Intervening With Confidence: Conformal Prescriptive Monitoring of Business Processes

Mahmoud Shoush[1], Marlon Dumas[1]

[1]University of Tartu, Narva mnt 18, 51009 Tartu, Estonia

Abstract

Prescriptive process monitoring methods seek to improve the performance of a process by selectively triggering interventions at runtime (e.g., offering a discount to a customer) to increase the probability of a desired case outcome (e.g., a customer making a purchase). The backbone of a prescriptive process monitoring method is an intervention policy, which determines for which cases and when an intervention should be executed. Existing methods rely on predictive models to define intervention policies; specifically, they consider policies that trigger an intervention when the probability of a negative outcome exceeds a threshold. However, the probabilities computed by a predictive model often come with low confidence, leading to unnecessary interventions and wasted effort, which is problematic when the resources available to execute interventions are limited. To tackle this shortcoming, this paper outlines an approach to extend existing prescriptive process monitoring methods with conformal predictions, i.e., predictions with confidence guarantees. A preliminary evaluation using real-life public datasets shows that conformal predictions enhance the net gain of prescriptive process monitoring methods under limited resources.

Keywords

Prescriptive Process Monitoring, Conformal Prediction, Causal Inference,

1. Introduction

Prescriptive process monitoring (PrPM) is a family of methods to optimize business processes by triggering runtime interventions with the goal of improving the percentage of cases that lead to a desired outcome [1]. For example, in a lead-to-order process, a PrPM method may recommend offering a discount (the intervention) to achieve sales (desired outcome). In contrast, in an unemployment benefits assessment process, a PrPM system may allocate a problematic case to a senior case handler (intervention) to avoid an appeal (undesired outcome).

Existing PrPM approaches [2, 3, 4, 5, 6] typically comprise at least two components: (i) a *predictive model* that estimates the probabilities that a case ends in a desired (d_{out}) or an undesired outcome (u_{out}); and (ii) an *intervention policy*, which determines for which ongoing cases an intervention should be triggered in view of optimizing a gain function. This gain function considers the benefit of more cases ending with a desired outcome (e.g., higher revenue from more cases ending in a sale) and the cost of the interventions (e.g., the discounts).

Typically, existing PrPM approaches rely on policies that trigger interventions when the

PMAI23@IJCAI: 2nd International Workshop on Process Management in the AI era, August 19, 2023, Macao, China

mahmoud.shoush@ut.ee (M. Shoush); marlon.dumas@ut.ee (M. Dumas)

https://github.com/mshoush (M. Shoush); https://kodu.ut.ee/~dumas/ (M. Dumas)

0000-0002-7423-9909 (M. Shoush); 0000-0002-9247-7476 (M. Dumas)

CEUR Workshop Proceedings (CEUR-WS.org)

probability of a negative outcome exceeds a threshold [7, 5]. A shortcoming of this approach is that the probabilities computed by predictive models often come with low confidence. This leads to unnecessary interventions and, thus, wasted effort. This wasted effort is particularly problematic in settings where the resources available to execute interventions are limited, which means that allocating a resource to intervene in a case (based on a low-confidence probability) may result in this resource being unable to intervene in other cases.

This paper addresses the above shortcoming by outlining an approach to integrate conformal prediction methods [8] into a PrPM system. Conformal prediction methods allow us to associate confidence guarantees with predictions, thus tackling the abovementioned shortcoming regarding triggering unnecessary interventions. The paper reports on an empirical evaluation to test the hypothesis that the use of conformal predictions leads to a higher net gain from interventions in a resource-constrained PrPM system.

2. Related Work

PrPM techniques can be classified into three groups based on intervention policy and improving business value [9]. The first group focuses on control flow for optimal action recommendations [10, 11, 4]. The second group prioritizes resource allocation decisions [12, 13]. The third group combines control flow and resources to mitigate undesired outcomes [14, 6, 2, 5]. This paper falls into the third group.

Studies in the third group use predictive models trained on historical process data (*event logs*) to determine when and for which cases interventions should be triggered. Fahrenkrog et al. [5] propose a PrPM approach based on predictions from an outcome-oriented model [15]. These methods trigger interventions if the probability of an undesired outcome exceeds a threshold. The threshold is determined through empirical thresholding, which explores multiple thresholds over a subset of the event log to maximize a reward function. However, these techniques overlook the inherent uncertainty in prediction models.

Metzger et al. [2] propose using reliability estimates, prediction scores, and other features in an online RL method. However, their reliability estimates lack confidence guarantees, and their black-box policy learned through neural networks lacks explainability. Additionally, their approach involves online RL, while we focus on offline policy discovery based on past data.

In previous work [14, 6], we presented a PrPM technique that considers the tradeoff between triggering an intervention now versus later when resources are limited. This technique relies on estimates including the intervention effect (or conditional average treatment effect, i.e., $CATE$), *total uncertainty* (determined as the entropy of the average prediction from an ensemble of machine learning (ML) classifiers), and the probability of undesired outcomes. However, these uncertainty estimates do not come with confidence guarantees. This latter approach is used as a baseline in the empirical evaluation reported later in this paper.

3. Approach

In line with existing ML approaches, the method consists of three phases, as illustrated in Fig. 1: Training, Calibration, and Testing, which we discuss below in turn.

3.1. Training phase

During training, the event log is used to train predictive and causal models after data cleaning and enrichment. The predictive model estimates the probability of an ongoing case resulting in an undesired outcome ($\mathbb{P}(u_{out})$). In contrast, the causal model measures the effect of triggering an intervention on the probability of a positive outcome ($CATE$). The training process is described in our previous work [14, 6] and is summarized below.

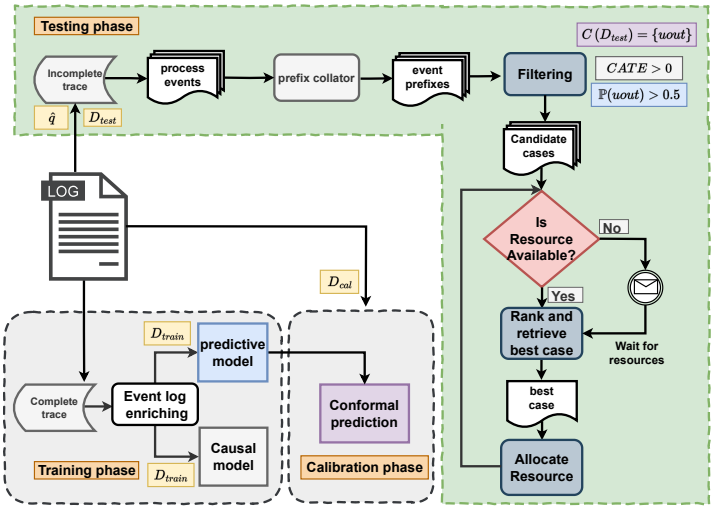

Figure 1: An overview of the proposed approach.

3.1.1. Event Log Enrichment

This step includes *data preparation, prefix extraction, enrichment, and encoding*. In data preparation, we clean the event log by removing incomplete traces and outliers (e.g., events with abnormal timestamp values). We extract prefixes of length K from each case to simulate real-life scenarios. The prefixes are enriched with attributes related to temporal context and inter-case information. Finally, the prefixes are encoded into a fixed-size feature vector using an aggregate encoding method [15] for training machine learning algorithms. The output is a preprocessed dataset containing tuples ((X_i, T_i, Y_i)), each consisting of a feature vector X_i (original and enriched features), and an intervention T_i that can positively impact the outcome Y_i. The dataset is then divided into three folds: $D_{train}, D_{cal}, D_{test}$ with $N = n_{train} + n_{cal} + n_{test}$ samples. Each fold is used in the training, calibration, and testing phases, respectively.

3.1.2. Predictive Model

The predictive model aims to estimate the probability of an ongoing case ending in an undesired outcome based on its corresponding prefix. To train the predictive model, a gradient-boosted tree algorithm is applied to the training fold D_{train}. The objective is to minimize a loss function $\mathcal{L}(Y, \hat{Y})$, where Y represents the actual outcome, and $\hat{Y}$ represents the predicted outcome. The result is a predictive model ($\hat{f}$) that generates a prediction score (probability) for both the undesired outcome, $\mathbb{P}(u_{out})$, and the desired outcome, $\mathbb{P}(d_{out})$.

3.1.3. Causal Model

The causal model determines the impact of the intervention, i.e., $CATE$. It represents the percentage increase in the probability of achieving the desired outcome when the intervention is applied. For example, in a lead-to-order process with an initial sales probability of 0.4, an $CATE$ of 0.3 indicates that the intervention would raise the sales probability to 0.7. To estimate the $CATE$, a causal model is trained to predict the probabilities of undesired outcomes with and without the intervention ($T = 1$ and $T = 0$). The difference between these probabilities, considering the current case state characterized by X, provides the $CATE$.

3.2. Calibration phase

In this phase (Fig. 2), we use an Inductive Conformal Prediction (ICP) algorithm [16, 8, 17]. ICP methods can be applied as a post-processing step to any predictive model, such as random forests or gradient-boosting, to provide predictions with confidence guarantees.

The ICP method uses a user-defined significance level (α) and a predictive model ($\hat{f}$) to create a prediction set (C that contains the actual outcome with a confidence level of $1 - \alpha$. For example, if the user desires a confidence level of 90%, then they would set α to 0.1. This α value isn't for hyperparameter optimization in total gain but reflects the user's preferred level of conservatism, indicating their willingness to act on less certain predictions.

Reducing the significance level increases confidence but also enlarges the prediction set to encompass all possible outcomes. In our context, we aim to create prediction sets exclusively containing only the undesired outcome to ensure high certainty that a case will end undesirably before allocating costly resources. Accordingly, we adopt a conservative approach for *risk-averse* users, triggering interventions only when we're highly confident of the undesired outcome. This leads to prediction sets consisting solely of the undesired outcome. In contrast, *risk-prone* users may opt for intervention even when positive outcomes are possible without it.

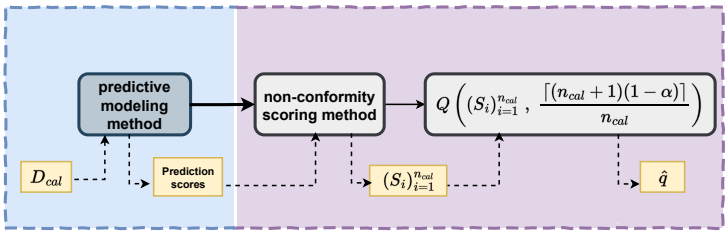

Figure 2: The inductive conformal prediction method.

An ICP method consists of two steps, as shown in Fig. 2. In the first step, non-conformity scores (S) and a non-conformity quantile ($\hat{q}$) are calculated. The predictive model assigns outcome probabilities (prediction scores) to the calibration data, and a *non-conformity scoring* method generates non-conformity scores $s \in (S_i)_{i=1}^{n_{cal}}$ for each sample. Higher non-conformity scores indicate greater uncertainty. The non-conformity quantile $\hat{q}$ is determined based on the significance level (α) as per Eq. 1.

$$\hat{q} = Q\left((S_i)_{i=1}^{n_{cal}}, \frac{\lceil (n_{cal}+1)(1-\alpha)\rceil}{n_{cal}}\right) \tag{1}$$

In the second step, the value of $\hat{q}$ determines the outcomes included in the prediction set. Using the marginal coverage guarantee property [18]. ICP generates a prediction set with $1-\alpha$ confidence. This property ensures that the actual outcome Y_{test} will be included in the prediction set $C(X_{test})$ with $1-\alpha$ confidence. A higher confidence level increases the size of the prediction set to accommodate all possible outcomes. In outcome-oriented PrPM tasks, the focus is on identifying $C(X_{test}) = \{u_{out}\}$ with greater certainty to allocate resources efficiently. For example, if the desired outcome is the actual outcome and $\alpha = 0.2$, Eq. 2 guarantees that d_{out} will be included in C with at least 80% confidence. Lower α values indicate higher confidence but also result in larger prediction sets. In a PrPM task, the main goal is to confidently identify $C(X_{test}) = \{u_{out}\}$, indicating a higher certainty of an undesirable outcome.

$$\mathbb{P}(Y_{test} \in C(X_{test})) \geq 1-\alpha \tag{2}$$

ICP methods differ in how they calculate the non-conformity score and how they use the non-conformity quantile $\hat{q}$ to determine the prediction set. Below, we describe the specific ICP methods we employ in our approach.

3.2.1. Naive method

Fundamentally, in the outcome-oriented task, the predictive model approximates $\mathbb{P}(Y = out \mid X = x)\ \forall\ out \in \{d_{out}, u_{out}\}$. For example, given an instance of a given case x, what is the probability of it belonging to out? Then we perform a naive calibration step by setting the non-conformity score (S_n) to be one minus the prediction score of the actual outcome, as shown in Eq. 3, to obtain $\{(s_i)\}_{i=1}^{n_{cal}}$. Then calculate $\hat{q}$ according to Eq. 1.

$$S_n = 1 - \hat{f}(X_{cal})_{out_{true}} \quad \forall \quad X_{cal} \in D_{cal} \tag{3}$$

$$C_n(X_{test}) = \{out : \hat{f}(X_{test})_{out} \geq 1-\hat{q}\} \tag{4}$$

Then, the prediction set is constructed based on Eq. 4, where X_{test} is known, but Y_{test} is not. This means the prediction set will only include one outcome, desired or undesired, when the prediction score for one outcome satisfies the condition in Eq. 4., and the other outcome does not. For example, when $\hat{q} = 0.7$, the $\mathbb{P}(u_{out}) = 0.72$, and the $\mathbb{P}(d_{out}) = 0.28$, then the $C(X_{test}) = \{u_{out}\}$. Otherwise, the level of certainty about the prediction becomes insufficient to retain only one outcome and either include both or none.

3.2.2. Outcome-balanced method

This scoring (S_{ob}) method's principle is the same as the former; however, here, we perform the calibration step for each outcome separately to achieve outcome-balanced coverage, especially when the outcome of cases is imbalanced; thus, it guarantees (5) instead of (2). Hence, defining the non-conformity scores and non-conformity quantile for each outcome, as shown in Eq. 5., means we stratify by the outcome.

$$\mathbb{P}(Y_{test} \in C(X_{test}) \mid Y_{test} = out) \geq 1-\alpha,\ \forall\ out \in \{d_{out}, u_{out}\} \tag{5}$$

$$\hat{q}^{(out)} = Q\left((S_i^{(out)})_{i=1}^{n_{cal}(out)}, \ \frac{\lceil (n_{cal}(out)+1)(1-\alpha) \rceil}{n_{cal}(out)}\right) \tag{6}$$

According to the outcome-balanced scoring method, the prediction set is determined by Eq. 7., where we iterate over desired and undesired outcomes. Then it retains or not each outcome according to its quantiles. For example, assume $\hat{q}^{(d_{out})} = 0.7$, $\hat{q}^{(u_{out})} = 0.4$, the $\mathbb{P}(u_{out}) = 0.3$, and the $\mathbb{P}(d_{out}) = 0.7$. Then the prediction set examines each outcome with its prediction score and $\hat{q}$. Hence, the $\mathbb{P}(u_{out}) = 0.3$ is not greater than 1 minus 0.4; accordingly, the prediction set will discard the undesired outcome. Conversely, the $\mathbb{P}(d_{out}) = 0.7$ is greater than 1 minus 0.7; thus, the prediction set will retain the desired outcome, meaning $C(X_{test}) = \{d_{out}\}$ only.

$$C_{ob}(X_{test}) = \{out : \hat{f}(X_{test})_{out} \geq 1 - \hat{q}^{(out)}\} \tag{7}$$

3.2.3. Adaptive method

Unlike previous methods (S_n and S_{ob}) that consider only the prediction score for the actual outcome, this scoring method (S_a) considers all possible outcomes until the sum of their prediction scores exceeds the $1 - \alpha$ confidence. Eq. 8., shows how the non-conformity scores are calculated, where $\pi(x)$ is the permutation of all possible outcomes that orders $\hat{f}(X_{test})$ from the most likely outcome to the less likely. The next step is to compute $\hat{q}$ as (2), and the prediction set is formed according to Eq. 9.

$$S_a = \sum_{i=1}^{out} \pi(X_{cal})_i \tag{8}$$

$$C_a(X_{test}) = \{out : S_a \geq \hat{q}\} \tag{9}$$

Based on this scoring method, there is no empty prediction set because the prediction set will retain only one outcome when the level of certainty about it is high. Otherwise, it will retain both outcomes but with different orders. Specifically, we add outcomes one by one to the prediction set until the sum of their prediction score exceeds the $\hat{q}$. For example, assume $\hat{q} = 0.8$, $\mathbb{P}(u_{out}) = 0.45$, and the $\mathbb{P}(d_{out}) = 0.55$. We first sort the prediction scores from the most likely to the least, e.g., $\mathbb{P}(d_{out}) = 0.55$, followed by $\mathbb{P}(u_{out}) = 0.45$. Then we add the most likely outcome to the prediction set if its prediction score does not exceed $\hat{q} = 0.8$, meaning $\mathbb{P}(d_{out}) = 0.55 < 0.8$. Next, we sum the next outcome in order to the previous one, and if their sum does not exceed the $\hat{q} = 0.8$ will include it; otherwise, we stop and not adding any other outcomes to the prediction set. Since $0.45 + 0.55$ is greater than $\hat{q} = 0.8$, the prediction set will not include the second outcome in the prediction set; thus $C(X_{test}) = \{d_{out}\}$.

3.3. Testing phase

At runtime, the approach collates events for ongoing cases into case prefixes using a *prefix collator*, resulting in a stream of trace prefixes. For each incoming trace prefix (X_{test}), estimates of $\mathbb{P}(u_{out})$, $CATE$, and $C(X_{test})$ are obtained. These estimates are then used to *filter* ongoing

cases, identify intervention candidates, and *rank* them based on a gain function that considers the benefits of achieving desired outcomes and the costs of interventions.

To identify candidate cases for intervention, we check three conditions: (1) $\mathbb{P}(u_{out})$ is above a threshold, determined empirically, (2) $C(X_{test}) = \{uout\}$, and (3) $CATE > 0$. The candidate case with the highest gain is chosen, calculated as the benefit of avoiding an undesired outcome ($C_{u_{out}}$ multiplied by $CATE$, minus the intervention cost C_{in}, see Eq. 10.

$$gain = CATE * C_{u_{out}} - C_{in} \tag{10}$$

The parameters $C_{u_{out}}$ and C_{in} are user-defined and can vary between different processes. Tab. 1 provides an example of costs and gains for six case prefixes in an unemployment benefits process. The undesired outcome is when the customer lodges an appeal. Different decisions are made depending on the cost of creating an appeal and giving a discount. For example, in $CaseID = C$, giving a discount is preferred when its cost is lower than creating an appeal, while in $CaseID = E$, accepting the appeal is preferred.

Table 1
An example of costs and gains.

$CaseID$	$\mathbb{P}(uout) > \tau_{=0.5}$	$CATE$	$C(X_{test})$	C_{uout}	C_{in}	$gain$
A	0.52	5	{uout}	6	6	24
B	0.54	-1	{dout}	-	-	-
C	0.7	6	{uout}	10	5	55
D	0.7	3	{}	-	-	-
E	0.55	3	{uout}	2	12	-6
F	0.76	4	{uout}	10	5	35

Also, in Tab. 1, we have six cases with different $\mathbb{P}(u_{out})$, $CATE$ and $C(X_{test})$. $CaseID = B$ and $CaseID = D$ are excluded due to their negative intervention effects and empty prediction sets. With only one available resource for a phone call, we allocate it to the case with the highest gain, which is $CaseID = C$.

4. Evaluation

We report on an evaluation that addresses the following questions:

RQ1. What significance level (α) is appropriate for each non-conformity scoring method to align with the preferences of a risk-averse user?

RQ2. To what extent does conformal prediction improve the total gain w.r.t. existing baselines?

4.1. Datasets

We experimented with two real-life event logs from the banking industry: *BPIC2017*[1] *and BPIC2012*[2]. These logs represent the loan origination process and provide clear definitions for desired and undesired outcomes. They are large enough regarding the number of loan applications and include interventions that can reduce the probability of undesired outcomes. Table 2 provides an overview of the key characteristics of these logs.

[1]https://doi.org/10.4121/uuid:5f3067df-f10b-45da-b98b-86ae4c7a310b
[2]https://data.4tu.nl/articles/dataset/BPI_Challenge_2012/12689204/1

Table 2
Descriptive statistics of the loan application dataset.

dataset	# applications	min length	max length	last activity	outcome	intervention activity
BPIC2017	31,413	10	180	A_pending	desired	-
				A_Canceled A_Declnied"	undesired	Creat_Offer
BPIC2012	13,087	15	175	A_Approved	desired	-
				A_Canceled A_Declnied	undesired	Creat_Offer

The logs contain diverse case and event attributes. We use them in our experiments in addition to other extracted attributes, e.g., the number of sent offers, monthly loan interest, and temporal features, to enrich the logs. Then, we define outcomes according to each case's last activity and determine the intervention according to the *Creat_Offer* activity for cases labeled with undesired outcomes, as shown in Tab. 2. To avoid lengthy cases, we extract prefixes up to the 90th percentile. An *aggregate encoding* method is applied to capture maximum information from the logs, outperforming other techniques, as shown in previous research [15]. The resulting fixed-size feature vector serves as input for training the machine learning algorithms.

4.2. Experimental Setup

The experimental setup involves dividing the log into three categories: training (60%), calibration (20%), and testing (20%). The training set is used for model training, the calibration set is used to create the prediction set, and the testing set evaluates the intervention policy.

We use *Catboost* [19], a GBDT algorithm, to train the predictive model for estimating the probability of undesired outcomes ($\mathbb{P}(u_{out})$). For estimating $CATE$, we employ the *Orthogonal Random Forest* (ORF) algorithm from *EconMl*[3]. Both methods have shown good accuracy in predicting undesired outcomes and estimating intervention effects [15, 14].

During runtime, ongoing cases are filtered to identify candidates based on $\mathbb{P}(u_{out}) > 0.5$, $CATE > 0$, and $C(X_{test}) = \{u_{out}\}$. These estimates help prioritize cases likely to have undesired outcomes and be influenced by the intervention. Then we set the $C_{uout} = 20$, relatively high, to $C_{in} = 1$ to estimate the expected gain from resource allocation.

We evaluate the proposed approach using metrics such as AUC and $F - score$ to assess the ICP methods' performance. These metrics are suitable for imbalanced data and provide an unbiased evaluation. Additionally, we examine the number of cases in $C(X_{test})$ containing only the undesired outcome, targeting confident predictions of undesirable outcomes.

We evaluate the intervention policy with limited resources based on the *total gain* and the *(accuracy/resource)* ratio. The total gain represents the cumulative gains achieved per available resource. In contrast, the accuracy per resource ratio indicates the proportion of correctly allocated resources to undesired cases out of the total allocated cases.

4.3. Results

We analyze the impact of the user-defined significance level (α) on the prediction set (RQ1) and examine the improvement in total gain with finite resources using the intervention policy

[3]https://github.com/microsoft/EconML

based on conformal prediction (RQ2).

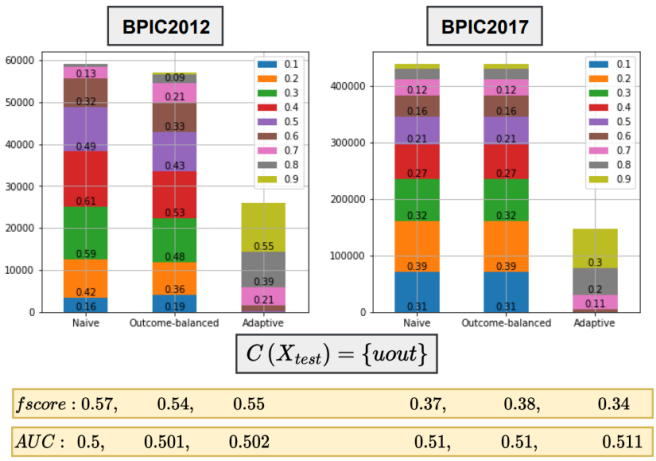

Figure 3: The histograms where $C(X_{test}) = \{uout\}$ and related metrics.

In Fig. 3, we present the impact of different non-conformity scoring methods on the retention of an undesired outcome in the prediction set, addressing RQ1. In other words, for a risk-averse user, we aim to find which significance level maximizes the number of cases in which the prediction set contains only a negative outcome. Our findings indicate that for the *naive* and *outcome-balanced* methods, the optimal significance levels (α) for maximizing the number of cases belonging to the prediction set, while retaining only undesired outcomes, are 0.4 for *BPIC2012* and 0.2 for *BPIC2017*. Conversely, the *adaptive* method achieves the maximum retention at $\alpha = 0.9$ for both logs. This disparity can be attributed to the construction of the prediction set in each method. The naive and outcome-balanced methods demonstrate less conservatism towards including a specific outcome in the prediction set as $\hat{q}$ approaches zero. In contrast, the adaptive method exhibits the opposite behavior. Moreover, we observe that these significance levels yield the highest *F-score* and *(AUC)* compared to other levels (detailed in the *supplementary material*[4]). As a result, for risk-averse users, an extreme alpha value must be selected, and these levels are used to assess the enhancement of conformal methods in terms of total gain and accuracy/resource.

To investigate RQ2, we analyze different approaches for improving the intervention policy. Firstly, we compare pure predictive methods targeting cases with $\mathbb{P}(u_{out}) > 0.5$ [5], with and without a threshold of $TotalUncertainty < 0.75$ [14]. We then evaluate the performance of predictive methods combined with the inductive conformal prediction (ICP), specifically when $C(X_{test})$ includes only undesired outcomes. This analysis is presented in Fig. 4, where the gain from interventions using $CATE$ is examined. Additionally, Fig. 5 compares the predictive approach ($\mathbb{P}(u_{out}) > 0.5$ and the $TotalUncertainty < 0.75$) combined with $CATE$ (when it is above 0) [3] and conformal prediction.

For the *BPIC2012* log, the *total gain* (on the left-hand side) improves when we combine any conformal method with pure predictive in Fig. 4 and $CATE$ in Fig. 5. In particular, when

[4]https://zenodo.org/record/7380386

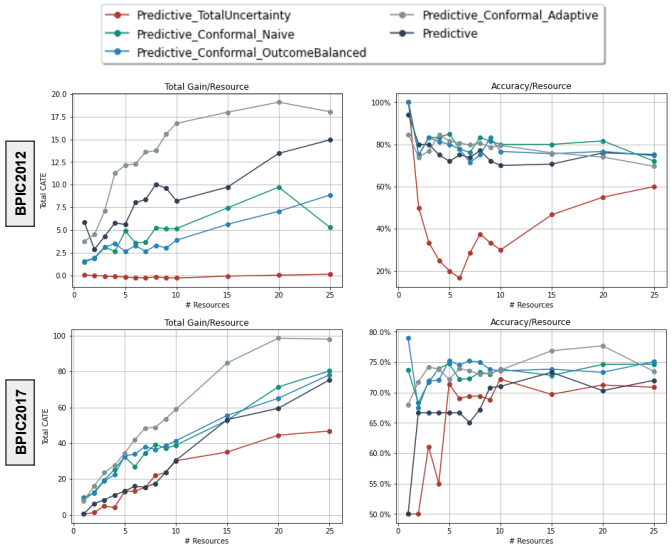

Figure 4: Pure predictive VS predictive plus conformal. X-axis: range of available resources, Y-axis: achieved total gain and accuracy per resource (left and right figures respectively).

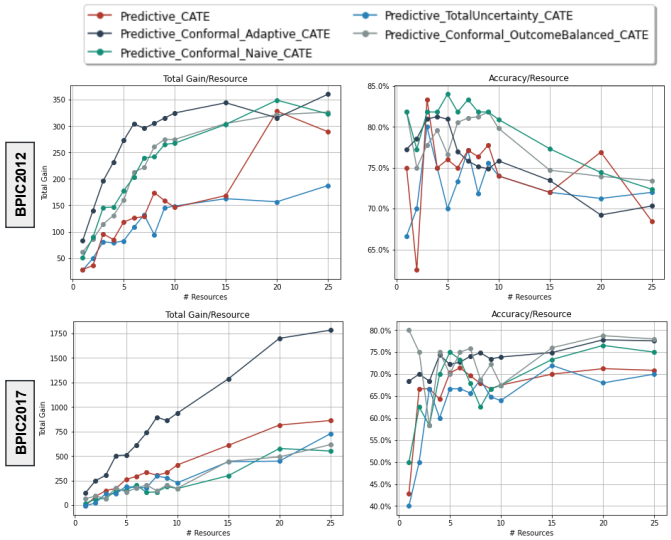

Figure 5: Predictive plus CATE VS Predictive plus CATE plus Conformal. X-axis: range of available resources, Y-axis: achieved total gain and accuracy per resource (left and right figures respectively).

resources are minimal, with a remarkable *accuracy/resource* compared to non-conformal methods. Also, the adaptive conformal method outperforms other methods w.r.t the total gain, and similar to other methods, w.r.t accuracy/resource. This is because the adaptive method's defined $\hat{q}$ is much higher than the naive and outcome-balanced methods; accordingly, more conservative in adding outcomes to the prediction set.

Moreover, when resources are not restricted, which is different from the situation in practice,

we find that non-conformal methods achieve good gains with reasonable accuracy per resource as conformal methods. However, the conformal methods are more conservative since they constrain the allocation of resources.

For the *BPIC2017* log, the adaptive method significantly improves the *total gain* with high accuracy in both limited and relaxed resource scenarios. In contrast, when resources are limited, the naive and outcome-balanced methods achieve comparable gains to non-conformal methods. Nevertheless, all conformal methods outperform non-conformal w.r.t accuracy per resource.

In summary, the proposed PrPM approach demonstrates superior performance compared to baselines regarding *total gain* and accuracy per resource, as shown in Fig. 4 and Fig. 5. Moreover, our approach outperforms the previous work [14], as indicated in the supplementary material. The use of conformal prediction to construct an intervention policy with limited resources further enhances the performance of PrPM methods, benefiting business processes.

5. Conclusion and Future Work

We studied the hypothesis that the use of conformal predictions can enhance the effectiveness of prescriptive process monitoring methods by preventing interventions from being triggered unnecessarily when the level of confidence is insufficient.

The empirical evaluation shows that intervention policies with conformal predictions outperform classic non-conformal methods, particularly when the number of resources available for performing the interventions is limited. The reported evaluation relied on two real-life event logs from the same domain (banking). We acknowledge that further experiments with a larger and more diverse array of datasets are required to achieve generalizability.

The proposal assumes that only one type of intervention is available (e.g., giving a customer discount). Also, it assumes that this intervention can only be triggered at most once in a case. In practice, cases may be subject to multiple interventions of different types (e.g., giving a discount, offering an upgrade, or a voucher for future purchases, etc.). Thus, a direction for future work is to extend the current approach to a multi-intervention setting, for example, using multi-armed bandit approaches. Another direction is to study the problem where the case outcome is not a categorical variable (e.g., positive vs. negative) but a numerical variable (e.g., cost, time).

In this paper, we've employed conformal prediction techniques to identify cases likely to result in a negative outcome. However, an intriguing avenue for further exploration involves applying conformal prediction to the *CATE* values themselves, thereby enhancing the overall prediction and decision-making process. Furthermore, we could expand upon this work by exploring the application of reinforcement learning, both with and without conformal methods, as an alternative to the rule-based approach for learning intervention policies.

Reproducibility. The source code required to reproduce the experiments can be found at: https://github.com/mshoush/conformal-prescriptive-monitoring.

Acknowledgments

This research is supported by the European Research Council (PIX Project).

References

[1] S. Athey, Beyond prediction: Using big data for policy problems, Science 355 (2017) 483–485.

[2] A. Metzger, T. Kley, A. Palm, Triggering proactive business process adaptations via online reinforcement learning, in: BPM, Springer, 2020, pp. 273–290.

[3] Z. D. Bozorgi, I. Teinemaa, M. Dumas, M. La Rosa, Prescriptive process monitoring for cost-aware cycle time reduction, in: ICPM, IEEE, 2021.

[4] M. de Leoni, M. Dees, L. Reulink, Design and evaluation of a process-aware recommender system based on prescriptive analytics, in: ICPM, IEEE, 2020.

[5] S. A. Fahrenkrog-Petersen, N. Tax, I. Teinemaa, M. Dumas, M. de Leoni, F. M. Maggi, M. Weidlich, Fire now, fire later: alarm-based systems for prescriptive process monitoring, Knowl. Inf. Syst. 64 (2022) 559–587.

[6] M. Shoush, M. Dumas, Prescriptive process monitoring under resource constraints: A causal inference approach, in: ICPM Workshops, Lect. Notes Bus. Inf. Process., Springer, 2021.

[7] I. Teinemaa, N. Tax, M. de Leoni, M. Dumas, F. M. Maggi, Alarm-based prescriptive process monitoring, in: BPM (Forum), Lect. Notes Bus. Inf. Process., Springer, 2018.

[8] G. Shafer, V. Vovk, A tutorial on conformal prediction, J. Mach. Learn. Res. 9 (2008) 371–421.

[9] K. Kubrak, F. Milani, A. Nolte, M. Dumas, Prescriptive process monitoring: *Quo vadis*?, PeerJ Comput. Sci. 8 (2022) e1097.

[10] S. Weinzierl, S. Dunzer, S. Zilker, M. Matzner, Prescriptive business process monitoring for recommending next best actions, in: BPM (Forum), volume 392 of *Lecture Notes in Business Information Processing*, Springer, 2020, pp. 193–209.

[11] P. Agarwal, A. Gupta, R. Sindhgatta, S. Dechu, Goal-oriented next best activity recommendation using reinforcement learning, CoRR abs/2205.03219 (2022).

[12] R. Sindhgatta, A. K. Ghose, H. K. Dam, Context-aware analysis of past process executions to aid resource allocation decisions, in: CAiSE, volume 9694 of *Lecture Notes in Computer Science*, Springer, 2016, pp. 575–589.

[13] G. Park, M. Song, Prediction-based resource allocation using LSTM and minimum cost and maximum flow algorithm, in: ICPM, IEEE, 2019, pp. 121–128.

[14] M. Shoush, M. Dumas, When to intervene? prescriptive process monitoring under uncertainty and resource constraints, in: BPM (Forum), 2022.

[15] I. Teinemaa, M. Dumas, M. L. Rosa, F. M. Maggi, Outcome-oriented predictive process monitoring: Review and benchmark, ACM Trans. Knowl. Discov. Data 13 (2019) 17:1–17:57.

[16] G. Zeni, M. Fontana, S. Vantini, Conformal prediction: a unified review of theory and new challenges, CoRR abs/2005.07972 (2020).

[17] R. J. Tibshirani, R. F. Barber, E. J. Candès, A. Ramdas, Conformal prediction under covariate shift, in: NeurIPS, 2019, pp. 2526–2536.

[18] V. Vovk, A. Gammerman, C. Saunders, Machine-learning applications of algorithmic randomness, in: ICML, Morgan Kaufmann, 1999, pp. 444–453.

[19] L. O. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, A. Gulin, Catboost: unbiased boosting with categorical features, in: NeurIPS, 2018.