Enriching Decision Making with Data-Based Thresholds of Process-Related KPIs

Adela del-Río-Ortega[1(✉)], Félix García[2], Manuel Resinas[1], Elmar Weber[3], Francisco Ruiz[2], and Antonio Ruiz-Cortés[1]

[1] Universidad de Sevilla, Seville, Spain
adeladelrio@us.es
[2] Universidad de Castilla La Mancha, Ciudad Real, Spain
[3] Cupenya, B.V., Amsterdam, The Netherlands

Abstract. The continuous performance improvement of business processes usually involves the definition of a set of process performance indicators (PPIs) with their target values. These PPIs can be classified into lag PPIs, which establish a goal that the organization is trying to achieve, though are not directly influenceable by process performers, and lead PPIs, which are influenceable by process performers and have a predictable impact on the lag indicator. Determining thresholds for lead PPIs that enable the fulfillment of the related lag PPI is a key task, which is usually done based on the experience and intuition of the process owners. However, the amount and nature of currently available data make it possible for data-driven decisions to be made in this regard. This paper proposes a method that applies statistical techniques for thresholds determination successfully employed in other domains. Its applicability has been evaluated in a real case study, where data from more than a thousand process executions was used.

Keywords: Thresholds · Process-related KPIs · Process performance indicators · Case study · Decision making · Decision support

1 Introduction

In process-oriented organisational settings, the evaluation of process performance plays a key role in obtaining information on the achievement of their strategic and operational goals. To carry out this evaluation, a performance measurement system (PMS) is implemented, so that business processes (BPs) can be continuously improved [1]. The implementation of this PMS includes the definition of

This work has received funding from the European Commission (FEDER), the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No. 645751 (RISE_BPM), Spanish, Andalusian and Castilla La Mancha R&D&I programmes (grants P12–TIC-1867 (COPAS), TIN2015-70560-R (BELI) and PEII-2014-050-P (INGENIOSO)).

E. Dubois and K. Pohl (Eds.): CAiSE 2017, LNCS 10253, pp. 193–209, 2017.
DOI: 10.1007/978-3-319-59536-8_13

a set of PPIs, their target values, and associated alarms that warn whenever certain predetermined value, named threshold [2], is exceeded [3]. These PPIs are quantifiable metrics that allow the efficiency and effectiveness of BPs to be evaluated and can be computed directly from data generated during their execution, either at an instance level (single-instance PPIs) or at a process level, i.e., computed applying certain functions to the execution data gathered from a set of instances (multi-instance PPIs) [4].

Based on these PPIs, several methodologies have been developed to continuously improve the process performance. One of the best known is based on the concept of lag and lead indicators [5]. Performance indicators defined for a business process can be broadly classified into two categories, namely: lag and lead indicators, also known as outcomes and performance drivers respectively. The former establishes a goal that the organization is trying to achieve and is usually linked to a critical success factor. For instance, one could have a PPI for a manuscript management process that specifies that its cycle time should be less than 40 working days in order to keep customer (author) satisfaction. However, the problem of lag indicators is that they tell the organization whether the goal has been achieved, but they are not directly influenceable by the performers of the process. On the contrary, lead indicators have two main characteristics. First, they are predictive in the sense that if the lead indicators are achieved, then it is likely the lag indicator is achieved as well. Second, they are influenceable by the performers of the process, meaning that they should be something that the performers of the process can actively do or not do. For instance, if we think that one major issue that prevents fulfilling the lag indicator is assigning the manuscript to an employee with a large queue of work and we know we can control the queue of work of each employee up to a certain point (e.g. by balancing the work amongst all employees), then reducing the workload could be a lead indicator for the cycle time lag indicator defined above. Each lag indicator may have one or more lead indicators that are influenceable by the process performers and help to predict its value. Therefore, if thresholds are established for those lead indicators, the focus will be on fulfilling lead indicators, which are actionable, and this will enable the fulfillment of the lag indicator.

Determining these thresholds appropriately is, thus, one of the key parts of the methodology. This is usually done based on the experience and intuition of the process owners. However, nowadays, the amount and nature of available data (e.g. event logs) make it possible for data-driven decisions to be made in this regard. Unfortunately, although a number of works to identify relationships between process characteristics and PPIs have been proposed in the last years, e.g. [4,6–8], the identification of proper thresholds for a PPI (lead) in order to support the achievement of another PPI (lag) has not been tackled up to date.

The goal of the presented research is to provide a method to determine the aforementioned thresholds, focusing on single-instance PPIs. To this end, we build on a set of statistical techniques successfully used in other domains for threshold determination [9–11]. In particular, we propose the use of Receiver Operating Characteristic (ROC) curves and the Bender method. While the

former allows the pursued thresholds to be determined, the latter provides ranges of values with the associated probabilities of fulfilling the target value. This information is specially useful when the changes required to reach the identified threshold cannot be implemented, since it gives hints on the risk taken.

In order to evaluate this approach, we have performed a case study in the context of the manuscript management process of an international publishing company. In this case, data from the execution of more than a thousand of instances of the selected business process are used to study the relationship between the workload, a lead indicator that measures how busy an employee is, and its cycle time, which is a lag indicator of the process. In this scenario, not only a threshold for the workload is identified, which allows for the achievement of the cycle time target value established. Furthermore, if the actions required to keep the workload under that threshold are not possible, our method provides information about the probabilities of achieving the cycle time target value depending on the range the workload value is located in.

The remainder of this paper is structured as follows. Section 2 discusses related work on both the problem and solution domains. Section 3 describes the method for threshold determination we propose. In Sect. 4, this method is applied in a case study to validate its usefulness. Finally, we conclude the paper and discuss future research directions in Sect. 5.

2 Related Work

This section describes previous research related to the work presented in this paper. Two main streams can be distinguished. One is focused on the problem domain and includes techniques developed to identify relationships between performance indicators. The other is focused on the solution domain and comments on some proposals for the definition of measures and associated thresholds.

2.1 Proposals for the Identification and Definition of PPI Relationships

Concerning the problem domain related research stream, there are a number of proposals that are focused on establishing relationships between PPIs. In particular, within the performance measurement context, there are some works that use different techniques, including correlation analysis or principal component analysis [6,12], for this purpose. In the context of process performance evaluation, there also exist some approaches to define relationships between PPIs such as: Popova and Sharpanskykh [7], where a variant of the first order sorted predicate language is employed to define cause, correlation or aggregation relationships; del-Río-Ortega et al. [4], which extracts PPI relationships with BP elements from their definition through description logic; Diamantini et al. [13], that allows for the explicit definition of algebraic relationships between PPIs using semantic techniques, or de Leoni et al. [8], who use decision and regression trees to correlate process or event characteristics. In addition, other approaches

[6,14] have been presented to quantify these relationships in magnitude and direction, providing information to determine their importance depending on whether the relationships are weak or strong. Although these works provide mechanisms to define and, somehow, quantify relationships between PPIs, none of them allow for the extraction of thresholds for PPIs from execution data. Our approach can be seen, therefore, as complementary to these previous works. Based on the PPI relationships identified with them, and given some objective to fulfill, our approach can provide thresholds for the influencing PPIs that lead to the achievement of the objective.

2.2 Thresholds Definition Proposals

Measurement of business processes is a vast research area and, in related literature, we can find numerous definitions of measures which support business process evaluation from both perspectives: modelling [15,16] and execution [4,7,17]. However, to facilitate a better decision making process from the assessment of the measurement results, it is necessary the specification of limit values or thresholds which indicate whether or not the measurement results are acceptable.

In this context, the research on thresholds associated with business process measures is more limited. Traditionally, the definition of thresholds has been applied in other disciplines such as medicine [9]. On the other hand, in the software engineering area, we can find several proposals mainly focused on measures for object oriented systems [10,11,18]. Several techniques are used for that purpose, including the mean and standard deviation, Bender Method, ROC curves, Linear Regression, clustering algorithms (k-means) and machine learning based methods.

From the business process modelling perspective, the application of techniques for threshold definition has been applied in [2,19,20]. In these works, thresholds for understandability, modifiability and correctness measures of BPMN models are extracted. To do so, Bender method, ROC curves and a new algorithm based on ANOVA called ATEMA are applied. In addition, the application of extracted thresholds to suggest improvement guidelines for business process models in a case study is presented in [21]. This research constitutes the background of the present work, which aims to apply the same threshold extraction techniques in the context of business process execution. The thresholds in this case are extracted from execution data and are aimed at assuring the fulfillment of a given PPI. To the best of our knowledge, there exists no previous work in this direction.

3 Threshold Determination Method

The method we propose is based on the concept of lag and lead indicators [5].

Specifically, our method takes as input the lag PPI, the set of performance indicators that, according to the knowledge of domain experts, can be considered

lead PPIs for that particular lag PPI and the values for those lag and lead PPIs computed from a set of process executions. This method includes the following steps: (1) preprocessing; (2) checking the relationship; (3) threshold extraction with Roc Curve; (4) application of the Bender Method to determine probabilities of errors for threshold ranges; (5) threshold validation. In the following, we describe these steps, which are performed for a pair lag PPI-lead PPI, and need to be repeated as may times as lead indicators provided as input.

3.1 Preprocessing

This step is twofold: first, some information need to be gathered in the format it will be required by the statistical techniques that will be applied, and second we need to divide our input data set (with the PPI values) into two. Regarding the former, we need to define a Boolean variable that represents the fulfillment of the lag indicator. In particular, for every process instance considered, we assign this variable the value 1 if the lag PPI is fulfilled, and 0 otherwise. We will refer to this variable as $fulfilledLagPPITargetValue$. As for the latter, we need to split our data set into two groups, one group will be used to define the thresholds and the other to validate them.

3.2 Checking the Relationship

The second step is to prove that the values of the lead PPI do actually have an influence on the fulfillment of the lag PPI. Actually, this is a required step for the two techniques we use later on. ROC curves and the Bender method involve a two-step approach. The first step is about estimating the discriminator function, that allows the aforementioned influence to be checked, and the second is the determination of thresholds and the associated probabilities, that will be described in the following steps (Subsects. 3.3 and 3.4).

We utilize *logistic regression* for estimating a discriminator function, in which the p-value should be lower than 0.05 to confirm that an influence exists. Logistic regression is a statistical model for estimating the probability of binary choices [22]. In our case, we are interested in the binary variable defined in the previous step whose range is $\{fulfillment, non-fulfillment\}$. The idea of a logistic regression is that this probability can be represented by the odds. This is the ratio of fulfillment probability divided by probability of non-fulfillment. The logistic regression estimates the odds based on the logit function, which is:

$$logit(p_i) = ln(\frac{p_i}{1-p_i}) = \alpha + \beta_1 x_{1,i} + \ldots + \beta_k x_{k,i}, \quad (1)$$

where α is called the intercept and β_1, β_2, β_3, etc., are called the regression coefficients of independent variables $x_{1,i}$, $x_{2,i}$, $x_{3,i}$ respectively. In our case we only consider one independent variable for every repetition of the steps, which corresponds to the lead PPI under analysis, i.e. $k = 1$, and observations from i business process instances.

3.3 Threshold Extraction with ROC Curve

"Receiver Operating Characteristics (ROC) curves provide a pure index of accuracy by demonstrating the limits of a test's ability to discriminate between alternative states" (fulfillment/non-fulfillment) [23]. In order to define an ROC curve, two variables need to be specified: one binary, which is the previously defined $fulfilledLagPPITargetValue$ variable, whose values correspond to the fulfillment or not fulfillment of the lag PPI target value; and another continuous, which is the estimated fulfillment probability function from the logistic regression of the lead PPI. In a ROC curve, the true positive rate (sensitivity) is plotted in function of the false positive rate (1-specificity). Each point in the ROC curve represents a pair of sensitivity and 1-specificity corresponding to a particular decision threshold, i.e. it represents the classification performance of any potential threshold.

Table 1. Confusion matrix for lead PPI and threshold.

Classified	Actual	
	Fulfillment	Non-fulfillment
Lead PPI $\leq$ threshold	True positives (TP)	False positives (FP)
Lead PPI $>$ threshold	False negatives (FN)	True negatives (TN)

The determination of the best threshold builds on the confusion matrix (Table 1), for which sensitivity and specificity values are calculated as follows: sensitivity = true positive (TP) rate = TP/(TP+FN), specificity = true negative (TN) rate = TN/(FP+TN), where TP is true positives, FN is false negatives, FP is false positives, and TN is true negatives. A TP is found when the assessment of a value of the lead PPI in relation to the threshold indicates that the lag PPI is likely to be fulfilled in that process instance, and that in fact it does have been fulfilled. Something similar, but with the non-fulfillment, happens to the TN, the assessment of a value of the lead indicator in relation to the threshold indicates that the lag indicator is likely to not be fulfilled in that process instance, and that in fact it has not been fulfilled. On the other hand, an FN indicates that the prediction says that for that value of the lead indicator the lag indicator is not fulfilled while indeed it is. Finally, an FP indicates that the process instance is predicted to fulfill the lag indicator and, actually, it has not fulfilled it.

The test performance is assessed using the Area Under the ROC Curve (AUC). AUC is a widely-used measure of performance of classification [24]. It ranges between 0 and 1, and can be used to assess how good threshold values are at discriminating between groups. According to [22], there exist rules of thumb for assessing the discriminative power of the lead indicator based on AUC. An AUC < 0.5 is considered no good, poor if $0.5 \leq AUC < 0.6$, fair if $0.6 \leq AUC < 0.7$, acceptable if $0.7 \leq AUC < 0.8$, excellent if $0.8 \leq AUC < 0.9$ and outstanding if $0.9 \leq AUC < 1$. The standard error or p-value is estimated using a 95% confidence interval. The test checks if the AUC is significantly different from 0.5.

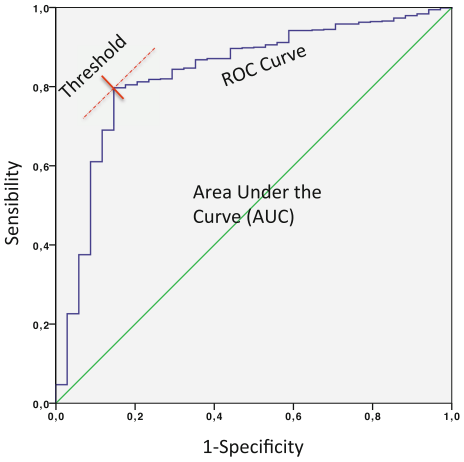

Fig. 1. ROC Curve and threshold.

Then, we can determine a threshold value for the lead PPI based on the ROC curve, but for doing so, wee need a criterion. The purpose is to maximize sensitivity and specificity, while at the same time [22] minimizing false positives and false negatives. Following [2,20], where sensitivity and specificity are considered to be equally important, we select the best threshold as depicted in Fig. 1. The best threshold is the point with the greatest distance from the 0.5 diagonal (that corresponds to a test without any ability to discriminate between the two alternatives).

Due to the involvement of humans in the process execution, one would not expect the same accuracy of predictions as in natural sciences like physics or chemistry [25]. Therefore, it is important to reflect upon the probability of errors associated with this threshold. This probability can be obtained by means of the Bender method as described in the following Subsect. 3.4.

3.4 Application of the Bender Method to Determine Probabilities of Errors for Threshold Ranges

The goal of this step is manifold. First we are interested in determining the probability associated to the threshold obtained in the previous step through the application of ROC curves. In addition, there are situations in which it is not possible to apply the changes required to reach that threshold. In those cases, it is important to provide the decision makers with information about the risk taken accepting other values lower or greater (depending if the threshold is a maximum or a minimum respectively) than the threshold. Therefore, this step also aims at providing other threshold values, or ranges, associated with different probabilities of the lag PPI fulfillment. To this end, the Bender method is applied.

The Bender method [9], developed for quantitative risk assessment in epidemiological studies, assumes that the risk of an event occurring is constant below a specific value (i.e. the threshold), and increases according to a logistic equation otherwise. By defining acceptable levels for the absolute risk, the corresponding benchmark values of the risk factor can be calculated by means of nonlinear functions of the logistic regression coefficients. Generally, a benchmark value is a characteristic point of the dose-response curve at which the risk of an event rises so steeply. The difficulty is to define what is meant by "so steeply". According to [9], one possibility to define benchmark values is based on the logistic curve. A benchmark can initially be defined as the "Value of an Acceptable Risk Level" (VARL) defined as Eq. (2), in which the acceptable risk level is given by a probability p_0.

$$VARL = \frac{1}{\beta}(ln(\frac{p_0}{1-p_0}) - \alpha) \tag{2}$$

$$p_0 = \frac{e^{\alpha+\beta x}}{1+e^{\alpha+\beta x}} \tag{3}$$

In Eq. 2, p_0 represents the probability of an event occurring. This value is indicated by the person who is applying that method and it can vary from 0 to 1. For example, applied to our case, $p_0 = 0.7$ indicates that there is a probability of 0.7 the lead PPI to be considered as appropriate, i.e., to lead to the fulfillment of the lag PPI. On the other hand, α and β are coefficients of a logistic regression equation, as was indicated in Eq. (1). The independent variable in the logistic regression model is the lead PPI for which we want to determine the threshold. The dependent variable must be a binary variable, in our case the $fulfilledLagPPITargetValue$ variable, that evaluates if the lag PPI was fulfilled or not.

We can then use this method to determine the probability associated to the threshold obtained through the application of ROC curves as follows. From the formula of Eq. 2 we can obtain Eq. 3 to calculate that probability, where x is the threshold value previously obtained, and α and β the coefficients also previously obtained. If, for instance, the resulting probability is 0.9, it means that when the lead indicator is lower or equal to the threshold obtained (considering it a maximum), there is a 90% of probability that the lag indicator is fulfilled.

Furthermore, as stated above, we can apply this method to identify other threshold values associated with different probabilities of the target value fulfillment, enriching the information provided to the manager to make a decision. For this purpose, the Bender method requires the definition of p_0, which indicates the probability of considering a BP instance as fulfilling lag indicator. Since there is no recommendation that can be used to configure this variable, we propose 9 values between 0 and 1 with the idea of obtaining a wide group of results. Therefore p_0 starts in 0.1, and 0.1 is added successively until reaching 0.9. Thus, we associate ranges of probability (from 10% to 90%) to different values of the lead PPI (see Table 4 to see the result in our case study).

3.5 Threshold Validation

In order to check the validity of the threshold obtained, we propose the application of cross-validation to that threshold. To this end, the second data set must be used. It is important to highlight that it contains information related to process instances different from those used for threshold determination.

We propose to approach the cross-validation of the thresholds by calculating precision and recall measures for assessing the quality of the prediction, as it is applied for evaluating a search result in information retrieval field [26]. *Precision* is the ratio of true positives to the sum of true and false positives ($Precision = TP/(TP + FP)$) [27]. In our context, this is the ratio of correctly predicted lag PPI fulfillments based on a threshold value in relation to all predicted lag PPI fulfillments. *Recall* is the ratio of true positives to the sum of true positives and false negatives ($Recall = TP/(TP + FN)$) [27]; i.e., the ratio of correctly predicted lag PPI fulfillments based on a threshold value in relation to all actual Lag PPI fulfillments.

To achieve accurate predictions, a technique should achieve both high precision and recall. However, an intrinsic relationship between precision and recall exists: increasing one of them may decrease the other. To combine precision and recall in a single value, literature thus recommends using measures such as the F-measure [28] (also known as F-score or F-1), which is defined in Eq. 4.

$$F\text{-}measure = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{4}$$

The above measures do not reflect a prediction technique's ability in predicting true negatives [29]. To complement our evaluation, we also propose to include specificity and accuracy measures. *Specificity* (Spec), as explained in Sect. 3, is calculated as the ratio of true negatives to the sum of false positives and true negatives ($Spec = TN/(FP + TN)$), and indicates how many actual non-fulfillments were correctly predicted as non-fulfillments. Finally, the *accuracy* (Acc) is a widespread measure of effectiveness, to evaluate a classifier's performance [30] and it is calculated as the sum of true positives and true negatives to the sum of true and false positives and true and false negatives ($Acc = (TP + TN)/(TP + FP + TN + FN)$), in other words, it is the percentage of correctly classified instances. Precision, recall, F-measure, specificity and accuracy are measures that are appropriate for computing the effectiveness of search results [26,29].

4 Evaluation with a Case Study

In order to evaluate the applicability of our approach, we conducted a case study. It was carried out in the context of an international publishing company[1] aiming at improving its core business processes. In particular, we focused on one of them,

[1] No further information can be provided about the company and its business processes due to privacy reasons.

the process associated to the management of manuscripts from the moment they are received by the editor to their publication (or rejection), trying to identify the relationship between cycle time and workload, as required by the publishing company quality manager. In this process, when a new instance arrives, the manager has to assign it to an employee, so manager's primary job is to divide the work optimally over her team. Currently, a manager is given an overview of his/her employee's progress using a report tool. This tool contains information like the number of instances his/her employees are working on or the subtasks durations. When assigning a given instance to an employee, the manager has to estimate how long this employee will take to finish the process (i.e. the cycle time). In order to help the manager to decide which employee will finish the process faster, it would be desirable to have information available to identify those PPIs or performance measures that have an influence on the value of the cycle time.

The guidelines proposed by Runeson and Höst [31] and Brereton et al. [32] were followed to design and conduct the case study, which is described in the following subsections.

4.1 Case Study Design

We carried out a holistic case study [33], with a single-case, in a single organization and in a single project of the organization. The object of the study was the improvement of the performance of the manuscript management process of this publishing company, and the main objective was to provide the publishing company's managers with additional performance information so that they can divide tasks between employees optimally, obtaining the pursued target value for cycle time. In this context, the cycle time was identified as a lag PPI and the workload as a lead PPI. Thus, the research question for this case study can be defined as follows: *"How does workload influence cycle time and what thresholds can be established for it to assure the fulfillment of the cycle time PPI?"*.

Regarding the case selected, the reasons for this selection are mainly two: first, the quality responsible was particularly interested in improving this process since it is one of the most critical processes from a customer/user point of view and can directly lead the company to success or failure; and second, a huge amount of execution data was available for the analysis. Furthermore, its lag PPI cycle time is very relevant for customer satisfaction according to the quality department of the company. Though there are probably other factors apart from the workload that influence the time it takes to an editor to complete the process, we focused on the workload because we were specifically asked to look at the relationship between workload and cycle time.

4.2 Data Collection and Analysis

The study presented in this paper consists of the application of our method to the execution data retrieved from 1080 process instances of the selected BP. Using

insights from a business analytics platform[2], we collected data from the object BP and the computation of workload and cycle time values for each execution. In particular, in the case of workload, the initial definition used was the *begin workload*, i.e., the number of process instances an employee is working on at the start of a new instance. However, we had to change it since, after a first analysis of the data, no apparent connection was found between the defined workload and the duration of a process instance (its cycle time). Instead, the *average workload* was used for this study. It can be defined as the weighted average of the number of instances an employee is working on during a process instance. Regarding cycle time, their values were obtained in milliseconds, as this is the unit provided by the information systems that gather the execution data in the publishing company. Finally, a pursued target value for the cycle time PPI was also provided by the quality department.

1. **Preprocessing**
   The Boolean variable in this case corresponds to the fulfillment of the lag PPI cycle time. The values of this variable were obtained by comparing the cycle time value of each BP instance with the target value established. We assign this variable the value 1 when the cycle time value is lower or equals to its target value, and 0 in other case. In addition, the data set described above was divided into two groups. The values from 700 BP instances were used for threshold extraction, and the values from the remaining 380 BP instances for threshold validation. Table 2 shows the average (μ) and standard deviation (γ) values for workload and cycle time in these two datasets. Workload values represent process instances (PI), and Cycle time values appear in milliseconds, as obtained from the execution data, and in weeks, for readability reasons.

Table 2. Average and standard deviation for workload and cycle time in the two datasets.

Dataset	Workload (PI)		Cycle time (ms)		Cycle time (weeks)	
	μ	γ	μ	γ	μ	γ
Extraction	29.46	12.72	5.43 E9	3.71 E9	8.98	6.14
Validation	25.65	13.67	4.95 E9	3.52 E9	8.19	5.82

2. **Checking the relationship**
   Here we have to prove that workload values do have an influence on the fulfillment of the cycle time target value. So as to apply logistic regression, we are interested in the binary variable $fulfilledCTTargetValue$ with the range $\{fulfillment, non-fulfillment\}$, the independent variable $Workload$, and observations from $i = 700$ business process instances.

[2] Its identity is not revealed for confidentiality restrictions.

Applying the logistic regression to our particular data, we obtain the coefficients (the intercept α and the only regression coefficient β in our case) represented in Table 3. The results show that there exists a correlation between both variables, the workload and the fulfillment of the cycle time target value, and that it is statistically significant, given the resulting p-value for the model of $0.000 < 0.05$. This proves that the workload have an influence on the cycle time.

Table 3. Coefficients of the logistic regression applied to our data.

Coeficients	Value	Std. error	p-value
α	7.994	1.068	0.000
β	−0.137	0.025	0.000

3. **Threshold extraction with ROC Curve**
   The ROC curve obtained from our data is depicted in Fig. 1. The resulting AUC value is 0.833, and the p-value 0.000 (<0.05), so the discriminative power of the workload can be considered excellent and significantly different from 0.5 from a statistical point of view. Now, we can determine a threshold value for the workload based on the ROC curve, selecting the point with the greatest distance from the 0.5 diagonal. In this case, this **threshold is 39.68**, which means that for a process instance assigned to an employe working on more than 39.68 instances on average during that instance, will likely not fulfill the pursued cycle time.
4. **Application of the Bender Method to determine probabilities of errors for threshold ranges**
   For the application of the Bender method in our case, the independent variable in the logistic regression model is the workload for which we want to determine the threshold, and the dependent variable is $fulfilledCTTargetValue$. From Eq. 3 we get a **probability of 0.93**. This can be interpreted as "if the employee's workload is lower than or equal to 39,68, there is a 93% of probability that she finishes the BP instance in less than the target value of the cycle time".

   Furthermore, as stated above, we can apply this method to identify other threshold values associated with different probabilities of the target value fulfillment. Table 4 depicts this information for our case and can be interpreted as follows. For a given instance, If the workload is approximately 74, then the probability of fulfilling the cycle time target value for that instance is 10%, which indicate that the workload is not appropriate at all. Conversely, if the workload is about 48, there is a probability of 80% that the BP instance fulfills the cycle time target value.
5. **Threshold validation**
   Finally, the calculations of the different measures defined in Sect. 3.5 in our case study result in the values contained in Table 5. From all the BP instances

Table 4. Workload thresholds with associated probabilities extracted with the Bender method.

Probability of considering the cycle time fulfilled	10%	20%	30%	40%	50%	60%	70%	80%	90%
Workload	74.39	68.47	64.54	61.31	58.35	55.39	52.17	48.23	42.31

Table 5. Values for Precision, Recall, F-measure, Specificity and Accuracy for the extracted threshold.

Precision	Recall	F-measure	Specificity	Accuracy
0.98	0.87	0.92	0.59	0.86

predicted as fulfilling the cycle time target value, 98% of the cases actually fulfilled it. In addition, from all the BP instances that really fulfilled the cycle time target value, 87% were correctly predicted. The lowest value is obtained for the specificity. In this case, from all the non-fulfilments, about 60% are correctly predicted. As for the accuracy, 86% of the cases were correctly predicted. Though there is no existing benchmark to which compare these values, they can be considered acceptable values taking into account they are in general high values. Taking these results into consideration, this approach can be used as a predictive model that supports the decision-making process of the managers in the publishing company, and can be improved in the future with data extracted from further process executions.

4.3 Interpretation of Results

The threshold obtained for the workload can be used to provide a more confident answer to the research question put forth in Sect. 4.1. This information supports managers during the assignment of new manuscripts to editors as follows. When a new manuscript needs to be assigned, the corresponding manager will check workload values for his/her editors, and will select the one with the lowest value. When possible, this workload value should be lower than 39.68, which is the obtained threshold. Otherwise, two options are available: either hiring new editors, which is not the common case at all; or taking certain risk. Our approach also provides information in this direction thanks to the results obtained from the Bender method (c.f. Sect. 3.4). If the manuscript is assigned to an editor with a workload about 48, the probability to fulfill the cycle time target value is 80%, if the workload is closer to 52, the probability of fulfillment is closer to 70%, and so on. In this way, the manager is aware of the risk taken when necessary.

These provided thresholds can serve as a starting point for application in practice, and they should be continuously gauged according to feedback obtained

from the practical experience derived from its usage as well as from data produced in future process executions.

4.4 Threats to Validity

In the context of the presented case study, the following types of validity threats can be considered. With regards to the *conclusion validity*, the size of the sample data used to perform the case study is of 1080 execution instances (700 for threshold extraction and 380 for validation), which is a considerable size for these cases, however, the study could be enriched by varying the sizes of the partitions and the samples.

In relation to *construct validity*, which is about reflecting our ability to measure what we want to measure, the measures used in this study (workload and cycle time) are relevant measures used in related literature, and they were measured or computed according to definitions in the related literature (e.g. [34]).

Internal validity concerns whether the effect measured is due to changes caused by the researcher, or from some other unknown cause. The possible threats to internal validity were: ROC curves are used and a possible disadvantage is that the discrimination (sensitivity, specificity) is not the only criterion for a good prediction. A curve with a larger AUC (which is apparently better) could be obtained even though the alternative may show superior performance over almost the entire range of values of the classification threshold. This has been mitigated with the validation of the obtained threshold. In addition, the application of ROC curves mitigates some negative aspects of other statistical techniques which require the setting of several input parameter values, which has the risk of obtaining unrealistic results for a bad setting of such parameters. In addition, ROC curves have a more intuitive interpretation of the results. With regard to the application of the Bender method, the main limitation could be the need of a binary variable as input which requires dichotomization in cases in which this binary variable is not available, with the consequent loss of information. This was not our case, as a binary variable was used as input.

Finally, regarding *external validity*, which describes the possibility of generalizing its results, in this research real data have been used from a representative business process of a company and a useful threshold has been obtained to support decision making in such process, which reinforces its validity. However, the main threat is related to the fact that each business process is particular in each organisation, and the same happens with the PPIs defined for each business process and their associated target values. In other words, the extracted threshold is context-dependent and it is not generalizable to other business processes or companies, but the threshold determination method used in this research could be reused for obtaining thresholds for other representative processes in this company, or even in other companies and domains whenever enough execution data is available. Actually, the organization where our case study was carried out presents several characteristics of organizations that would be interested in applying the same method. For example, there is a set of representative BPs

with associated PPIs, from which execution data is recorded on different information systems and from where it is possible to be gathered. Another important characteristic is that the publishing company already has a quality department, which is a key factor for providing key information about PPIs and objectives to be fulfilled.

5 Conclusions and Future Work

In this paper we proposed a method to extract thresholds for lead PPIs that allow the fulfillment of a lag PPI. This method was validated through a case study performed in the context of an international publishing company, using 700 process instances to extract the threshold and 380 for its validation. The extracted thresholds and associated probabilities allow the publishing company managers to decide how to regulate workload levels to achieve the desired cycle time target value, and when to assume certain risks, being aware of the exact risk, according to the probabilities provided.

This method for threshold determination can be also applicable to other domains such as SLAs, where a guarantee term is provided, and It must be fulfilled to avoid penalties. This guarantee term could be seen as the lag PPI and is defined on the basis of other measures, which would be analogous to our lead PPIs. This is part of our future work. Furthermore, we plan to define a tool to support the methodology presented, extend it for multi-instance PPIs and apply it to other different domains.

References

1. Parmenter, D.: Key Performance Indicators (KPI): Developing, Implementing, and Using Winning KPIs. Wiley, Hoboken (2010)
2. Sánchez-González, L., García, F., Ruiz, F., Mendling, J.: A study of the effectiveness of two threshold definition techniques. In: 16th International Conference on Evaluation & Assessment in Software Engineering, EASE 2012, pp. 197–205 (2012)
3. Wetzstein, B., Leitner, P., Rosenberg, F., Dustdar, S., Leymann, F.: Identifying influential factors of business process performance using dependency analysis. Enterp. IS **5**(1), 79–98 (2011)
4. del Río-Ortega, A., Resinas, M., Cabanillas, C., Ruiz-Cortés, A.: On the definition and design-time analysis of process performance indicators. Inf. Syst. **38**(4), 470–490 (2013)
5. McChesney, C., Covey, S., Huling, J.: The 4 Disciplines of Execution: Achieving Your Wildly Important Goals. Simon and Schuster, New York (2012)
6. Rodriguez, R.R., Saiz, J.J.A., Bas, A.O.: Quantitative relationships between key performance indicators for supporting decision-making processes. Comput. Ind. **60**(2), 104–113 (2009)
7. Popova, V., Sharpanskykh, A.: Modeling organizational performance indicators. Inf. Syst. **35**(4), 505–527 (2010)
8. de Leoni, M., van der Aalst, W.M.P., Dees, M.: A general process mining framework for correlating, predicting and clustering dynamic behavior based on event logs. Inf. Syst. **56**, 235–257 (2016)

9. Bender, R.: Quantitative risk assessment in epidemiological studies investigating threshold effects. Biometrical J. **41**(3), 305–319 (1999)
10. Shatnawi, R., Li, W., Swain, J., Newman, T.: Finding software metrics threshold values using ROC curves. J. Softw. Maint. Evol. **22**(1), 1–16 (2010)
11. Catal, C., Alan, O., Balkan, K.: Class noise detection based on software metrics and ROC curves. Inf. Sci. **181**(21), 4867–4877 (2011)
12. Youngblood, A.D., Collins, T.R.: Addressing balanced scorecard trade-off issues between performance metrics using multi-attribute utility theory. Eng. Manag. J. **15**(1), 11–17 (2003)
13. Diamantini, C., Genga, L., Potena, D., Storti, E.: Collaborative building of an ontology of key performance indicators. In: Meersman, R., Panetto, H., Dillon, T., Missikoff, M., Liu, L., Pastor, O., Cuzzocrea, A., Sellis, T. (eds.) OTM 2014. LNCS, vol. 8841, pp. 148–165. Springer, Heidelberg (2014). doi:10.1007/978-3-662-45563-0_9
14. Patel, B., Chaussalet, T., Millard, P.: Balancing the NHS balanced scorecard!. Eur. J. Oper. Res. **185**(3), 905–914 (2008)
15. Sánchez-González, L., García, F., Ruiz, F., Piattini, M.: Toward a quality framework for business process models. Int. J. Coop. Inf. Syst. **22**(01), 1350003 (2013)
16. Mendling, J.: Metrics for Process Models: Empirical Foundations of Verification, Error Prediction, and Guidelines for Correctness. LNBIP, vol. 6. Springer, Heidelberg (2008)
17. Delgado, A., Weber, B., Ruiz, F., de Guzmán, I.G.R., Piattini, M.: An integrated approach based on execution measures for the continuous improvement of business processes realized by services. Inf. Softw. Technol. **56**(2), 134–162 (2014)
18. Herbold, S., Grabowski, J., Waack, S.: Calculation and optimization of thresholds for sets of software metrics. Empir. Softw. Eng. **16**(6), 812–841 (2011)
19. Sánchez-González, L., García, F., Ruiz, F., Mendling, J.: Quality indicators for business process models from a gateway complexity perspective. Inf. Softw. Technol. **54**(11), 1159–1174 (2012)
20. Mendling, J., Sánchez-González, L., García, F., Rosa, M.L.: Thresholds for error probability measures of business process models. J. Syst. Softw. **85**(5), 1188–1197 (2012)
21. Sánchez-González, L., García, F., Ruiz, F., Piattini, M.: A case study about the improvement of business process models driven by indicators. Softw. Syst. Model., 1–30 (2015). doi:10.1007/s10270-015-0482-0
22. Hosmer, D., Lemeshow, S.: Applied Logistic Regression. Wiley, Hoboken (2004)
23. Zweig, M.H., Campbell, G.: Receiver-operating characteristic (ROC) plots: a fundamental evaluation tool in clinical medicine. Clin. Chem. **39**(4), 561–577 (1993)
24. Hand, D.J.: Measuring classifier performance: a coherent alternative to the area under the ROC curve. Mach. Learn. **77**(1), 103–123 (2009)
25. Morasca, S., Ruhe, G.: Introduction: knowledge discovery from empirical software engineering data. Int. J. Softw. Eng. Knowl. Eng. **09**(05), 495–498 (1999)
26. Baeza-Yates, R.A., Ribeiro-Neto, B.A.: Modern Information Retrieval. ACM Press/Addison-Wesley, New York/Boston (1999)
27. Olson, D.L., Delen, D.: Advanced Data Mining Techniques, 1st edn. Springe, Heidelberg (2008). Incorporated
28. Salfner, F., Lenk, M., Malek, M.: A survey of online failure prediction methods. ACM Comput. Surv. **42**(3), 10:1–10:42 (2010)
29. Metzger, A., Leitner, P., Ivanovic, D., Schmieders, E., Franklin, R., Carro, M., Dustdar, S., Pohl, K.: Comparing and combining predictive business process monitoring techniques. IEEE Trans. Syst. Man Cybern.: Syst. **45**(2), 276–290 (2015)

30. Michie, D., Spiegelhalter, D.J., Taylor, C.C., Campbell, J. (eds.): Machine Learning, Neural and Statistical Classification. Ellis Horwood, Upper Saddle River (1994)
31. Runeson, P., Höst, M.: Guidelines for conducting and reporting case study research in software engineering. Empir. Softw. Eng. **14**(2), 131–164 (2009)
32. Brereton, P., Kitchenham, B., Budgen, D.: Using a protocol template for case study planning. In: Proceedings of EASE 2008, BCS-eWiC (2008)
33. Yin, R.: Case Study Research: Design and Methods. Applied Social Research Methods. SAGE Publications, Thousand Oaks (2009)
34. Nakatumba, J.: Resource-aware business process management: analysis and support. Ph.D. thesis, Eindhoven University of Technology (2014)