

Conformance Checking for a Medical Training Process using Petri Net Simulation and Sequence Alignment

An Nguyen¹[0000-0002-8759-7641] and Bjoern M. Eskofier^{2,3}[0000-0002-0417-0336]

Department of Computer Science,
Friedrich-Alexander-University Erlangen-Nürnberg (FAU), Erlangen, Germany
{an.nguyen, bjoern.eskofier}@fau.de

Abstract. Process Mining has recently gained popularity in healthcare due to its potential to provide a transparent, data-based view on processes. A common question is how actual process executions deviate from existing guidelines. Conformance checking is a sub discipline of process mining which aims to answer such questions. In this work, a medical training process provided by the Conformance Checking Challenge 2019 (CCC19) is analyzed. Ten students were trained to install a Central Venous Catheters (CVC) with ultrasound. Event log data was collected at an initial test after they were taught how to perform the procedure by their instructors and after a following, individual practice period. Students and instructors are particularly interested in the comparison of performance between these two trials with respect to the process guideline. In order to provide objective performance measures for that, we formulated an optimal, global sequence alignment problem inspired by approaches in bioinformatics. An event log can be modeled as a set of sequences. Therefore, the Petri Net model representation of the medical process guideline was used to simulate a representative set of guideline conform sequences (event log). Next, the optimal, global sequence alignment of the recorded and representative event logs was calculated. Finally, the output measures were used to provide objective feedback for the students and instructors.

Keywords: Process Mining · Conformance Checking · Bioinformatics · Sequence Alignment · Medical Process

1 Introduction

Process mining deals with the extraction of insights from event logs. Its major groups of methods are process discovery, conformance checking and process enhancement [1]. Event logs analyzed in process mining are usually grouped into so called *case IDs*, which consists of a sequence of *activities* including *time stamps*. Process mining gained recently popularity for healthcare applications as indicated in the reviews [2] and [3]. There exist many guidelines for the many

processes in hospitals. A common question is whether the executed processes are compliant with existing guidelines. Conformance checking is an approach which aims to check how the recorded event log data from actual process executions match a given normative model representing a guideline. Such models are often given in workflow languages like Petri nets and BPMN [1]. Most conformance checking methods in process mining are based on replaying an event log on a Petri net [1] and many of them are implemented in the freely available process mining tool ProM¹. There is also a process mining framework in Python currently in development². In this work, a medical training process provided by the Conformance Checking Challenge 2019 (CCC19) [4] was analyzed as a case study. The goal was to provide objective performance measures for both students and instructors. This was done by formulating the conformance checking problem as an optimal, global alignment problem.

2 Preliminaries

In the following are definitions given for activities, sequences and event logs as referred to in this work. Formal definitions are given followed by an example for illustration.

Definition 1. (event logs and sequences) Let \mathbb{A} be the set of all activities of a process \mathbb{P} . A sequence of activities is described by a mapping $\sigma \in \{1, \dots, n\} \rightarrow \mathbb{A}$, where n is the length of the sequence ($\text{len}(\sigma) = n$). $\sigma(i)$ denotes the i^{th} element of the sequence, for $1 \leq i \leq n$. A sequence σ can be denoted by $\langle a_1, a_2, \dots, a_n \rangle$, where $a_i = \sigma(i)$. An event log \mathbb{L} is a multiset of sequences σ . We denote $\sigma_{i,\mathbb{L}}$ as the i^{th} sequence in the event log \mathbb{L} . $N_{\text{seq}} = |\mathbb{L}|$ and $N_{\text{act}} = |\mathbb{A}|$ are the number of sequences and unique activities in an event log \mathbb{L} and process \mathbb{P} respectively.

Definition 2. (stages and stage event log) Let \mathbb{A} be the set of all activities of a process \mathbb{P} . We map each activity in \mathbb{A} to a finite set of stages $S_j \in \mathbb{S}$ for $1 \leq j \leq m$, where m denotes the number of stages in a process \mathbb{P} . Assume an event log \mathbb{L} over a set of activities \mathbb{A} with stages $S_j \in \mathbb{S}$. We define the 'Stage event log' \mathbb{L}_{S_j} as the event log, where the sequences $\sigma_{i,\mathbb{L}_{S_j}} = \langle a_{S_j, \text{first}}, \dots, a_{S_j, \text{last}} \rangle$ are derived from sequences $\sigma_{i,\mathbb{L}} = \langle a_1, \dots, a_n \rangle$ with $a_{S_j, \text{first}}$ and $a_{S_j, \text{last}}$ being the first and last activity in $\sigma_{i,\mathbb{L}}$ assigned to stage S_j respectively. $\sigma_{i,\mathbb{L}_{S_j}}$ includes all activities from $\sigma_{i,\mathbb{L}}$ including and between $a_{S_j, \text{first}}$ and $a_{S_j, \text{last}}$.

As an example assume the simplified process of a doctor visit \mathbb{P} , with a set of activities $\mathbb{A} = \{\text{enter building (a), show ID (b), go to waiting room (c), wait (d), examination (e), check out (f), get prescription (g)}\}$ and stages $S_1 = \{a, b, c, d\}$ and $S_2 = \{e, f, g\}$. Let $\mathbb{L}_{\mathbb{P}} = \{\langle a, b, c, e, g \rangle \langle b, c, f, g \rangle \langle a, c, d, e, f, g \rangle\}$ be an event log recorded during one hour. Due to Definition 2 the corresponding stage event logs are $\mathbb{L}_{\mathbb{P}, S_1} = \{\langle a, b, c \rangle \langle b, c \rangle \langle a, c, d \rangle\}$ and $\mathbb{L}_{\mathbb{P}, S_2} = \{\langle e, g \rangle \langle f, g \rangle \langle e, f, g \rangle\}$.

¹ <http://www.promtools.org/doku.php>

² <http://www.pm4py.org/>

In this work we considered two event logs for analysis (plus their stage event logs). The first one was based on the student recordings $\mathbb{L}_{student}$ and the second one was based on the simulation of the provided Petri net \mathbb{L}_{norm} . Given two sequences σ_1 and σ_2 , an (optimal) alignment can be calculated as defined as follows.

Definition 3. (alignment) Given two sequences $\sigma_1 = \langle a_1, a_2, \dots, a_n \rangle$ and $\sigma_2 = \langle b_1, b_2, \dots, b_m \rangle$ of lengths n and m respectively. An alignment is the assignment of gaps denoted as '-' to positions $0, \dots, \max(m, n)$ in σ_1 and σ_2 , such that each activity in one sequence is lined up with either an activity or gap in the other sequence. The resulting sequences based on such alignment are referred to as σ_1^* and σ_2^* .

Definition 4. (optimal alignment) Given two aligned sequences σ_1^* and σ_2^* , both of length N . Let A be the set of all unique activities from the underlying process \mathbb{P} . We will define the following score between two activities $\sigma_1^*(i)$ and $\sigma_2^*(i)$ for $1 \leq i \leq N$.

$$d(a, b) = \begin{cases} \text{match} \in \mathbb{R}^+ & \text{if } a = b \text{ and } a, b \in A \\ \text{gap} \in \mathbb{R}^- & \text{if } a = '-' \text{ or } b = '-' \\ \text{mismatch} \in \mathbb{R}^- & \text{if } a \neq b \text{ and } a, b \in A \end{cases}$$

An optimal alignment of two sequences $\sigma_1 = \langle a_1, a_2, \dots, a_n \rangle$ and $\sigma_2 = \langle b_1, b_2, \dots, b_m \rangle$ is defined by:

$$\max_{\sigma_1^*, \sigma_2^*} \sum_{i=1}^N (d(\sigma_1^*(i), \sigma_2^*(i)))$$

Definition 5. (identity) Given two aligned sequences σ_1^* and σ_2^* of length N . We define the identity as the percentage of matching activities in these sequences.

$$\text{identity} = \frac{100}{N} \cdot \sum [\sigma_1^*(i) = \sigma_2^*(i)], \sigma_1^*(i) \text{ and } \sigma_2^*(i) \in A$$

3 Data

The data analyzed in this study was provided by the Conformance Checking Challenge 2019 (CCC19) [4]. Ten students were trained to install a Central Venous Catheters (CVC) with ultrasound. Event log data was collected at an initial test after they were taught how to perform the procedure by their instructors (pre trial) and after a following, individual practice period (post trial) based on tagged video recordings. The resulting event log $\mathbb{L}_{student}$ contains therefore 20 sequences. The process can be divided into six subsequent stages: 'Operator and Patient Preparation' (S_1), 'Ultrasound Preparation' (S_2), 'Locate Structures' (S_3), 'Venous puncture' (S_4), 'Install Guidewire' (S_5) and 'Install Catheter' (S_6). Each of these stages include a set of activities. All *activities* $\in \mathbb{A}$ and their corresponding stages in the process are listed in Table 2. Students and instructors are particularly interested in the comparison of performance between both trials with respect to the process guideline. The normative process model represented

as a Petri net is shown in Figure 1 including all activities and stages of the process. Note that the activities in the Petri net model 'INVISIBLE No good position' and 'INVISIBLE No Return' were not recorded in the given event log of the students. Table 1 shows the reference for the students used in the event log and this work. Furthermore, an event log \mathbb{L}_{norm} based on simulations of the given Petri net was analyzed as a representation of the process guideline as described in the following section.

Table 1: Mapping of the 'RESOURCE' column in original event log with references used in this work.

RESOURCE	Student
R_13_1C	1
R_14_1D	2
R_21_1F	3
R_31_1G	4
R_32_1H	5
R_33_1L	6
R_45_2A	7
R_46_2B	8
R_47_2C	9
R_48_2D	10

4 Methods

The overall approach of this work was to use the given Petri net representing the process guideline to simulate an representative event log \mathbb{L}_{norm} . The maximal global alignment between each case (sequence) of the event log recorded from the process executions of the students $\mathbb{L}_{student}$ and all simulated normative cases in \mathbb{L}_{norm} was then computed to provide objective performance measures.

4.1 Event log simulation

The Petri net shown in Figure 1 was used in combination with the ProM plug-in '*Perform a simple simulation of a (stochastic) Petri net*' to generate a representative event log \mathbb{L}_{norm} of the normative model for the medical process analyzed in this study. The number of sequences (cases) to be simulated was set to 1.000 and the maximum number of activities as 65. In a post processing step only simulated sequences which completed the process (ended with the activity 'Check catheter position') were kept for further analysis in \mathbb{L}_{norm} . Furthermore, the activities 'INVISIBLE No good position' and 'INVISIBLE No Return' were removed from \mathbb{L}_{norm} . We observed that the shortest and longest student sequence

Table 2: All activities of the process with the assigned stages and used abbreviations.

Activity	Stage	Representation
Prepare Implements	Operator and Patient Preparation (S_1)	1a
Hand washing	Operator and Patient Preparation (S_1)	1b
Get in sterile clothes	Operator and Patient Preparation (S_1)	1c
Clean puncture area	Operator and Patient Preparation (S_1)	1d
Drap puncture area	Operator and Patient Preparation (S_1)	1e
Ultrasound configuration	Ultrasound Preparation (S_2)	2a
Gel in probe	Ultrasound Preparation (S_2)	2b
Cover probe	Ultrasound Preparation (S_2)	2c
Put sterile gel	Ultrasound Preparation (S_2)	2d
Position Probe	Ultrasound Preparation (S_2)	2e
Position patient	Locate Structures (S_3)	3a
Anatomic identification	Locate Structures (S_3)	3b
Doppler identification	Locate Structures (S_3)	3c
Compression identification	Locate Structures (S_3)	3d
Anesthetize	Venous puncture (S_4)	4a
Puncture	Venous puncture (S_4)	4b
Blood return	Venous puncture (S_4)	4c
Drop probe	Install Guidewire (S_5)	5a
Remove syringe	Install Guidewire (S_5)	5b
Guidewire install	Install Guidewire (S_5)	5c
Remove trocar	Install Guidewire (S_5)	5d
Check wire in short axis	Install Guidewire (S_5)	5e
Check wire in long axis	Install Guidewire (S_5)	5f
Wire in good position	Install Guidewire (S_5)	5g
Widen pathway	Install Catheter (S_6)	6a
Advance catheter	Install Catheter (S_6)	6b
Remove guidewire	Install Catheter (S_6)	6c
Check flow and reflow	Install Catheter (S_6)	6d
Check catheter position	Install Catheter (S_6)	6e

in $\mathbb{L}_{student}$ were of lengths 26 and 59 respectively. The distribution of sequence lengths per case in \mathbb{L}_{norm} covered all possible and valid values in that range after post processing. One can observe from the given Petri net that there are only 4 places (p12, p16, p20 and p22) where due to the process guidelines a decision can be made. Therefore, \mathbb{L}_{norm} inherently models these constrains and only specific and valid sequence lengths can be generated via simulation of the Petri net.

4.2 Sequence alignment

In this work we aimed to provide an objective measure on how well a student installed a CVC with ultrasound with respect to given guidelines. For the overall performance analysis we used the event logs $\mathbb{L}_{student}$ and \mathbb{L}_{norm} . In order to give

more detailed feedback based on the performance in the different stages (S_1, \dots, S_6) we extracted the corresponding event logs $\mathbb{L}_{student_{S_i}}$ and $\mathbb{L}_{norm_{S_i}}$ for $1 \leq i \leq 6$ according to definition 2. In order to compute the optimal alignment between two sequences as in Definitions 3 and 4, we applied the Needleman-Wunsch algorithm [5, 6] with the scores: $match = 1$, $gap = -2$ and $mismatch = -2$. The Needleman-Wunsch algorithm is commonly used in bioinformatics to align protein or nucleotides sequences. The algorithm follows a dynamic programming approach to find an optimal (global) alignment of two sequences. Its worst-case complexity is $O(mn)$ with m and n being the lengths of the two sequences to be aligned. The two main steps of this algorithm are:

- construction of a scoring matrix
- finding the optimal path by maximizing the score as in definition 4 in the scoring matrix

We refer to the original publications of the algorithm for more details on these steps [5, 6]. In this work a Python implementation of the algorithm ³ was used which can handle a sequence of strings as an input. For each sequence in the student event logs $\mathbb{L}_{student}$ and $\mathbb{L}_{student_{S_i}}$, we calculated the optimal alignment with all sequences in the corresponding normative event logs \mathbb{L}_{norm} and $\mathbb{L}_{norm_{S_i}}$. Specifically, we found the maximum identity (Definition 5) of each sequence in $\mathbb{L}_{student}$ or $\mathbb{L}_{student_{S_i}}$ with all sequences in \mathbb{L}_{norm} or $\mathbb{L}_{norm_{S_i}}$ respectively. The assumption was that this would yield the maximum conformance of a specific student sequence with the guidelines modeled by \mathbb{L}_{norm} and $\mathbb{L}_{norm_{S_i}}$. Furthermore, we calculated the duration of each sequence in $\mathbb{L}_{student}$ and $\mathbb{L}_{student_{S_i}}$. Finally, the maximum identity and duration for the whole process and each of the stages are visualized as a feedback for the students and instructors.

5 Results

The output generated by our analysis provides different objective performance measures for the students and instructors. First of all, the optimal aligned sequences of student executions in $\mathbb{L}_{student}$ and the guidelines modeled as \mathbb{L}_{norm} are given. Then analysis results of the duration and identity are presented for the whole process based on $\mathbb{L}_{student}$ and \mathbb{L}_{norm} , and the stages based on $\mathbb{L}_{student_{S_i}}$ and $\mathbb{L}_{norm_{S_i}}$.

5.1 Optimal Alignment Sequences

As an example output of the optimal alignment approach, are the optimal aligned sequences for student 1 and post trial (from $\mathbb{L}_{student}$) and the sequence from \mathbb{L}_{norm} with maximum identity shown:

³ <https://gist.github.com/aziele/6192a38862ce569fe1b9cbe377339fbe>

student 1 Post: [1a, 1b, 1c, 1a, 1d, 1e, 1a, 2b, 2c, 2d, 2e, 2a, 3b, 3d, 3c, 4a, 1a, 4b, 4c, 5a, 5b, 5c, 5d, 5e, 5f, 5g, -, 6b, 6c, 6d, 6e]

normative: [1a, 1b, 1c, -, 1d, 1e, 2a, 2b, 2c, 2d, 2e, -, -, 3a, 3c, 4a, -, 4b, 4c, 5a, 5b, 5c, 5d, -, 5f, 5g, 6a, 6b, 6c, 6d, 6e]

The maximum identity for this student sequence with \mathbb{L}_{norm} is 74%. The results for the other trials (optimal aligned sequences) are given in Appendix A to provide a detailed conformance analysis for each student trial. The same results also exist on a stage level and could be provided as well. This would lead to additional (6 stages * 2 rounds * 10 students * 2 (norminal and student)) 240 sequences listed, which we decided not to do.

5.2 Duration Analysis

In Figures 2 and 3 are the durations visualized for each student's pre and post trial for the whole process and each stage respectively. For the overall process it can be observed that all students except student 2 were able to perform the procedure faster after the individual training period. The mean duration over all students for pre and post trial are 17.7 min and 13.5 min respectively. For all stages except S_6 the majority of students increased their speed when comparing pre and post trials. The mean duration decreased for all stages from pre to post trial. The durations of 0 min in stage 3 can be explained by the fact that the activity $3a$ was often skipped and the other activities in stage 3 are logged with the same start and end timestamps.

5.3 Identity Analysis

Figures 4 and 5 show the maximum identity of each sequence in $\mathbb{L}_{student}$ and $\mathbb{L}_{students_i}$ with \mathbb{L}_{norm} and \mathbb{L}_{norms_i} respectively. For the whole process it can be observed that most students improved their performance in terms of measured conformance with the guidelines from pre to post trial. The mean overall improvement for the whole process is around 10% (Fig. 4). For all of the 6 stages an improvement in terms of computed identity can be seen in Figure 5. Considering the post trial, students seem to perform best in stage S_6 (mean identity = 87.2%) and worst in stage S_3 (mean identity = 32.5%).

5.4 Combined Identity and Duration Analysis

Figures 6 - 12 are showing the identity vs duration of the whole process and all of the 6 stages for all student sequences. An optimal performance would aim a point in the top left corner, indicating a large conformance with the guideline and quick execution time. For the whole process (Fig. 6) it is clearly observable that the students improved substantially from pre to post trial. The same statement can generally be made when considering the different stages (Fig. 7- 12) as well.

6 Conclusion and Future work

In this work a medical training process for installing a Central Venous Catheters with ultrasound was analyzed. Data was provided by the Conformance Checking Challenge 2019 (CCC19). The guideline for this process was modeled by a provided Petri net which was used to simulate a normative event log for the overall process and its stages (\mathbb{L}_{norm} and $\mathbb{L}_{norm_{S_i}}$). The maximum identity of each sequence of the student process instances ($\mathbb{L}_{student}$ and $\mathbb{L}_{student_{S_i}}$) with the normative event logs (\mathbb{L}_{norm} and $\mathbb{L}_{norm_{S_i}}$) were computed via the Needleman-Wunsch algorithm, which is commonly used for aligning protein and nucleotide sequences in bioinformatics. The identity values were used in combination with the execution times (durations) to provide both students and instructors with detailed and objective performance measures. This approach could easily be applied to other similar problems. As a next step it would be interesting to get feedback from students and instructors if they find the provided performance measures helpful. Furthermore, it would be interesting to compactly visualize the sequences in Appendix A. With that it might also be possible to provide the optimal aligned stage level sequences in a concise manner.

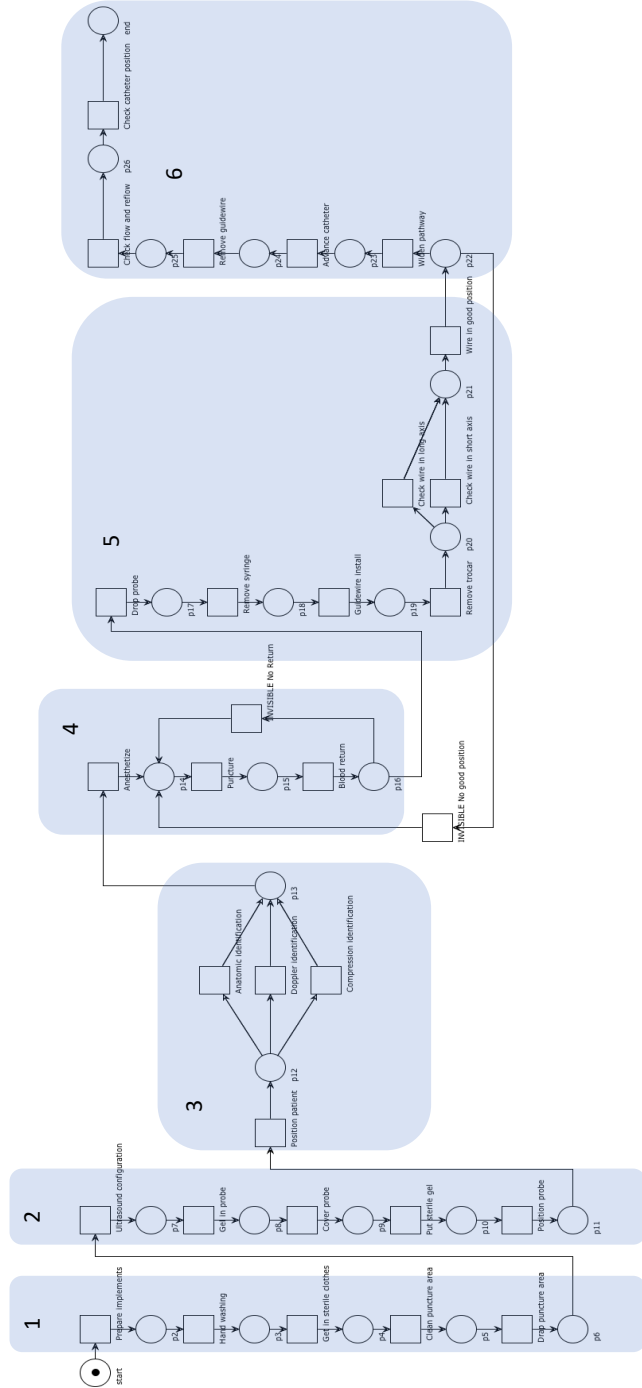


Fig. 1: Petri net representation of the normative model for the Central Venous Catheters (CVC) with ultrasound process. Activities are modeled as transitions and the initial marking includes one single token at the first place. The six subsequent stages of the process: 'Operator and Patient Preparation' (1), 'Ultrasound Preparation' (2), 'Locate Structures' (3), 'Venous puncture' (4), 'Install Guidewire' (5) and 'Install Catheter' (6) are highlighted.

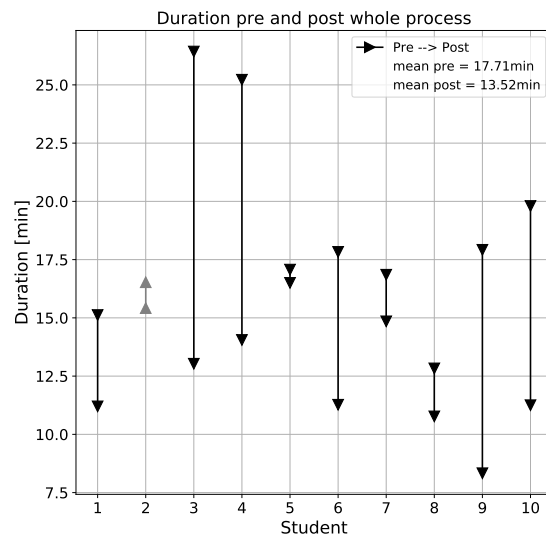


Fig. 2: Duration of the whole process for each student's pre and post trial. The markers are pointing from pre to post. Black indicates an improvement and gray a decline in performance from pre to post trial.

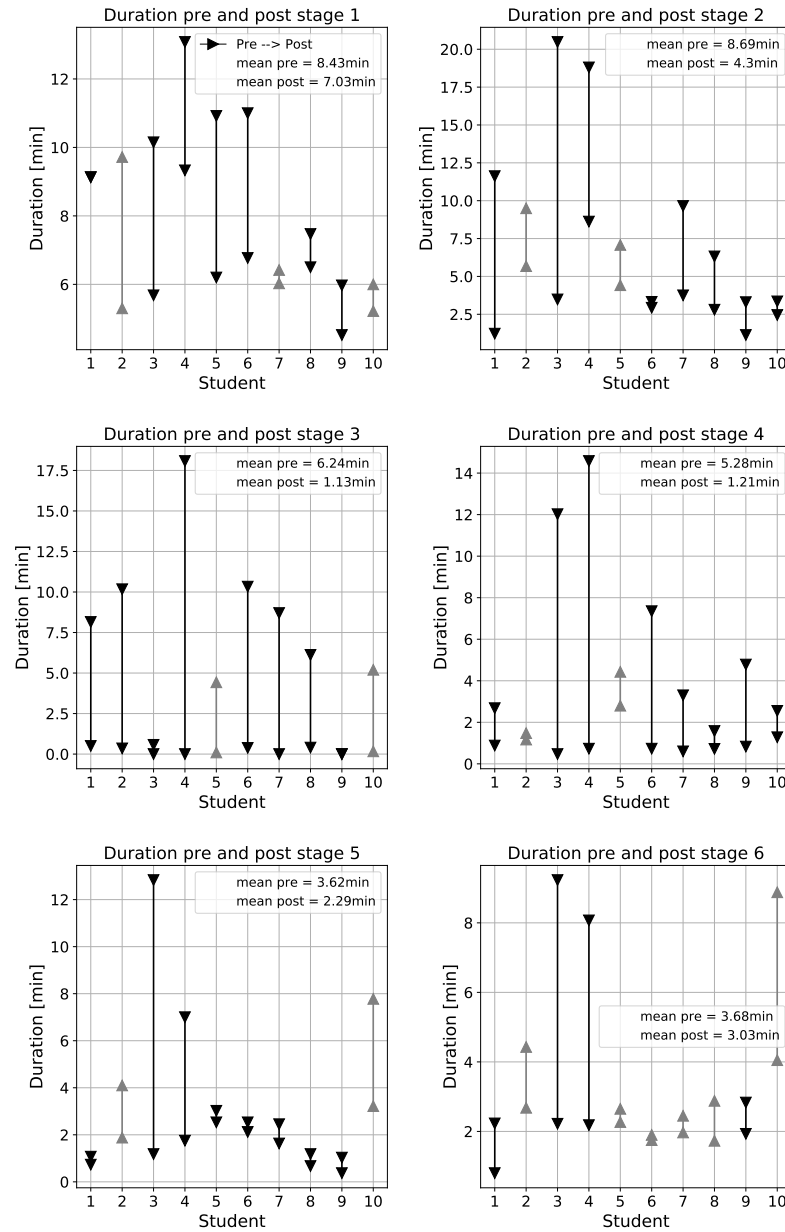


Fig. 3: Duration of all 6 process stages for each student's pre and post trial. The markers are pointing from pre to post. Black indicates an improvement and gray a decline in performance from from pre to post trial.

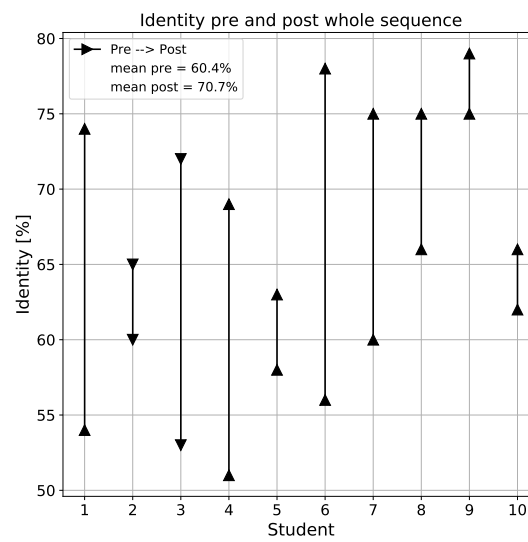


Fig. 4: Identity of the whole process for each student's pre and post trial. The markers are pointing from pre to post. Black indicates an improvement and gray a decline in performance from pre to post trial.

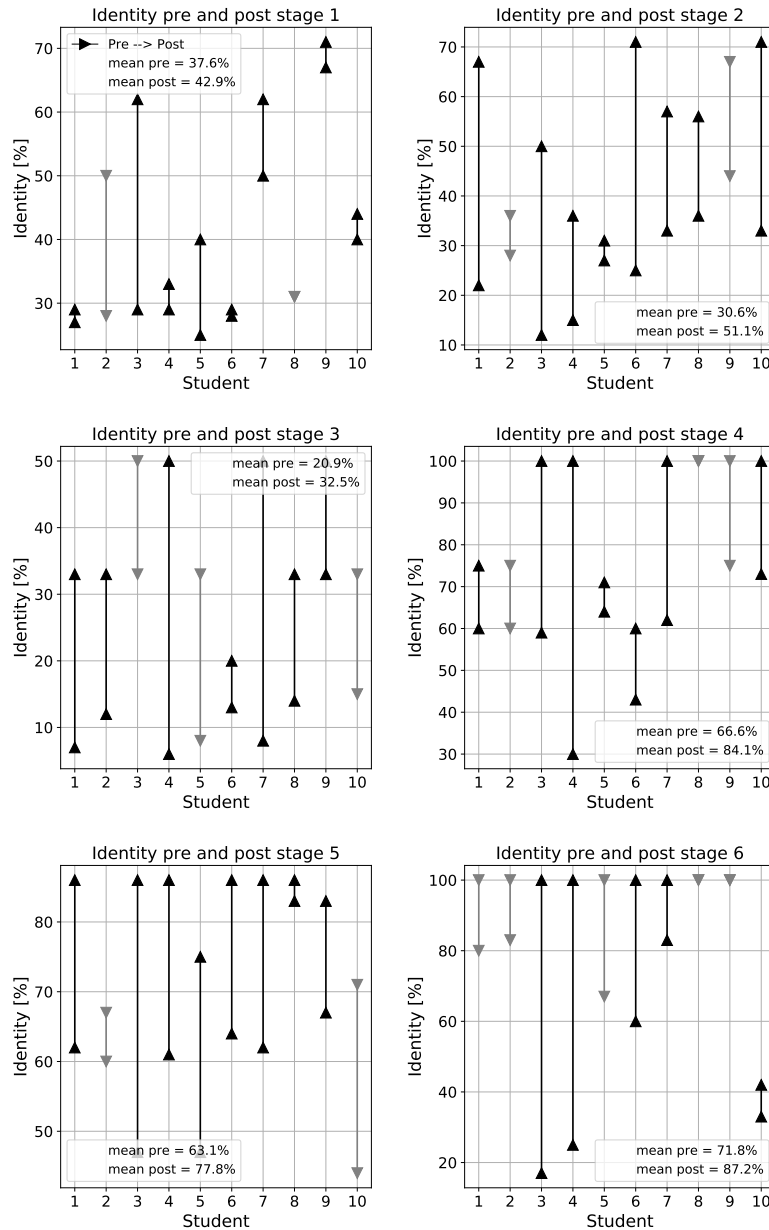


Fig. 5: Identity of all 6 process stages for each student's pre and post trial. The markers are pointing from pre to post. Black indicates an improvement and gray a decline in performance from pre to post trial.

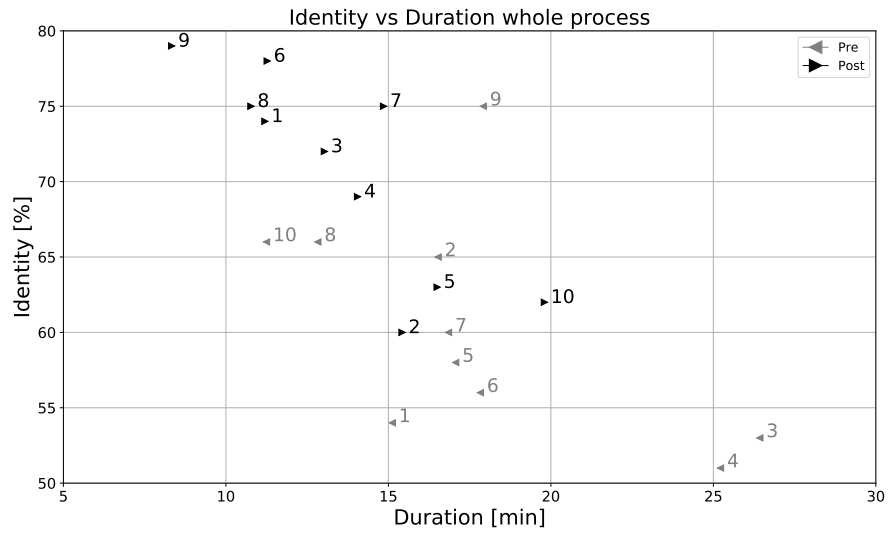


Fig. 6: Identity vs Duration for the students pre (gray) and post (black) trial.

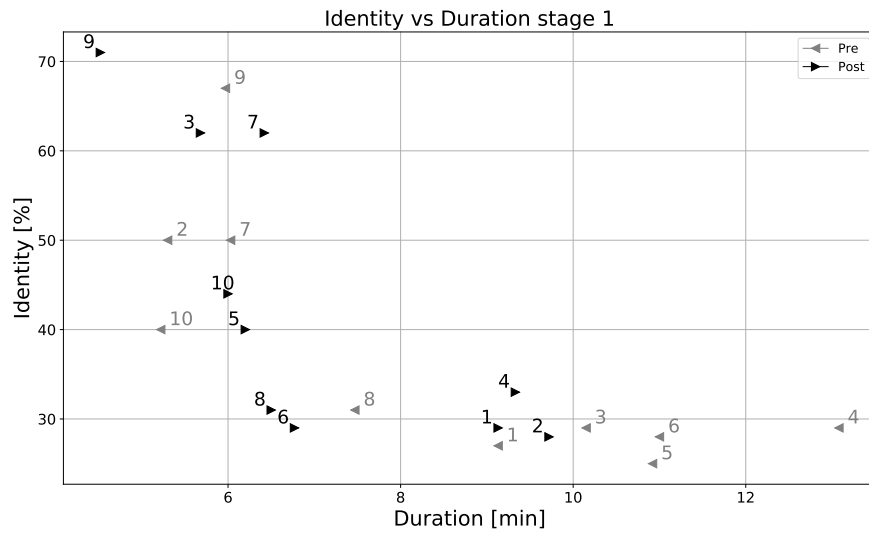


Fig. 7: Identity vs Duration for the students pre (gray) and post (black) trial.

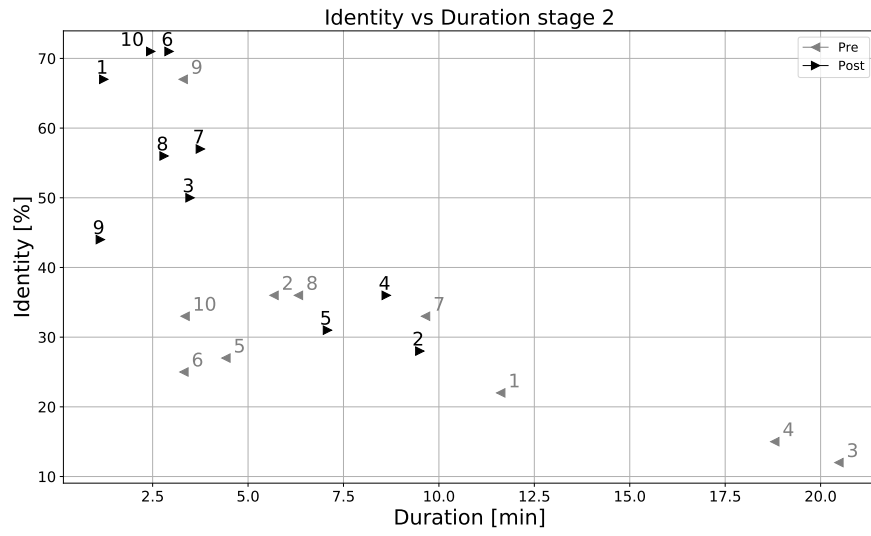


Fig. 8: Identity vs Duration for the students pre (gray) and post (black) trial.

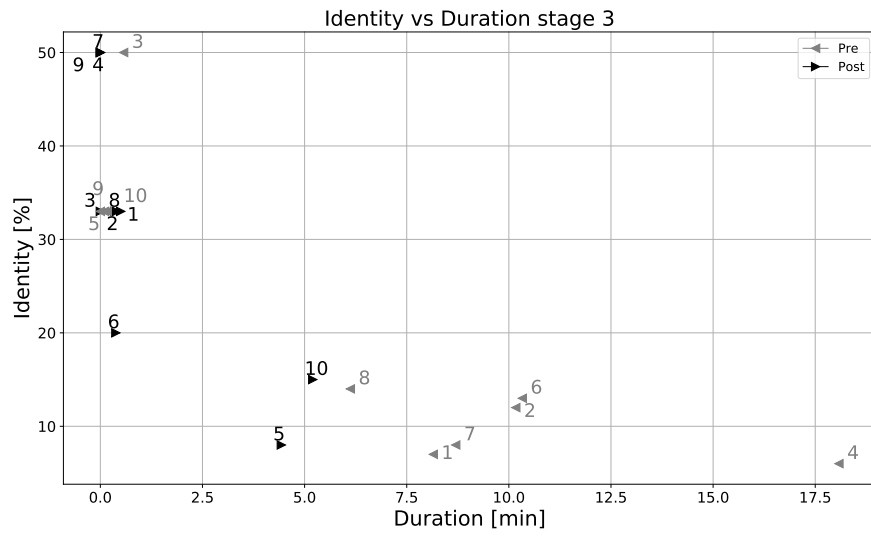


Fig. 9: Identity vs Duration for the students pre (gray) and post (black) trial.

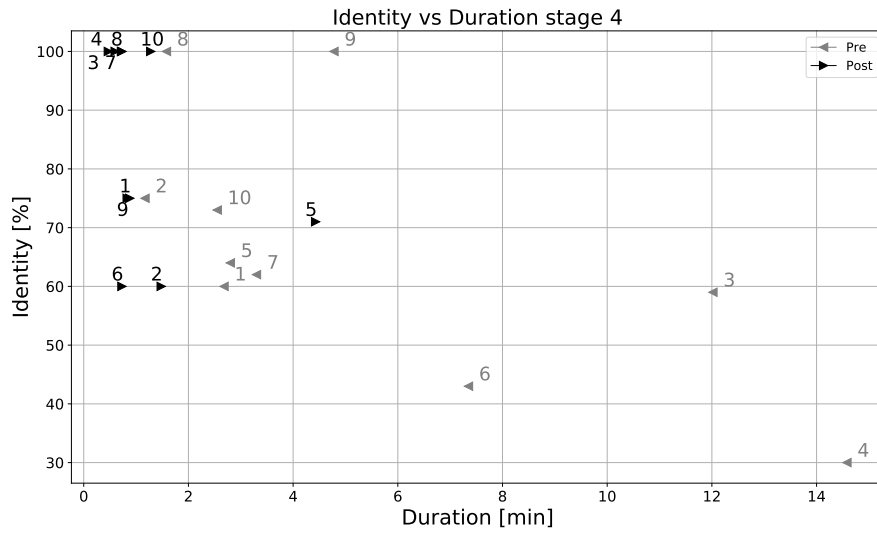


Fig. 10: Identity vs Duration for the students pre (gray) and post (black) trial.

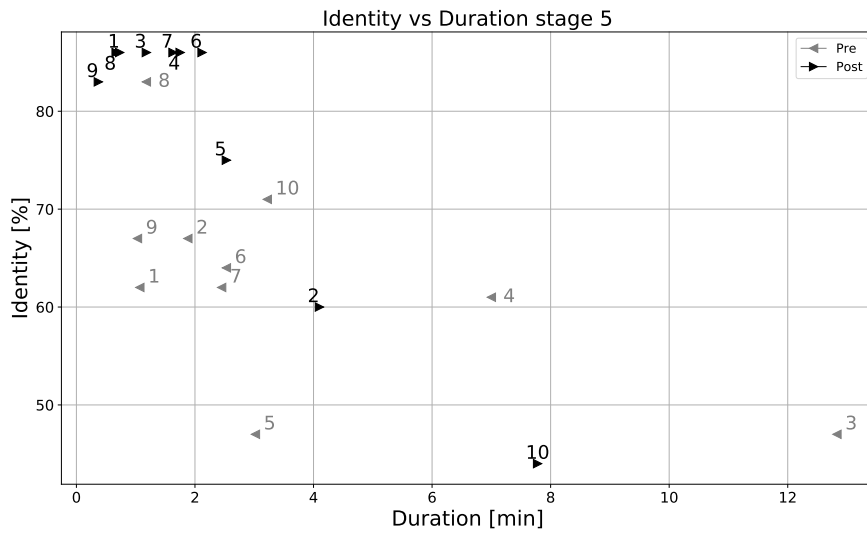


Fig. 11: Identity vs Duration for the students pre (gray) and post (black) trial.

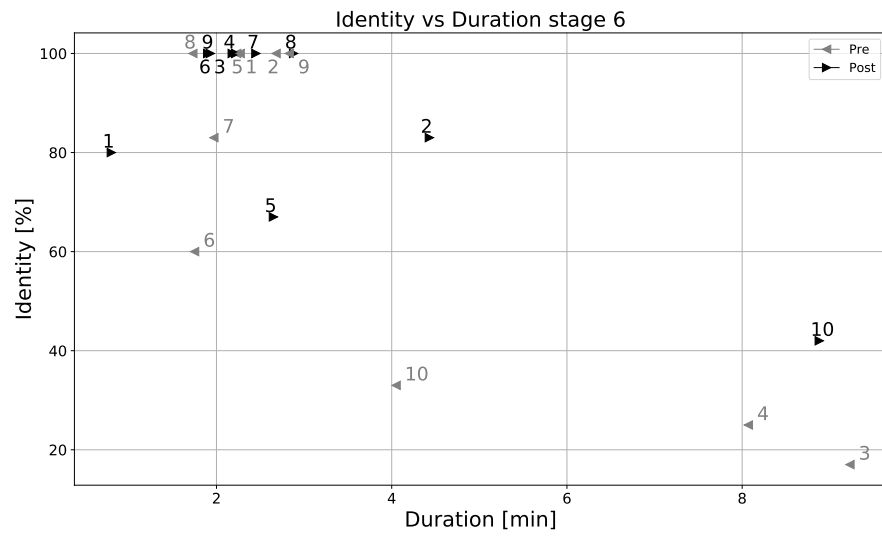


Fig. 12: Identity vs Duration for the students pre (gray) and post (black) trial.

References

1. Aalst, Wil van der. *Process Mining: Data Science in Action*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2016. <https://doi.org/10.1007/978-3-662-49851-4>.
2. Rojas, Eric, Jorge Munoz-Gama, Marcos Sepúlveda, and Daniel Capurro. "Process Mining in Healthcare: A Literature Review." *Journal of Biomedical Informatics* 61 (June 1, 2016): 224–36. <https://doi.org/10.1016/j.jbi.2016.04.007>.
3. Richard, Williams, Rojas Eric, Peek Niels, and Johnson Owen A. "Process Mining in Primary Care: A Literature Review." *Studies in Health Technology and Informatics*, 2018, 376–380. <https://doi.org/10.3233/978-1-61499-852-5-376>.
4. Munoz-Gama, J., de la Fuente, R., Sepúlveda, M., Fuentes, R., Conformance Checking Challenge 2019. 4TU.Centre for Research Data. <https://doi.org/10.4121/uuid:c923af09-ce93-44c3-ace0-c5508cf103ad>
5. Saul B. Needleman, Christian D. Wunsch, A general method applicable to the search for similarities in the amino acid sequence of two proteins, *Journal of Molecular Biology*, Volume 48, Issue 3, 1970, Pages 443-453, ISSN 0022-2836, [https://doi.org/10.1016/0022-2836\(70\)90057-4](https://doi.org/10.1016/0022-2836(70)90057-4).
6. Osamu Gotoh, An improved algorithm for matching biological sequences, *Journal of Molecular Biology*, Volume 162, Issue 3, 1982, Pages 705-708, ISSN 0022-2836, [https://doi.org/10.1016/0022-2836\(82\)90398-9](https://doi.org/10.1016/0022-2836(82)90398-9).

A Optimal Aligned Sequences

A.1 Pre Trial

student 1

student: [1b, 2a, 3b, 3d, 2b, 1c, 1b, 1c, 1d, 1e, -, -, 2c, 2d, 3b, 2a, 1a, 4a, 3b, 4b, 2a, 4c, 5a, 5b, 5c, 5e, 5f, 5d, 5e, 5f, 6a, 6b, 6c, 6d, 6e]
 normative: [-, -, -, -, 1a, 1b, 1c, 1d, 1e, 2a, 2b, 2c, 2d, 2e, 3a, 3b, 4a, -, 4b, -, 4c, 5a, 5b, 5c, -, -, 5d, 5e, 5g, 6a, 6b, 6c, 6d, 6e]

student 2

student: [1c, 1b, 3a, 1a, 1b, 1c, 1a, 1d, 1e, 1a, 2c, 2a, 2b, -, 2d, 2e, 2a, 3b, 3d, 4a, 2e, 4b, 4c, 5a, 5b, 5c, 5d, -, -, 6a, 6b, 6c, 6d, 6e]
 normative: [-, -, -, 1a, 1b, 1c, -, 1d, 1e, -, -, 2a, 2b, 2c, 2d, 2e, -, 3a, 3d, 4a, -, 4b, 4c, 5a, 5b, 5c, 5d, 5f, 5g, 6a, 6b, 6c, 6d, 6e]

student 3

student: [1a, 1c, 2a, 2b, 2a, 1b, 1c, 1a, 1d, 1e, -, 1a, 2c, 2d, 2e, 3b, 3d, 1a, 2e, 4a, 4b, 4c, 5a, 4b, 4c, 5a, -, 5c, -, -, 2e, 4b, 4c, 5a, 5b, 5c, 5d, 5e, 5f, 5g, 6a, 6b, 5e, 5f, 6c, 2e, 4b, 4c, 5a, 5b, 5c, 5e, 5f, 5d, 5f, 5e, 5f, 6b, 5e, 5f, 6b, 6c, 6d, -]
 normative: [1a, -, -, -, 1b, 1c, -, 1d, 1e, 2a, 2b, 2c, 2d, 2e, 3a, 3d, -, -, 4a, 4b, 4c, -, 4b, 4c, 5a, 5b, 5c, 5d, 5e, 5g, 4b, 4c, 5a, 5b, 5c, 5d, -, 5f, 5g, 4b, 4c, 4b, 4c, 4b, 4c, 4b, 4c, 5a, 5b, 5c, -, -, 5d, -, -, 5f, -, 5g, 6a, 6b, 6c, 6d, 6e]

student 4

student: [1a, 2a, 3a, 1b, 1c, 1a, 1d, 1a, 4a, 1e, 1a, 2b, 2c, 2d, 1a, 4a, 1a, 2d, 2e,

3b, 2a, 3b, 3d, 3c, 4b, 4c, 5a, 5b, 5c, 5d, 6a, 6b, 6c, 6d, 2e, 3b, 4b, 4c, 5a, 5b, 5c, 5d, 5e, 5f, 6a, 6b, -, -, -]

normative: [1a, -, -, 1b, 1c, -, 1d, -, -, 1e, 2a, 2b, 2c, -, -, -, 2d, 2e, -, -, 3a, 3d, 4a, 4b, 4c, 5a, 5b, 5c, 5d, -, -, -, 5f, 5g, 4b, 4c, 5a, 5b, 5c, 5d, 5e, 5g, 6a, 6b, 6c, 6d, 6e]

student 5

student: [1c, 1b, 1c, 1d, 1e, 1a, 2c, 2b, 1a, 2a, 2d, 2e, 3b, 2a, 3d, 1a, 2e, 4a, 4b, 4c, 5a, 5b, 5e, 5f, 5c, 4c, 5c, 5e, 5d, 5e, 5f, 6a, 6b, 6c, 6d, 6e]

normative: [1a, 1b, 1c, 1d, 1e, -, 2a, 2b, -, 2c, 2d, 2e, -, 3a, 3d, -, -, 4a, 4b, 4c, 5a, 5b, -, -, -, 5c, -, 5d, 5e, 5g, 6a, 6b, 6c, 6d, 6e]

student 6

student: [1a, 3a, 1b, 1a, 1c, 1d, 1a, 1e, 1a, 4a, 2c, 2b, 2d, 3b, 2a, 3d, 2d, 1a, 4b, 4c, 5a, 5b, 5c, -, -, 4c, 4b, 4c, 5a, 5b, 5c, 5e, 5f, 5d, 6a, 6b, 6c, -, -]

normative: [1a, -, 1b, -, 1c, 1d, -, 1e, 2a, 2b, 2c, -, 2d, 2e, 3a, 3d, -, 4a, 4b, 4c, 5a, 5b, 5c, 5d, 5e, 5g, 4b, 4c, 5a, 5b, 5c, 5d, 5f, 5g, 6a, 6b, 6c, 6d, 6e]

student 7

student: [2a, 3b, 3d, 1c, 1b, 1c, 1a, 1e, -, 1a, 2c, 2d, 2e, -, 3b, 3d, 4b, 4b, 4b, 4c, 5a, 5b, 5c, -, 5e, 5f, 4b, -, 5a, 5b, 5c, 5f, 5f, 5g, 5f, 5d, 6b, 6a, 6b, 6c, 6d, 6e]

normative: [-, -, -, 1a, 1b, 1c, 1d, 1e, 2a, 2b, 2c, 2d, 2e, 3a, 3b, 4a, 4b, 4c, 4b, 4c, 5a, 5b, 5c, 5d, 5e, 5g, 4b, 4c, 5a, 5b, 5c, 5d, 5f, 5g, -, -, -, 6a, 6b, 6c, 6d, 6e]

student 8

student: [3a, 2a, 2b, 1c, 1b, 1c, 1a, 1d, 1e, -, -, 2c, 2d, 2e, 3b, 3d, 2a, 1a, 4a, 4b, 4c, 5a, 5b, 5c, 5d, 5e, 5f, 6a, 6b, 6c, 6d, 6e]

normative: [-, -, -, 1a, 1b, 1c, -, 1d, 1e, 2a, 2b, 2c, 2d, 2e, 3a, 3d, -, -, 4a, 4b, 4c, 5a, 5b, 5c, 5d, 5e, 5g, 6a, 6b, 6c, 6d, 6e]

student 9

student: [1c, 1b, 1c, 1d, 1e, 1a, 2b, 2c, 2a, 2d, 2e, 3b, 3d, 3c, 4a, 4b, 4c, 5a, 5b, 5c, 5d, -, -, 6a, 6b, 6c, 6d, 6e]

normative: [1a, 1b, 1c, 1d, 1e, 2a, 2b, 2c, -, 2d, 2e, -, 3a, 3c, 4a, 4b, 4c, 5a, 5b, 5c, 5d, 5f, 5g, 6a, 6b, 6c, 6d, 6e]

student 10

student: [1c, 1b, 1c, 1a, 1d, 1e, 2c, 2b, -, 2d, 1a, 2a, 2e, 3b, 2a, 3d, -, 4b, 4c, 5a, 5b, 5c, 5d, 6a, 6b, 4b, 4c, 5a, 5b, 5c, 5e, 5f, 5d, 6a, 6b, 6c, 6d, 6e]

normative: [1a, 1b, 1c, -, 1d, 1e, 2a, 2b, 2c, 2d, -, -, 2e, -, 3a, 3d, 4a, 4b, 4c, 5a, 5b, 5c, 5d, 5e, 5g, 4b, 4c, 5a, 5b, 5c, 5d, 5f, 5g, 6a, 6b, 6c, 6d, 6e]

A.2 Post Trial

student 1

student: [1a, 1b, 1c, 1a, 1d, 1e, 1a, 2b, 2c, 2d, 2e, 2a, 3b, 3d, 3c, 4a, 1a, 4b, 4c, 5a, 5b, 5c, 5d, 5e, 5f, 5g, -, 6b, 6c, 6d, 6e]

normative: [1a, 1b, 1c, -, 1d, 1e, 2a, 2b, 2c, 2d, 2e, -, -, 3a, 3c, 4a, -, 4b, 4c, 5a, 5b, 5c, 5d, -, 5f, 5g, 6a, 6b, 6c, 6d, 6e]

student 2

student: [1a, 2a, 3b, 3d, 3c, 1c, 1b, 1c, 1a, 1d, 1e, 1a, 2b, 2c, 1a, 2d, -, 4a, 1a, 2e, 4b, 4c, 5a, 5b, 5c, 5d, 5e, 5f, 5g, 6a, 6b, 5f, 6c, 6d, 6e]

normative: [1a, -, -, -, -, 1b, 1c, -, 1d, 1e, 2a, 2b, 2c, -, 2d, 2e, 3a, 3c, 4a, 4b, 4c, 5a, 5b, 5c, 5d, -, 5f, 5g, 6a, 6b, -, 6c, 6d, 6e]

student 3

student: [1c, 1a, 1b, 1c, 1a, 1d, 1e, -, 1a, 2c, 2b, 2d, 2e, 2a, 3d, 3c, 3b, 4a, 4b, 4c, 5a, 5b, 5c, 5d, 5e, 5f, 5g, 6a, 6b, 6c, 6d, 6e]

normative: [-, 1a, 1b, 1c, -, 1d, 1e, 2a, 2b, 2c, -, 2d, 2e, -, 3a, 3c, -, 4a, 4b, 4c, 5a, 5b, 5c, 5d, -, 5f, 5g, 6a, 6b, 6c, 6d, 6e]

student 4

student: [1c, 1a, 2a, 3b, 2b, 1b, 1c, 1a, 1d, 1e, -, 1a, 2c, 2d, 2e, 1a, 2e, 4a, 4b, 4c, 5a, 5b, 5c, 5d, 5e, 5f, 5g, 6a, 6b, 6c, 6d, 6e]

normative: [-, 1a, -, -, -, 1b, 1c, -, 1d, 1e, 2a, 2b, 2c, 2d, 2e, 3a, 3c, 4a, 4b, 4c, 5a, 5b, 5c, 5d, -, 5f, 5g, 6a, 6b, 6c, 6d, 6e]

student 5

student: [1a, 1c, 2a, 1a, 1c, 1b, 1a, 1d, 1a, 1e, 2c, 2b, -, 2d, 2e, 2a, 3b, 4a, 3d, 4b, 4c, 5a, 5b, 5c, 5d, 5e, 5f, 3d, 3b, 4b, 4c, 5a, 5b, 5c, 5d, 5e, 5f, 6a, 6b, 6c, -, 6e, 6d]

normative: [-, -, -, 1a, -, 1b, 1c, 1d, -, 1e, 2a, 2b, 2c, 2d, 2e, 3a, 3b, 4a, -, 4b, 4c, 5a, 5b, 5c, 5d, 5e, -, -, 5g, 4b, 4c, 5a, 5b, 5c, 5d, 5e, 5g, 6a, 6b, 6c, 6d, 6e, -]

student 6

student: [1a, 1c, 1b, 1c, 1a, 1d, 1e, 2a, 2b, 2c, 1a, 2d, 2e, 2a, 3d, 4a, 1a, 3a, 4b, 4c, 5a, 5b, 5c, 5d, 5e, 5f, 5g, 6a, 6b, 6c, 6d, 6e]

normative: [1a, -, 1b, 1c, -, 1d, 1e, 2a, 2b, 2c, -, 2d, 2e, 3a, 3d, 4a, -, -, 4b, 4c, 5a, 5b, 5c, 5d, -, 5f, 5g, 6a, 6b, 6c, 6d, 6e]

student 7

student: [1a, 1c, 1b, 1c, 1a, 1d, 1e, 1a, 2a, -, 2c, 2b, 2d, 2e, 2a, 3b, 3d, 4a, 4b, 4c, 5a, 5b, 5c, 5d, 5e, 5f, 5g, 6a, 6b, 6c, 6d, 6e]

normative: [1a, -, 1b, 1c, -, 1d, 1e, -, 2a, 2b, 2c, -, 2d, 2e, -, 3a, 3d, 4a, 4b, 4c, 5a, 5b, 5c, 5d, -, 5f, 5g, 6a, 6b, 6c, 6d, 6e]

student 8

student: [1a, 1c, 1b, 1c, 1a, 2a, 1d, 1e, -, 2b, 2c, 2d, 2e, 3b, 2a, 3d, 1a, 4a, 4b, 4c, 5a, 5b, 5c, 5d, 5e, 5f, 5g, 6a, 6b, 6c, 6d, 6e]

normative: [1a, -, 1b, 1c, -, -, 1d, 1e, 2a, 2b, 2c, 2d, 2e, -, 3a, 3d, -, 4a, 4b, 4c, 5a, 5b, 5c, 5d, -, 5f, 5g, 6a, 6b, 6c, 6d, 6e]

student 9

student: [1a, 1c, 1b, 1c, 1a, 1d, 1e, -, 2b, 2c, 2d, 2e, 3d, 3b, 4a, 2a, 4b, 4c, 5a, 5b, 5c, 5d, 5f, -, 6a, 6b, 6c, 6d, 6e]

normative: [1a, -, 1b, 1c, -, 1d, 1e, 2a, 2b, 2c, 2d, 2e, 3a, 3b, 4a, -, 4b, 4c, 5a, 5b, 5c, 5d, 5f, 5g, 6a, 6b, 6c, 6d, 6e]

student 10

student: [1c, 1b, 1c, 3a, 1d, 1e, 1a, 2a, 1a, 2b, 2c, 2d, 2e, 2a, 3b, 3d, 4a, 4b, 4c, 5a, 5b, 5c, 5d, 5e, 5f, 5e, 5g, 6a, 6b, 6b, 6b, 6b, 5e, 5f, 6b, 6b, 6c, 6d, 6e]

normative: [1a, 1b, 1c, -, 1d, 1e, -, 2a, -, 2b, 2c, 2d, 2e, -, 3a, 3d, 4a, 4b, 4c, 5a, 5b, 5c, 5d, -, 5f, -, 5g, 6a, -, -, -, -, -, -, 6b, 6c, 6d, 6e]