Unraveling temporal processes using probabilistic graphical models

de Paula Bueno, M.L.

7

EXCEPTIONAL MODEL MINING USING DYNAMIC BAYESIAN NETWORKS

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. This task has clear relevance nowadays, facing the current need for interpretable AI models. In this chapter, we introduce temporal exceptional model mining *to capture not only multiple targets, but also complex temporal relationships among the targets. Temporal exceptional model mining opens new avenues for discovering groups that deviate from the crowd, in domains such as medical treatments and industrial processes, where repeated measurements of a set of variables might be available. The contributions of this chapter are three-fold: (i) a new definition of the task of temporal exceptional model mining is provided; (ii) we characterize the discovery of exceptional dynamic Bayesian networks by means of a new interestingness score, and (iii) the practical value of the proposed method is demonstrated based on process data of funding applications and by comparisons with previous EMM approaches.*

7.1 INTRODUCTION

Subgroup discovery (SD, for short) is the task of identifying subsets of a dataset that have unusual distributions with respect to a target variable [87]. Subgroup discovery and clustering have different goals [181] as clustering seeks subsets of data that are internally homogeneous, while in SD the models that allow for *interpreting differences* are sought, as they support explaining why an object belongs to a subgroup. Interpretability is essential in artificial intelligence, even with successful, yet less interpretable models as deep neural networks [88, 114], which justifies the relevance of SD research.

In many real-world applications one has to deal with multiple and complex targets. This has led to the generalization of subgroup discovery known as *exceptional model mining* (EMM, for short) [110]. EMM aims to identify subgroups with models fitted on the targets that are unusual compared to a reference model (typically the model fitted on the whole dataset).

The computational burden of SD lies in subgroup search [87], as determining whether a subgroup is unusual is often straightforward. In EMM, however,

models over multiple variables are fitted on subgroup data, which results in a two-fold challenge: (*i*) the choice of suitable model classes, as model learning is now an integral part of the framework, and (*ii*) how to determine whether a subgroup model is exceptional. This increased complexity has been compensated by the discovery of useful exceptional models, e.g., based on linear regression [110], Bayesian networks [57], subgraph mining [9, 100], social networks [4] and preferences [149].

Remarkably, little research has been done in exploiting temporal submodels for EMM. Submodels based on Markov chains (MCs, for short) have been investigated [112], as well as latent variable-modeling by means of hidden Markov models [161]. In this chapter, we introduce the discovery of temporal submodels by means of the *temporal exceptional model mining* task (TEMM, for short), which is demonstrated by means of dynamic Bayesian networks (DBNs, for short) as model class. We argue that using DBNs allows for a general and intuitive representation of subgroups obtained from multiple and temporal observations. The DBN representation allows for extra, qualitative information that can be gleaned from the model structure.

The contributions of this chapter are as follows. First, the novel task of temporal exceptional model mining is defined, which can be seen as a generalization of previous research in EMM. Then we introduce the usage of DBNs for TEMM by proposing an *interestingness score* for identifying exceptional DBNs. Finally, the proposed methods are demonstrated by analyzing data of funding applications.

This chapter is organized as follows. In Section 7.2, we discuss the related work. In Section 7.3, we define the task of TEMM. In Section 7.4, we introduce a distance measure for exceptional DBNs. In Section 7.5 we present a search approach for exceptional DBNs. The experiments based on simulations and real data are discussed in Sections 7.6 and 7.7. The conclusions and future work are discussed in Section 7.8.

7.1.1 *Motivating example*

We describe next a running example which is also used in experiments with real-world data. In the European Union, farmers can apply for direct payments [56], which provide them additional income and incentives for sustainable production. A *funding application* is described by **Land Area**, **Young Farmer** (yes/no), and **Small Farmer** (yes/no). An application is submitted in a **Year** and is checked for eligibility by a **Department**. The work flow of an application is a sequence of *events* described by **Activity** and **Doc Type**.

We would like to know whether there are applications whose work flow (i.e. the dynamics of Activity and Doc Type) deviates considerably from the work flow of the whole population of applications. It could be the case that applications handled by a certain department take much longer than the average, or that applications submitted in a particular year have a specific work flow. By *automatically* discovering these subgroups, we could learn more about the process, which could e.g. help the organization to improve the process quality.

7.2 RELATED WORK

As a generalization of SD, exceptional model mining [58] 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 [110] and the discovery of subgroups with Bayesian networks that have significant structural differences [57]. A more specialized application of EMM is tailored at sequential problems, yet over a single target, where discrete Markov chains with significantly different transition patterns have been investigated [112].

The aforementioned EMM research can be seen as *parameter-based approaches*, because subgroups are characterized based on the unusualness of some of the model parameters, e.g. regression slope, network structure, etc. On the other hand, model-based subgroup discovery [161] is an *evaluation-driven approach* that compares the distribution of subgroups by means of proper scoring rules.

Some body of research has dealt with *subgroup search*, whose aims include making the search more efficient, reducing the number of redundant subgroups, etc. Research has been done on providing bounds for some interestingness scores in the context of numerical targets that can be used for search pruning [111]. Subgroup search has also been formulated in terms of game theory [18], 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 [113] is tailored for relational data and can extract very general structured patterns of subgroups. More recently, exceptional graph mining [9, 100] 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) [9]. Research has been done on the discovery of exceptional social behavior from spatio-temporal [98], which helps understand networked interactions (e.g. as in how people interact in a neighborhood). Recently, EMM has been applied to finding subsets of data related to exceptional convolutional layers in convolutional neural networks [167], which might help the interpretation of such models.

7.3 TEMPORAL EXCEPTIONAL MODEL MINING

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

7.3.1 *Temporal targets*

In order to represent subgroups in SD and EMM 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 each 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 normally models next to $\mathbf{A}$ a single variable X called *target variable*, while in EMM a *set of targets variables* $\mathbf{X} = \{X_1, \ldots, X_n\}$ is used instead. For example, in EMM for regression [110], the predictor and response variables are the target variables.

In TEMM, we assume that the target variables are the result of a temporal process that changes a set of basis variables $\mathbf{X}$.

Definition 7.1 (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 7.1, 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, 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 the last temporal target. Thus, each data point of D has a particular number of temporal targets. An example is given next.

Example 7.1. *Consider the dataset for the application described in Section 7.1.1 with descriptors $\mathbf{A} = \{$**Year**, **Department**, **Number Parcels**, **Land Area**$\}$ and targets $\mathbf{X} = \{$**Activity**, **Doc Type**$\}$. Table 7.1 shows two data points of this dataset.*

7.3.2 *Subgroups*

A subgroup can be described by different pattern languages [57], depending on the type of data and the kind of patterns one wants to discover. In spite of different existing languages (see, e.g., [9, 113]), the attribute-value pattern language [58, 61, 128] is still very relevant in SD and EMM. In this work, we use this propositional language, which is defined based on the space of descriptor variables $\mathbf{A}$ as follows.

Definition 7.2 (Subgroup)**.** *Let $D = \{d_1, \ldots, d_m\}$ be a dataset (multiset) with each record d_i a collection of variable-value pairs $A_j[i] = a_j$ and $\mathbf{A} = \{A_1, \ldots, A_k\}$. 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 to φ is defined as:*

$$G_\varphi = \left\{ d_i \in D \mid (A_{p_1}[i] = a_{p_1} \wedge \ldots \wedge A_{p_q}[i] = a_{p_q}) \right\} \tag{7.1}$$

We say that the number of descriptors of G_φ is q.

We refer to a subgroup defined by the expression φ either by G_φ or by the expression φ itself. For convenience, the domain of a binary descriptor such as A is denoted by $\mathrm{dom}(A) = \{a^-, a^+\}$. For example, an expression $(a_1^+ \wedge a_2^+ \wedge a_3^-)$ represents a subgroup with 3 binary descriptors. In Definition 7.2, a subgroup

Year	Department	# Parcels	Area	Activity	Doc Type
2016	4e	31	97.8	mail valid	Payment application
				initialize	Geo parcel document
				finish editing	Control summary
				performed	Reference alignment
				finish editing	Geo parcel document
				performed	Department control parcels
				finish editing	Geo parcel document
				calculate	Payment application
				decide	Payment application
				revoke decision	Payment application
				calculate	Payment application
				decide	Payment application
				begin payment	Payment application
				abort payment	Payment application
				begin payment	Payment application
				insert document	Payment application
				finish payment	Payment application

Year	Department	# Parcels	Area	Activity	Doc Type
2016	e7	37	97.8	mail valid	Payment application
				initialize	Geo parcel document
				finish editing	Control summary
				performed	Reference alignment
				performed	Department control parcels
				calculate	Payment application
				decide	Payment application
				revoke decision	Payment application
				calculate	Payment application
				decide	Payment application
				begin payment	Payment application
				insert document	Payment application
				finish payment	Payment application

Year	Department	# Parcels	Area	Activity	Doc Type
2017	6b	7	9.1	mail valid	Payment application
				pre-check	Geo parcel document
				finish editing	Control summary
				finish editing	Geo parcel document
				performed	Reference alignment
				initialize	Payment application
				finish editing	Geo parcel document
				calculate	Payment application
				finish editing	Geo parcel document
				calculate	Payment application
				decide	Payment application
				begin payment	Payment application
				insert document	Payment application
				finish payment	Payment application

Table 7.1: Data points of a process dataset, with $\mathbf{A} = \{\textbf{Year}, \textbf{Department}, \textbf{Number Parcels}, \textbf{Land Area}\}$ and $\mathbf{X} = \{\textbf{Activity}, \textbf{Doc Type}\}$. The temporal targets correspond to the work flow of events in the order they occurred.

is a subset of data points of D selected according to a propositional expression formed by a conjunction of attribute-value pairs. Not all attributes of $\mathbf{A}$ need to be involved in the subgroup expression, hence $p_q \leq k$. If $q = 1$ we say that the subgroup is *unitary*, otherwise the subgroup is *specialized*. The subscript φ is omitted from G_φ if no risk of ambiguity arises.

Each data point of G_φ is associated to a configuration of temporal targets for which notation is introduced next.

Definition 7.3 (Subgroup sequences)**.** *The subgroup sequences of a subgroup G_φ of D are given by:*

$$S(G_\varphi) = \{\mathbf{x}[i]^{(0:m_i)} \mid d_i \in G_\varphi\} \tag{7.2}$$

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

7.3.3 *Comparing subgroups*

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

The notion of exceptional subgroups involves comparing subgroups based on some notion of distance. We define a distance notion with some *desirable properties* that serves as a basis for the development of distance measures for specific class of temporal models.

Definition 7.4 (Distance function)**.** *Given a multiset D, the distance function between two subgroups G and H of D is a real number denoted by $d(G, H)$. This distance has the following properties:*

$$d(G, H) \geq 0 \qquad \textit{non-negativity} \tag{7.3}$$

$$d(G, H) = 0 \textit{ if } G = H \qquad \textit{weak identity of indiscernibles} \tag{7.4}$$

$$d(G, H) = d(H, G) \qquad \textit{symmetry} \tag{7.5}$$

Other properties can be added to the above ones depending on the desired characteristics of the distance function. For example, by strengthening the second assumption and adding the triangle inequality, one would arrive at a distance function that would be a *metric*. The distance function should, however, be designed in such a way to support these properties.

7.3.4 *Exceptional subgroups*

One way to determine whether a subgroup G is exceptional is by considering a *reference* subgroup upon which the distance to G can be computed. We introduce

the notion of exceptional relation next, which has a few *desirable properties* of interest.

Definition 7.5 (Exceptional subgroup)**.** *Given a multiset D, we define a relation $ex \subseteq 2^D \times 2^D$, called exceptionality which has the following properties for two any subgroups G and H of D:*

$$ex(G,H) \implies ex(H,G) \quad \textit{(symmetry)} \tag{7.6}$$

$$\neg ex(G,G) \quad \textit{(anti-reflexive)} \tag{7.7}$$

If $ex(G,H)$ holds, we say that G is an exceptional subgroup with regard to the subgroup H.

The precise definition of which subgroups are exceptional depends on the definition of the distance function. An exceptionality relation will be defined in Section 7.4.4.

7.3.5 *Problem statement*

In TEMM, we wish to find all the subgroups G which are exceptional with regard to the population. One additional requirement is that every exceptional 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 some kind of preference for more specialized or more general subgroups (see, e.g., [112]).

7.4 EXCEPTIONAL DYNAMIC BAYESIAN NETWORKS

In this work, we consider dynamic Bayesian networks [68, 104, 124] as model class to represent subgroup models. We define a distance function for DBNs and instantiate it for a scoring function, allowing for the discovery of *exceptional dynamic Bayesian networks*.

7.4.1 *Dynamic Bayesian networks*

Dynamic Bayesian networks extend Bayesian networks for modeling processes with uncertainty. In this work, DBNs model the temporal targets from Definition 7.1.

In order to keep the model compact, a few assumptions are considered in dynamic systems such as 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 the *time homogeneity*, which holds in a dynamic system if the transitions $P(\mathbf{X}^{(t+1)} \mid \mathbf{X}^{(t)})$ are fixed for every $t \geq 0$. We refer the reader to Section 2.4 for more details on DBNs.

7.4.2 *Distance function*

Definition 7.6 (Mismatch score)**.** *Let D be a multiset 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 learned from G and H respectively by maximizing some scoring function. The mismatch score between M_G and M_H is:*

$$\begin{aligned} mismatch(M_G, M_H) = (score(M_G : G) - score(M_H : G)) \\ + (score(M_H : H) - score(M_G : H)) \end{aligned} \tag{7.8}$$

where score$(M : G)$ refers to the score of model M based on data G. The mismatch distance resembles the idea of learning and validation sets (e.g. as used in cross-validation [104])). However, here we are considering a more general situation, because we assume that G and H might not have come from the same distribution. In fact that is what we want to evaluate: the *error* that a model makes when given data not used to learn it. Intuitively, if the DBNs induced from G and H are similar one would expect a small mismatch value, while a high mismatch would be obtained had the models been too different. A few properties regarding the mismatch score are given next.

Proposition 7.1 (Weak identity of indiscernibles)**.** *Let M_G be a DBN fitted to the subgroup G of D. Then it holds that:*

$$mismatch(M_G, M_G) = 0 \tag{7.9}$$

Proof. Directly from the definition of mismatch score. □

Proposition 7.1 means that the weak identity of indiscernibles holds for the mismatch. However, it is not the case that a mismatch equal to zero implies that the subgroups G and H are the same. This is because D is a multiset, hence G and H might be associated to the same sequences while being two different subsets of D. Another relevant property is symmetry, which is formalised in the next proposition.

Proposition 7.2 (Symmetry)**.** *Given two DBNs M_G and M_H learned from two subgroups G and H of D, it holds that:*

$$mismatch(M_G, M_H) = mismatch(M_H, M_G) \tag{7.10}$$

Proof. Directly from the definition of mismatch score. □

A relevant property concerns the sign of the mismatch distance is given as follows.

Proposition 7.3 (Non-negativity)**.** *Let M_G and M_H be the DBNs learned from the subgroups G and H of D. Then it holds that:*

$$mismatch(M_G, M_H) \geq 0 \tag{7.11}$$

Proof. If $G = H$, the claim holds by Proposition 7.1. Otherwise, if M_G is the model learned from G, then it must hold that $\text{score}(M_G : G) \geq \text{score}(M_H : G)$ for any model M_H. This is because by Definition 7.6 M_G was learned by maximizing the score given the data G, then no other model can have better score given G. □

As the mismatch distance is non-negative, symmetric and has the weak identity of indiscernibles property, it follows that it can be taken as a distance function for TEMM, as discussed in Section 7.3.3.

7.4.3 *Scoring function*

In this work, we use Bayesian information criterion as scoring function (see Section 2.3.2), which is proportional to the log-likelihood of the model and includes a penalty to control for model complexity. For convenience, we repeat the definition of the BIC of a model M_G given data G as follows:

$$\text{BIC}(M_G : G) = 2 \log \mathcal{L}(M_G : G) - |M_G| \log |G| \tag{7.12}$$

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 number of observations of G. The negative value of the standard BIC was taken for the convenience of maximizing the score.

We assume that M_G is fitted by maximizing the BIC score as denoted by $\text{BIC}(M_G : G)$, and we shall denote by $\text{BIC}(M_G : H)$, with $H \neq G$, the score of M_G given data H different than that used to fit M_G. The BIC score corresponds to the score term of Definition 7.6.

7.4.4 *Exceptional subgroups*

We define next a general notion of exceptional DBNs.

Definition 7.7 (Exceptional subgroups)**.** *Consider the exceptionality relation $ex \subseteq 2^D \times 2^D$. 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.*

Definition 7.7 implements the idea of exceptional subgroups delineated by Definition 7.5 applied to exceptional DBNs. It is straightforward to verify that the exceptionality relation just defined is symmetric and anti-reflexive, hence the relationship has the desired properties as discussed in Section 7.3.4.

In EMM, the reference subgroup used for determining the exceptionality of a subgroup is typically the full data D, also referred to as *population* [161]. 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 denoted by $\bar{G}$ [57], which results in a comparison involving two disjoint subgroups. TEMM uses the population as reference subgroup as well,

thus for determining whether a subgroup G is exceptional we compare the subgroup models of G and $\bar{G}$.

7.5 IDENTIFYING EXCEPTIONAL SUBGROUPS

In this section, we discuss how the exceptionality of DBNs can be identified from data by considering reasonable assumptions on what can be seen as exceptional in real-world situations.

7.5.1 *Distribution of false discoveries*

In practice, one way to use Definition 7.7 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. The reason for shifting the focus to significantly different subgroups is that the true distribution of subgroups is unknown, and we therefore need to account for the error in the estimated model. Based on these ideas, the identification of exceptional subgroups is described next.

To determine how exceptional a subgroup G is, a sampling-based approach with the *distribution of false discoveries* (DFD, for short) [59, 112] is used. Suppose G has size $|G|$, then random subgroups of size $|G|$ are drawn without replacement from D, such that for each random subgroup its mismatch distance is computed. In order to compute the mismatch of each random subgroup, we fit a DBN on the random subgroup data and another DBN on its complement data. This sampling procedure approximates the distribution of mismatch distances that characterizes the mismatch 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., [59, 112]). We can then compute a z-score for the mismatch of G, from which we can obtain a p-value. If the p-value of G is smaller than a significance level α, we conclude that G is an exceptional subgroup.

7.5.2 *Subgroup search*

In order to generate subgroups and test their exceptionality, we introduce a general search algorithm outlined in Algorithm 3. The central idea of Algorithm 3 is to specialize all exceptional subgroups that have been found so far, until there are no further exceptional subgroups to be specialized. The algorithm does not specialize subgroups considered as non-exceptional.

Algorithm 3 starts with $c = \emptyset$ as the current subgroup, i.e. the total population. By entering the outer loop, new candidate subgroups are generated by

specializing c with the addition of one descriptor that is not in the descriptor set of c (Line 8). For brevity sake, Line 8 in fact generates several subgroups, one for each value from the domain of the new descriptor. Then, each new candidate subgroup is tested for a minimal size σ and for exceptionality. If the candidate subgroup passes these tests, it is stored into the set E', which keeps the exceptional subgroups found so far. The new exceptional subgroup is also added to F, which stores the subgroups to further expand. Once the new exceptional subgroups have been processed, a subgroup to be further specialized is picked at random from F. While $F \neq \emptyset$, the whole specialization process is repeated.

Algorithm 3 Subgroup search

Input: D: a dataset of data points of the form $\{\mathbf{A}, \mathbf{X}^{(0)}, \mathbf{X}^{(1)}, \ldots\}$; σ: minimal size threshold; α: significance level for exceptionality test.
Output: E: set of exceptional subgroups.

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

7.5.3 *Exceptionality test*

Algorithm 3 makes use of an exceptionality test, which is detailed in Algorithm 4. Algorithm 4 does intensive computation as it learns subgroup models, calculates their mismatch distances, and calculates the DFDs. These steps are necessary to assess how unusual the mismatch of a particular subgroup is compared to random subgroups.

Algorithm 4 Exceptionality test

Input: G: a subgroup; D: a dataset of data points of the form $\{\mathbf{A}, \mathbf{X}^{(0)}, \mathbf{X}^{(1)}, \ldots\}$; α: significance level for exceptionality test.
Output: a Boolean value indicating whether G is exceptional.

$M_G \leftarrow \text{learn\_dbn}(S(G))$
$M_{\bar{G}} \leftarrow \text{learn\_dbn}(S(\bar{G}))$
$d \leftarrow \text{mismatch}(M_G, M_{\bar{G}})$
// Distribution of false discoveries
Sample subgroups from D with size $|G|$.
for all sampled subgroup H **do**
    $M_H \leftarrow \text{learn\_dbn}(S(H))$
    $M_{\bar{H}} \leftarrow \text{learn\_dbn}(S(\bar{H}))$
    $d_H \leftarrow \text{mismatch}(M_H, M_{\bar{H}})$
Calculate the mean and standard deviation from the set of distances d_H, and denote them by $\overline{x}$ and s respectively.
$z \leftarrow \dfrac{d - \overline{x}}{s}$ // z-score of the subgroup
Calculate the p-value corresponding to the z-score.
if p-value $< \alpha$ **then**
    return *true*
return *false*

7.5.4 *Search optimization*

The computation of DFDs is a costly step of the exceptionality test used by Algorithm 3. In order to evaluate the exceptionality of a subgroup G, we check whether a subgroup H with $|H| = |G|$ has been considered before during search. If so, we can reuse the previously computed DFD of H as the DFD of G, because the DFD is a function of the subgroup size. This can save substantial computation because in problems with several descriptor variables (the set $\mathbf{A}$), one would expect that some subgroups have the same size. We can take advantage of this fact by storing a list of sizes and a DFD for each size, so that a DFD is actually computed only when it is not found in this list.

By Proposition 7.2, the mismatch distance is symmetric. This means that if we ask whether a subgroup G with size $|G|$ is exceptional, we could equivalently ask whether the complementary subgroup (which has size $|D| - |G|$) is exceptional. This means that when we look up for a DFD in our table of stored DFDs, we can look up for DFDs associated to size $|G|$ *and* to DFDs associated to size $|D| - |G|$. This yields additional computational savings.

7.6 EXPERIMENTS WITH SIMULATED DATA

7.6.1 *Data*

We consider two simulation scenarios to assess the method by varying the set $\mathbf{X} = \{X_1, \ldots, X_n\}$, with X_i binary. In the first scenario, we use $n = 10$ variables inspired by previous research [112] which used Markov chains with 1,024 states. In the second scenario, we consider 100 times more MC states, requiring $n = \log_2 100 \cdot 1024 \simeq 17$ variables, allowing for a more comprehensive evaluation.

In order to build a dataset for a scenario, simulated data was generated from two ground truth DBNs based on the variables $\mathbf{X}$. The number of time points was 10 for both $n = 10$ and $n = 17$. The structure of each DBN was generated by uniformly sampling DAGs [122], while node parameters are sampled from Beta distributions.

The next step is to define the descriptor space. We defined a descriptor variable A_1 such that the sequences from one DBN were assigned to the subgroup $(A_1 = a_1^-)$ and the sequences from the other DBN to $(A_1 = a_1^+)$. The same amount of data was generated for these subgroups. We also added 5 binary descriptors $R_1, \ldots, R_5$ to act as noisy variables by randomly assigning the generated sequences to the noisy variables (with uniform probability).

Given a scenario, we now assign *ground truth labels* to unitary subgroups as follows:

- The subgroups (a_1^+) and (a_1^-) are seen as *positive instances*, as the sequences of each come from a single DBN, thus making these subgroups exceptional by definition.
- The subgroups described by R_i, such as $R_1 = r_1^+$ and $R_2 = r_2^-$, are seen as *negative instances*, as they correspond to random selections of sequences.

Based on the true and predicted labels, we measure how well we can identify exceptional subgroups (described by A_1) and non-exceptional subgroups (described by R_i). Further, by having only one descriptor for exceptional subgroups (A_1) and multiple ones for non-exceptional subgroups (R_i), it becomes more challenging to distinguish the two types of subgroups. This way we evaluate the robustness of the proposed algorithm.

Based on the described procedure, simulated data for a scenario consists of data points over the variables $\{A_1, R_1, \ldots, R_5, \mathbf{X}^{(0)}, \ldots, \mathbf{X}^{(9)}\}$. The whole simulation process, including the generation of ground truth models, was executed 10 times for better assessment of each scenario.

7.6.2 *Evaluation*

Algorithm 3 always generates unitary subgroups, which allows for evaluating the labeling done by the proposed method using several metrics. The AUC-ROC

	MC $n = 10$		DBN $n = 10$		MC $n = 17$		DBN $n = 17$	
Seq	**Pr**	**Rec**	**Pr**	**Rec**	**Pr**	**Rec**	**Pr**	**Rec**
10	0.6	0.6	0.82	1	0	0	0.9	1
20	0.87	1	0.95	1	0	0	0.9	1
40	0.85	1	0.87	1	0.15	0.2	0.9	1
60	0.92	1	0.87	1	0.53	0.6	0.89	1
80	0.92	1	0.83	1	0.75	0.85	0.88	1

Table 7.2: Precision (**Pr**) and recall (**Rec**) achieved by Markov chains and DBNs on simulated data. **Seq** = number of data sequences.

(area under the ROC curve) evaluates how the method separates the positive from the negative instances. We also compute *precision* and *recall* values, where precision is TP/(TP+FP) and recall is TP/(TP+FN) and TP, FP and FN denote the number of true positives, false positives, and false negatives.

Algorithm 3 also generates specialized subgroups if unitary exceptional subgroups are found. Specialized subgroups described by A_1 are also considered as exceptional. A subgroup such as (a_1^+, r_1^-) can be seen as a selection of half the sequences of subgroup (a_1^+), making the models of (a_1^+, r_1^-) and (a_1^+) similar. By opposition, specialized subgroups without A_1 are considered as non-exceptional. To facilitate comparisons, we evaluate unitary and specialized subgroups separately as the number of generated specialized subgroups can vary over different simulations. We used a size threshold $\sigma = 0.05$.

As a baseline, we consider Markov chains for representing the temporal targets instead of a DBN. In this case, the search algorithm is the same but the temporal targets are represented by a MC. To learn a MC, each variable $\mathbf{X}^{(t)}$ was mapped into a single variable $X'^{(t)}$ which has as domain the Cartesian product of the domains of $X_1, \ldots, X_n$. As a result, the state space of this MC can have up to 1,024 and 131,072 states for $n = 10$ and $n = 17$ respectively. Then, the temporal data of each sequence $\mathbf{X}^{(0)}, \mathbf{X}^{(1)}, \ldots$ was mapped into $X'^{(0)}, X'^{(1)}, \ldots$. This allows for an additional assessment of the DBN representation. To avoid zero probabilities, a Laplace smoothing [104] with $\lambda = 1$ is used in MC and DBN learning.

7.6.3 *Results*

Figure 7.1 shows the results based on simulated data for unitary subgroups. 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. Table 7.2 shows the precision and recall of MCs and DBNs based on the threshold $\alpha = 0.05$ of Algorithm 4.

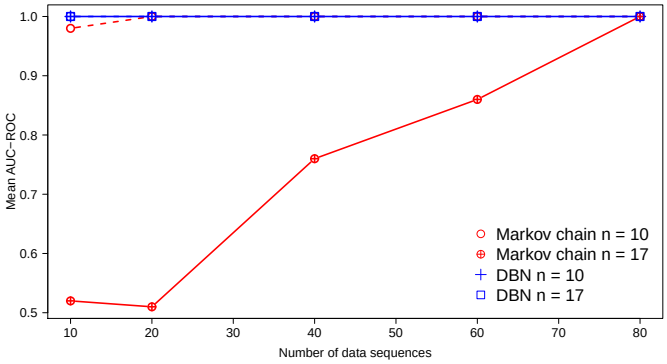

Figure 7.1: Effect of the amount of simulated data on the AUC-ROC of Markov chains and DBNs. Every sequence has 10 time points.

As previously discussed, specialized subgroups that include A_1 are supposed to be labeled as exceptional subgroups. Figure 7.2 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 significantly more specialized exceptional subgroups.

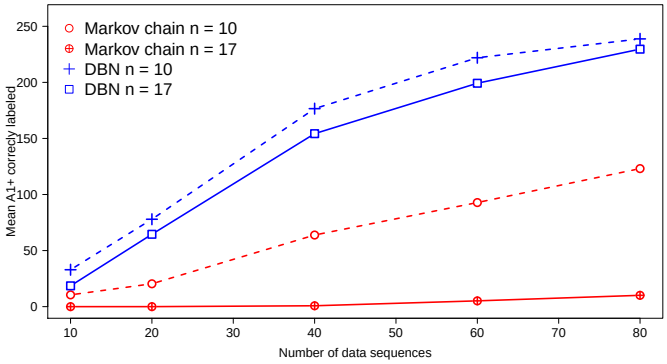

Figure 7.2: Mean number of specialized subgroups with A_1 which were labeled as exceptional (simulated data).

7.6.4 *Similar ground truth models*

Now we consider simulations where we control how similar the ground truth models are. This allows for a complementary evaluation of the search algorithm than that where we essentially varied the amount of data supplied to the algorithm. As before, two ground truth models are associated to the binary descriptor A_1.

In the following experiments, the second ground truth DBN was defined by copying the structure and parameters of the first DBN. For a variable X_i in the second DBN we have $p = P(X_i^{(0)} = x_i^- \mid \pi(x_i^{(0)}))$ and $p' = P(X_i^{(0)} = x_i^+ \mid \pi(x_i^{(0)}))$. These parameters were 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 $p = p + change$ and $p' = p' - change$. The lower the threshold δ, the more similar the DBNs are. It is straightforward to see that the modified p and p' values constitute a valid probability distribution.

Based on the previous results, we focus the analysis on DBNs in the remaining of this chapter. Figure 7.3 shows the AUC-ROC of simulations based on different maximal change thresholds. The results suggest that the search algorithm achieved better results with higher δ, which is expected because with more dissimilar ground truth models detecting exceptional behavior becomes more straightforward. On the other hand, the method made more mistakes under lower δ, particularly when there was little data, which can be seen as difficult situations for the method. In general, with larger amounts of data the method had better performance with any δ, which supports a behavior consistent with the previous experiments.

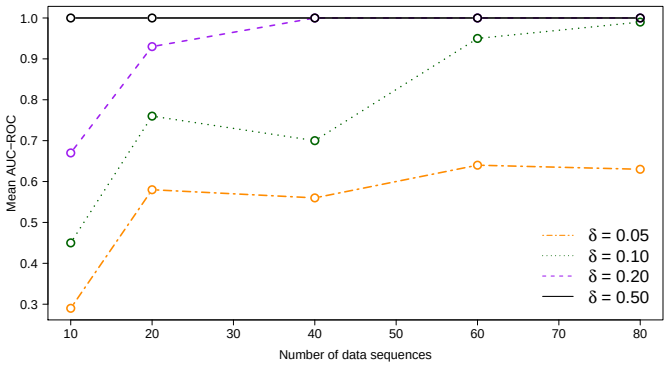

Figure 7.3: AUC-ROC achieved by DBNs on simulated data from different ground truth models. δ = maximal change threshold.

7.6.5 *Discussion*

Table 7.3 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.

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

Subgroup	Size	z-score	p-value	Labels (I & 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

Table 7.3: A simulation iteration based on DBNs ($n = 17$, 80 data sequences). **Size** = subgroup size normalized by $|D|$, **Labels (I&T)** = inferred and true labels respectively. The labels '1' and '0' indicate positive and negative instances respectively.

[112] 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. This is because the dimension of the transition matrices of MCs is prone to become very large even with a moderate number of target variables (e.g. $n = 17$).

As experiments have shown, this parameter issue makes the MC representation to scale poorly, particularly when n is larger and there is a reduced availability of data for model learning. Furthermore, the DBN-based search made substantially less mistakes in the simulations, which makes this representation suitable for TEMM.

7.7 DATA OF FUNDING APPLICATIONS

In order to evaluate the proposed TEMM method, we consider data from the *business process intelligence challenge* (BPIC18, for short) [56]. The BPIC18 dataset contains event log data of applications submitted to the European Union for direct payments for German farmers in 2015, 2016 and 2017. *The goal of applying TEMM to the BPIC18 data is to identify the subgroups in which the dynamic of events is exceptional*.

7.7.1 *Data*

Each application in the BPIC18 data is associated to 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 to *events* related to workflow activities, where an event is described by the multinomial variables (domain size): **Activity** (41), **Doctype** (8), **Subprocess** (8). Each application is associated to one or more events, which are the temporal targets of the data. Hence, the ith data point of this dataset has the form $\{\text{Land Area}, \ldots, \text{Young Farmer}, \text{Activity}^{(0:m_i)}, \ldots, \text{Subprocess}^{(0:m_i)}\}$, where m_i is its last time point.

The BPIC18 dataset has 4,800 applications randomly selected from the original dataset, with an equal number of applications per year. The dataset considered for the experiments has 275,226 events in total (mean [StDv] length of each application: 57.3 [49.5] events).

7.7.2 *Discovered subgroups*

Table 7.4 shows the exceptional and non-exceptional subgroups that were discovered from the BPIC18 data based on a minimal size $\sigma = 0.05$. The results suggest that the most exceptional subgroups are unitary and described by a particular year, be it 2015, 2016 or 2017. This might suggest that significant changes took place in application processing between different years, such as changes in application structure, time spent in application tasks, funding policies, etc. Regardless of the year, each department has its own dynamics, as all unitary subgroups (Department) were exceptional. However, their the exceptionality was not as strong as that of (Year) subgroups.

As Table 7.4 shows, unitary subgroups of (Young Farmer) were not exceptional, which suggests that the exceptionality of subgroups as $(\text{Year} = 2017 \wedge \text{Young Farmer}^{-})$ is only caused by other attributes. Due to the large size of $(\text{young.farmer}^{-})$, we conjecture that some specialized subgroups of (Young Farmer) have distributions similar to their generalized subgroups without (Young Farmer), which would make such specialized subgroups redundant.

7.7.3 *Validation*

The BPIC18 data provider [56] claims that the underlying process changed between years due to changes implemented in the structure of the application procedure. This is evidence that supports the exceptional subgroups found in this chapter described by (Year), as shown in Table 7.4.

Such discovered exceptional subgroups are also in line with previous research [135] applied to this dataset, which was able to identify concept drifts precisely between each year of the data. Other research [174] has analyzed how the workflow of applications submitted in different years has changed, also suggesting that differences exist in the workflow structure between years.

Differently than the other analyses from the literature on the BPIC18 data, the method proposed in this chapter can be seen as a principled one due to its automated nature. However, the discussed validation of the subgroups found should be seen as a partial validation, as the true exceptional subgroups of real-world data are usually unknown.

7.8 CONCLUSIONS

In this chapter, we proposed TEMM, a generalization of EMM to allow for the representation of multiple and temporal targets. We proposed a method able to

Exceptional subgroups	**Size**	**z-score**
year = 2017	0.34	773.6
year = 2015	0.35	524.1
year = 2016	0.30	479.0
department = e7	0.30	23.4
department = d4	0.16	21.3
department = 4e	0.30	13.1
department = 6b	0.24	11.3
number_parcels = 2	0.06	7.2
year = 2017 $\wedge$ young.farmer$^-$	0.31	385.0
year = 2015 $\wedge$ young.farmer$^-$	0.32	363.7
department = e7 $\wedge$ year = 2017	0.10	166.6
department = 6b $\wedge$ year = 2017	0.09	110.9
department = 6b $\wedge$ year = 2016	0.07	106.7
department = 6b $\wedge$ young.farmer$^-$ $\wedge$ year = 2016	0.06	147.6
department = e7 $\wedge$ young.farmer$^-$ $\wedge$ year = 2017	0.09	128.2
department = 4e $\wedge$ young.farmer$^-$ $\wedge$ year = 2017	0.09	124.9
department = 6b $\wedge$ young.farmer$^-$ $\wedge$ year = 2017	0.08	118.3
department = e7 $\wedge$ young.farmer$^-$ $\wedge$ year = 2016	0.08	69.6
Non-exceptional subgroups	**Size**	**z-score**
young.farmer$^-$	0.91	1.7
young.farmer$^+$	0.09	1.3
number_parcels = 3	0.06	0.9
department = e7 $\wedge$ year = 2015	0.11	0.2
department = 4e $\wedge$ year = 2015	0.10	-1.6

Table 7.4: Exceptional (34) and non-exceptional (5) subgroups from the BPIC18 dataset. For better visualization, the 5 most specialized subgroups are shown. **Size** = subgroup size normalized by $|D|$. All p-values < 0.001 (exceptional subgroups) and ≥ 0.05 (non-exceptional subgroups).

identify exceptional DBNs from temporal data, which allows for an intuitive and sound model class for TEMM.

The proposed TEMM method was empirically evaluated on simulated data and a 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 better explain why models are considered as exceptional, e.g., by looking at relevant structural or numerical parameters of the DBNs. We also wish to summarize exceptional subgroups that might reflect the same DBN distribution, e.g., by merging exceptional subgroups during search or post-processing. Moreover, by investigating the relation between subgroup size and the mismatch distance, the search mechanism could be further optimized.