Temporal Exceptional Model Mining using Dynamic Bayesian Networks

Marcos L. P. Bueno[1,2], Arjen Hommersom[3], and Peter J. F. Lucas[4,5]

[1] MCS, Eindhoven University of Technology, The Netherlands
[2] iCIS, Radboud University Nijmegen, The Netherlands
[3] Department of Computer Science, Open Universiteit, The Netherlands
[4] Faculty of EEMCS, University of Twente, The Netherlands
[5] LIACS, Leiden University, The Netherlands
m.l.de.paula.bueno@tue.nl, Arjen.Hommersom@ou.nl, peter.lucas@utwente.nl

Abstract. The discovery of subsets of data that are characterized by models that differ significantly from the entire dataset, is the goal of exceptional model mining. With the increasing availability of temporal data, this task has clear relevance in discovering *deviating temporal subprocesses* that can bring insight into industrial processes, medical treatments, etc. As temporal data is often noisy, high-dimensional and has complex statistical dependencies, discovering such temporal subprocesses is challenging for current exceptional model mining methods. In this paper, we introduce Temporal Exceptional Model Mining to capture multiple and complex relationships among temporal variables of a dataset in a principled way. Our contributions are as follows: (*i*) we define the new task of temporal exceptional model mining; (*ii*) we characterize the discovery of exceptional temporal submodels using dynamic Bayesian networks by means of a new distance measure, (*iii*) we introduce a search procedure for exceptional dynamic Bayesian networks optimized by properties of the proposed distance, and (*iv*) the practical value of the proposed method is demonstrated based on simulated data and process data of funding applications and by comparisons with other exceptional model mining methods.

Keywords: Machine learning · Graphical models · Bayesian networks · Temporal data · Subgroup discovery · Exceptional model mining

1 Introduction

In many domains such as health care, engineering and workflow processes, there is an increasing availability of temporal data, often mixed with non-temporal ones, such as gender and geographical location. In such cases there may be a need for **discovering subgroups with deviant temporal dynamics** [6, 12]. Examples are male patients for which some symptom takes longer to wane in comparison to female patients, or workflow processes of department A having excessive payment failures in comparison to other departments. This identification is clearly relevant, e.g., to support treatment selection, cost reduction and fraud detection.

As temporal data is often noisy, high-dimensional and has complex statistical dependencies, discovering deviant subprocesses is challenging making many standard statistical and machine learning methods unsuitable. **Exceptional model mining** (EMM) [9, 2, 10] allows for the discovery of *exceptional* (i.e., deviant) models from temporal data, however restricted to a single temporal observation modeled as a Markov chain (MC) [12]. The MC representation imposes severe limitations for temporal settings, as correlations among multiple observations are invisible as they are collapsed into a single observation. Moreover, scaling to larger problems with MCs is infeasible due to the required number of parameters. On the other hand, temporal submodels with latent variables have been investigated [16], yet interpreting latent states is often not trivial.

One distinguishing feature of EMM is that it supports *interpreting model differences*, explaining *why* an object belongs to a subgroup. The challenge now is: how to represent exceptional temporal subprocesses in EMM with reasonable generality, and yet in an interpretable way? In this paper, we introduce the task of **temporal exceptional model mining** (**TEMM**) for the discovery of exceptional temporal subprocesses. Our definition of TEMM enables the representation of a range of temporal subprocesses. We demonstrate TEMM by means of dynamic Bayesian networks (DBNs) [8] to represent temporal submodels. DBNs are graphical models that fulfill several properties: they can capture arbitrary probability distributions, and are interpretable.

The contributions of this paper are as follows. First, TEMM is presented as a setting for representing exceptional temporal subprocesses in EMM. Then, a distance function that measures the exceptionality of a DBN is introduced. We give a procedure for searching for exceptional DBNs in data that is optimized by exploiting properties of the designed distance. An empirical evaluation demonstrates the proposed method, by a broad comparison with baselines on simulated data and a case based on real workflow process data.

This paper is organized as follows. A running example is described in Section 2. In Section 3, we define the task of TEMM. In Section 4, we introduce a distance measure and a search approach for exceptional DBNs. The experiments based on simulations and real data are discussed in Sections 5 and 6. In Section 7, the related work is reviewed. The conclusions are discussed in Section 8.

2 Motivating example: the business process intelligence challenge

In the European Union farmers can apply for direct payments, which provide basic income decoupled from production. A *funding application* is described by **Land Area** and **Number Parcels**, is submitted in a **Year** and is handled by a **Department**. The workflow of an application is a set of documents (**Doc Type**), each one having a state (**Subprocess**) that allows for certain actions (**Activity**). For each document, there are one or more subprocesses. This is the basis for the *business process intelligence challenge* (BPIC18) [4].

Typically, the workflow starts with the *payment application* document, with activities such as mail exchange and validation. An application normally requests subsidies for a number of parcels, stored in a *(geo) parcel document*. Checks regarding the validity of parcels are stored in a *department control parcels* document. The stated parcels are also aligned based on a known reference, and this is kept in a *reference alignment*. The result of these and other checks are summarized in the *control summary*. In any document, *editing* and *calculations* are frequent activities. Eventually, a *decision* is made for the case, leading to *payment* activities. Deviations can occur, e.g., a percentage of cases has an *inspection* document with *on-site* or *remote* subprocesses, or the case might also be reopened due to a legal objection. Figure 1 shows this workflow dynamics. Our general goal is to identify the overall dynamics and whether there are subgroups of the data whose dynamics is substantially different from the general one.

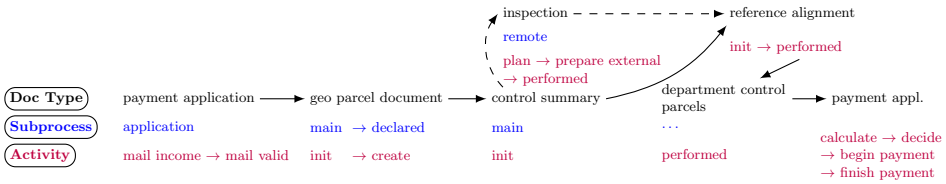


Fig. 1. Typical workflow of the funding example (simplified). Each document occurs with multiple subprocesses and activities. Dashed arrows show process deviations.

3 Temporal exceptional model mining

In this section we describe relevant background notions and define the task of temporal exceptional model mining.

3.1 Temporal targets

In order to represent subgroups we define descriptor and target variables. The set of descriptor variables is a set $\mathbf{A}$ of random variables $\{A_1, \ldots, A_k\}$, where A_i is a *descriptor variable* and has a domain $\mathrm{dom}(A_i)$. We denote values of the domain by lower-case letters such as $a_i \in \mathrm{dom}(A_i)$. In standard SD, one models next to $\mathbf{A}$ a single variable X called *target variable*, while in EMM a *set of target variables* $\mathbf{X} = \{X_1, \ldots, X_n\}$ is used instead. For example, in EMM for regression [10], the predictor and response variables are the target variables. In TEMM, the target variables $\mathbf{X}$ are the result of a temporal process as defined next.

Definition 1 (Temporal targets). *Let $\mathbf{X}$ be a set of random variables. We assume that there is a process that changes $\mathbf{X}$ at regular time points, resulting in the variables $\mathbf{X}^{(0)}, \mathbf{X}^{(1)}, \ldots$. The variable $X_i^{(t)}$ denotes X_i at time t, and we*

denote by $X_i^{(t_1:t_2)}$ *the variables* X_i *occurring from time* t_1 *up to* t_2*. The variables* $X_i^{(t)}$*, for* $t \geq 0$*, have the same domain. We call each* $\mathbf{X}^{(t)}$ *a* temporal target*.*

Based on Definition 1, we define the space of variables in TEMM as $\{\mathbf{A}, \mathbf{X}^{(0)}, \mathbf{X}^{(1)}, \ldots\}$. In practice, a data point in TEMM corresponds to configurations of $\mathbf{A}$ and a finite number of temporal targets. Based on this, we consider a multiset D of data points (called dataset in the following), where the ith data point is denoted by $(\mathbf{a}[i], \mathbf{x}[i]^{(0)}, \ldots, \mathbf{x}[i]^{(m_i)})$, in which m_i is its last temporal target.

Example 1. Reconsider the problem of Section 2 with descriptors $\mathbf{A} = \{$**Year**, **Department**, **Number Parcels**, **Land Area**$\}$ and targets $\mathbf{X} = \{$**Activity**, **Doc Type**, **Subprocess**$\}$. Figure 2 shows an example of a data point.

time

Year = 2016, **Department** = e7, **Number Parcels** = 37, **Area** = 97.85

Doc Type	**Subprocess**	**Activity**
payment application	application	mail income → mail valid
geo parcel document	main	initialize
geo parcel document	declared	create
control summary	main	initialize
reference alignment	main	initialize → performed
department control parcels	main	performed
payment application	application	initialize → calculate → decide → revoke decision → calculate → decide → begin payment → insert document → finish payment

Fig. 2. A data point of the funding process. The temporal targets are {Doc type, Subprocess, Activity}. Arrows indicate transitions between instances of temporal targets. All the activities of a row are associated with the same Doc Type and Subprocess.

3.2 Subgroups

A subgroup can be described by different pattern languages, depending on the data being explored and on the patterns one wishes to discover [5]. Although other languages exist (see, e.g., [2, 13]), the attribute-value pattern language is still very relevant in EMM [14, 6]. In this work, we use this propositional language, which is defined based on the space of descriptor variables $\mathbf{A}$ as follows.

Definition 2 (Subgroup). *Let* $D = \{d_1, \ldots, d_m\}$ *be a dataset with each records* $d_i = (\mathbf{a}[i], \mathbf{x}[i]^{(0)}, \ldots, \mathbf{x}[i]^{(m_i)})$*. Let* φ *denote an expression of the form* $(A_{p_1} = a_{p_1} \wedge \cdots \wedge A_{p_q} = a_{p_q})$*, where* $\{p_1, \ldots, p_q\} \subseteq \{1, \ldots, k\}$*. The* subgroup *associated with* φ *is defined as:*

$$G_\varphi = \left\{ d_i \in D \mid (A_{p_1}[i] = a_{p_1} \wedge \ldots \wedge A_{p_q}[i] = a_{p_q}) \right\} \tag{1}$$

We say that the number of descriptors *of* G_φ *is equal to* q*.*

We refer to a subgroup either by G_φ, by the expression φ that defines it, or simply by G if no confusion arises. For convenience, the domain of a binary descriptor such as A is denoted by $\text{dom}(A) = \{a^-, a^+\}$. For example, an expression $(a_1^+ \wedge a_2^+ \wedge a_3^-)$ represents a subgroup with 3 binary descriptors. In Definition 2, a subgroup is a subset of data points of D selected according to a propositional expression formed by a conjunction of attribute-value pairs. If $q = 1$ we say that the subgroup is *unitary*, otherwise the subgroup is *specialized*.

Definition 3 (Subgroup sequences). *The* subgroup sequences *of a subgroup G_φ of D are given by:*

$$S(G_\varphi) = \{\mathbf{x}[i]^{(0:m_i)} \mid d_i \in G_\varphi\} \tag{2}$$

The size *of subgroup G_φ is $\sum_{d_i \in G_\varphi} (m_i + 1)$ and is denoted by $|G_\varphi|$.*

In TEMM, given a subgroup G a model shall be fitted on the subgroup's sequences $S(G)$ and is called the *subgroup model*. When we wish to compare subgroups in TEMM, we shall compare the subgroup models associated with these subgroups, hence this comparison is based on the space of temporal targets.

3.3 Problem statement

In TEMM, we wish to find all the subgroups G whose models have a distribution that differs from the distribution of the subgroup model associated with the rest of the data. Additionally, every subgroup G must have a minimal size, i.e. $|G| \geq \sigma|D|$, where $\sigma \in [0, 1]$ is the *minimal size threshold*. One can also specify a preference for more specialized or more general subgroups (see, e.g., [12]).

4 Exceptional dynamic Bayesian networks

In this work, dynamic Bayesian networks (DBNs) are studied as model class to represent *temporal* subgroup models. Then, we define a distance notion for DBNs, allowing for the discovery of *exceptional dynamic Bayesian networks*.

4.1 Dynamic Bayesian networks

Dynamic Bayesian networks extend Bayesian networks (BNs) to model processes with uncertainty [8]: the temporal targets of Definition 1. In order to keep the model compact, a few assumptions are adopted in DBNs. We say that a dynamic system over the temporal targets $\mathbf{X}$ is **Markovian** if $P(\mathbf{X}^{(t+1)} \mid \mathbf{X}^{(0:t)}) = P(\mathbf{X}^{(t+1)} \mid \mathbf{X}^{(t)})$, for all $t \geq 0$. This means that predicting the future state depends only on the current state. Another useful assumption is **time homogeneity**, which holds in a dynamic system if the transitions $P(\mathbf{X}^{(t+1)} \mid \mathbf{X}^{(t)})$ are invariant for every $t \geq 0$.

Definition 4 (Dynamic Bayesian network). *A* **dynamic Bayesian network** *M is a Markovian time-homogeneous system $M = (\mathcal{B}_0, \mathcal{B}_\rightarrow)$, where: (i) $\mathcal{B}_0 = (\mathcal{G}_0, P_0)$ is a BN over the variables $\mathbf{X}^{(0)}$ called* **initial network***; (ii) $\mathcal{B}_\rightarrow = (\mathcal{G}_\rightarrow, P_\rightarrow)$ is a BN over the variables $\{\mathbf{X}^{(t+1)}, \mathbf{X}^{(t)}\}$ called* **transition network***. The variables of $\mathbf{X}^{(t)}$ have no parents in the transition network.*

Based on the previous notions, a DBN can be unrolled for any discrete horizon $\{0, \ldots, m\}$ with the following joint distribution:

$$P(\mathbf{X}^{(0:m)}) = \prod_{i=1}^{n} P_0(X_i^{(0)} \mid \pi(X_i^{(0)})) \prod_{t=0}^{m-1} \prod_{i=1}^{n} P_\rightarrow(X_i^{(t+1)} \mid \pi(X_i^{(t+1)})) \tag{3}$$

where $\pi(X_i^{(t)})$ denotes the parents of node $X_i^{(t)}$ in $\mathcal{G}_0$ or $\mathcal{G}_\rightarrow$.

4.2 Distance function

Definition 5 (Mismatch score). *Let D be a dataset over $\{\mathbf{A}, \mathbf{X}^{(0)}, \mathbf{X}^{(1)}, \ldots\}$ and G, H be two subgroups of D. Further, let us denote by M_G and M_H the dynamic Bayesian networks with maximum score given subgroups G and H respectively. The* **mismatch score** *between M_G and M_H is:*

$$\begin{aligned} \mathit{mismatch}(M_G, M_H) = (\mathit{score}(M_G \colon G) - \mathit{score}(M_H \colon G)) \\ + (\mathit{score}(M_H \colon H) - \mathit{score}(M_G \colon H)) \end{aligned} \tag{4}$$

where $\text{score}(M \colon D)$ refers to the score of model M based on subgroup D. Note that given a subgroup G, it holds by definition that $\text{score}(M_G \colon G) \geq \text{score}(M_H \colon G)$ for any model M_H. In practice, it might be difficult to identify the model M_G of subgroup G, as we discuss in Section 4.3.

The mismatch score assess the *error* that a model makes when given data different than that which gives the maximum score. Intuitively, if the DBNs of subgroups G and H are similar one would expect a small mismatch, while a high mismatch indicates the models to be highly different.

Proposition 1 (Weak identity of indiscernibles). *Let M_G be the DBN of subgroup G of dataset D. Then it holds that:*

$$\mathit{mismatch}(M_G, M_G) = 0 \tag{5}$$

Note that a mismatch equal to zero does not imply that the subgroups G and H are the same. This is because a dataset D is a multiset, hence G and H might be associated with the same sequences while being two different parts of D.

Proposition 2 (Symmetry). *Given the DBNs M_G and M_H of the subgroups G and H of dataset D, it holds that:*

$$\mathit{mismatch}(M_G, M_H) = \mathit{mismatch}(M_H, M_G) \tag{6}$$

The proofs of Propositions 1 and 2 follow directly from Definition 5.

Proposition 3 (Non-negativity). *Let M_G and M_H be the DBNs of the subgroups G and H of dataset D. Then it holds that:*

$$mismatch(M_G, M_H) \geq 0 \tag{7}$$

Proof. From the assumptions of Definition 5, M_G has the maximum score given G, i.e., $\text{score}(M_G : G) \geq \text{score}(M_H : G)$ for any model M_H. Analogously, it holds that $\text{score}(M_H : H) \geq \text{score}(M_G : H)$ for any M_G, which completes the proof.

In the next sections, these properties will appear useful for developing a search strategy for identifying exceptional DBNs.

4.3 Scoring function

In practice, DBNs can be learned by maximizing a penalized scoring function. In this work, we use the Bayesian information criterion (BIC) [8] as scoring function. The BIC of a model M_G given data G is defined as follows:

$$\text{BIC}(M_G : G) = 2 \log \mathcal{L}(M_G : G) - |M_G| \log |G| \tag{8}$$

where $\log \mathcal{L}(M_G : G)$ denotes the log-likelihood of the model M_G, $|M_G|$ the number of parameters of M_G, and $|G|$ is the size of G. We assume that M_G is fitted by maximizing the BIC score on data G. We denote by $\text{BIC}(M_G : H)$, with $H \neq G$, the score of M_G given data H different from data G that was used to fit M_G. The BIC score is the score term in Definition 5.

DBN learning is a hard computational problem. In practice, heuristic search is often used. We refer the reader for further detail on DBN learning [8].

4.4 Exceptional subgroups

We define next a general notion of exceptional DBNs.

Definition 6 (Exceptional subgroups). *Given a dataset D, we define a relation $ex \subseteq 2^D \times 2^D$, called* exceptionality*. We say that G is an* exceptional subgroup *with regard to a subgroup H, denoted by $ex(G, H)$, if the distribution of the DBN M_G is different from the distribution of the DBN M_H.*

It is straightforward to verify that the exceptionality relation just defined is symmetric and anti-reflexive. In EMM, the reference subgroup used for determining the exceptionality of a subgroup is typically the full data D, also referred to as *population* [16]. This means that a subgroup of interest G would be compared with D; however, this comparison is made more convenient by instead comparing G with its complement $\bar{G}$ [5], which results in a comparison involving two disjoint subgroups. This approach will be used in TEMM as well.

4.5 Distribution of false discoveries

In practice, one way to use Definition 6 for identifying exceptionality is to consider the extent to which subgroup models differ from the population model. In this case, we would like to identify models which are significantly different from the population model. This is because the true distribution of subgroups is unknown, and we therefore need to account for the error in the estimated model.

To determine how exceptional a subgroup G is, a sampling-based approach with the *distribution of false discoveries* (DFD) [7, 12] is used. Suppose G has size $|G|$, then random subgroups of size $|G|$ are drawn without replacement from D. The mismatch distance of a random subgroup is computed by fitting a DBN on its data and another DBN on the subgroup's complement data. This procedure approximates the distribution of mismatch distances of subgroups with size $|G|$.

By constructing a distribution of distances of random subgroups, we are able to assess how unusual the mismatch distance of a subgroup G is. In order to do so, we execute a hypothesis testing procedure as follows. By taking large enough number of sampled subgroups, the resulting distribution of random mismatch distances will be approximately Normal (see, e.g., [12, 7]). We can then compute a z-score for the mismatch of G, and then a p-value. If the p-value of G is smaller than a significance level α, we conclude that G is an exceptional subgroup.

4.6 Subgroup search

We introduce a bottom-up search method in Algorithm 1 to identify exceptional subgroups from a dataset D. The central idea of Algorithm 1 is to specialize all exceptional subgroups that have been found so far, until there are no exceptional subgroups to be specialized. Each generated subgroup is predicted as exceptional or non-exceptional using Algorithm 2 (Line 9). The algorithm does not specialize subgroups predicted as non-exceptional. For brevity sake, Line 8 generates several subgroups, one for each value of the new descriptor.

Algorithm 1 Subgroup search

Input: D: a dataset $\{\mathbf{A}, \mathbf{X}^{(0)}, \mathbf{X}^{(1)}, \ldots\}$; σ: minimal size threshold; α: significance level.
Output: E: set of subgroups predicted as exceptional.

1: $E \leftarrow \emptyset$
2: $F \leftarrow \emptyset$ // Exceptional subgroups to further expand
3: $C \leftarrow \emptyset$ // Current subgroup
4: cand_descs $\leftarrow \{A_1, \ldots, A_k\}$
5: **do**
6: $\quad E' \leftarrow \emptyset$
7: $\quad$ **for all** $A_i \in$ get_cand_descriptors(c) **do**
8: $\quad\quad G \leftarrow C \cup \{A_i = a_i\}$, for each $a_i \in \mathrm{dom}(A_i)$ // Specialize current subgroup C
9: $\quad\quad$ **if** check_size(G, D, σ) and exceptionality_test(G, D, α) **then**
10: $\quad\quad\quad E' \leftarrow E' \cup \{G\}$
$\quad$ // Add new exceptionals and select new one for expansion
11: $\quad E \leftarrow E \cup E'$
12: $\quad F \leftarrow F \cup E'$
13: $\quad C \leftarrow$ select_random(F)
14: $\quad F \leftarrow F - \{C\}$
15: **while** $F \neq \emptyset$
16: **return** E

4.7 Exceptionality test

Algorithm 2 predicts the exceptionality of a subgroup using the statistical test of Section 4.5. The test assesses how unusual the mismatch of a subgroup is compared to the distribution of mismatch distances of random subgroups, by constructing a DFD. We sample 100 subgroups in our experiments to build each DFD.

Computing a DFD from the scratch is costly due to multiple DBN learning calls. However, we can avoid this by noting that the DFD is a function of the subgroup size, hence when asking for the DFD of a subgroup G we can *reuse* the previously computed DFD of a subgroup H if $|G| = |H|$, which enables substantial computation savings. Moreover, by Proposition 2 the mismatch distance is symmetric, hence when we look up for a DFD in our table of stored DFDs, we can look up for DFDs associated with size $|G|$ *and* to DFDs associated with size $|D| - |G|$. This yields additional computation savings.

Algorithm 2 Exceptionality test

Input: G: a subgroup; D: a dataset $\{\mathbf{A}, \mathbf{X}^{(0)}, \mathbf{X}^{(1)}, \ldots\}$; α: significance level.
Output: the exceptionality prediction of G.

$M_G \leftarrow$ learn_dbn($S(G)$)
$M_{\bar{G}} \leftarrow$ learn_dbn($S(\bar{G})$)
$d \leftarrow$ mismatch($M_G, M_{\bar{G}}$)
// Distribution of false discoveries
if dfd_exists($|G|$) **then** // By Proposition 2, also search for a DFD with size $|D| - |G|$
    $\mathbf{d}_s \leftarrow$ get_stored_mismatch_distances($|G|$) // Reuse DFD
else // Reuse not possible: compute DFD from scratch
    Sample subgroups from D with size $|G|$ and make $\mathbf{d}_s \leftarrow \emptyset$
    for all sampled subgroup H **do**
        $M_H \leftarrow$ learn_dbn($S(H)$)
        $M_{\bar{H}} \leftarrow$ learn_dbn($S(\bar{H})$)
        $d_H \leftarrow$ mismatch($M_H, M_{\bar{H}}$)
        $\mathbf{d}_s \leftarrow \mathbf{d}_s \cup \{d_H\}$
    store_mismatch_distances($|G|, \mathbf{d}_s$)
Calculate the mean $\overline{x}$ and standard deviation s from the set of distances $\mathbf{d}_s$
$z \leftarrow \frac{d - \overline{x}}{s}$ // z-score of the subgroup
Calculate the p-value corresponding to the z-score.
if p-value $< \alpha$ **then**
    return *true* // Subgroup predicted as exceptional
return *false* // Subgroup predicted as non-exceptional

5 Experiments with simulated data

5.1 Data generating procedure

We consider two *simulation scenarios* for assessing the method[6]. First, the number of temporal targets n in $\mathbf{X} = \{X_1, \ldots, X_n\}$, with X_i binary, is set to $n = 10$

[6] Source code and datasets available at: https://github.com/marcoslbueno/temm

inspired by previous research [12] which used Markov chains with 1,024 states. Second, we consider 100 times more states for a broader evaluation, requiring $n = \log_2 100 \cdot 1024 \simeq 17$ temporal targets. For each scenario, two ground truth DBNs on $\mathbf{X}$ were built, with model structure generated by uniformly sampling directed acyclic graphs and node parameters sampled from Beta distributions. Data sequences were sampled from the DBNs, with duration of 10 time points. The same amount of data was sampled from each DBN.

Next, we include the descriptor variable A_1 such that $A_1 = a_1^-$ for all the sequences from one DBN, and $A_1 = a_1^+$ for all the sequences of the other DBN. We also added 5 binary descriptors $R_1, \ldots, R_5$ to act as noisy variables, such that the value of R_i on each sequence is assigned uniformly at random. Based on this procedure, simulated data for a scenario consists of data points over $\{A_1,\ R_1, \ldots, R_5,\ \mathbf{X}^{(0)}, \ldots, \mathbf{X}^{(9)}\}$, where $m = 9$ (the last time point) and the cardinality of $\mathbf{X}$ is n.

5.2 Evaluation

The ultimate goal of TEMM is to recover the exceptional subgroups. For evaluation purposes, we see this as a classification problem on the *space of descriptors*, such that each subgroup is either a *positive* or a *negative* instance. We assigned *ground truth labels* to unitary subgroups as follows:

- *Positive instances*: subgroups (a_1^+) and (a_1^-), as the sequences of each come from different DBNs, making these subgroups exceptional by definition.
- *Negative instances*: subgroups described by R_i, such as (r_1^+) and (r_1^-) as they contain sequences from both DBNs selected at random.

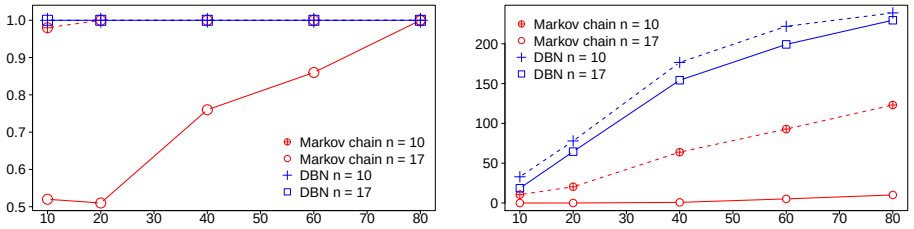
The predicted labels of unitary subgroups by Algorithm 1 are used to evaluate the proposed method. The AUROC (area under the ROC curve) was computed, allowing us to measure how well we can identify exceptional subgroups. We also evaluated the specialized subgroups that Algorithm 1 generates if exceptional unitary subgroups are found. Analogously, positive instances are specialized subgroups that include A_1, and negative instances are all the other specialized subgroups. We evaluate unitary and specialized subgroups separately as the number of specialized ones is typically much larger.

Baseline. Markov chains were used as baseline for representing the temporal targets instead of DBNs. For both MC and DBNs, we applied the mismatch score from Definition 5 to identify subgroups. To avoid zero probabilities, Laplace smoothing with smoothing parameter $\lambda = 1$ is used in both MC and DBN parameter estimation. The whole simulation process was executed 10 times for better assessment, each time with different ground truth models.

5.3 Results

Figure 3a shows the results based on simulated data for unitary subgroups. Note that the X axis shows the number of sequences in each ground-truth subgroup,

hence the total dataset size is twice that amount. The results suggest that the DBN and the MC representation achieved good results with datasets of $n = 10$ target variables (or 1,024 MC states). However, substantial differences arose with $n = 17$ variables (or 131,072 MC states), a situation where DBNs were able to provide optimal AUC values even with the minimal amount of data, as opposed to MCs. In this case, MCs had to count on substantially larger amounts of data in order to provide comparable AUC values to those of DBNs. The threshold $\alpha = 0.05$ was used in Algorithm 2.



(a) Number of sequences (X axis) and mean AUROC (Y axis) on *unitary* subgroups.

(b) Mean number of exceptional *specialized* subgroups correctly predicted (Y).

Fig. 3. Results of Markov Chains and DBNs on simulated data (10 simulations).

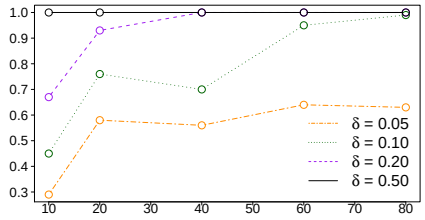
Figure 3b shows the mean number of specialized subgroups which include A_1 and were labeled as exceptional. As the amount of data increases, the results show that more subgroups were produced by both the MC and DBN representations. However, it is clear that DBNs were able to capture substantially more specialized exceptional subgroups.

Figure 4b shows a fragment of subgroups from a simulation iteration using DBNs, together with their mismatch distances. This shows that the method is robust at identifying exceptional subgroups even when most of other subgroups are noisy subgroups. Moreover, the mismatch distances of exceptional subgroups are usually very different from those of non-exceptional subgroups.

5.4 Impact of (dis)similar models on prediction

Now we consider simulations where we control the similarity of the ground truth models. To this end, the second ground truth DBN was defined by copying the structure and parameters of the first DBN. Then, for each variable X_i in the second DBN let $p \leftarrow P(X_i^{(0)} = x_i^- \mid \pi(x_i^{(0)}))$ and $p' \leftarrow P(X_i^{(0)} = x_i^+ \mid \pi(x_i^{(0)}))$. Then, these parameters are changed by picking at random a real number called *change* from the interval $[0, \min(\delta, 1-p)]$, with uniform probability, where $\delta \in [0, 1]$ is the *maximal change threshold*. Next, we set $p \leftarrow p + change$ and $p' \leftarrow p' - change$. The lower the threshold δ, the more similar the DBNs are.

Except for the way ground truth DBNs are generated, we follow the data generating procedure of Section 5.1 and restrict ourselves to learning DBNs and use $n = 17$ temporal targets. Figure 4a shows the AUROC of simulations based on different δ values. The results suggest that extreme cases (low δ, little data) are challenging for the proposed method. In the remainder cases, the method achieved good to optimal results, which suggests that the method is robust at detecting exceptional behavior.



(a) X axis: Number of sequences in dataset; Y axis: AUROC for different values of δ (maximal change threshold).

Subgroup	Size	z-score	p-value	Labels (P&T)	
(a_1^+)	0.50	195.8	$\simeq 0$	1	1
(a_1^-, r_2^-)	0.27	49.4	$\simeq 0$	1	1
(a_1^+, r_1^+, r_2^+)	0.11	15.1	$\simeq 0$	1	1
(r_2^-)	0.49	-1.2	0.22	0	0
(r_3^-)	0.49	0.5	0.64	0	0

(b) A simulation iteration ($n = 17$, 80 data sequences). **Size** = subgroup size normalized by $|D|$, **Labels (P&T)** = predicted and true labels respectively. Label 1 (0) = positive (negative) instance.

Fig. 4. Results of DBNs on simulated data with varying similarity of ground truth.

6 Data of funding applications

In order to evaluate the proposed TEMM method, we consider data from the *business process intelligence challenge* (BPIC18) [4], already briefly described in Section 2. The BPIC18 dataset contains event log data of applications submitted to the European Union for direct payments for German farmers in 2015–2017. *The goal of applying TEMM to the BPIC18 data is to identify subgroups whose dynamics of events is exceptional.*

6.1 Data

Each application in the BPIC18 data is associated with descriptor variables (domain size) as follows: **Land Area** (437), **Department** (4), **Number of Parcels** (74), **Redistribution** (2), **Year of Submission** (3), **Success** (2), **Small Farmer** (2), and **Young Farmer** (2). Applications are also associated with *events* related to workflow activities, where an event is described by the multinomial variables (domain size): **Doc Type** (8), **Subprocess** (8) and **Activity** (33). From the original set of 41 activities, we filtered out some repetitive and generic activities, such as *editing* and *save*.

Each application is associated with one or more events, which are the temporal targets of the data. Hence, the ith data point of this dataset has the

form {Land Area, ..., Young Farmer, Activity$^{(0:m_i)}$, ..., Subprocess$^{(0:m_i)}$}.The BPIC18 dataset has 4,800 applications randomly selected from the original dataset, with an equal number of applications per year. There are 145,980 events in total (mean [StDv] length of each application: 30.4 [8.4] events). Again, Laplace smoothing with $\lambda = 1$ was used in model learning.

6.2 Discovered subgroups

Table 1a shows an excerpt of the exceptional subgroups discovered from the BPIC18 data based on a minimal size $\sigma = 0.05$. The results show that the most exceptional subgroups are unitary and described by a particular year, be it 2015, 2016 or 2017. This suggests that significant changes took place in application dynamics across years manifested in the sequential behavior of the target variables. This could be explained, e.g., by changes in the business process and funding policies. Each department also has its own dynamics, as all unitary subgroups with this descriptor were exceptional. However, their exceptionality was not as strong as that of year subgroups.

Exceptional subgroups	**Size**	**z-score**
Year=2015	0.37	2461.47
Year=2016	0.33	1327.07
Year=2017	0.30	2411.69
Department=4e	0.32	33.28
Department=e7	0.28	35.03
Department=6b	0.25	24.29
Department=d4	0.16	28.00
Number Parcels=2	0.06	12.15
Number Parcels=3	0.06	25.10
Number Parcels=1	0.05	2.15
Year=2015 ∧ Young Farmer=False	0.34	2107.47
Year=2017 ∧ Young Farmer=False	0.27	1844.72
Year=2016 ∧Young Farmer=False	0.30	1144.32
Department=4e ∧Year=2015	0.11	730.81
Department=e7 ∧ Year=2015	0.11	647.71

(a) **Size** = subgroup size normalized by $|D|$.

Doc Type	**2015**	**2016**	**2017**
payment application	16	20.8	12.1
entitlement application	10.5	0.3	0.1
parcel document	2.6	0	0
control summary	1	1	1
reference alignment	2.2	2.1	2
department control parcels	1	1	0
inspection	0.6	1	0.8
geo parcel document	0	3.8	11.3

(b) Average number of document types per application in each year.

Table 1. Results on the BPIC18 dataset, where 38 exeptional subgroups were discovered. For better visualization, only the 5 most exceptional specialized subgroups are shown. All p-values < 0.001, except (Number Parcels=1).

6.3 Comparison to previous analyses

While the ground truth exceptional subgroups are not available for the BPIC dataset, there is evidence that the subgroups described by *year* as shown in Table 1a are exceptional. First, the BPIC18 data provider [4] claims that the underlying process changed between years due to changes implemented in the structure of the application procedure. This is in line with previous research [15] on this dataset, where concept drifts were identified precisely between each year of the data. Other research [19] has analyzed how the workflow of applications submitted in different years has changed, also suggesting that differences exist

in these workflow structures. Based on these previous analyzes, we conclude the proposed method is able to detect true exceptional subgroups.

Differently than the other analyses from the literature on the BPIC18 data, the method proposed in this paper can be seen as a principled one due to its statistical foundations.

6.4 Subgroup differences

Based on subgroup's data, Table 1b shows the frequency of each Doc Type value for the most exceptional subgroups. One strong difference is that the *geo parcel document* vanished in applications from 2015, while it was increasingly used in applications from 2016 and 2017. On the other hand, the *parcel document* was adopted only in 2015, and the *document control parcels* vanished in 2017. All these changes are expected due to known changes in the funding process [4].

Table 1b also reveals a remarkable reduction in the frequency of *entitlement application* over the years. This could reflect that subprocesses such as *objection* and *change* of entitlement application are moved to application payment, as the latter is the only other type of document which has such subprocesses. Other changes include more *inspections* in 2016 and 2017, which might indicate changes in funding policies as only a small percentage of cases are to be inspected.

7 Related work

As a generalization of SD, exceptional model mining [6] is an active area of research and has been applied to different target variable representations. Earlier research includes the discovery of exceptional linear regression models [10] and the discovery of subgroups with Bayesian networks that have significant structural differences [5]. A more specialized usage of EMM is tailored at sequential problems, yet over a single target, where discrete Markov chains with significantly different transition patterns have been investigated [12].

The aforementioned EMM research can be seen as *parameter-based approaches*, because subgroups are characterized based on the unusualness of model parameters, such as regression slope and network structure. On the other hand, model-based subgroup discovery [16] is an *evaluation-driven approach* that compares the distribution of subgroups by means of proper scoring rules. The latter is related to data mining research where the minimum description length (MDL) was applied to identify differences between databases [18]. In this paper we consider more general model selection criteria, where MDL is a special case.

Some body of research has dealt with *subgroup search*, whose aims include making the search more efficient and reducing the number of redundant subgroups. Research has been done on providing bounds for some interestingness scores in the context of numerical targets that can be used for search pruning [11]. Subgroup search has also been formulated in terms of game theory [3], which allows for guiding the search toward the interestingness of subgroups while improving the lack of diversity that search might face.

Other extensions to SD and EMM operate on data other than the common attribute-value data. The approach in [13] is tailored for relational data and can extract very general structured patterns of subgroups. More recently, exceptional graph mining [2] has been proposed to allow for the discovery of graph neighborhoods that are similar internally but exceptional to the general attributed graph (i.e. graphs with non-trivial vertices such as a list of attribute-value pairs). Recently, EMM has been applied to finding subsets of data related to exceptional convolutional layers in convolutional neural networks [17], which might help the interpretation of such models.

The proposed mismatch score can be seen as a *data-based* score, as it is computed based on goodness-of-fit scores (the BIC score). By opposition, previous research [12] for discovering exceptional MCs used a measure based on statistical distance between transition distributions. While structure learning is not required for MC learning, the number of parameters in DBNs is typically substantially lower due to its factorized representation. As experiments have shown, this parameter issue makes the MC representation to scale poorly, particularly when the number of temporal targets n is larger and there is a less data for model learning. Furthermore, the DBN-based search made substantially less mistakes in the simulations, which makes this representation suitable for TEMM.

One task that has some resemblance to TEMM is sequential pattern mining [1]. However, the mined rules might not correspond to actual subgroups or even actual processes from the dataset, as opposed to TEMM and subgroup discovery. This makes it not possible to directly compare the results of these approaches.

8 Conclusions

In this paper, we proposed temporal exceptional model mining to enable the representation of temporal observations in EMM in a principled way. For capturing the temporal dependencies in TEMM, dynamic Bayesian networks were used, which allows for an intuitive and interpretable model class for TEMM.

The proposed method was empirically evaluated on simulated data and process data based on funding applications, showing that the identifiability of the method in different scenarios is robust. Our method was able to discover exceptional subgroups from the funding data in accordance to previous research, as well other, yet less exceptional subgroups. Furthermore, our approach solved this practical problem in a more principled manner.

As future work, we would like to explain in more detail why models are considered as exceptional. This could involve looking at relevant structural or numerical parameters of the DBNs. We wish to quantify the savings of the optimizations employed during search to reduce the computation of distribution of false discoveries. Finally, we would like to investigate if further improvements to the search algorithm are possible based on properties of the mismatch distance.

References

1. van der Aalst, W.: Process Mining: Discovery, Conformance and Enhancement of Business Processes. Springer Publishing Company, Incorporated (2011)
2. Bendimerad, A., Plantevit, M., Robardet, C.: Mining exceptional closed patterns in attributed graphs. Knowledge and Information Systems **56**(1), 1–25 (Jul 2018)
3. Bosc, G., Boulicaut, J.F., Raïssi, C., Kaytoue, M.: Anytime discovery of a diverse set of patterns with monte carlo tree search. Data Mining and Knowledge Discovery **32**(3), 604–650 (May 2018)
4. van Dongen, B., Borchert, F.: BPI Challenge 2018 (2018), https://data.4tu.nl/repository/uuid:3301445f-95e8-4ff0-98a4-901f1f204972
5. Duivesteijn, W., Knobbe, A., Feelders, A., van Leeuwen, M.: Subgroup Discovery Meets Bayesian Networks – An Exceptional Model Mining Approach. In: 2010 IEEE International Conference on Data Mining. pp. 158–167 (Dec 2010)
6. Duivesteijn, W., Feelders, A.J., Knobbe, A.: Exceptional model mining. Data Min. Knowl. Discov. **30**(1), 47–98 (Jan 2016)
7. Duivesteijn, W., Knobbe, A.: Exploiting False Discoveries – Statistical Validation of Patterns and Quality Measures in Subgroup Discovery. In: Proc. IEEE 11th International Conference on Data Mining. pp. 151–160. ICDM '11 (2011)
8. Friedman, N., Murphy, K., Russell, S.: Learning the structure of dynamic probabilistic networks. In: Proc. of the 14th Conference on Uncertainty in Artificial Intelligence. pp. 139–147. UAI'98 (1998)
9. Herrera, F., Carmona, C.J., González, P., del Jesus, M.J.: An overview on subgroup discovery: foundations and applications. Knowl Inf Syst **29**(3), 495–525 (Dec 2011)
10. Leman, D., Feelders, A., Knobbe, A.: Exceptional model mining. In: Daelemans, W., Goethals, B., Morik, K. (eds.) Machine Learning and Knowledge Discovery in Databases. pp. 1–16. Springer Berlin Heidelberg, Berlin, Heidelberg (2008)
11. Lemmerich, F., Atzmueller, M., Puppe, F.: Fast exhaustive subgroup discovery with numerical target concepts. Data Min Knowl Disc **30**(3), 711–762 (2016)
12. Lemmerich, F., et al.: Mining subgroups with exceptional transition behavior. In: Proc. 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 965–974. KDD '16 (2016)
13. Lijffijt, J., Spyropoulou, E., Kang, B., De Bie, T.: P-N-RMiner: a generic framework for mining interesting structured relational patterns. International Journal of Data Science and Analytics **1**(1), 61–76 (Apr 2016)
14. Novak, P.K., Lavrač, N., Webb, G.I.: Supervised descriptive rule discovery: A unifying survey of contrast set, emerging pattern and subgroup mining. J. Mach. Learn. Res. **10**, 377–403 (Jun 2009)
15. Pauwels, S., Calders, T.: An Anomaly Detection Technique for Business Processes based on Extended Dynamic Bayesian Networks. In: The 34th ACM/SIGAPP Symposium on Applied Computing (SAC '19). pp. 1–8. Limassol, Cyprus (2019)
16. Song, H.: Model-Based Subgroup Discovery. PhD thesis, University of Bristol (11 2017)
17. van Strien, B.: Exceptional Model Mining of Convolutional Neural Networks. MSc thesis, Eindhoven University of Technology (2019)
18. Vreeken, J., Van Leeuwen, M., Siebes, A.: Characterising the difference. In: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 765–774 (2007)
19. Wangikar, L., Dhuwalia, S., Yadav, A., Dikshit, B., Yadav, D.: Faster Payments to Farmers: Analysis of the Direct Payments Process of EU's Agricultural Guarantee Fund – Business Process Intelligence Challenge 2018 (2018)