BPI Challenge 2017
Process Variability Analysis over Offers

Alfredo Bolt

Eindhoven University of Technology, Eindhoven, The Netherlands
a.bolt@tue.nl

Abstract. The "Offer" process is described and characterized using basic process mining and statistical techniques, answering some (but not all) of the questions asked by the company that provided such event log. The main content of this submission is narrower and is oriented towards the identification of variability reduction points and their location within the process. Concretely, characterized process variants were identified using a process-variant detection technique developed by the author. Such variants are analyzed and interpreted without any type of knowledge about the process, therefore some interpretations are built based on assumptions (which are clearly identified and explained).

1 Introduction

The BPI Challenge provides participants with a real-life event log, and they are asked to analyze this data using whatever techniques available, focusing on one or more of the process owner's questions or proving other unique insights into the process captured in the event log. This year, the data was obtained from a financial institute (i.e., **The Company**) and relates to a loan application handling process. The basic flow of applications is the following: Customers apply for a loan for which the company sends them one or more offers. If the customer accepts an offer, the loan is approved. Loan applications and offers can also be cancelled at any time, and applications can be rejected according to certain criteria.

As state in this year's BPI Challenge website[1], the company is particularly interested in answers to the following questions:

- **Q1:** 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 the user and the time spent waiting on input from the applicant as this is currently unclear.
- **Q2:** 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.

[1] BPI Challenge 2017 website: https://www.win.tue.nl/bpi/doku.php?id=2017:challenge

- **Q3:** 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?
- **Q4:** Any other interesting trends, dependencies etc.

The event log provided for this year's challenge contains all applications filed in 2016, and their subsequent handling up to February 2nd 2017. In total, there are 1,202,267 events contained in 31,509 loan applications. For these applications, a total of 42,995 offers were created. There are three types of events, namely *Application* state changes, *Offer* state changes and *Workflow* events. There are 149 originators in the data, i.e. employees or systems of the company.

The data is provided in two files:

- *The Application event log*: This event log contains all events with the application as the case ID. Any event related to an offer also refers to an OfferID.
- *The Offer event log*: This event log contains all events related to offers, with these offers as case ID. For each offer, a corresponding application is available. Please note that there may be multiple offers per application. However, at most one of them should always be accepted.

The scope of this paper is focused on analyzing the relation between all the data attributes mentioned above and the acceptance of an offer. Therefore the focus is only on the Offers event log, for which Q1 and Q4 will be answered only in the context of offers. Note that Q2 and Q3 apply mostly to applications, hence they are out of the scope of this paper.

The Offers log has the following data attributes for each event:

- An offer ID
- The state of the offer
- The offered amount
- The initial withdrawal amount
- The number of payback terms agreed to
- The monthly costs
- The credit score of the customer
- The employee who created the offer
- Whether the offer was selected by the customer
- Whether the offer was accepted by the customer

The event log was preprocessed using a RapidMiner workflow (available upon request) which extends the list of data attributes mentioned before with the following derived data attributes:

- The interest rate, calculated as the total payment ($monthlyCost * numberOfTerms$) divided by the offered amount
- The withdrawal rate, calculated as the initial withdrawal amount divided by the offered amount

- The elapsed time of an offer, calculated for each event as the time difference between the current event and the first event related to that offer (i.e., creation)
- The state duration, calculated as the time spend in a current offer state.
- The next state, calculated for each event (except the last event related to an offer) as the state corresponding to the next event related to an offer (sorted by timestamp)

The remainder of this paper is organized as follows. First, Section 2 introduces the process of the lifecycle of offers, and characterizes it using basic analysis and statistics, answering Q1. Then, Section 3 discusses the concrete insights that were found using a specific process mining tool, answering Q4 without using any domain knowledge. Finally, Section 4 reflects on these results and concludes this paper.

2 Basic Analysis of Offers

This section describes general properties of the process such as frequencies and performance of the process. In this paper, we use transition systems to represent the lifecycle of offers. Transition systems are composed by states and transitions between them, and their formal definition is out of the scope of this paper. Transition systems can be annotated with event attributes that can be used for process analysis such as prediction or comparison.

Figure 1 illustrates the lifecycle of offers as a transition system, where the thickness of states (nodes) and transitions (arcs) represents the percentage of offers that reach such state. In other words, thicker elements indicate that they are more frequent.

The Offer handling process is described as follows. First, an offer is created ("O_Create Offer" and "O_Created"). Note that these two states have a 100% of frequency i.e., they are reached by all offers. We can deduce that this is just a duplication and they correspond to the same concept. Then, the offer is sent to the customer. In total, 97% of the offers are actually sent to the customer. Note that the remainder of the offers are either cancelled ("O_Cancelled") or refused ("O_Refused") before they are even sent. There are two ways to do this depending on the media channel used: online or physical mail. 4.71% of the offers are sent via online only ("O_Sent (online only)"), and 92.35% of the offers are sent via online and physical mail ("O_Sent (mail and online)").

Once the offer has been sent, it can be returned by the customer ("O_Returned"), cancelled or refused. In total, only 54.2% of the offers are returned. When a customer returns an offer, it can be accepted ("O_Accepted"), refused or cancelled. Note that, in total, 40% of all the offers are accepted, 11% are refused and 49% are cancelled. Note that an offer can be accepted only if it is returned. If an offer is returned, the most likely outcome is acceptance (73% of the returned offers are accepted).

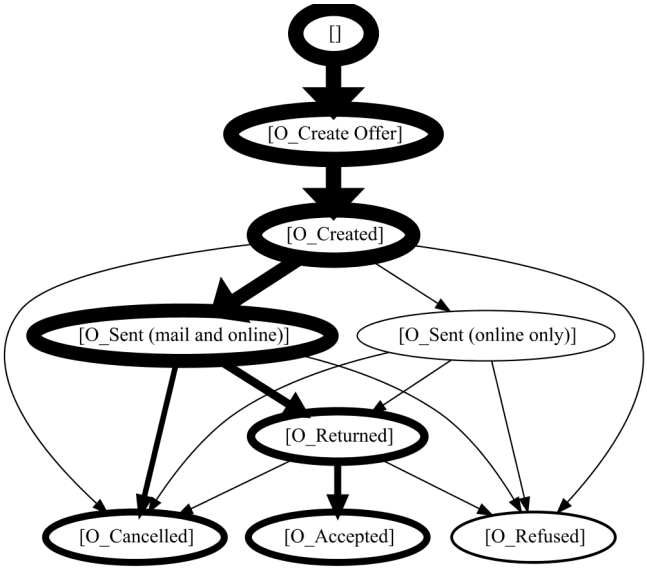

Fig. 1: Transition system representing the lifecycle of offers. Thickness indicate frequency. Thicker "paths" indicate the most frequent behavior.

Also note that a small percentage of the sent offers are refused without being returned (2.5% of all offers). This could be caused by customers being contacted via phone call, stating that they refuse the offer, hence not returning it.

Figure 2 show an alternative representation of the lifecycle of offers as a transition system, where the thickness of states (nodes) represent the elapsed time of an offer since it has been created until it has reached that state, and the thickness of transitions (arcs) represents the difference between the elapsed times of the two states (i.e., duration).

The time difference between the states "O_Create Offer" and "O_Created" is 1 second in average. This confirms our supposition that both states relate to the same activity and are probably bind by system-level triggers.

The reader can note that the thickest state corresponds to an order being cancelled ("O_Cancelled"). This means that cancelled offers are the ones that in general have the longest throughput time (23 days in average). Note that there are considerable time differences depending on which was the last state before the cancellation. The throughput time of Offers that are:

- Cancelled after being created is 5 days in average.
- Cancelled after being sent via online only, is 17 days in average.
- Cancelled after being sent via online and physical mail is 25 days in average.
- Cancelled after being returned is 24 days in average.

Also note that once a customer returns an offer, it takes some time to be processed by the company. Concretely, in average it takes 15 days to realize that

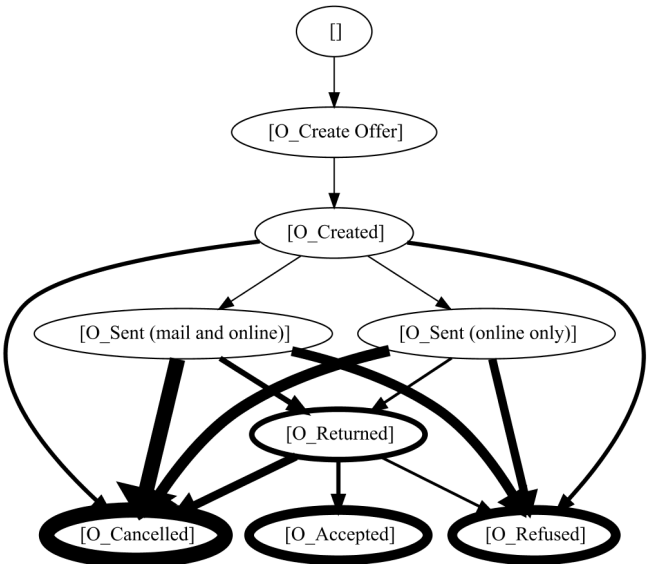

Fig. 2: Transition system representing the lifecycle of offers. Thickness indicate elapsed time for states and duration for transitions.

a returned offer is cancelled, 6 days to realize that a returned offer is accepted and 4.8 days to realize that a returned offer has been refused. This processing time could be improved by the company in order to take more immediate action if needed. For example, if a customer rejects an offer, a new offer could be sent the same day to maintain the attention and interest of the customer.

Finally, note that the time it takes for a customer to return an offer also depends on the way it was sent: offers sent via online only are returned in 3 days (avg), whereas offers sent by physical mail and online are returned after 9 days (avg).

Analysis of the "Accepted" and "Selected" Data Attributes

The process description states that offers contain, among other attributes, wether an offer was *selected* and/or *accepted* by the customer. To validate such data attributes, we initially filtered the event log to keep only those offers that were selected by the customer. From the 42,995 offers recorded in the log, only 21,768 were marked as *selected* (i.e., the attribute "Selected" has a value "true").

Figure 3 shows the count of all the reached states (i.e., activities) of this filtered event log. Note that, even all offers are returned, not all offers were accepted (several offers were cancelled or refused). There are no loops in the process, so offers that are refuser or cancelled cannot be accepted afterwards. Therefore we cannot rely on the "Selected" attribute to consider if an offer was accepted or not.

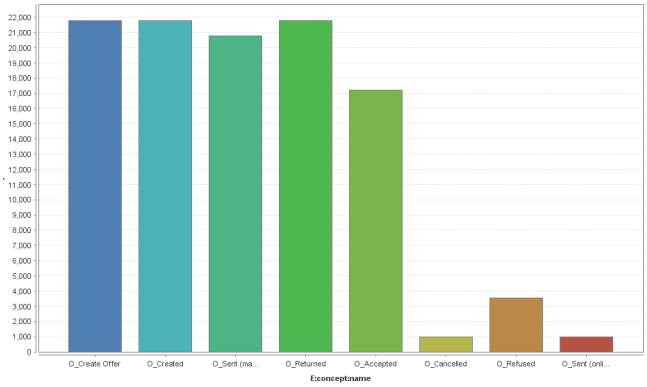

Fig. 3: State count for all offers marked as "Selected = true" in the event log.

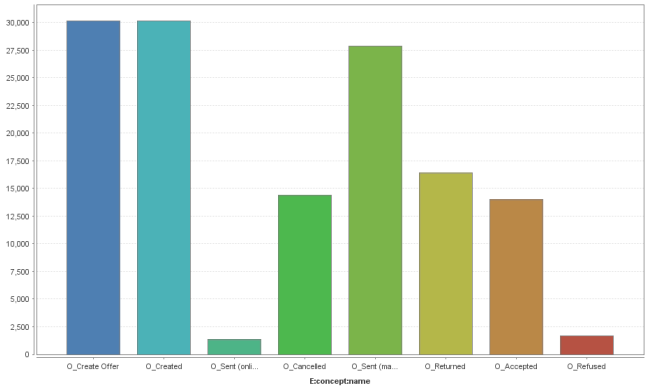

Fig. 4: State count for all offers marked as "Accepted = true" in the event log.

Similarly, we filtered the event log to keep only the offers that were accepted by the customer. In total, 30,136 offers were marked as *accepted* (i.e., the attribute "Accepted" has a value "true").

Figure 4 shows the count of all the reached states (i.e., activities) of this filtered event log. Note that this time not all offers are returned (a requirement to actually accept an offer), and many times they are refused or cancelled.

Again, we cannot rely on he "Accepted" attribute to consider if an offer has been accepted or not. Therefore, offers will be considered as *accepted* if and only if they reach the "O_Accepted" state.

3 Detecting Offer Variants

The Offers log was analyzed using a ProM plugin named "Process Variant Finder", built by the author if this paper (it is assumed that the reader is

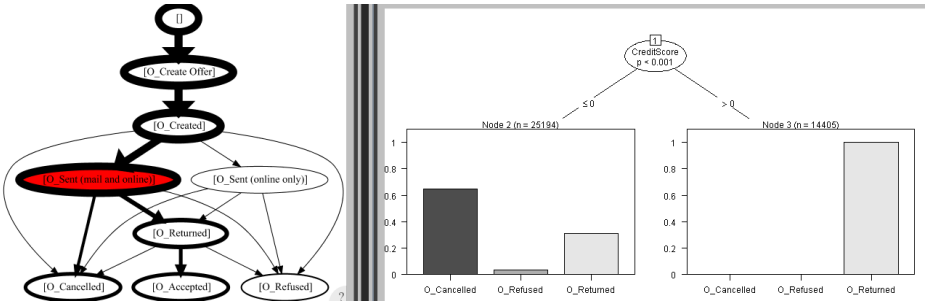

Fig. 5: Control-flow variants identified in the "O_Sent (mail and online)" state (left). The next state to be reached is highly determined by the credit score of the customer (right). Note that the Y axis represents the probability of each possible next state (X axis).

familiar with the ProM framework[2]). It creates an Event-annotated Transition System from an event log (i.e., where each state and transition is related to a set of events) and performs recursive partitioning by conditional inference using the data attributes available in the events. The formal definitions and mechanisms inherent in this tool are still unpublished and the work is still in progress.

The idea is to determine if the variability of any event attribute can be reduced by splitting any other event attribute without any extra knowledge of the company or their customers. If so, such variability reductions can considered as process variants. The remainder of this section describes three types of process variants found in the offers event log, depending on which perspective the variability is reduced: control-flow, performance and context. Note that such analysis is performed without any type of domain knowledge about financial institutes.

3.1 Detecting Variants with Control-flow Differences

Control-flow is usually associated to the activity to which an event refers to. However, this can be extended (as shown in Section 1) and control-flow dimensions such as the next activity to be executed can be added to events as data attributes. Concretely, the variability of the next activity (i.e., state) that will be reached is reduced by splitting other available data attributes. This analysis is illustrated by the findings described as follows.

Figure 5 shows that if an offer is sent to the customer, the probability of the offer being returned is related to the credit score of the customer. The results define the existence of two variants: the first being offers with a credit score = 0 and the second being offers with credit score > 0.

Note that offers that are sent via physical mail and online (92% of all offers) to customers with credit score higher than 0 are always returned. On the

[2] visit `http://www.promtools.org` for more information on ProM.

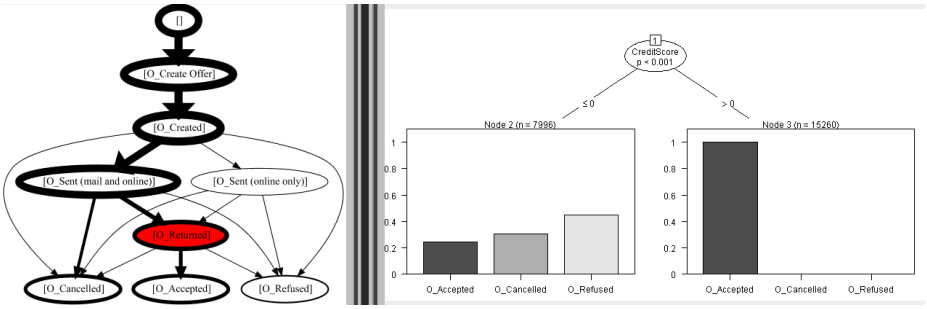

Fig. 6: Control-flow variants identified in the "O_Returned" state (left). The next state to be reached is highly determined by the credit score of the customer (right). Note that the Y axis represents the probability of each possible next state (X axis).

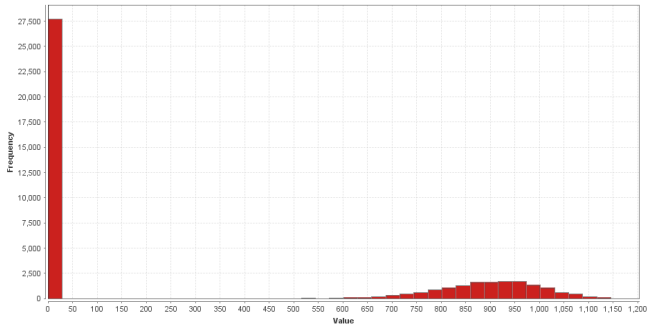

Fig. 7: Histogram of offers based on their credit score attribute.

other hand, offers sent to customers with credit score of 0 are more likely to be cancelled.

Similar variants were found for returned offers. Figure 6 shows that if an offer has been returned (i.e., is in the "O_Returned" state) the credit score of the customer also plays a determining factor on the next state to be reached. The results again define the existence of the same two variants. Returned offers (54% of all offers) are always accepted by customers with a credit score higher than 0. To the contrary, even if the offer is returned, most customers with a credit score of 0 will either refuse or cancel the offer.

So far, credit scores seem to be a good predictor for control flow, however, its distribution needs to be analyzed. Figure 7 shows a histogram of offers grouped by their credit score (in bins). It can first be noted that a large percentage of offers (approx 64%) are sent to customers with credit score = 0. It can also be observed that the non-zero credit scores resemble a normal distribution.

Assumption: A credit score of 0 does not mean that a customer is too risky. If that would be the case, the histogram would also contain values close to 0,

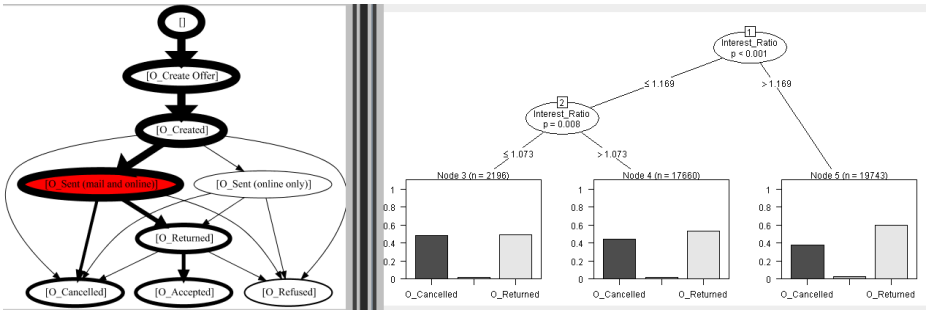

Fig. 8: Control-flow variants identified in the "O_Sent (mail and online)" state (left). The next state to be reached is highly determined by the total interest rate (right). Note that the Y axis represents the probability of each possible next state (X axis).

which is not the case. It is assumed then, that a credit score of 0 means that the customer has not had a credit risk evaluation yet. In other words, they are new potential customers.

Another attribute that was analyzed was the *interest rate* (described in Section 1). Figure 8 shows that if an offer has been sent to the customer via online and physical mail, the interest rate of the offer also plays a determining factor on the next state to be reached. Here, the results indicate the existence of three variants: corresponding to offers with an interest rate of ≤ 1.073, between 1.73 and 1.169 (including), and > 1.169. It can be observed in the variants described above that the chance of an offer being returned increases with the interest rate. Surprisingly, loans with lower interest rates are more likely to be cancelled, and loans with larger interest rates are more likely to be accepted (one would assume that lower interest rates are more appealing to customers).

Assumption: Loans with larger total interest rates have longer number of terms than loans with lower interest rates, because of composite interest. In other words, if two loans have the same monthly interest rate, but are paid in different number of months (i.e., terms) the total interest rate will be higher in the longer loans.

Given this assumption, one can deduce that customers are more likely to return longer-term offers, thus, with larger total interest rate). The effect of the "number of terms" attribute was also analyzed con confirm this assumption. Figure 9 shows the results such analysis. As suspected, sent offers are more likely to be returned for longer-term loans. Therefore, we can discard the previous analysis related to interest rate, since the analysis of the number of terms provides a simpler explanation for the same phenomenon (Occam's Razor).

Next, the impact of elapsed time in control flow was analyzed. Figure 10 shows that the longer that an offer takes to be returned (only for returned offers), the less likely that it will be accepted. This is through the analysis of the three detected variants for different elapsed time ranges. Note that the elapsed

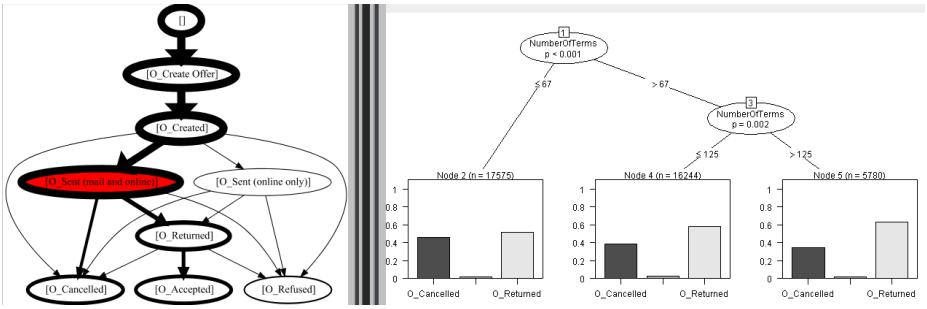

Fig. 9: Control-flow variants identified in the "O_Sent (mail and online)" state (left). The next state to be reached is related to the number of terms of the loan (right). Note that the Y axis represents the probability of each possible next state (X axis).

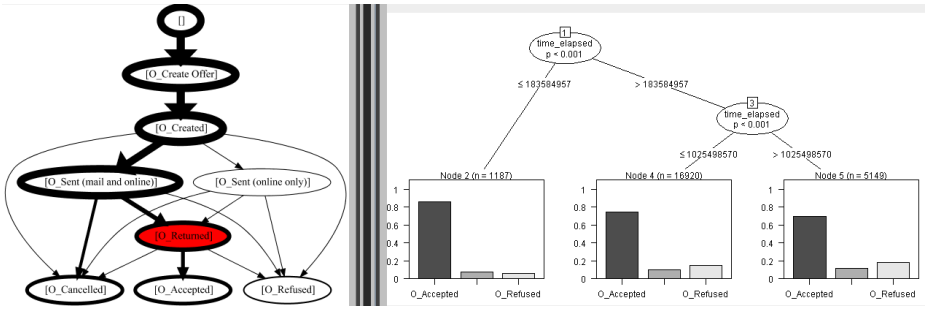

Fig. 10: Control-flow variants identified in the "O_Returned" state (left). The next state to be reached is related to the elapsed time of an offer until it is returned (right). Note that the Y axis represents the probability of each possible next state (X axis).

time is measured in milliseconds (18358957 ms = 5 hours, 1025498570 ms = 12 days approx). As observed in Section 2, most of the time spent by an offer until is returned is spent on the customer's side. Therefore, the company is recommended to work on contacting customers in order for them to return the offer as soon as possible.

We also analyzed the impact of other data attributes in control-flow, but the results indicated that there were no significant variants among them.

3.2 Detecting Variants with Performance Differences

Performance is usually associated to time, and it can be measured in several ways (e.g., elapsed time, remaining time). In this paper, we extended the data attributes related to events with two derived data attributes: elapsed time and duration (see Section 1 for details on how they are calculated).

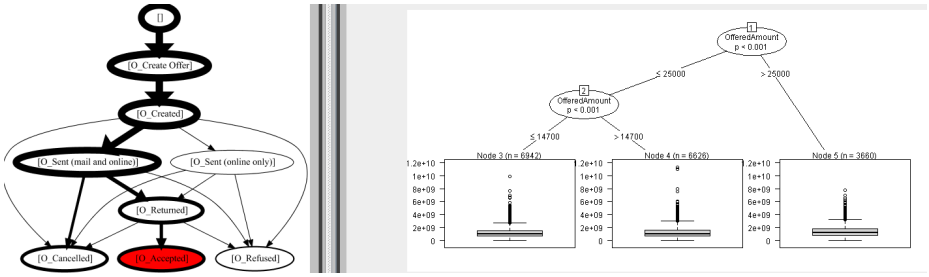

Fig. 11: Performance variants identified in the "O_Returned" state (left). The elapsed time of offers that are returned is correlated to the ammount offered (right). For each variant (defined by the offered amount) a box plot is shown.

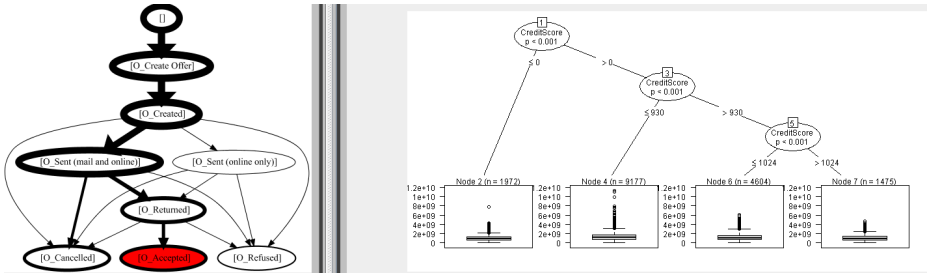

Fig. 12: Performance variants identified in the "O_Accepted" state (left). The throughput time of offers that are accepted is correlated to the credit score of the customer (right). For each variant (defined by the credit score range) a box plot is shown.

This section is dedicated to the analysis of wether by splitting data attributes, variants with different performance can be identified. Figure 11 shows that offers that are returned take longer to be returned for larger loans (i.e., higher offered ammount).

Returned offers of less than 14,700 take a shorter time to actually being returned. Offers with an amount higher than 25,000 take the longest. This could be expected as it makes sense that customers take more time to analyze and consider bigger loans. This may not be clearly visible in the chart, as the scale is expanded to include outliers.

Next, the effect of credit score in performance was analyzed.. Figure 12 shows that accepted offers with a non-zero credit score have a shorter throughput time when the credit score is higher.

Note that accepted offers for customers with a credit score of 0 are the fastest. Then, for non-zero scores, the lower scores tend to be slower.

Assumption: The higher the credit score is, the better is the customer. Based on the findings presented above and in the previous section, customers

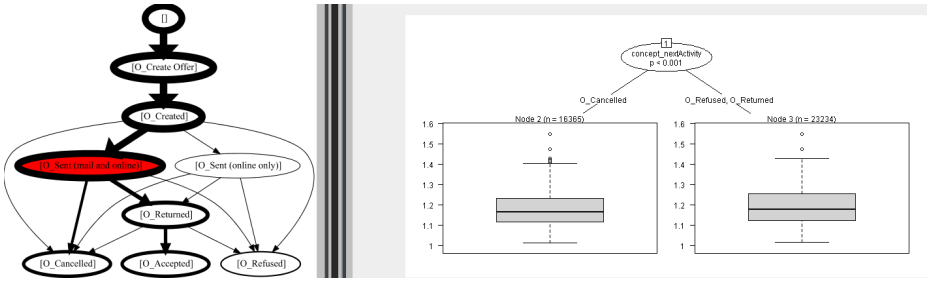

Fig. 13: Context variants identified in the "O_Sent (mail and online)" state (left). The interest rate of offers is related to the next state to be reached (right). For each variant (defined by the next state to be reached) a box plot of interest rates is shown.

with higher credit score are more likely to accept offers and take less time to do so.

3.3 Detecting Variants with Context Differences

In this paper, Context is defined as all the event data attributes that are not related to control-flow or performance. In this section, context attributes are analyzed in order to detect variants with a reduced variability.

Figure 13 shows that sent offers that are returned tend to have higher interest rates. This result is related to the one presented in Figure 8, where the same interaction was detected, but with inverted roles: control-flow variants were detected based on the interest rate. Now, context (i.e., interest rate) variants are detected based on the control-flow. Like before, the effect of the number of terms was also analyzed, but no significant differences were found this time.

Assumption: This effect is truly related to the total interest rate of the offer, since there is no context variants based on different control-flow for attributes such as number of terms, monthly cost or offered amount. It would be interesting to analyze the true interest per term. Some basic financial formulaes can be used for that purpose (e.g., *fixed compound interest*) but without domain knowledge is difficult to know wether such calculation reflects reality or not.

4 Conclusions

The offers event log has been analyzed; most frequent paths and biggest delays have been identified in Section 2. Also, several process variants have been detected in Section 3. Most notably, there seems to be a considerable effect of the credit score on the performance and control-flow of offers. However, the scope of the analysis is limited to offers as individual instances, which is not true in reality: several offers may be connected to the same log. Such information can

indeed be used to further detect variants. Also, the lifecycle of the application can be studied in detail.

This limited scope is decided on purpose, as this paper serves as a proof of concept for the evaluation of the process variant detection tool created by the author. The company is strongly suggested to extend the scope of this analysis if any changes in the process are to be made.