original and adjusted probability distributions, we further determined for our example classifier, the distance $CD(\Pr^o, \Pr)$ as a function of the desired value of the output probability $\Pr(cd\,|\,fgh)$ through tuning with the parameter x, with the parameter combination

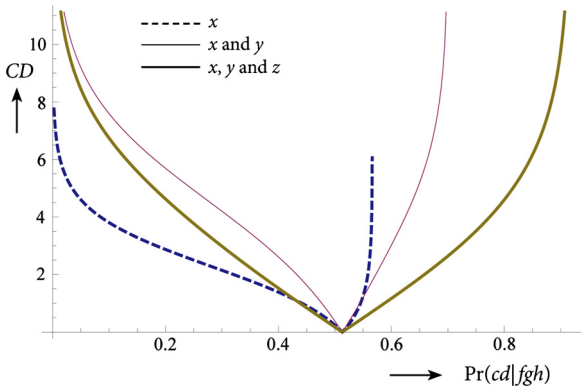

Fig. 6. The distance $CD(\Pr^o, \Pr)$ as a function of the desired value of the output probability $\Pr(cd\,|\,fgh)$, given guaranteed monotone changes of x, of x, y, and of x, y, z, respectively, in the classifier from Fig. 4.

x, y, and with x, y, z; the resulting functions are depicted in Fig. 6. We observe that for tuning $\Pr(cd\,|\,fgh)$ to a higher value, adding the parameter y, respectively the parameter combination y, z, to the balancing scheme indeed results in a smaller CD-distance between the original and adjusted probability distributions. In tuning to a value below 0.43 however, a more levelled change is found upon tuning with just the parameter x. This may be explained by the relatively small contribution of increasing the parameters y and z, to the decrease of $\Pr(cd\,|\,fgh)$ when compared to the contribution of the decrease of x.

To conclude, we recall that in our previous work [3], we proposed sliced sensitivity functions and indicated the use of such functions for tuning purposes. A sliced sensitivity function specifies an output probability of a Bayesian network in multiple linearly related parameters where the linear relationship for tuning purposes is determined by the gradient vector of the underlying higher-order sensitivity function given the original parameter values. By means of our example, we now illustrate the advantage of balanced sensitivity functions over sliced sensitivity functions. In contrast to appropriately balanced functions, sliced sensitivity functions are not guaranteed to cover the same output range as the corresponding higher-order functions. For example, capturing the output probability $\Pr(cd\,|\,fgh)$ as a function of the parameter x, with y changing linearly with x along the gradient of $\Pr(cd\,|\,fgh)(x, y)$ in (x^o, y^o), results in the sliced sensitivity function

$$\Pr(cd\,|\,fgh)(x \parallel y^s) = \frac{1.64 \cdot x}{0.70 + 2.43 \cdot x - 0.34 \cdot x^2}$$

This function attains its maximum $\Pr(cd\,|\,fgh)(x \parallel y^s) = 0.59$ at $x = 1$. The balanced sensitivity function which resulted from negatively balancing the parameter y with x however, had at $x = 1$ its maximum $\Pr(cd\,|\,fgh))(x \parallel y^-) = 0.7$.

6.2. A real-world example

In a European project, we had previously developed a Bayesian network for the early detection of Classical Swine Fever, or CSF for short, in pigs, in close collaboration with veterinary researchers and practitioners [11]. Classical Swine Fever is a highly infectious pig disease with a potential for rapid spread and with major socio-economic consequences upon an outbreak; CSF is a notifiable disease, which means that any suspicion of its presence should be reported immediately to the agricultural authorities and control measures should be installed. For the present paper, we constructed, from the Classical Swine Fever network, a multi-dimensional Bayesian network classifier. The resulting classifier is depicted in Fig. 7. The class variables CSF_i, $i = 1, \ldots, 5$, denote the five phases which are commonly distinguished in the progression of an infection with Classical Swine Fever in individual animals. Each of these phases is associated with one or more clinical symptoms which are known to more or less persist over the subsequent phases of the disease. The symptoms are modelled by the classifier's feature variables; these variables are shown in the figure in the order in which the symptoms tend to show in a diseased animal. The conditional probability distributions for the class and feature variables were computed from the original Bayesian network, conditioned on environmental factors not included in the classifier.

The original Bayesian network for Classical Swine Fever was constructed by hand with the help of veterinary experts throughout the European Union. During its construction, extensive case reviews were conducted with swine practitioners, both with and without clinical CSF experience. We now use one of these cases for tuning the output probabilities of the constructed multi-dimensional classifier. The case pertains to an animal showing mild symptoms from just the first phase of a CSF infection and two symptoms which may be associated with the two final phases of the disease; no signs are seen from the intermediate phases of an infection with Classical Swine Fever. During the case reviews, the experts had indicated that the probability of this animal actually having Classical Swine Fever would be quite small; the probability $\Pr(\bar{c}_1, \ldots, \bar{c}_5\,|\,Case)$, with $\bar{c}_i$ denoting the absence of a CSF infection of phase i, should thus be quite high and was in fact estimated by the experts at 0.80. From the constructed multi-dimensional classifier however, this output probability was found to be equal to 0.17. For tuning the output probability of the classifier to the insight of the experts, we selected two parameters:

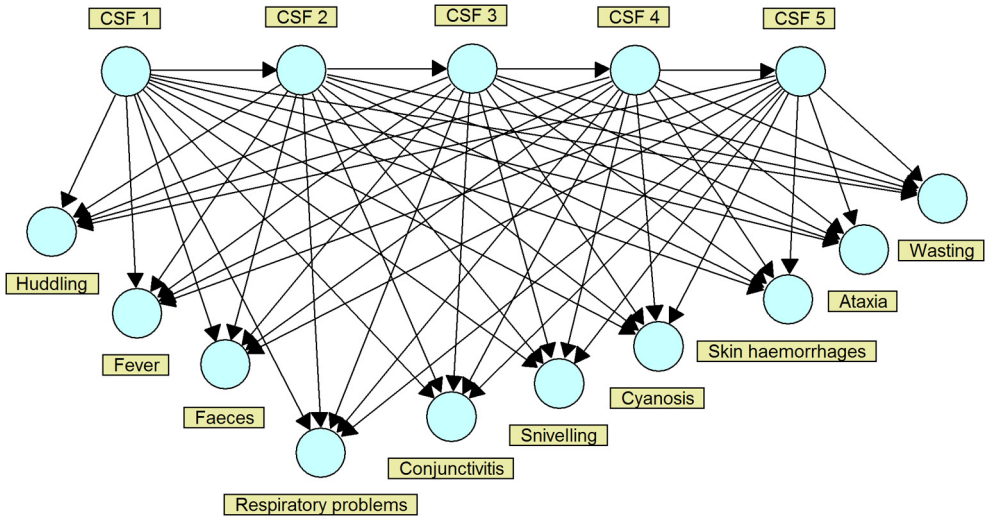

Fig. 7. A multi-dimensional Bayesian network classifier for Classical Swine Fever in pigs.

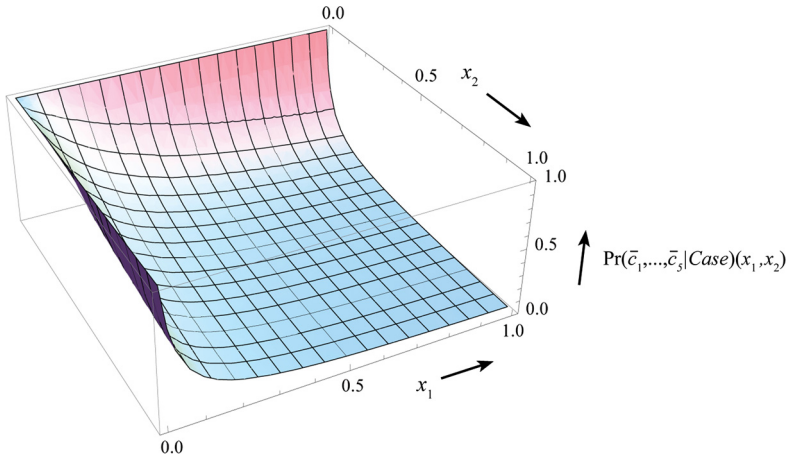

Fig. 8. The second-order sensitivity function $\Pr(\bar{c}_1, \ldots, \bar{c}_5 \,|\, Case)(x_1, x_2)$ of the multi-dimensional classifier for Classical Swine Fever from Fig. 7.

$$x_1 = \Pr(w \,|\, c_1, \ldots, c_5)$$
$$x_2 = \Pr(f_n \,|\, c_1, \ldots, c_5)$$

where w denotes the clinical symptom of *Wasting* being *present* and f_n indicates the value *normal* for the animal's *Faeces*. These two parameters were selected for tuning because the absence of abnormal faeces and the animal showing signs of wasting point in different directions with respect to a diagnosis of Classical Swine Fever. The original values of the selected parameters are $x_1^o = 0.90$ and $x_2^o = 0.21$. The second-order sensitivity function expressing the output probability of interest in these two parameters equals

$$\Pr(\bar{c}_1, \ldots, \bar{c}_5 \,|\, Case)(x_1, x_2) = \frac{p_1^o \cdot x_1^o \cdot x_2^o}{(1 - p_6^o) \cdot x_1^o \cdot x_2^o + p_6^o \cdot x_1 \cdot x_2} = \frac{0.3348}{0.3364 + 8.2460 \cdot x_1 \cdot x_2}$$

where $p_1 = \Pr(\bar{c}_1, \ldots, \bar{c}_5 \,|\, Case)$ with $p_1^o = 0.17$ and $p_6 = \Pr(c_1, \ldots, c_5 \,|\, Case)$ with $p_6^o = 0.82$. The function is depicted in Fig. 8, and ranges from the minimum value $\Pr(\bar{c}_1, \ldots, \bar{c}_5 \,|\, Case)(x_1, x_2) = 0.039$ at $x_1 = 1$, $x_2 = 1$, to the maximum value $\Pr(\bar{c}_1, \ldots, \bar{c}_5 \,|\, Case)(x_1, x_2) = 0.995$ at $x_1 = 0$ or $x_2 = 0$ which shows that the desired value 0.80 for the output probability at hand indeed is attainable by tuning with x_1 and x_2.

To find values for the two parameters x_1 and x_2 such that the desired value $\Pr(\bar{c}_1, \ldots, \bar{c}_5 \,|\, Case) = 0.80$ for the output probability is attained, we construct a balanced sensitivity function for further manipulation. Since both parameters x_1, x_2 are incompatible with the class value combination $\bar{c}_1, \ldots, \bar{c}_5$ under study, the two parameters should be adjusted in the

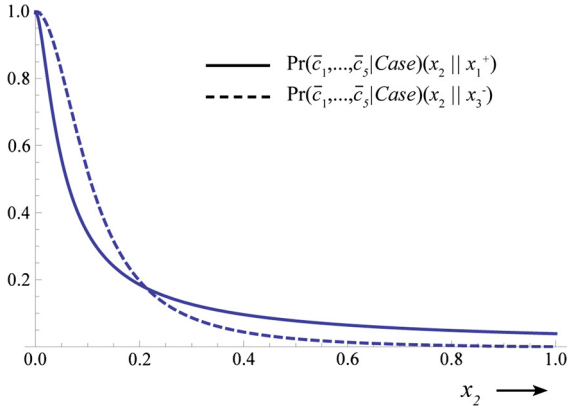

Fig. 9. The balanced sensitivity functions $\Pr(\bar{c}_1,\ldots,\bar{c}_5\,|\,Case)(x_2\|x_1^+)$ (solid) and $\Pr(\bar{c}_1,\ldots,\bar{c}_5\,|\,Case)(x_2\|x_3^-)$ (dashed), for the multi-dimensional classifier for Classical Swine Fever.

same direction and hence are balanced positively. The balanced sensitivity function in x_2 which results from this balancing scheme equals

$$\Pr(\bar{c}_1,\ldots,\bar{c}_5\,|\,Case)(x_2 \parallel x_1^+) = \frac{0.714 + 23.000\cdot x_2}{0.717 + 23.107\cdot x_2 + 583.990\cdot x_2^2}$$

The function is shown in Fig. 9. The output probability $\Pr(\bar{c}_1,\ldots,\bar{c}_5\,|\,Case) = 0.80$ is attained at $x_1 = 0.437$ and $x_2 = 0.023$. We note that compared to their original values, the two parameters have been adjusted quite strongly. The extent of change actually is as large as $\alpha(x_2 = 0.437) = 11.595$, from which we find that the distance between the original probability distribution and the adjusted one is bounded by $2\cdot\ln(\alpha(x_2)) = 4.90$.

For tuning the output probability of the classifier to the insight of the experts, we also consider another parameter combination:

$$\begin{aligned} x_2 &= \Pr(f_n\,|\,c_1,\ldots,c_5) \\ x_3 &= \Pr(s\,|\,\bar{c}_1,\ldots,\bar{c}_5) \end{aligned}$$

where x_2 is the same parameter as before, pertaining to the absence of abnormal faeces in the case under study, and the parameter $x_3 = \Pr(s\,|\,\bar{c}_1,\ldots,\bar{c}_5)$ pertains to the absence of skin haemorrhages, denoted by s, in the case; the original value of the parameter x_3 equals $x_3^o = 0.001$. Once more, these parameters have been selected because the absence of abnormal faeces and the animal showing skin haemorrhages point in different directions with respect to a diagnosis of Classical Swine Fever. In the multi-dimensional classifier, the clinical symptom of skin haemorrhages is modelled as a strong indication against the absence of a Classical Swine Fever infection. Since we are now tuning an output probability for the absence of CSF in the presence of skin haemorrhages, we expect a relatively strong effect from adjusting the parameter x_3. In fact, we expect that for this particular parameter combination tuning will require a smaller extent of change and will hence incur a smaller overall shift in the modelled distribution.

The second-order sensitivity function expressing the output probability of interest in the two parameters x_2 and x_3 equals

$$\Pr(\bar{c}_1,\ldots,\bar{c}_5\,|\,Case)(x_2,x_3) = \frac{0.037\cdot x_3}{1.71\cdot 10^{-7} + 0.001\cdot x_2 + 0.0037\cdot x_3}$$

To find values for the two parameters such that the desired value 0.80 for the output probability is attained, again a balanced sensitivity function is constructed for further manipulation. Since the two parameters have an opposite effect on the output probability, they should be adjusted in opposite directions and hence are balanced negatively. The balanced sensitivity function resulting from this balancing scheme equals

$$\Pr(\bar{c}_1,\ldots,\bar{c}_5\,|\,Case)(x_2 \parallel x_3^-) = \frac{7.961 - 7.961\cdot x_2}{7.961 - 7.645\cdot x_2 + 650.642\cdot x_2^2}$$

and is shown in Fig. 9 as well. The output probability $\Pr(\bar{c}_1,\ldots,\bar{c}_5\,|\,Case) = 0.80$ is attained at $x_2 = 0.054$ and $x_3 = 0.005$. The extent of this change equals $\alpha(x_2 = 0.054) = 4.79$. We note that the extent of change for this particular parameter combination is smaller than that for the parameter pair x_1, x_2, as expected. We find that the distance between the original probability distribution and the adjusted one now is bounded by $2\cdot\ln(\alpha(x_2)) = 3.13$.

7. Discussion and conclusions

Motivated by the observation that available data sets often prove problematic for learning multi-dimensional Bayesian network classifiers, we presented an elegant method for tuning their output probabilities of interest based on expert-provided insights. We showed that the topological properties and dedicated use of a multi-dimensional classifier induce higher-order sensitivity functions of restricted functional form which can be established efficiently. We further formalised a scheme of balanced parameter adjustment, by which a higher-order sensitivity function is reduced to an insightful single-parameter function which is readily used for further manipulation. The idea of balancing multiple parameters gave rise to a suitable heuristic for tuning, which was shown to incur changes within guaranteed bounds in all output probabilities over the class variables. We would like to note that given an output probability which pertains to a joint value assignment to all non-observed variables, that is, given an output MPE-probability, any Bayesian network can, by evidence absorption [12], be reduced to a network with the topological structure of a multi-dimensional classifier. The higher-order sensitivity function proposed for MDCs in this paper, and the results based on this function therefore in fact apply to output MPE-probabilities of Bayesian networks in general.

The tuning method developed in this paper does not as yet provide for selecting appropriate parameters for tuning. Parameter selection may be based upon various considerations. An example criterion may be to select parameters which give the smallest difference between the original and adjusted probability distributions. Yet, parameters may also be selected based on the sizes of the samples from which they were estimated originally. In the reviewed real-world example, we further took the underlying domain knowledge into consideration for selecting parameters for tuning. We plan to investigate the effects of these and other criteria in various applications of multi-dimensional network classifiers.

Although balanced tuning is a simple, generally applicable heuristic which gives a levelled change of the parameters involved, a balanced adjustment is not necessarily optimal with respect to minimising the distance between the original probability distribution and the distribution after tuning. As the heuristic is not tailored to the probability distribution at hand, it may well be that, upon attaining the same value for some output probability of interest, an increased contribution of a particular parameter change is more than compensated by a decrease of the contribution of another parameter change. In future research we would like to develop a more sophisticated heuristic for balanced tuning which incorporates knowledge of the distribution at hand.

Acknowledgements

This work was supported by the Netherlands Organisation for Scientific Research, grant number 612.066.928. We would like to thank the reviewers for their valuable comments which definitely helped to improve our paper.

Appendix

As review in Section 2.3, the *CD*-distance between an original distribution $\Pr^o$ and an adjusted distribution $\Pr$ bounds the odds ratio of any output probability $\Pr(\mathbf{w}\,|\,\mathbf{u})$ by

$$e^{-CD(\Pr^o,\Pr)} \leq \frac{O(\mathbf{w}\,|\,\mathbf{u})}{O^o(\mathbf{w}\,|\,\mathbf{u})} \leq e^{CD(\Pr^o,\Pr)}$$

The indicated bounds are not tight in general in the sense that in any case there would be a probability $\Pr(\mathbf{w}\,|\,\mathbf{u})$ for which

$$CD(\Pr^o,\Pr) = \ln\Big(\frac{\Pr^o(\mathbf{w}\,|\,\mathbf{u})}{1-\Pr^o(\mathbf{w}\,|\,\mathbf{u})} \cdot \frac{1-\Pr(\mathbf{w}\,|\,\mathbf{u})}{\Pr(\mathbf{w}\,|\,\mathbf{u})}\Big)$$

or

$$CD(\Pr^o,\Pr) = \ln\Big(\frac{\Pr(\mathbf{w}\,|\,\mathbf{u})}{1-\Pr(\mathbf{w}\,|\,\mathbf{u})} \cdot \frac{1-\Pr^o(\mathbf{w}\,|\,\mathbf{u})}{\Pr^o(\mathbf{w}\,|\,\mathbf{u})}\Big)$$

As an example, consider a Bayesian network with a binary variable A which has a three-valued variable B for its child, and original and adjusted conditional probability tables are as follows:

	Original value	New value
$\Pr(a_1)$	0.5	0.6
$\Pr(a_2)$	0.5	0.4
$\Pr(b_1\,|\,a_1)$	0.2	0.1
$\Pr(b_2\,|\,a_1)$	0.4	0.2
$\Pr(b_3\,|\,a_1)$	0.4	0.7
$\Pr(b_1\,|\,a_2)$	0.3	0.1
$\Pr(b_2\,|\,a_2)$	0.6	0.4
$\Pr(b_3\,|\,a_2)$	0.1	0.5

From the definition of the *CD*-distance

$$CD(\Pr^o, \Pr) \;=\; \ln \max_{a_i b_j \in \mathcal{AB}} \left(\frac{\Pr(a_i b_j)}{\Pr^o(a_i b_j)} \right) \;-\; \ln \min_{a_i b_j \in \mathcal{AB}} \left(\frac{\Pr(a_i b_j)}{\Pr^o(a_i b_j)} \right)$$

we find that

$$CD(\Pr^o, \Pr) = \ln \left(\frac{\Pr(a_2 b_3)}{\Pr^o(a_2 b_3)} \Big/ \frac{\Pr(a_2 b_1)}{\Pr^o(a_2 b_1)} \right) = \ln(15)$$

Systematically computing all odds ratios for the original and adjusted probabilities yields a maximum odds ratio of ln(9), however, found for the probability $\Pr(b_3 \,|\, a_2)$. For the example network therefore, the established *CD*-distance, of ln(15) is not reached by any output probability of interest.

References

[1] C. Bielza, G. Li, P. Larrañaga, Multi-dimensional classification with Bayesian networks, Int. J. Approx. Reason. 52 (2011) 705–727.
[2] J. De Bock, C.P. de Campos, A. Antonucci, Global sensitivity analysis for MAP inference in graphical models, in: Z. Ghahramani, M. Welling, C. Cortes, N.D. Lawrence, K.Q. Weinberger (Eds.), Advances in Neural Information Processing Systems, vol. 27, 2014, pp. 2690–2698.
[3] J.H. Bolt, S. Renooij, Local sensitivity of Bayesian networks to multiple parameter shifts, in: L.C. van der Gaag, A.J. Feelders (Eds.), Proceedings of the Seventh European Workshop on Probabilistic Graphical Models, 2014, pp. 65–80.
[4] H. Borchani, C. Bielza, C. Toro, P. Larrañaga, Predicting human immunodeficiency virus inhibitors using multi-dimensional Bayesian network classifiers, Artif. Intell. Med. 57 (2013) 219–229.
[5] H. Borchani, P. Larrañaga, J. Gama, C. Bielza, Mining multi-dimensional concept-drifting data streams using Bayesian network classifiers, Intell. Data Anal. 20 (2) (2016) 257–280.
[6] H. Chan, A. Darwiche, Sensitivity analysis in Bayesian networks: from single to multiple parameters, in: M. Chickering, J. Halpern (Eds.), Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence, AUAI Press, Arlington, VA, 2004, pp. 67–75.
[7] H. Chan, A. Darwiche, A distance measure for bounding probabilistic belief change, Int. J. Approx. Reason. 38 (2005) 149–174.
[8] G. Corani, A. Antonucci, D.D. Mauá, S. Gabaglio, Trading off speed and accuracy in multilabel classification, in: L.C. van der Gaag, A. Feelders (Eds.), Proceedings of the 7th European Workshop on Probabilistic Graphical Models, in: LNAI, vol. 8754, Springer, Berlin, 2014, pp. 145–159.
[9] V.M.H. Coupé, L.C. van der Gaag, Properties of sensitivity analysis of Bayesian belief networks, Ann. Math. Artif. Intell. 36 (2002) 323–356.
[10] P. Fernandez-Gonzalez, C. Bielza, P. Larrañaga, Multidimensional classifiers for neuroanatomical data, in: ICML Workshop on Statistics, Machine Learning and Neuroscience, 2015.
[11] L.C. van der Gaag, J. Bolt, W.L. Loeffen, A. Elbers, Modelling patterns of evidence in Bayesian networks: a case-study in Classical Swine Fever, in: E. Hüllermeier, R. Kruse, F. Hoffmann (Eds.), Computational Intelligence for Knowledge-Based Systems Design, in: Lecture Notes in Artificial Intelligence, vol. 6178, Springer-Verlag, Berlin, 2010, pp. 675–684.
[12] L.C. van der Gaag, On evidence absorption for belief networks, Int. J. Approx. Reason. 15 (1996) 265–286.
[13] L.C. van der Gaag, P.R. de Waal, Multi-dimensional Bayesian network classifiers, in: J. Vomlel, M. Studený (Eds.), Proceedings of the Third European Workshop on Probabilistic Graphical Models, 2006, pp. 107–114.
[14] A. Pastink, L.C. van der Gaag, Multi-classifiers of small treewidth, in: S. Destercke, Th. Denoeux (Eds.), Proceedings of the 13th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty, in: LNAI, vol. 9161, Springer, Berlin, 2015, pp. 199–209.
[15] J. Pearl, Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference, Morgan Kaufmann Publishers, Palo Alto, 1988.
[16] S. Renooij, Co-variation for sensitivity analysis in Bayesian networks: properties, consequences and alternatives, Int. J. Approx. Reason. 55 (2014) 1022–1042.
[17] J.D. Rodriguez, A. Perez, D. Arteta, D. Tejedor, J.A. Lozano, Using multidimensional Bayesian network classifiers to assist the treatment of multiple sclerosis, IEEE Trans. Syst. Man Cybern. 42 (2012) 1705–1715.
[18] P.R. de Waal, L.C. van der Gaag, Inference and learning in multi-dimensional Bayesian network classifiers, in: K. Mellouli (Ed.), Proceedings of the European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty, in: LNCS, vol. 4724, Springer, Berlin, 2007, pp. 501–511.