Tractable Classification with Non-Ignorable Missing Data Using Generative Random Forests

Julissa Villanueva[1] and Denis Mauá[2]

[1] Universidade de São Paulo, Brazil
jgville@ime.usp.br
[2] Universidade de São Paulo, Brazil
ddm@ime.usp.br

Abstract. Missing data is abundant in predictive tasks. Typical approaches assume that the missingness process is ignorable or non-informative and handle missing data either by marginalization or heuristically. Yet, data is often missing in a non-ignorable way, which introduces bias in prediction if not treated properly. In this paper, we develop a new method to perform tractable predictive inference under non-ignorable missing data using probabilistic circuits derived from Decision Tree Classifiers and a partially specified response model of missingness. We show empirically that our method delivers less biased (probabilistic) classifications than approaches that assume missing at random and are more determinate than similar existing overcautious approaches.

Categories and Subject Descriptors: I.2.6 [**Artificial Intelligence**]: Inference

Keywords: generative random forests, probabilistic circuits, non-ignorable missing data

1. INTRODUCTION

Probabilistic Circuits (PCs) are probabilistic graphical models that allow for linear time marginal inference [Choi et al. 2020; Liang and Van den Broeck 2019]. Sum-Product Networks [Poon and Domingos 2011], Probabilistic Sentential Decision Diagrams [Kisa et al. 2014] and Cutset Networks [Rahman et al. 2014] are all special cases of Probabilistic Circuits. These models have obtained notable results in several machine learning tasks due to their ability to compactly represent multidimensional distributions and efficiently produce reliable inference [Peharz et al. 2020; Poon and Domingos 2011; Shao et al. 2020; Shen et al. 2017; Shen et al. 2019; Zheng et al. 2018]. Generative Random Forests (GeFs) are yet another special class of Probabilistic Circuits [Correia et al. 2020]. GeFs extend standard Random Forests Classifiers and Regressors into full statistical models, and allow, among other queries, to efficiently compute exact predictive inference under missing data by marginalizing non-observed feature variables.

Handling missing data in classification in a principled and efficient data is an old and unsolved problem in machine learning. The standard approaches are (multiple) imputation, which produces one or more complete datasets, and marginalization of non-observed variables [Azur et al. 2011; Khosravi et al. 2019]. Those approaches (and many others) are justified under the assumption that non-observed data is *missing at random* (MAR), that states that the probability of the missing values only depends on the observed values for a certain prediction [Rubin 1976]. While MAR is popular, it is often violated in practical scenarios. For example, in personalized recommendation, users have a strong bias towards rating items which they either strongly like or strongly dislike [Marlin et al.

This work was partly supported by CAPES Finance 001, CNPq grant #304012/2019-0, and the joint FAPESP-IBM grant #2019/07665-4 (C4AI).

2011]. In such cases, called *non-ignorable missing data* or missing not at random (MNAR), imputing or averaging over completions can lead to biased and (overly confident) inconsistent estimates.

An arguably more principled approach to coping with MNAR data is to suspend judgment (i.e., avoid making a prediction) whenever different completions of missing values could result in different decisions (i.e., classification) according to the predictions of the model [Zaffalon 2002]. Following that principle, we developed in previous work [Villanueva et al. 2021] a linear-time algorithm to decide whether to suspend judgment in PCs derived from GeFs for classifications problems with non-ignorable missing data. Our approach thus avoids making unwarranted and untested hypothesis (such as MAR); however it produces too overcautious and often too non-informative predictions (e.g., the set of all classes).

In this work, we improve on our previous approach to MNAR data by assuming the availability of a partially specified response model in the form of probability intervals of the probability of observing/measuring a variable conditional on each possible value. Such intervals can often be obtained from expert domain knowledge, or derived from specially curated training data (e.g., a small sample survey with users of a recommendation system). We devise a polynomial-time procedure to perform exact inference *at prediction time* in the presence of non-ignorable missing data with Decision Tree Classifiers, assuming such a partially specified response model is available. The procedure can be used to determine the robustness of classifications of a target variable to the (full) specification of the response model. We compare our results with our previous more conservative approach described in [Villanueva et al. 2021], by comparing accuracy of the classifications made by marginalizing non-observed variables (i..e, the approach taken in [Correia et al. 2020]) according to the robustness of each instance as judged by either method. Our experiments show that our new algorithm obtains reliable conclusions often more accurate than criteria that ignores or marginalizes missing variables, while being more informative than our previous approach.

2. GENERATIVE RANDOM FORESTS AND CREDAL PROBABILISTIC CIRCUITS

We start by establishing some notation and terminology. We denote random variables (RVs) by upper-case letters (e.g., X_i, X), and their values by lower case (e.g., x_i, x). Sets of random variables are written in boldface (e.g., $\mathbf{X}$), as well as their instantiations (e.g., $\mathbf{x}$). In this work we assume that RVs take on a finite number of values, denoted as $val(X)$ for random variable X. We associate every discrete random variable X with a set of indicator functions $\{[\![X = x]\!] : x \in val(X)\}$, where the notation $[\![X = x]\!]$ describes the indicator function that returns 1 if X takes value x and 0 otherwise. We review Generative Random Forests and Credal Probabilistic Circuits, the backbones of our proposal.

2.1 Generative Random Forests and Probabilistic Circuits

A Probabilistic Circuit (PC) $\mathtt{M}$ over a set of categorical random variables $\mathbf{X}$ is a rooted weighted acyclic directed graph whose leaves are associated with indicator functions $[\![X_i = x_i]\!]$ of variables in $\mathbf{X}$, and the internal nodes are associated to either sum or product operations. The arcs $i \to j$ leaving sum node i are associated with non-negative weights w_{ij}. We write $\mathtt{M}_i$ to denote the sub-PC rooted at node i . The *scope* of a PC is the set of random variables associated with the indicator variables at the leaves, and the scope of a node is the scope of the respective sub-PC. A PC represents a joint distribution of $\mathbf{X}$ by $P_{\mathtt{M}}(\mathbf{x}) = \mathtt{M}(\mathbf{x})/(\sum_{\mathbf{x}'} \mathtt{M}(\mathbf{x}'))$. We assume here w.l.o.g. that PCs are normalized, which implies that $\sum_{\mathbf{x}'} \mathtt{M}(\mathbf{x}') = 1$. The value $\mathtt{M}(\mathbf{x})$, called the evaluation of the circuit at $\mathbf{x}$, is defined inductively in the size of the circuit as:

$$\mathtt{M}(\mathbf{x}) = \begin{cases} [\![X_i = x_i]\!](\mathbf{x}_i) & \text{if is a leaf node} [\![X_i = x_i]\!]\,, \\ \sum_j w_{ij} \mathtt{M}_j(\mathbf{x}) & \text{if s a circuit rooted at a sum node } i \text{ with children } j\,, \\ \prod_j \mathtt{M}_j(\mathbf{x}) & \text{if s a circuit rooted at a product node } i \text{ with children } j\,. \end{cases}$$

Smoothness, a.k.a. *completeness*, states that the scopes of any two children of a sum node are identical. *Decomposability* states that the scopes of any two children of a product node are disjoint. To ensure that marginal inference is computed in linear time in the size of the circuit, it suffices that the circuit satisfies smoothness and decomposability [Kisa et al. 2014; Poon and Domingos 2011]. For the rest of this paper, *we assume that PCs are smooth and decomposable. Determinism*, a.k.a. *selectivity*, states that each sum node has at most one child that evaluates to non-zero at any (complete) realization of its scope. It ensures that maximum likelihood estimates for the weights can be obtained in closed-form under complete data, and that a most probable realization is linear-time computable, a task otherwise NP-hard [Peharz et al. 2014].

A Generative Random Forest is a mixture of selective, smooth and decomposable PCs, each of them constructed from a Decision Tree as follows. Given a Decision Tree mapping features $\mathbf{X}$ to a target variable Y, convert each decision node into a sum node and each leaf into a sub-PC whose support is the partition induced by the corresponding path of the Decision Tree. The sub-PCs at the leaves can be learned with any structure learning algorithm for PCs (e.g., Learn-SPN) or take simple forms such as fully factorized distributions. Figure 1 illustrates this idea. We say that a PC $\mathtt{M}$ is *strong selective* with respect to a variable Y, where $\text{scope}(\mathtt{M}) = \{Y, \mathbf{X}\}$, if for each sum node $\mathtt{M}^i$, each pair of values y', y'' of Y and and each realization $\mathbf{x}$, at most one child of that node evaluates to a non-zero value. This is akin to (and implied by) the concept of class-factorized GeFs [Correia et al. 2020].

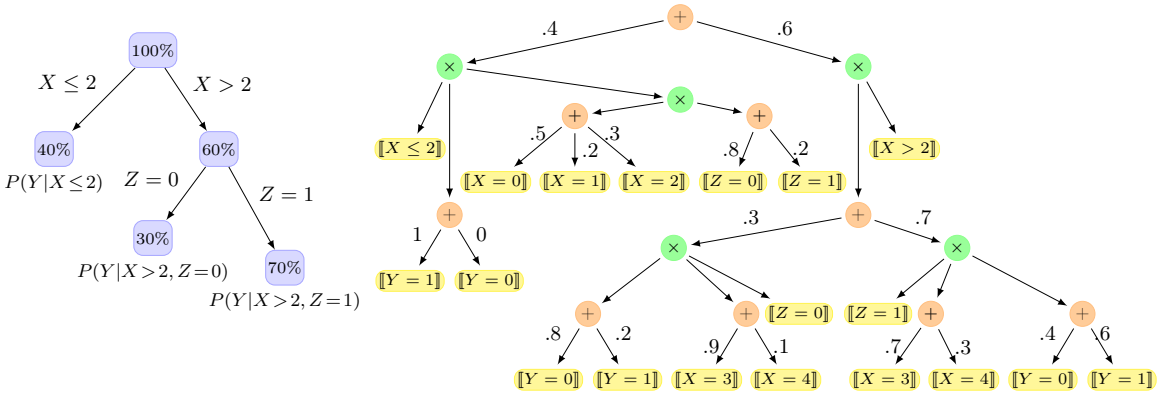

Fig. 1. A Decision Tree for classifying Y given X and Z (left) and its class-factorized Generative Decision Tree (right).

2.2 Credal Probabilistic Circuits

A *Credal Probabilistic Circuit* (CPC) is a set of complete, decomposable and normalized PCs which all share the same network structure [Mauá et al. 2018]. We denote a CPC as $\{\mathtt{M}_{\mathbf{w}} : \mathbf{w} \in \mathcal{C}\}$, where $\mathcal{C}$ is some space of weights (i.e., parametrizations). The set $\mathcal{C}$ is usually taken as the Cartesian product of closed and convex sets of weights $\mathcal{C}_i$, one for each sum node i in the network. Since the networks are normalized, a CPC induces a credal set (CS) (i.e., a set of probability distributions) [Levi 1980]. An example of a CPC is shown in Figure 2. We say that a CPC is (strong) selective if for any choice of weights $\mathbf{w}$ the corresponding PC $\mathtt{M}_{\mathbf{w}}$ is (strong) selective. Note that if some PC with positive weights in a CPC is (strong) selective, then by definition, all other PCs must also be (strong) selective.

2.2.1 *Credal Classification.* A most common task that PCs perform is probabilistic classification, that consists in obtaining the most probable class associated to a given observation using the model. Since CPCs define more than one single model, they induce several possible maximizers [Troffaes 2007]. A very popular criteria for decision making with imprecise probability models is *Credal Classification*, which is based on the principle of maximality [Zaffalon 2002]. Given a class variable Y, evidence

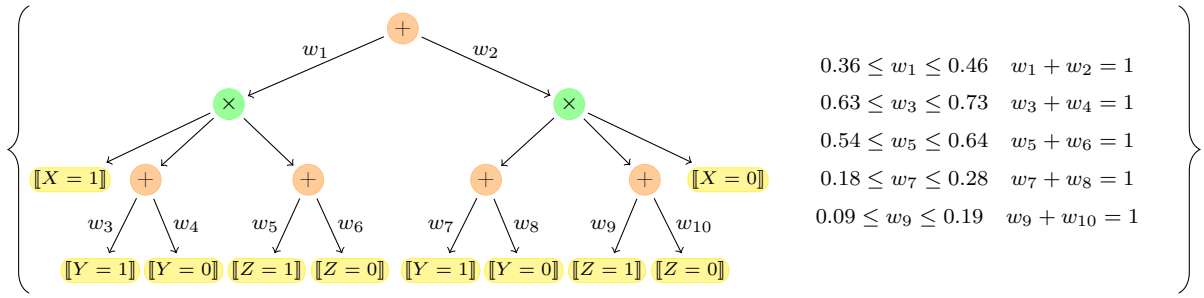

Fig. 2. Selective Credal Probabilistic Circuit obtained by 0.1-contamination of a PC

$\mathbf{X}^{\mathrm{e}} = \mathbf{e}$, and a CPC $\{\mathtt{M}_{\mathbf{w}} : \mathbf{w} \in \mathcal{C}\}$ with scope $\mathbf{X}^{\mathrm{e}}, Y$, we say that an assignment y' for Y credally dominates another assignment y'' if

$$\delta_{\mathtt{M}}(y', y'') = \min_{\mathbf{w} \in \mathcal{C}} [\mathtt{M}_{\mathbf{w}}(y', \mathbf{e}) - \mathtt{M}_{\mathbf{w}}(y'', \mathbf{e})] > 0\,. \tag{1}$$

A maximal set consists of classes that are not credibly dominated by any other class.

In [Mauá et al. 2018], the authors derived a polynomial-time algorithm for deciding the equation above in the case of CPCs whose network structure is tree-shaped (i.e., every internal node has at most one parent). We now review that algorithm, which propagates information from the leaves (input) to the root node (output), differentiating between the type of node at which it operates. For a **sum node** $\mathtt{M}$ with children $\mathtt{M}_1, \ldots, \mathtt{M}_n$ and local weights $\mathbf{w}_i \in \mathcal{C}_i$, the algorithm computes:

$$\delta_{\mathtt{M}}(y', y'') = \min_{\mathbf{w} \in \mathcal{C}} \sum_{i=1}^{n} \mathbf{w}_i [\mathtt{M}_i(y', \mathbf{e}) - \mathtt{M}_i(y'', \mathbf{e})] = \min_{\mathbf{w}_i \in \mathcal{C}_i} \sum_{i=1}^{n} \mathbf{w}_i \delta_{\mathtt{M}_i}(y', y'')\,. \tag{2}$$

For a **product node** $\mathtt{M}$ with children $\mathtt{M}_1, \ldots, \mathtt{M}_n$ such that Y is in the scope of $\mathtt{M}_1$ (and no other), the algorithm computes:

$$\delta_{\mathtt{M}}(y', y'') = \min_{\mathbf{w} \in \mathcal{C}} [\mathtt{M}_1(y', \mathbf{e}_1) - \mathtt{M}_1(y'', \mathbf{e}_1)] = \delta_{\mathtt{M}_1}(y', y'') \prod_{i=2}^{n} \text{opt } \mathtt{M}_i(\mathbf{e}_i)\,, \tag{3}$$

where $\mathbf{e}_i$ denotes the projection of $\mathbf{e}$ into the scope of $\mathtt{M}_i$, and opt $=$ max if $\delta_{\mathtt{M}_1}(y', y'') < 0$ and opt $=$ min if $\delta_{\mathtt{M}_1}(y', y'') \geq 0$. The first term in Eq. 3 denotes the recursive computation on the sub-PC $\mathtt{M}_1$. Finally, if $\mathtt{M}$ is a **leaf node** representing an indicator variable then the algorithm computes:

$$\delta_{\mathtt{M}}(y', y'') = \begin{cases} -1 & \text{if } \mathtt{M} \text{ is } [\![Y = y'']\!], \\ 1 & \text{if } \mathtt{M} \text{ is } [\![Y = y']\!] \text{ or is consistent with } \mathbf{e}, \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

3. HANDLING NON-IGNORABLE MISSING DATA USING GENERATIVE RANDOM FOREST

We now review some background on handling non-ignorable missing data, and describe our contribution in this work.

3.1 Response Model

The presence or absence of a response (observation) for a given variable can be represented as an associated Boolean random variable, by extending the sample space and generative model accordingly. Let $\mathbf{x}$ be a dataset with missing and observed values $\mathbf{x} = \{\mathbf{x}^{\text{obs}}, \mathbf{x}^{\text{miss}}\}$, where x_j^i represents the value of variable X_j in the instance i, and $x_{j,v}$ is an indicator variable that represents that X_j takes on

value v. Let $\mathbf{r}$ be a matrix of response-data indicators where $r_j^i = 1$ if x_j^i is observed and 0 otherwise. The joint distribution over the data and response indicators is given by:

$$P_{\mathbf{w},\mu}(\mathbf{X}, \mathbf{R}) = P_{\mathbf{w}}(\mathbf{X}) P_{\mu}(\mathbf{R}|\mathbf{X}) \,, \tag{5}$$

where $P_{\mathbf{w}}(\mathbf{X})$ is the *data model* (or sampling model), $P_{\mu}(\mathbf{R}|\mathbf{X})$ is the *selection model* or *response model*, and w and μ are parameters that specify the models.

We categorize missing data according to the conditional independences observed or assumed in the response model [Little and Rubin 2014]. Missing data are missing completely at random (MCAR) when the missing-data mechanism does not depend on any variable. Missingness obtained at random (MAR) is produced when the missing data mechanism depend only on the observed values We say that the data missingness is non-ignorable or Missing not at Random (MNAR) when the distribution of missing data mechanism does depend on missing and observed values [Rubin 1976], that is,

$$P_{\mu}(\mathbf{r}|\mathbf{x}) = P_{\mu}(\mathbf{r}|\mathbf{x}^{\text{obs}}, \mathbf{x}^{\text{miss}}). \tag{6}$$

To incorporate the response model in a joint distribution specified as a PC, we propose to augment the data model, learned from complete or MAR data, with a response model, as follows. Starting with this fixed structure and parameters, the augmentation consists in replacing each leave indicator node $[\![X_j = v]\!]$ by a circuit rooted with a product node with two children, the first one is the original indicator $[\![X_j = v]\!]$ and the last one is a sum node with response indicators $[\![R_{j,v} = 0]\!]$ and $[\![R_{j,v} = 1]\!]$ as children and weights $\mu_{j,v}$ and $1-\mu_{j,v}$. This structure implicitly assumes that each response variable R_j depends only on its corresponding data model variable, X_j, and of no other. More sophisticated response models can be built if additional information is present. Figure 3 shows the augmentation of data model structure for variables X and Z. The parameters associated to the response indicators

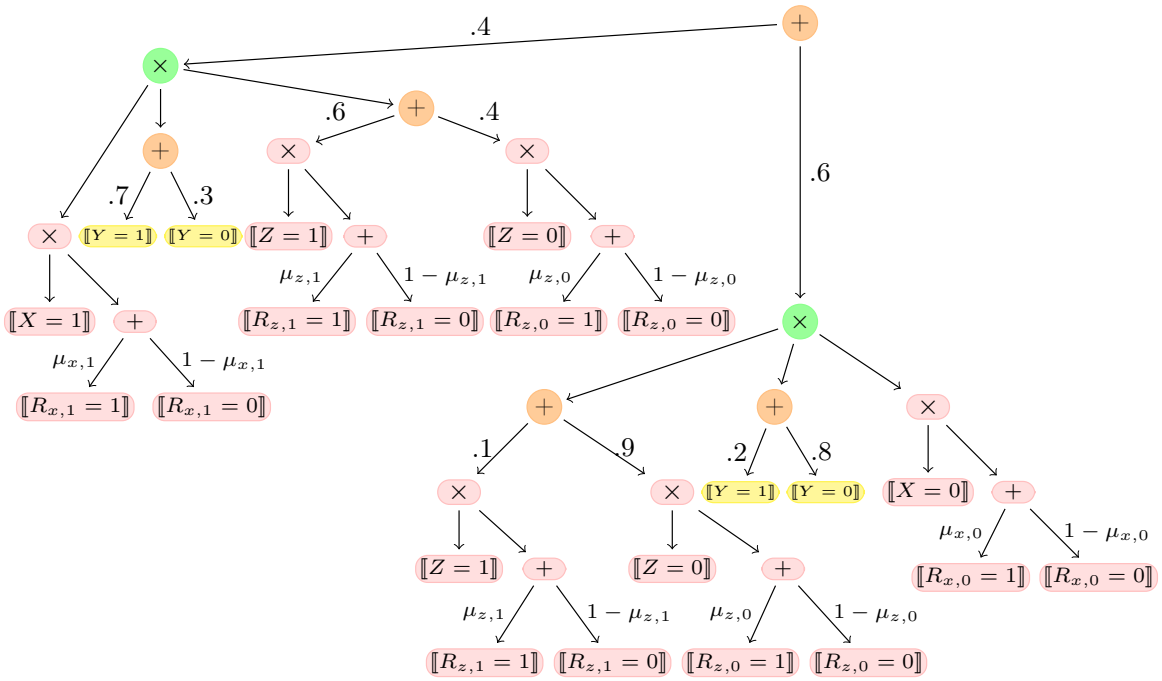

Fig. 3. Augmentation of PC with response model (pink nodes) associated to missingness processes for variables X and Z. In this example, the missing values for Y are generated by missing complete at random process. The resulting structure is not tree-shaped, although some leaves are shown duplicated, for the sake of clarity.

are obtained from a significant MNAR sample as:

$$\mu_{j,v} = P_{\mu_j}(R_j = 1|X_j = v) = \frac{P^{\text{MNAR}}(X_j = v, R_j = 1)}{P^{\text{MNAR}}(X_j = v)}\,. \tag{7}$$

The probabilities in the numerator can be estimated from the MNAR sample as relative frequencies as they mentioned only observable quantities. The denominator in Equation 7 however includes cases where the value of X_j is missing not at random (i.e., $R_j = 0$) and thus a relative frequency estimator is biased and inconsistent. In the presence of a complete or missing at random sample we can compute the denominator of Equation 7, by noting that:

$$P^{\text{MNAR}}(X_j = v) = P^{\text{complete}}(X_j = v) = P^{\text{MAR}}(X_j = v) = P^{\text{MAR}}(X_j = v|R_j = 1)\,.$$

In the augmented model, we perform inference by marginalizing missing variables at data indicators and setting the corespondent values in response indicators to deal with the missing process, the indicator $[\![R_{j,v} = 1]\!]$ propagates 1 when X_j is observable and $[\![R_{j,v} = 0]\!]$ when is missing.

3.2 CReM: A Partially Specified Response Model

Consider a PC $\mathtt{M}(\mathbf{X}, Y)$, possibly learned from some (complete or MAR incomplete) dataset of realizations of variables $\mathbf{X}, Y$, and representing our data model. Suppose we are interested in using our model to predict the value of a target variable Y given a configuration $\mathbf{x}$ of the variables such that some of its values are missing, and we do not want to assume MAR. Suppose further that we do not have access to the missingness mechanism. We can then augment our PC with the response model described and use it to draw inferences about Y. The only impediment is the the estimation of the parameters $\mu_{j,v}$ in Equation 7, as it relies on unseen and unavailable data. Very often, however, we can rely on expert domain knowledge to provide rough estimates for those parameters. Being only approximate, we are better subscribing for a partial specification in the form of probability intervals [Cozman 2000]. If we have access to an MNAR sample of data (say, at prediction time), we can instead derive such bounds from Equation 7, by considering all possible values for the denominator $P(X_j = v|R_j = 1)$:

$$\mu_{j,v} \in \left[\frac{n_{j,v}}{n_{j,v} + n_{\text{miss}}}, 1\right] \tag{8}$$

where $n_{j,v}$ denotes the number of occurrences of $X_j = v$ in the MNAR dataset of size n with n_{miss} missing values for X_j.

In either case, the result is an augmented Credal Probabilistic Circuit, where imprecision occurs only at edges connected to input indicator nodes (viz. those edges associated with response indicators) that we named CReM. As the respective network structure is not tree-shaped, we cannot in principle use the algorithm described in Section 2.2 to compute the dominance criterion in Equation 1. However, one can show, in much the same way as the conservative tractable rule in [Villanueva et al. 2021], that for the special case of GeDTs that satisfy determinism and strong selectivity, that algorithm correctly computes credal dominance in linear time. We omit the proof for lack of space.

4. EXPERIMENTS

We empirically evaluate the ability of our proposed method in producing set-valued classifications and in assessing the robustness of "precise" classifications, under the presence of non-ignorable missing feature values.

To this end, we learn class-factorized GeFs from some well-known complete datasets for density estimation and classification [Davis and Domingos 2010; Goldberg et al. 2001], using the algorithm in [Correia et al. 2020]. The characteristics of the datasets appear in Table I. Missing test values are

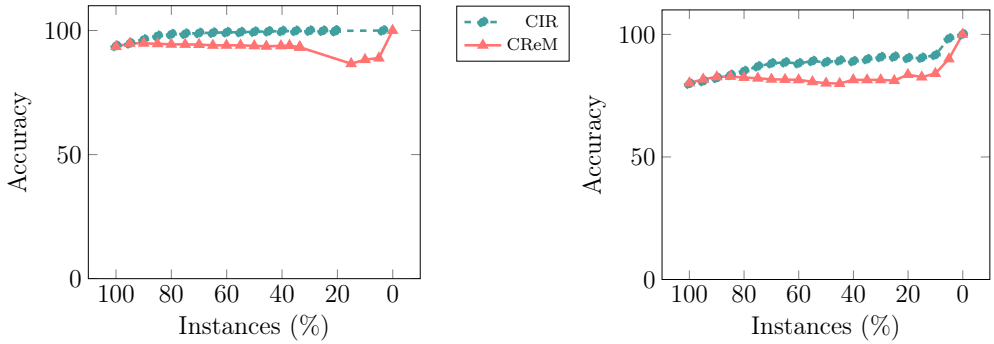

Fig. 4. Accuracy vs percentage ($\%Instances$) of the dominant instances for CIR vs Credal Response model using Nltcs(left) and Dna (right) datsets, where robustness satisfy $\delta_{\mathtt{M}}(y', y'') > \delta$, $\delta_{\mathtt{M}}$ is defined as in Eq. (1), and y' is the class with higher δ.

simulated using a MNAR mechanism. The average number of missing values per instance is denoted as AvM, and Y as number of classes.

Dataset	Vars	Test	AvM	Train	Y	Acc	CIR				CReM			
							$SAcc$	$DAcc$	DR	$RAcc$	$SAcc$	$DAcc$	DR	$RAcc$
Dna	180	1,186	24.5	1,600	2	79.1	92.3	76.9	69.1	88.9	84.4	80.2	91.7	83
Jester	10	7,486	3.6	17,467	5	36.4	96.4	21.3	2.7	41.5	59.6	31.7	36.6	37.9
Insurance	27	2,400	1.3	5,600	3	78.2	85.2	77.4	84.1	84.5	79.9	78	95.7	84.3
Nltcs	16	3,236	1.1	16,181	2	93.9	98.0	90.9	85.7	97.7	95.1	93.3	96.4	94.9

Table I. Characteristics of the datasets used and relevant performance metrics: accuracy (Acc), Set Accuracy ($SAcc$), Discounted Accuracy ($DAcc$), Determinacy Rate (DR), and accuracy of determinate classifications ($RAcc$). See text for explanation.

We also report in Table I the performance of the precise classifier, as well as relevant performance metrics of our Credal Response Model (CReM) predictions and the Conservative Inference Rule (CIR) proposed in [Villanueva et al. 2021]. Set Accuracy ($SAcc$) measures the percentage of (indeterminate or determinate) classifications that contains the true class, Discounted Accuracy ($DAcc$) measures the percentage of (determinate or indeterminate) of classifications that contain the true class, weighted by the reciprocal of set size, DR shows the percentage of determinate classifications (the ones with a single maximal class) and Precise Accuracy ($RAcc$) measures accuracy among determinate classifications.

According to the results, in comparison to CIR, CReM obtains smaller set and precise accuracies with a significantly higher determinacy rate, leading to an overall improved discounted accuracy. The effect of such a trade-off is also shown in the left plot in Figure 4, which displays precise accuracy of classifications made either by CReM or CIR selecting the class with the highest value of δ, and sorted by that same value for dataset Nltcs (left) and Dna (rigth). We see here that CReM is less adequate than CIR at judging robustness of such instances.

5. CONCLUSIONS

We developed a new tractable method to perform predictive inference under non-ignorable missing data with a partially specified response model of missingness using Generative Random Forest. Experiments with realistic data showed that our method delivers less biased (probabilistic) classifications than approaches that assume missing at random and are more determinate than similar existing overcautious approaches.

REFERENCES

Azur, M. J., Stuart, E. A., Frangakis, C., and Leaf, P. J. Multiple imputation by chained equations: what is it and how does it work? *International Journal of Methods in Psychiatric Research* 20 (1): 40–49, 2011.

Choi, Y., Vergari, A., and Van den Broeck, G. Probabilistic circuits: A unifying framework for tractable probabilistic models, 2020.

Correia, A., Peharz, R., and de Campos, C. P. Joints in random forests. In *Proceedings of the neural information processing systems*. Vol. 33. pp. 11404–11415, 2020.

Cozman, F. G. Credal networks. *Artificial intelligence* 120 (2): 199–233, 2000.

Davis, J. and Domingos, P. Bottom-up learning of Markov network structure. In *Proceedings of the 27th International Conference on Machine Learning (ICML)*. pp. 271–280, 2010.

Goldberg, K., Roeder, T., Gupta, D., and Perkins, C. Eigentaste: A constant time collaborative filtering algorithm. *information retrieval* 4 (2): 133–151, 2001.

Khosravi, P., Choi, Y., Liang, Y., Vergari, A., and Van den Broeck, G. On tractable computation of expected predictions. In *Advances in Neural Information Processing Systems 32 (NeurIPS)*, 2019.

Kisa, D., Van den Broeck, G., Choi, A., and Darwiche, A. Probabilistic sentential decision diagrams. In *Proceedings of the 14th International Conference on Principles of Knowledge Representation and Reasoning (PKDD)*. pp. 1–10, 2014.

Levi, I. *The Enterprise of Knowledge. An Essay on Knowledge, Credal Probability, and Chance*. MIT Press, Cambridge, 1980.

Liang, Y. and Van den Broeck, G. Learning logistic circuits. In *Proceedings of the 33rd Conference on Artificial Intelligence (AAAI)*, 2019.

Little, R. J. A. and Rubin, D. B. Nonignorable Missing-Data Models, 2014.

Marlin, B. M., Zemel, R. S., Roweis, S. T., and Slaney, M. Recommender systems: Missing data and statistical model estimation. In *Proceedings of the 22nd International Joint Conference in Artificial Intelligence (IJCAI)*, 2011.

Mauá, D. D., Conaty, D., Cozman, F. G., Poppenhaeger, K., and de Campos, C. P. Robustifying sum-product networks. *International Journal of Approximate Reasoning*, 2018.

Peharz, R., Gens, R., and Domingos, P. Learning selective sum-product networks. In *Proceedings of the Workshop on Learning Tractable Probabilistic Models*, 2014.

Peharz, R., Vergari, A., Stelzner, K., Molina, A., Shao, X., Trapp, M., Kersting, K., and Ghahramani, Z. Random sum-product networks: A simple and effective approach to probabilistic deep learning. In *Proceedings of The 35th Uncertainty in Artificial Intelligence Conference (UAI)*, 2020.

Poon, H. and Domingos, P. Sum-product networks: A new deep architecture. In *Proceedings of the 27th Conference on Uncertainty in Artificial Intelligence (UAI)*. pp. 337–346, 2011.

Rahman, T., Kothalkar, P., and Gogate, V. Cutset networks: A simple, tractable, and scalable approach for improving the accuracy of Chow-Liu trees. In *Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML-PKDD)*, T. Calders, F. Esposito, E. Hüllermeier, and R. Meo (Eds.). pp. 630–645, 2014.

Rubin, D. B. Inference and missing data. *Biometrika* 63 (3): 581–592, 1976.

Shao, X., Alejandro Molina, A. V., Stelzner, K., Peharz, R., Liebig, T., and Kersting, K. Conditional sum-product networks: Imposing structure on deep probabilistic architectures. In *Proceedings of the 10th International Conference on Probabilistic Graphical Models (PGM)*, 2020.

Shen, Y., Choi, A., and Darwiche, A. A tractable probabilistic model for subset selection. In *Proceedings of the 33rd Conference on Uncertainty in Artificial Intelligence (UAI)*, 2017.

Shen, Y., Goyanka, A., Darwiche, A., and Choi, A. Structured Bayesian networks: From inference to learning with routes. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence (AAAI)*, 2019.

Troffaes, M. C. Decision making under uncertainty using imprecise probabilities. *International journal of approximate reasoning* 45 (1): 17–29, 2007.

Villanueva, J., Mauá, D., and Antonucci, A. Cautious classification with data missing not at random using generative random forests. In *European Conference on Symbolic and Quantitative Approaches with Uncertainty*. Springer, pp. 284–298, 2021.

Zaffalon, M. The naive credal classifier. *Journal of statistical planning and inference* 105 (1): 5–21, 2002.

Zheng, K., Pronobis, A., and Rao, R. P. N. Learning graph-structured sum-product networks for probabilistic semantic maps. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI)*, 2018.