Balanced Tuning of Multi-dimensional Bayesian Network Classifiers

Janneke H. Bolt(✉) and Linda C. van der Gaag

Department of Information and Computing Sciences, Utrecht University,
P.O. Box 80.089, 3508 TB Utrecht, The Netherlands
{j.h.bolt,l.c.vandergaag}@uu.nl

Abstract. Multi-dimensional classifiers are Bayesian networks of restricted topological structure, for classifying data instances into multiple classes. We show that upon varying their parameter probabilities, the graphical properties of these classifiers induce higher-order sensitivity functions of restricted functional form. To allow ready interpretation of these functions, we introduce the concept of balanced sensitivity function in which parameter probabilities are related by the odds ratios of their original and new values. We demonstrate that these balanced functions provide a suitable heuristic for tuning multi-dimensional Bayesian network classifiers, with guaranteed bounds on the changes of all output probabilities.

1 Introduction

The family of multi-dimensional Bayesian network classifiers (MDCs) was introduced to generalise one-dimensional classifiers to application domains that require instances to be classified into multiple dimensions [6,9]. 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 amounts to computing the joint probability distribution over the class variables given the instance's features, and returning the most likely joint class combination. MDCs enjoy a growing interest as a suitable tool for multi-dimensional classification [1,4].

Like more traditional classifiers, multi-dimensional Bayesian network classifiers are typically learned from data. Tailored algorithms are available for fitting MDCs to the joint probability distributions reflected in the data at hand. While often available data prove suboptimal already for constructing a one-dimensional classifier, any skewness properties of the joint or conditional distributions over the class variables will prove especially problematic for learning multi-dimensional classifiers. Expert knowledge, for example of expected classifications, can then be instrumental in correcting unwanted biases by careful tuning of the parameter probabilities of the learned classifier.

Tuning the parameters of a multi-dimensional classifier requires detailed insight in the effects of changing their values on the classifier's output probabilities. For Bayesian networks in general, the technique of sensitivity analysis

S. Destercke and T. Denoeux (Eds.): ECSQARU 2015, LNAI 9161, pp. 210–220, 2015.
DOI: 10.1007/978-3-319-20807-7_19

has evolved as a practical tool for studying the effects of changes in a network's parameter probabilities. Research so far has focused on one-way sensitivity analyses in which a single parameter is varied. The effects of systematic variation of multiple parameters have received far less attention, mostly due to the computational burden of establishing the functions describing these effects. A recent exception is [2] in which an efficient algorithm for studying the effects of multiple changes, within a fixed interval, on an established MPE is given. For tuning the parameters of a multi-dimensional classifier however, more detailed insights in the effects of simultaneously varying multiple parameters is preferred or even necessary.

In this paper, we present an elegant method for tuning the output probabilities of a multi-dimensional Bayesian network classifier by simultaneous parameter adjustment. We begin by showing that the topological properties of an MDC induce higher-order sensitivity functions of restricted functional form which can be established efficiently. By employing a carefully balanced scheme of parameter adjustment, such a function is reduced to an insightful single-parameter balanced sensitivity function which can be readily exploited as a suitable heuristic for tuning. The heuristic is shown to incur changes within guaranteed bounds in all output probabilities over the class variables, thereby providing global insight in the change in the network's output distributions.

The paper is organised as follows. In Sect. 2 we review multi-dimensional classifiers, and sensitivity functions of Bayesian networks in general. In Sect. 3 we derive the general form of a higher-order sensitivity function for MDCs. In Sect. 4, the concept of balanced sensitivity function is introduced; we describe how such a function is used for effective parameter tuning in a multi-dimensional classifier and prove bounds on the changes induced in all output probabilities. Section 5 illustrates the basic idea of balanced parameter tuning by means of an example, and Sect. 6 concludes the paper.

2 Preliminaries

We briefly review multi-dimensional classifiers and thereby introduce our notations. We further describe higher-order sensitivity functions for Bayesian networks in general.

2.1 Bayesian Networks and Multi-dimensional Classifiers

We consider a set of 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; we will write v and $\bar{v}$ for the two values of a binary variable V. A joint value assignment to $\mathbf{V}$ is indicated by $\mathbf{v}$. In the sequel, we will use V_i and $\mathbf{V}$ also to indicate the set of possible value assignments to V_i and $\mathbf{V}$, respectively.

A Bayesian network is a graphical model of a joint probability distribution Pr over a set of random variables $\mathbf{V}$. Each variable from $\mathbf{V}$ 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 [7]. Each variable $V_i \in \mathbf{V}$ further has associated a set of conditional probability distributions $\Pr(V_i \mid \pi_{V_i})$, where π_{V_i} denotes the set of parents of V_i in the graph; the separate probabilities in these distributions are termed the network's parameters. The joint probability distribution Pr now factorises over the network's graph as

$$\Pr(\mathbf{V}) = \prod_{V_i \in \mathbf{V}} \Pr(V_i \mid \pi_{V_i})$$

where V_i and π_{V_i} take their value assignments compatible with $\mathbf{V}$. We will use $\sim$ and $\nsim$ to indicate compatibility and incompatibility of value assignments, respectively.

A multi-dimensional classifier now is a Bayesian network of restricted topology. Its set of variables is partitioned into a set $\mathbf{C}$ of class variables and a set $\mathbf{F}$ of feature variables, and its digraph does not allow the feature variables to have class children [6,9]. An MDC is used to assign a joint value assigment, or instance, $\mathbf{f}$ to a most likely combination of class values $\mathbf{c}$, that is, it is used to establish $\mathrm{argmax}_{\mathbf{c}} \Pr(\mathbf{c} \,|\, \mathbf{f})$. In this paper we focus specifically on classifiers without any direct relationships between their class variables, yet in which no further topological assumptions are made; we will denote such classifiers by $MDC(\mathbf{C}, \mathbf{F})$. For a feature variable $F_i \in \mathbf{F}$, we will use $\mathbf{F}_{F_i} = \mathbf{F} \cap \pi_{F_i}$ to denote its set of feature parents, and $\mathbf{C}_{F_i}$ to denote its parents from $\mathbf{C}$. Specific value assignments to these sets are indicated by $\mathbf{f}_{F_i}$ and $\mathbf{c}_{F_i}$ respectively.

2.2 Sensitivity Functions of Bayesian Networks

Upon systematically varying multiple parameter probabilities $\mathbf{x} = \{x_1, \ldots, x_n\}$, $n \geq 1$, of a Bayesian network in general, a higher-order sensitivity function results which expresses an output probability $\Pr(\mathbf{y} \mid \mathbf{e})$ of interest in terms of these parameters $\mathbf{x}$. More specifically, the result is a function of the following form:

$$\Pr(\mathbf{y} \mid \mathbf{e})(\mathbf{x}) = \frac{\sum_{\mathbf{x}_k \in \mathcal{P}(\mathbf{x})} \left(c_k \cdot \prod_{x_i \in \mathbf{x}_k} x_i\right)}{\sum_{\mathbf{x}_k \in \mathcal{P}(\mathbf{x})} \left(d_k \cdot \prod_{x_i \in \mathbf{x}_k} x_i\right)}$$

where $\mathcal{P}(\mathbf{x})$ is the powerset of the set of network parameters $\mathbf{x}$ and where the constants c_k, d_k are determined by the non-varied network parameters. We will use $\mathbf{x^o} = \{x_1^o, \ldots, x_n^o\}$ to indicate the original values of the parameters $\mathbf{x}$ in the network under study, $\Pr^o$ to indicate original probabilities, that is, probabilities computed with the original values of all parameters involved, and O^o to indicate original odds.

Upon varying a parameter x_j for a variable V_i, the other parameters of the same conditional distribution over V_i are co-varied to let the distribution sum to 1. In the most commonly used co-variation scheme, these parameters are varied proportionally with x_j. Since other schemes may also be appropriate [8], we will formulate our results in the sequel without assuming any specific scheme of co-variation.

3 The n-way Sensitivity Function of an MDC

Establishing a higher-order sensitivity function for a Bayesian network in general is computationally expensive, 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. In this section, we show that, due to its restricted topological structure and dedicated use, a multi-dimensional classifier allows more ready calculation of the n-way sensitivity functions for its output probabilities. We show more specifically, that an output probability $\Pr(\mathbf{c} \,|\, \mathbf{f})$ for a given $\mathbf{c}$ can be expressed in terms of the original output probability and the original and new values of all parameters compatible with the instance $\mathbf{f}$. The form of the sensitivity function is given in the proposition below; the proofs of all propositions in this paper are provided in the appendix.

Proposition 1. *Let* $MDC(\mathbf{C}, \mathbf{F})$ *be a multi-dimensional classifier as defined above. Let* $\mathbf{f}$ *be an instance of* $\mathbf{F}$*, and let* $\mathbf{x} = \{x_1, \ldots, x_n\}$*,* $n \geq 1$*, be the set of network parameters compatible with* $\mathbf{f}$*. Then, for all* $\mathbf{c} \in \mathbf{C}$*,*

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

The sensitivity function $\Pr(\mathbf{c} \,|\, \mathbf{f})(\mathbf{x})$ stated above includes all parameters of the feature variables which are compatible with the instance $\mathbf{f}$ to be classified. The parameters $\Pr(f_i' \mid \pi_{F_i})$ of a feature variable F_i with f_i' incompatible with $\mathbf{f}$ do not occur in the function since these parameters are not involved directly in the computation of the output probability: upon variation of such a parameter, the output probability is affected only indirectly by the co-variation of $\Pr(f_i \mid \pi_{F_i})$ with $f_i \sim \mathbf{f}$. Without loss of generality, we thus include just the parameters compatible with $\mathbf{f}$, which implies that the proposition holds for any co-variation scheme used for the parameters of the feature variables. Also all parameters $\Pr(c_i)$ of a class variable C_i are included in the sensitivity function. These parameters cannot be varied freely however, as their sum should remain 1. By assuming a specific co-variation scheme, we could have included the dependent parameters implicitly, as with the feature parameters. By their explicit inclusion, however, the function is independent of the co-variation scheme used for the class parameters and can be further tailored to a specific scheme upon practical application.

Although the function stated above includes all parameters compatible with the instance to be classified, it is easily adapted to a sensitivity function involving only 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 cancels out. The sensitivity function is also readily adapted to output probabilities $\Pr(\mathbf{c} \mid \mathbf{g})$ with $\mathbf{G} \subset \mathbf{F}$, provided there are no observed feature variables with unobserved feature parents. The parameters of the unobserved feature variables then should simply be excluded.

The sensitivity function stated in Proposition 1 reveals that an output probability of a multi-dimensional classifier changes monotonically with specific parameter adjustments. Proposition 2 details this property of monotonicity.

Proposition 2. *Let MDC*$(\mathbf{C},\mathbf{F})$ *be a classifier as before, and let* $\Pr(\mathbf{c}\mid\mathbf{f})$ *be its output probability of interest. Let* $\mathbf{x}=\{x_1\ldots,x_n\}$, $n\geq 1$, *be the parameters of MDC*$(\mathbf{C},\mathbf{F})$ *compatible with* $\mathbf{f}$, *and let* $\mathbf{x}'$, $\mathbf{x}^*$ *be two sets of values for these parameters. Then,*

$$x_i'\leq x_i^* \textit{ for all } x_i\sim\mathbf{c} \;\textit{ and }\; x_j'\geq x_j^* \textit{ for all } x_j\not\sim\mathbf{c} \Leftrightarrow \Pr(\mathbf{c}\mid\mathbf{f})(\mathbf{x}')\leq\Pr(\mathbf{c}\mid\mathbf{f})(\mathbf{x}^*)$$

The proposition states that by increasing the parameters in $\mathbf{x}$ compatible with $\mathbf{c}$ and decreasing the incompatible ones, the output probability of the class combination $\mathbf{c}$ increases. Such a parameter change will be called monotone with respect to the output probability $\Pr(\mathbf{c}\,|\,\mathbf{f})$. We note that the monotonicity property of a parameter change provides information about the direction in which the separate parameters need to be adjusted to arrive at the intended effect on the output probability. The following corollary states that this probability takes its maximum at the parameters' extreme values.

Corollary 1. *Let MDC*$(\mathbf{C},\mathbf{F})$, $\Pr(\mathbf{c}\mid\mathbf{f})$ *and* $\mathbf{x}$ *be as before. The sensitivity function* $\Pr(\mathbf{c}\mid\mathbf{f})(\mathbf{x})$ *attains its maximum at* $x_i=1$ *for all* $x_i\sim\mathbf{c}$ *and* $x_j=0$ *for all* $x_j\not\sim\mathbf{c}$. *A similar property holds for the minimum of the function.*

4 Balanced Tuning of MDCs

In the previous section we showed that the output probability $\Pr(\mathbf{c}\mid\mathbf{f})$ of a multi-dimensional classifier changes monotonically given a monotone parameter adjustment. While this property indicates the direction in which parameters have to be adjusted, it does not yet suggest the amount of adjustment for arriving at the intended output. We now introduce for this purpose the concept of a balancing scheme for parameter adjustment. A balancing scheme governs a simultaneous change in all parameters x involved, by amounts defined by their odds ratios $\frac{x^o\cdot(1-x)}{(1-x^o)\cdot x}$. Balancing the parameters of a classifier constitutes a simple and generally applicable approach to parameter tuning; we will show moreover that the approach comes with guaranteed bounds on the changes of all possible output probabilities. We now first define the concept of balancing scheme.

Definition 1. *Let* $x,y\in\langle 0,1\rangle$ *be parameters of an MDC, and let* x^o *and* y^o *be their original values. We say that a scheme for parameter adjustment balances* y *positively with* x *if* $\frac{x^o\cdot(1-x)}{(1-x^o)\cdot x}=\frac{y^o\cdot(1-y)}{(1-y^o)\cdot y}$; *it balances* y *negatively with* x *if* $\frac{x^o\cdot(1-x)}{(1-x^o)\cdot x}=\frac{(1-y^o)\cdot y}{y^o\cdot(1-y)}$.

An important property of a balancing scheme for parameter adjustment 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 of y is not constrained by balancing y with x; this property is illustrated for $x^o=0.7$

and $y^o = 0.8$ in Fig. 1. We note that we assume that a balancing scheme does not adjust deterministic parameters and that non-deterministic parameters will not adopt deterministic values.

Building upon balancing schemes, we now define a balanced sensitivity function[1].

Definition 2. *Let* $\Pr(\mathbf{c}\,|\,\mathbf{f})$ *be the output probability of an MDC as before, and let* $\mathbf{x} = \{x_1, \ldots, x_n\}$*,* $n \geq 1$*, be its parameters compatible with* $\mathbf{f}$*. A balanced sensitivity function for* $\Pr(\mathbf{c} \mid \mathbf{f})$ *is a function* $\Pr(\mathbf{c} \mid \mathbf{f})(x_i)$ *in a single parameter* $x_i \in \mathbf{x}$*, with all parameters* $x_j \in \mathbf{x}$ *balanced with* x_i*.*

A balanced function $\Pr(\mathbf{c} \mid \mathbf{f})(x_i)$ is the intersection of the n-way function $\Pr(\mathbf{c} \mid \mathbf{f})(\mathbf{x})$ with the (curved) surface defined by the balancing scheme. It takes the following form:

$$\Pr(\mathbf{c} \mid \mathbf{f})(x_i) = \frac{c_0 + c_1 \cdot x_i^1 + \ldots + c_m \cdot x_i^m}{d_0 + d_1 \cdot x_i^1 + \ldots + d_m \cdot x_i^m}$$

where the constants c_j, d_j again are determined by the non-varied parameters, each x_i^k is a multiplicative term of degree k, and m is the number of probability tables from which the parameters are chosen. As an example, Fig. 2 depicts the two-way sensitivity function $\Pr(cd\,|\,fgh)(x, y)$ of the MDC from Fig. 3, in the two parameters $x = \Pr(f\,|\,c)$ and $y = \Pr(g\,|\,c\bar{d})$. The figure further depicts the two surfaces determining the balanced sensitivity functions in x and in y separately, that are derived from the two-way function given a positive and a negative balancing scheme for the two parameters.

A balanced sensitivity function provides insight in the effects of varying multiple parameter probabilities according to a balanced scheme of adjustment. For a required change in the output probability of interest $\Pr(\mathbf{c} \mid \mathbf{f})$, the amount by which the parameter x_i is to be adjusted is readily established; the balanced scheme of adjustment then enforces the other parameter probabilities to be adjusted accordingly. To guarantee that the balanced sensitivity function covers the same value range for the output probability as the underlying n-way function, all parameters have to be balanced monotonically with the output probability of interest.

Given a (not necessarily monotone) balanced change, the changes incurred in all output probabilities over the class variables are bounded, in terms of the odds ratio of the original and new probabilities, as stated in the following proposition.

Proposition 3. *Let* $MDC(\mathbf{C}, \mathbf{F})$ *be a multi-dimensional classifier and let* $\mathbf{G} \subseteq \mathbf{F}$*. Let parameters* $\mathbf{x}$ *be balanced with the parameter* x *and let* $\alpha \geq 1$ *be such that either* $\frac{x\cdot(1-x^o)}{(1-x)\cdot x^o} = \alpha$ *or* $\frac{(1-x)\cdot x^o}{x\cdot(1-x^o)} = \alpha$*. Then,*

$$\frac{1}{\alpha^k} \leq \frac{O(\mathbf{C} \mid \mathbf{G})(x)}{O^o(\mathbf{C} \mid \mathbf{G})} \leq \alpha^k$$

[1] In earlier research, we introduced the related concept of *sliced sensitivity function* [3] which specifies an output probability of a Bayesian network in n linearly related parameters.

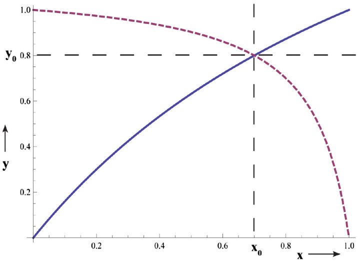

Fig. 1. Positively (solid line) and negatively (dashed line) balanced parameters x and y, with $x^o = 0.7$ and $y^o = 0.8$.

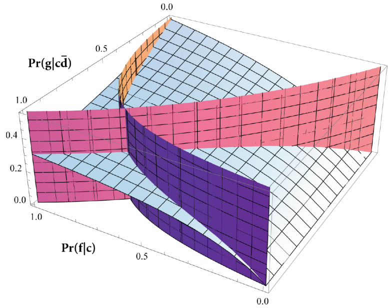

Fig. 2. A two-way sensitivity function in $x = \Pr(f|c)$, $y = \Pr(g \,|\, c\bar{d})$, and the surfaces defining the balanced sensitivity functions with $x^o = 0.7$ and $y^o = 0.8$.

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

Although the bounds stated above are not strict, they do give insight in the overall perturbation of the classifier's output distributions.

The idea of measuring the distance between two probability distributions by their odds ratio was introduced before by Chan and Darwiche [5]. More specifically, they proposed a measure which strictly bounds the odds ratio of an arbitrary probability of interest. Given changes in just a single probability table, their bounds are readily computed from just those changes; computing these bounds given multiple parameter changes however, is computationally expensive in general.

5 Tuning an Example Multi-dimensional Classifier

We consider the example classifier from Fig. 3 and its output probability of interest $\Pr(cd \mid fgh)$. With the original parameter values, we find that $\Pr(cd \mid fgh) = 0.29$. Now suppose that domain experts indicate that this probability should be 0.40, and that we would like to arrive at this value by adjusting the parameters $x = \Pr(f \,|\, c)$, $y = \Pr(g \,|\, c\bar{d})$ and $z = \Pr(h \,|\, g\bar{d})$. By Proposition 1, we find the sensitivity function:

$$\begin{aligned}\Pr(cd \,|\, fgh)(x,y,z) &= \frac{p_1^o \cdot x \cdot y^o \cdot z^o}{p_1^o \cdot x \cdot y^o \cdot z^o + p_2^o \cdot x \cdot y \cdot z + p_3^o \cdot x^o \cdot y^o \cdot z^o + p_4^o \cdot x^o \cdot y^o \cdot z} \\ &= \frac{0.94 \cdot x}{0.94 \cdot x + 3.47 \cdot x \cdot y \cdot z + 0.25 + 1.39 \cdot z}\end{aligned}$$

where $p_1^o = \Pr^o(cd \mid fgh)$, $p_2^o = \Pr^o(c\bar{d} \mid fgh)$, $p_3^o = \Pr^o(\bar{c}d \mid fgh)$ and $p_4^o = \Pr^o(\bar{c}\bar{d} \,|\, fgh)$. From this higher-order function, we now derive a balanced

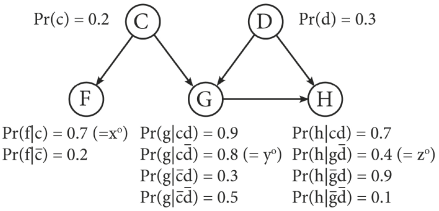

Fig. 3. An example multi-dimensional classifier, with $\Pr(cd \,|\, fgh)$ for its probability of interest to be tuned.

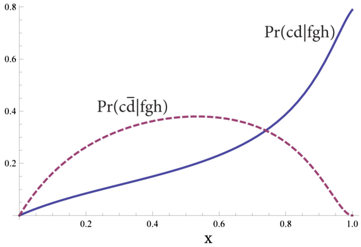

Fig. 4. Balanced functions for $\Pr(cd \mid fgh)$ and $\Pr(c\bar{d} \,|\, fgh)$, given a monotone balancing scheme for x, y, z with respect to $\Pr(cd \,|\, fgh)$.

sensitivity function $\Pr(cd| \, fgh)(x)$ by appropriately balancing the parameters y and z with x. Since $x \sim \Pr(cd \,|\, fgh)$, $y \nsim \Pr(cd \,|\, fgh)$ and $z \nsim \Pr(cd \,|\, fgh)$, we balance both y and z negatively with x, to guarantee that the output probability retains the same value range as with the corresponding higher-order sensitivity function. We find the balanced function

$$\Pr(cd \mid fgh)(x) = \frac{0.15 \cdot x - 0.184 \cdot x^2 + 0.05 \cdot x^3}{0.26 + 0.23 \cdot x - 1.07 \cdot x^2 + 0.59 \cdot x^3}$$

which is depicted in Fig. 4. The expert-provided value 0.4 for $\Pr(cd \mid fgh)$ is attained at $x = 0.81$; the other parameters then take the values $y = 0.69$ and $z = 0.27$. The value α of the adjustment is 1.83. As we changed a single parameter from three CPTs, we find that $[1/\alpha^k, \alpha^k] = [0.16, 6.10]$. In addition to the monotonically balanced sensitivity function $\Pr(cd \,|\, fgh)(x)$, the figure also depicts the function $\Pr(c\bar{d} \,|\, fgh)(x)$ found with the same balancing scheme for the parameters x, y, z. Since this scheme is non-monotone for $\Pr(c\bar{d} \,|\, fgh)$, the resulting balanced function is no longer monotone.

To attain the desired output probability $\Pr(cd \mid fgh) = 0.40$, also another combination of parameters can be varied. Varying other parameter combinations will generally result in another α and hence in other bounds on the changes in all output probabilities. For example, the desired probability is also found with $\Pr(f \mid \bar{c}) = 0.11$, $\Pr(g \mid \bar{c}\bar{d}) = 0.34$ and $\Pr(h \mid cd) = 0.82$. For this parameter combination $\alpha = 1.97$ is found, from which the interval $[1/\alpha^k, \alpha^k] = [0.13, 7.60]$ is established. In uncertainty of the actual changes therefore, the first tuning option is preferred.

6 Conclusions

Motivated by the observation that available data sets often prove problematic for learning multi-dimensional classifiers, we presented an elegant method for tuning their parameter probabilities based on expert-provided information. We showed that the topological properties and dedicated use of an MDC induce higher-order sensitivity functions of restricted functional form which can be established efficiently. We further designed a scheme of balanced parameter adjustment, by which a higher-order sensitivity function is reduced to an insightful single-parameter function which is readily exploited as a suitable heuristic for tuning. The heuristic was shown to incur changes within guaranteed bounds in all output probabilities over the class variables. Although not strict, these bounds do give insight in the changes in the classifier's output distributions which are incurred by balanced adjustment of different sets of parameters. In our future research, we would like to study these bounds with the aim of further tightening them. We also plan to study optimality properties of balancing parameter probabilities by their odds ratios in view of the odds-ratio based measure on the output.

The tuning method developed in this paper does not as yet provide for selecting parameters for tuning. Parameter selection may be based upon various considerations. An example criterion may be to select parameters which give the smallest changes in the output distribution as a whole, as was already suggested in our example. Yet, parameters may also be selected based on the sizes of the samples from which they were estimated originally. We plan to investigate the effects of these and other criteria in various real-world applications of multi-dimensional network classifiers.

Acknowledgements. This work was supported by the Netherlands Organisation for Scientific Research.

Appendix

Proof of Proposition 1. Let $\mathit{MDC}(\mathbf{C}, \mathbf{F})$ be a multi-dimensional classifier as before. Writing the output probability $\Pr(\mathbf{c} \mid \mathbf{f})$ for a given $\mathbf{c}$ and $\mathbf{f}$ as $\Pr(\mathbf{c} \mid \mathbf{f}) = (\Pr(\mathbf{f} \,|\, \mathbf{c}) \cdot \Pr(\mathbf{c})) \,/\, (\sum_{\mathbf{C}} \Pr(\mathbf{f} \,|\, \mathbf{C}) \cdot \Pr(\mathbf{C}))$, and including terms involving the original probability values $\Pr^o(\mathbf{c} \mid \mathbf{f})$ and $\Pr^o(\mathbf{c})$, results in

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

Rearranging its summands into a single fraction gives for the denominator

$$\frac{\sum\limits_{\mathbf{c}^* \in \mathbf{C}} \left(\Pr^o(\mathbf{c}^* \mid \mathbf{f}) \cdot \Pr(\mathbf{f} \mid \mathbf{c}^*) \cdot \Pr(\mathbf{c}^*) \cdot \prod\limits_{\mathbf{C} \setminus \mathbf{c}^*} \Pr^o(\mathbf{f} \mid \mathbf{C}) \cdot \Pr^o(\mathbf{C}) \right)}{\prod\limits_{\mathbf{C}} \Pr^o(\mathbf{f} \mid \mathbf{C}) \cdot \Pr^o(\mathbf{C})}$$

where $\mathbf{C}\backslash\mathbf{c}^*$ is used to denote the set of all joint assignments to $\mathbf{C}$ except $\mathbf{c}^*$. Substitution and simplification now gives

$$\begin{aligned}\Pr(\mathbf{c}|\,\mathbf{f}) \;&=\; \frac{\Pr^o(\mathbf{c}\mid\mathbf{f})\cdot\Pr(\mathbf{f}\mid\mathbf{c})\cdot\Pr(\mathbf{c})\cdot\prod_{\mathbf{C}\backslash\mathbf{c}}\Pr^o(\mathbf{f}\mid\mathbf{C})\cdot\Pr^o(\mathbf{C})}{\sum_{\mathbf{c}^*\in\mathbf{C}}\Pr^o(\mathbf{c}^*\mid\mathbf{f})\cdot\Pr(\mathbf{f}\mid\mathbf{c}^*)\cdot\Pr(\mathbf{c}^*)\cdot\prod_{\mathbf{C}\backslash\mathbf{c}^*}\Pr^o(\mathbf{f}\mid\mathbf{C})\cdot\Pr^o(\mathbf{C})}\\ &= \frac{\Pr^o(\mathbf{c}\mid\mathbf{f})\cdot\prod_i\Pr(f_i\mid\mathbf{c},\mathbf{f}_{F_i})\cdot\Pr(\mathbf{c})\cdot\prod_{\mathbf{C}\backslash\mathbf{c}}\prod_i\Pr^o(f_i\mid\mathbf{C})\cdot\Pr^o(\mathbf{C})}{\sum_{\mathbf{c}^*\in\mathbf{C}}\left(\Pr^o(\mathbf{c}^*\mid\mathbf{f})\cdot\prod_i\Pr(f_i\mid\mathbf{c}^*,\mathbf{f}_{F_i})\cdot\Pr(\mathbf{c}^*)\cdot\prod_{\mathbf{C}\backslash\mathbf{c}^*}\prod_i\Pr^o(f_i\mid\mathbf{C},\mathbf{f}_{F_i})\cdot\Pr^o(\mathbf{C})\right)}\end{aligned}$$

in which we used that $\Pr(\mathbf{f}\mid\mathbf{c}) = \prod_i \Pr(f_i\mid\mathbf{c},\mathbf{f}_{F_i})$ with $f_i,\mathbf{f}_{F_i}\sim\mathbf{f}$, and that $\Pr(\mathbf{c}) = \prod_j \Pr(c_j)$ with $c_j\sim\mathbf{c}$. We then find that

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

Proof of Proposition 2. For the one-way sensitivity function describing the output probability $\Pr(\mathbf{c}\mid\mathbf{f})$ of an MDC in a parameter $x\sim\mathbf{c}$, we have that $\Pr(\mathbf{c}\mid\mathbf{f})(x) = (x\cdot r)/(x\cdot s+t)$, where $r,s,t\geq 0$ since these constants arise from multiplication and addition of probabilities. The function's first derivative equals $\Pr(\mathbf{c}\mid\mathbf{f})'(x) = (r\cdot t)/(s\cdot x+t)^2$, which is always positive. Irrespective of the values of the other parameters in the classifier therefore, an increase in value of $x\sim\mathbf{c}$ will result in an increase of $\Pr(\mathbf{c}\mid\mathbf{f})$. Similarly, the output probability increases with a decrease in value of $x\nsim\mathbf{c}$. □

Proof of Proposition 3. Let $MDC(\mathbf{C},\mathbf{F})$, $\mathbf{G}$ and $\mathbf{x}$ be as stated in the proposition, and let $\mathbf{H}$ be such that $\mathbf{H}=\mathbf{F}\backslash\mathbf{G}$. We first show that the proposition holds for any value combination $\mathbf{c}\in\mathbf{C}$ given a fixed instance $\mathbf{f}$. Using Proposition 1 we find that

$$O(\mathbf{c}\mid\mathbf{f})(\mathbf{x}) = \frac{\Pr(\mathbf{c}\mid\mathbf{f})(\mathbf{x})}{1-\Pr(\mathbf{c}\mid\mathbf{f})(\mathbf{x})} = \frac{\Pr^o(\mathbf{c}\mid\mathbf{f})\cdot\prod_{x_i\sim\mathbf{c},x_j\nsim\mathbf{c}} x_i\cdot x_j^o}{\sum_{\mathbf{c}^*\in\mathbf{C}\backslash\mathbf{c}}\left(\Pr^o(\mathbf{c}^*\mid\mathbf{f})\cdot\prod_{x_i\sim\mathbf{c}^*,x_j\nsim\mathbf{c}^*} x_i\cdot x_j^o\right)}$$

from which we find

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

and hence

$$\min\nolimits_{\mathbf{c}^*\in\mathbf{C}\backslash\mathbf{c}}\frac{\prod\limits_{x_i\sim\mathbf{c},x_j\nsim\mathbf{c}} x_i\cdot x_j^o}{\prod\limits_{x_i\sim\mathbf{c}^*,x_j\nsim\mathbf{c}^*} x_i\cdot x_j^o} \;\leq\; \frac{O(\mathbf{c}\mid\mathbf{f})(\mathbf{x})}{O^o(\mathbf{c}\mid\mathbf{f})} \;\leq\; \max\nolimits_{\mathbf{c}^*\in\mathbf{C}\backslash\mathbf{c}}\frac{\prod\limits_{x_i\sim\mathbf{c},x_j\nsim\mathbf{c}} x_i\cdot x_j^o}{\prod\limits_{x_i\sim\mathbf{c}^*,x_j\nsim\mathbf{c}^*} x_i\cdot x_j^o}$$

If $\mathbf{x}$ includes all parameters of the classifier, from each probability table exactely two parameters will not cancel out from the fraction $(\prod_{x_i \sim \mathbf{c}, x_j \not\sim \mathbf{c}} x_i \cdot x_j^o) \,/\, (\prod_{x_i \sim \mathbf{c}^*, x_j \not\sim \mathbf{c}^*} x_i \cdot x_j^o)$. For each such parameter x, the fraction includes either $\frac{x}{x^o}$ or $\frac{x^o}{x}$. Now, for $\alpha \geq 1$, we have that $\frac{x}{x^o}, \frac{x^o}{x} \in [1/\alpha, \alpha]$. With a balanced sensitivity function therefore, the minimum of the fraction equals $1/\alpha^k$ and the maximum is α^k, where k is two times the number of probability tables. If $\mathbf{x}$ includes just a subset of the classifier's parameters, we find that $k = s + 2 \cdot t$, where s is the number of probability tables from which just a single parameter is in $\mathbf{x}$ and t is the number of tables with two or more parameters in $\mathbf{x}$.

For an instance $\mathbf{f}' \not\sim \mathbf{f}$, we find $\Pr(\mathbf{c} \,|\, \mathbf{f}')$ by replacing (some of) the parameters in the fraction above by their proportional co-variant, which gives $\frac{1-x}{1-x^o}$ or its reciprocal. Since for $\alpha \geq 1$, these fractions are in $[1/\alpha, \alpha]$ as well, the proof above generalises to all instances in $\mathbf{F}$. For a partial instance $\mathbf{g}$ we have that $\Pr(\mathbf{C} \mid \mathbf{g}) = \sum_{\mathbf{H}} \Pr(\mathbf{C} \mid \mathbf{g}, \mathbf{H}) \cdot \Pr(\mathbf{H} \mid \mathbf{g})$. Since $(O(\mathbf{C} \mid \mathbf{gH}))/(O^o(\mathbf{C} \mid \mathbf{gH})) \in [1/\alpha^k, \alpha^k]$ and $\sum_{\mathbf{H}} \Pr(\mathbf{H} \mid \mathbf{g}) = 1$, we further find that $(O(\mathbf{C} \mid \mathbf{g}))/(O^o(\mathbf{C} \mid \mathbf{g})) \in [1/\alpha^k, \alpha^k]$ for all $\mathbf{g} \in \mathbf{G}$. □

References

1. Bielza, C., Li, G., Larrañaga, P.: Multi-dimensional classification with Bayesian networks. Int. J. Approximate Reasoning **52**, 705–727 (2011)
2. De Bock, J., de Campos, C.P., Antonucci, A.: Global sensitivity analysis for MAP inference in graphical models. In: Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N.D., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems, vol. 27, pp. 2690–2698 (2014)
3. Bolt, J.H., Renooij, S.: Local sensitivity of Bayesian networks to multiple parameter shifts. In: van der Gaag, L.C., Feelders, A.J. (eds.) PGM 2014. LNCS, vol. 8754, pp. 65–80. Springer, Switzerland (2014)
4. Borchani, H., Bielza, C., Toro, C., Larrañaga, P.: Predicting human immunodeficiency virus inhibitors using multi-dimensional Bayesian network classifiers. Artif. Intell. Med. **57**, 219–229 (2013)
5. Chan, H., Darwiche, A.: A distance measure for bounding probabilistic belief change. Int. J. Approximate Reasoning **38**, 149–174 (2005)
6. van der Gaag, L.C., de Waal, P.R.: Multi-dimensional Bayesian network classifiers. In: Vomlel, J., Studený, M. (eds.) Proceedings of the Third European Workshop in Probabilistic Graphical Models, pp. 107–114 (2006)
7. Pearl, J.: Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann Publishers, Palo Alto (1988)
8. Renooij, S.: Co-variation for sensitivity analysis in Bayesian networks: properties, consequences and alternatives. Int. J. Approximate Reasoning **55**, 1022–1042 (2014)
9. de Waal, P.R., van der Gaag, L.C.: Inference and learning in multi-dimensional Bayesian network classifiers. In: Mellouli, K. (ed.) ECSQARU 2007. LNCS (LNAI), vol. 4724, pp. 501–511. Springer, Heidelberg (2007)