

xPM: A Framework for Process Mining with Exogenous Data

Adam Banham¹[0000–0001–9912–8220], Sander J.J. Leemans¹[0000–0002–5201–7125], Moe T. Wynn¹[0000–0002–7205–8821], and Robert Andrews¹[0000–0001–7743–5772]

Queensland University of Technology, Brisbane, Australia

Abstract. Process mining facilitates analysis of business processes using event logs derived from historical records of process executions stored in organisations’ information systems. Most existing process mining techniques only consider data directly related to process execution (endogenous data). Data not directly representable as attributes of either events or traces (which includes exogenous data), are generally not considered. Exogenous data may be used by process participants in making decisions about execution paths. However, as exogenous data is not represented in event logs, its impact on such decision making is opaque and cannot currently be assessed by existing process mining techniques. This paper shows how exogenous data can be used in process mining, in particular discovery and enhancement techniques, to understand its influence on process decisions. In particular, we focus on time series which represent periodic observations of e.g. weather measurements, city health alerts or patient vital signs. We show that exogenous time series can be aligned and transformed into new attributes to annotate events in an event log. Then, we use these attributes to discover preconditions in a Petri net with exogenous data (xDPN), thus revealing the exogenous data’s influence on the process. Using our framework and a real-life data set from the medical domain, we evaluate the influence of exogenous data on decision points that are non-deterministic in an xDPN.

Keywords: Process mining · Decision mining · Petri nets with data · Context awareness · Time series data

1 Introduction

Process mining is a field that uses historical event data extracted from an organisation about a business function (process) to better understand its behaviour and performance [1]. Process mining techniques rely on a single ‘source of truth’, an event log containing process instances (traces), and a sequence of events (the “what” happened and “when” it happened) for each process instance.

Process discovery techniques [25,13] exploit event sequences presented in an event log to recreate the structure of a business process. Conformance techniques [4,3] use a process model and an event log to create aligned event se-

quences that follow the possibilities described in the process model. Enhancement techniques [1,20] enrich a process model with additional influences such as the performance or the resource utilisation of events.

In our work, we use the following definitions to distinguish between data that can, and can not be represented in event logs effectively. We define *endogenous* data as data internal to a process, meaning they have a direct link to a specific process’s progress towards its goal. For example, endogenous data could include: the time that an event occurred, the resource which performed the activity, any information needed to perform the activity or the cost of completing the activity.

In contrast, we define *exogenous* data as data external to a process, meaning that they are not tied to a specific process, but record contextual data. For example, exogenous data could be the temperature and humidity readings inside a food delivery truck, or periodic readings from a sensor monitoring a patient’s heart rate, or the noise levels in an employee’s work space. The purpose in recording exogenous data is to describe the context as clearly as possible over time, meaning that records are taken as frequently as possible (i.e., time series) rather than more selective point-in-time recordings usually associated with endogenous data. While data-aware or context-aware techniques exist, such as techniques presented in [20], [24] or [25], we have not found any studies which use exogenous data in conjunction with these techniques.

In this paper, we study the potential of exogenous data to improve our understanding of complex decision points in processes. In particular, we focus on a particular type of exogenous data, i.e., numerical time series. We proposed a novel process mining framework, *xPM*, that translates exogenous time series data and links them to relevant events in an event log for automated process discovery and enhancement. A data-aware process discovery technique can then be used to discover a process model in which the decision points are annotated with preconditions using exogenous data. Finally, an enhancement step which visualises related exogenous data for transitions on a process model is envisioned. We instantiated xPM and evaluated the influence of exogenous data on the quality of the discovered process model using a real-life data set from the medical domain.

The remainder of the paper is organised as follows: Section II outlines related work. Section III defines the preliminaries. Section IV presents xPM. Section V discusses the evaluation, and Section VI concludes the paper.

2 Related Work

Categorisation of data sources used to describe businesses has been discussed in several studies. The ‘onion skin’ model in [22] conceptualises the relationship between data and process as the viewpoint is moved further away from a process. This conceptualisation is then applied to process mining in [2], where data is categorised according to the likelihood of cause and effect between variables with the process. However, these frameworks are not seen as essential to process mining in recent reviews of the field, and the contextual component remains an

optional consideration during event data extraction [7,9,21]. Our contribution is that we support separate entities for endogenous and exogenous data sources such that they can be studied separately or in combination.

The benefits of including a variety of data categories are discussed in [16] which (i) motivates the use of data attributes for distinguishing between noise and conditional behaviour, (ii) considers if data attributes influence decision points by creating an internal state as the process executes through boolean expressions and decision trees, and (iii) studied how alignments [3] can be extended such that they balance both the control flow perspective and the data perspective. The techniques described in [16] were implemented using Petri Nets with Data (DPN) modelling language (see [6,16,10] for a complete definition).

Our study extends the concepts presented in [16] and shows how exogenous data can be incorporated (instead of being limited to only the endogenous perspective of an event log). In particular, we focus on extending guard conditions in DPNs to include external factors not represented in the endogenous event log.

Methodologies that encourage contextual data collection and log enrichment are few in number. However, some recent studies have focused on the enrichment of an event log with new types of data. In [23], the authors present a framework for intra- and inter-trace predictive monitoring and introduce the notion of *bi-dimensional coding* to deal with intra- and inter-trace dependencies. In [8], the authors suggest that not all events within an event log are about the control flow, and are instead, about the data flow of a process. They use the concept of *context events* to deal with the two types of events and show how distinguishing between the two can lead to less complex discovered models. However, this approach would incorporate exogenous context into the control flow perspective instead of clarifying whether the context influences process execution.

The benefits of having additional data attributes that can be seen in the recent evolution of techniques using such data, such as [14,25]. In [14], the authors present a discovery algorithm that uses data attributes to create a hierarchical model to improve the simplicity of outcomes. Another approach in [25], was to create a constraint operator for process trees notation, whereby data semantics can be expressed. While these techniques can create control flow sequences based on data attributes, no extensions have been proposed to use exogenous sources outside what can be found in the events within an event log.

3 Preliminaries

This section introduces event logs, Petri nets, xDPNs and exogenous data sets.

Event logs. The execution of each process step can be recorded as an event. An end-to-end execution of a process is called a *trace*. A trace is a sequence of events $\langle e_1, \dots, e_n \rangle$. An event log is a collection of such traces. Both traces and events can have attributes to store data.

Exogenous Data Sets. A *time series* is a sequence of timestamped values $\langle m_1^{@t_1} \dots m_i^{@t_i} \rangle$ for measures m_i and timestamps t_i . For example, $\langle 22^\circ\text{C}_1^{01-01-2021}, 15^\circ\text{C}_2^{02-01-2021}, 10^\circ\text{C}_3^{03-01-2021} \rangle$. A time series can have attributes and uses the

notation of a trace to describe the i th measurement of a time series. A collection of time series describing the same exogenous context is an *exogenous data set*. For example, a collection of time series for wind speed, where each series is recording wind speeds for a local government area is an exogenous data set.

Labelled Petri Nets. A *Petri net* is a triple $N = (P, T, F)$, where P is a finite set of places, T is a finite set of transitions such that $P \cap T = \emptyset$ and $F \subset (P \times T) \cup (T \times P)$ is a set of directed arcs, called a flow relation [1]. A *labelled Petri net* is a quintuple, $(P, T, F, \Sigma, \lambda)$, where (P, T, F) is a Petri net, Σ is a set of observed activity names and λ is a event labelling function $T \rightarrow \Sigma$ [1]. Places may hold tokens, which are produced and consumed when transitions fire according to the flow relation. A transition is *enabled* if each input place contains a token. The state of a Petri net is a *marking*, which records what places have tokens and how many. An enabled transition l can *fire*, which updates the marking according to the flow relation F and, if l is labelled by λ , denotes the execution of activity $\lambda(l)$. An initial marking denotes the initial state of a Petri net before the first transition is fired.

Petri Nets with Exogenous Data (xDPN). A *precondition* is a boolean expression describing a subset of values for attributes (e.g. temperature is higher than 20°C). A *Petri Net with Exogenous Data* (xDPN) is a sextuple $(P, T, F, \Sigma, \lambda, \Phi)$, where $(P, T, F, \Sigma, \lambda)$ is a labelled Petri net and $\Phi : T \rightarrow \phi$ associates a transition with a precondition. A transition is *data enabled* if the precondition attached to a transition is satisfied by the current assignment of attributes or if there is no attached precondition. In an xDPN, a transition can fire if it is enabled and data enabled. The state of an xDPN is described by a marking, and an endogenous and exogenous data state. An xDPN is a sub-formalism of DPN (a complete formalisation of DPN can be found in [6,16,10]): in contrast to xDPN, DPN consider distinctions between attributes states (e.g. **read** or **written**). Furthermore, xDPNs do not enforce that transitions in the model update variable assignments, allowing exogenous data attributes to be updated during execution.

4 A Framework for Process Mining with Exogenous Data

In this section we introduce xPM, which considers how exogenous data can be used by process mining techniques. Figure 1 shows an overview of xPM. xPM takes as input an event log and a collection of exogenous data sets (\mathcal{X}). xPM uses a number of quadruples $(x, \mathcal{L}, \mathcal{S}, \mathcal{T})$, where $x \in \mathcal{X}$ is an exogenous data set; \mathcal{L} is a linking function, which links traces to exogenous time series that are relevant for that trace; \mathcal{S} is a slicing function, which, for each event, returns sub-time series relevant for that event; and \mathcal{T} is a transformation function, which summarises each sub-time series into a set of transformed attributes. For each such quadruple, xPM annotates each event (that has non-empty sub-time series) with their exogenous sub-time series and transformed attributes, creating a *exogenous-annotated log* (xlog). Next, a discovery function \mathcal{D} discovers an xDPN. Finally, an enhancement step \mathcal{E} aligns a log and a xDPN. Then for each

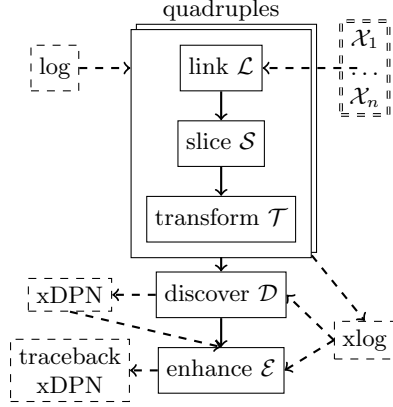


Fig. 1: The xPM framework.

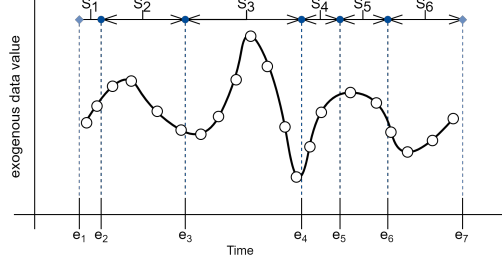


Fig. 2: Visualisation of exogenous data sliced by \mathcal{S} to create sub-time series s_i . Each e_i is annotated with s_{i-1} .

aligned event, we *trace back* from the exogenous transformed attribute to sub-time series. Finally, \mathcal{E} visualise the subset of exogenous data set relevant for each transition (*traceback* xDPN).

4.1 Linking

The first step of xPM is to find a subset of an exogenous data set related to each trace using a linking function \mathcal{L} . This linking function can consider many different aspects of a trace when creating this subset, and it may also consider if a trace has or has not been linked to other exogenous data sets. For example, an event log could be capturing how an insurance company handles claims. Then, an exogenous data set could capture time series of weather predictions for local government areas. An \mathcal{L} would link the time series in this data set to claims. To create a subset of time series, \mathcal{L} might compare the location of a claim and the location of weather predictions. However, a more complex \mathcal{L} could find adjacent government areas and interpolate between weather predictions to predict if an extreme weather event will likely occur.

In case an \mathcal{L} links two or more time series to a particular trace, this \mathcal{L} must merge these time series into a single time series. Simply combining all time series onto a single timeline is insufficient as multiple values could be recorded at a timestamp. Handling this case is not trivial and will require a thorough understanding of exogenous data or domain knowledge. As such, in this paper we limit the scope of exogenous data to time series of numerical data and limit \mathcal{L} to link only one time series to each trace. We acknowledge that this is a simplified view of exogenous data and does not account for all types of exogenous data possible; an extension of xPM could consider how an additional internal step could compress larger subsets into a single time series.

4.2 Slicing

The second step of xPM is to annotate events with relevant exogenous data. That is, events will be annotated with the sub-time series using a *slicing* function \mathcal{S} . Figure 2 illustrates an example of a simple slicing function: each event e_i of a trace $\langle e_1, \dots, e_n \rangle$ is annotated with the sub-time series between the previous event e_{i-1} and e_i .

More elaborate slicing functions could use a process model to ignore concurrent events when determining the previous event, or to only annotate events relevant to decision points in the model. Other possibilities include taking a fixed time window, for instance, for an event, taking the past two days of rainfall measurements to watch for flash flooding. Another possible slicing algorithm could use knowledge of activity instances (i.e. start and completion events) in order to create sub-time series observed during a execution of an activity.

These examples are not exhaustive; however, we highlight the potential of creating an extensive array of slicing algorithms to suit the needs of an analyst. Domain knowledge then informs the choice of slicing functions; assisting this choice is an interesting area of further research.

4.3 Transformation

Next, a transformation function, \mathcal{T} , transforms a sub-time series for an event into attributes and annotates the event with these attributes. Each new attribute created in this way for a event is referred to as a *transformed* attribute. The \mathcal{T} function needs to provide a name for each attribute it creates (which can be trivially met by adding a suffix to the exogenous data set’s name). Furthermore, transformations should reference sub-time series by an identifier so that outcomes that use transformations can be traced back to the original sub-time series for further analysis.

We identified three forms that a \mathcal{T} can take: (i) \mathcal{T} can return a single value to annotate an event; such a transformation might return the minimum, maximum or mean of a sub-time series; (ii) \mathcal{T} can return a set of attributes to annotate an event; such a transformation might be the n th Taylor polynomial of the sub-time series, with each of the necessary coefficients; (iii) \mathcal{T} can be recursive, which applies several recursions in order to meet either case (i) or (ii). Such a transformation finds the n th derivative of the sub-time series (where the sub-time series is a continuous function) then applies any previously mentioned functions.

4.4 Discovery

The output of several quadruples $(x, \mathcal{L}, \mathcal{S}, \mathcal{T})$ is an event log, with some events annotated with (i) sub-time series and (ii) transformed attributes. We refer to such an event log as an *exogenous-annotated* log (xlog). To this xlog, a discovery function \mathcal{D} is applied. This study only considers \mathcal{D} functions that use data-aware discovery techniques to obtain a process model with preconditions for transitions using the transformed attributes. Examples of such techniques

are [17] and [19]. Preconditions found by these techniques do not create boolean expressions between `written` or `read` and as such can be translated into an xDPN. Furthermore, when discovering preconditions using these techniques, the attributes that have been set by preceding events inform the discovery. As data leading up to the event is not considered, future decision mining techniques capable could handle differences in exogenous and endogenous attributes.

4.5 Enhancing

As the final step of xPM an enhancement step \mathcal{E} visualises the sub-time series from an xlog using the outcome of \mathcal{D} to highlight points of interest. To create a connection between the events in an exogenous-annotated log and a discovered xDPN, we need to use process conformance techniques to find alignments. While data-aware alignments exist (e.g. [4,16]), [4] only considers the writing of attributes by transitions (and not whether preconditions hold) and [16,4] correct the data written by transitions using Integer Linear Programming. In contrast, in our context, exogenous data should not be adjusted in conformance techniques as it occurs outside the internal process execution. Therefore, to verify whether preconditions are met, our approach first computes alignments [3], after which we verify preconditions separately.

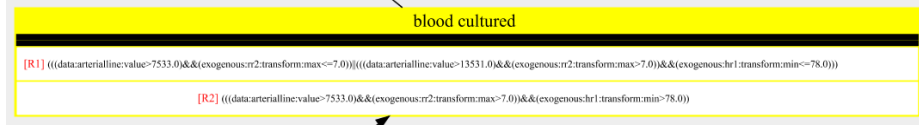
Given that the alignments proposed in [3] do not consider the data perspective, we present following example of \mathcal{E} which uses alignments. First, an alignment between all traces and an xDPN is computed. Then for each aligned transition in the xDPN, we collect the most recent sub-time series in preceding events and plot all series from the same exogenous data set on a graph. Then we consider the type of alignment move that occurred in the alignment for that transition. If we see a synchronous move and this transition has a precondition, we check the following. (1) If the precondition was satisfied then sub-time series related to the aligned event of this move is plotted in green. (2) If the precondition was not satisfied then sub-time series related to the aligned event of this move is plotted in red. (3) Otherwise – e.g. a non-synchronous move – we plot the related sub-time series in black. Figure 3 is an example of such a visualisation, which has been implemented in a ProM plugin, Exogenous Data.

5 Evaluation

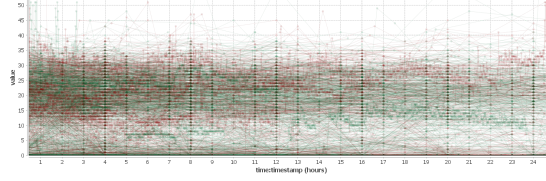
In this section, we instantiate xPM presented in Section 4. Then we evaluate, using two event logs from a real-life data set in the medical domain and existing DPN discovery techniques, the influence of exogenous data on the quality of the discovered xDPNs.

5.1 Procedure

We used the event logs either (i) as an event log with endogenous data attributes (*endo*), (ii) as an event log with exogenous attributes where endogenous



(a) A transition with a disjunctive precondition of R1 and R2.



(b) An exogenous data set's (RR2) sub-time series for Figure 3a.

Fig. 3: An example of \mathcal{E} for a transition, showing exogenous sliced time series.

attributes have been removed (*exo*), and (iii) as an event log with both endogenous and exogenous attributes (*endo+exo*).

Our instantiation of xPM is as follows:

- \mathcal{L} For each exogenous data set, a linking function was defined that linked data sets to the patient of the trace and that occurred during the admission.
- \mathcal{S} We included two slicing functions. Let $\langle e_1 \dots e_n \rangle$ be a trace. Then, for event e_i the first slicing function (S_1) finds sub-time series between events e_{i-1} and e_i , while the second slicing function (S_2) finds the sub-time series between e_1 and e_i .
- \mathcal{T} We included four transformation functions: minimum, average, maximum and the cumulative sum of a Fourier transform¹ [11].
- \mathcal{D} To discover a control-flow model, we applied the Inductive Miner - infrequent [13] with path filtering of 0.25. To discover an xDPN, we applied two Data Petri Net discovery techniques: Mutually Exclusive Decision Tree (**dt**) [5] and Overlapping Rules Decision Tree (**or**) [19]. These techniques each take a parameter *min instances* (mi) that sets the minimum level of observed decision point instances that support a clause in a precondition. We repeated the experiment for $mi \in \{0.05, 0.15, 0.25\}$.
- \mathcal{E} was not part of this experiment.

Thus, in total, 18 xDPNs were discovered for each of the two logs. A visual breakdown of our instantiation can be seen in Figure 4.

5.2 Quality Measures

We assessed the quality of the discovered xDPNs using fitness, precision and determinism. For fitness, we used balanced multi-perspective conformance checking [18]. For precision, we used the multi-perspective precision [16].

¹ <https://docs.scipy.org/doc/scipy/reference/generated/scipy.signal.stft.html#scipy.signal.stft>

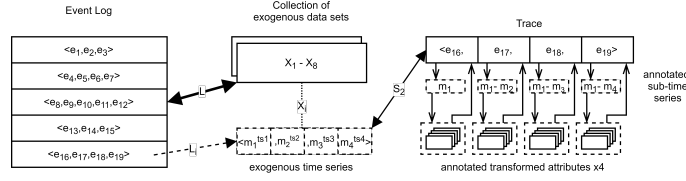


Fig. 4: Visual breakdown of quadruples applied in xPM.

For determinism, we propose the following measure, which expresses the decision points in the model that are deterministic. That is, a fraction of places in the model with more than two outgoing arcs (decision points) that have at least one outgoing arc to a transition that has *no* precondition. Formally, let $N = (P, T, F)$ be a Petri net.

$$\text{decision points or } dp(P) = \{p \mid p \in P \wedge p^\bullet \geq 2\} \quad (1)$$

$$\text{weight or } w(p) = \frac{|\{t \mid t \in p^\bullet \wedge \Phi(t) \neq \perp\}|}{|p^\bullet|} \quad (2)$$

$$\text{Determinism or } D(P) = \begin{cases} \frac{\sum_{\{p \in dp(P)\}} w(p)}{|dp(P)|} & \text{if } |dp(P)| > 0 \\ 1 & \text{otherwise} \end{cases} \quad (3)$$

A D value of 1 implies that all transitions that are involved in choices in the model have preconditions, while a value of 0 indicates that no transition that is involved in a choice has a precondition.

5.3 Event Logs & Exogenous Data

The data for our experiments is derived from the MIMIC-III data set [12]. MIMIC-III records patient demographics, admissions, ward stays, clinical observations, labs, imaging, prescriptions, caregiver notes, etc., for over forty thousand patients who stayed in critical care units between 2001 to 2012.

We created two event logs: a log of patient movements (*movements log*) and a log of procedures for respiratory failures (*procedures log*). The extraction scripts for these two event logs can be found in this repository ². The *movements log* captures the movements of patients between ICU wards within a single hospital admission, and contains 24 271 traces, 290 462 events, 65 activities and 6 endogenous attributes. The *procedures log* captures a process which describes the procedures that a patient received during a single hospital admission and contains 65 traces, 610 events, 34 event classes and 4 endogenous attributes. Both logs have 8 exogenous data sets (respiratory rate, 3x heart rate, 2x oxygen saturation, 2x arterial blood pressure). The *movements log* has 25 684 680 exogenous data points; the *procedures log* has 590 285 exogenous data points.

² <https://github.com/adamBanham/icpm2021>

Table 1: Experimental Results.

| Variant | \mathcal{D} | mi | Movements log | | | Procedures log | | |
|--------------|---------------|------|---------------|--------------|--------------|----------------|--------------|--------------|
| | | | fit. | pre. | D | fit. | pre. | D |
| endo | dt | 0.25 | 0.586 | 0.644 | 0.048 | 0.739 | 0.395 | 0.119 |
| | | 0.15 | 0.573 | 0.656 | 0.108 | 0.739 | 0.392 | 0.119 |
| | | 0.05 | 0.573 | 0.644 | 0.140 | 0.761 | 0.418 | 0.167 |
| | or | 0.25 | 0.586 | 0.657 | 0.048 | 0.785 | 0.393 | 0.131 |
| | | 0.15 | 0.583 | 0.656 | 0.108 | 0.785 | 0.393 | 0.131 |
| | | 0.05 | 0.587 | 0.647 | 0.140 | 0.761 | 0.419 | 0.167 |
| endo +exo | dt | 0.25 | 0.575 | 0.591 | 0.079 | 0.692 | 0.409 | 0.155 |
| | | 0.15 | 0.518 | 0.537 | 0.156 | 0.705 | 0.411 | 0.155 |
| | | 0.05 | 0.465 | 0.533 | 0.283 | 0.717 | 0.459 | 0.238 |
| | or | 0.25 | 0.586 | 0.649 | 0.048 | 0.731 | 0.445 | 0.214 |
| | | 0.15 | 0.536 | 0.559 | 0.156 | 0.739 | 0.430 | 0.179 |
| | | 0.05 | 0.465 | 0.550 | 0.259 | 0.717 | 0.463 | 0.238 |
| exo | dt | 0.25 | 0.654 | 0.640 | 0.000 | 0.722 | 0.413 | 0.143 |
| | | 0.15 | 0.654 | 0.623 | 0.000 | 0.697 | 0.436 | 0.143 |
| | | 0.05 | 0.173 | 0.567 | 0.222 | 0.709 | 0.420 | 0.214 |
| | or | 0.25 | 0.654 | 0.628 | 0.000 | 0.701 | 0.512 | 0.274 |
| | | 0.15 | 0.654 | 0.639 | 0.000 | 0.697 | 0.425 | 0.143 |
| | | 0.05 | 0.099 | 0.582 | 0.198 | 0.709 | 0.424 | 0.214 |

5.4 Results & Discussion

Table 1 shows the results. The best results for each log appear in **boldface**. When considering the *movements log*, using exogenous data only (exo) does not introduce preconditions in most cases, and henceforth the fitness and precision values are high. In cases where it does introduce preconditions, fitness is very low but precision is competitive. We conclude that for this log, the exogenous data by itself does not suffice. For exo+endo, typically more preconditions are discovered, which lowers fitness and precision (at most 0.11 lower than endo). This is to be expected, as adding more preconditions to the xDPN means that multi-perspective measures will consider more data attributes from the event log, thus increasing the state space on which precision is based.

When considering the *procedures log*, surprisingly, larger values of the parameter mi did not always decrease the number of preconditions found (D , or, endo+exo) as is to be expected as mi is a support threshold. We suspect that the rather small size of the procedures log and the nature of the overlapping rules (**or**) algorithm is at play here, which after building a first precondition, there is not enough observations left for a second precondition to meet the mi threshold. For this log, using the exogenous data increased the determinism and hence the number of preconditions found (exo+endo and exo vs. endo). Consequently, for endo and endo+exo, fitness goes up with mi for **dt** and goes down for **or**, but for exo these patterns are not there. If we consider exo and endo+exo vs. endo, then fitness consistently decreases, precision consistently increases, and determinism consistently increases. We suspect that the preconditions cover a larger fraction of the increased state space than for the *movements log*.

A possible extension of our analysis would be to understand if cohort analysis [15], separating patients into distinct care groups, would change the efficacy

of our approach, allowing us to consider if observations in *procedures log* can be seen in medically relevant cohorts of patients.

6 Conclusion

In previous studies, exogenous data has been undeveloped when considering guard conditions in DPNs. As such, exogenous data's influence on process participants and decision-making in a process execution has not been considered in depth. This paper presents xPM, a framework for using exogenous data in process mining techniques that does not limit analysis opportunities. xPM allows for complex analysis of exogenous data and process executions, using existing process mining techniques and increased traceability – by means of slicing functions – between events and exogenous data. We evaluated the influence of exogenous data on process model discovery by measuring the difference in process model quality. Our evaluation showed that we could understand more decision points by including exogenous data and can improve fitness.

We see several extensions in future work. The semantics of xDPN could be expanded to introduce ways of expressing exogenous data sets alongside the process execution rather than solely within preconditions. Other data-aware process mining techniques could be used instead of the proposed techniques in our instantiation, such as [14,25]. Decision mining techniques for discovering preconditions could be extended to consider if an transformed attribute or exogenous data set correlates with the process activity before discovering a precondition. A variety of visualisation for the enhancement step could exist, and analysis could be expanded to consider more than satisfaction of a precondition to creating new modes of engagement with domain experts.

References

1. van der Aalst, W.M.P.: Process Mining. Springer Berlin Heidelberg (2016)
2. van der Aalst, W.M.P., Dustdar, S.: Process mining put into context. *IEEE Internet Computing* **16**(1), 82–86 (2012)
3. Adriansyah, A.: Aligning observed and modeled behavior. Ph.D. thesis, Technische Universiteit Eindhoven (2014)
4. De Leoni, M., van der Aalst, W.M.P.: Aligning event logs and process models for multi-perspective conformance checking: An approach based on integer linear programming. In: *Lecture Notes in Computer Science*, pp. 113–129. Springer (2013)
5. De Leoni, M., van der Aalst, W.M.P.: Data-aware process mining. In: *Proceedings of the 28th Annual ACM Symp. on Applied Computing - SAC '13*. ACM (2013)
6. De Leoni, M., Felli, P., Montali, M.: A holistic approach for soundness verification of decision-aware process models. In: *Concept. Model.*, pp. 219–235. Springer (2018)
7. De Smedt, J., van den Broucke, S.K.L.M., Obregon, J., Kim, A., Jung, J.Y., Vanthienen, J.: Decision mining in a broader context: An overview of the current landscape and future directions. In: *BPM Workshops*, pp. 197–207. Springer (2017)
8. Dees, M., Hompes, B., van der Aalst, W.M.P.: Events put into context (EPiC). In: *ICPM*. IEEE (2020)

9. Diba, K., Batoulis, K., Weidlich, M., Weske, M.: Extraction, correlation, and abstraction of event data for process mining. *WIREs Data Mining and Knowl. Discovery* **10**(3) (2019)
10. Felli, P., De Leoni, M., Montali, M.: Soundness verification of decision-aware process models with var.-to-var. conditions. In: 2019 19th Int. Conf. on ACSD. IEEE (2019)
11. Griffin, D., Lim, J.: Signal estimation from modified short-time fourier transform. *IEEE Trans. on Acoustics, Speech, and Signal Processing* **32**(2), 236–243 (1984)
12. Johnson, A.E., Pollard, T.J., Shen, L., wei H. Lehman, L., Feng, M., Ghassemi, M., Moody, B., Szolovits, P., Celi, L.A., Mark, R.G.: MIMIC-III, a freely accessible critical care database. *Scientific Data* **3**(1) (2016)
13. Leemans, S.J.J., Fahland, D., van der Aalst, W.M.P.: Discovering block-structured process models from event logs containing infrequent behaviour. In: BPM Workshops. Lecture Notes in Bus. Inf. Process., vol. 171, pp. 66–78. Springer (2013)
14. Leemans, S.J.J., Goel, K., van Zelst, S.J.: Using multi-level information in hierarchical process mining: Balancing behavioural quality and model complexity. In: 2020 2nd ICPM. IEEE (2020)
15. Leemans, S.J.J., Shabaninejad, S., Goel, K., Khosravi, H., Sadiq, S.W., Wynn, M.T.: Identifying cohorts: Recommending drill-downs based on differences in behaviour for process mining. In: ER. pp. 92–102. Springer (2020)
16. Mannhardt, F.: Multi-perspective process mining. Ph.D. thesis, Technische Universiteit Eindhoven (2018)
17. Mannhardt, F., De Leoni, M., Reijers, H.A.: The multi-perspective process explorer. In: BPM Conference Demos. vol. 1418, pp. 130 – 134. CEUR-WS (2015)
18. Mannhardt, F., De Leoni, M., Reijers, H.A., van der Aalst, W.M.P.: Balanced multi-perspective checking of process conformance. *Computing* **98**(4), 407–437 (2015)
19. Mannhardt, F., De Leoni, M., Reijers, H.A., van der Aalst, W.M.P.: Decision Mining Revisited - Discovering Overlapping Rules. In: Adv. Inf. Systems Engineering. pp. 377–392. Springer (2016)
20. Mannhardt, F., De Leoni, M., Reijers, H.A., van der Aalst, W.M.P.: Data-driven process discovery - revealing conditional infrequent behavior from event logs. In: Adv. Inf. Systems Engineering, pp. 545–560. Springer (2017)
21. Marquez-Chamorro, A.E., Revoredo, K., Resinas, M., Del-Rio-Ortega, A., Santoro, F.M., Ruiz-Cortes, A.: Context-aware process performance indicator prediction. *IEEE Access* **8**, 222050–222063 (2020)
22. Rosemann, M., Recker, J., Flender, C.: Contextualisation of business processes. *Inter. J. of Business Process Integration and Management* **3**(1), 47–60 (2008)
23. Senderovich, A., Di Francescomarino, C., Maggi, F.M.: From knowledge-driven to data-driven inter-case feature encoding in predictive process monitoring. *Inf. Systems* **84**, 255–264 (2019)
24. Senderovich, A., Weidlich, M., Gal, A.: Context-aware temporal network representation of event logs: Model and methods for process performance analysis. *Inf. Systems* **84**, 240–254 (2019)
25. Shraga, R., Gal, A., Schumacher, D., Senderovich, A., Weidlich, M.: Inductive context-aware process discovery. In: 2019 1st ICPM. IEEE (2019)