HAL
open science

Interactive anomaly detection in mixed tabular data using Bayesian networks

Evan Dufraisse, Philippe Leray, Raphaël Nedellec, Tarek Benkhelif

To cite this version:

Evan Dufraisse, Philippe Leray, Raphaël Nedellec, Tarek Benkhelif. Interactive anomaly detection in mixed tabular data using Bayesian networks. 10th International Conference on Probabilistic Graphical Models (PGM 2020), Sep 2020, Aalborg, Denmark. hal-03014622

HAL Id: hal-03014622

https://hal.science/hal-03014622v1

Submitted on 17 Mar 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Interactive Anomaly Detection in Mixed Tabular Data using Bayesian Networks

Evan Dufraisse[1] EDUFRAISSE@GMAIL.COM
Philippe Leray[1] PHILIPPE.LERAY@LS2N.FR
Raphaël Nedellec[2] RNEDELLEC@TALEND.COM
Tarek Benkhelif[2] TBENKHELIF@TALEND.COM
[1]*Université de Nantes, LS2N UMR CNRS 6004, Nantes, France*
[2]*Talend, Nantes, France*

Abstract

The last decades improvements in processing abilities have quickly led to an increasing use of data analyses implying massive data-sets. To retrieve insightful information from any data driven approach, a pivotal aspect to ensure is good data quality. Manual correction of massive data-sets requires tremendous efforts, is prone to errors, and results being really costly. If knowledge in a specific field can often allow the development of efficient models for anomaly detection and data correction, this knowledge can sometimes be unavailable and a more generic approach should be found. This paper presents a novel approach to anomaly detection and correction in mixed tabular data using Bayesian Networks. We present an algorithm for detecting anomalies and offering correction hints based on Jensen scores computed within the Markov Blankets of considered variables. We also discuss the incremental corrections of detection model using user's feedback, as well as additional aspects related to discretization in mixed data and its effects on detection efficiency. Finally we also discuss how functional dependencies can be managed to detect errors while improving faithfulness and computation speed.

Keywords: Bayesian Networks, Anomaly Detection, Outlier Detection, Functional Dependencies, Mixed Data

1. Introduction

When considering tabular data, anomalies (also refered as outliers in the literature) are often thought as odd and isolated points within a multi-dimensional space. This has led many anomaly detection algorithms to be designed using exclusively space-based approaches relying on distance and density measures within neighborhoods of data points. It has been shown by (Lu et al., 2018) that the detection of probabilistic dependencies violations using Bayesian Networks was complementary to space-based approaches and could allow the detection of a broader range of anomalies when combined. The idea of combining anomaly detection algorithms isn't new and ensemble techniques are already used as an attempt to join strengths and reduce bias of different algorithms (Aggarwal and Sathe, 2017). In this respect, considering the core difference and complementarity between distance-based and probabilistic-based detection algorithms, it seems both natural and essential to combine these approaches.

Functional dependencies are not rare when dealing with tabular data. This kind of dependency can be useful for detecting anomalies, but also problematic if we try to learn probabilistic dependencies. Our objective is to interactively detect and correct anomalies by using both functional and

probabilistic dependencies facing mixed data. Several approaches have been described for continuous or categorical data, but none to our knowledge tries to tackle the case of mixed data.

In former studies, various scoring methods have been employed to identify anomalies : direct likelihood ranking of individuals (Cansado and Soto, 2008), patterns recognition from manually defined thresholds (Babbar and Chawla, 2012) and normalized likelihood-based scores (Kirk et al., 2014; Rashidi et al., 2011). Manual threshold pattern recognition suffers from the need of defining rules on what is anomalous and direct likelihood estimation is prone to detect "rare" rather than "incoherent" individuals. We thus decided to use a normalized likelihood-based score similar to what Kirk et al. (2014) and Rashidi et al. (2011) mention, and often called Jensen score. We offer a new usage of this score by defining a different exploration strategy built around an extended definition of Markov blankets for a particular Bayesian Network construction. Our article also distinguishes itself by presenting a way of integrating functional dependencies and adding value through interactivity with an expert in the process of anomaly detection.

When used with Bayesian Networks, mixed data is generally subjected to a discretization step for which several uni-modal or multi-modal algorithms exist (Boullé, 2006; Mabrouk et al., 2014; Chen et al., 2017). Inspired by the work of Mabrouk et al. (2014), we've designed an new uni-modal discretization process to be used for anomaly detection networks.

Motivated by the large presence of pairwise functional dependencies in today's databases, we mainly focused on the treatment of this type of dependence. For this aspect, we rely on the results of Rahier (2018), who described a way of learning a Bayesian network when facing data including functional dependencies, by first detecting these dependencies, and then by restricting a discrete Bayesian network structure learning algorithm to a subset of functionally independent variables to discover their probabilistic dependencies. Other publications focus on the discovery of functional dependencies involving more variables (Caruccio et al., 2016).

Section 2.2 offers a general overview of our interactive anomaly detection system dealing with mixed tabular data. Section 2.3 describes the way we decide to deal with mixed data by discretizing continuous variables. Section 2.4 is dedicated to functional dependencies detection and their use for identifying a first set of anomalies. Once functional dependencies have been processed, section 2.5 presents the hybrid Bayesian network we learn to capture the probabilistic dependencies with original variables. In section 2.6, we give details about our anomaly detection approach based on an adapted use of the Jensen score to detect anomalies in extended Markov blankets of our hybrid Bayesian network. Section 2.7 briefly deal with the model update when the expert decides to correct values in the data-set. Section 3 is dedicated to the empirical evaluation of our proposition. Section 4 concludes on the contribution of this paper and our perspectives.

2. Contribution

2.1 Notations

Let us introduce the notations used in the following sections. Our algorithm is dealing with n random variables, with l discrete variables $\{X_1,.., X_l\}$ and $n - l$ continuous ones $\{\mathring{X}_{l+1},.., \mathring{X}_n\}$. $\{X_{l+1},.., X_n\}$ will denote the discretized counterpart of these continuous variables.

$\mathring{x}_i$ will represent an instance of a continuous variable, when x_i will represent its discretized value, or the value of a discrete variable. $Val(X_i)$ will denote the set of possible values for the discrete variable X_i.

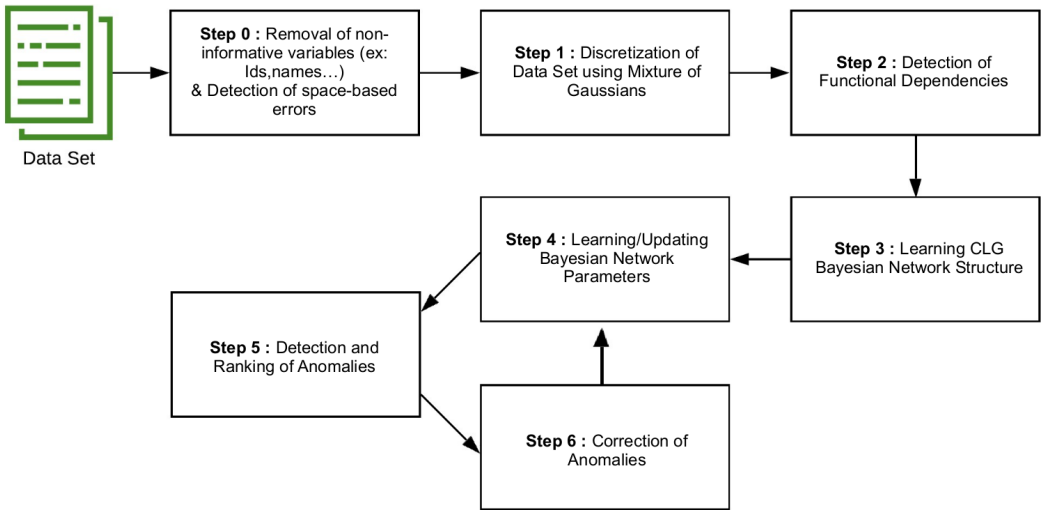

Figure 1: Architecture of our interactive anomaly detection system dealing with mixed data.

$\mathring{\mathbf{X}} = \{X_1,.., X_l, \mathring{X}_{l+1},.., \mathring{X}_n\}$ is the set of original variables, $\mathbf{X} = \{X_1,.., X_l, X_{l+1},.., X_n\}$ is the set of the discrete and discretized variables, and $\overline{\mathring{\mathbf{X}}} = \{X_1,.., X_l, \mathring{X}_{l+1},.., \mathring{X}_n, X_{l+1},.., X_n\}$ is the original set of variables extended with the discretized ones.

$\mathring{D}$ and D respectively denote the original mixed data-set (over $\mathring{\mathbf{X}}$) and the discretized one (over $\mathbf{X}$), and N is the number of instances in the data-set.

$\mathring{\mathbf{X}}_{|s}$, $\mathbf{X}_{|s}$, $\overline{\mathring{\mathbf{X}}}_{|s}$ will correspond to subsets of variables, with the corresponding datasets $\mathring{D}_{|s}$, $D_{|s}$.

2.2 Overview

Figure 1 presents the general overview of our interactive anomaly detection system dealing with mixed tabular data.

As explained in introduction, and represented in the step 0 in Figure 1, we assume that distance-based errors, which can't be detected using probabilistic-based approaches, have been detected and corrected as a pre-processing step. Removing non-informative variables prior to execution can also speed up the process depending on the implementation.

Our proposition really starts with the discretization of the considered mixed data-set in step 1. We proceed here to an unimodal discretization that will be described in section 2.3.

Once the data-set is discretized, we detect in step 2 functional dependencies which could exist within the data-set. As part of this study we are only considering functional dependencies between pairs of variables, as presented in section 2.4. At the end of this step, an outlier detection and correction can be performed based on functional dependencies only. Finally, one subset of "functionally independent" variables is selected.

In step 3, we then learn the structure of a Conditional Linear Gaussian Bayesian Network (CLGBN) capturing all the probabilistic dependencies between the previously selected variables. Section 2.5 gives more details about this step.

Step 4 is dedicated to the parameter learning of this Bayesian network, from the original dataset, or its update when the dataset will be corrected in step 6 (section 2.7).

The anomaly detection task (step 5) presented in section 2.6 uses probabilistic dependencies previously discovered to score and rank potential anomalies and offer hints of correction.

Step 6 is the interactive part of the process, where some detected and ranked anomalies are presented to one expert in charge of validating or not the proposition in order to correct the anomalies. Based on the expert's correction choices, probability tables can be updated in step 4 and the process can iterate until no more anomalies are validated by the expert.

2.3 Discretization

In the context of mixed data, discretization is in itself an important subject in Bayesian network learning and more generally in Machine Learning. Using an arbitrary discretization by frequency or by even intervals generally leads to sub-optimal networks. Even though this discretization arbitrariness can be reduced using multi-modal approaches (Boullé, 2006; Mabrouk et al., 2015; Chen et al., 2017), these often come at a considerable computing cost.

For this reason, in this first version of our anomaly detection system, we use a simple uni-modal discretization approach based on the Expectation-Maximization algorithm (Dempster et al., 1977; McLachlan and Krishnan, 2008). For each continuous variable $\mathring{X}_i \in \{\mathring{X}_{l+1},.., \mathring{X}_n\}$ we learn a mixture of $G_i \geq 2$ Gaussians and discretize them by assigning the label of the most explaining Gaussian to entry i of every instances x.

In order to prevent over-fitting, the number of Gaussians G_i for each mixture is determined by applying an usual stepwise greedy search to optimize the BIC criterion, but some other criteria can be used (Celeux et al., 2018).

The parameters of the Gaussian distributions have been initialized by using a classical grid initialization between the bounds of the considered variable for the mean and a constant standard deviation.

2.4 Functional Dependencies

A functional dependency between one variable X_i and a set of variables $\boldsymbol{W}$ is usually defined by the fact that for every valid instance of $\boldsymbol{W}$, that instance of $\boldsymbol{W}$ uniquely determines the value of X_i. This can be seen as a special case of probabilistic dependency and be detected if the conditional entropy $H(X_i|\boldsymbol{W}) = 0$.

Presence of functional dependencies are known to render conditional independence tests between variables unreliable, and thus would lead to unfaithful graph structures when learning our Bayesian network in the next step (Luo, 2006; Rodrigues de Morais et al., 2008; Mabrouk et al., 2014; Rahier, 2018). For this reason, we offer to detect in this step functional dependencies and correct found functional violations.

Being interested in detecting functional dependencies in the context of anomaly detection, we need to expand our definition of functional dependencies to what is often defined as "quasi-determinism" or "relaxed functional dependencies", so that potential errors don't prevent their detection. Relaxed functional dependencies are functional dependencies that almost hold except on a "small" subset of entries. Several algorithms have been published for relaxed dependencies mining (Caruccio et al., 2016; Bleifuß et al., 2016; Caruccio et al., 2019).

In our architecture, we propose to use the results of Rahier (2018) which offer a simple approach to detect and cluster pairwise quasi-deterministic dependencies between discrete variables by relying on conditional entropy computations. The algorithm essentially considers each pair of variables

(X_i, X_j) and check if the conditional entropy is inferior to a given ϵ. In short, if $H(X_i|X_j) < \epsilon$, X_j is a "functional" parent candidate for X_i.

Rahier (2018) finally structures the functional dependencies discovered as a spanning forest F where each X_i has at most one "functional" parent $Pa_F(X_i)$. Variables without "functional" parents are called roots, and are functionally independent. We will denote later R this set of roots, $\mathbf{X}_{|R}$ the subset of functionally independent discrete variables and $\mathring{\mathbf{X}}_{|R}$ the corresponding subset in the original variables.

After their detection, we learn the CPD $P(X_i|Pa_F(X_i))$ of each functional dependency present in F and we create a Boolean function f_i that allows the identification of values (x_i, x_j) of $(X_i, Pa_F(X_i))$ that violate the quasi-deterministic functional dependency discovered.

$$f_i(X_i = x_i, Pa_F(X_i) = x_k) = 1 - \delta_{x_i, x_i^k} \tag{1}$$

where

$$x_i^k = \underset{x_i \in Val(X_i)}{\operatorname{argmax}} \; P(X_i = x_i|Pa|F(X_i) = x_k) \tag{2}$$

For each instance in D we then check the values of f_i functions, whenever a f_i returns true we report the error to the user along with the involved variables $(X_i = x_i, Pa_F(X_i) = x_k)$ and ask for a correction of this anomaly.

2.5 CLGBN structure Learning

The objective of this section is to learn a Conditional Linear Gaussian Bayesian Network (CLGBN, Lauritzen and Wermuth 1989) that captures the probabilistic dependencies between variables $\mathring{\mathbf{X}}_{|R}$, i.e. once the functional dependencies have been processed in section 2.4. Restricting ourselves to this subset of variables ensures the absence of functional dependencies and allows to maintain the same amount of information while reducing the number of variables.

Rather than learning directly this CLGBN from the mixed data, we present a two-step alternative process inspired from the work of Mabrouk et al. (2014) and relying on the Gaussian mixtures found during discretization step (section 2.3). First, a simple discrete Bayesian Network structure is learned on $\mathbf{X}_{|R}$, the discretized version of the variables This structure learning step is generic and can be made using any preferred algorithm.

Second step consists in creating the final CLGBN over variables $\overline{\mathring{\mathbf{X}}}_{|R}$ by augmenting the discrete Bayesian Network learned over $\mathbf{X}_{|R}$ with the continuous variables present in $\mathring{\mathbf{X}}_{|R}$ and adding a conditional probabilistic dependency $P(\mathring{X}_i|X_i)$ for each of these continuous variables. These conditional distribution will be defined with the Gaussian mixtures learned in section 2.3.

The obtained CLGBN contains both the probabilistic dependencies between the continuous variables and their discrete counterparts, and the probabilistic dependencies discovered between all the discrete or discretized variables.

2.6 Anomaly Detection and Correction

Anomaly detection using Bayesian Network whether on continuous/discrete data or temporal series has been subject of several publications (Kirk et al., 2014; Das and Schneider, 2007; Rashidi et al., 2011; Babbar and Chawla, 2012). Some approaches have been based solely on reporting individuals with lowest likelihoods of occurrence, others on defining manually patterns rules to identify anomalous individuals (Babbar and Chawla, 2012). Our detection approach distinguishes itself both from

the use of the CLGBN learned in the previous section and from the application of an anomalousness scoring that considers Markov Blanket of variables.

2.6.1 MARKOV BLANKET BASED SCORING OF ANOMALOUSNESS

As part of this architecture, we use a scoring method defined in eqn 3 relying on a normalized likelihood score called the Jensen score (Jensen and Nielsen, 2007; Kirk et al., 2014; Rashidi et al., 2011).

$$J(x_1, x_2, ..., x_n) = \log(\frac{P(x_1)P(x_2)...P(x_n)}{P(x_1, x_2, ..., x_n)}) \tag{3}$$

The intuition behind the use of this score is the following. Given a conjunction of variables and their values, if the probability of occurrence of this conjunction is inferior to the product of their marginal probabilities of occurrence, we can flag the conjunction as incoherent and should report it as abnormal to the user. In other words, a conjunction of values must have a Jensen score below zero to be considered coherent.

In a Bayesian network defined over a set of variables $\mathbf{X}$, the Markov Blanket of one node X_i is by definition the set of variables $MB(X_i)$ given which X_i becomes conditionally independent from $\mathbf{X} \setminus MB(X_i)$. It means that when the values of $\mathbf{X}$ are all observed, the values of $MB(X_i)$ are the only ones that are necessary to estimate the posterior probability of X_i. A side effect is that we can estimate $J(X_i, MB(X_i))$ to determine if there is a probabilistic anomaly between the value observed for X_i and the observation of its Markov blanket.

In our CLGBN, the discretized variables present in $\overline{\mathring{\mathbf{X}}}_{|R}$ are unobserved. So, we propose to modify this Markov blanket by replacing the discretized (and unobserved) nodes $\{X_j\}$ present in $\{MB(X_i) \cup X_i\}$ by their continuous counterpart $\{\mathring{X}_j\}$ in order to obtain an extended Markov blanket $eMB(X_i)$.

In this definition, we are considering each information given by the observation of continuous variable $\mathring{X}_j = \mathring{x}_j$ as a soft evidence $P(X_j|\mathring{x}_j)$ sufficient to determine the posterior distribution of its discretized variable X_j, whatever the values of other variables dependent with X_j (as defined by Valtorta et al. (2002), by opposition to the virtual evidence defined by Pearl (1988). The use of this soft evidence is also a way to reduce the arbitrariness of the previous discretization step by not using directly the discretized value as an evidence.

By this way, this extended Markov blanket $eMB(X_i)$ contains all the necessary information to estimate the posterior distribution of X_i.

With this reasoning, we propose to compute $J(eMB(X_i) = emb_i)$ to determine if there is a probabilistic anomaly between the value observed for X_i and the observation of its extended Markov blanket, where emb_i is the observation of $eMB(X_i)$.

2.6.2 DETECTION, RANKING AND CORRECTION PROPOSAL

For each variable X_i in the original mixed variable set (restricted to the functionally independent variables) $\mathring{\mathbf{X}}_{|R}$ the scoring of anomalousness $J(eMB(X_i) = emb_i)$ for a observed configuration emb_i is computed for all the N instances of the dataset and normalized (cf. eqn 4).

$$j(eMB(X_i) = emb_i) = \frac{J(eMB(X_i) = emb_i) - E[J(eMB(X_i))]}{std(J(eMB(X_i)))} \tag{4}$$

As explained in section 2.6.1, only the instances mb_i with a positive value of the Jensen score $J(eMB(X_i) = mb_i) > 0$ are considered and detected as anomalies. In order to rank these anomalies even if they are raised by different $eMB(X_i)$, we propose to rank them with respect to their normalized scoring function $j(eMB(X_i) = emb_i)$.

One anomaly can be detected in several $eMB(X_i)$, but the more important is that one of them is proposed to the expert in order to correct it.

This potential anomaly is then proposed to the expert as a set of variables and their observed values $eMB(X_i) = emb_i$. The expert is then in charge of looking at all these variables and values and checking what pair of variable and value (X_j, x_j) could be the anomaly (where X_j belongs to eMB_i). We decide here to help the expert by proposing him the posterior distribution $P(X_j| \{emb_i \setminus x_j\})$ of each X_j candidate (or its discretized counterpart) without its potential abnormal value x_i.

2.7 Model updating

With the list of information proposed to him in the previous section, the expert can decide to correct one variable X_j with a new value x_j^* instead of the previous value x_j. We then proceed with the change of our CLBGN parameters that are dependent with this variable by updating the sufficient statistics classically used and stored during the parameter estimation phase. We make here the assumption, that such a small change in the dataset (and the repetition of these small changes caused by anomaly correction) will not change the underlying structure of our model.

3. Experiments

3.1 Datasets and implementation

For these experiments, we used the NFL2016 dataset[1] previously pre-processed which contains here about 2330 NFL players entries. Information includes both continuous variables (Jersey number, Age, Height, Weight, ...) and discrete ones (Position, Position-Group, Side, ...).

We know that three functional dependencies are present in this data-set (Position $\rightarrow$ Position-Group, Years $\rightarrow$ Level and Position-Group $\rightarrow$ Side).

Even if we don't know exactly what entries in this data-set have anomalies, we have selected this data-set because of the existence of several external sources about NFL players that allows us to fact-check all the possible anomalies detected.

We are currently working with several other data-sets, but fact-checking takes a long time, so we decided to provide here only results for this NFL data-set. As we don't know exactly where are all the possible anomalies in this data-set, the only performance measurements we can use are True and False Positive (TP, FP), after our fact-checking step.

The implementation of our architecture has been performed with our C++ library dedicated to Probabilistic Graphical Models, PILGRIM[2], with the help of ProBT[3] library for handling CLGBNs.

The BN structure learning used during step 3 is our implementation of the classical Greedy search algorithm initially proposed by Chickering et al. (1995), or an exact search with BIC scoring

1. https://data.world/tarek-benkhelif/nflplayers
2. http://pilgrim.univ-nantes.fr/
3. https://www.probayes.com/

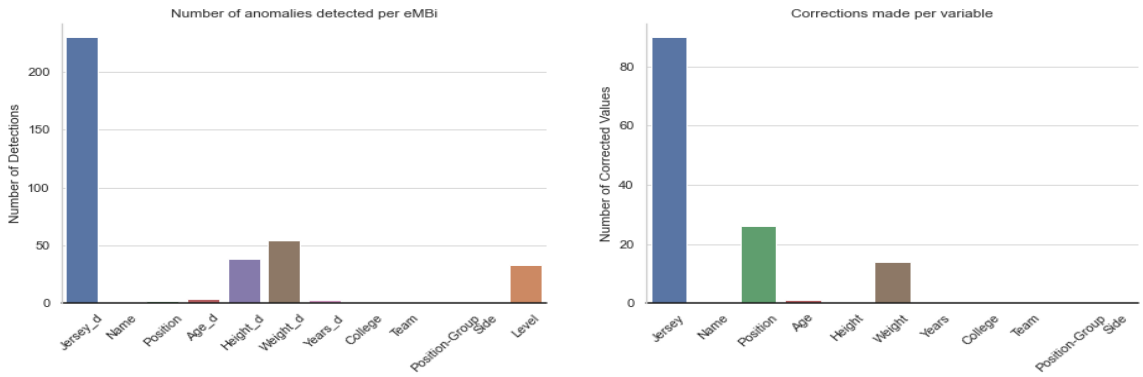

Figure 2: Distribution of detection and real anomalies detected among the variables

function. The threshold used for the detection of quasi-deterministic functional dependencies is $\epsilon = 0.02$.

3.2 Experimental design and results

3.2.1 PROBABILISTIC DETECTION OF ANOMALIES

In this experiment, we've decided to run our system without Step 2 (related to functional dependencies) in order to detect, rank and propose anomalies only with the help of our CLGBN.

Among the 2330 instances (NFL players), 364 anomalies have been detected (with some anomalies related to the same player) and reported to the expert. After fact-checking the information related to each anomaly detected, 131 detections are real anomalies, 20 detections are unsure ones, 140 are false detection, and 73 are considered as "surprising" instances, i.e. unusual values, but corresponding to the reality. In Figure 2, we can also see that we are able to detect real anomalies from both discrete (Position) and continuous (Jersey and Weight) variables.

Even if we know that this dataset includes some functional dependencies, we were able in this first experiment to illustrate the interest of the use of our CLGBN in order to detect anomalies in discrete or continuous framework.

3.2.2 INTEREST OF THE EXTENDED JENSEN SCORE

This second experiment is the same as the previous one but with a specific focus on the Jensen score values. The anomalies detected by our Jensen scores are filtered with respect to a given threshold.

Figure 3 shows us the evolution of True and False positive anomaly detection when focusing on the extended Markov blanket of Jersey variable. We can see that our scoring function is helpful for ranking anomalies. By increasing the detection threshold, only a fraction of the previous real anomalies ($TPmax$) are detected but the quality of the detection (ratio $TP/(TP+FP)$) increases.

Figure 4 presents the distribution of the (un-normalized) Jensen score computed on the extended Markov blanket for two variables Height and Weight, for real and false detections. Confirming the previous results, this Jensen score helps to detect real anomalies (the higher this score is, the more True positives we have). But, by comparing distributions between both variables, we can see that our score distributions are not comparable between two different extended Markov blankets. This observation confirms the interest of using the normalized version described in eq. 4.

Figure 3: Evolution of detection performances related to Jersey variable when Jensen scores are filtered with respect to a given threshold.

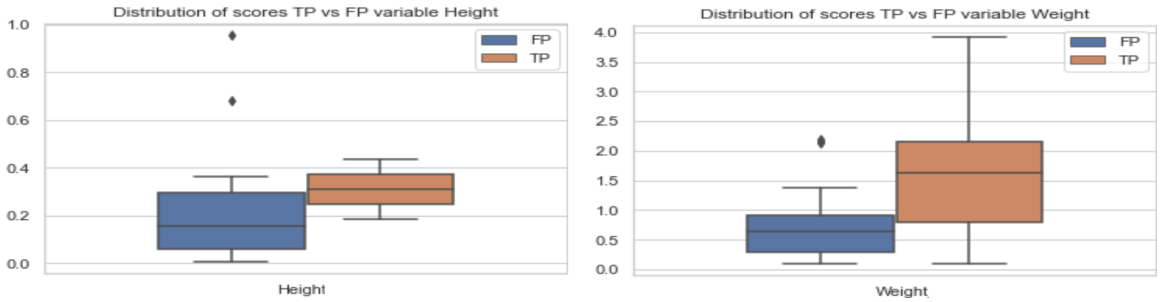

Figure 4: Comparison of (un-normalized) Scores $J(eMB)$ distribution for variables Height and Weigth for real (TP) and false detections

3.2.3 FUNCTIONAL AND PROBABILISTIC DETECTION

In this third experiment, we run once again our anomaly detection system, by also taking into account functional dependency detection.

As a first result, the three expected functional dependencies are detected in our dataset : Position $\rightarrow$ Position-Group $\rightarrow$ Side and Years $\rightarrow$ Level. No anomalies were detected during this phase, which is coherent with the results of our previous experiments where the 131 real anomalies discovered were not involving variables present in the existing functional dependencies.

The CLGBN obtained during step 3 is described in Figure 5 with or without taking into account the functional dependency detection. We can notice that both structures capture similar probabilistic dependencies, except the one between Level and Jersey. As partially observed in figure 2, some anomalies have been detected with the first CLGBN regarding Level Markov blanket. All these anomalies where False Positive. Our new CLGBN doesn't contain this dependency and those False Positive are no longer detected.

Without any filtering or action from the user, by using only our scoring function, 920 anomalies would be detected, which is about 3 times the number of detection made in the previous experiment.

We identified that 126 of the 131 FP were rightly identified, 14 of the 20 unsure and 192 of the previous 213 FP. But more than 480 detections have been made scoring the $eMB(Height)$, whereas our first experiment only detected about 40 potential anomalies for it. This surprising result

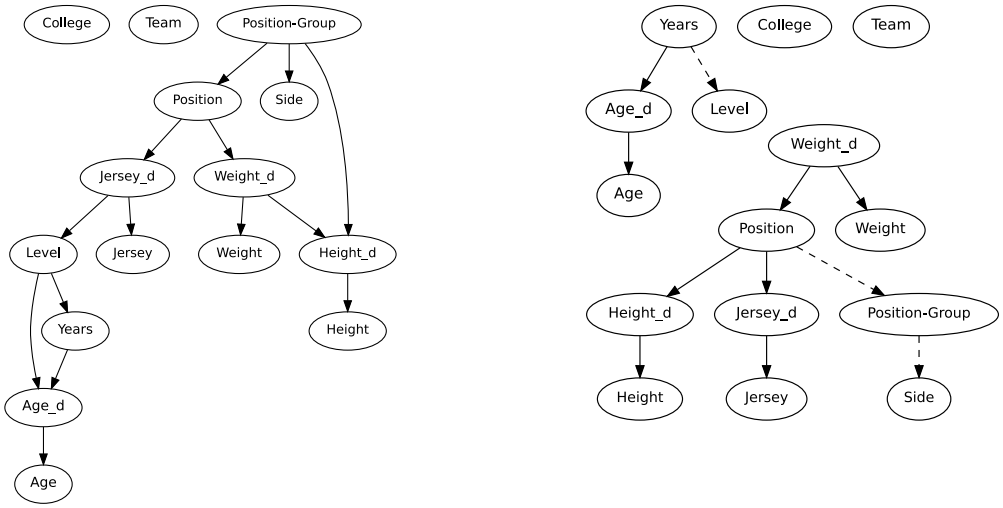

Figure 5: CLGBN obtained during step 3, without taking into account functional dependencies (left) and by taking into account the functional dependencies (right). The dotted dependencies added in the right figure correspond to the functional dependencies

is explained by the use of the Greedy Search structure learning algorithm and the disappearance of Weight in the extended Markov Blanket of Height. In the second model, Height and Weight are only conditionally independent to Position. This loss of dependency between Height and Weight generates an increase of false positives, and enforced us in our decision of using an exact structure learning algorithm when the number of variables is low. The results of this experiment are not shown due to lack of space, but the CLGBN learnt here contains the direct dependency between Weight and Height, and the anomalies detected are similar to the one detected without taking functional dependencies into account.

4. Conclusion

We proposed in this paper an architecture dedicated to interactive anomaly detection system dealing with mixed tabular data, and taking into account both functional dependencies and probabilistic ones for anomaly detection. As functional dependencies are known to perturb Bayesian network structure learning algorithms, we first detect an correct them independently. We deal with mixed data by discretizing continuous variables. We then detect functional dependencies prior to BN structure learning, considering only discrete and discretized variables. We soften the effect of discretization boundaries for the anomaly detection phase by extending the learned BN with direct dependencies between discretized variables and their continuous counterparts. This Continuous Linear Gaussian BN is finally used for detecting probabilistic anomalies. For this task, we adapted existing Jensen score to focus on specific subsets of variables, taking into account Markov blankets both for detection and correction hints.

In our first experiments with NFL dataset, we were able to show that even the probabilistic part of our architecture is able to detect real anomalies in a dataset. The same experiments confirmed the interest of our extended Jensen score, and its normalized version. Another experiment highlighted

the importance of functional dependencies, and the fact we have to resort to an improved Bayesian Network structure learning algorithm or an exact one when the number of variables is low, to ensure the quality of the Markov blanket.

This work is the first step of the development of our architecture, with several perspectives. In a very short term, we intend to perform larger experiments with other datasets in order to consolidate the interest of our proposition.

We are also planning to work on improvements of our discretization process, for instance with a multivariate process intertwined with the BN learning (Mabrouk et al., 2015) or by using mixtures of truncated Gaussians (Gonzales, 2019) instead of Gaussians in order to avoid spurious discretization effects.

Detecting more complex functional dependencies could also be interesting, even though we know we will face complexity issues.

Finally, this architecture can also be improved by working on the interaction with the expert, improving his guidance and better using his feedback to score anomalies.

References

C. C. Aggarwal and S. Sathe. *Outlier Ensembles*. Springer International Publishing, 2017. ISBN 978-3-319-54764-0.

S. Babbar and S. Chawla. Mining causal outliers using Gaussian Bayesian networks. In *IEEE ICTAI 2012*, volume 1, pages 97–104, 2012.

T. Bleifuß, S. Bülow, J. Frohnhofen, J. Risch, G. Wiese, S. Kruse, T. Papenbrock, and F. Naumann. Approximate discovery of functional dependencies for large datasets. In *Proceedings of CIKM '16*, page 1803–1812. ACM Press, 2016.

M. Boullé. MODL: A Bayes optimal discretization method for continuous attributes. *Machine Learning*, 65(1):131–165, Oct 2006.

A. Cansado and A. Soto. Unsupervised anomaly detection using bayesian networks. *Applied Artificial Intelligence*, 22(4):309–330, 2008.

L. Caruccio, V. Deufemia, and G. Polese. Relaxed functional dependencies—a survey of approaches. *IEEE Transactions on Knowledge and Data Engineering*, 28(1):147–165, Jan 2016.

L. Caruccio, V. Deufemia, and G. Polese. Mining relaxed functional dependencies from data. *Data Mining and Knowledge Discovery*, Dec 2019.

G. Celeux, S. Frühwirth-Schnatter, and C. Robert. Model Selection for Mixture Models-Perspectives and Strategies. In *Handbook of Mixture Analysis*. CRC Press, Dec. 2018.

Y.-C. Chen, T. A. Wheeler, and M. J. Kochenderfer. Learning discrete Bayesian networks from continuous data. *Journal of Artificial Intelligence Research*, 59:103–132, Jun 2017.

D. Chickering, D. Geiger, and D. Heckerman. Learning Bayesian networks: Search methods and experimental results. In *Proceedings of Fifth Conference on Artificial Intelligence and Statistics*, pages 112–128, 1995.

K. Das and J. Schneider. Detecting anomalous records in categorical datasets. In *Proceedings of ACM KDD '07*, page 220. ACM Press, 2007.

A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B*, 39:1–38, 1977.

C. Gonzales. Dealing with continuous variables in graphical models. In N. Ben Amor, B. Quost, and M. Theobald, editors, *Scalable Uncertainty Management*, pages 404–408. Springer, 2019.

F. V. Jensen and T. D. Nielsen. *Bayesian networks and decision graphs*. Information science and statistics. Springer, 2nd ed edition, 2007.

A. Kirk, J. Legg, and E. El-Mahassni. Anomaly detection and attribution using Bayesian networks. Technical report, Defence Science and Technology Organisation, Canberra, Australia, 2014.

S. Lauritzen and N. Wermuth. Graphical models for associations between variables, some of which are qualitative and some quantitative. *The Annals of Statistics*, 17:31–57, 1989.

S. Lu, L. Liu, J. Li, and T. D. Le. Effective outlier detection based on Bayesian network and proximity. In *2018 IEEE International Conference on Big Data*, pages 134–139, 2018.

W. Luo. Learning bayesian networks in semi-deterministic systems. In L. Lamontagne and M. Marchand, editors, *Advances in Artificial Intelligence*, pages 230–241. Springer, 2006.

A. Mabrouk, C. Gonzales, K. Jabet-Chevalier, and E. Chojnacki. An efficient Bayesian network structure learning algorithm in the presence of deterministic relations. In *Proceedings of ECAI'14*, page 567–572. IOS Press, 2014.

A. Mabrouk, C. Gonzales, K. Jabet-Chevalier, and E. Chojnaki. Multivariate cluster-based discretization for Bayesian network structure learning. In C. Beierle and A. Dekhtyar, editors, *Scalable Uncertainty Management*, volume 9310, page 155–169. Springer, 2015.

G. McLachlan and T. Krishnan. *The EM algorithm and extensions*. Wiley series in probability and statistics. Wiley, Hoboken, NJ, 2. ed edition, 2008.

J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufman Publishers, San Mateo, CA, 1988.

T. Rahier. *Bayesian networks for static and temporal data fusion*. PhD thesis, Grenoble Alpes University, France, 2018.

L. Rashidi, S. Hashemi, and A. Hamzeh. Anomaly detection in categorical datasets using bayesian networks. In H. Deng, D. Miao, J. Lei, and F. L. Wang, editors, *Artificial Intelligence and Computational Intelligence*, pages 610–619. Springer, 2011.

S. Rodrigues de Morais, A. Aussem, and M. Corbex. Handling almost-deterministic relationships in constraint-based Bayesian network discovery : Application to cancer risk factor identification. In *Proceedings of ESANN'08*, pages 101–106, 2008.

M. Valtorta, Y.-G. Kim, and J. Vomlel. Soft evidential update for probabilistic multiagent systems. *International Journal of Approximate Reasoning*, 29(1):71 – 106, 2002.