Advanced Process Mining SS22

Report Assignment Part 1

Submitted To:
Viki Peeva M.Sc.
Tsunghao Huang M.Sc.
Tobias Brockhoff M.Sc.
Sebastiaan J. van Zelst Dr. ir.

Submitted By:
Mengru Ji (427504)
Guoqing Jin (427508)
Shengnan Lu (405608)

Q1 Process Overview

⟨This section will briefly view our vaccination process event log from several aspects, including the statistics and several dotted charts corresponding to different scales.⟩

a) ⟨Several statistics relating to this event log have been investigated at first. This event log owns **4960 cases** and **20 different activities** in total. The **minimum, maximum, and average event count for each trace** are **3, 35, and 18.74**, respectively. The following histogram, Figure 1, shows the distribution of trace duration. The **maximum duration is 7806.85 hours**, **the minimum duration is 0.42 hours**, and **the traces have a duration of 1551.03 hours on average**. ⟩

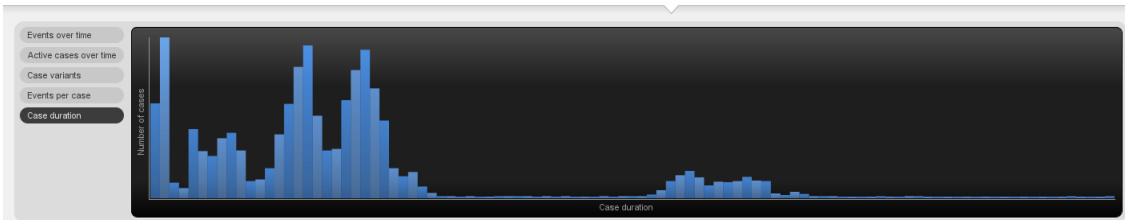

Figure 1: Distribution of Trace Duration

b) ⟨Utilizing dotted chart analysis, we can have an in sight in the over all event through the distribution of events over all time (See Figure 2). Each row of Figure 2 depicts one of the traces displayed on the time scale and all the events have been colored according to the concept:name. We observed that the entire process took nearly 8 months for most instances. Besides, almost all traces have the same starting events (See Figure 3), which are "Enter to System" (yellow mark), "Insurance Check Private" (orange mark) or "Insurance Check Private Statutory" (dark blue mark), and "Notify about vacc. possibilities" (blue mark). After around 3-5 days, people usually receive confirmation of their vaccine possibility, sometimes even longer. Some people also choose to refuse the vaccination before receiving the acknowledgement. In addition, we found that nearly every trace of no declining vaccination events has at least one vaccination process (start with "Prepare Vaccine", end with "Checkout") (See Figure 4), most patients had two vaccinations, some also had three. And patients usually waited a long time for their third vaccinations after they have the second vaccination. Furthermore, all patients would receive a vaccination certificate after their second vaccination. If someone had a third vaccination, they would receive a new vaccination certificate.⟩

Figure 3: Starting Events

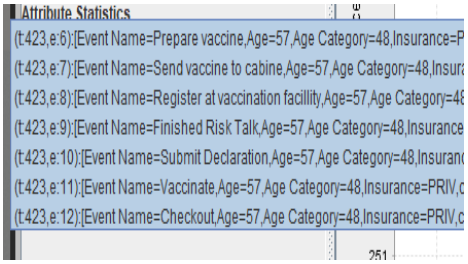

Figure 4: Vaccination Events

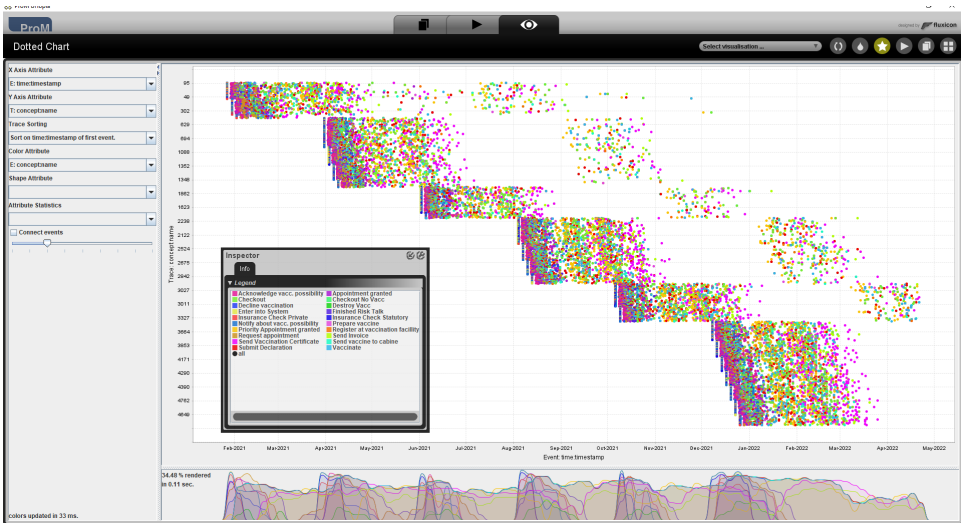

Figure 2: Dotted Chart Analysis over All Events Categorized by "E:concept:name"

c) ⟨(From the very top to the very bottom in Figure 5) In the first part: (1)Starting on January 27, 2021, people's age span is vast, ranging from 20 to 60. (2)Beginning on January 30, 2021, persons over 40 were vaccinated. Besides, a small number of young people were also vaccinated. (3)Beginning February 2, 2021, only the elderly over 88 were eligible for immunization. The bulk of those vaccinated in the second (from 31.03.2021 to 10.04.2021) and third parts (from 02.06.2021 to 05.06.2021) are above 60 years old. All of the recorded traces in the fourth part (from 04.08.2021 to 13.08.2021) and the fifth part (from 06.10.2021 to 09.10.2021) are related to adults aged 40 to 60. Young adults between the ages of 20 and 40 are the last to be vaccinated. From the above description, we found that the elderly are the priority group to be considered for vaccination, followed by the middle-aged, and finally, the young. However, as we discovered, this does not apply to the first group (the first part (1)) to be vaccinated. We dived deeper by changing the class attribute to "Health Worker" and discovered that this phenomenon is related to the occupation

of the vaccinated person. (See Figure 6 with health worker (blue mark), and not health worker (light blue mark)) Since the health workers might be considered to have a higher risk than other people, they should be vaccinated at first.⟩

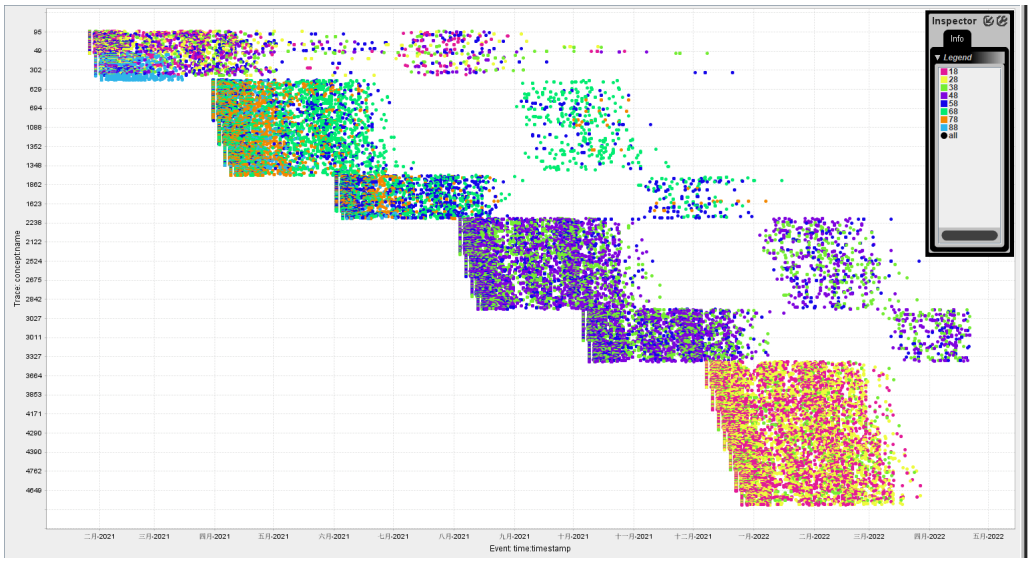

Figure 5: Dotted Chart Analysis over All Events Categorized by "E:Age Category"

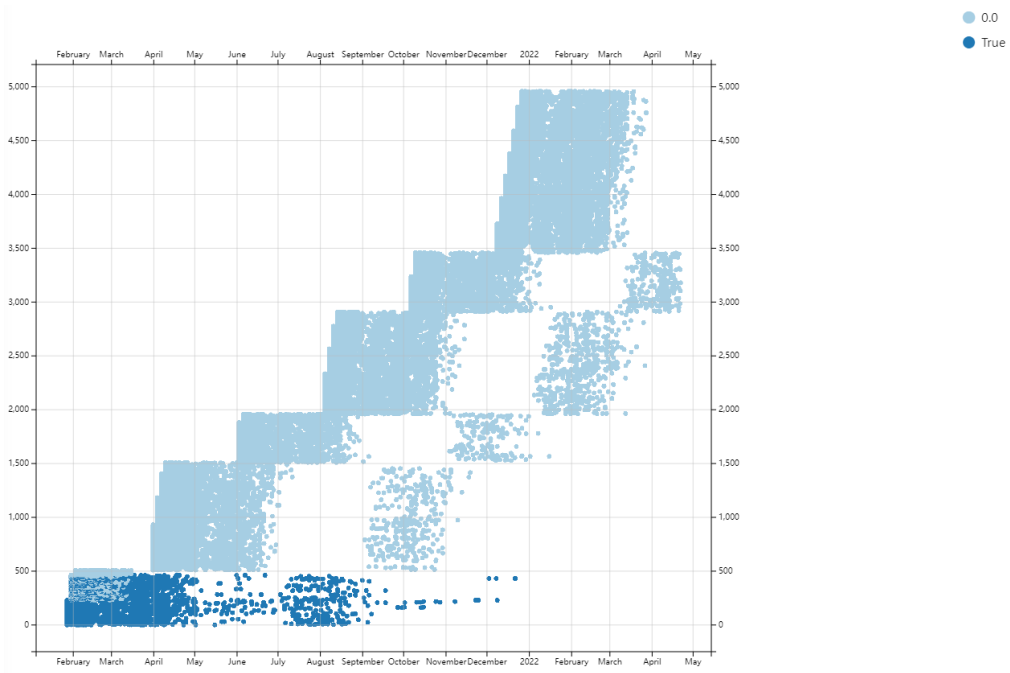

Figure 6: Dotted Chart Analysis over All Events Categorized by Health Worker

d) ⟨Setting of the dotted chart (Figure 7): **X-axis: Throughput time (Elapsed time), Y-axis: E:concept:name, Color attribute: E:concept:name, Trace sorting: Sort on time:timestamp of the first event**.

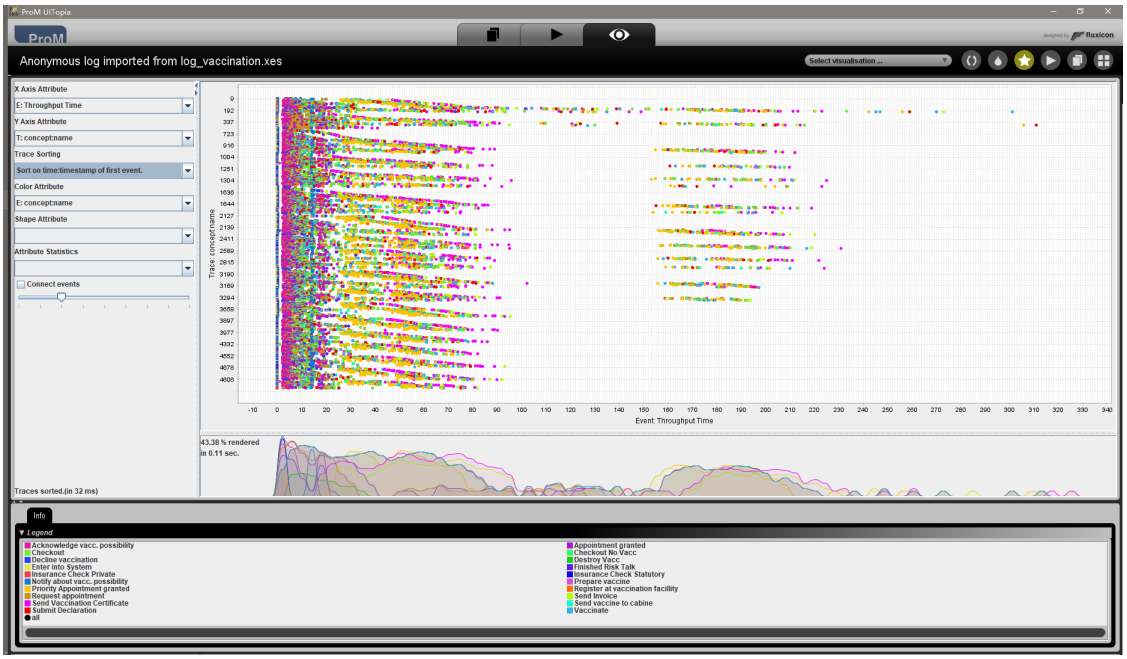

Figure 7: Dotted Chart Analysis over All Events with Throughput Time as x-axis

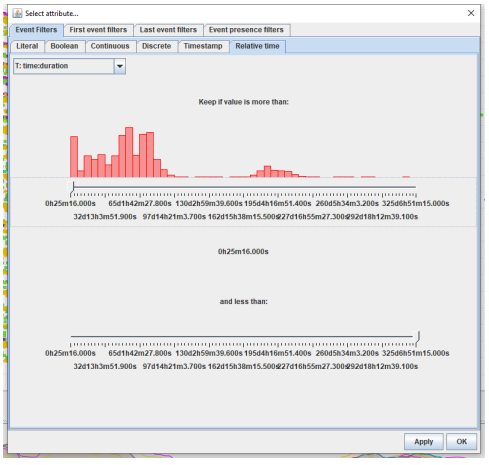

Figure 8: Filter on Elapsed Time

Since most of the traces have a duration less than 90 days, we utilized the filter function of the dotted chart and only focused on the traces that ran over 90 days

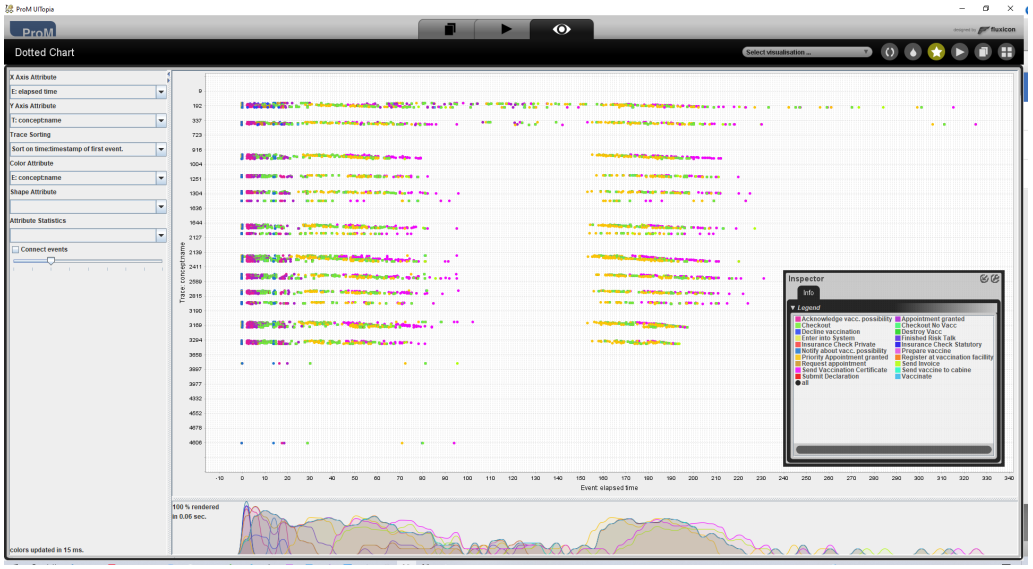

Figure 9: Dotted Chart Analysis over All Events with Throughput Time as x-axis after Filtering

(See setting in Figure 8). Figure 9 illustrates the long lasting run traces. One of the factors contributing to the long-lasting throughput time is that some patients were vaccinated three times. In addition, we found that when the vaccine was first introduced, a small number of patients finally chose to be vaccinated after declining the vaccination for a while (including some patients who repeatedly refused to be vaccinated and regretted their decisions). This phenomenon may be because people are always skeptical about new drugs, so they are in a wait-and-see or tangled state. ⟩

Q2 Process Discovery

⟨This section will compare the performance of several process discovery models evaluated on the given event log according to their four quality dimensions. ⟩

a) ⟨**982 cases** are removed, **3978 cases** were still remained in the event log. The set of end activities for removed traces consists of *Checkout, Send Invoice, Priority appointment granted, Vaccinate.*⟩

b) ⟨Table 1 - 5 displays the value of replay fitness, precision, generalization, and simplicity of *Inductive Miner*, *Heuristic Miner*, *Alpha Miner*, *ILP Miner*, *State Region Miner*. (Petri Net see Appendix Q2b)

- *Inductive Miner*: The resulted model is in neither a spaghetti nor a flower look.

	Replay Fitness	Precision	Generalization	Simplicity
Token-based	0.9655	0.9194	0.9371	0.7959
Alignments	0.9455	0.9201	\	\

Table 1: Four Quality Dimensions for Inductive Miner

- *Heuristic Miner*: The resulted model is in neither a spaghetti nor a flower look.

	Replay Fitness	Precision	Generalization	Simplicity
Token-based	0.9668	0.7994	0.9322	0.6646
Alignments	0.8280	0.7994	\	\

Table 2: Four Quality Dimensions for Heuristic Miner

- *Alpha Miner*: The resulted model looks spaghetti-like. Alignment cannot be applied to this model since the resulted Petri net is not a easy sound net.

	Replay Fitness	Precision	Generalization	Simplicity
Token-based	0.7714	0.3995	0.9727	0.3986
Alignments	\	\	\	\

Table 3: Four Quality Dimensions for Alpha Miner

- *eST Miner*: This algorithm is time-consuming and requires more than 15 minutes to get the result. Since the eST miner needs to find all possible places, it will need exponential time when the number of different events is enormous.
- *ILP Miner*: The resulted model looks spaghetti-like.

	Replay Fitness	Precision	Generalization	Simplicity
Token-based	0.9651	0.9102	0.9846	0.2510
Alignments	0.8835	0.9372	\	\

Table 4: Four Quality Dimensions for ILP Miner

- *State Region Miner*: The resulted model looks spaghetti-like. Alignment cannot be applied to this model since the resulted Petri net is not a easy sound net.

	Replay Fitness	Precision	Generalization	Simplicity
Token-based	0.9548	0.5406	0.8718	0.4737
Alignments	\	\	\	\

Table 5: Four Quality Dimensions for State-Region Miner

Inductive miner results in sound workflow nets based on the process tree generated from the DFG. It guarantees that the original event log can be fully replayed.
Heuristic miner, an enhancement of Alpha miner, considers the order of corresponding traces and the frequency, which means that it is more consistent with the provided event log. This phenomenon can be proven by their values of replay fitness, precision, and generalization of token-based replay. As it is shown in Table 2 and 3, heuristic miner's replay fitness and precision are higher than alpha miner's, but the value of generalization is lower. Furthermore, Alpha miner creates a transition for each activity, and this can result in a lower simplicity value.
ILP miner utilizes the integer linear programming approach, and it can guarantee the consistency between the discovered model and the event log. This will result in an unstructured and unreadable model, explaining why the replay fitness and precision are high. Still, the simplicity is much lower than models discovered by other algorithms.
State region miner derives Petri-net based on a derived transition system. It reduces a lot of concurrency by grouping the places into regions. According to Table 5, the discovered model can have way more behaviors than that in the given event log, explaining the low value of precision. ⟩

c) ⟨The filtered event log now contains **3316 cases** and **ten different events**, including *Register at vaccination facillity*, *Checkout*, *Submit Declaration*, *Vaccinate*, *Enter into System*, *Notify about vacc.possibility*, *Request appointment*, *Finished Risk Talk*, *Acknowledge vacc.possibility. Decline vaccination.*
We did this in PMTK, Figure 10 illustrates how to remove the traces with *Notify about vacc.possibility* followed by itself. In Figure 11, only the traces starting with *Enter into System* are remain. Figure 12 depicts how to remove the specific event from the event log. Here we only provide an example of removing *Insurance Check Statutory.* For renaming, since there is no function provided in PMTK, we did this with PM4PY.⟩

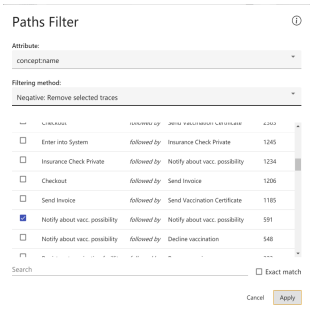

Figure 10: Direct-Follow Filter

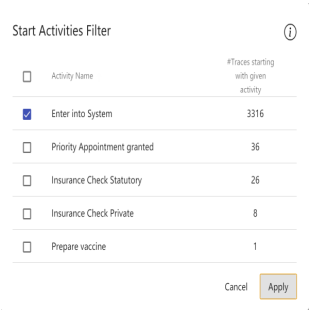

Figure 11: Start Activity Filter

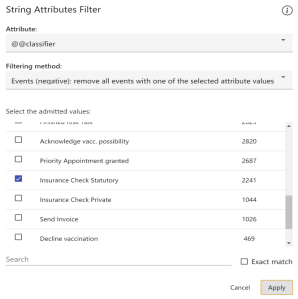

Figure 12: Remove Activity Example

d) ⟨Table 6 - 10 displays the value of replay fitness, precision, generalization, and simplicity of *Inductive Miner*, *Heuristic Miner*, *Alpha Miner*, *ILP Miner*, *State Region Miner*. (Petri Net see Appendix Q2d)

- *Inductive Miner*

	Replay Fitness	Precision	Generalization	Simplicity
Token-based	0.7853	0.8559	0.9812	0.8286
Alignments	0.7833	0.8559	\	\

Table 6: Four Quality Dimensions for Inductive Miner

- *Heuristic Miner*: Alignment cannot be applied to this model since the resulted Petri net is not a easy sound net.

	Replay Fitness	Precision	Generalization	Simplicity
Token-based	0.9566	0.6971	0.8563	0.6944
Alignments	\	\	\	\

Table 7: Four Quality Dimensions for Heuristic Miner

- *Alpha Miner*: Alignment cannot be applied to this model since the resulted Petri net is not a easy sound net.

	Replay Fitness	Precision	Generalization	Simplicity
Token-based	0.7299	0.4485	0.9813	0.5111
Alignments	\	\	\	\

Table 8: Four Quality Dimensions for Alpha Miner

- *ILP Miner*

	Replay Fitness	Precision	Generalization	Simplicity
Token-based	0.9750	0.9990	0.9836	0.3333
Alignments	0.9315	0.9994	\	\

Table 9: Four Quality Dimensions for ILP Miner

- *State Region Miner*: Alignment cannot be applied to this model since the resulted Petri net is not a easy sound net.

	Replay Fitness	Precision	Generalization	Simplicity
Token-based	0.9434	0.8382	0.9246	0.6000
Alignments	\	\	\	\

Table 10: Four Quality Dimensions for State-Region Miner

- *Small Hand Model*

	Replay Fitness	Precision	Generalization	Simplicity
Token-based	0.9934	0.9966	0.9592	0.7255
Alignments	0.9929	0.9966	\	\

Table 11: Four Quality Dimensions for Small Hand Model

behavior modeled by the hand-made model but cannot be discovered by *Inductive Miner*, *Heuristic Miner*, *ILP Miner*:
Start → Enter to system → Notify about vacc. probability → Decline vaccination → End

First, concerning what has been discussed in 2(b) about the ILP miner, it is clear why most performance metrics (except simplicity) are similar to the hand-made model at a satisfying level. In addition, we observed that all discovered models have a delightful generalization value. Since it provides more traces than in the given event log, it helps to avoid overfitting. However, as the number of traces generated by the models is much larger than that in the event log, this can result in lower precision and simplicity values than the hand-made model. ⟩

e) ⟨Four thresholds ranged from 0.0 to 1.0 have been chosen, including **0.0, 0.3, 0.6, 0.9**. From Table 12, we discovered that, as the noise threshold increases, the

performance of the inductive miner model does not change. But the resulted Petri-nets are different.⟩

Noise Threshold	Type	Replay Fitness	Precision	Generalization	Simplicity
0.0	Token-based	1.0	0.1954	0.9366	0.6164
	Alignments	0.9998	0.1954	\	\
0.3	Token-based	1.0	0.1954	0.9366	0.6164
	Alignments	0.9998	0.1954	\	\
0.6	Token-based	1.0	0.1954	0.9366	0.6164
	Alignments	0.9998	0.1954	\	\
0.9	Token-based	1.0	0.1954	0.9366	0.6164
	Alignments	0.9998	0.1954	\	\

Table 12: Four Quality Dimensions for Inductive Miner with Different Noise Threshold

Q3 Conformance Checking

Conformance checking is used to compare a process model with an event log of the same process. To check if the event log conforms to the model. Here we discuss footprint, token-based and alignments.

a) The footprint could compare the relationships between any pair of activities of our log or model, such as directly-follow, directly-before and parallel. According to the Table 13 we could find out that the sequence number of Heuristic Miner is greater than Inductive Miner, while the parallel numbers are the same(both 0). Considering the differences of these two discover methods. Inductive Miner relies on the detection of various cuts on the directly follows graph created using the event log. But Heuristic miner is a noise tolerant algorithm, it is applied to demonstrate the process behavior in noise data. Therefore there are more sequence numbers in Heuristic Miner than in Inductive Miner.

	Inductive Miner	Heuristic Miner
Num. of sequence constructs	22	47
Num. of parallel constructs	0	0

Table 13: Comparison of the models discovered by Inductive Miner and Heuristic Miner

b) Footprint matrix displays the causal dependency of two activities in an event log. Based on the footprint matrices, one can reason that if an event log conforms

with a regarded process model. The fitness is computed as $1 - \frac{\#deviations}{\#relations}$.
Toke-based replay: The fitness between an event log and process model is computed as: $\frac{1}{2}\left(1 - \frac{m}{c}\right) + \frac{1}{2}\left(1 - \frac{r}{p}\right)$, where m is the number of missing tokens, c is the number of consumed tokens, r is the number of remaining tokens, p is the number of produced tokens.The token-based reply technique does not consider the suitable path generated by the model for the unfit cases.

Alignments-based technique can be applied for any process modeling notation and it is guaranteed to find out the most related model run in comparison to the trace.

The three different conformance checking techniques using different calculating metrics for fitness, but from the result of comparison we can see that the fitness are generally over 90%.

Based on footprint matrix

- Fitness: 90.48%
- Precision: 100%

Token based replay

- percentage fit traces: 57.34%
- Average trace fitness: 96.20%
- Log fitness: 96.96%
- Percentage of fitting traces: 57.24%
- Precision: 96.56 %

Alignment

- percentage of fitting traces: 57.34%
- Average trace fitness: 95.84%
- Average fitness: 95.84%
- log fitness: 95.99%
- Precision: 96.56 %

c)

- Flower model and log vaccination finished cases:
  footprints-based
  - Fitness: 100%
  - Precision: 17%

  Alignments-based
  - Fitness: 100.00%
  - Precision: 13.26%

  Compare with the results from task b, we could find out that the fitness here are higher than before(task b), but the precision is extremely lower. First of all if we check the flower model we could find out that all the relationships of each pair of the activities are parallel. The flower model can describe all the behaviors which might happen. Therefore the fitness are 100%. But at the same time, the flower model is too generalized so that the precision could be super low.
- Full hand model and one trace log using token-based, alignment-based.
  Token-based reply
  - percentage of fitting traces: 0%
  - Average trace fitness: 93.33%
  - Log fitness: 93.33%
  - Precision: 57.14%

  Alignment
  - percentage of fitting traces: 0%
  - Average trace fitness: 62.07%
  - Log fitness: 62.07%
  - Precision: 57.14%

According to the results above we could find out these two methods have the same precision but meanwhile alignment has the lower fitness. Considering the differences between token-based replay and alignments: The token-based technique is based on an assessment of the states of the process model when replaying the events of a trace. The model doesn't consider the behaviour recorded in the event log when the log unfit the model. And alignment considers not only the model but also the recorded behaviour.

In our case if we compute roughly, the fitness of token based reply, we could find out that there is only 1 missing token and 0 remaining token, since the fitness is $\frac{1}{2}\left(1-\frac{m}{c}\right)+\frac{1}{2}\left(1-\frac{r}{p}\right)$, the result could be very large. But for alignment, the shortest path of model is 3, the length of our trace is 25. If we observe the trace we could see that there is no cost for the first 3 movements, then for the following 11 activities all repeat once in log and related to the model we could have at least 11 cost. Therefore the fitness of alignment is $1-\frac{11}{3+25} \approx 60\%$.

d.i) By observing fig. 13 and fig.14, we could find that the places(indicated by the red arrows) are with lower fitness and precision respectively.
According to fig.13 the places right after "Insurance Check Private" and "Insurance Check Statutory" are with lowest places fitness, which are around 78% and 70%.

According to fig.14 For example place 2(indicated by the red arrow) with lower precision around 50.4%.

The mentioned places could be the problematic part of this process.

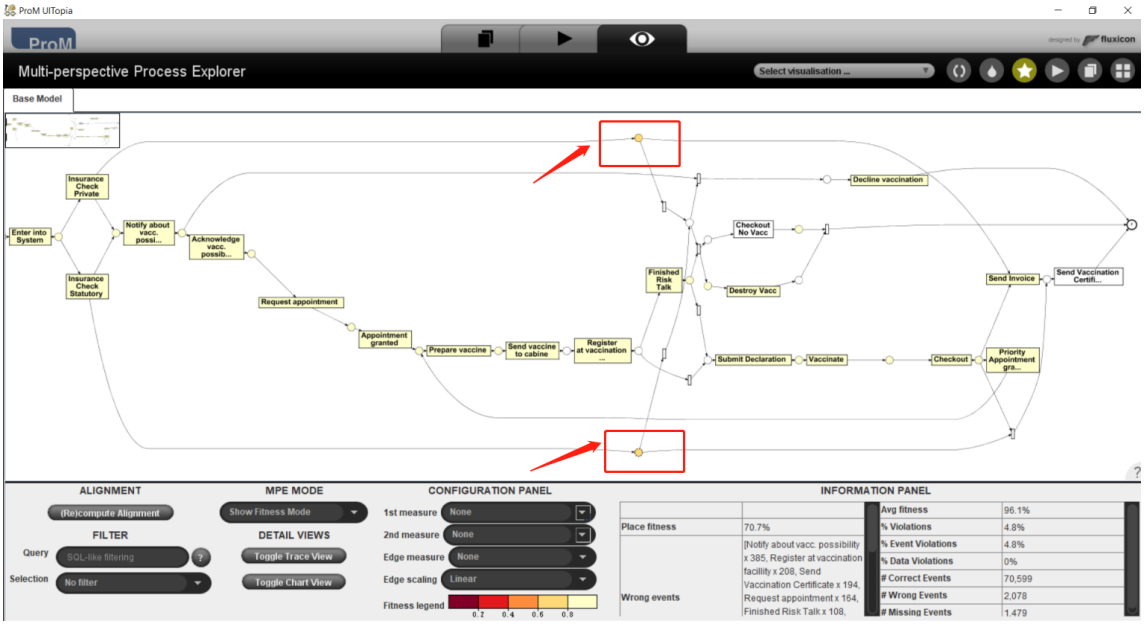

Figure 13: Q3.d.i Multi-perspective Process Explorer_fitness

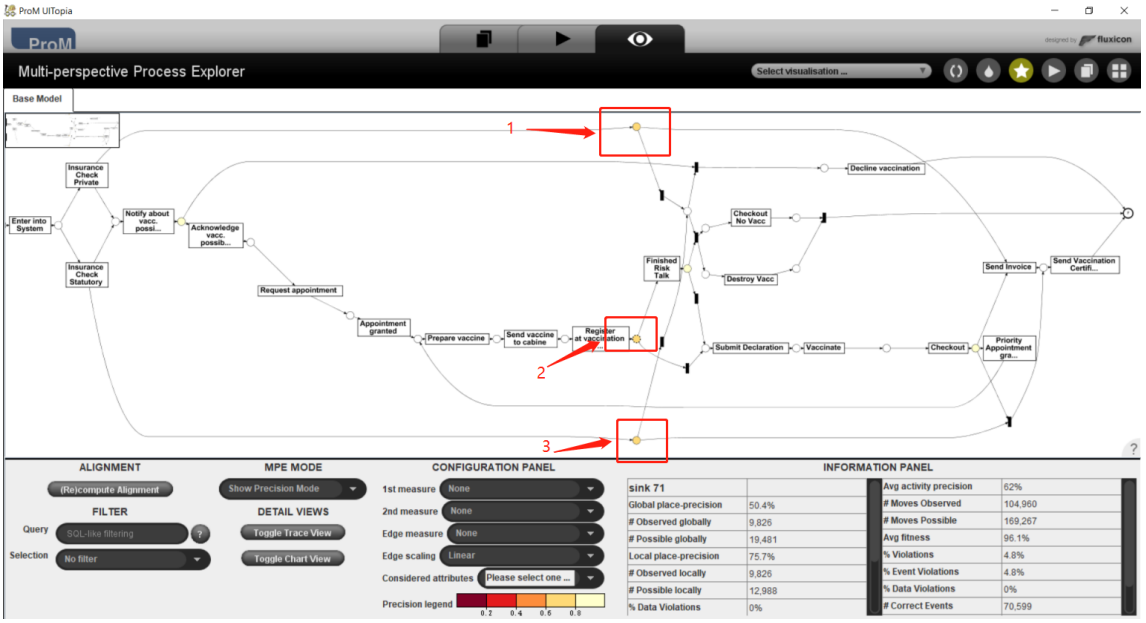

Figure 14: Q3.d.i Multi-perspective Process Explorer_precision

d.ii) We choose the fitness ascending, then got fig.15.

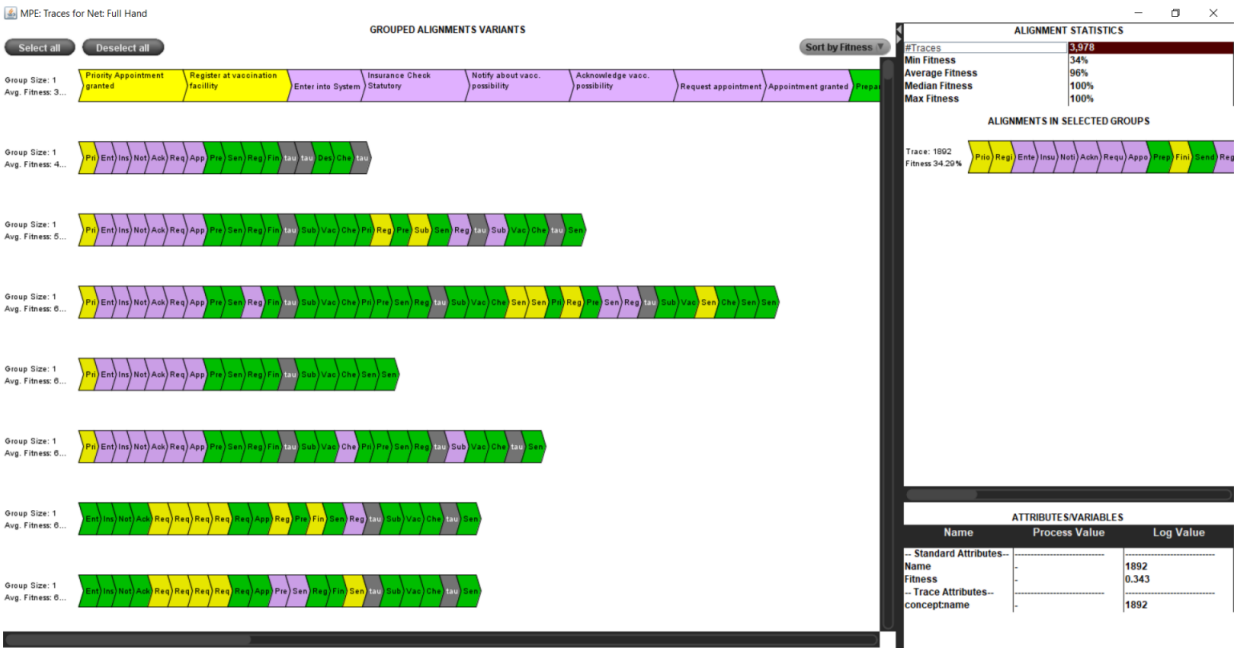

Figure 15: Fitness ascending

1. Then choose the first one, which is with the lowest fitness(fig.16). We noticed that the first two activities are yellow, which means the wrong event, only log move. It indicates that this trace doesn't start with the proper activity i.e. *Enter into System* but starts with *Priority Appointment granted*, is directly followed with the *Prepare vaccine*. According to the fig.15 we could see that the first six traces with low fitness all have the same problem which could be a noise problem.

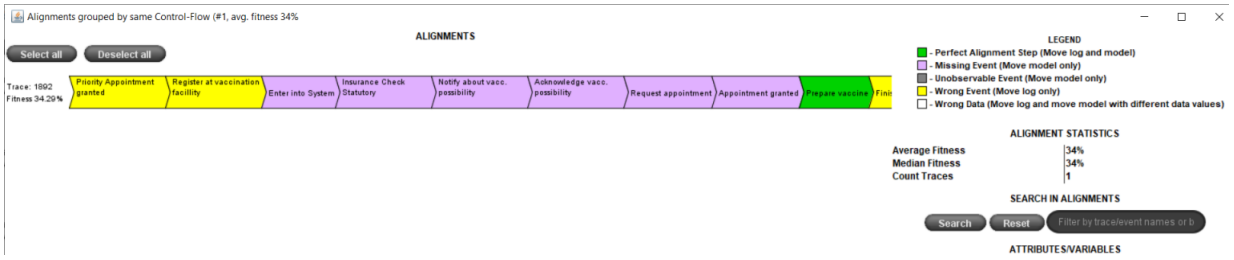

Figure 16: Problem1

2. Still the first trace, according to the fig.17, that *Finished Risk Talk* happens before *Send vaccine to cabine*. If we check the model, the order is wrong. The right order should be ... $\rightarrow$ *Prepare vaccine* $\rightarrow$ *Send vaccine to cabine* $\rightarrow$ *Register at vaccination facility* $\rightarrow$ *Finished Risk Talk* $\rightarrow$ ....

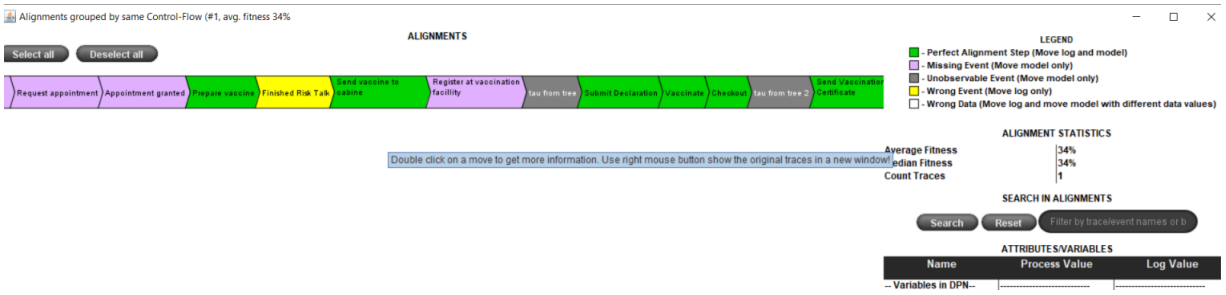

Figure 17: Problem2

3. Another traces(fig.18), we could see that *Notify about vacc.possibility* may happen twice. Since the first one is in yellow, only log move, then the second one is green, the perfect alignment. This problem could be a desired behaviour, since there is possibility that the system send the notice to the patient but he doesn't give any reaction or ignore it. Then the system may do it again.

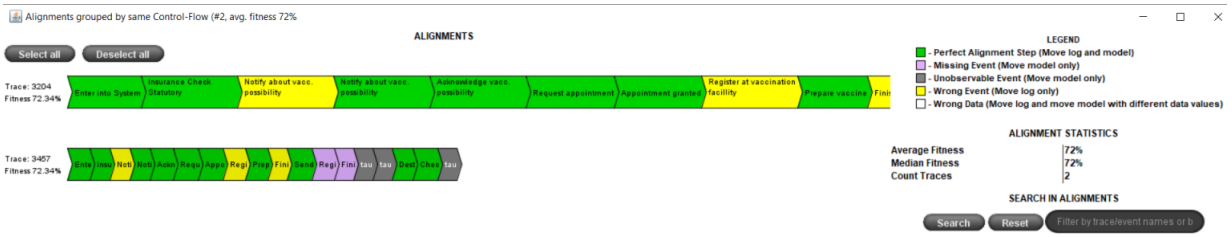

Figure 18: Problem3

4. According to the fig.19 we could see that *Request appointment* happens multi times. But the system doesn't deal with it, which should be improved.

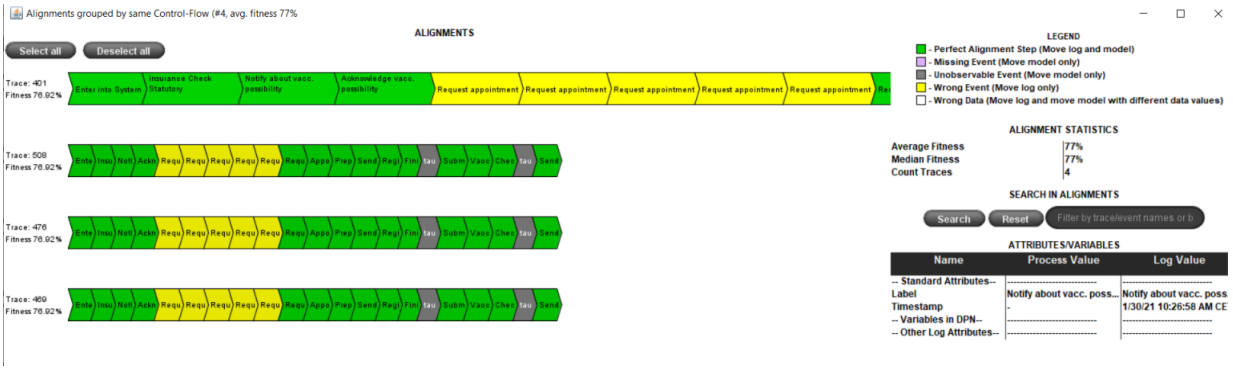

Figure 19: Problem4

Q4 Performance Analysis

Based on performance analysis which mainly focuses on the throughput time between several events, we can detect the bottlenecks within the process and the potential reason of them.

a) ⟨Figure 20 illustrates the performance spectrum of *Checkout* → *Send Invoice*. Based on the petri-net, the activity Checkout is followed by Send Invoice 1501 times. Among these instances, we discovered that after execute the activity *Checkout*, the invoice can be sent to most patients (87% of paths between "Checkout" and "Send Invoice") within one month. 8% of the instances will execute send invoice more than one month but within two months after Checkout. However, there are still 5% of instances where patients need to wait 2-3 months to have their invoice to be sent. The minimal and maximal duration between these two activities are 13 days 23 hours and 90 days 4 hours.

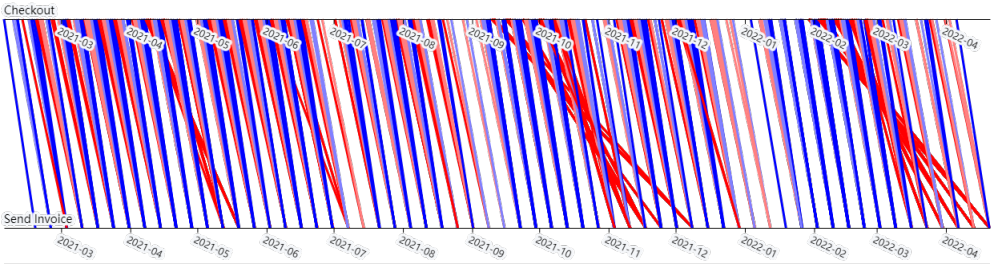

Figure 20: Performance Spectrum of *Checkout* → *Send Invoice*

Figure 21 depicts the performance range of *Checkout* → *Priority Appointment granted*. According to the different timestamps of Checkout, it can be divided into six similar parts (see Figure 22 for an example part), each of which consists of two months. In each part, the time interval between Checkout and Priority Appointment are shorter in the first month but relatively much longer in the second month. One factor contributing to this is that some patients checkout both in the first month and the second month (see Figure 23), which means they have been vaccinated twice, so the next appointment should be for their third vaccination. As discussed in Q1, we know that people always wait longer for the third vaccine. As discussed above, some cases in which Checkout is completed in the first month need to be handled in the second month. As a result, the workload will be higher, explaining why in some cases, when Checkout was completed for the first time in the second month, patients still waited several months to get an appointment.

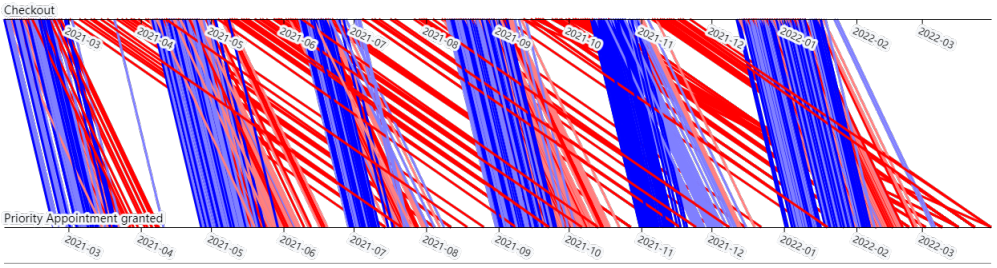

Figure 21: Performance Spectrum of *Checkout* → *Priority Appointment granted*

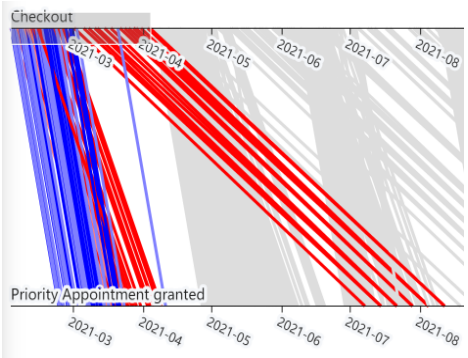

Figure 22: Performance Spectrum of *Checkout → Priority Appointment granted* Part Selected1

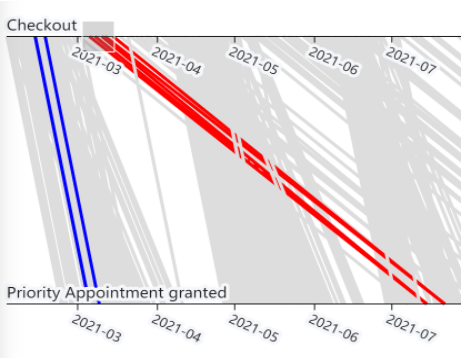

Figure 23: Performance Spectrum of *Checkout → Priority Appointment granted* Part Selected2

⟩

b) ⟨In this event log, the only events following *Decline Vaccination* is *Request appointment*. Figure 24 depicts the performance spectrum of the path *Decline Vaccination → Request Appointment*. We discovered that all decline happened between 07.02,2021 and 16.02.2021, which was the starting period of the whole vaccination process. Among these, after declining, 50 % of patients decided to request an appointment for vaccination within 80 days; others did this after more than this threshold.⟩

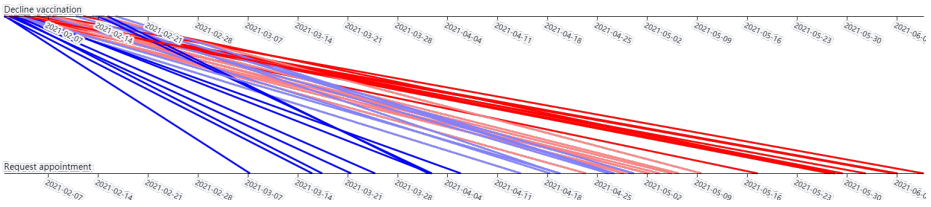

Figure 24: Performance Spectrum of *Decline Vaccination → Request Appointment*

c) ⟨Figure 25 depicts the resulted decision tree (X[4]: 'event:org:resource@Luka', X[64]: 'succession:concept:name@Send Invoice#Send Vaccination Certificate', X[50]: 'succession:concept:name@Insurance Check Statutory#Notify about vacc. possibility', X[19]: 'event:concept:name@Insurance Check Statutory'), where the blue nodes represent the traces with an illegal process, and the orange nodes represent the traces with the legal process. This decision tree demonstrates that if the event log has a resource from Luka and Luka did not notify the patients about the possibility of the vaccination directly after checking the statutory of their insurance, it has a

high probability of violating the rules set by the organization. Only 55 out of 4209 instances are classified as illegal if Luka does not handle them.⟩

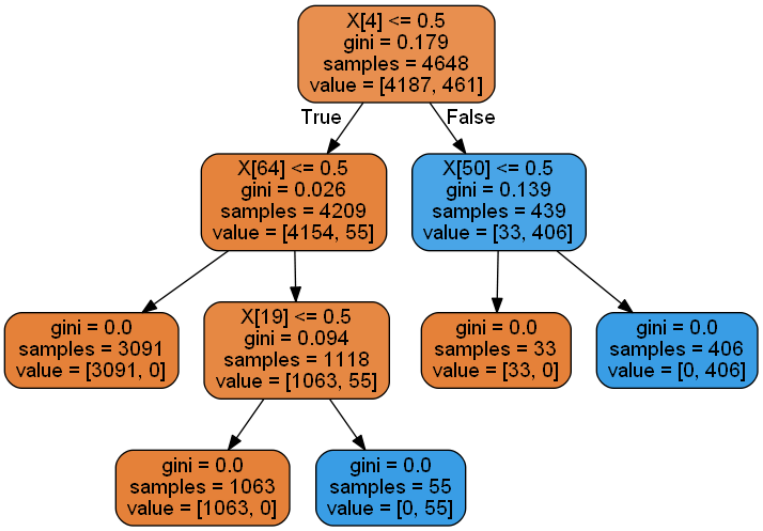

Figure 25: Decision Tree

Q5 Conclusion

In the first we got the overview of the process, include the basic statics of the event log, find out the some usual patterns for traces, for example usually start with *Enter to System Insurance Check Private/Insurance Check Private Statutory Notify about vacc. possibilities*,followed by a appointment granted sub-process. After that, patients can have their first vaccination. They can receive a certificate after their second vaccination. Besides, elderly have higher priority to be vaccinated, but health worker are the first group considered to be vaccinated. We also discovered that, the possible reason for the long lasting cases could be the multiple times decline of vaccination by patient or a three-times vaccination.

In the second part we utilized several different algorithms to discover the process models for the given event log. Then based on the four dimension evaluated different models. According to the evaluation matrix we find out, without considering the simplicity, the ILP Miner has the best performance. If consider it then the Inductive Miner could be the best. Besides, prepossessing of the event log could influence performance of the models. So appropriate filter on the event log is necessary under some situations.

Then we do the conformance checking on the discovered models. We compared the differences of three different methods, including footprint, token-based and align-

ments. We find out that footprint could describe the relationships of each pair activities, and token-based focus on the traces replay with the state of the process model, the limitation is that if the event log is unfit the model then the model doesn't consider. So alignment improve this problem, consider both the model and the recorded behaviour.

At last we based on performance analysis which mainly focus on the throughput time, and find some bottlenecks, for example, some patients may wait 2-3 months to receive their invoices after checkout. Besides, they perhaps don't distinguish the appointments of the second and third vaccination, so a lot of cases were backlogged. Lots of patients needed to wait for long times more than three or four months to get an appointment for their second vaccination, which should be avoided.

Finally above all the analysis we could find some problems such as illegal processes which have an appointment without first entering the system, should be noticed. There has been a problem with respect to that system doesn't react the request of appointment from the patient sides, therefore we would recommend to hire more employees. Besides we discover that there are some inappropriate orders of events or some employees made mistakes when they dealt with some cases, which may violate the rules of the organization. This can be avoided by the training of the employees or enhancing the management.

Appendix

Q2b:

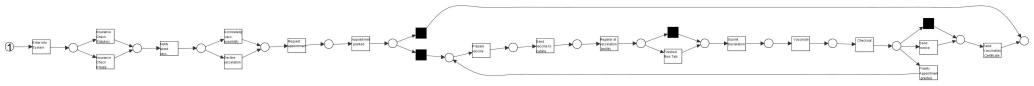

Figure 26: Q2b Inductive Miner

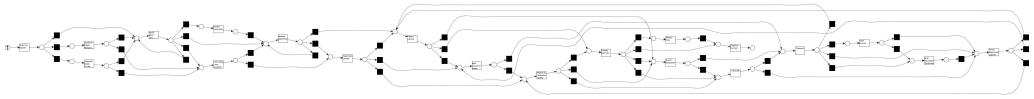

Figure 27: Q2b Heuristic convert

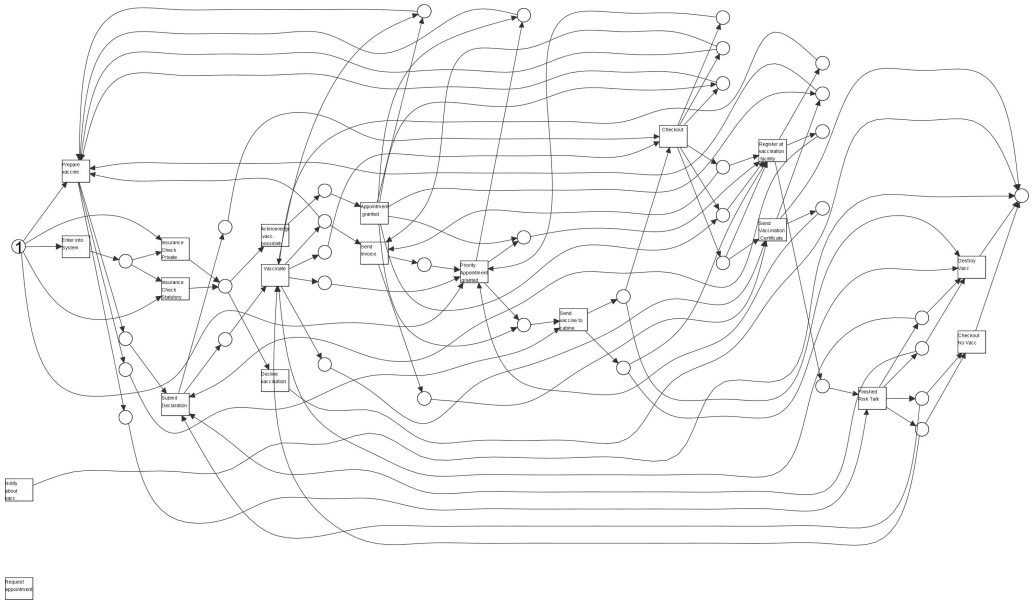

Figure 28: Q2b Alpha

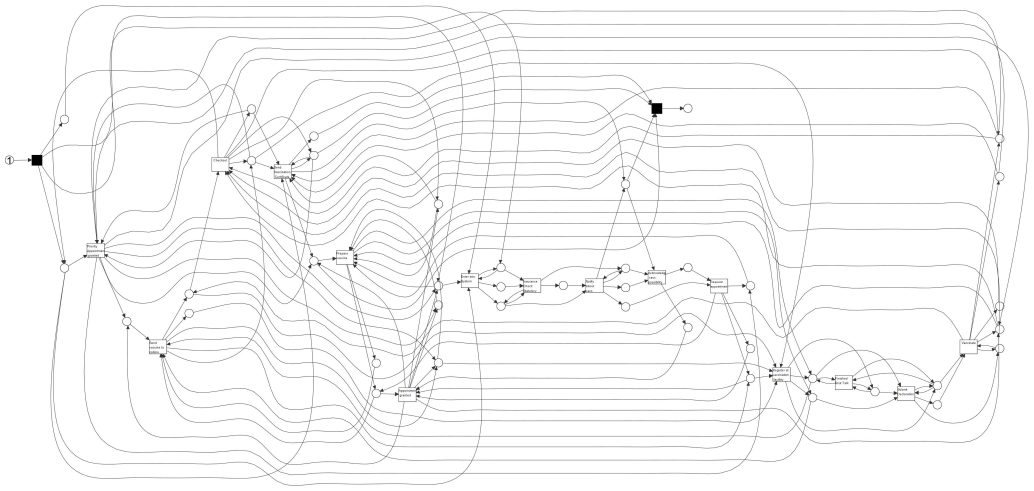

Figure 29: Q2b ILP

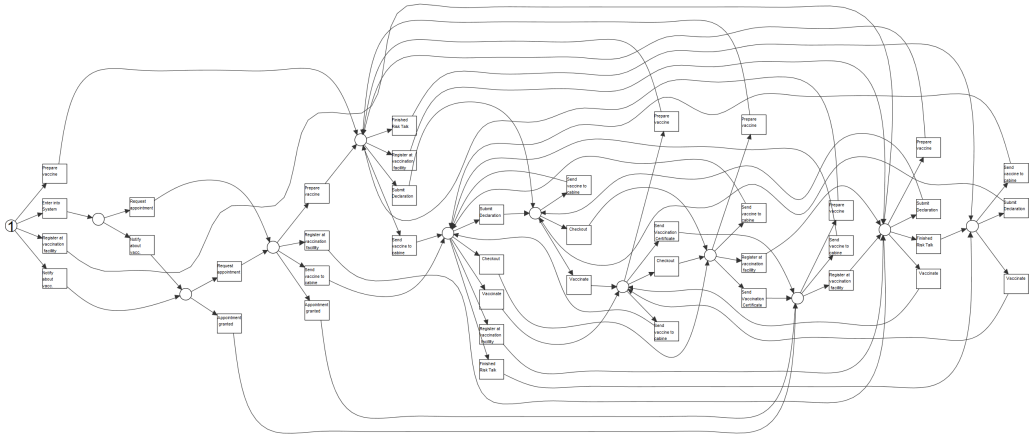

Figure 30: Q2b State Region

Q2d:

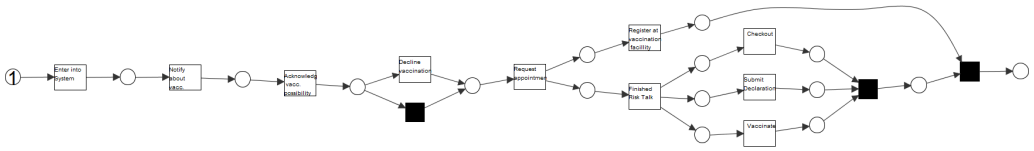

Figure 31: Q2d Inductive Miner

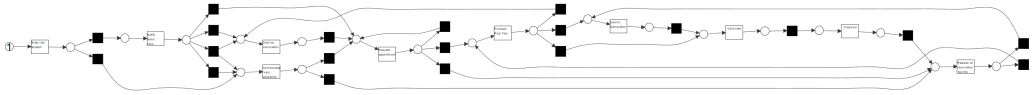

Figure 32: Q2d Heuristic convert

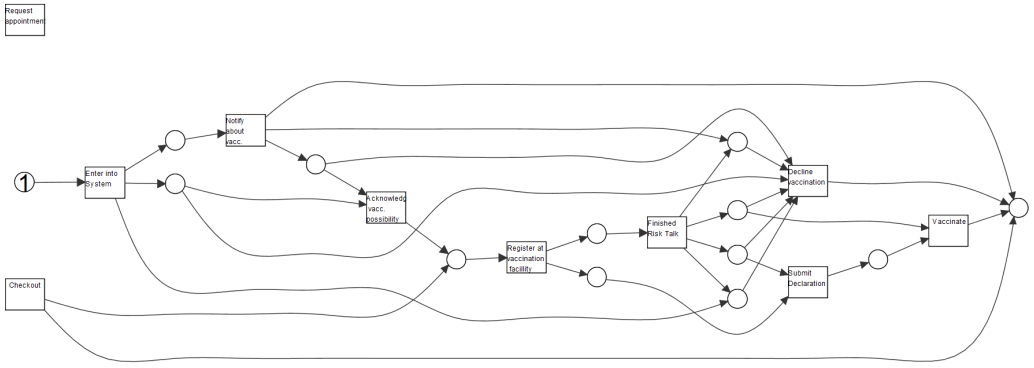

Figure 33: Q2d Alpha

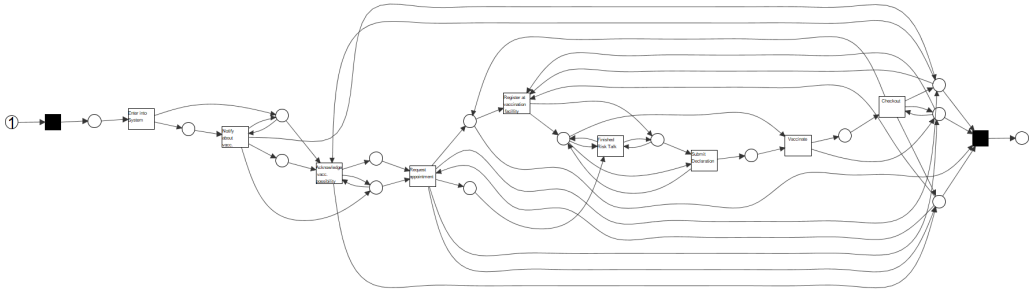

Figure 34: Q2d ILP

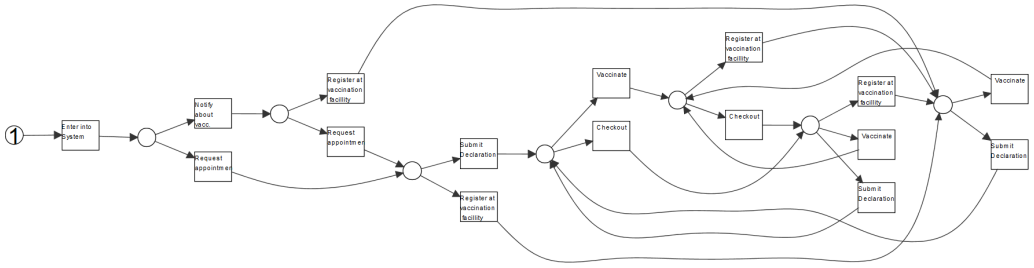

Figure 35: Q2d State Region