

# VIT-PLA: Visual Interactive Tool for Process Log Analysis

Sen Yang<sup>1</sup>, Xin Dong<sup>1</sup>, Moliang Zhou<sup>1</sup>, Xinyu Li<sup>1</sup>, Shuhong Chen<sup>1</sup>, Rachel Webman<sup>3</sup>,  
Aleksandra Sarcevic<sup>2</sup>, Ivan Marsic<sup>1</sup> and Randall S. Burd<sup>3</sup>

<sup>1</sup>Rutgers University  
Piscataway, NJ, USA  
{sy358, xd48, mz330, xl264,  
sc1624, marsic}@rutgers.edu

<sup>2</sup>Drexel University  
Philadelphia, PA, USA  
aleksarc@drexel.edu

<sup>3</sup>Children's National Medical Center  
Washington, DC, USA  
{rwebman, rburd}  
@childrensnational.org

## ABSTRACT

Techniques for analyzing and visualizing process or workflow data have been developed and applied in a wide range of domains. Visual analysis of large process logs and integration of statistical analysis, however, have been limited. We introduce the Visual Interactive Tool for Process Log Analysis (VIT-PLA) that provides a simplified process log visualization and performs statistical correlation analysis on process attributes. We demonstrate its use by applying it to an artificial dataset and running a preliminary analysis of trauma team task data collected from a medical emergency department.

## Keywords

Interactive Workflow Data Visualization; Trace Alignment; Trace Clustering; Correlation Analysis

## 1. INTRODUCTION

### 1.1 Motivation

Many contemporary information systems record activity logs, including personal calendars and electronic health records (EHR). Process mining techniques attempt to extract non-trivial knowledge and insights from these activity logs and use them for further analyses [1]. Most research in process mining has focused on workflow discovery and process execution visualization [1][2]. When visualized, real-world workflow often produces “spaghetti-like” graphics that are difficult to analyze and do not provide useful observations or insights. In addition to graphical visualization, other efforts have also been made to produce different visualizations for process executions or workflow data [3][4][5][6][7][8][9]. Although these systems have been shown to work well with focused processes and relatively small event logs, little work has been done with large process logs with many execution traces (typically hundreds or thousands of different process cases). Simply displaying all traces at once does not make a useful visualization. We observed that only several dozen traces can fit intelligibly on one screen at a time. Even if the symbols were distinguishable, the amount of displayed data make it inconvenient for human interpretation. When working with large workflow

datasets, it is often useful to obtain a concise visualization that summarizes the data into an easily interpretable format. We present an approach for visualizing a summary of large process logs by aggregating the data with a trace clustering method. Process traces are clustered based on the similarity or proximity between their elements (i.e. process tasks). Each cluster is represented using a “representative” or “average” trace extracted from the corresponding cluster. Using this approach, we are able to usefully visualize large process logs. To help users better understand the clusters, we also included tools for running statistical tests on the clusters and their associated process attributes. These statistical test results can reveal significant and interesting correlations between process executions and process attributes. We implemented these approaches in a Java-based application, named VIT-PLA.

### 1.2 Related Work

Recent advances have been made in the development of workflow data visualization techniques. EventFlow [3] visualizes temporal events on a timeline and can simplify workflow executions into an aggregated display. Outflow [5] aggregates events into a graph with integrated statistics. Frequency [6] and Care Pathway Explorer [7] are user interfaces for information exploration that integrate interactive visualizations with data mining to find frequent event sequence patterns. Dotted Chart [8] uses colored dots to visualize process traces in a fast and simple implementation. The trace alignment plugin for the ProM framework [9] is designed to align process traces so as to optimize interpretability and facilitate exploration. Despite extensive work on interactive visualization, little has been done to directly integrate statistical analysis into these applications. Some data visualization applications can show general statistics [5][8], but few can provide more sophisticated ones [4]. CoCo [4] can be used to find similarities and differences between two groups (“cohorts”) of process traces and to highlight their significant distinguishing features (e.g. activity order, frequency, and duration).

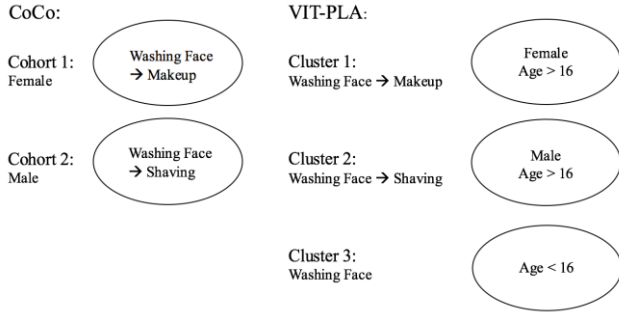
From the perspective of workflow visualization, Eventflow [3] and ProM's Trace Alignment [9] plugin are closest related to our work. Neither are suitable for visualizing large process logs with many traces, because both visualize all activities in the log at once. Without data aggregation and summarization strategies, the size of the dataset that can be handled is always limited. From our previous experience with Eventflow and ProM, visualizations using a standard sized computer monitor (24") generally become uninterpretable when the number of unique process traces exceeds 100. EventFlow can be used to visualize logs with >100 process traces, but only if there are many repeated traces [21]. Eventflow visualizes the activities on a timeline without advanced processing of the data. ProM visualizes the alignment and also clusters the

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

*KDD 2016 Workshop on Interactive Data Exploration and Analytics (IDEA'16)*, August 14th, 2016, San Francisco, CA, USA.

Copyright is held by the owner/author(s).

DOI: <http://dx.doi.org/10.1145/12345.67890>



**Figure 1. A simple example showing the differences between the statistical analysis in CoCo and VIT-PLA. This example describes a morning skincare ritual. The workflow includes three different activities (washing face, makeup, and shaving) and two different attributes (gender and age).**

process traces, but does not provide any statistical analyses that can help the user better understand their data. When visualizing clusters of process traces, ProM shows all traces in each cluster without any data aggregation or simplification. In contrast, our approach displays each cluster’s cluster “prototype” [8], i.e., an execution trace that is representative of the other traces in the cluster (the representative trace is not necessarily one of the original process traces in the input log). This strategy enables visualization of large process logs. This visualization also helps to identify key characteristics of each cluster and key differences between clusters.

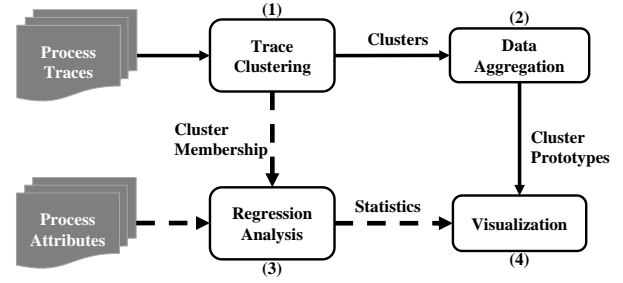
From the perspective of statistical analysis, CoCo is closest related to our work. Both CoCo and VIT-PLA seek to correlate trace structural features (e.g., sequential order of activities, their frequencies and durations) with process attributes (e.g., patient gender, age, etc.). The two approaches to statistical analysis are different (Figure 1). CoCo first splits the data into strictly two cohorts based on a background attribute (in this case gender). It then finds significant associations between the cohorts’ trace structures and attributes. It may identify a structural pattern (e.g., “Washing Face → Makeup”) as significantly belonging to one cohort (female), as opposed to the opposite (male). In contrast, our implementation first separates the data into clusters based on trace structure, and then associates cluster membership with background attributes. For example, the sequence “Washing Face → Makeup” is executed mostly by females over age 16.

Unlike CoCo that can only make these associations based on cohort pairs, our system uses multinomial or binomial logistic regression to make associations based on multiple clusters. VIT-PLA allows for more comprehensive attribute-structure correlation, bringing the previously unusable age attribute into the analysis (see example above [Figure 1]). In this way, VIT-PLA’s approach reveals potential relationships missed by CoCo’s binary analysis.

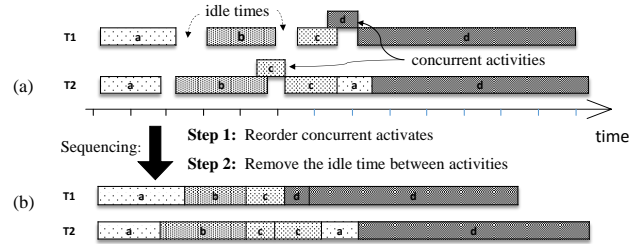
Our statistical analysis is important because it facilitates the discovery of significant correlations between clusters and background attributes. Given the trace attributes, we may determine what workflow practices (represented by the cluster prototype) are more likely to be observed, which is useful information for analyzing the workflow data and extracting insights.

### 1.3 Contribution

Our main contribution is a novel approach to producing summarized visualizations of large process logs and directly integrating statistical analyses into the visualization. These features help users discover attributes associated with specific sequence progressions and deviations within the dataset.



**Figure 2. Flowchart outlining the core methods implemented in VIT-PLA and their corresponding inputs and outputs.**



**Figure 3. Two steps of sequencing the traces with concurrent activities (such as  $d$  in  $T_1$  and  $c$  in  $T_2$ ) and idle times (white spaces between activities). (a) Example process traces before sequencing. (b) The same process traces after sequencing.**

The paper is organized as follows. Section 2 introduces our approach to process trace visualization and attribute analysis. Section 3 discusses our implementation and user interface design. Section 4 shows preliminary results from using VIT-PLA on an artificial dataset and a trauma resuscitation process log. Section 5 summarizes the paper and discusses the limitations of our current work.

## 2. METHODOLOGY

The core methods implemented in VIT-PLA can be summarized as follows (Figure 2): (1) clustering of process traces (workflow data) based on proximity of data objects, (2) aggregation of process traces and selection of cluster prototype, (3) regression analysis to explore underlying knowledge, (4) interactive visualization of process traces and statistical analysis results. This section will describe (1), (2), and (3); (4) will be discussed in Section 3.

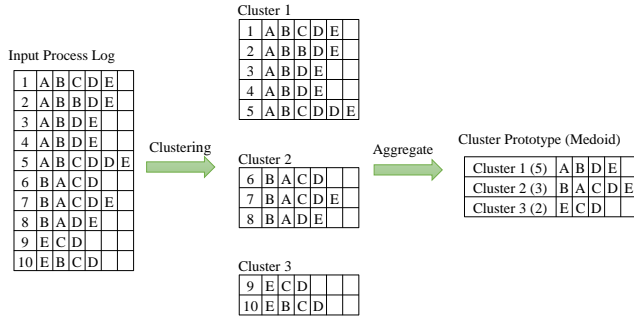
### 2.1 Data Preprocessing: Sequencing of Traces

Process sequencing is necessary before more advanced processing. Activities coded in a process log usually have start and end timestamps (some logs may not include end time) for each activity. Idle time may exist between activities, and some activities may be executed concurrently (Figure 3(a)). In process mining, process traces are usually sequenced by ascending order of the start time of activities (Figure 3(b)).

### 2.2 Summary Visualization of Process Logs

#### 2.2.1 Process Trace Clustering

Our approach uses clustering techniques to simplify the process trace visualizations. Clustering provides an abstraction from the original data objects to generalized data representatives, i.e. cluster prototypes. In most data mining problems, data clusters are calculated based on the data objects’ feature set. However, to aggregate process traces that follow an underlying workflow model, we cluster the traces based on the similarity of their



**Figure 4.** An example showing data clustering and aggregation. The cluster prototype used here is cluster medoid.

constituent tasks in terms of task type and sequential order of execution [10]. That is to say, our sole feature used for clustering is the structure of each trace’s task sequence, not the process attributes.

In VIT-PLA, the clustering algorithm we use is agglomerative hierarchical clustering [15] with Ward’s method [22] as clustering criterion. We calculate the similarity of process traces based on Edit Distance [8] (a.k.a. Levenshtein Distance [11]). If activity duration information is also available, the similarity can be calculated with “Duration-Aware Edit Distance” [16], a metric derived from Edit Distance that penalizes dissimilarity between durations of the same activity type.

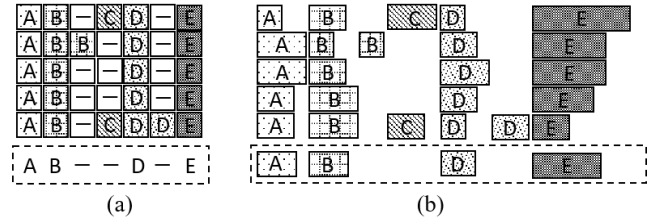
### 2.2.2 Cluster Prototype and Trace Alignment

After clustering, each cluster can be characterized by a cluster prototype (Figure 4). Because it is not practical to visualize all the data objects on a single computer screen, a substantial reduction in the data size is needed. The deployment of cluster prototypes helps compress the dataset.

Several candidates can be considered as cluster prototype, such as the widely-used cluster centroid [14], the center of a cluster. There is, however, a great chance that there may not be an actual data point at the cluster’s center. In this case, the centroid location is calculated from the data in the cluster with the aim of minimizing the sum-squared distance to other points.

Note that for categorical data and event-based data, the notion of a center (centroid) does not apply [14]. For example, the centroid of categorical data (e.g. {orange, apple, banana}) cannot be determined. In this case, we may use the cluster medoid, the most representative data object in the cluster, i.e. a data point with minimal average dissimilarity to all other objects in the cluster. The medoid, however, may not be adequate if the cluster does not contain an “appropriate” representative.

To ensure that the chosen sequence is representative of the cluster, we used the consensus sequence as the cluster prototype even though it may not be an observed trace from the data. The consensus sequence, a concept derived from aligning biological sequences (e.g. DNA) in bioinformatics, is a sequence of the most frequent residues found in the alignment matrix’s columns. In process mining, consensus sequences may be considered the “average” or “common” sequence of tasks [9] (Figure 5). To find the consensus sequence for each cluster, trace alignment [9][16] needs to be performed using traces from each cluster respectively. Trace alignment reformats the original data by placing the same or similar activities of all traces to the same column of the alignment matrix. If a matching activity cannot be found, a gap symbol “-” is inserted. Bose and Van der Aalst [9] have shown how to use trace alignment techniques to visualize and analyze process traces



**Figure 5.** An example of two types of trace alignment: (a) Context-Aware and (b) Duration-Aware. The sequences at the bottom of (a) and (b) are consensus sequences derived from the data. A gap symbol “-” or white space is inserted if a match cannot be found. The five process traces shown here are from Cluster 1 in Figure 4.

(Figure 5(a)). In our previous work, we extended their work by introducing a duration-aware trace alignment algorithm [16] that also takes activity duration into consideration. In our implementation, the alignment algorithm can work for data either with or without activity durations (Figure 5).

## 2.3 Association between Trace Clusters and Trace Attributes

In addition to visualization, VIT-PLA also provides statistical analysis functions. The goal of our statistical analyses is to help the user discover the underlying associations between data cluster membership and trace attributes. This goal is accomplished using either multinomial or binary logistic regression. The user chooses between these two statistical methods depending on the domain question being asked. Multinomial logistic regression works for binary comparison between two clusters (one-vs.-one cluster comparison), while binomial logistic regression works for binary comparison between one cluster and the rest of the clusters (one-vs.-rest). Using both logistic regression models can help discover attributes associated with particular clusters.

### 2.3.1 Multinomial logistic regression

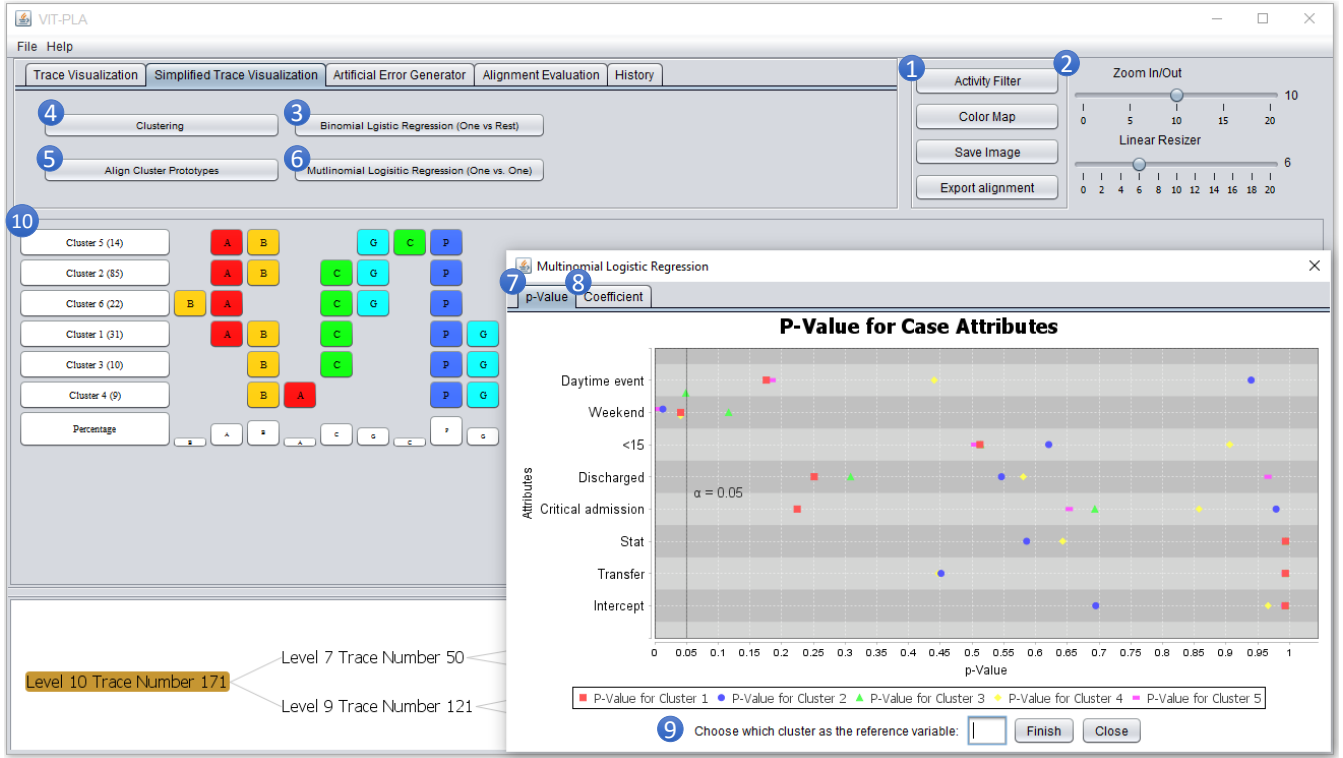
In multinomial logistic regression [12], let  $K$  denote the number of independent variables, and let  $J$  denote the number of discrete categories of the dependent variable, where  $J \geq 2$ . In our case, the independent variables correspond to the trace attributes and the dependent variables correspond to the trace cluster membership. The number of trace attributes is  $K$  and the number of clusters is  $J$ . By default, we define the last category (the  $J$ th cluster) to be the reference category, against which logits of the first  $J-1$  categories are compared. Let  $C$  denote cluster membership. Represented formally:

$$\ln \left( \frac{P(C=i)}{P(C=J)} \right) = \ln \left( \frac{P(C=i)}{1 - \sum_{j=1}^{J-1} P(C=j)} \right) = \beta_{i0} + \beta_{i1}x_{i1} + \beta_{i2}x_{i2} + \dots + \beta_{iK}x_{iK}, \quad i = 1, \dots, K-1 \quad (1)$$

where  $x_i$  are trace attributes, and  $\beta_i$  are regression coefficients for each of the trace attributes. In VIT-PLA, users can also choose which cluster to use as the reference category.

### 2.3.2 Binomial logistic regression

Binary logistic regression [12] is a special case of multinomial logistic regression, in which there are only two categories ( $J = 2$ ). In our problem, one category is the target cluster of interest and the other category is all other clusters. Let  $K$  denote the total number of independent variables and  $C$  denote cluster membership. Represented formally:



**Figure 6. VIT-PLA Graphical User Interface showing aggregated data, hierarchical clustering results, and statistics from the multinomial logistic regression analysis. The data shown here is the same as the data in our 2<sup>nd</sup> case study. Please note that there are other functions of VIT-PLA that are not displayed in this figure.**

$$\ln\left(\frac{P(C=i)}{P(C \neq i)}\right) = \ln\left(\frac{P(C=i)}{1-P(C=i)}\right) = \beta_{i0} + \beta_{i1}x_{i1} + \beta_{i2}x_{i2} + \dots + \beta_{iK}x_{iK}, \quad i = 1, \dots, K \quad (2)$$

where the parameters have the same meaning as in Eq.1.

### 2.3.3 Hypothesis Test

To identify which trace attributes are significantly associated with cluster membership, we use the Wald test [13] for logistic regression, which is defined as:

$$W = \frac{(\hat{\beta}_i - \beta_i)}{\text{se}(\hat{\beta}_i)}$$

where  $\hat{\beta}_i$  is the regression coefficient for trace attributes  $x_i$ ;  $\beta_i = 0$  is the null hypothesis, i.e. the trace attribute  $x_i$  has a corresponding coefficient of zero;  $se$  is standard error. In our implementation, we use a normal distribution and  $z$ -values for calculating  $p$ -values. The null hypothesis can be rejected when  $p$ -value is less than or equal to alpha, the significance level which is most often set at 0.05.

## 3. VISUAL INTERFACE DESIGN

During software development, we received feedback from domain experts and continuously improved our design. In this section, we describe the first prototype of VIT-PLA. The visual interface design (Figure 6) was developed with three main goals:

- G1. Interactive visualization of raw process traces, the basic visualization functionality.
- G2. Simplified visualization of process traces (for large data applications).
- G3. Visualization of trace cluster vs. trace attribute association statistics.

Although VIT-PLA has many other functions, the rest of this paper focuses on how its design achieves these three goals.

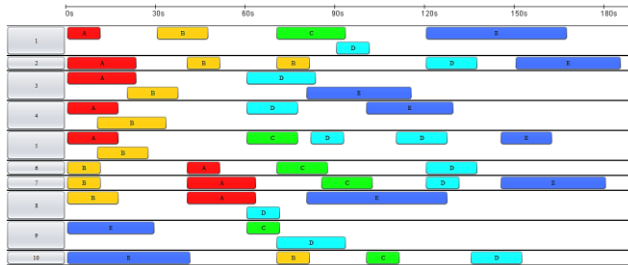
### 3.1 G1: Three Common Ways to Visualize Raw Process Traces

VIT-PLA provides three common ways of visualizing raw process traces. We refer to the data as “raw process traces” to distinguish goal G1 from G2, where the data is visualized in an aggregated format. The three visualization methods are:

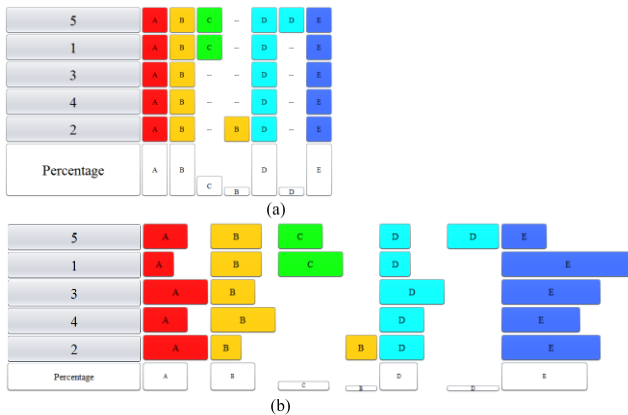
- 1) *Simple stack of activities in the process traces (Error! Reference source not found.*(a) without activity duration, and **Error! Reference source not found.**(b) with activity duration). This approach is one of the simplest ways to visualize process traces. Activities are stacked based on their occurrence time. Activity information can be accessed with a mouse click on the corresponding symbol. This visualization is easily interpretable and computationally efficient, but it cannot provide deep insights into the data.
- 2) *Overlay of the process execution on the timeline (Error! Reference source not found.*). Activities are scaled based on duration and aligned to the timeline according to their start and end times. The advantage of this visualization approach is that it clearly shows the concurrent activities in each process.
- 3) *Process trace alignment (Error! Reference source not found.*(a) context-aware alignment and **Error! Reference source not found.**(b) duration-aware alignment). The context-aware trace alignment algorithm is based on Bose and Van der Aalst’s work [9] and the duration-aware trace alignment algorithm proposed in our previous research [16]. The duration of each activity in the consensus sequence (bottom line of **Error! Reference source not found.**(b)) of duration-



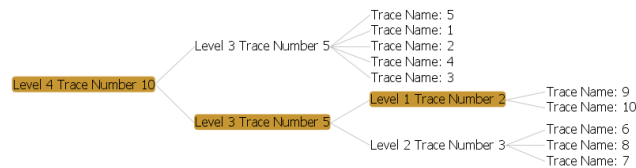
**Figure 7. Simple stack (a) Process executions are stacked (b) Process executions are stacked and symbol blocks are scaled based on activity duration. Each row represents a single trace and each block represents a single activity. The data comes from Cluster 1 in Figure 4.**



**Figure 8. Visualize process traces on a timeline. The top scale is the timeline with second as the unit. Each row, separated by a bold line, represents a single process. Each block represents a single activity. Symbol blocks that are vertically stacked in one process are activities occurring simultaneously. The data comes from the input log in Figure 4.**



**Figure 9. Alignment (a) Process trace alignment (b) Duration-aware trace alignment. Each row represents a single process and each block represents an activity. The bottom line of each figure is the consensus sequence. Dashes or spaces are introduced to achieve alignment of the activities. The data comes from Cluster 1 in Figure 4.**



**Figure 10. Hierarchical Tree Structure (we cited the same source code from ProM [9] here and made modifications showing only the number of clusters specified by the user). The result is based on the data in Figure 4.**

aware trace alignment is the mean activity duration of the corresponding column. Compared with the previous two visualizations, the alignment view makes it easier to interpret process traces and extract insights. When considering algorithm execution time, our previous research found that for a moderately-sized dataset (e.g. 50,000 activities, ~1,000 traces and ~50 activity for each trace), the alignment can be effectively calculated in  $25.5 \pm 1.5$  seconds [16]. This time is not instantaneous (which would be ideal), but is still reasonable.

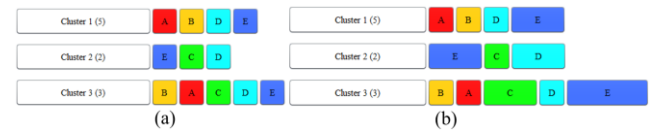
### 3.2 G2: Simplified Visualization of Process Traces

The first interactive visualization feature in G2 is the selection of cluster number (clicking button ① in Figure 6 and inputting cluster number  $k$  in the pop-up dialogue). A hierarchical tree structure with  $k$  clusters will be shown at the bottom panel (Figure 6 and **Error! Reference source not found.**) where the non-leaf (a.k.a. internal) nodes show the current height (a.k.a. depth) and process traces included under this node.  $k$  leaf nodes correspond to the  $k$  clusters and display all the process IDs in the cluster.

After clustering, each cluster is represented with its own cluster prototype. By default, the cluster prototypes are visualized as activity stacks (Figure 11). The prototypes can also be visualized in alignment view (Figure 6 and Figure 12) by clicking on the button “Align Cluster Prototype” (② in Figure 6). Another interactive function allows the user to check the pre-aggregated traces under a certain cluster. This feature may be accessed by clicking on the buttons showing the cluster information (③ in Figure 6).

### 3.3 G3: Visualization of Statistics of Trace Clusters vs. Trace Attributes.

Users can access statistics of trace clusters and trace attributes by clicking on the button “Multi-Logistic Regression” (⑤ in Figure 6)



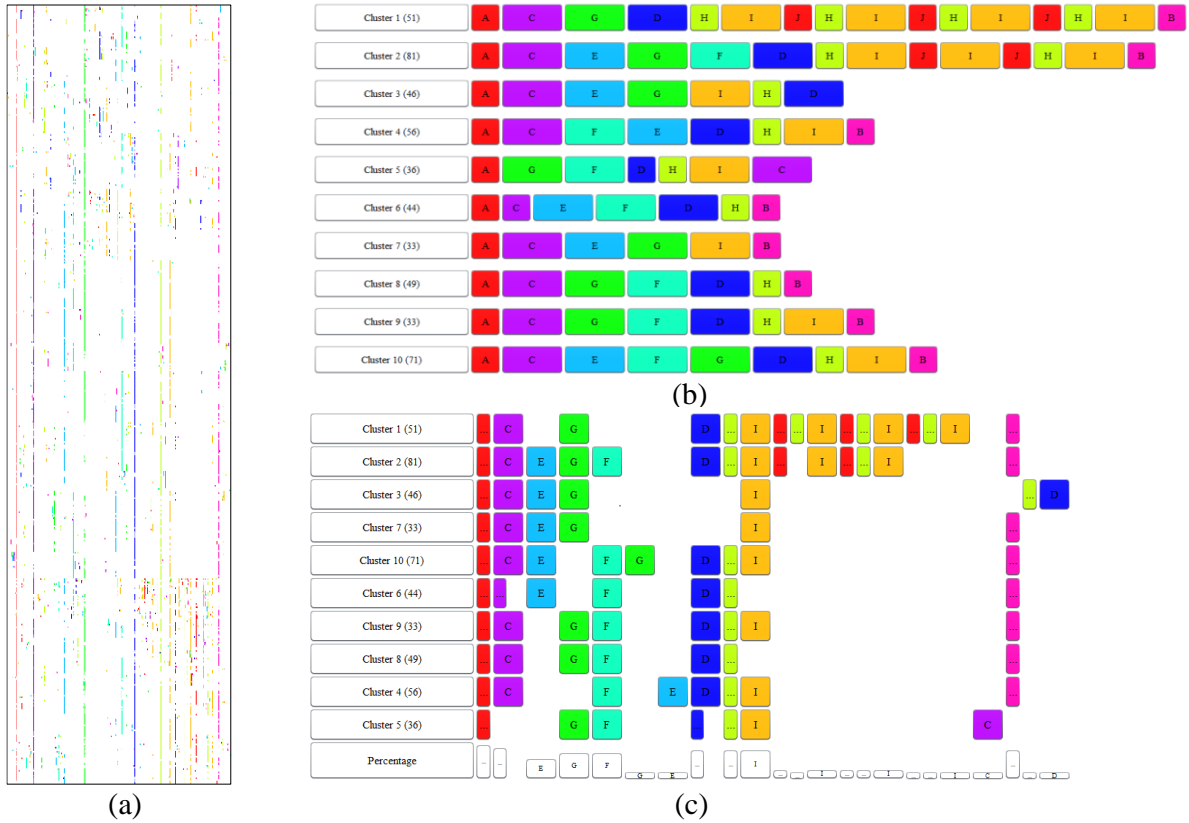
**Figure 11. Simplified visualization of raw process traces. Each row is a cluster's prototype. The information in the white block before the prototypes shows the cluster ID that each prototype represents and the number of process traces in that cluster. (a) Cluster prototypes are consensus sequences calculated from context-aware alignment (Error! Reference source not found.(a)); (b) Cluster prototypes are consensus sequences calculated from duration-aware alignment (Error! Reference source not found.(b)). The data comes from Figure 4.**



**Figure 12. Alignment view of the cluster prototypes in Figure 11(a). The data comes from Figure 4.**

	p-value(one vs all)	Coefficient						
Cluster 1	-16.38544	14.42526	14.31447	-1.15607	-1.0814	0.63728	-0.03663	1.10754
Cluster 2	0.3829	-2.3596	-1.2306	0.53868	0.22207	-0.89724	0.10001	-1.37452
Cluster 3	-1.40379	0.65426	0.44483	0.25073	0.61827	0.3251	0.39179	0.21481
Cluster 4	-0.37907	-0.56444	-0.55853	0.29681	-0.07276	-0.03752	-0.55082	-0.4284

**Figure 13. Statistics for regression coefficients**



**Figure 14. Visualization of artificially generated dataset. (a) Alignment view of all 500 process traces; (b) Simplified visualization of 500 process traces using 10 cluster prototypes; (c) Alignment view of 10 cluster prototypes.**

or on “Binomial Logistic Regression” (④ in Figure 6). The number of clusters is decided by the user. The significance tests for trace attributes on trace clusters ( $p$ -value statistics) are shown in a chart (⑥ in Figure 6, JFreeChart library [18] is used). The horizontal axis represents the  $p$ -value, while the vertical axis represents the trace attributes. The  $p$ -value of different clusters is denoted with different shapes and colors. Because  $\alpha = 0.05$  is widely used as the significance level, we placed a highlighted line at this level. When performing multinomial logistic regression, the reference category is set to the last-numbered category by default. Users, however, may change the reference category manually (⑦ in Figure 6). In addition to  $p$ -values for each trace attribute, the regression coefficients of the logistic regression model are also listed in a table (⑧ in Figure 6 and Figure 13).

### 3.4 Additional supportive functions

In addition to the three main goals, VIT-PLA also includes several useful supportive functions. The Activity Filter (⑨ in Figure 6) allows the user to include and exclude activities in the visualization and analysis. The Color Map (⑩ in Figure 6) allows the user to recolor the activity symbols. The Zoom Slider (⑪ in Figure 6) enables the user to resize the activity symbols in the visualization panel (the sliders in the top-right corner control the size of the activity symbols).

## 4. PRELIMINARY CASE STUDY

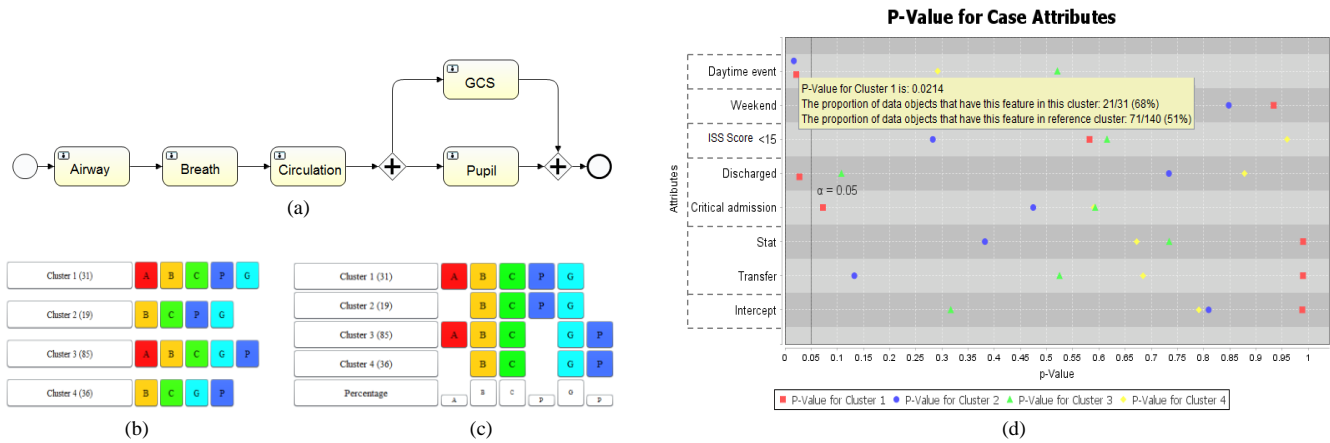
### 4.1 Case Study I: Artificial Data

#### 4.1.1 Data Description

This dataset was artificially generated using the Process Log Generator (PLG) [17]. It includes 500 process traces consisting of 10 different activity types. The drawback of this artificial data is that it does not have background attributes associated with each process trace. For this reason, we only focus on the simplification of trace visualization when using this dataset.

#### 4.1.2 Results and Discussion

The visualization of 500 process traces without data aggregation strategies can lead to extremely large and complex visualization results (Figure 14(a)). When represented this way, the symbols are too small to identify, making it difficult to extract useful information. To improve visualization, we used clustering to aggregate the original dataset into a small number of representative process traces (Figure 14(b)). In this example, we arbitrarily chose 10 clusters, a manageable number of clusters to understand. The visualization becomes clearer when put into the alignment view (Figure 14(c)). From these two simplified visualizations (Figure 14(b) and Figure 14(c)), it is easy to extract some interesting insights: (1) the sequential order of consensus tasks (tasks that occur more than or equal to 50% in the column) is “ACEGFDHIB”; (2) the pattern “HIJ” is repeated in two of the ten clusters (cluster 1 and cluster 2); (3) activity C is performed late in one cluster (cluster 5); and (4) activity D is performed late in one cluster (cluster 3) and omitted in another (cluster 7).



**Figure 15. (a) Workflow model (drawn based on BPMN) given by domain expert describing the initial evaluation of trauma, (b) Simplified visualization of 171 traces using four cluster prototypes, (c) Alignment view of four cluster prototypes (d)  $p$ -value for binomial logistic regression coefficients**

## 4.2 Case Study II: Trauma Resuscitation Workflow Data

### 4.2.1 Data Description

We used a trace log obtained from video analysis of 171 child trauma resuscitations between May and August 2013 at Children’s National Medical Center in Washington, DC. An event log of five activities typically performed during the initial evaluation was created and used as the dataset for this case study. We obtained the workflow model for these activities from domain experts (Figure 15(a)). Activities “Airway, Breath, Circulation” follow a sequential order. Activities “GCS” and “Pupil check” are parallel and should be performed after the previous three activities. We also obtained from the medical chart review several patient and resuscitation attributes (including pre-hospital triage level, the resuscitation’s time of day and day of week, Injury Severity Score [ISS], and patient admission status after the resuscitation) (Table 1). This dataset is not a “large process log,” but we chose it for our preliminary analysis to demonstrate how our approach can be integrated with medical domain knowledge.

### 4.2.2 Results and Discussion

#### 4.2.2.1 Data Interpretation from Visual Analysis

Four cluster prototypes were generated (Figure 15(b) and (c)). Prototypes of clusters 1 and 3 conform to our expert model, but clusters 2 and 4 do not. From the alignment view of prototypes, we can observe that the sequential order of activity GCS (G) and pupil assessment (P) is interchangeable, which conforms with the parallel structure in our expert model. Visualizations of pre-aggregated

**Table 1 Process trace attributes**

Attribute List	Values		
Weekend Event	1	0	
Daytime Event	1	0	
ISS Score	<15	≥15	
Activation Level <sup>a</sup>	Attending Stat	Stat	Transfer
EDDISPGroup <sup>b</sup>	Non-critical Admission	Critical Admission	Discharged

<sup>a</sup>. Activation level = pre-hospital triage level  
<sup>b</sup>. EDDISPGroup = admission status of patients after ED care

traces for each prototype are not displayed, but users can visualize the traces by clicking on the cluster button at the front of each row (Figure 15(b) and (c)).

With the attribute data for these process traces, we can perform statistical analysis to explore the underlying correlation between the trace attributes and trace cluster membership. The following are examples of the statistical findings, followed by feedback from domain experts:

**Observation #1:** Attribute “Daytime Event” is statistically significant ( $p$ -value = 0.021, red square point in row “Daytime event” in Figure 15) for cluster 1. The regression coefficient of Daytime Event is 1.108 (Figure 13). This attribute is statistically significant because the proportion of data objects that have this feature (daytime = 1) in this cluster is 12/31 (68%), while the proportion of data objects that have this feature (daytime = 1) in the reference category (all other cluster) is 71/140 (51%).

**Observation #2:** Attribute “Daytime Event” is statistically significant ( $p$ -value = 0.017, blue circle point in row “Daytime event” in Figure 15) for cluster 2. The regression coefficient of Daytime Event is  $-1.375$  (Figure 13). This attribute is significantly significant because the proportion of data objects that have this feature (daytime = 1) in this cluster is 6/19 (31%), while the proportion of data objects that have this feature (daytime = 1) in the reference category (all other cluster) is 86/152 (57%).

**Medical expert feedback:** For the care of injured patients, improved outcomes are associated with compliance with the Advanced Trauma Life Support model [19], represented here as the expert model. We find that one cluster (cluster 1) whose cluster prototype follows the model occurs more often during the day and another cluster (cluster 2) whose cluster prototype deviates from the model occurs more often at night. This association finding supports previous work showing decreased compliance with trauma protocols at night [20].

#### 4.2.2.2 Domain Expert Feedback on VIT-PLA

##### Design:

To evaluate the quality of our design, we had two medical domain experts evaluate a prototype of VIT-PLA. Both positive and negative feedback was received.

Both domain experts liked the visualization’s flexibility and interactivity. They found that its data clustering, activity filtering, symbol resizing, and recoloring functions were very useful. They

were also found that with the knowledge uncovered by the program's statistical analysis was useful. One domain expert found it useful to switch between the aggregated data and the original traces, and also commented on the helpfulness of the cluster's "average sequence".

Most negative comments focused on our approach for statistical analysis. One domain expert felt that data-driven clustering approach lacked consistency because its result varied when different clustering algorithms or similarity metrics were used. Also, the domain expert found that some small clusters did not have sufficient data to support the statistical hypothesis test correlating trace clusters and trace attributes.

## 5. SUMMARY AND FUTURE WORK

As process mining finds increased usage in many domains, visual analytic tools for process sequences are in high demand. We introduced VIT-PLA, a visual and interactive workflow data analysis tool that is able to visualize large process logs. With these visualizations and integrated statistical testing, VIT-PLA is able to obtain results not revealed by simple observation.

The limitation of our current work is that we only implemented the hierarchical clustering approach with two process trace proximity metrics. In our future work, we will evaluate other clustering algorithms (e.g. KNN, feature-based k-means, HMM-based clustering). Also, the determination of cluster number, a typically non-trivial task, is still manual. In the future, we plan on building a function that suggests cluster number based on some cluster metric.

## 6. ACKNOWLEDGMENTS:

This research is supported by National Institutes of Health under grant number 1R01LM011834-01A1.

## 7. REFERENCES

- [1] Van Der Aalst, Wil. Process mining: discovery, conformance and enhancement of business processes. Springer Science & Business Media, 2011
- [2] Van der Aalst, Wil, Ton Weijters, and Laura Maruster. "Workflow mining: Discovering process models from event logs." *Knowledge and Data Engineering*, IEEE Transactions on 16.9 (2004): 1128-1142.
- [3] Monroe, Megan, et al. "Temporal event sequence simplification." *Visualization and Computer Graphics*, IEEE Transactions on 19.12 (2013): 2227-2236.
- [4] Malik, Sana, et al. "Cohort comparison of event sequences with balanced integration of visual analytics and statistics." *Proceedings of the 20th International Conference on Intelligent User Interfaces*. ACM, 2015.
- [5] Wongsuphasawat, Krist, and David Gotz. "Exploring flow, factors, and outcomes of temporal event sequences with the outflow visualization." *Visualization and Computer Graphics*, IEEE Transactions on 18.12 (2012): 2659-2668.
- [6] Perer, Adam, and Fei Wang. "Frequency: interactive mining and visualization of temporal frequent event sequences." *Proceedings of the 19th international conference on Intelligent User Interfaces*. ACM, 2014.
- [7] Perer, Adam, Fei Wang, and Jiaying Hu. "Mining and exploring care pathways from electronic medical records with visual analytics." *Journal of biomedical informatics* 56 (2015): 369-378.
- [8] Song, Minseok, and Wil MP van der Aalst. "Supporting process mining by showing events at a glance." *7th Annual Workshop on Information Technologies and Systems*. 2007.
- [9] Bose, RP Jagadeesh Chandra, and Wil MP van der Aalst. "Process diagnostics using trace alignment: opportunities, issues, and challenges." *Information Systems* 37.2 (2012): 117-141
- [10] Bose, RP Jagadeesh Chandra, and Wil MP van der Aalst. "Context Aware Trace Clustering: Towards Improving Process Mining Results." *SDM*. 2009.
- [11] Levenshtein, Vladimir I. "Binary codes capable of correcting deletions, insertions, and reversals." *Soviet physics doklady*. Vol. 10. No. 8. 1966.
- [12] Czepiel, Scott A. "Maximum likelihood estimation of logistic regression models: theory and implementation." Available at [czep.net/stat/mlelr.pdf](http://czep.net/stat/mlelr.pdf)(2002).
- [13] Wasserman, Larry. *All of statistics: a concise course in statistical inference*. Springer Science & Business Media, 2013.
- [14] Tan, Pang-Ning, Michael Steinbach, and Vipin Kumar. *Introduction to data mining*. Vol. 1. Boston: Pearson Addison Wesley, 2006.
- [15] Jain, Anil K., and Richard C. Dubes. *Algorithms for clustering data*. Prentice-Hall, Inc., 1988.
- [16] Sen Yang, Moliang Zhou, Rachel Webman, JaeWon Yang, Aleksandra Sarcevic, Ivan Marsic and Randall S. Burd, "Duration-Aware Alignment of Process Traces", Accepted for Industrial Conference on Data Mining (ICDM 2016), New York, NY, July 13-17, 2016
- [17] Burattin, Andrea, and Alessandro Sperduti. *PLG: a Process Log Generator*. Tech. rep, 2010.
- [18] <http://www.jfree.org/>
- [19] van Olden, Ger DJ, et al. "Clinical impact of advanced trauma life support." *The American journal of emergency medicine* 22.7 (2004): 522-525.
- [20] Carter, Elizabeth A., et al. "Adherence to ATLS primary and secondary surveys during pediatric trauma resuscitation." *Resuscitation* 84.1 (2013): 66-71.
- [21] Du, Fan, et al. "Coping with Volume and Variety in Temporal Event Sequences: Strategies for Sharpening Analytic Focus."
- [22] Ward Jr, Joe H. "Hierarchical grouping to optimize an objective function." *Journal of the American statistical association* 58.301 (1963): 236-244.