Suggestions for Improving a Bank's Loan Application Process based on a Process Mining Analysis

Gregor Scheithauer[1], Roman Henne[1], Arled Kerciku[1],
Robert Waldenmaier[1], Ulrich Riedel[1]

[1] metafinanz Informationssysteme GmbH, 80804 Munich, Germany

{gregor.scheithauer, roman.henne, urlich.riedel,
robert.waldenmaier, arled.kerciku}@metafinanz.de

Abstract. Every year, the International Workshop on Business Process Intelligence (BPI) sets out a challenge for students, researchers, and practitioners. Participants should demonstrate novel tools, approaches, and algorithms to solve the challenge. This year's challenge provides anonymized loan application data from a Dutch financial institution. In this paper, we demonstrate how we apply process mining technology, data visualization, and statistical models to determine the actual process duration and wait times, to show the impact on requested customer information and customer conversion rate, and to show how many offers made to the applicant will grant a successful application. Based on our analysis we derive suggestions for the bank to improve the process.

Keywords: BPI Challenge, process mining, data mining, loan application management, process optimization, RStats.

1 Introduction

Process analysis is not trivial. In general, it involves many resources, takes time, and findings are often ambiguous or even unreliable. On the other hand, companies would rather spend time and resources on realizing benefits than analyzing as-is processes.

A data-oriented approach can overcome these barriers to some degree. Process mining is a data-oriented process analysis technique "*[…] to discover, monitor and improve real processes (i.e. not assumed processes) by extracting knowledge from event logs readily available in today's (information) systems […]*" [1]. Since it works with facts (i.e. event logs) it involves on average fewer resources and can be automated. Hence, it is faster and findings are less ambiguous.

Figure 1 shows the basic three process mining use cases [1]. **Discovery** uses event logs from one or more systems to derive a process model that satisfies processes that were executed in a period found in the event log. This is very helpful in cases were no process overview and no transparency about the process flow exist. **Conformance checking** describes how executed processes match a given normative model. Deviations or non-conformal process executions are highlighted and diagnostics can be used

to determine the reasons why the processes were not compliant. **Enhancement** describes a way to enrich a process model based on a given normative model and on concrete process executions. The resulting model covers all possibilities and is a better fit for existing process executions.

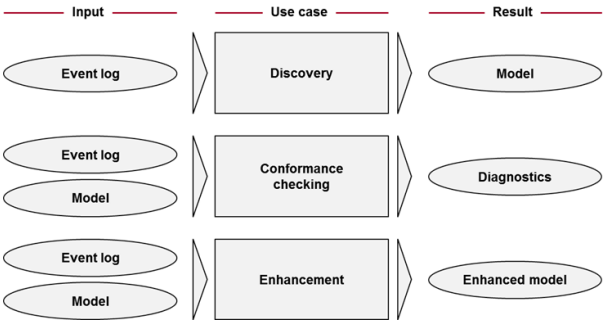

Fig. 1. Overview of three basic process mining use cases [1].

Process mining projects touch on a lot of varied expertise. Firstly, domain knowledge is necessary to identify what to look for in the data, to collect the necessary data as well as interpret analysis results. Secondly, data handling, such as cleansing, masking and transforming is necessary to put the available data into the correct form. Thirdly, process management expertise to guide the project and implement improvements into analyzed processes.

Every year, the International Workshop on Business Process Intelligence (BPI) sets out a challenge for students, researchers, and practitioners. Participants should demonstrate novel tools, approaches, and algorithms to solve the challenge. This year's challenge provides anonymized loan application data from a Dutch financial institute. We are a group of practitioners in the fields of process management, data science and reporting and are intrigued by the BPI Challenge 2017. In this paper, we would like to document what we have learned about process mining, apply our methods and tools as well as present our findings.

The remainder of this paper is structured as follows: in the next section, we outline our method and tool chain. In section three, we present the available loan application data as well as general findings and give a process overview based on the data. In the subsequent three sections, we address the three given key questions, outline the analysis approach and present findings. We discuss our findings and recommend possible measures to address our findings. We conclude our paper in the final section.

2 Process Mining Approach and Setup

This section illustrates our approach as well as the technical setup that guides and supports us in addressing the given questions.

2.1 Process Mining Approach

Our approach to plan and execute process mining projects can be described in six steps (cf. figure 2). Depending on the nature of the project, all or some of the steps are considered. In the context of this challenge only steps four and five are considered, since relevant questions and corresponding data were provided. In the following section, we define each step briefly.

Fig. 2. Process mining approach overview

Find business-relevant questions, hypotheses. Experience shows us to try to initiate every process mining project with questions or hypotheses where the answers are of interest to the business. This course of action is also suggested in the process mining manifesto [1]. The benefits of bringing together domain experts are that it is possible to create a shared understanding of the process challenges under investigation, discuss transparent problems (such as missed agreed service levels) and supposed root causes as well as to prioritize the investigation thereof by ranking them according to potential business benefit if they could be solved. Another outcome of this step is the understanding of what data is needed to investigate these questions.

Get **necessary data that answers** questions **or prove hypotheses, respectively.** The business-relevant focus allows to derive necessary data in width and granularity. If the business is interested in supplier performance, for example, then the data needs to be extensive enough to hold information about supplier names (data wideness). If the questions can be solved on an abstract level it might be sufficient to focus on process instance data such as overall duration or number of activities. However, if the question demands a more profound look, e.g. at activity performance and number of different process flows, information on activity (or even workflow) level is required. Once the necessary data width and granularity is planned, it is possible to determine systems that could provide the data. If no such system exists, we suggest two courses of action: (1) arrange existing systems to collect the necessary data or (2) limit the analysis to data that is available.

Mask and anonymize data, when necessary. Usually, data includes references to personal and sensitive information, including companies' employees and customers. This data needs to be dealt with carefully. In most cases, this concrete information may not be employed and is not necessary for analyses. The variance in this data is of interest rather than the concrete values. Based on the data privacy requirements, we employ six different methods as needed:

1. Masking out - mask a certain number of characters
2. Number and date variance - modify each number or date value by a random percentage of the real value
3. Substitution - replace actual data with random data

4. Shuffling - randomly move data within one column between rows
5. Encryption - use symmetric encryption
6. Nulling out - delete certain information

Analyze data. In case of missing values or untidy data, data is manipulated to meet certain quality criteria, a process often referred to as data cleansing. Furthermore, additional information can be calculated that facilitates further analysis, e.g., if start and end times for activities exist it is possible to calculate the duration of each activity. Then we address each question or hypothesis and draw a way of how to find the required answers. This includes several statistics (e.g. quartiles, mean, min, max), data visualizations (e.g. histograms, scatter plots for relationships, and process flows), and predictive models (e.g. random forests). Their application can be found in the following four sections.

Draw conclusions and derive measures for improvement. Following the analysis step, conclusions are drawn from data analyses. Quite often, data inconsistencies reveal themselves and need to be addressed. Answers are prepared and hypotheses are tested based on analyses results. Additionally, further points of interest are explored that amend drawn conclusions. Based on the results and domain expertise we derive measures to improve potential findings. These measures could include a number of things, such as but not limited to:

- Train users / improve work instructions to reduce undesirable process variants
- Implement measures to improve overall data quality to reduce re-work
- Implement controls to prevent fraud or improve customer feedback
- Remove ineffective controls to improve performance
- Redo incentive systems (i.e. Key Performance Indicators)
- Raise degree of automation to improve performance and quality

Implement process mining for continuous improvement. Process mining is a powerful tool for finding the root causes of process inefficiencies. Often, such an analysis reveals important insights that were unavailable before and can be a foundation for several improvement projects. To fully exploit the technology, we recommend executing such analyses on a regular basis. This allows companies to react to actual process inefficiencies and problems in a timely manner rather than identifying inefficiencies in the distant past. Depending on companies' settings, implementing process mining could comprise:

1. Developing a target picture
2. Deciding on distributed or centralized process mining team
3. Training and hiring experts
4. Selecting appropriate tools (including software that is already available in the company)
5. Integrating process mining into existing reporting landscape
6. Integrating expert team and tool chain into process management governance

2.2 Setup

Our technical goal is to have repeatable data gathering, masking, and manipulating and plot generation whenever needed by automating as much as possible, even though this initially requires additional effort. This is helpful as it saves re-work when additional data is provided later or some of the steps need to be reconfigured based on received feedback.

In general, our setup consists of tools that support data cleansing, masking, and analysis. Depending on the context, these tools may vary. For this paper, we make use of R [3], a powerful language to support all aforementioned tasks, including their automation. Especially the notebook functionality [8] is very helpful. Alternative tools to R include e.g. RapidMiner [4], SAS [6], or MS Excel. To analyze process control flow, we use Fluxicon's tool Disco [2]. Alternative tools to Disco include e.g. Celonis [5] or ProM [7]. Outputs of R and Disco can be found in the following section.

Table 1. Overview of selected event log variables.

Selected variable	Description	Example
Case ID	Identifier of for each application	Application_652823628
Activity	Name of the activity that was performed for one application	*A_Create Application*
Resource	Identifier of an employee or system (anonymized)	User_1
Start Timestamp	Start time of performed activity	2016-01-01 10:51:15.303
Complete Timestamp	End time of performed activity	2016-01-01 10:51:15.303
Application Type	Indicates whether this application is for a new credit or a raise for an existing credit	New credit
Loan Goal	Applicant's reasons for the loan	Existing loan takeover
Requested Amount	Applicant's requested loan amount	20.000
Credit Score	Describes whether an applicant is dependable	NA
OfferID	Identifier for each offer	NA
Offered Amount	The amount of money offered to an applicant	NA

3 General Process Analysis

This section introduces the **loan application process**. The process covers the application of loans, the application's validation, and the decision whether to make an offer or not, the reply of the applicant, as well as validating the applicants' decisions whether to accept the offer. The data provided contains **31.509 different process instances** and covers the time from **January 2016** to **February 2017**. The following paragraphs introduce the event log, present a process overview, and discuss process performance.

The event log is provided via two files in XES format. The loan application contains all information regarding the process. The additional loan offer file is a subset of the former file and only contains events related to offers. For this analysis, we concentrate on the loan application log, since it contains all the information, and filter out the necessary information for each analysis as needed.

Table 2. Application-relevant activity overview.

Activity	Description	Start or end
A_Create Application	Depicts the start of an application process.	Start activity
A_Submitted	Applicant submits an application on the website	
A_Concept	A first, automatic assessment of the application has been done and an employee calls the customer to complete the application.	
A_Accepted	Following the call with the applicant, the application is re-assessed.	
A_Complete	The offers have been sent to the customer and the bank waits for the customer to return a signed offer along with the remaining documents.	
A_Validating	Evaluation of the received documents by the bank.	
A_Incomplete	Received documents are not correct or incomplete and the applicant has to send more documents.	
A_Pending	All documents have been received and the assessment is positive. The loan is paid to the customer.	End activity
A_Denied	The application doesn't match the acceptance criteria	End activity
A_Cancelled	The application is canceled if the applicant does not get back to the bank after an offer was sent out	End activity

The event log [9] is described as "*This event log pertains to a loan application process of a Dutch financial institute. The data contains all applications filed through an online system in 2016 and their subsequent events until February 1st, 2017*" The data has **561.671 events** and **23 variables**. Table 1 shows selected variables.

The process has 26 distinct activities that can be divided into three categories: application-relevant activities, offer-relevant activities, and workflow-relevant activities. Naturally, not every activity is performed with the same frequency. Figures 3 and 4 suggest that the most frequent activities are *O_Created* and *O_Create offer*, followed by *O_Sent* (mail and online), *W_Validate application*, and *A_Validating*. The activity with the lowest frequency is *W_Personal Loan collection*. In Table 2 we explain the application-relevant process activities of the loan application process.

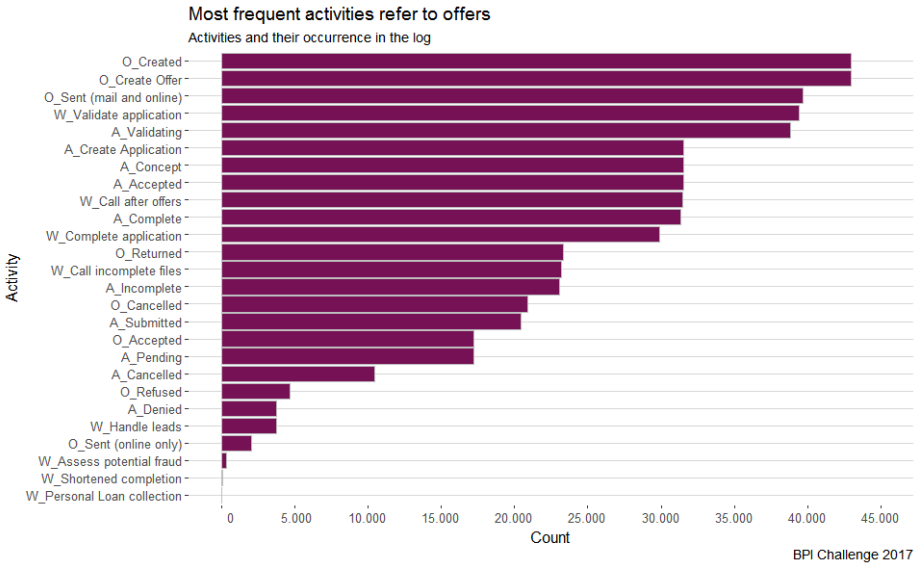

Fig. 3. Activities and their occurrence in the log.

In addition, the log tells us that there are **4.047 process variants** with different frequencies. Figure 5 shows the 75 most frequent process variants. The most frequent variant (variant one) covers over eleven percent of all process instances (3.656 instances) and the 75 most frequent variants cover 72 percent of all instances (22.611 instances).

Another way of looking at the process is to distinguish between entry channels and the way the process was ended, i.e. no offer was made to the customer, customer refused an offer or customer accepted one offer. The two start scenarios are: (1) apply via website – an applicant applied for a loan via the bank's website – User 1 (system resource) responsible for initial activities and waiting time in hours between *A_Concept* and *W_Complete application* (e.g. variants one, two, and three) - 20.423 instances, and (2) apply via bank - an applicant applies in person and a clerk enters the application - no activity *A_Submitted* present and User 1 not responsible for initial activities (e.g. variant 4) - 11.086 (cf. also figure 6).

The most frequent process end points (cf. figure 6) are: (1) application denied - the loan cannot be offered to the customer - 3.752 instances, (2) application canceled - offer was made to the applicant but the applicant did not get back to the bank within 30 days

- 10.431 instances, and (3) application pending - all documents are received and the assessment is positive, the loan is final and paid out to the customer - 17.228 instances.

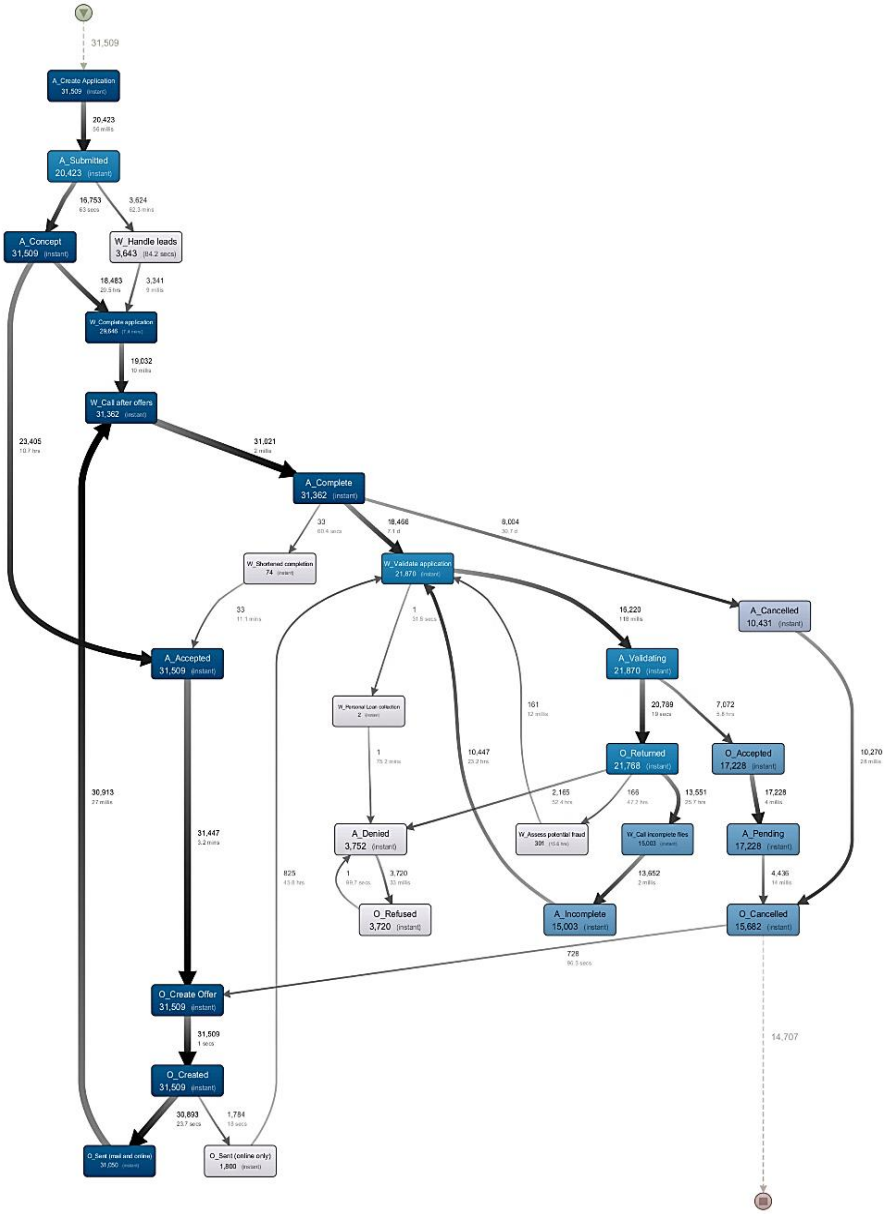

Fig. 4. Process overview with frequency statistics, limited to most frequent paths (second metric: median duration and wait times).

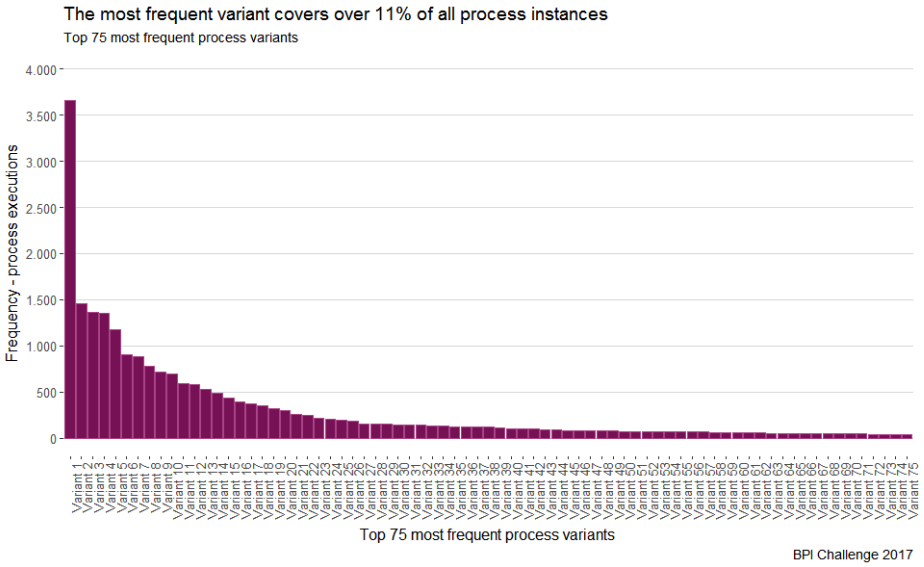

Fig. 5. Overview of the 75 most frequent process variants.

Table 3 provides an overview of the six generic process variants and their frequencies. It abstracts from the many ways a process instance can go and focuses instead on possible start and end points. The most frequent process variant is that applications are made via website and are accepted by the customer (application pending). It also shows that applications that are made via website have a conversion rate of 49 percent in comparison to applications that are made via bank that have a conversion rate of 65 percent. Overall the conversion rate is 55 percent, and hence on average, applications via bank have a higher conversation rate.

Table 3. Process instance distribution via process start and end points.

Input channel/ Outcome	Application denied	Application canceled	Application pending	Other end points	Sum
Apply via website	2.702	7.573	10.064	84	**20.423**
Apply via bank	1.050	2.858	7.164	14	**11.086**
Sum	3.752	10.431	17.228	98	**31.509**

There are 561.671 activities in the log file amounting to 31.509 applications. On average, 18 activities are performed for each application. The process with the fewest activities has two activities, which suggests that there might be incomplete process instances in the dataset. 75 percent of all processes have 20 or fewer activities. One process instance with the highest number of activities performed counts 61 activities. Figure 7 shows the distribution of number of activities per application.

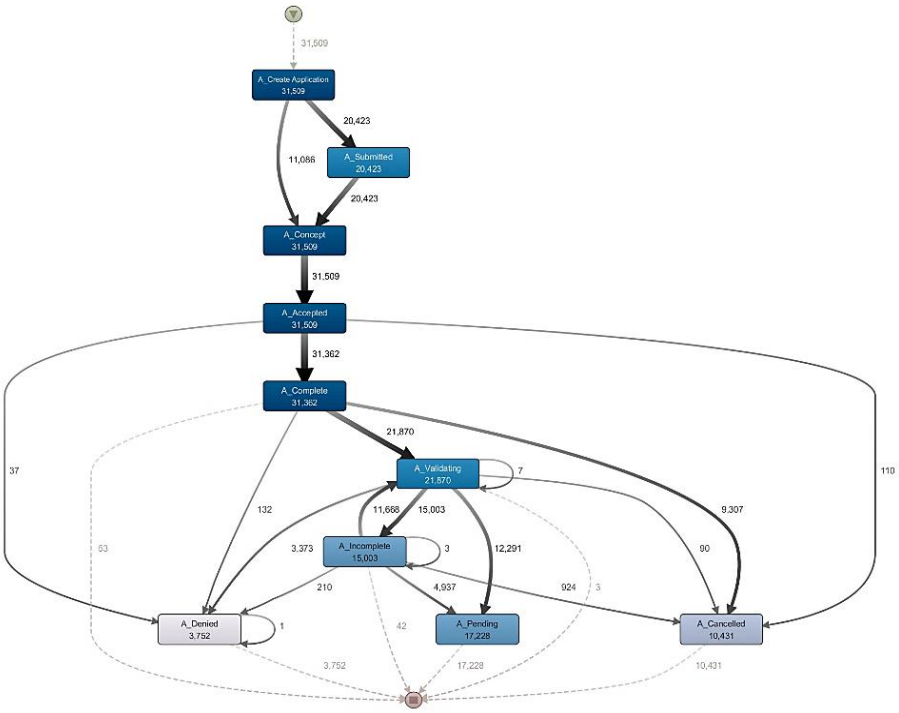

Fig. 6. Process activities and their frequencies (only application-relevant activities are shown).

Considering process performance, the data shows that, on average, instances took 21 days to complete. Figure 8 shows a peak in ten days and in 31 days. Looking back at the possible process endpoints, applications that were canceled due to a missing applicant response wait for a response for exactly 30 days. This would explain the peak around 31 days. The distribution around ten days shows instances where applications were either successfully accepted by applicants or denied. Further statistics show that 25 percent of process instances can be concluded in ten days, the 25 percent with the longest process durations were between 31 and 169 days, and the median is 18 days. However, since there are many different variants of the application process, e.g. different start points, different outcomes or different loan goals, it is necessary to break down the performance analysis to specific variants for comparison.

Figure 4 also shows median activity durations and median wait times between activities. It is important to note that the median wait time is 20,5 hours between the activities *A_Concept* and *W_Complete application*. On further inspection, it turned out that if applications are submitted via website it takes a day before the bank looks at the application to mark it as complete.

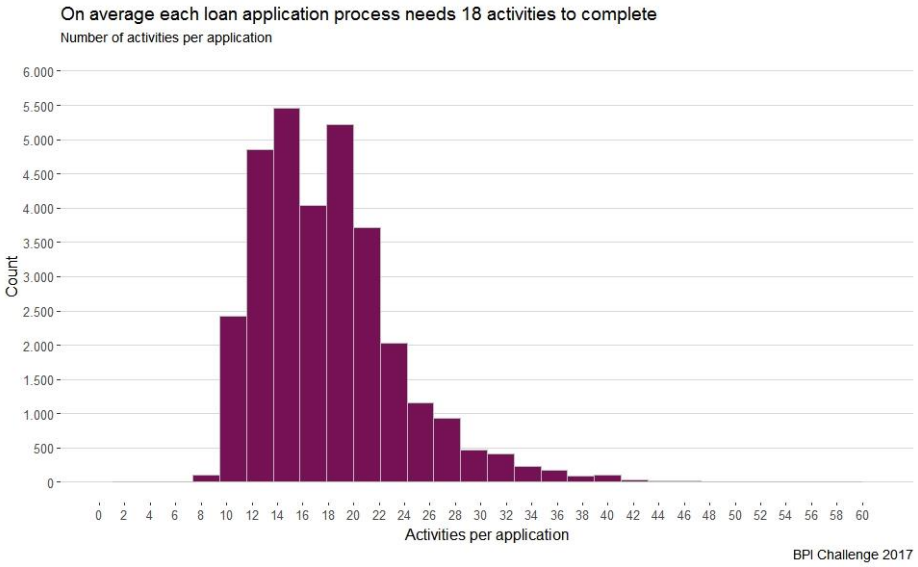

Fig. 7. Number of activities per application – distribution.

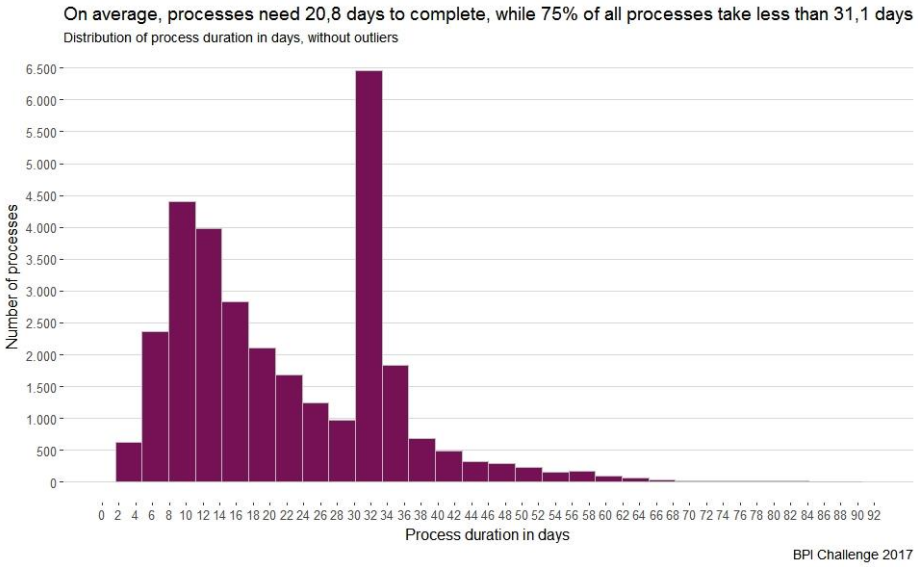

Fig. 8. Distribution overview of process duration in days.

4 Throughput Times Analysis

In this section, we address the first question of the BPI Challenge: "*What are the throughput times per part of the process, in particular the difference between the time*

spent in the company's systems waiting for processing by a user, and the time spent waiting on input from the applicant as this is currently unclear?". In order to answer the question, we went through the following steps:

1. Identify and analyze process parts where an application is waiting to be processed by the bank (either an employee or a system)
2. Identify and analyze process parts where the bank is waiting for input from the applicant

The loan application process based on the dataset at hand consists of 26 process activities. As already explained in the previous sections, these activities are divided into three main categories: application activities (*A_*), offer activities (*O_*) and workflow activities (*W_*). The initiation of workflow activities indicates that a certain workflow has started. Each of the workflow activities generally consists of application and/or offer activities. If we compare Figure 4 and Figure 9, one can observe that in terms of frequency or median duration, they are the same. Therefore, we omitted the workflow activities in this chapter and concentrated only on application and offer activities. There are 18 application-relevant and offer-relevant activities in total. Each activity is either initiated by the bank (e.g. *A_Accepted*, *O_Created*, and *A_Validating*) or by the applicant (e.g. *A_Create Application*). The loan application process starts with the creation of the application (*A_Create Application*) and ends in most cases with one of the following endpoints: *A_Pending* (offer has been accepted), *A_Cancelled* (offer has been canceled) and *A_Denied* (offer has been denied). In this section of the paper, we only focus on the waiting times between these activities.

4.1 Identify and analyze process parts where an application is waiting to be processed by the bank (either an employee or a system)

The waiting time between each of the 18 activities occur either because the bank is waiting for input from the applicant or the application is waiting to be processed by the bank (by either an employee or a system). In this subsection, we focus on the latter. Depending on the granularity of the process, the number of process parts varies. We focused on those process parts that have a considerable impact in the overall throughput time of the process, and identified five such process parts:

- B1->B2 [*A_Concept - A_Accepted*]: Waiting time after the application was created and before the offer creation process is started by a bank employee.
- B3->B4 [*A_Validating - O_Accepted*]: The waiting time after the validation process has finished and the offer accepting process has started. If the same offer is returned by the applicant with the additional missing documents, and the validation is successfull, the status changes from validating to accepted. The same offer will not be marked as *O_Returned* twice.
- B5->B4 [*O_Returned - O_Accepted*]: The waiting time after the validation process has finished and the appropriate process for uncompleted applications has started. If the same offer is sent only once and it was accepted, the status changes from *O_Returned* to *O_Accepted*. The same offer will not be marked as *O_Returned* twice.

- B5->B6 [*O_Returned - A_Incomplete*]: The waiting time after the validation process has finished and the appropriate process for uncompleted applications has started.
- B6->B4 [*A_Incomplete - O_Accepted*]: The waiting time before an uncompleted offer is accepted.

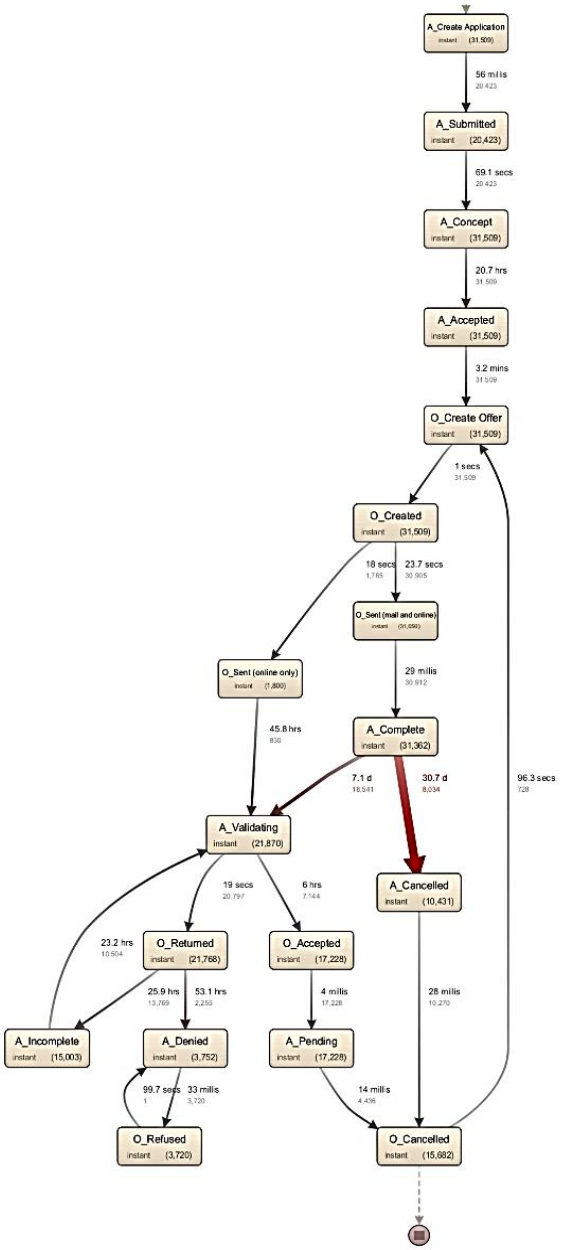

Fig. 9. Median duration and instance frequency of data set.

Table 4. Mean and median Duration (in days) of bank-related process parts.

ID	From activity	To activity	Median duration in days	Mean duration in days	Instance frequency
B1 -> B2	*A_Concept*	*A_Accepted*	0,86	1,41	31.509
B3 -> B4	*A_Validating*	*O_Accepted*	0,25	0,91	7.144
B5 -> B4	*O_Returned*	*O_Accepted*	1,09	2,05	5.290
B5-> B6	*O_Returned*	*A_Incomplete*	1,08	2,00	13.769
B6-> B4	*A_Incomplete*	*O_Accepted*	3,70	5,90	4.781

By comparing the median and mean duration in each of the process parts we notice that the mean is approximately twice as long as the median. This suggests that the dataset has outliers with relatively high durations, causing the mean to be higher. Thus, the median is more robust than the mean and will therefore be used for further analysis in this section. For example, it takes on average over 1,4 days to accept an offer that was created, but 50 percent of all applications only take 0,86 days or.

Due to the high instance frequency of the process part B1 -> B2 (31.509), lowering the waiting time by a small percentage would have a big effect in the overall throughput time. Here we need to differentiate between applications submitted via the website and applications created at the bank. For the latter, there is a median wait time about four minutes. The median wait time for applications submitted by website is 29 hours (1,2 days). From this we deduct that 50 percent of all applications submitted via website sit for 1,2 days or more before they are picked up and processed, and hence, the bank loses more than one day to get back to the customer to make an offer. For one hour saved in B1->B2 the applications submitted via website (20.423 applications) overall throughput time can be reduced by up to 851 days. One of the possibilities to reduce the waiting time is to employ more people responsible for the offer creation process. On the other hand, the fixed costs for labor will be higher. Another option is process automation. The bank could implement artificial intelligence tools that use previous data to automatically create one or more offers. Additionally, checks could be performed for the acceptance process of the application in an automated fashion, making use of robotics or a process automation tool. Moreover, one could implement a system which ensures that applicants check that the application is in order and meets all acceptance criteria. The bank could request more documents during the application creation part for the AI tool to make better decisions.

Applications with missing documents cause relatively high waiting times. If we take both process parts, B5->B6 and B6->B4, into account, they have median waiting times of 1,08 and 3,70 days, respectively. If the bank were to reduce the number of iterations due to document incompleteness, the waiting time would be reduced considerably. One of the measures the bank could take, is to initially request that applicants send additional documents. Partial process automation is a possible solution. The bank could either identify loans with certain properties (e.g. loan goal = home improvement, loan amount <= 5.000) or certain applicant profiles (age 30-40, income > 2.000 etc.) If the loan

properties and the applicant profile match, the loan is granted, if not, further validation/document is needed.

Process parts B3->B4 and B5->B4 cannot happen at the same time. If an offer is sent only once (there were no missing documents) and the loan is granted, then process part B5->B4 occurs. If, however, the offer is sent more than once due to incomplete documents, and the loan is granted, then process part B3->B4 is executed. The fact that the waiting time in B5->B4 is higher (one-day median duration) than the waiting time in B3->B4 (median 0,25) is due to the fact that when an offer comes back to the bank for a second or third time, part of the validation has already been done in the previous validation steps.

4.2 Identify and analyze process parts where an application is waiting to be processed by the applicant

In addition to process parts where an application is waiting to be processed by the bank, there are also process parts where the bank is waiting for the applicant to send an offer, missing documents, etc. We have identified four parts of the process and considered them as relevant for further analysis. Those process steps are:

- A1->A2 [*A_Complete - A_Validating*]: Bank has sent the offer, and is waiting for the applicant to sign the documents and send them back.
- A3->A2 [*A_Incomplete - A_Validating*]: Bank is waiting for the applicant to send missing documents in order to continue with the validation process.
- A1->A3 [*A_Complete - A_Canceled*]: Waiting time for a response from the applicant before the application is canceled.
- A1->A4 [*A_Complete - O_Create Offer*]: Waiting time before the bank creates a second offer for the same application.

Table 5. Duration of the process parts where the bank is waiting for input from the applicant.

ID	From	To	Median Duration	Mean Duration	Instance Frequency
A1 -> A2	*A_Complete*	*A_Validating*	7,1	8,8	31.509
A3 -> A2	*A_Incomplete*	*A_Validating*	1,0	2,5	10.504
A1 -> A3	*A_Complete*	*A_Cancelled*	30,3	27,7	8.034
A1 -> A4	*A_Complete*	*O_Create Offer*	4,1	7,2	4.135

The median and mean duration of process parts A1->A2 and A1->A3 is similar. In contrast, process parts A3->A2 and A1->A4 show that the median is considerably longer than the mean, indicating a skewed distribution, possibly with outliers with a high waiting time.

The process part with the highest waiting time is A1->A3. These are applications where the bank sends out the offer and waits for a response by the applicant. After the offer was sent, the bank does not contact the applicant again, and after 30 days of no

response the offer is canceled. The analysis of other chapters (see conversion rate analysis based on incomplete files) shows that if the bank contacts the applicant, the conversion rate is higher on average. On the other hand, as seen in process part A1->A3 (Figure 10), if the bank does not contact the applicant at all after sending out the offer, the applicant might forget to return the offer and after 30 days this offer is canceled.

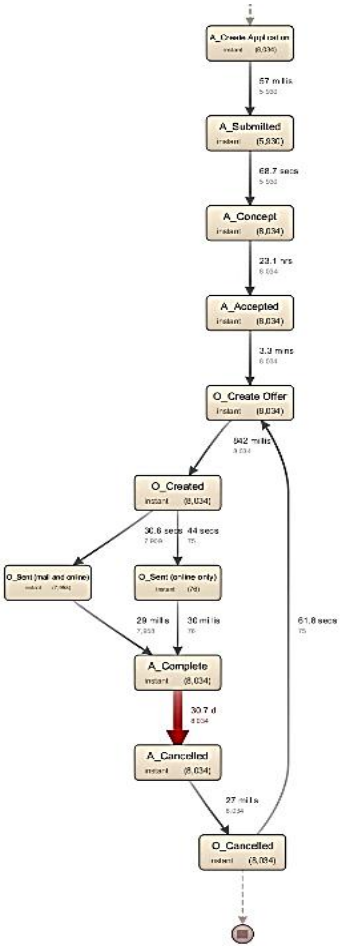

Fig.10. Median duration and Instance frequency of canceled offers.

According to process part A1->A2, applicants send in the requested documents one week after the offer has been sent out. To prevent having to cancel an offer, the bank could remind applicants by contacting them via phone or mail one week after the offer was made, if there is no response on the part of the applicants. Another possibility to increase the response rate of applicants is to use incentives. The bank could grant a cash back for loans whose offer is sent back to the bank within the first week.

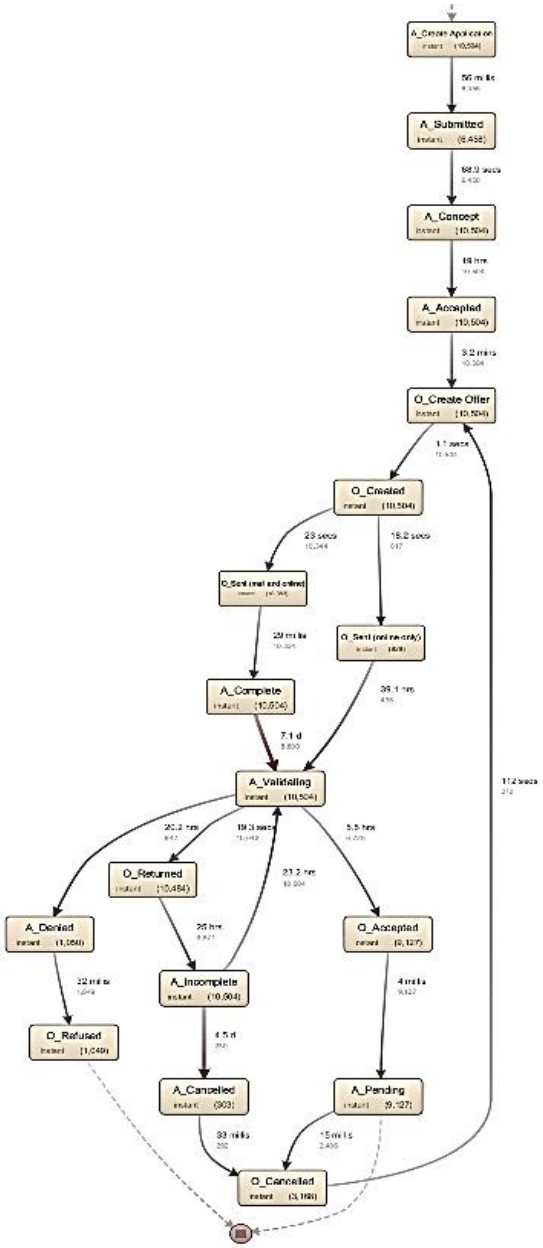

Fig. 11. Median duration and instance frequency of offers with incomplete files.

Process part A1->A2 contains offers that are sent online only (*O_Sent* (online only)) and offers that are sent online and per mail (*O_Sent* (online and mail)). Data analysis shows that on average, offers that are sent online only take one day less to be sent back to the bank. For every offer sent online only, the throughput time of the whole process

is reduced by one day. The data also shows that most of the offers are sent online and per mail (cf. figure 9). Out of 31.509 applications, only 830 have offers sent online only (only three percent). The bank could encourage applicants to receive offers online by offering incentives.

Figure 11 shows that the incompleteness of documents also increases the waiting time for input from the applicant. This is understandable since in these instances the applicant must send in missing documents and a part of the loan application process must be repeated. A closer look at this process part (figure 11) shows that applications with missing documents have a better conversion rate. By analyzing and identifying more documents that are essential for the decision to accept or decline an offer, and by requesting these documents when the offer is first sent out, the throughput time can be reduced and conversion rate can be increased. At the same time, the applicant is more satisfied since the bank responds faster and they are not asked to send in more documents or wait longer for a response.

4.3 Findings and recommendations

The throughput time of the loan application process includes waiting times from process parts where the bank is waiting for input from the applicant and waiting times from process parts were the applications are waiting to be processed by the bank. We analyzed both process parts separately by focusing on the process parts with the most impact in the overall throughput time of the process. Below we give an overview of the main findings followed by recommendations.

Finding 1: Applications where the customer is never contacted after the offer was sent have a high probability of getting canceled because the applicant is most likely not going to return the offer. We recommend that the bank reminds applicants about received offers on a weekly basis. The additional touchpoint between the applicant and the bank might have a big effect on the conversion rate.

Finding 2: An application submitted via website waits around one day before a user picks it up to start the offer creation process. Since this process part has a high frequency (every application has at least one offer), a reduction of the waiting time by a small percentage can have a big effect in the overall throughput time of the process. We recommend that the bank automates this process part. This can be achieved by implementing tools (e.g. artificial intelligence tools) that are fed with historical data and determine the information needed to not only create and submit an application but also create at least one offer. The bank can offer chat bots during the offer creation process to assist applicants. The applicant should also have the possibility to call a user (bank employee) if the applicant has any questions. Through the automation of the offer creation process part, the bank saves resources and the throughput time is reduced by eliminating the one-day waiting period for a bank user to process the application. Also, customer satisfaction is increased, by enabling them to instantly create an offer without waiting for the bank to call the them.

Finding 3: Every time the order is sent back to the applicant because of missing documents, the validation process part is repeated, thus increasing the throughput time of the process. The bank should analyze and identify documents that are frequently

missing and which are essential for the decision of whether or not to accept the offer. As part of the automation of the offer creation process (finding 2), the bank can mark the essential document as mandatory. Hence, the applicant cannot submit an application without these documents. Moreover, the bank could even grant loans automatically. With the help of AI Tools, the bank can create applicant profiles and loan profiles based on historical data. If the profiles match the loan is granted, without further validation.

Finding 4: For every offer that is sent out via email only, the throughput time of the loan application process is on average one day shorter. Currently, offers that are sent online only make just three percent of all the offers sent. By offering incentives, the bank could push the applicants to receive offers online only.

5 Conversion Rate Analysis Based on Incomplete Files

This section addresses the second question of the BPI Challenge: "*What is the influence on the frequency of incompleteness to the final outcome. The hypothesis here is that if applicants are confronted with more requests for completion, they are more likely to not accept the final offer.*" By reducing the requests for missing documents, the bank could benefit in two ways if the hypothesis is in fact true. Firstly, costs could be reduced due to less application tracking, fewer calls, and less resource usage. Secondly, if fewer requests for missing documents are needed the conversion rate would increase, which would boost the sales rate. To address the question, we pursued the following steps:

1. Identify the number of requests for additional documents for each application
2. Calculate conversion rates based on the number of request for additional documents
3. Identify additional patterns

5.1 Identify the Number of Requests for Additional Documents for each Application

To identify process instances where applicants were asked for additional (missing) documents, we look for the activity *A_Incomplete*. If this activity never occurs, no documents were missing; if the activity occurs in the process, the number of occurrences is exactly the number of requests for additional documents.

As shown in figure 4, this was true for 15.003 application processes of the overall 31.509 instances. This indicates that for 16.506 application processes the documents provided were sufficient, which is approximately 52 percent of the instances. In 9.317 instances, or 30 percent of the total instance log, additional documents were requested once, 3.970 instances or 13 percent of all instances had documents which were requested twice, 1.234 or 4 percent of the instances had documents which were requested three times, and the highest number of requests to add missing documents was seven times, which occurred in nine instances. Figure 12 depicts the distribution of the instances as a combination of the number of instances and the number of requests to supply missing documents. In 95 percent of all application processes, the bank requested additional documents not at all, once or twice.

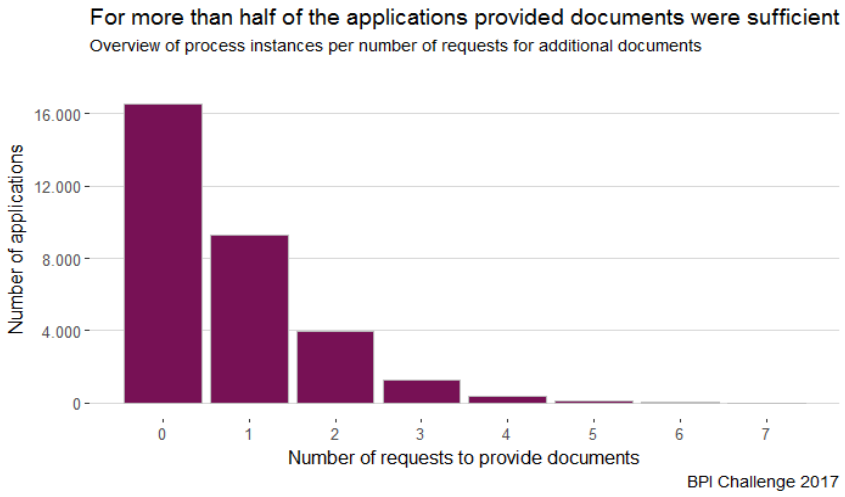

Fig. 12. Overview of process instances per number of requests for additional documents.

5.2 Calculate conversion rates based on the number of document requests

As noted previously, a loan application process is successful once the offer was accepted by the applicant, which is denoted by the activity *A_Pending*. The data shows a conversion rate of around 54,7 percent, which means that 17.228 of 31.509 applications were successful. Figure 13 shows the endpoints with the respective frequencies.

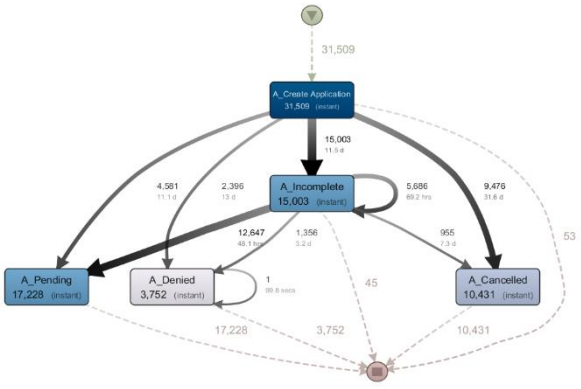

Fig. 13. Process end points and their frequencies.

The following figure 14 then shows the conversion rate in combination with the number of requests of (the submission of) missing documents. While application process instances with complete documents make for 52 percent of all applications, the

conversion rate is considerably low at 28 percent, compared with the overall rate of 55 percent. Application process instances, for which the bank requested additional documents at least once, the conversion rate is around three times higher, i.e. 84 percent. For instances with more than one request to provide documents, the conversion rate is around 85 percent. The mean conversion rates for applications with six or seven requests for additional documents stray up and down and are not considered meaningful because of their low frequencies.

This analysis contradicts the bank's hypothesis that the more applicants are confronted with requests the less likely they are to accept an offer. Rather the opposite seems to be the case. On average, the conversion rate is quite low if no request was made. For one or several requests, the conversion rate is considerably high.

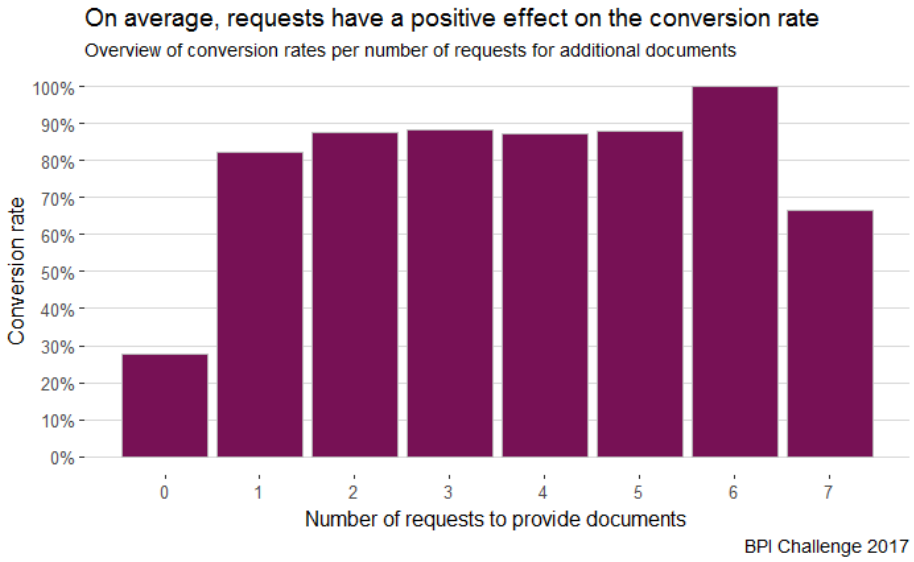

Fig. 14. Overview of conversion rates with number of requests for additional documents.

5.3 Identify additional patterns

In addition, we searched for patterns that would further explain the drivers behind increased conversion rates: is the behavior similar or can we narrow it down further to more specific contexts? The log shows information regarding loan goals, loan amounts requested, and application types, which are all used in our analysis.

Loan goals. The analysis of missing documents can be distinguished by different loan goals. One might suppose that missing documents occur more often for certain categories than others. Only the seven most common loan goals were considered: car (9.328 instances), existing loan takeover (5.601 instances), home improvement (7.669 instances), remaining debt home (842 instances), not specified (1.065 instances), other, see explanation (2.985 instances), and unknown (2.365 instances). These make for 95

percent of all instances. To enhance readability, the last three loan goals mentioned are summarized as 'other'.

Figure 15 shows the conversion rates for each aforementioned loan goal per number of requests. It clearly shows that for instances with no requests to provide further documents, the conversion rates are low. Instances with the loan goal 'remaining debt home' at around 13 percent have a much lower conversion rate than those from 'home improvement' at around 33 percent, and hence, a spread of 20 percentage points. For instances with requests to provide documents the conversion rates then narrow to low levels of variation from 82 to 86 percent, i.e. a spread of 4 percentage points. We can learn two things from this. First, the conversion rates for 'home improvement' are the highest if at least one request was made, and the lowest if no request was made. Second, the conversion rates for all loan goals improve considerably if at least one request was made, and hence, the pattern is independent of the loan goal.

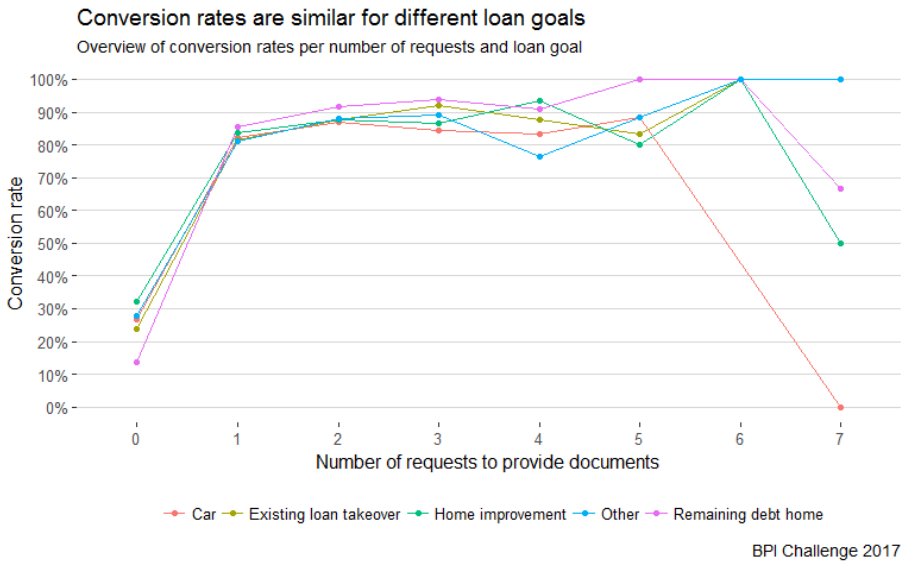

Fig. 15. Overview of conversion rates per number of requests and loan goal

Requested amount. Another data point worth inspecting is the relationship of requested credit amount and number of requests for additional documents. Figure 16 shows that there is indeed a positive relationship. We can deduce that the larger the requested amount, the greater the need for documents that would minimize the risk of loan default, such as information about bails or securities.

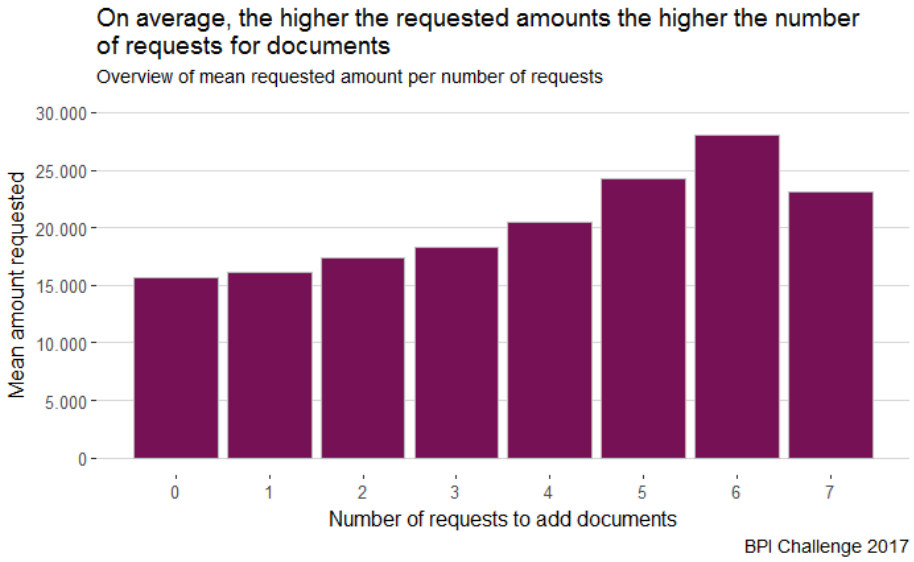

Fig. 16. Overview of mean requested amount per number of requests.

Application types. Another combination worth investigating is the conversion rate for applications depending on whether the loan is a limit raise or a new credit. Figure 17 shows that there is indeed a trend: increasing an existing loan leads to accepted offers in 65 percent without requests for additional documents. Compared with the conversion rate for new credits with no request for additional documents, the conversion for new credits is remarkably low at 20 percent. In 65 percent of instances, increasing an existing loan results in an accepted offer without a request for additional documents.

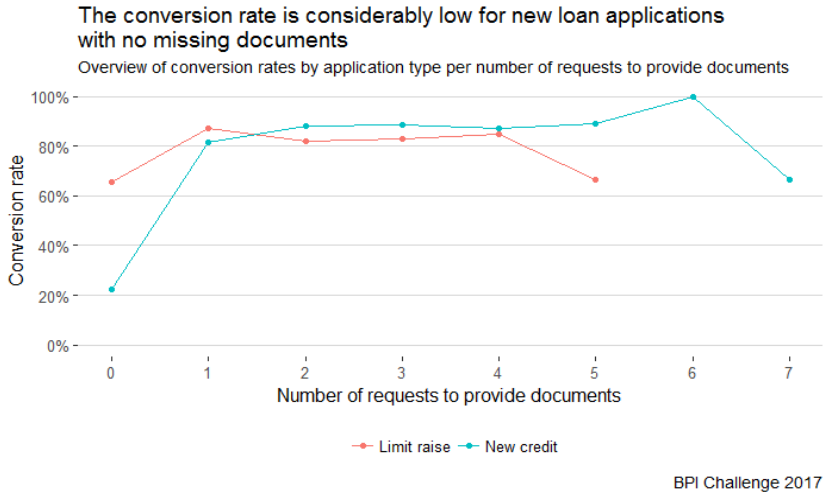

Fig. 17. Overview of conversion rates by application type per number of requests.

5.4 Findings and recommendations

The analysis has shown that conversion rates are not adversely affected by requests for additional documents. On the contrary, once an applicant is approached, even to provide further documents, the likelihood of an accepted offer on average increases by 2,5 times. Therefore, we could not prove the hypotheses, that a higher number of requests negatively influences the conversion rate.

However, from a cost perspective this part of the process should still be optimized, as every incidence of incomplete documents leads to a new validation and more calls following an offer. These costs can potentially be reduced by optimizing the process in such a way that applicants are informed of what kind of documents to provide, thereby reducing the iterations. This potentially applies to all instances with missing documents, which are around 50 percent. It is safe to assume that these repetitions are an extra cost factor. Due to these findings, we recommend the following:

1. Based on additional data (not provided within the given data set), determine the kind of documents which are missing the most often. Use this to compose a questionnaire or checklist for applicants and bank employees to make sure that all documents are provided before the offering process. This might reduce time and cost by preventing extra work for requesting documents, as well as improve applicant satisfaction.
2. Based on the information that on average, the conversion rate increases per additional request for missing documents, we suggest reassessing the customer journey – independently of any missing documents – and implementing additional touchpoints with applicants to increase the conversion rate.

6 Conversion Rate Analysis Based on Number of Offers

This section elaborates on the third question of the BPI Challenge: "*How many customers ask for more than one offer (where it matters if these offers are asked for in a single conversation or in multiple conversations)? How does the conversion compare between applicants for whom a single offer is made and applicants for whom multiple offers are made?*". To tackle these questions, we pursue the following steps:

1. Identify number of applications with more than one offer
2. Differentiate between whether offer was made in a single or in multiple conversations
3. Identify the overall conversion rate
4. Identify conversion rate for single-offer applications and multi-offer applications

6.1 Identify number of applications with more than one offer

From a total of 31.509 process instances, all instances had at least one offer. 8.559 instances had more than one offer (27 percent). This means that for 22.950 instances only one offer was made. For 6.578 applications, exactly two offers were made, for 1.348 applications exactly three offers were made, for 443 applications exactly four

offers were made, for 126 applications exactly five offers were made, until for two applications exactly ten offers were made. See figure 18 for an overview.

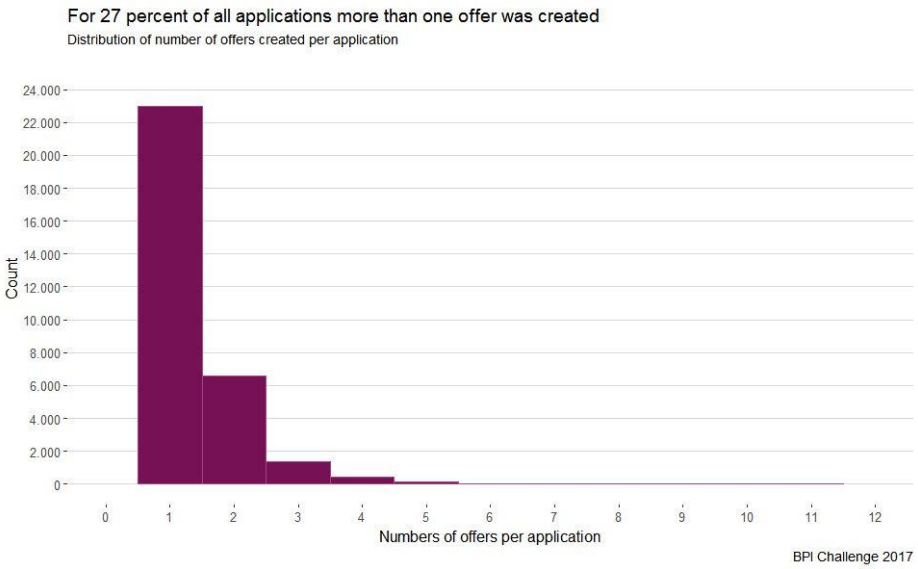

Fig. 18. Process instance count per number of offer per application.

6.2 Differentiate between whether offer was made in a single or in multiple conversations

In order to further distinguish between whether offers were made in a single conversation or in multiple conversations we looked into the order of specific activity flows. If an offer is created (*O_Create Offer* and *O_Created*) and directly followed by another offer creation, it indicates single-conversation-offer applications. On the other hand, if one or more offers are created, the activity *A_Complete* indicates the end of the conversation. If an offer is created after the activity *A_Complete*, it indicates an additional conversation, and hence, is considered a multi-conversation-offer application. Figure 19 shows the activity flow that was filtered accordingly, i.e. the reduced number of activities shown and a filter rule that says to keep only process instances where the activity *O_Created* is eventually followed by *O_Create Offer*. The data shows that for the 8.559 applications with more than one offer **5.882 applications were multi-conversation-offers** and **2.677 were single-conversation-offers** (not shown in the figure)**.**

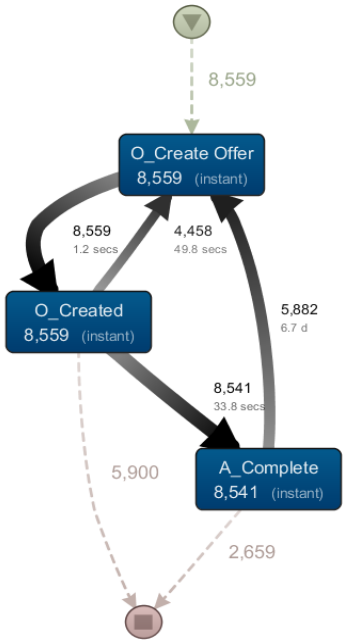

Fig. 19. Overview of process flows regarding single- and multi-offer applications.

6.3 Identify the overall conversion rate

We define the conversion rate as the ratio between the number of applications where an offer was accepted by the applicant and the number of all applications. Furthermore, if the activity *A_Pending* is reached in the process it is considered that the applicant accepted an offer. Table 3 shows that this is true for 17.228 applications. Considering a total of 31.509 applications, the **overall conversion rate is 54,7 percent.**

6.4 Identify conversion rate for single-offer applications and multi-offer applications

Now that the overall conversion rate is known and we are also in the position to distinguish between single- and multiple-offer applications, and also distinguish between single- and multiple-conversation applications for multiple-offer applications, we are able to calculate the conversion rate for each part. Figure 20 shows filtered process flows, one for single- and one for multi-offer applications. For each flow, frequencies for specific endpoints are shown that allow us to differentiate whether an application was successful (i.e. *A_Pending*) or not (i.e. *A_Cancelled*, *A_Denied*, other).

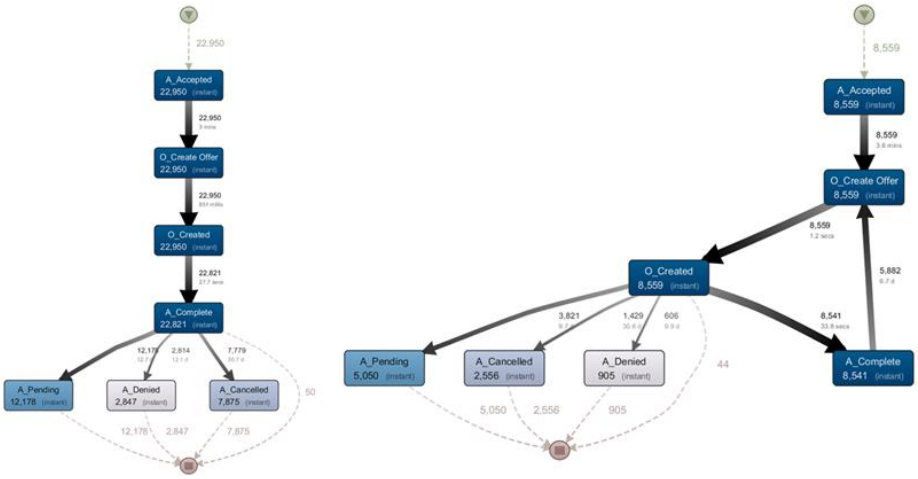

Fig. 20. Filtered process flows, left single-offer applications, right multi-offer applications.

The numbers are shown in table 6. Single-offer applications occur 2,7 times more often than multi-offer applications. For single-offer applications, the conversion rate is 53,1 percent, and therefore slightly lower than the overall rate (54,7 percent). **Multi-offer applications show a rate of 59,0 percent, and are thus almost six percentage points above the single-offer application rate.**

Table 6. Process instance distribution via single- or multi-offer applications and end points.

Offers/ Out-come	Application denied	Application canceled	Application pending	Other end points	Sum	Conver-sion rate
Single-offer applications	2.847	7.875	12.178	50	**22.950**	**53,1%**
Multi-offer applications	905	2.556	5.050	48	**8.559**	**59,0%**
All	**3.752**	**10.431**	**17.228**	**98**	**31.509**	**54,7%**

In the next step, we show the differentiation between single- and multiple-conversation offers. This can be obtained by applying the same filtering rules as mentioned at the beginning of this section. Table 7 shows that there are twice as many multiple-conversation-offer applications than single-conversation-offer applications. Also, multiple-conversation-offer applications show a 65,0 percent conversion rate that is six percentage points above the multiple-offer application rate and over ten percentage points above the overall conversion rate. Multiple-offer single-conversation application conversion rate is 45,9 percent and low compared to the overall conversion rate (54,7 percent).

Table 7. Distribution for single- or multi-conversation applications and end points.

Offers/ Outcome	Application denied	Application canceled	Application pending	Other end points	Sum	Conversion rate
Multi-offer-single conversation applications	303	1.141	1.229	4	**2.677**	**45,9%**
Multi-offer-multi conversation applications	602	1.415	3.821	44	**5.882**	**65,0%**
All	**905**	**2.556**	**5.050**	**48**	**8.559**	**59,0%**

6.5 Findings and recommendations

Overall, the number of applications with exactly one offer is high compared to the overall number of applications (72,8 percent), i.e. 22.950 of 31.509 applications. The data gives no indication whether this is based on the preference of the applicants or the bank itself. However, the data on average shows a better conversion rate for applications with more than one offer, i.e. 59,0 versus 53,1 percent of successful applications. An even higher conversion rate has applications with more than one offer that were discussed in more than one conversation: a 65,0 versus 45,9 percent conversion rate.

One interpretation would be to always make applicants more than one offer. For example, one could use available data and derive often selected offers based on amount, loan goal, duration, and applicant-specific metrics (e.g. location), then handing out such standard offers next to the one that was actually discussed (along with the information of why this standard offer is of value to most other customers).

A different interpretation would be that the number of offers is not the only driver for a better conversion rate in comparison with the number of contacts with the applicant in the customer journey. Table 7 clearly shows that the segment of applications with multiple offers and multiple conversations performs best with a rate of 65,0 percent. One suggestion would be to always contact applicants after one or more offers were made in the first conversation, just to check and to support the applicant in any way as an additional touchpoint. Based on the interpretations, we recommend the following:

1. Derive a set of standard template-offers for different loan goals and customer segments that have been successful in the past. Encourage employees to discuss and offer such templates to promising applicants.
2. Change the current process in such a way that after the initial conversation, where one or more offers were made, applicants are contacted after a certain amount of time to follow-up on the offers if applicants did not get back (cf. previous section).
3. Study the changed behavior and resulting conversion rate carefully and interpret the results. If necessary, make adjustments.

7 Conclusion

In this report, we showed our understanding of how to approach data-driven process analysis projects. In particular, we demonstrated our method as well as one possible toolset to perform the analysis. The context of the analysis was given by the data provided by a Dutch bank for loan applications. The data spans from January 2016 until February 2017 with a total of 31.509 application instances.
We investigated in four different directions. Firstly, we analyzed the general process structure and considered the three questions provided by the bank. In the general analysis, we learned that the process has a total of 4.047 variants but the 75 most frequent variants cover over 70 percent of all applications. Furthermore, we showed that the process is either initiated by an applicant at the bank or via the bank's website. Each applicant gets at least one offer that is either accepted, denied or never returned. We also learned that on average, applications get processed within 21 days, and that the drivers behind this value are the long wait times for applicants to return an accepted offer and that on average, applications submitted via website are not processed for over 20 hours. We also considered loan goal categories, where several categories, namely 'unknown', 'not specified', and 'other' complicated further analysis.

Secondly, we looked into process performance, especially regarding wait times on the part of the bank as well as wait times on the part of the applicant. We focused on nine process parts where waiting time and instance frequency were considered for analysis purposes. In five of these process parts, the application is waiting to be processed by the bank (employee or system), and in the other four process parts, the bank is waiting for input from the applicant. Our analysis shows that if an applicant is not contacted after he or she received the offer and the offer is not sent back to the bank, this offer will never be sent back and will be canceled after one week. We also found out that applications created via website wait one day before a bank’s employee or system picks them up for further processing. Furthermore, we found that the incompleteness of documents increases the waiting time by approximately five days. We also learned that it takes one day less for offers that are sent online only to reach the bank, compared with offers sent out via online and mail.

Thirdly, we looked into whether asking applicants for missing documents has an impact on the conversion rate. The analysis showed that the conversion rate rises from 30 percent, where no documents are missing, to around 85 percent where documents were requested, regardless of whether there were six requests to provide missing documents or just one. Hence, we cannot confirm the bank's hypothesis that the conversion rate decreases when making requests to applicants. Nevertheless, we believe that costs for processing these requests could be reduced if applicants hand in required documents during submission.

Fourthly, we investigated whether the number of offers for one application has an impact on the conversion rate. We found that the overall conversion rate is 55 percent (17.228 applications out of 31.509). The conversion rate for applications with exactly one offer is 53 percent (12.178 out of 22.950 applications). For applications with more than offer per application, the conversion rate is higher, i.e. 59 percent (5.050 out of 8.559 applications). However, when we further distinguish multiple-offer applications

into whether offers were made in one or multiple conversations we learn that multiple-conversation offer applications perform better than single-conversation offer applications, 65 and 46 percent, respectively. With the overall analysis, we make the following recommendations to the bank:

1. Shorten the timespan between application submission via website and completion of the application. Implement an AI Tool to automate acceptance, and where possible, the offer creation process. A decrease in waiting time until the application is accepted and an offer has been created will have a big impact on the whole process performance.
2. Merge the categories 'unknown', 'not specified', and 'other' into one category and encourage employees to make sure to specify the correct loan goal.
3. Based on additional data (not provided within the given data set), find out what kind of documents are missing most often. Use this to compose a questionnaire or checklist for applicants and bank employees to make sure that all documents are submitted before the offering process. This might reduce time and cost by preventing extra work for requesting documents and improve applicant satisfaction.
4. Based on the information that on average, the conversion rate is higher per additional request for missing documents or additional offers, we suggest reassessing the customer journey and implementing additional touchpoints with applicants to increase the conversion rate.
5. Increase the number of offers that are sent online only. For each offer sent online only the throughput time is reduced by one day, on average.
6. Derive a set of standard template offers that were successful in the past for different loan goals and customer segments. Encourage employees to discuss and offer such templates to promising applicants.
7. Another field to look at would be to understand how the bank's employees are measured and what effect this would have on the application process, and try to realign these Key Performance Indicators (KPIs) to improve the overall process outcome.

References

1. W. van der Aalst, et.al., “Process Mining Manifesto,” in *BPM WS*, 2012, pp. 169–194.
2. C. W. Günther and A. Rozinat, “Disco: Discover your processes,” in *CEUR Workshop Proceedings 940*, 2012, pp. 40–44.
3. G. Grolemund and H. Wickham, “R for Data Science,” *O’Reilly*, 2016.
4. M. Hofmann; and R. Klinkenberg;, RapidMiner: Data Mining Use Cases and Business Analytics Applications. Chapman and Hall/CRC, 2013.
5. “Celonis website,” [Online]. Available: http://celonis.com/. [Accessed: 31-May-2017].
6. R. Cody, Learning SAS by Example: A Programmer’s Guide, vol. 50. 2007.
7. A. Rozinat and W. M. P. van der Aalst, “Decision Mining in ProM,” in *Proceedings of the 4th Int. Conf. on Business Process Management (BPM 2006)*, 2006, vol. 4102, pp. 420–425.
8. “R Notebooks.” http://rmarkdown.rstudio.com/r_notebooks.html. [Accessed: 04-Jun-2017].
9. B. F. van Dongen, “BPI Challenge 2017 dataset,” 2017.