Business Process Intelligence

SS23 Assignment Part 1

Prof. Dr. Wil van der Aalst, Bianka Bakkulari M. Sc., Harry Beyel M. Sc.,
Benedikt Knopp M. Sc., Nina Graves M. Sc., Christopher Schwanen M. Sc.
Chair of Process and Data Science
RWTH Aachen University

April 27, 2023

Introduction

In this assignment, you deal with event data from a company located in the telecommunications sector. The company sells three types of products and has now decided to offer the option to buy a refurbished product instead of a new one. To enable this, customers return their old product which they no longer use. These products are then refurbished in the described refurbishment process. You are asked to take a look at this process. As a process miner, you try to get an understanding of the process before looking at the data. Therefore, you ask the process owner to describe the process to you. She gives you the following description:

”The process is initiated when a customer sends a used product. Once a returned product is registered, it is sent to the quality assessment department to detect the type of improvement the product needs. The improvements can be distinguished in ten categories. After categorization, the sales department is notified, and the product is refurbished by one of the two refurbishment teams. The refurbishment department has two teams: One of the teams merely does design polishings and the other team does more complex minor repairs. Some of the improvements can be done by both teams. Once a product is refurbished, it is returned to the quality assessment department where an employee checks whether the product is indeed in a state in which it can be resold. If it is not, it is returned to the Refurbishment department. This cycle is only performed a limited number of times as it is not considered profitable to spend too much time in the refurbishment of a product that is sold as new for a cheaper price anyway. Once the product is deemed ready to be resold or the maximum number of cycles is achieved, the case is documented, so that the sales department is informed on the current status.“

Submission Details

- Total number of points obtainable: 100
- Group size: 2-3
- Input: this assignment PDF, data (*activity_table.csv, case_table.csv, situation_table.csv, product_refurbishment.xes*)
- Deliverables: one PDF report

There is a bonus exercise with 4 additional points, so the points sum up to 104. However, awarded points are limited to 100.

Tools

In this assignment, we use *Celonis*, *RapidMiner* and *ProM.*

Deliverables

The deliverables are **a single PDF report of at most 20 pages** containing your answers to the assignment questions. Please include the screenshots and figures to support your answers in the PDF file.

Your written report will be the basis for grading. Make sure to include **the names of all group members on the title page**. In the report, you should present your methods, results, and explanations. Doing so, **clearly indicate which answer belongs to which question.** Moreover, the answers should be **self-contained** (i.e., they should not require references to external files, if not stated otherwise in the task description).

Besides, please mind the following criteria when preparing your report:

- Proper spelling, punctuation, readability
- Comprehensive structure - the report should not exceed the page limit
- Use of **adequate** visualizations for showing (aggregated) results or illustrating methods
- Axes have labels, diagrams have headers
- Figure quality (e.g., resolution and relevance)
- All used figures, tables and similar are numbered / labeled and referred to in your text.

Do not re-upload the event logs that we provided.

When answering the questions, document what you did and carefully describe and explain your results. In particular, explain how you derived your results and on which facts you base your claims.

Results from previous questions can be referred to, to improve your discussion and explanation.

Question 1: PQL

In this task, you use the tool offered by *Celonis*. Use the following files for this task: *case_table.csv* and *activity_table.csv*. Upload the files as shown in the instruction, create a data model, load the model, and create a new analysis.

(a) (4 points) As shown in the corresponding instruction, create a *case-based situation table* in Celonis based on these files. Create an OLAP-table and provide PQL formulas for the following dimensions:

- Case ID: Name/ID of the cases.
- Defect Type: Defect type of the case.
- Phone Type: Phone type of the case.
- Document Happened: Assign 1 if the activity “Document” happened in a trace
- Defect Fixed: Last update in the activities if the defect was fixed.
- Throughput Time: Throughput time of cases in minutes.
- Number of Refurbish (Simple): Number of activities named “Refurbish (Simple)” in a trace
- Number of Refurbish (Complex): Number of activities named “Refurbish (Complex)” in a trace

(b) (2 points) Using your data model, explore the process. To do so, use the *Process Explorer*. Set the setting of activities to 100%. Provide a screenshot of a model showing 80.6% connections. Moreover, provide a screenshot of a model with 84.1% connections. What changed?

(c) (2 points) Next, you want to investigate the different variants of the process in Celonis. To investigate variants, you use the *Variant Explorer* by considering frequency as measurement. Provide a screenshot of the processes depicting the most common variant. Additionally, provide a second screenshot of the process using the two most common variants. What are the differences between these two models? Are there changes in numbers?

Total for Question 1: 8

Question 2: Decision Tree

The process owner asks you to investigate why in some refurbishment processes, the defect got fixed but did not in others. To explain the reasons in more detail, you want to use decision trees. You decide to use *RapidMiner* for this task. The process owner prepared a dataset, *situation_tables.csv*, that you use for this task.

(a) (3 points) Before creating a decision tree, you want to explore your data set visually. Create a 3D-scatter visualization with the number of complex refurbishments on the x-axis, the number of simple refurbishments on the y-axis, and throughput time as value column. Color your instances based on whether a defect is fixed or not. What property do instances that could not be fixed have? Only focus on the attributes you use for the visualization.

(b) (8 points) In the next step, you want to discover the reasons in more detail. You consider the following attributes: *Defect Fixed, Defect Type, Phone Type, Document Happened, Throughput Time, Number of Refurbish (Simple), and Number of Refurbish (Complex).* You want to try out different parameter combinations to select a tree.

Use the *Decision Tree* operator to create a tree. Try out different parameters using the *Loop Parameters* operator. For evaluating your trees, use the *Cross Validation* operator and the *Performance* operator.

Your settings are as follows:

- Depth of trees between 1 and 7 in one step size,
- Criteria of trees are gain ratio, gini index, and information gain,
- Cross-validation is performed with five folds to have more reliable results,
- Sampling of cross-validation is set to automatic with the local seed 2023.

Show the processes in RapidMiner. Show a simple scatter chart with the depth of trees on the x-axis, the accuracy on the y-axis, and the criterion as color attribute. Which setting has the highest accuracy? Show the tree with the highest accuracy.

(c) (2 points) For the two tree leaves with the highest accuracy, state what characteristics the instances grouped in each leaf have.

Total for Question 2: 13

Question 3: Clustering

The process manager asks you to cluster instances based on their *number of simple refurbishments*, *number of complex refurbishments*, *throughput time.* Again, you decide to use *RapidMiner* and *situation_tables.csv.*

(a) (6 points) To cluster instances, you start with k-Means. Use the operator *k-Means* in RapidMiner. You set k equal to three. Show the process. Create a 3D-scatter visualization with *Number of Refurbish (Complex)* on the x-axis, *Number of Refurbish (Simple)* on the y-axis,*Throughput Time* as value column, and *cluster* as color. Provide a picture of the clusters. Explain the clusters' meaning. For the visualization, show the unnormalized values.

(b) (4 points) Besides k-Means, other clustering algorithms are available, for example, DBSCAN. Using the *Clustering (DBSCAN)* operator, set *epsilon* equal to one and *minimal points* to five to create clusters. Show your pipeline in RapidMiner. Create a 3D-scatter visualization with the number of complex refurbishments on the x-axis, the number of simple refurbishments on the y-axis, throughput time as value column, and cluster as color. Provide a picture of the cluster. How many clusters do you end up with? What do the clusters mean? For the visualization, show the unnormalized values.

Total for Question 3: 10

Question 4: Association Rules

(a) (6.5 points) The process manager asks you to find out which behavior occurs together. In particular, she asks you to find relationships between the number of simple and complex refurbishments, the phone and defect type, and if the defect is repaired at the end or not. For each attribute, she suggests the following:

- Number of simple or complex refurbishments: If there is no refurbishment, denote it with *no simple/complex refurbishment.* If there is one, denote it with *one simple/complex refurbishment.* If there are two, denote it with *two simple/complex refurbishments.* If there are more than two, denote it with *many simple/complex refurbishments.*
- If defectFixed=1, denote it with *Defect fixed.* If defectFixed=0, denote it with *Defect not fixed.*
- Leave defectType as numbers.
- Leave phoneType as it is.

You aim to find rules with minimum support of 0.7 and minimum confidence of 0.8. Show your process and sort your results from highest to lowest lift value. Provide a screenshot of the first rule and explain two rules. Again, you decide to use *RapidMiner* and *situation_tables.csv.*

(b) (2.5 points) Your manager only wants to know the rules when the defect is not fixed. Filter your data set for cases with unfixed defects and receive rules again. Show the process in RapidMiner. Provide an overview of your process and a screenshot of your first rules sorted by lift (highest to lowest). Additionally, filter for conclusions which *Defect not fixed* is part of. What is your observation concerning the lift values? What is the resulting implication for the rules? Explain the interplay of support, confidence, and lift for these rules.
Hint: Consider how support, confidence, and lift are computed. Use the filtered data set to check your assumptions.

Total for Question 4: 9

Question 5: Process Exploration

In the following, we use ProM to explore and analyze the event data stored in the file *product_refurbishment.xes.* Let that log be denoted by L.

(a) (5 points) Answer the following questions regarding L. You may use any of the ProM Visualization tools to derive your answers, for instance the log visualizers or directly-follows graphs. For each question, describe how you derived your answer.

- (i) What activities are there in the log?
- (ii) How many trace variants are there?
- (iii) Which attributes are featured at which events?
- (iv) How many different end activities are there? What is the most frequent end activity?
- (v) What activity is most frequently performed after *Detect Required Improvement*?

(b) (3 points) A trace variant may be considered as noise if it is infrequent and it is unlikely to capture behavior observable in reality. For a simpler and more concise analysis, we could rid L of potential noise.

- (i) Name a trace variant occurring in L that may be considered as noise and explain your choice. Relate your explanation to the process description in the introduction.
- (ii) Filter L by both removing all trace variants occurring less than ten times, and keeping only those variants with the most frequent end activity. From the filtered log, show the five most frequent and the five least frequent trace variants together with their absolute and relative frequencies.

(c) (9 points) In the following, continue working with the original log L.

Answer the following questions regarding L using the *Dotted Chart* of the ProM Visualization tools. For each question, explain how you plotted the data (i.e., attributes/labels of axes and sorting criterion) and how you visually derived your answer.

Note: You do not need to provide precise evaluations of the data. Instead, choose meaningful visualizations. You may also use screenshots to support your answers.

- (i) What are the company's business hours?
- (ii) What fraction of cases are handled within the first 12 hours after the case starts?
- (iii) What activities is each employee responsible for?
- (iv) What activity having a *duration in minutes* takes on average the most time to be completed?
- (v) Look at the long term behavior of the process (from June to September). Apart from non-working hours, are there time windows in which certain activities were not performed at all?
- (vi) What employee responsible for *Refurbish (Simple)* takes the most time, which one takes the least time to complete their task?

(d) (3 points) We would like to analyze how long the staff effectively needs to work on cases, i.e., the duration of traces. However, since new phones are delivered and registered batchwise at the start of each working day, the total trace durations are misleading. Hence, we first filter out the registration activity from the log, and analyze the durations of the truncated traces.

- (i) Filter out the activity *Register Arrival.* Then, use again the dotted chart to plot sorted trace durations of the filtered log. Show your plotted chart.
- (ii) What are the predominant time intervals for trace durations? Relate your explanation to your observations from *c(i).*

Total for Question 5: 20

Question 6: Alpha Miner

In the following, L again refers to the log provided in *product_refurbishment.xes.*

(a) (2 points) Mine a Petri net on L using the Alpha miner plugin in ProM. Discuss:

(i) Is $\alpha(L)$ a workflow net?

(ii) Is the state space of $\alpha(L)$ finite or infinite? If $\alpha(L)$ is a workflow net, assume that the initial marking has exactly one token residing in the source place.

(b) (4 points) We would like to investigate all properly documented cases where products were refurbished in a single pass, i.e., after one refurbishment action. Therefore, let L' be the log containing those cases in L which end with activity *Document* and do not include *Restart Refurbish.*

(i) Using ProM, compute L' as well as $\alpha(L')$. You should obtain a sound workflow net accepting finitely many traces. Show the trace variants of L' as well as $\alpha(L')$.

(ii) How good are fitness and precision of the model with respect to the log? Explain your answer.

(c) (8 points) Consider the following log L_3, featuring the three most frequent trace variants in L:

$$
\begin{aligned}
L_3 = [&\langle \mathit{Register\ Arrival}, \mathit{Detect\ Required\ Improvement}, \mathit{Notify\ Sales}, \mathit{Refurbish}\ (\mathit{Complex}),\\
&\mathit{Assess\ Product\ Quality}, \mathit{Document}\rangle,\\
&\langle \mathit{Register\ Arrival}, \mathit{Detect\ Required\ Improvement}, \mathit{Notify\ Sales}, \mathit{Refurbish}\ (\mathit{Simple}),\\
&\mathit{Assess\ Product\ Quality}, \mathit{Document}\rangle,\\
&\langle \mathit{Register\ Arrival}, \mathit{Detect\ Required\ Improvement}, \mathit{Refurbish}\ (\mathit{Complex}),\\
&\mathit{Assess\ Product\ Quality}, \mathit{Notify\ Sales}, \mathit{Document}\rangle]
\end{aligned}
$$

Apply the Alpha algorithm on L_3. You may use shorthand notations for activities / transition labels, e.g. R for *Register Arrival*, C for *Refurbish (Complex)* etc. Proceed as follows:

(i) Give the footprint matrix of L_3.

(ii) Derive the set X of pairs of transition sets as defined for the Alpha algorithm.

(iii) Select the set Y of maximal pairs of transition sets as defined for the Alpha algorithm.

(iv) Show the resulting Petri net $\alpha(L_3)$.

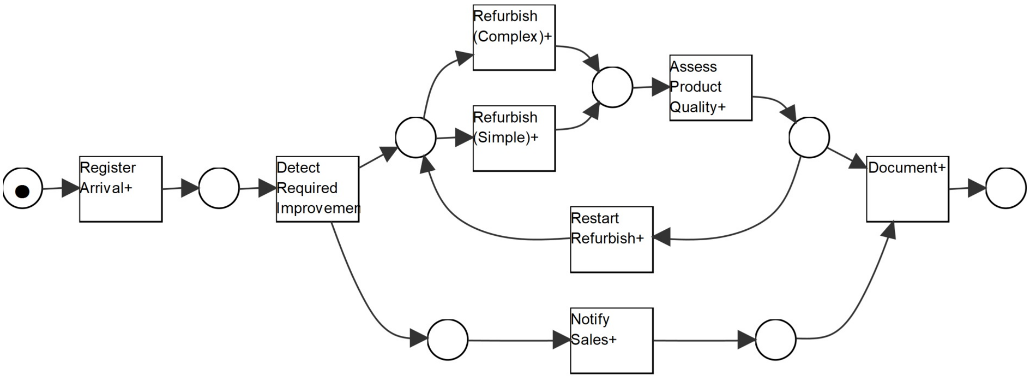
Hint: (iv) is graded based on the consistency between $\alpha(L_3)$ and Y. You may, however, validate your result with ProM and in the following work with the ProM output.

(d) (6 points) Follow up on the previous task with a discussion of $\alpha(L_3)$.

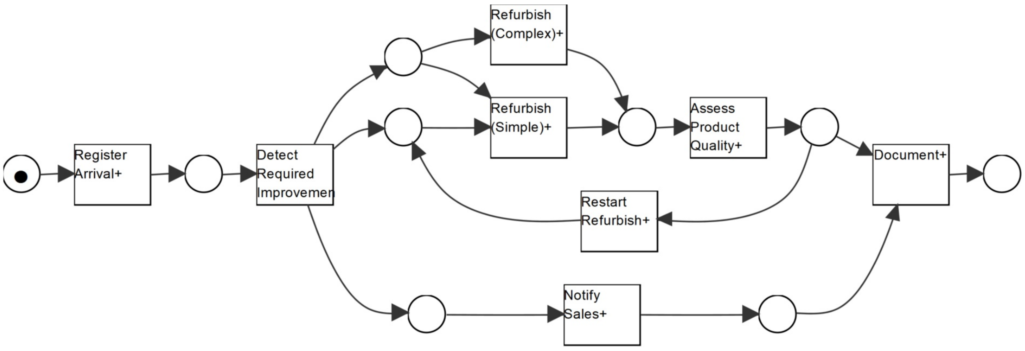
(i) For each of the soundness subproperties *safeness, option to complete, proper completion* and *freedom of dead transitions*, discuss whether $\alpha(L_3)$ satisfies that property. Is the net sound?

(ii) Is there a trace variant accepted by $\alpha(L_3)$? Which of the trace variants in L_3 are accepted by $\alpha(L_3)$?

(e) (4 points) (Bonus Exercise) Let L_1, L_2 be two event logs such that $\alpha(L_1)$, $\alpha(L_2)$ are as depicted below.



(a) $\alpha(L_1)$



(b) $\alpha(L_2)$

Figure 3: Two Petri nets discovered by the Alpha miner. L_1, L_2 were obtained from L by filtering out traces with less than 10 and 20 occurrences, respectively, and by retaining only traces ending with *Document*.

(i) Name two activities a, b where $a \rightarrow b$ holds in L_1, but $a \rightarrow b$ does not hold in L_2.

(ii) Let L_2' be a log that has the same footprint matrix as L_2, except that $a \rightarrow b$. Is $\alpha(L_2') = \alpha(L_1)$? Explain your answer without explicitly writing down the footprint matrices.

Total for Question 6: 24

Question 7: Heuristic Miner

Use the "Interactive Data-aware Heuristic Miner" ProM Plugin presented to you in the lecture and load the event log *product_refurbishment.xes*. For this task, please do not set any data-aware configurations, set the conditions slider to zero and **do not** tick any additional settings.

You have learned in the lecture that a causal net is built from a dependency graph and that your dependency graph can take different shapes depending on some parameters you chose. Change the output process model to "Dependency Graph". Please note that the dependency graph is the basis for a C-net, requiring a graph from the start to the end node. If the passed parameters violate this requirement, the plugin ensures that this requirement is fulfilled based on internal logic.

While working on your process analysis, the process owner walks by. Intrigued by what she sees on your screen, she wants to know more about the dependency graph.

(a) (4 points) Explain to her what you control when using the frequency slider. Refer to the change in the dependency graph when switching the frequency from 0.2 to 0.5 to demonstrate its effect.
Hint: By double-clicking the slider value you can enter the exact frequency.

(b) (4 points) Having understood the frequency slider, she asks about the dependency slider. Explain the what the dependency slider does as well as the meaning of the weights displayed on the arcs. What does the weight of 0.998 on the arc from *Detect Required Improvement* to *Notify Sales* tell us compared to the 0.145 from *Assess Product Quality* to *Notify Sales*?

You tell the process owner that you have to get on working now and switch to the causal net output. With the bindings slider, you can determine what bindings are included relative to the most frequent one (per activity and binding type). For the following, set the frequency slider to 0.4.

(c) (2 points) How often is a complex refurbishment and how often is a simple one needed? Use the overview of all output bindings associated with *Detect Required Improvement* to provide the absolute and relative frequencies for all bindings involving these activities.
Hint: Right-clicking activities provides additional information on bindings.

(d) (1.5 points) Set the dependency slider to 0.7, and the bindings slider to 0. Consider the resulting C-net. Is the following trace σ_1 valid on this C-net?

$$\sigma_1 = \langle \textit{Start},\ \textit{Register Arrival},\ \textit{Detect Required Improvement},\ \textit{Refurbish}\,(\textit{Complex}),\\ \textit{Refurbish}\,(\textit{Simple}), \textit{Assess Product Quality},\ \textit{Notify Sales},\ \textit{Document},\ \textit{End}\rangle$$

If yes, provide the corresponding binding sequence, if not, provide the obligation that cannot be fulfilled.

(e) (1.5 points) You want to create a causal net that describes the way the process works. Considering the process description and the output bindings of *Detect Required Improvements* – what value do you suggest setting the bindings slider to? Explain your choice.

(f) (1.5 points) When the sales team is notified about a product, is it more likely that a simple or a complex refurbishment is required? How can you see this in the causal net displayed in ProM?

(g) (2 points) Explain how infrequent behavior can be excluded when using the Alpha miner and when using the Heuristic miner.

(h) (3.5 points) As we know from the process owner's description, it is possible that refurbishment has to be restarted which leads to a loop in the process model. In the ProM Plugin, set the frequency slider to 0.2, the dependency slider to 0.7 and the bindings slider to 0.3 and switch to the output "Petri net".
Compare the resulting Petri net with the two Petri nets $\alpha(L_1), \alpha(L_2)$ from Fig. 3a, 3b.

(i) In each of the nets, what type of refurbishment can be started after the activity "Restart Refurbish" was completed?

(ii) What is the main difference between the structure of the Petri nets mined by the Alpha miner and the one displaying the result of the Heuristic miner that makes this difference in net behavior possible?

Total for Question 7: 20