Interactive Visual Patient Cohort Analysis

Zhiyuan Zhang, David Gotz, and Adam Perer

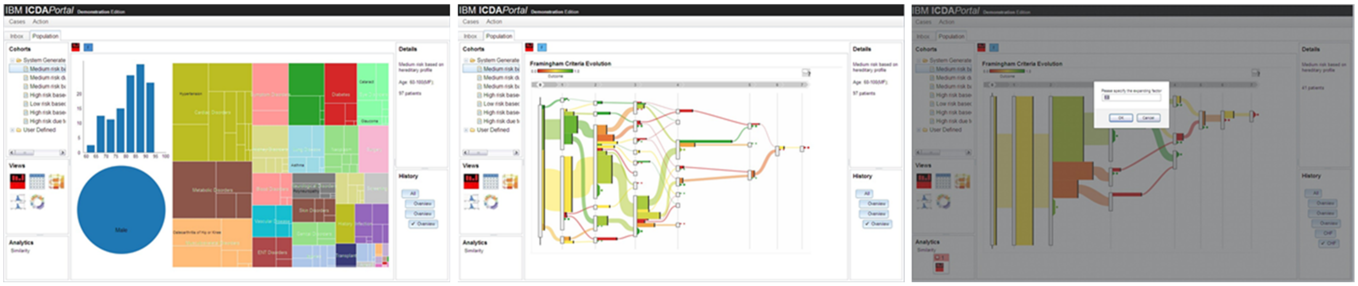

Figure 1. (a) Summary view of a cohort of males over 60, visual filtered from an initial cohort of "medium risk of heart failure" patients. (b) The same sub-cohort visualized for symptom progression. (c) After further filtering reduces the cohort below a statistically significant size, similarity analytics can be used to grow the cohort by querying for additional similar patients.

Abstract—Retrospective patient cohort analysis is a widely used technique in many healthcare studies. Due to its data intensive nature, the traditional analytical pipeline requires expertise from several areas, such as databases, data mining, software development, statistics, and domain knowledge. As a result, domain experts often rely on a team of technologists to help perform such studies which can make the process slow and cumbersome. To allow domain experts to perform faster and more flexible analyses, we designed an integrated system that combines visual exploration and data analytics with an intuitive user interface. Our system lets clinicians interactively visualize and refine cohorts, request analytics on those cohorts, and make new discoveries.

Index Terms—Cohort Study, Retrospective Patient Cohort Analysis, Visual Analytics, Interactive Cohort Definition and Refinement.

1 Introduction

Retrospective patient cohort analysis [2] is the analysis of patients' medical and diagnostic histories to make healthcare discoveries. In the traditional pipeline, analysts work manually to define specific cohort constraints (e.g., "female patients over age 70") or apply specialized batch analytics to computationally determine a meaningful group of patients (e.g., high-utilization cohorts [1]). Unfortunately, both methods have limitations. For the definition of the cohort constraints, it's difficult to select the attributes that are to be queried from a list of hundreds or thousands of patient attributes. For batch analytics that behave like a "black box", users have few ways to apply their domain expertise to influence the process.

Once a patient cohort has been defined, the next step in the analysis process is to apply specific statistics or data mining techniques to the cohort and look at the results to uncover insights. These steps can often be unintuitive for clinical users. As a result, exploratory analysis can often require several iterations of work between domain experts and computational staffs. This can significantly slow down the process and limit the clinical analyst's flexibility to explore.

To address these challenges, we have designed an integrated system that combines visual exploration and data analytics with an intuitive user interface. The system empowers clinical users to quickly and efficiently perform interactive visual patient cohort analyses. Using our system:

- ♦ Patient cohorts can be interactively defined and modified at any step of an analysis;
- ♦ Cohorts can be visualized in various ways and users can pivot easily between different visualization metaphors; and
- ♦ Analytics can be applied to cohorts for on-demand processing at any time in an analysis,

The system's user interface supports these tasks by allow direct manipulation of three key artifacts: (1) ***cohorts***, (2) ***views***, and (3) ***analytics***. Cohorts represent sets of patients and their associated information. Views are visualization components used to graphically represent and interactively refine the cohorts. Analytics operate on cohorts and are used to generate new cohorts, produce additional data for a specific cohort, or to otherwise modify (e.g., expand or segment) an existing cohort. The remainder of this paper describes our system design and user interaction model, and presents a use case that demonstrates the utility of our approach.

2 System Design

Our system design manages three types of artifacts—cohorts, views, and analytics—and connects them within a single integrated user interface. This section describes the artifact types in more detail and discusses how they connect within our architecture.

2.1 Cohorts

Patient cohorts are groups of patients and their associated information, such as gender, age, diagnoses, and treatments. A cohort serves as the underlying data structure that is used to pass data throughout the system's pipeline. They are the objects on which the other two artifacts—analytics and views—operate. Included within each patient cohort representation is a list of individual patient identification numbers that can be used to connect cohort members with more detailed clinical data located in a remote data store.

2.2 Analytics

Analytics are computational components that operate on cohorts in various ways. Our system supports two main types of analytics: (1) ***batch analytics*** and (2) ***on-demand analytics***. Batch analytics are components that are executed automatically in the background by our system (e.g., nightly as new patient data is imported to the system). They process an entire patient population and identify groups of interest. For example, a batch analytic may be used to perform risk stratification, generating lists of patients that have common sets of risk factors. The batch analytics components generate new cohorts that can serve as starting points for a user's exploratory analysis.

• Zhiyuan Zhang is with Stony Brook University.
• David Gotz and Adam Perer are with IBM Research.

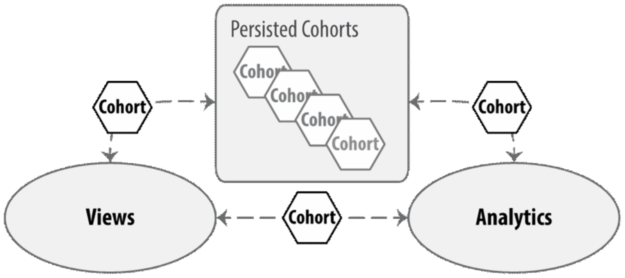

Figure 2. Patient cohorts are passed between views and analytics via drag-and-drop interactions as an analysis unfolds. Cohorts can also be persisted for future reference.

On-demand analytics, in contrast, are performed in ad-hoc fashion at the specific request of a user. On-demand analytics take as input a specific patient cohort, plus an optional set of input parameters. In response, an on-demand analytics can produce additional information about patients in the cohort (e.g., calculate risk scores) and/or refine the membership of the cohort (e.g., query for additional similar patients).

2.3 Views

Views are visualization components that offer specific targeted ways to graphically depict and interact with a patient cohort. Each view is designed to take a single cohort as input and render a specific subset of patient features. Views also provide interactive capabilities through which users can selectively brush and filter to explore and refine the set of patients in the cohort.

For example, our system includes a patient cohort summary view that depicts general information about a group of patients such as age and gender distributions along with Treemap [3] summarizing diagnosis code statistics. This view provides multiple coordinated visualizations through which users can refine the set of patients in a cohort (e.g., “filter to only male patients over age 50 with specific classes of cancer”). Our system also provides a generic table view to look at a detailed list of patients in a cohort including individual patient identification numbers.

Beyond these generic views, additional components are provided for use-case specific visualizations. For example, another view provided by our prototype is the Outflow visualization for exploring patient symptom evolution [4].

Each view has the additional ability to export the set of patients being visualized at any given point in time. Therefore, from a data perspective, views are very much like on-demand analytics in that they both take a cohort as input and produce a cohort as output.

2.4 Interactions

The similarity in data flows for both on-demand analytics and views are critical to the design of our system. This commonality allows users to chain together views and analytics—both serving as operators on cohorts—into arbitrary sequences. This lets users interactively perform complex and ad hoc exploratory analysis processes that mix visual interactions and filtering with computational analysis routines. The approach is illustrated in Figure 2.

Users interact with our system primarily via a drag and drop model that connects our three types of artifacts. Users drag cohorts to views to visualize them, and drag cohorts to analytics to process them. Users can select a cohort from either the current view or from a list of saved cohorts in a sidebar. The sidebar contains both system generated cohorts (via batch analytics) and those that have been manually defined (via prior user interaction).

In addition, individual views allow selection, brushing, and filtering. Callouts and a dedicated sidebar panel are also used to provide more information about moused-over elements. A user’s analytic history is summarized in a sidebar, capturing the provenance of the currently viewed cohort and allowing a user to revisit prior stages of his/her investigation.

3 Use Case Demonstrating Typical Workflow

We have developed a prototype implementation of our design as a web-based interactive visualization application. This section reviews an example use case (see Figure 1) to demonstrate the typical workflow that our system supports.

A session starts with a user selecting a cohort from the sidebar located on the left side of the user interface, such as a group of patients flagged as being at risk of developing heart disease. The user can drag and drop the cohort from the sidebar onto the view icon for our cohort summary visualization. The user can interactively explore the aggregate information about the cohort and apply filters to modify the cohort for a specific analytic task. For example, Figure 1(a) shows the summary view after filtering to only male patients over the age of 60.

The cohort created by the filtering step can then be dragged-and-dropped onto an Outflow view to visualize the variations in disease progression for the set of identified patients. It is clear from the view shown in Figure 1(b) that significant variations have been observed. A user can then select a specific pathway from the Outflow visualization and apply additional filters. For example, a user might select the largest pathway, which has mixed outcomes to perform further analysis.

This iterative filtering process using multiple views of the data can let users quickly and intuitively identify a population of interest. However, additional information is often needed that was not contained in the original cohort. For example, in the sequence described here the user has applied several filters that have reduced the size of the cohort population significantly. This reduces its statistical power.

In order to re-grow the patient population, which would allow the user to draw more meaningful conclusions, the user can take advantage of our system’s on-demand analytics to retrieve additional patients that are similar to those in the current cohort but that were left out of the initial cohort that was first used to start the investigation.

On-demand analytics are initiated when a user drags the cohort from the current view (or a persisted cohort from the sidebar) to the analysis component of their choice. All available on-demand analytics are listed in the lower left sidebar. After a cohort is dropped on a specific analytic component, the system immediately begins the analysis process. If additional input parameters are required by a given analytics, a dialog box is displayed to gather the needed user input. For example, Figure 1(c) shows the dialog box shown for our system’s patient similarity analytic component. This analysis algorithm needs to know an “expansion factor” that specifies how many similar patients to retrieve as it grows the input cohort. For example, an expansion factor of 0.2 will grow the size of a cohort by 20%.

After the similarity computation completes, an expanded cohort is returned and immediately visualized using the same view that was active prior to the analytics request.

References

[1] J. Hu, F. Wang, J. Sun, R. Sorrentino, and S. Ebadollahi. “A Healthcare Utilization Analysis Framework for Hot Spotting and Contextual Anomaly Detection.” AMIA, 2012.

[2] M. Porta (editor). A Dictionary of Epidemiology. 5th. ed. New York: *Oxford University Press*, 2008.

[3] B. Shneiderman. “Tree Visualization with Tree-Maps: 2-d Space-filling Approach.” *ACM Transactions on Graphics*, 11(1), pp. 92-99, 1992.

[4] K. Wongsuphasawat and D. Gotz. “Exploring Flow, Factors, and Outcomes of Temporal Event Sequences with the Outflow Visualization.” *IEEE Information Visualization,* 2012.