Check for updates

A New Score for Adaptive Tests in Bayesian and Credal Networks

Alessandro Antonucci(✉), Francesca Mangili, Claudio Bonesana, and Giorgia Adorni

Istituto Dalle Molle di Studi sull'Intelligenza Artificiale, Lugano, Switzerland
{alessandro,francesca,claudio.bonesana,giorgia.adorni}@idsia.ch

Abstract. A test is *adaptive* when the sequence and number of questions is dynamically tuned on the basis of the estimated skills of the taker. Graphical models, such as Bayesian networks, are used for adaptive tests as they allow to model the uncertainty about the questions and the skills in an explainable fashion, especially when coping with multiple skills. A better elicitation of the uncertainty in the question/skills relations can be achieved by interval probabilities. This turns the model into a *credal* network, thus increasing the inferential complexity of the queries required to select questions. This is especially the case for the information-theoretic quantities used as *scores* to drive the adaptive mechanism. We present an alternative family of scores, based on the mode of the posterior probabilities, and hence easier to explain. This makes considerably simpler the evaluation in the credal case, without significantly affecting the quality of the adaptive process. Numerical tests on synthetic and real-world data are used to support this claim.

Keywords: Adaptive tests · Information theory · Credal networks · Bayesian networks · Index of qualitative variation

1 Introduction

A test or an exam can be naturally intended as a measurement process, with the questions acting as sensors measuring the skills of the test taker in a particular discipline. Such measurement is typically imperfect with the skills modeled as latent variables whose actual values cannot be revealed in a perfectly reliable way. The role of the questions, whose answers are regarded instead as manifest variables, is to reduce the uncertainty about the latent skills. Following this perspective, probabilistic models are an obvious framework to describe tests. Consider for instance the example in Fig. 1, where a Bayesian network evaluates the probability that the test taker knows how to multiply integers. In such framework making the test *adaptive*, i.e., picking a next question on the basis of the

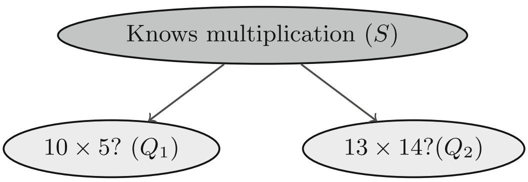
G. Adorni—We thank the Swiss National Science Foundation (grant no. 187246) for support

J. Vejnarová and N. Wilson (Eds.): ECSQARU 2021, LNAI 12897, pp. 399–412, 2021.
https://doi.org/10.1007/978-3-030-86772-0_29

current knowledge level of the test taker is also very natural. The information gain for the available questions might be used to select the question leading to the more informative results (e.g., according to Table 1, Q_1 is more informative than Q_2 no matter what the answer is). This might also be done before the answer on the basis of expectations over the possible alternatives.

A critical point when coping with such approaches is to provide a realistic assessment for the probabilistic parameters associated with the modeling of the relations between the questions and the skills. Having to provide sharp numerical values for these probabilities might be difficult. As the skill is a latent quantity, complete data are not available for a statistical learning and a direct elicitation should be provided by experts (e.g., a teacher). Yet, it might be not obvious to express such a domain knowledge by single numbers and a more robust elicitation, such as a probability interval (e.g., $P(Q_1 = 1|S_1 = 1) \in [0.85, 0.95]$), might add realism and robustness to the modeling process [15]. With such generalized assessments of the parameters a Bayesian network simply becomes a *credal* network [22]. The counterpart of such increased realism is the higher computational complexity of inference in credal networks [21]. This is an issue especially when coping with information-theoretic measures such as the information gain, whose computation in credal networks might lead to complex non-linear optimization tasks [19].

The goal of this paper is to investigate the potential of alternatives to the information-theoretic scores driving the question selection in adaptive tests based on directed graphical models, no matter whether these are Bayesian or credal networks. In particular, we consider a family of scores based on the (expected) mode of the posterior distributions over the skills. We show that, when coping with credal networks, the computation of these scores can be reduced to a sequence of linear programming task. Moreover, we show that these scores benefit of better explainability properties, thus allowing for a more transparent process in the question selection.



$$P(Q_1 = 1|S = 1) = 0.9$$
$$P(Q_1 = 1|S = 0) = 0.3$$
$$P(Q_2 = 1|S = 1) = 0.6$$
$$P(Q_2 = 1|S = 0) = 0.4$$

Fig. 1. A Bayesian network over Boolean variables modeling a simple test to evaluate integer multiplication skill. Probabilities of correct answers are also depicted.

The paper is organized as follows. A critical discussion about the existing work in this area is in Sect. 2. The necessary background material is reviewed in Sect. 3. The adaptive testing concepts are introduced in Sect. 4 and specialized to graphical models in 5. The technical part of the paper is in Sect. 6, where the new scores are discussed and specialized to the credal case, while the experiments are in Sect. 7. Conclusions and outlooks are in Sect. 8.

Table 1. Posterior probabilities of the skill after one or two questions in the test based on the Bayesian network in Fig. 1. A uniform prior over the skill is considered. Probabilities are regarded as grades and sorted from the lowest one. Bounds obtained with a perturbation $\epsilon = \pm 0.05$ of all the input parameters are also reported.

Q_1	Q_2	$P(S=1\|q_1,q_2)$	$\underline{P}(S=1\|q_1,q_2)$	$\overline{P}(S=1\|q_1,q_2)$
0	0	0.087	0.028	0.187
0	–	0.125	0.052	0.220
0	1	0.176	0.092	0.256
–	0	0.400	0.306	0.506
–	1	0.600	0.599	0.603
1	0	0.667	0.626	0.708
1	–	0.750	0.748	0.757
1	1	0.818	0.784	0.852

2 Related Work

Tests are modeled as a process relating latent and manifest variables since the classical *item response theory* (IRT), that has been widely used even to implement adaptive sequences [14]. Despite its success related to the ease of implementation and inference, IRT might be inadequate when coping with multiple latent skills, especially when these are dependent. This moved researchers towards the area of probabilistic graphical models [17], as practical tools to implement IRT in more complex setups [2]. Eventually, Bayesian networks have been identified as a suitable formalism to model tests, even behind the IRT framework [25], this being especially the case for adaptive models [26] and coached solving [12]. In order to cope with latent skills, some authors successfully adopted EM approaches to these models [23], this also involving the extreme situation of no ground truth information about the answers [6]. As an alternative approach to the same issue, some authors considered relaxations of the Bayesian formalism, such as fuzzy models [7] and imprecise probabilities [19]. The latter is the direction we consider here, but trying to overcome the computational limitations of that approach when coping with information-theoretic scores. This has some analogy with the approach in [11], that is focused on the Bayesian case only, but whose score, based on the *same-decision* problem, appears hard to be extended to the imprecise framework without increasing the computational complexity.

3 Background on Bayesian and Credal Networks

We denote variables by Latin uppercase letters, while using lowercase for their generic values, and calligraphic for the set of their possible values. Thus, $v \in \mathcal{V}$ is a possible value of V. Here we only consider discrete variables.[1]

[1] IRT uses instead continuous skills. Yet, with probabilistic models, discrete skills do not prevent evaluations to range over continuous domains. E.g., see Table 1, where the grade corresponds to a (continuous) probability.

3.1 Bayesian Networks

A probability mass function (PMF) over V is denoted as $P(V)$, while $P(v)$ is the probability assigned to state v. Given a function f of V, its expectation with respect to $P(V)$ is $\mathbb{E}_P(f) := \sum_{v\in\mathcal{V}} P(v)f(v)$. The expectation of $-\log_b[P(V)]$ is called *entropy* and denoted also as $H(V)$. In particular we assume $b := |\mathcal{V}|$ to have the maximum of the entropy, achieved for uniform PMFs, equal to one.

Given a joint PMF $P(U,V)$, the marginal PMF $P(V)$ is obtained by summing out the other variable, i.e., $P(v) = \sum_{u\in\mathcal{U}} P(u,v)$. Conditional PMFs such as $P(U|v)$ are similarly obtained by Bayes's rule, i.e., $P(u|v) = P(u,v)/P(v)$ provided that $P(v) > 0$. The notation $P(U|V) := \{P(U|v)\}_{v\in\mathcal{V}}$ is used for such a conditional probability table (CPT). The entropy of a conditional PMF is defined as in the unconditional case and denoted as $H(U|v)$. The conditional entropy is a weighted average of entropies of the conditional PMFs, i.e., $H(U|V) := \sum_{v\in\mathcal{V}} H(U|v)P(v)$. If $P(u,v) = P(u)P(v)$ for each $u \in \mathcal{U}$ and $v \in \mathcal{V}$, variables U and V are independent. Conditional formulations are also considered.

We assume the set of variables $\boldsymbol{V} := (V_1,\ldots,V_r)$ to be in one-to-one correspondence with a directed acyclic graph $\mathcal{G}$. For each $V \in \boldsymbol{V}$, the parents of V, i.e., the predecessors of V in $\mathcal{G}$, are denoted as Pa_V. The graph $\mathcal{G}$ together with the collection of CPTs $\{P(V|\mathrm{Pa}_V)\}_{V\in\boldsymbol{V}}$ provides a Bayesian network (BN) specification [17]. Under the Markov condition, i.e., every variable is conditionally independent of its non-descendants non-parents given its parents, a BN compactly defines a joint PMF $P(\boldsymbol{V})$ that factorizes as $P(\boldsymbol{v}) = \prod_{V\in\boldsymbol{V}} P(v|\mathrm{pa}_V)$.

Inference, intended as the computation of the posterior PMF of a single (queried) variable given some evidence about other variables, is in general NP-hard, but exact and approximate schemes are available (see [17] for details).

3.2 Credal Sets and Credal Networks

A set of PMFs over V is denoted as $K(V)$ and called *credal set* (CS). Expectations based on CSs are the bounds of the PMF expectations with respect to the CS. Thus $\underline{\mathbb{E}}[f] := \inf_{P(V)\in K(V)} \mathbb{E}[f]$ and similarly for the supremum $\overline{\mathbb{E}}$. Expectations of events are in particular called lower and upper probabilities and denoted as $\underline{P}$ and $\overline{P}$. Notation $K(U|v)$ is used for a set of conditional CSs, while $K(U|V) := \{K(U|v)\}_{v\in\mathcal{V}}$ is a credal CPT (CCPT).

Analogously to a BN, a credal network (CN) is specified by graph $\mathcal{G}$ together with a family of CCPTs $\{K(V|\mathrm{Pa}_V)\}_{V\in\boldsymbol{V}}$ [13]. A CN defines a joint CS $K(\boldsymbol{V})$ corresponding to all the joint PMFs induced by BNs whose CPTs are consistent with the CN CCPTs.

For CNs, we intend inference as the computation of the lower and upper posterior probabilities. The task generalizes BN inference being therefore NP-hard, see [21] for a deeper characterization. Yet, exact and approximate schemes are also available to practically compute inferences [4,5,16].

4 Testing Algorithms

A typical test aims at evaluating the knowledge level of a test taker σ on the basis of her answers to a number of questions. Let $\boldsymbol{Q}$ denote a repository of questions available to the instructor. The order and the number of questions picked from $\boldsymbol{Q}$ to be asked to σ might not be defined in advance. We call *testing algorithm* (TA) a procedure taking care of the selection of the sequence of questions asked to the test taker, and deciding when the test stops. Algorithm 1 depicts a general TA scheme, with $\boldsymbol{e}$ denoting the array of the answers collected from taker σ.

Algorithm 1. General TA: given the profile σ and repository $\boldsymbol{Q}$, an evaluation based on answers $\boldsymbol{e}$ is returned.

$\boldsymbol{e} \leftarrow \emptyset$
while not $\texttt{Stopping}(\boldsymbol{e})$ **do**
  $Q^* \leftarrow \texttt{Pick}(\boldsymbol{Q}, \boldsymbol{e})$
  $q^* \leftarrow \texttt{Answer}(Q^*, \sigma)$
  $\boldsymbol{e} \leftarrow \boldsymbol{e} \cup \{Q^* = q^*\}$
  $\boldsymbol{Q} \leftarrow \boldsymbol{Q} \setminus \{Q^*\}$
end while
return $\texttt{Evaluate}(\boldsymbol{e})$

The Boolean function `Stopping` decides whether the test should end, this choice being possibly based on the previous answers in $\boldsymbol{e}$. Trivial stopping rules might be based on the number of questions asked to the test taker ($\texttt{Stopping}(\boldsymbol{e}) = 1$ if and only if $|\boldsymbol{e}| > n$) or on the number of correct answers provided that a maximum number of questions is not exceeded. Function `Pick` selects instead the question to be asked to the student from the repository $\boldsymbol{Q}$. A TA is called *adaptive* when this function takes into account the previous answers $\boldsymbol{e}$. Trivial non-adaptive strategies might consist in randomly picking an element of $\boldsymbol{Q}$ or following a fixed order. The function `Answer` is simply collecting (or simulating) the answer of test taker σ to a particular question Q. In our assumptions, this answer is independent of the previous answers to other questions.[2]

Finally, `Evaluate` is a function returning the overall judgment of the test (e.g., a numerical grade or a pass/fail Boolean) on the basis of all the answers collected after the test termination. Trivial examples of such functions are the percentage of correct answers or a Boolean that is true when a sufficient number of correct answers has been provided. Note also that in our assumptions the TA is *exchangeable*, i.e., the stopping rule, the question finder and the evaluation function are invariant with respect to permutations in $\boldsymbol{e}$ [24]. In other words,

[2] Generalized setups where the quality of the student answer is affected by the previous answers will be discussed at the end of the paper. This might include a *fatigue* model negatively affecting the quality of the answers when many questions have been already answered as well as the presence of *revealing* questions that might improve the quality of other answers [18].

the same next question, the same evaluation and the same stopping decision is produced for any two students who provided the same list of answers in two different orders.

A TA is supposed to achieve reliable evaluation of taker σ from the answers $\boldsymbol{e}$. As each answer is individually assumed to improve such quality, asking all the questions, no matter the order because of the exchangeability assumption, is an obvious choice. Yet, this might be impractical (e.g., because of time limitations) or just provide an unnecessary burden to the test taker. The goal of a good TA is therefore to trade off the evaluation accuracy and the number of questions.[3]

5 Adaptive Testing in Bayesian and Credal Networks

The general TA setup in Algorithm 1 can be easily specialized to BNs as follows. First, we identify the profile σ of the test taker with the actual states of a number of latent discrete variables, called *skills*. Let $\boldsymbol{S} = \{S_j\}_{j=1}^m$ denote these skill variables, and $\boldsymbol{s}_\sigma$ the actual values of the skills for the taker. Skills are typically ordinal variables, whose states correspond to increasing knowledge levels. Questions in $\boldsymbol{Q}$ are still described as manifest variables whose actual values are returned by the `Answer` function. This is achieved by a (possibly stochastic) function of the actual profile $\boldsymbol{s}_\sigma$. This reflects the taker perspective, while the teacher has clearly no access to $\boldsymbol{s}_\sigma$. As a remark, note that we might often coarsen the set of possible values $\mathcal{Q}$ for each $Q \in \boldsymbol{Q}$: for instance, a multiple choice question with three options might have a single right answer, the two other answers being indistinguishable from the evaluation point of view.[4]

A joint PMF over the skills $\boldsymbol{S}$ and the questions $\boldsymbol{Q}$ is supposed to be available. In particular we assume this to correspond to a BN whose graph has the questions as leaf nodes. Thus, for each $Q \in \boldsymbol{Q}$, $Pa_Q \subseteq \boldsymbol{S}$ and we call Pa_Q the *scope* of question Q. Note that this assumption about the graph is simply reflecting a statement about the conditional independence between (the answer to) a question and all the other skills and questions given scope of the question. This basically means that the answers to other questions are not directly affecting the answer to a particular question.[5]

As the available data are typically incomplete because of the latent nature of the skills, dedicated learning strategies, such as various forms of constrained EM should be considered to train a BN from data. We refer the reader to the various contributions of Plajner and Vomlel in this field (e.g., [23]) for a complete discussion of that approach. Here we assume the BN quantification available.

[3] In some generalized setups, other elements such as a *serendipity* in choice in order to avoid tedious sequences of questions might be also considered [8].

[4] The case of *abstention* to an answer and the consequent problem of modeling the incompleteness is a topic we do not consider here for the sake of conciseness. Yet, general approaches based on the ideas in [20] could be easily adopted.

[5] Moving to other setups would not be really critical because of the separation properties of observed nodes in Bayesian and credal networks, see for instance [3,9].

In such a BN framework, `Stopping`($\boldsymbol{e}$) might be naturally based on an evaluation of the posterior PMF $P(\boldsymbol{S}|\boldsymbol{e})$, this being also the case for `Evaluate`. Regarding the question selection, `Pick` might be similarly based on the (posterior) CPT $P(\boldsymbol{S}|Q,\boldsymbol{e})$, whose values for the different answers to Q might be weighted by the marginal $P(Q|\boldsymbol{e})$. More specifically, entropies and conditional entropies are considered by Algorithm 2, while the evaluation is based on a conditional expectation for a given utility function.

Algorithm 2. Information Theoretic TA in a BN over the questions $\boldsymbol{Q}$ and the skills $\boldsymbol{S}$: given the student profile $\boldsymbol{s}_\sigma$, the algorithm returns an evaluation corresponding to the expectation of an evaluation function f with respect to the posterior for the skills given the answers $\boldsymbol{e}$.

1: $\boldsymbol{e} = \emptyset$
2: **while** $H(\boldsymbol{S}|\boldsymbol{e}) > H^*$ **do**
3: $\quad Q^* \leftarrow \arg\max_{Q \in \boldsymbol{Q}} [H(\boldsymbol{S}|\boldsymbol{e}) - H(\boldsymbol{S}|Q,\boldsymbol{e})]$
4: $\quad q^* \leftarrow$ `Answer`($Q^*, \boldsymbol{s}_\sigma$)
5: $\quad \boldsymbol{e} \leftarrow \boldsymbol{e} \cup \{Q^* = q^*\}$
6: $\quad \boldsymbol{Q} \leftarrow \boldsymbol{Q} \setminus \{Q^*\}$
7: **end while**
8: **return** $\mathbb{E}_{P(\boldsymbol{S}|\boldsymbol{e})}[f(\boldsymbol{S})]$

When no data are available for the BN training, elicitation techniques should be considered instead. As already discussed, CNs might offer a better formalism to capture domain knowledge, especially by providing interval-valued probabilities instead of sharp values. If this is the case, a CN version of Algorithm 2 can be considered. To achieve that, in line 3, a score taking into account the fact that the entropies in a CN are not anymore precisely specified should be adopted. Similar considerations apply to the evaluation function in line 8.

The price of such increased realism in the elicitation is the higher complexity characterizing inferences based on CNs. The work in [19] offers a critical discussion of those issues, that are only partially addressed by heuristic techniques used there to approximate the upper bounds of conditional entropies. In the next section we consider an alternative approach to cope with CNs and adaptive TAs based on different scores used to select the questions.

6 A New Score for Testing Algorithms

Following [27], we can regard the PMF entropy (and its conditional version) used by Algorithm 2 as an example of an index of *qualitative variation* (IQV). An IQV is just a normalized number that takes value zero for degenerate PMFs, one on uniform ones, being independent on the number of possible states (and samples for empirical models). The closer to uniform is the PMF, the higher is the index and vice versa.

In order to bypass the computational issues related to its application with CNs, we want to consider alternative IQVs to replace entropy in Algorithm 2. Wilkox *deviation from the mode* (DM) appears a sensible option. Given a PMF $P(V)$, this corresponds to:

$$M(V) := 1 - \sum_{v \in \mathcal{V}} \frac{\max_{v' \in \mathcal{V}} P(v') - P(v)}{|\mathcal{V}| - 1} \,. \tag{1}$$

It is a trivial exercise to check that this is a proper IQV, with the same unimodal behavior of the entropy. In terms of explainability, being a linear function of the modal probability, the numerical value of the DM offers a more transparent interpretation than the entropy. From a computational point of view, for both marginal and conditional PMFs, both the entropy and the DM can be directly obtained from the probabilities of the singletons.

The situation is different when computing the bounds of these quantities with respect to a CS. For the upper bound, by simple algebra, we obtain:

$$\overline{M}(V) := \max_{P(V) \in K(V)} M(V) = \frac{1 - \min_{P(V) \in K(V)} \max_{v' \in \mathcal{V}} P(v')}{(|\mathcal{V}| - 1)/|\mathcal{V}|} \,. \tag{2}$$

As we assume CSs defined by a finite number of linear constraints, such a *minimax* objective function can be easily reduced to a linear programming task by adding an auxiliary variable corresponding to the maximum and the constraints modeling the fact that this variable is the maximum. The situation is even simpler for the lower bound $\underline{M}(V)$, which reduces to a *maximax* corresponding to the identification of the singleton state with the highest upper probability. Optimizing entropy requires instead a non-trivial, but convex, optimization. See for instance [1] for an iterative procedure to find the maximum when coping with CSs defined by probability intervals. The situation is even more critical for the minimization, that has been proved to be NP-hard in [28].

The optimization becomes more challenging for conditional entropies, as these are mixtures of entropies of conditional distributions based on imprecise weights. Consequently, in [19], only an inner approximation for the upper bound have been derived. The situation is different for conditional DMs. The following result offers a feasible approach in a simplified setup, to be later extended to the general case.

Theorem 1. *Under the setup of Sect. 5, consider a CN with a single skill S and a single question Q, that is a child of S. Let $K(S)$ and $K(Q|S)$ be the CCPTs of such CN. Let also $\mathcal{Q} = \{q^1, \ldots, q^n\}$ and $\mathcal{S} = \{s^1, \ldots, s^m\}$. The upper conditional DM, i.e.,*

$$\overline{M}(S|Q) := 1 - \min_{\substack{P(S) \in K(S) \\ P(Q|S) \in K(Q|S)}} \sum_{i=1}^{n} \left[\max_{j \in \{1,\ldots,m\}} P(s_j|q_i) \right] P(q_i) \,, \tag{3}$$

where the denominator in (2) was omitted for the sake of brevity, is such that:

$$\overline{M}(S|Q) := 1 - \min_{\hat{j}_1,\ldots,\hat{j}_n \in \{1,\ldots,m\}} \Omega(\hat{j}_1, \ldots, \hat{j}_n) \,, \tag{4}$$

where $\Omega(\hat{j}_1, \ldots, \hat{j}_n)$ is the solution of the following linear programming task.

$$\begin{aligned}
\min \quad & \sum_i x_{i\hat{j}_i} & & \\
\text{s.t.} \quad & \sum_{ij} x_{ij} = 1 & & (5)\\
& x_{ij} \geq 0 & \forall i,j & \quad (6)\\
& \sum_k x_{kj} \geq \underline{P}(s_j) & \forall j & \quad (7)\\
& \sum_k x_{kj} \leq \overline{P}(s_j) & \forall j & \quad (8)\\
& \underline{P}(q_i|s_j) \sum_k x_{kj} \leq x_{ij} & \forall i,j & \quad (9)\\
& \overline{P}(q_i|s_j) \sum_k x_{kj} \geq x_{ij} & \forall i,j & \quad (10)\\
& x_{i\hat{j}_i} \geq x_{ij} & \forall i, j \neq \hat{j}_i & \quad (11)
\end{aligned}$$

Assignments of $(\hat{j}_1, \ldots, \hat{j}_n)$ such that the corresponding linear programming task is unfeasible are just removed from the minimization in Eq. (4). Note that the bounds on the sums over the indexes and on the universal quantifiers are also omitted for the sake of brevity.

Proof. *Equation (3) rewrites as:*

$$\overline{M}(S|Q) = 1 - \min_{\substack{P(S) \in K(S) \\ P(Q|S) \in K(Q|S)}} \sum_{i=1}^{n} \left[\max_{j=1,\ldots,m} P(s_j) P(q_i|s_j) \right] . \tag{12}$$

We define the variables of such constrained optimization as $x_{ij} := P(s_j) \cdot P(q_i|s_j)$ for each $i \in \{1, \ldots, n\}$ and $j \in \{1, \ldots, m\}$. The CCPT constraints can be easily reformulated with respect to such new variables by noticing that $x_{ij} = P(s_j, q_i)$, and hence $P(s_j) = \sum_i x_{ij}$ and $P(q_i|s_j) = x_{ij}/(\sum_k x_{kj})$. Consequently, the interval constraints on $P(S)$ correspond to the linear constraints in Eqs. (7) and (8). Similarly, for $P(Q|S)$, we obtain:

$$\underline{P}(q_i|s_j) \leq \frac{x_{ij}}{\sum_k x_{kj}} \leq \overline{P}(q_i|s_j) , \tag{13}$$

that easily gives the linear constraints in Eqs. (9) and (10) (as we cope with strictly positive probabilities of the skills, the denominator in Eq. (13) cannot be zero). The non-negativity of the probabilities corresponds to Eq. (6), while Eq. (5) gives the normalization of $P(S,Q)$ and the normalization of $P(Q|S)$ is by construction. Equation (12) rewrites therefore as:

$$\overline{M}(S|Q) = 1 - \min_{\{x_{ij}\} \in \Gamma} \sum_i \max_j x_{ij} , \tag{14}$$

where Γ denotes the linear constraints in Eqs. (5)–(10)*. If* $\hat{j}_i := \arg\max_j x_{ij}$*, Eq.* (14) *rewrites as:*

$$\overline{M}(S|Q) = 1 - \min_{\{x_{ij}\} \in \Gamma'} \sum_i x_{i\hat{j}_i} \,, \tag{15}$$

where Γ' are the constraints in Γ with the additional (linear) constraints in Eq. (11) *implementing the definition of $\hat{j}$. The minimization in the right-hand side of Eq.* (15) *is not a linear programming task, as the values of the indexes $\hat{j}_i$ are potentially different for different assignments of the optimization variables consistent with the constraints in Γ. Yet, we might address such optimization as a brute-force task with respect to all the possible assignation of the indexes $\hat{j}_i$. This is exactly what is done by Eq.* (4) *where all the m^n possible assignations are considered. Finally, regarding the feasibility of the constraints, as the constraints in Eq.* (14) *are feasible by construction, there is at least a value of $(\hat{j}_1, \ldots, \hat{j}_n)$, i.e., the one corresponding to the optimum, for which also the constraints in Eq.* (15) *are feasible.* □

An analogous result holds for the computation of $\underline{M}(S|Q)$. In that case a maximum should replace the minimum in both Eq. (3) and in the linear programming tasks. The overall complexity is $O(m^n)$ with $n := |\mathcal{Q}|$. This means quadratic complexity for any test where only the difference between a wrong and a right answer is considered from an elicitation perspective, and tractable computations provided that the number of distinct answers to the same question is bounded by a small constant. Coping with multiple questions becomes trivial by means of the results in [3], that allows to merge multiple observed children into a single one. Finally, the case of multiple skills might be similarly considered by using the marginal bounds of the single skills in Eqs. (7) and (8).

7 Experiments

We validate the ideas outlined in the previous section in order to check whether or not the DM can be used for TAs as a sensible alternative to information-theoretic scores such as the entropy. In the BN context, this is achieved by computing the necessary posterior probabilities, while Theorem 1 is used instead for CNs.

7.1 Single-Skill Experiments on Synthetic Data

For a very first validation of our approach, we consider a simple setup made of a single Boolean skill S and a repository with 18 Boolean questions based on nine different parametrizations (two questions for each parametrization). In such a BN, the CPT of a question can be parametrized by two numbers. E.g., in the example in Fig. 1, we used the probabilities of correctly answering the question given that the skill is present or not, i.e., $P(Q = 1|S = 1)$ and $P(Q = 1|S = 0)$.

A more interpretable parametrization can be obtained as follows:

$$\delta := 1 - \frac{1}{2}[P(Q=1|\,S=1) + P(Q=1|\,S=0)]\,, \tag{16}$$

$$\kappa := P(Q=1|\,S=1) - P(Q=1|\,S=0)\,. \tag{17}$$

Note that $P(Q=1|S=1) > P(Q=1|S=0)$ is an obvious rationality constraint for questions, otherwise having the skill would make it less likely to answer properly to a question. Both parameters are therefore non-negative. The parameter δ, corresponding to the average probability of a wrong answer over the different skill values, can be regarded as a normalized index of the question *difficulty*. E.g., in Fig. 1, Q_1 ($\delta = 0.4$) is less difficult than Q_2 ($\delta = 0.5$). The parameter κ can be instead regarded as a descriptor of the difference of the conditional PMFs associated with the different skill values. In the most extreme case $\kappa = 1$, the CPT $P(Q|S)$ is diagonal implementing an identity mapping between the skill and the question. We therefore regard κ as a indicator of the *discriminative* power of the question. In our tests, for the BN quantification, we consider the nine possible parametrizations corresponding to $(\delta, \gamma) \in [0.4, 0.5, 0.6]^2$. For $P(S)$ we use instead a uniform PMF. For the CN approach we perturb all the BN parameters with $\epsilon = \pm 0.05$, thus obtaining a CN quantification. A group of 1024 simulated students, half of them having $S=0$ and half with $S=1$ is used for simulations. The student answers are sampled from the CPT of the asked question on the basis of the student profile. Figure 2 (left) depicts the accuracy of the BN and CN approaches based on both the entropy and the DM scores. To force credal models to give a single output, decisions are based on the mid-point between the lower and the upper probability, while lower entropies are used. Pure credal approaches returning multiple options will be considered in a future work. We notably see all the adaptive approaches outperforming a non-adaptive, random, choice of the questions. To better investigate the strong overlap between these trajectories, in Fig. 2 (right) we compute the Brier score and we observe a strong similarity between DM and entropy approaches in both the Bayesian and the credal case, with the credal slightly outperforming the Bayesian approach.

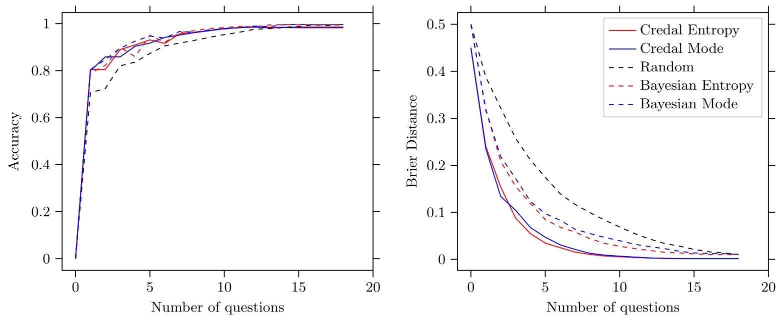


Fig. 2. Accuracy (left) and Brier distance (right) of TAs for a single-skill BN/CN

7.2 Multi-skill Experiments on Real Data

For a validation on real data, we consider an online German language placement test (see also [19]). Four different Boolean skills associated with different abilities (vocabulary, communication, listening and reading) are considered and modeled by a chain-shaped graph, for which BN and CN quantification are already available. A repository of 64 Boolean questions, 16 for each skill, with four different levels of difficulty and discriminative power, have been used. Experiments have been achieved by means of the CREMA library for credal networks [16].[6] The Java code used for the simulations is available together with the Python scripts used to analyze the results and the model specifications.[7] Performances are evaluated as for the previous model, the only difference being that here the accuracy is aggregated by average over the separate accuracies for the four skills. The results (Fig. 3) are analogous to those for the single-skill case: entropy-based and mode-based scores are providing similar results, with the credal approach typically leading to more accurate evaluations.

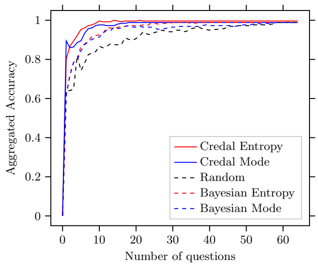


Fig. 3. Aggregated accuracy for a multi-skill TA

8 Outlooks and Conclusions

A new score for adaptive testing in Bayesian and credal networks has been proposed. Our proposal is based on indexes of qualitative variation, being in particular focused on the modal probability for their explainability features. An algorithm to evaluate this quantity in the credal case is derived. Our experiments show that moving to these scores does not really affect the quality of the selection process. Besides a deeper experimental validation, a necessary future work consists in the derivation of simpler elicitation strategies for these models in order to promote their application to real-world testing environments. To achieve that we also intend to embed these new scores in a software we recently developed for the practical implementation of web-based adaptive tests [10].[8]

---

6 https://github.com/IDSIA/crema.

7 https://github.com/IDSIA/crema-adaptive.

8 https://github.com/IDSIA/adapquest.

References

1. Abellan, J., Moral, S.: Maximum of entropy for credal sets. Int. J. Uncertainty Fuzziness Knowl.-Based Syst. **11**(05), 587–597 (2003)
2. Almond, R.G., Mislevy, R.J.: Graphical models and computerized adaptive testing. Appl. Psychol. Meas. **23**(3), 223–237 (1999)
3. Antonucci, A., Piatti, A.: Modeling unreliable observations in Bayesian networks by credal networks. In: Godo, L., Pugliese, A. (eds.) SUM 2009. LNCS (LNAI), vol. 5785, pp. 28–39. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-04388-8_4
4. Antonucci, A., de Campos, C.P., Huber, D., Zaffalon, M.: Approximating credal network inferences by linear programming. In: van der Gaag, L.C. (ed.) ECSQARU 2013. LNCS (LNAI), vol. 7958, pp. 13–24. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-39091-3_2
5. Antonucci, A., de Campos, C.P., Huber, D., Zaffalon, M.: Approximate credal network updating by linear programming with applications to decision making. Int. J. Approximate Reasoning **58**, 25–38 (2015)
6. Bachrach, Y., Graepel, T., Minka, T., Guiver, J.: How to grade a test without knowing the answers–a Bayesian graphical model for adaptive crowdsourcing and aptitude testing. arXiv preprint arXiv:1206.6386 (2012)
7. Badaracco, M., Martínez, L.: A fuzzy linguistic algorithm for adaptive test in intelligent tutoring system based on competences. Expert Syst. Appl. **40**(8), 3073–3086 (2013)
8. Badran, M.E.K., Abdo, J.B., Al Jurdi, W., Demerjian, J.: Adaptive serendipity for recommender systems: Let it find you. In: ICAART (2), pp. 739–745 (2019)
9. Bolt, J.H., De Bock, J., Renooij, S.: Exploiting Bayesian network sensitivity functions for inference in credal networks. In: Proceedings of the Twenty-Second European Conference on Artificial Intelligence (ECAI), vol. 285, pp. 646–654. IOS Press (2016)
10. Bonesana, C., Mangili, F., Antonucci, A.: ADAPQUEST: a software for web-based adaptive questionnaires based on Bayesian networks. In: IJCAI 2021 Workshop Artificial Intelligence for Education (2021)
11. Chen, S.J., Choi, A., Darwiche, A.: Computer adaptive testing using the same-decision probability. In: BMA@ UAI, pp. 34–43 (2015)
12. Conati, C., Gertner, A.S., VanLehn, K., Druzdzel, M.J.: On-line student modeling for coached problem solving using Bayesian networks. In: Jameson, A., Paris, C., Tasso, C. (eds.) User Modeling. ICMS, vol. 383, pp. 231–242. Springer, Vienna (1997). https://doi.org/10.1007/978-3-7091-2670-7_24
13. Cozman, F.G.: Credal networks. Artif. Intell. **120**(2), 199–233 (2000)
14. Embretson, S.E., Reise, S.P.: Item Response Theory. Psychology Press, Hove (2013)
15. Hájek, A., Smithson, M.: Rationality and indeterminate probabilities. Synthese **187**(1), 33–48 (2012)
16. Huber, D., Cabañas, R., Antonucci, A., Zaffalon, M.: CREMA: a Java library for credal network inference. In: Jaeger, M., Nielsen, T. (eds.) Proceedings of the 10th International Conference on Probabilistic Graphical Models (PGM 2020). Proceedings of Machine Learning Research, PMLR, Aalborg, Denmark (2020)
17. Koller, D., Friedman, N.: Probabilistic Graphical Models: Principles and Techniques. MIT Press, Cambridge (2009)
18. Laitusis, C.C., Morgan, D.L., Bridgeman, B., Zanna, J., Stone, E.: Examination of fatigue effects from extended-time accommodations on the SAT reasoning test. ETS Research Report Series **2007**(2), i–13 (2007)

19. Mangili, F., Bonesana, C., Antonucci, A.: Reliable knowledge-based adaptive tests by credal networks. In: Antonucci, A., Cholvy, L., Papini, O. (eds.) ECSQARU 2017. LNCS (LNAI), vol. 10369, pp. 282–291. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-61581-3_26
20. Marchetti, S., Antonucci, A.: Reliable uncertain evidence modeling in Bayesian networks by credal networks. In: Brawner, K.W., Rus, V. (eds.) Proceedings of the Thirty-First International Florida Artificial Intelligence Research Society Conference (FLAIRS-31), pp. 513–518. AAAI Press, Melbourne, Florida, USA (2018)
21. Mauá, D.D., De Campos, C.P., Benavoli, A., Antonucci, A.: Probabilistic inference in credal networks: new complexity results. J. Artif. Intell. Res. **50**, 603–637 (2014)
22. Piatti, A., Antonucci, A., Zaffalon, M.: Building knowledge-based expert systems by credal networks: a tutorial. In: Baswell, A. (ed.) Advances in Mathematics Research, vol. 11, chap. 2. Nova Science Publishers, New York (2010)
23. Plajner, M., Vomlel, J.: Monotonicity in practice of adaptive testing. arXiv preprint arXiv:2009.06981 (2020)
24. Sawatzky, R., Ratner, P.A., Kopec, J.A., Wu, A.D., Zumbo, B.D.: The accuracy of computerized adaptive testing in heterogeneous populations: a mixture item-response theory analysis. PLoS ONE **11**(3), e0150563 (2016)
25. Vomlel, J.: Bayesian networks in educational testing. Int. J. Uncertain. Fuzziness Knowl.-Based Syst. **12**(supp01), 83–100 (2004)
26. Vomlel, J.: Building adaptive tests using Bayesian networks. Kybernetika **40**(3), 333–348 (2004)
27. Wilcox, A.R.: Indices of qualitative variation and political measurement. Western Political Q. **26**(2), 325–343 (1973)
28. Xiang, G., Kosheleva, O., Klir, G.J.: Estimating information amount under interval uncertainty: algorithmic solvability and computational complexity. Technical report 158, Departmental Technical Reports (CS) (2006)