Advanced Process Mining - 12.25111

Practical Assignment; Part I

Deadline: 03.06.2022, 23:59 CET

V. Peeva M.Sc., T. Huang M.Sc.,
T. Brockhoff M.Sc., Dr. ir. S.J. van Zelst

Chair of Process and Data Science
RWTH Aachen University

Introduction

In this assignment, we investigate a *large-scale vaccination process*. The responsible healthcare institution has extracted several event logs provided in the assignment `.zip` file for further analysis. They already have some experience with analyzing vaccination processes with the help of Stella. However, after last year's analysis, Stella retired and left Petre in charge. Hence, they provide you with the initial information below and occasional support from Petre. It is your task to make sense of the data and to help the institution understand its vaccination process. In doing so, you will mainly use the Process Mining for Python (PM4Py) library as well as the Process Mining Workbench (ProM). However, for some of the tasks, you can use a tool of your choice as long you can produce the requested results. You will carry out the Python part of your analysis in a Python notebook using project Jupyter's JupyterLab/Notebook. A template has already been set up for you. In addition, you have to deliver a written report explaining your methods and results.

The Data

Each event, recorded in the context of the vaccination process, has the following attributes:

Attribute Name	Attribute Description
Patient	Patient id of the patient that is related to this event (case id)
time:timestamp	Timestamp of the event
concept:name	Activity corresponding to the event
lifecycle:transition	Lifecycle transition name. (Here only "complete")
Age	Age of the patient
Insurance	Type of the patient's insurance (statutory or private)
Health Worker	Boolean flag that describes whether the patient is a health worker

Assignment Details

- Total number of points obtainable: 100 (20% of final course grade)
- Group size: 2-3 group members
- Input: Jupyter notebook template, event log, LaTeX report template
- Deliverables: PDF report, Jupyter(Lab) notebook, figures folder

Tools

In this assignment we will use ProM 6.11 and PM4Py version 2.2.20.1.

Deliverables

The deliverables comprise a report, a notebook, and a folder with high resolution figures.

Report Your written report serves as the main basis for grading. In the report, you present your methods, motivation, results, and explanations. Doing so, **clearly indicate which answer belongs to which question.** Moreover, it should be **self-contained** (i.e., it should not require references to the notebook or to the figures folder). The length should be at **most 20 pages**, including the title page, excluding appendices. Make sure to include **all group members names with the student IDs on the title page**. If you do not want to use the provided LaTeX template, create a document mimicking its structure by a method of your choice. Besides, 10 points are reserved for style. The following criteria (among others) are considered when judging the style of the report:

- Proper spelling, punctuation, readability
- Comprehensive structure
- Use of **adequate** visualizations for showing (aggregated) results or illustrating methods
- Figures have captions, axes have labels, diagrams have headers
- Figure quality (e.g., resolution and relevance)
- All figures, tables and similar are numbered and properly referred to in text

Notebook Your Jupyter(Lab) notebook is used for reference and potentially testing your code. Therefore, it should satisfy the following requirements:

- Commented and structured code (if not, this will be penalized in the style points)
- Questions separated by markdown headers
- Top-to-bottom runnable cells to reproduce your results
- Additional packages free and installable from standard online repositories using "pip install" or "conda install" (NO further dependencies)

Do not re-upload the event log. Notebooks that intentionally access files outside of the notebook's directory or do any harm will be graded with zero points.

Figures All larger output figures are also saved and submitted as .pdf or .png in the provided figures folder.

Hints

When answering the questions, document what you did and carefully describe and explain your results. In particular, explain how you derived your results, on which facts you base your claims, and motivate the methods you used. Results from previous questions can (and should) be referred to, to improve your discussion and explanation.
The template notebook already contains many imports, which you will need to use, and a few examples of helpful code snippets that have not been explicitly part of the official documentation when publishing the assignment.

Optional Resources

- Jupyter: `https://jupyter.org/index.html`
- PM4Py installation guide: `https://pm4py.fit.fraunhofer.de/install`
- PM4Py documentation: `https://pm4py.fit.fraunhofer.de/docs` — **Consider to use the simplified interface** (see template notebook for an explanation)
- PM4Py source code: `https://github.com/pm4py/pm4py-source/tree/release/pm4py`
- ProM: `http://www.promtools.org/doku.php`
- PMTK: `https://pmtk.fit.fraunhofer.de/`
- Disco: `https://fluxicon.com/disco/`
- See [2] for a short introduction into the `Muli-Perspective Process Explorer`. Note that we do not use the possiblity to add additional perspectives; however, this tool gives a nice visualization of alignments
- See [1] for more details on the `Interactive Performance Spectrum` plugin

Question 1: Process Overview

Before Stella retired, she gave a few tips to Petre. One tip was to get an overview of the data, before diving deeper into the analysis of the vaccination process. Therefore, Petre provides you with the event log `log_vaccination.xes` and shares some of Stella's wisdom with you.

Note: You are free to use and combine tools of your choice as long you deliver the result. Some possibilities are writing code in Python + PM4Py, using ProM, Disco, PMTK, etc.

(a) (4 points) First of all, you want to get familiar with the process and convince Petre that you are the right person for this job. Therefore, you inspect a few cases. Give a high-level description of the process. Additionally, you calculate several statistics on the provided data to get a better overview. The APM Team suggests you to investigate the number of cases, the number of different activities, the minimum, maximum, and average event count for each trace, and the distribution of the trace duration in the form of a histogram.

(b) (3 points) A good way to visualize the overall process without model bias is by means of a *Dotted Chart*. Load the event log `log_vaccination.xes` into ProM and create a *Dotted Chart* with the `Project Log on Dotted Chart` plugin, using the following configuration: *X Axis Attribute* $\mapsto$ *E: time:timestamp*, *Y Axis Attribute* $\mapsto$ *T: concept:name*, *Trace Sorting* $\mapsto$ *time:timestamp of first event*, *Color Attribute* $\mapsto$ *E: concept:name*. In this configuration, every case constitutes a row of dots in the chart where a dot is positioned according to the timestamp of the corresponding event. Moreover, each activity is assigned a color. Present, describe, and explain the chart. In particular, describe any interesting patterns that you discover and explain their meaning.

(c) (4 points) One of the first things Petre learned from Stella is that a powerful way to reveal additional information using a *Dotted Chart* is to enhance the event log with extra attributes. We expect *Age* to be an important factor in the vaccination process. Hence, by using the same configuration for the *Dotted Chart* as in Question 1.b and just changing the *Color Attribute* to *E: Age Category*, we can get significant insights into our process. Note, by using *E: Age Category*, the *Dotted Chart* automatically creates an extra attribute based on binning the *Age* attribute. Discuss what we can learn from the visualization. Are there any noticeable patterns? Are there any exceptions that deviate from the usual patterns? If yes, can you explain this behavior by changing the coloring attribute?

(d) (4 points) The *Dotted Chart* can give us valuable initial insights in the performance of the process. Therefore, Petre wants you to create a *Dotted Chart* that is suitable for investigating the duration of the cases and on which long-lasting cases are clearly visible. Explain the configuration of your *Dotted Chart* and describe it. In particular, describe any patterns or special properties you can identify with respect to performance. Can you already detect potential reasons for difference in case duration?
Hint: To be able to explore long cases you need to enhance your event log. The `Add Elapsed Time in Trace as Attribute to all Events` plugin might help with that.

Total for Question 1: 15

Question 2: Process Discovery

Next, you and Petre try to visualize the control flow of the process using different algorithms. Additionally, you analyze the strengths and weaknesses of each of the algorithms.

(a) (2 points) Usually, we analyze running processes. Hence, although we use historic data, the process has not finished and some cases might not be fully contained in the extracted event log. In process discovery, such running cases will usually affect the discovered model. Therefore, you and Petre decide to filter the event log. Filter the event log such that you retain only the traces, for which the last executed activity is among the following activities: *Checkout No Vacc*, *Decline Vaccination*, *Destroy Vacc*, *Notify about vacc. possibility*, or *Send Vaccination Certificate*. Report, how many cases were removed, how many are still present in the log, and the set of end activities for the removed traces.

(b) (12 points) Petre finally asks you to discover a model from the event log. However, still not too trusting of your abilities, he asked the APM Team to provide the event log generated by the filtering in the previous task: `log_vaccination_finished_cases.xes`. In APM lectures, you learned about different discovery algorithms and you are struggling to decide which is the best for the vaccination process. Therefore, you decide to try the *Inductive Miner*, *Heuristic Miner*, *Alpha Miner*, *eST Miner*, *ILP Miner*, and *State-based Region Miner*. Report for each discovered model whether it looks flowery (allows all kind of behavior) or spaghetti-like. Discuss the four quality dimensions for each of the discovered models. Report a timeout if an algorithm needs more than 15 minutes for a result to be returned and discuss what might be the cause. Use the following ProM plugins with default parameters:

- *Inductive Miner* - `Mine Petri net with Inductive Miner`
- *Heuristic Miner* - `Mine for a Heuristic Net using Heuristic Miner`, and then apply `Convert Heuristic net into Petri net`
- *Alpha Miner* - `Alpha Miner`
- *eST Miner* - `eST Miner - Standard`
- *ILP Miner* - `ILP-Based Process Discovery`
- *State Region Miner* - `Mine Transition System` and `Convert to Petri Net using Regions`

and calculate fitness (token-based and alignments), precision (token-based and alignments), generalization and simplicity using PM4Py.

(c) (4 points) Petre also wants to apply process discovery on a more refined event log and focus on the following aspects:

1. Focus on active patients: Remove traces in which *Notify about vacc. possibility* is directly followed by itself.
2. Ignore patients with illegal start: Remove traces that do not start with *Enter into System* activity.
3. Focus on the patient perspective: Remove the following activities: *Insurance Check Private*, *Insurance Check Statutory*, *Prepare vaccine*, *Send vaccine to cabine*, *Send Vaccination Certificate*, *Appointment granted*, *Priority Appointment granted*, *Destroy Vacc* and *Send Invoice*.
4. Unify declining vaccination: Rename the activity *Checkout No Vacc* to *Decline vaccination*

Report the different activities and number of cases and events in the filtered event log.

(d) (8 points) To make sure you proceed with correctly filtered event log, Petre decided to do the filtering in parallel with you. You are once again going to apply process discovery, but on the postprocessed event log Petre provided: `log_vaccination_2c.xes`. Use the same algorithms as in Question 2.b, except for the *eST Miner*. Additionally, Petre provides a hand-made model (`small_hand.pnml`). Compare this model to the ones that you discovered with respect to the quality dimensions. Additionally, for three of the used discovery algorithms pinpoint behavior modeled by the hand-made model that they cannot discover. Use the same plugins in ProM and code in PM4Py as in Question 2.b.

(e) (6 points) Petre is interested in the impact of different threshold parameters for the *Inductive Miner*. Use the event log `log_vaccination_finished_cases.xes` and the implementation for *Inductive Miner* in PM4Py. Try at least four different thresholds, including 0. Calculate values for each of the quality dimensions. How does the model change as the noise threshold increases?

Total for Question 2: 32

Question 3: Conformance Checking

One would think that after working on all those tasks with Petre, you would not hide information from one another. Petre disagrees. All this time, he had a hand-made model for the process, which he wants you to use for conformance checking. He made you discover several models so that you understand the vaccination process better and also to compare the discovered models with the one the organization has. At least Petre was nice enough to provide you a Petri net for the hand-drawn model given in `full_hand.pnml`.

(a) (4 points) Petre looks at the different models, and he notices differences. However, he would be very happy if you can provide some statistics on the model difference. You decide to do model comparison via footprint matrices in PM4Py. He is mostly interested in comparison with the models discovered by `Inductive Miner` and `Heuristic Miner` for the `log_vaccination_finished_cases.xes` event log. Provide the results and describe your findings.
Hint: Looking at the number of sequence and parallel constructs might be helpful to explain the simplicity of some models when compared to others.

(b) (3 points) Petre thinks it is time for him to get some numbers telling him how much the process recorded in `log_vaccination_finished_cases.xes` conforms to the model he provided you with. Apply conformance checking in PM4Py using a footprint matrix, token-based replay, and alignments. Report and discuss the results.

(c) (6 points) Petre is not really happy that you gave him three different numbers. He doesn't understand why you couldn't give him a single number. Simply said, he does not know what to do with them. It is your job to explain to Petre that each number provides additional information for him and that is better to have them all instead of one, at least if he doesn't do more in-depth analysis. To help you, the APM team provides you with the model `flower.pnml` and the `one_trace_log.xes` event log containing only a single trace. Compute the conformance for the `flower.pnml` model and the `log_vaccination_finished_cases.xes` event log using footprint and alignments. Describe the results you get. Is there a difference, and if yes what causes it? To explain at least one difference between token-based replay and alignments to Petre, compute conformance between the `full_hand.pnml` model and the `one_trace_log.xes` event log using the two algorithms. Explain why the two algorithms yield different values.

(d) Petre is now interested in the in-depth conformance analysis you mentioned earlier. To this end, you are going to identify in which parts of the model de-

viations occur and what kind of deviations these are. You again analyze the `log_vaccination_finished_cases.xes` event log on the `full_hand.pnml` model.

i. (2 points) Use the `Multi-perspective Process Explorer` plugin in ProM to compute alignments and project them on the model. Describe the results. Can you already point out which parts of the model are most problematic? *Note: The `Multi-perspective Process Explorer` plugin sometimes has trouble working with non-standard attributes. Therefore, the APM Team suggests to use the `Remove all 'non-standard' attributes (In Place)` plugin to remove such attributes.*

ii. (8 points) Inspect the alignments manually (you can access the alignments by clicking the *Toggle Trace View* button in the `Multi-perspective Process Explorer` plugin). Can you spot some general problems? Describe what you can find and note whether in your opinion the problem is noise, a real problem that the organization needs to solve, or simply a desired behavior and the model should be updated to include it.

Total for Question 3: 23

Question 4: Performance Analysis

According to Petre it is time for performance analysis. He wants to use the *Interactive Performance Spectrum* plugin in ProM, and you are supposed to help him. First of all, Petre provides you with a new event log in which for the *Send Invoice* activity also a resource is recorded. You complain to Petre and say you have privacy concerns. He is reluctant to listen to you and tells you that he is aware that this is your APM assignment and if you want to pass the course you have to do it. You being you, with your strong convictions about protecting the privacy of the employees, talk to the APM Team. We agree with you, so we provide you a post-processed event log (`log_resources.xes`) in which the resources are renamed, and 5% of them are changed with a randomly chosen existing resource.
Note: Similar functionality as in the `Interactive Performance Spectrum` plugin is also available in PM4Py and PMTK.

(a) (7 points) To use the *Interactive Performance Spectrum* plugin, you need to discover a *Petri net*. Use the *Mine with Inductive visual Miner* plugin where *paths* is set to 0.8 and export a *Petri net* (not an *Accepting Petri net*). Next, run the *Interactive Performance Spectrum* plugin and analyze the event log. Petre is especially interested in the *Checkout* $\rightarrow$ *Send Invoice* and *Checkout* $\rightarrow$ *Priority Appointment* subpaths. Describe your findings.

(b) (3 points) Petre's own conformance analysis discovered that sometimes the *Decline vaccination* activity happens in the middle of a trace. Find all activities that directly follow *Decline vaccination*. Then, with the help of the *Interactive Performance Spectrum* plugin, investigate the spectrum between *Decline vaccination* and each of the activities found.

(c) (5 points) Petre received a complaint from a few patients that they got an invoice although they have a statutory health insurance. This does not comply with the rules the organization has set, so Petre asks you to investigate the problem. You decide to do a root-cause analysis in PM4Py. To this end, enhance the event log with an additional attribute called *illegal*. Annotate traces that contain events corresponding to both *Insurance Check Statutory* and *Send Invoice* activity as *illegal*. Train a *Decision tree* with *illegal* as the target attribute. Discuss the results (keep in mind that there was some resource swapping).
Hint: You may want to have a look at the Decision Trees - Root Cause Analysis section in the PM4Py documentation.

Total for Question 4: 15

Question 5: Conclusion

To conclude your report for Petre, briefly summarize your findings. In particular, focus on the major problems that you discovered and give recommendations on how to address them. As a small outlook on the second part of the assignment, keep in mind that even this small artificial process slightly evolves over time. Therefore, ensure that you frame your recommendations accordingly. For example, a recommendation that concerns problems existing only in the beginning of the process could be framed as: There *has been* a problem with respect to ...; therefore, we would recommend to do ... in order to alleviate this problem if this situation occurs *again* or ... to avoid this situation completely.

Total for Question 5: 5

References

[1] Wil M.P. van der Aalst et al. "Visualizing Token Flows Using Interactive Performance Spectra". In: *Application and Theory of Petri Nets and Concurrency - 41st International Conference, PETRI NETS 2020, Proceedings*. Available at `http://www.padsweb.rwth-aachen.de/wvdaalst/publications/p1130.pdf`. 2020, pp. 369–380.

[2] Felix Mannhardt, Massimiliano De Leoni, and Hajo A. Reijers. "The Multi-perspective Process Explorer". In: vol. 1418. Available at `https://www.researchgate.net/profile/Felix_Mannhardt/publication/282283719_The_Multi-perspective_Process_Explorer/links/560a5fe808ae576ce63fc926/The-Multi-perspective-Process-Explorer.pdf`. 2015, pp. 130–134.