Selection Bias Tracking and Detailed Subset Comparison for High-Dimensional Data

David Borland, Wenyuan Wang, Jonathan Zhang, Joshua Shrestha, and David Gotz

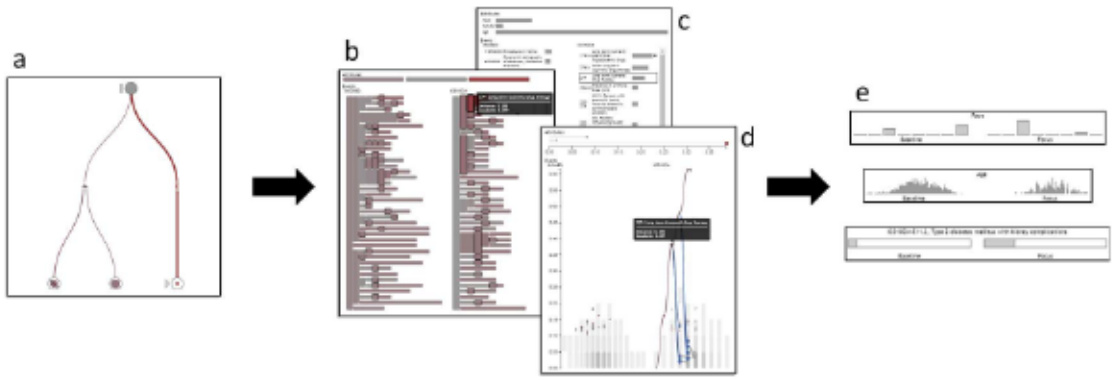

Fig. 1. Exploratory cohort selection in high-dimensional datasets can lead to selection bias–unintended side-effects in variable distributions–that may go unnoticed by the user. Our selection bias tracking system and detailed cohort comparison visualizations, deployed in a medical temporal event sequence visual analytics tool, include (a) a cohort provenance tree to keep track of created cohorts and indicate when selection bias may have occurred, (b-d) a suite of high-dimensional cohort comparison visualizations that employ hierarchical aggregation to display the differences between two cohorts in detail, and (e) data-type dependent comparisons of individual variable distributions.

Abstract— The collection of large, complex datasets has become common across a wide variety of domains. Visual analytics tools increasingly play a key role in exploring and answering complex questions about these large datasets. However, many visualizations are not designed to concurrently visualize the large number of dimensions present in complex datasets (e.g. tens of thousands of distinct codes in an electronic health record system). This fact, combined with the ability of many visual analytics systems to enable rapid, ad-hoc specification of groups, or cohorts, of individuals based on a small subset of visualized dimensions, leads to the possibility of introducing selection bias–when the user creates a cohort based on a specified set of dimensions, differences across many other unseen dimensions may also be introduced. These unintended side effects may result in the cohort no longer being representative of the larger population intended to be studied, which can negatively affect the validity of subsequent analyses. We present techniques for selection bias tracking and visualization that can be incorporated into high-dimensional exploratory visual analytics systems, with a focus on medical data with existing data hierarchies. These techniques include: (1) tree-based cohort provenance and visualization, including a user-specified baseline cohort that all other cohorts are compared against, and visual encoding of cohort "drift", which indicates where selection bias may have occurred, and (2) a set of visualizations, including a novel icicle-plot based visualization, to compare in detail the per-dimension differences between the baseline and a user-specified focus cohort. These techniques are integrated into a medical temporal event sequence visual analytics tool. We present example use cases and report findings from domain expert user interviews.

Index Terms—High-dimensional visualization, visual analytics, cohort selection, medical informatics, selection bias

1 Introduction

The collection of large, complex datasets has become common across a wide variety of domains, such as advertising, security, and healthcare. In addition to analytical approaches such as statistics, data mining, and machine learning, visual analytics increasingly plays a key role in exploring and answering complex questions about such large datasets to support precision, evidence-based decision making [25, 49].

Two distinct challenges encountered with large datasets are data *volume* and data *complexity*. Volume typically refers to the number of records in a given dataset, e.g. the number of patients in an electronic health record (EHR) system. Complexity, meanwhile, reflects the number of dimensions, e.g. the various demographic, diagnosis, procedure, lab, and medication data stored for each patient in an EHR. For many analytical tasks increased volume can be managed with improved processing power. Moreover, many visualizations scale well with increasing volume because they depict aggregate values—a bar chart of a binary variable works equally well with ten records as with 1 billion records. Data complexity, however, presents a more fundamental challenge. For example, many EHRs use coding systems that can contain hundreds of thousands of unique codes to represent different diagnoses, procedures, medications, etc. (e.g. [1, 45]). Although many high-dimensional visualization techniques exist (e.g. [11, 28] provide reviews), they can typically only show a relatively small subset of the

- *David Borland is with RENCI at the University of North Carolina at Chapel Hill. E-mail: borland@renci.org.*
- *Wenyuan Wang is with the School of Information and Library Science at the University of North Carolina at Chapel Hill. E-mail: vaapad@live.unc.edu.*
- *Jonathan Zhang is with the Dept. of Biostatistics at the University of North Carolina at Chapel Hill. E-mail: jzhang42@live.unc.edu.*
- *Joshua Shrestha is with the Dept. of Computer Science at the University of North Carolina at Chapel Hill. E-mail: prayash@live.unc.edu.*
- *David Gotz is with the School of Information and Library Science at the University of North Carolina at Chapel Hill. E-mail: gotz@unc.edu.*

Manuscript received 31 Mar. 2019; accepted 1 Aug. 2019.
Date of publication 16 Aug. 2019; date of current version 20 Oct. 2019.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TVCG.2019.2934209

dimensions represented in such complex datasets at any given time, via techniques such as dimension selection or projection.

At the same time, a wide variety of visual analytics applications have successfully employed Shneiderman's mantra of "overview first, zoom and filter, then details-on-demand" [42] to help manage the complexity of high-dimensional datasets. Indeed, filtering is a common step in many analytic workflows.

However, the ability of sophisticated visual analytics systems to enable rapid, ad-hoc filtering of complex datasets while simultaneously focusing the user's attention on a narrow subspace of the overall dataset at any given point in time can create a situation ripe for issues such as *selection bias*. Selection bias occurs when the users selects a sample for analysis in such a way that the sample is not representative of the larger group that is intended to be analyzed [22]. For example, in the medical domain, certain medications cannot be used together due to drug-drug interactions. As a result, filtering patient data on one drug during a medical analysis will result in a patient population with a much lower-than-typical rate of the second drug. This shift in the distribution of the second drug may be an *unintended side-effect* of the filtering operation, and may go unnoticed by the user, especially if data for the second drug are not currently shown in the visualization.

More generally, when a user applies filters to create data subsets based on a small set of dimensions, there may be large shifts in the distributions of other "unseen" correlated dimensions. If such side-effects go unnoticed, they could threaten the validity of subsequent analyses. In the medical domain, such side-effects can be especially problematic because there are many inter-related dimensions, and selection bias could lead to incorrect evidence for medical decision making.

Contextual visualization methods [3] are one proposed approach for addressing selection bias and related challenges. One such method, deployed in a medical application to address selection bias, is adaptive contextualization [16, 17]. In this approach, subsets of patients, or cohorts, are interactively generated by the user of a visual analytics interface depicting medical event sequences. The system keeps track of each generated cohort, and computes a distance measure indicating to what degree the variable distribution of each cohort has "drifted" from a baseline cohort, which may indicate unintended selection bias. The user can inspect a list of event types to see the degree of drift for each individual variable. Although shown to be effective, the system exhibits two critical limitations: (1) it does not enable key types of bias comparisons required by various fields, including medical cohort analysis [31], because it only considers a linear sequence of selection steps, and (2) it does not account for important variable interactions, including hierarchical relationships, focusing instead on independent univariate measures.

This paper builds on this earlier work, introducing a new approach that addresses these limitations. Selection bias is a known issue in various medical domains (e.g. [22, 33, 50, 51]), and we focus our work on cohort selection from medical event sequence data, taking advantage of existing medical data hierarchies. Given a visual analytics interface that enables the user to create multiple cohorts of patients via various filter paths from a root dataset, the primary goals of the work in this paper are as follows: (1) Enable visualization of the *created cohorts and the steps taken to create them*. (2) Enable the user to identify the *degree to which the variable distribution for each cohort has drifted* from a user-specified baseline–an indicator of potential selection bias. (3) Enable the user to identify *which filter operations contribute the most* to the potential selection bias for any cohort. (4) For a user-selected focus data cohort, enable the user to investigate in-depth the *drift for each dimension* compared to the current baseline, prioritizing dimensions which have drifted most drastically. (5) Leverage *hierarchical relationships* between dimensions to better communicate areas of drift within high-dimensional data. Accomplishing these goals will alert the user to any selection bias introduced in their current analysis, enabling them to correct for such bias in subsequent analyses.

The key research contributions presented in furtherance of these goals include:

- A tree-based cohort provenance visualization, showing the full non-linear selection process from an initial query result to one or more final cohorts, along with the amount of selection bias introduced at each step. The user can interactively select a baseline cohort against which all cohorts are compared, and a focus cohort for detailed comparison with the baseline.
- A set of visualizations for communicating selection bias between pairs of cohorts from a high-dimensional dataset, including a novel visualization technique adapted from icicle plots. A user-specified aggregation level is incorporated into each visualization to scale effectively for high-dimensional datasets, while prioritizing salient dimensions with respect to selection bias.
- Integration of these, and other relevant cohort comparison visualizations, within *Cadence*, a temporal event sequence visual analytics and cohort selection tool.

This paper describes in detail the contributions listed above, presents example use cases, and reports findings from medical domain expert interviews.

2 Related Work

The section provides a brief overview of related work that is most relevant to the contributions of this paper: hierarchical visualization, visual comparison, bias, and medical cohort management.

2.1 Hierarchical Visualization

Both the cohort provenance (Section 6.1) and two of the detailed cohort comparison visualizations (Sections 6.3 and 6.4) presented in this paper involve the visualization of hierarchies. A hierarchy consists of a set of nodes N and directed edges E. A single *root* node has only outgoing edges; all other nodes must have one incoming edge, from its *parent*, and zero or more outgoing edges, to its *children*. Hierarchies are used across a wide range of domains to organize data, and hierarchical aggregation is often used to manage visual complexity in data visualization [10, 57]. The various visualization approaches developed to communicate hierarchical structures are often classified as node-link representations or implicit/space-filling techniques [38].

Node-link diagrams represent each node in the hierarchy as a glyph, and each hierarchical relationship as a mark (e.g. line, curve, etc.) connecting each child node glyph to its parent. Layout strategies for node-link diagrams include phylogenetic trees (e.g., [47]) and dendrograms, which draw all leaf nodes at the same depth and are often used for displaying hierarchical clustering output (e.g., [37]). In contrast, layouts such as tidy trees [35] draw each level in the tree at the same depth, potentially resulting in a "ragged" appearance for leaf nodes. Our cohort provenance tree uses a node-link diagram to enable the encoding of information along each link in the tree, with a tidy-tree layout to emphasize the sequences of steps taken to form each cohort.

Implicit techniques include treemaps [24, 41] and icicle plots [27]. These techniques do not draw parent-child relationships directly, instead encoding them using enclosure or adjacency. As a result, they offer compact representations and are often suitable for the visualization of large hierarchies. However, the ability to encode information for each link is hindered. For our detailed cohort comparison visualizations we explore the use of implicit (Section 6.3) and node-link (Section 6.4) representations, incorporating a user-specified aggregation level while prioritizing salient features in the hierarchy.

2.2 Visual Comparison

Comparison is a fundamental process for many visualization tasks, and a variety of visual comparison systems and approaches have been developed to meet specific needs. In light of such diversity, Gleicher et al. developed a taxonomy of comparative visualization techniques: (1) juxtaposition, by showing objects separately, (2) superposition, by overlaying objects in the same space, and (3) explicit encoding, by directly representing the relationships between objects [13].

The cohort provenance visualization described in this paper adopts a tree structure that is inspired in part by CONSORT diagrams (Section 2.4). We therefore employ an explicit encoding of comparison between nodes in the tree (representing cohort drift) that is overlaid onto the tree structure.

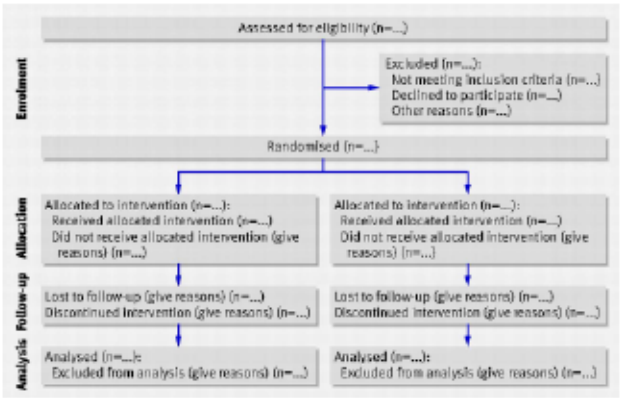

Fig. 2. CONSORT flow diagram for a two-group randomized trial [31].

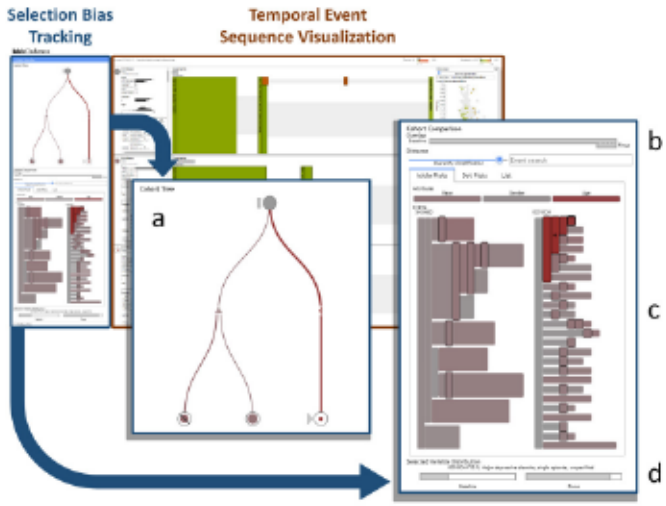

Fig. 3. Overview of the *Cadence* visual analytics tool, with selection bias tracking (the focus of this paper) on the left, and temporal event sequence visualization on the right. Closeup images of the selection bias tracking components show the (a) cohort provenance tree, (b) cohort overlap, (c) detailed cohort distance, and (d) selected variable distribution visualizations.

Various approaches have been developed for the visual comparison of hierarchical data, including static topology over time (e.g. [2,6,56]), interactive tree comparison (e.g. [4,23]), stable treemap layouts for improved juxtaposition comparisons (e.g. [19,21,44,46,48,52]), and dynamic topology (e.g. [26,29]). Cui et al. employ dynamic simplification of hierarchies evolving over time, with the ability to interactively zoom to lower levels of detail [7]. Our detailed cohort comparison visualizations adopt a similar strategy, employing user-controlled hierarchical aggregation and interactive level-of-detail exploration. However, our focus is on conveying where in the hierarchical data structure two cohorts are most different. Both the split icicle plot (Section 6.3) and dot plot (Section 6.4) visualization designs explicitly encode the differences between each cohort, conveyed via a shared hierarchical structure.

2.3 Bias

Bias is the (unfair) disproportionate weighting in favor of or against one thing compared to another. Various forms of bias have been recognized, including cognitive and statistical biases. Although domain expertise represents a form of bias that can help an analyst complete a task, e.g. via heuristic short cuts for filtering out unnecessary information [12], bias can also negatively affect the integrity of an analytical process and the validity of analytical results. For this reason, bias has increasingly become a topic of study across a variety of disciplines, including machine learning and visual analytics (e.g. [8,9,55]). Much of the focus in the visualization community has focused on characterizing and addressing cognitive biases [5,53,54]. Expanding on prior work in adaptive contextualization [16,17], the work presented in the paper focuses on the issue of selection bias–when individuals are selected for analysis in such a way that the sample is not representative of the population intended to be analyzed–an issue exacerbated by the interactive analysis of large, complex datasets, and of particular concern within the medical domain.

2.4 Medical Patient Cohort Management

The randomized control trial (RCT) is a widely used experimental design in fields such as epidemiology. The Consolidated Standards of Reporting Trials (CONSORT) statement [31,39] has been introduced for rigorous reporting of RCTs to avoid biased results (e.g. [40]). The CONSORT flow diagram presents how RCTs progress through various phases (Figure 2). The tree-structure of the diagram shows the construction of cohorts with different inclusion and exclusion criteria. Interactive cohort selection tools for medical research, such as i2b2 [20,32], enable researchers to query EHR databases and define cohorts based on various criteria. However, they do not typically include visual comparisons of constructed cohorts. Given the familiarity of CONSORT diagrams to the medical target audience for the techniques presented in this paper, they motivate in part the visual design adopted in the cohort provenance visualization (Section 6.1). This influence is reflected in the tree-based structure, the notions of inclusion and exclusion criteria, and the concepts of included and excluded cohorts.

Within the visualization community, a number of approaches have been developed to support patient cohort analysis (Rind et al. provide a review [36]), including various approaches for temporal event sequences (e.g. [15,30]), and cross-sectional phenotype studies [14]. However, these systems are not designed to explicitly track and visualize selection bias.

3 System Overview

The selection bias tracking and detailed cohort comparison methods described in this paper are integrated into *Cadence*, a medical temporal event sequence visual analytics tool (Figure 3). *Cadence* enables the selection of multiple cohorts of patients from an initial query result by filtering based on demographic attributes (*Race*, *Gender*, and *Age*) and sequences of temporal events (*Diagnoses* and *Procedures*). The precise nature of the methods implemented for visualizing temporal events and specifying cohorts is beyond the scope of this paper, and described elsewhere [18]. Instead, treating the cohort selection mechanisms as a black box, the bias tracking system that is the focus of this paper takes as its input (1) a set of cohorts $C = \{c_i\}$, where each cohort c_i is defined by a set of patients (each with their own respective attributes and events), and (2) a set of operators $O = \{o_i\}$, where each operator o_i corresponds to a filter (based on an attribute value or the presence or absence of an event) that defines the transformation of a parent cohort c_j to a child cohort c_k.

3.1 Data Description

In *Cadence*, medical data is represented using standardized coding systems including ICD-10-CM for diagnoses [1] and SNOMED-CT for procedures [45]. Both coding systems are hierarchical in structure, with ICD-10-CM containing over 70,000 distinct codes and SNOMED-CT over 300,000 distinct codes (of which a subset corresponds to medical procedures). Typically, data from EHR systems can contain codes from various levels of the coding hierarchy. For example, one patient may be diagnosed with the ICD-10-CM code *150: Heart Failure*, whereas another may be diagnosed with a more specific code, such as *150.32: Chronic diastolic (congestive) heart failure*. For each patient, medical event types can be viewed as binary variables: *present* (recorded in the patient's data) or *absent* (not recorded in the patient's data). Moreover, if a given code is present for a patient, all ancestors of that code in the code hierarchy are also considered present. For example, a patient with *150.32* would also be considered to have the more generic *150*

	Patients	Distinct Event Types		
		SNOMED-CT	ICD-10-CM	Total
Minimum	1,732	3,594	8,159	11,753
Average	4,936	4,305	9,692	13,997
Maximum	8,360	4,750	10,626	15,376

Table 1. Statistics summarizing event data returned by 12 queries using *Cadence*, each of which would form the initial cohort for an analysis.

diagnosis code. Finally, the data are relatively sparse, with some codes used frequently, others rarely, and many not at all within a given cohort.

To provide a sense for the complexity of the data in this domain, Table 1 summarizes statistics for a dozen queries run against real-world medical data in *Cadence*. The smallest query result returned 3,594 distinct codes from the SNOMED-CT hierarchy and 8,159 from the ICD-10-CM hierarchy, whereas the largest contained 4,750 unique SNOMED-CT codes and over 10,000 unique ICD-10-CM codes. The large numbers of unique codes in a typical query result motivated the development of the hierarchical aggregation methods used for the split icicle plot (Section 6.3) and hierarchical dot plot visualizations (Section 6.4) described later in this paper.

4 Requirements for Selection Bias Tracking and Detailed Cohort Comparison

Our system aims to alert the user to potential selection bias that can occur during rapid, ad-hoc cohort selection in high-dimensional datasets, leading to the following design requirements:

- **R1** Provide an intuitive interface for keeping track of created cohorts and the steps taken to create them.
- **R2** Enable identification of which cohorts may be subject to selection bias.
- **R3** Enable identification of which filter operations contribute the most to potential selection bias for a given cohort.
- **R4** Enable in-depth investigation of the distribution drift for each dimension between any two cohorts.
- **R5** Leverage hierarchical relationships between dimensions to better communicate areas of drift in high-dimensional data.

5 Measuring Selection Bias

A fundamental need for addressing the proposed requirements is the ability to measure potential selection bias. In the following sections we describe such a metric, and discuss specific considerations for analyzing and visualizing selection bias in the context of hierarchical data.

5.1 Selection Bias Metric

In order to detect selection bias, we must be able to quantify shifts in variable (attribute and event) distributions between cohorts. In a manner similar to adaptive contextualization [16, 17], we use the Hellinger distance [34, 43]. However, we compute the distance at each level in the hierarchy for aggregate values, as discussed in Section 3.1, rather than only considering each variable independently. The Hellinger distance is an established statistical metric that quantifies the similarity between two probability distributions. Its discrete form computes a distance between two discrete probability distributions $P = (p_1, ..., p_n)$ and $Q = (q_1, ..., q_n)$, where n is the number of possible values for the variable (e.g. $n = 2$ for a binary variable), as follows:

$$H(P,Q) = \sqrt{\frac{1}{2}\sum_{i=1}^{n}(\sqrt{p_i} - \sqrt{q_i})^2} \quad (1)$$

Regardless of the value of n, $H = 0$ when P and Q are identical, and $H = 1$ when P and Q are maximally different. This characteristic makes the distance comparable when applied to heterogeneous variable types, and the discrete form of H can be applied to categorical, ordinal, and ratio (with binning to discretize the distribution) variables.

When comparing two cohorts c_i and c_j, H is computed for all m dimensions d in the dataset. These individual distances are used in the detailed cohort comparison visualizations (Sections 6.3-6.5). However, since there can be thousands of individual distances, it is difficult to map data at this granularity onto the cohort provenance tree (Section 6.1) for detecting potential selection bias. We therefore compute the average Hellinger distance between two cohorts c_i and c_j as:

$$H_{avg}(c_i, c_j) = \frac{1}{m}\sum_{k=1}^{m} H(d_{ik}, d_{jk}) \quad (2)$$

This univariate metric is used in the cohort provenance tree as an indication of potential selection bias. The distances for each individual variable can then be investigated in more depth via the detailed cohort comparison techniques.

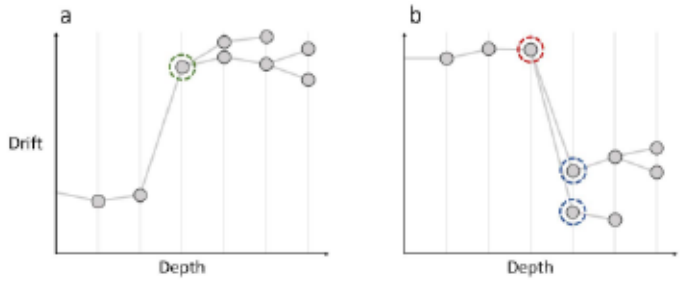

Fig. 4. Examples of changing drift in a section of a hierarchy. Each dimension is a circle, with x-position the depth in the hierarchy and y-position the amount of drift. Links indicate parent-child relationships. (a) The green node does not have the highest drift, but does *differ* the most from its parent, indicating an area of the hierarchy where drift has increased. (b) Similarly, the circled red and blue nodes indicate an area where drift has decreased.

5.2 Selection Bias in Hierarchical Data

When comparing the distributions drifts between two cohorts for every dimension, it is important to convey which dimensions have drifted the most (measured by Equation 1), as has been considered in previous work [16, 17]. However, looking at each of these individual univariate distances in isolation does not provide information on *where in the hierarchy areas of drift have been introduced*, nor on *drift that emerges at higher levels of aggregation*, where variations in coding—e.g., many small drifts in different kinds of heart failure that could add up to a large drift in heart failure at a higher level of representation—could normally cause drift to be overlooked (R5). In addition to areas where drift increases (Figure 4a), areas where drift decreases (Figure 4b) may also help define areas where drift has aggregated, by indicating smaller drifts that may have accumulated.

In general, patterns of drift through a hierarchy can be varied, but areas where the amount of drift *changes* substantially can be suggestive of informative portions of the hierarchy with respect to detecting the presence of selection bias. In order to identify such areas, when comparing two cohorts we explicitly compute the difference in drift value—the drift gradient—between each child dimension and parent dimension in the hierarchy. Based on the selection bias distance metric (Equation 1), we define the drift gradient between a child dimension d_c and its parent d_p for two cohorts c_i and c_j as:

$$\Delta H(d_c, d_p) = H(d_{ic}, d_{jc}) - H(d_{ip}, d_{jp}) \quad (3)$$

To capture (1) areas of the hierarchy with large increases in drift and (2) areas of the hierarchy with large decreases in drift, we define a gradient-based saliency criterion for a dimension d_i, given a single parent dimension d_p, and a set of child dimensions $D_c = \{d_{cj}\}$ (empty for leaf nodes):

$$S(d_i) = \Delta H(d_p, d_i) \geq t_s \vee \exists d_{cj} \in D_c \; -\Delta H(d_i, d_{cj}) \leq -t_s\,, \quad (4)$$

where t_s is a user-specified threshold. This equation detects whether a given node has increased (Figure 4a, green) or any of its children have

decreased (Figure 4b, red), by an amount $\geq t_s$. The user can adjust t_s to determine the degree of aggregation used for the hierarchical visualization methods described in Sections 6.3 and 6.4.

5.3 Constraints

Typically, the dimensions included in the filter constraints used to define cohorts will exhibit the greatest amount of drift. For example, if a user derives one cohort from another by constraining by *Gender = Female*, the *Gender* dimension would, by design, be expected to exhibit a very large drift in distribution. However, when tracking selection bias, the goal is to convey the magnitude of *unintended side effects*.

Previous work has dealt with this issue by simply excluding all constrained dimensions from consideration when tracking or visualizing selection bias [16, 17]. However, when filtering using hierarchically related dimensions, this naive approach is not sufficient. In particular, large drifts in distribution can often occur for the descendants of a constrained dimension. Moreover, it can be assumed that the drift in descendant dimensions is expected by the user. If not, the user could have filtered using dimensions at a lower and more specific level in the hierarchy. We therefore exclude from the calculation of H_{avg} both: (1) dimensions explicitly referenced by a constraint, and (2) any descendants of the explicitly referenced dimensions.

In contrast, when *visualizing* these metrics in detail, excluding constrained dimensions is not desirable because the drift in distributions in their descendant dimensions may be informative to users. Rather than removing constrained dimensions from our hierarchical visualizations, we instead implement two design choices: (1) indicating visually which dimensions are constrained, and (2) normalizing scales to reduce the extent to which constrained variables and their descendants dominate the visualization. For (1), in all detailed cohort comparison visualizations (Sections 6.3-6.5), constraints are indicated with a diamond symbol ♦. For (2), all color scales are normalized to the maximum distance for non-constrained and non-constrained-descendant dimensions. In this manner, the constraints remain part of the visualization—enabling the user to see variations in drift between constrained descendants—while still enabling the user to effectively find non-constrained dimensions that may be subject to selection bias.

5.4 Baseline and Focus

Selection bias occurs when a cohort is selected in such a way that proper randomization (of the non-constraining dimensions in the data) is not conducted, making the cohort no-longer representative of the larger population intended to be analyzed. This larger population can be thought of as a *baseline* against which cohorts can be compared to check for potential selection bias. During a given analysis, it may be useful to explore different baselines for comparison. It is therefore necessary to support a *flexible* approach for choosing the current baseline. In the cohort provenance tree (Section 6.1), the initial queried dataset serves as the baseline by default, however the user can select any created cohort to serve as the baseline against which all other cohorts are compared.

In addition, to enable more in-depth comparison of any given cohort against the baseline, we adopt the concept of a *focus* cohort. By default the focus cohort is the most recently created, however the user can select any cohort to serve as the focus. The visualizations supporting in-depth comparison of the potential selection bias between the focus and baseline cohorts are described in Sections 6.3-6.6.

6 Visualization Design and Implementation

The visual interfaces for the selection bias tracking and cohort comparison features in *Cadence* are motivated by the design requirements from Section 4 and build upon the selection bias measures presented in Section 5. The Cohort Tree panel (Figure 3a) contains the cohort provenance and selection bias tracking visualizations, which address R1, R2, and R3. The Cohort Comparison panel (Figure 3, b-d) enables in-depth comparison of the current baseline and focus cohorts, addressing R4 and R5. It consists of a cohort overlap visualization (Section 6.2), three tabbed views for in-depth exploration of the degree of drift across all dimensions in the dataset (Sections 6.3-6.5), and a detailed univariate comparison view that provides data-type-dependent visualizations to compare the distributions for a single dimension (Section 6.6).

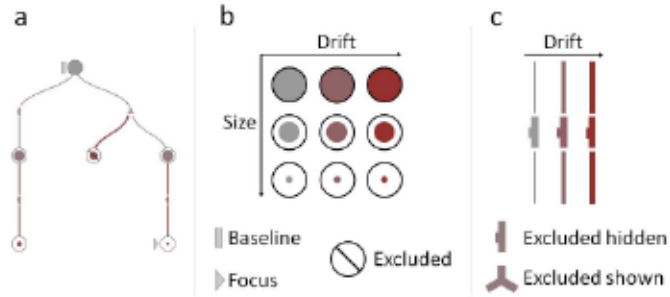

Fig. 5. Cohort provenance visualization and iconography. (a) Each cohort and filter operation is shown as a node-link tree diagram. (b) Cohort glyph inner circle area is proportional to cohort size. Drift is encoded by a grey-red color map. The current baseline and focus cohorts are indicated by rectangular and triangular icons. Excluded cohorts that have been made visible are indicated by a diagonal slash. (c) Link color and thickness, and filter glyph color, indicate the amount of drift introduced at that filter step. Glyph appearance changes to indicate excluded cohort visibility.

6.1 Cohort Provenance Tree

To address R1 (Section 4), we show the provenance of each cohort created by the user via the *Cadence* interface in a node-link tree diagram (Figure 5). Thisdesign is influenced by CONSORT diagrams (Figure 2), which are familiar to many medical experts in the target user population for *Cadence*. A tidy-tree [35] provides a compact layout while showing each step in the cohort-creation process at a distinct level. Each node in the tree represents a cohort, with links representing dependencies (e.g., new cohort B was derived from previous cohort A).

Nodes are represented by circular glyphs of unit size. The glyphs include an inner circle with area proportional to the number of patients in the cohort (Figure 5b). The inner circles are color-coded based on H_{avg} (Equation 2), as computed by comparing the glyph's cohort to the current baseline. This design supports R2.

The links between cohorts include glyphs representing the filter operations used to derive each link's corresponding new cohort. The filter glyphs and the links themselves are both color-mapped by ΔH_{avg}, the difference in H_{avg} between the two cohorts that the link connects. In this way, the edges visualize the amount of drift introduced directly by the link's corresponding filter operation. The encoding is shown in Figure 5c. Mousing over the filter glyph displays a tooltip with information about the constraints applied at that step in the cohort creation process, as well as the amount of drift introduced in response. This design supports R3.

Critically, each filter operation creates both a set of included patients (those matching the filter criteria), and a set of excluded patients (those not matching the criteria). By default only the included cohort is shown. However, the user can show any excluded cohort via a context menu available for each filter glyph. This feature can help users understand the effect of a filter (as evidenced by the specification of excluded groups within the CONSORT flow diagram). Excluded cohorts are depicted with a diagonal slash across the cohort glyph. Moreover, a filter glyph's appearance changes to indicate that the excluded cohort is being shown (Figure 5c).

The cohort tree also indicates the currently selected baseline and focus cohorts. The baseline is indicated with a rectangular icon to the left of the baseline cohort glyph, and the focus with a triangular icon (Figure 5b). By default, the first cohort used at the start of a visual analysis session is marked as the baseline. Meanwhile, the focus is updated (by default) to reference each new cohort as it is created. However, users are able to interactively select any cohort in the tree at any time to serve as the baseline or focus through a popup context menu

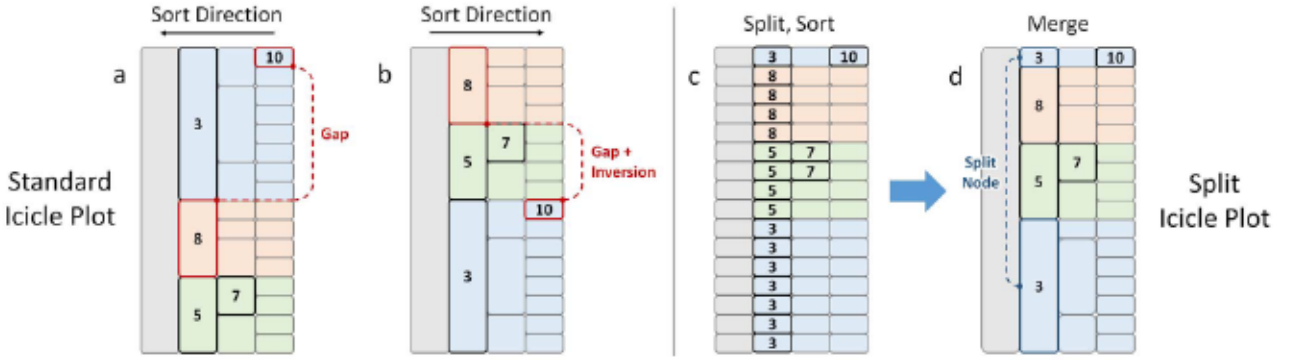

Fig. 6. Sorting by descending order in an icicle plot. The highest 5 values are displayed. With a standard icicle plot (left), recursive sorting strategies can be used to sort (a) from leaf to parent or (b) from parent to leaf. However, nodes may not be correctly ordered, and nodes with high values can become "buried" far from the top, resulting in gaps and inversions. With the split icicle plot layout (right), (c) the path from each leaf node to the root is first separated, and then sorted based on the maximum value along that path. (d) Adjacent nodes are then merged back together. For a given node, all paths above it will contain a value greater than or equal to its value, avoiding issues of gaps and inversions. However, some nodes may remain split to achieve this guarantee.

Fig. 7. The cohort overlap visualization compares the focus and baseline cohorts, showing the relative sizes of the cohorts and proportion of individuals that belong to both. This includes cases where (a) the focus is a subset of the baseline, (b) the two cohorts partially overlap, and (c) the cohorts are disjoint.

available for each cohort glyph. This functionality provides control over which cohorts are the baseline and focus used for the detailed cohort comparison visualizations described in the remainder of this section.

6.2 Cohort Overlap

The first element of the Cohort Comparison panel, which enables users to compare the baseline and focus cohorts, is the cohort overlap visualization. This view depicts the proportion of patients that are members of the baseline and focus cohorts. Unlike previous work with a linear provenance model, in which the focus was restricted by definition to be a subset of the baseline (e.g. [16, 17]), our tree-based approach enables a more flexible set of comparisons. As a result, there are three possible overlap relationships to be visualized. Figure 7 shows examples of these three conditions: (a) one cohort (typically the focus) is a subset of the other (typically the baseline), (b) there is partial overlap, and (c) the two cohorts are disjoint. This visualization conveys the type of overlap the two cohorts have, the relative size of the cohorts, and the proportion of each cohort that falls within the overlapping subset.

6.3 Split Icicle Plot

A key element of the visual design for comparing cohorts is to help users identify where in the dimension hierarchy the largest drifts in distribution have occurred. Given the large hierarchies in our data (Table 1), our initial attempts to visualize this information employed tree maps [41] and icicle plots [27], which are both space-filling techniques with compact representations. However, despite their compact representations, the number of dimensions in the hierarchies made them too big to fit on screen without over-plotting issues.

Attempting to remedy this problem, we first implemented a scrollable icicle plot, which revealed another problem: the inability of icicle plots (and other space-filling hierarchical representations) to enforce a strict ordering of dimensions by value. Icicle plots partition space into a rectangular node for each dimension, with each node placed adjacent to its parent along one axis and sized along the other axis with a length proportional to the number of descendant leaf nodes. Recursive strategies for ordering sibling nodes are possible (e.g., Figure 6, a and b). However, nodes can become "buried" by the hierarchical structure of the visualization, leading to *gaps* and *inversions*. Figure 6a shows how a leaf-to-root sorting strategy can produce a large gap between the largest (10) and 2nd largest (8) nodes by value. Similarly, Figure 6b shows how a root-to-leaf sorting strategy can produce both an inversion and a gap between nodes that are neighbors in sort order.

This issue is problematic for visualizing drift in large hierarchies for two reasons: (1) ordering is an important cue for understanding which dimensions have shifted the most, and it is useful for users to be able to scan from top-to bottom to identify areas of potential selection bias, and (2) when using a scrollable interface to avoid over-plotting, users will need to scroll to find dimensions with large amounts of drift if they are not ordered with largest drift first.

6.3.1 Algorithm and Example

To address these issues, we developed the split icicle plot (Figure 6, c and d). The split icicle plot algorithm proceeds in three stages: *split*, *sort*, and *merge*. The hierarchical data are first *split* into paths from each leaf to the root. The paths are then *sorted* by the maximum value along each path. Adjacent nodes are then *merged* back together. The result is a layout in which, for a given node in the visualization, all paths to the root above it will contain a value greater than or equal to its value, avoiding issues of gaps or inversions. As a tradeoff however, some nodes may be "split" to achieve this guarantee (Figure 6d).

Figure 8 shows an actual example of the split icicle plot visualizing differences in ICD-10-CM diagnoses between two cohorts. In addition to sorting the leaf-to-root paths in descending order by maximum drift (placing largest drift values at the top), the drift for each individual variable in the hierarchy is encoded using a grey-red color map, scaled to the non-constrained variable with the most drift. The one constrained variable in this example is indicated with a ♦ symbol, as explained in Section 5.3. Split nodes are indicated by dashed lines that join the split sections on mouseover. In this example, a parent node of the constraint has been split, enabling a variable that would otherwise have been buried below this node to be *extricated* and revealed as the most distant variable that is not an ancestor of the constraint. Split nodes "break" the strict hierarchical layout traditionally imposed by icicle plots, which could potentially hinder interpretation. However in practice the node splits tend to form subgroups of children with similar drift values which aids the user's comprehension of the visualization.

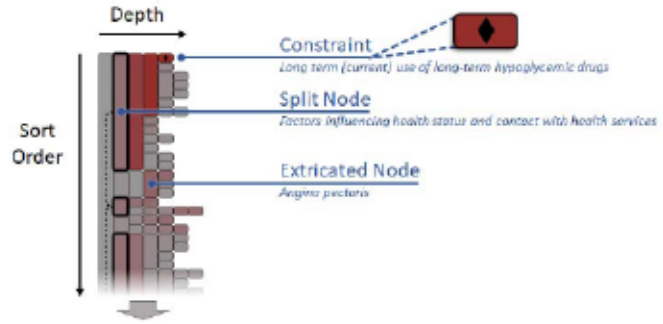

Fig. 8. Split icicle plot visualizing drift between two cohorts in the ICD-10-CM hierarchy. Drift is encoded with a grey-red color map, and constraints are indicated with a ♦ symbol. Split nodes are indicated by dashed lines joining the split sections of the node on mouseover. A split ancestor of the constraint enables an otherwise buried node with high drift to be *extricated*.

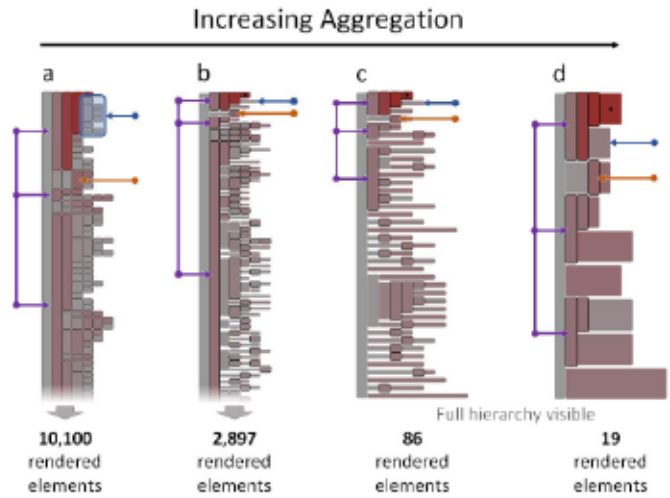

Fig. 9. Hierarchical aggregation using gradient-based saliency. Nodes that do not meet the saliency criterion (Equation 4) are merged into groups. Nodes that do meet the criterion are outlined in black. The blue arrows indicate a group of nodes that is merged from (a) to (b), maintained from (b) to (c), and then merged with another group from (c) to (d). The orange arrows indicate the same *extricated* node from Figure 8, which is maintained from (a-d), as is the split node from the same figure, indicated in purple.

6.3.2 Hierarchical Aggregation

The split icicle plot solves issues related to over-plotting and ordering, and provides an effective visualization of drift in hierarchical data. However, two issues remain when this approach is applied to the large hierarchies found in our application: (1) drawing individual nodes for each dimension in a large hierarchy can hinder performance during interaction, and (2) the large size of the visualization and the use of scrolling (to help overcome over-plotting) can make it difficult for users to obtain a broad overview of the data. To address these issues we have developed a hierarchical aggregation technique with a user-controlled level of aggregation to simplify the visual representation (Figure 9).

The aggregation algorithm relies on gradient-based saliency as defined in Equation 4. Beginning with a split, sorted, and unmerged icicle plot (Figure 6c), we have developed two aggregation methods: *breadth-first*, and *depth-first*. Figure 10 demonstrates both methods. Breadth-first aggregation merges adjacent nodes at the same level in the hierarchy if they either (1) share a common salient descendant, or (2) have no salient descendants. These merged groups are propagated down the tree until a salient node or leaf node is encountered, at which point new groups are created. In constrast, depth-first aggregation merges

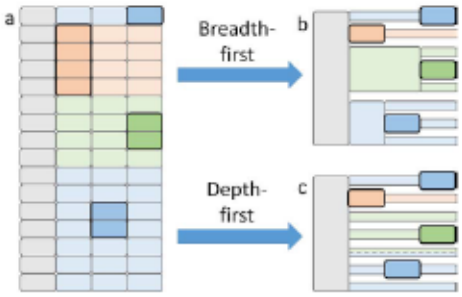

Fig. 10. Two hierarchical aggregation methods for split icicle plots. Given (a) a split, sorted, and unmerged icicle plot with salient nodes outlined in black, (b) breadth-first aggregation prioritizes the merging of adjacent non-salient nodes within the same level of the hierarchy. This preserves much of the hierarchical structure. (c) Depth-first aggregation, which can be more efficient for deep hierarchies, prioritizes the merging of non-salient nodes along each path from the root to a leaf.

nodes along each path from the root until either (1) a salient node is encountered, at which point new paths are created for any children, or (b) a leaf node is encountered. Any groups that share the same starting or ending point are then merged together. In practice, breadth-first tends to preserve more of the hierarchical structure, as adjacent nodes for the same dimension are merged. In contrast, depth-first is more likely to split nodes for the same dimension, but can be more efficient in aggregating hierarchies with large depths. The examples used in this paper all use breadth-first aggregation.

Figure 11 shows an example of the split icicle plot with hierarchical aggregation. Salient nodes are drawn with black outlines (Figure 11a), whereas merged groups are drawn with reduced heights to further emphasize the salient nodes (Figure 11b). Each node or group is colored by drift with a grey-red color map scaled to the maximum drift for a non-constrained dimension. For groups, the color is determined by the maximum drift in the group to indicate areas where large drift may have occurred that did not meet the current saliency definition, such as a chain of relatively small increases in drift that combine to produce a large total drift. The user can inspect the contents of each group with the cursor, which will display a standard icicle plot of the nodes contained within the group (Figure 11c). The user can select a node from within the expanded group to manually define the corresponding dimension as salient. In response, the overall layout will be updated accordingly.

6.4 Hierarchical Dot Plot

One drawback of the split icicle plot visualization is that it depends on color to encode drift, which can make it difficult to distinguish smaller value differences. We therefore developed a hierarchical dot plot visualization as an alternative that uses position to encode drift (Figure 1d).

As with the split icicle plot, hierarchical aggregation is used to reduce over-plotting and increase rendering performance using the same gradient-based saliency method. The visual design is similar to that of Figure 4. Salient nodes are rendered as dots, with x-position representing the depth of the node in the hierarchy, and y-position the amount of drift. The size of the node is proportional to $|\Delta H(p,e)|$, and $\Delta H(p,e)$ is mapped to a blue-grey-red color map, with blue for negative values and red for positive values. The user can highlight any visible node via mouseover to show links to all of its ancestors and descendants. Nodes below the saliency threshold are aggregated and displayed as a background heat-map. Any heat-map cell can be expanded to reveal the nodes inside, and those interior nodes can themselves be selected. By iteratively selecting ancestor/descendant nodes, the user can interactively explore the hierarchy (addressing R4 and R5). This dot plot design provides better visual fidelity for drift values when compared to the split icicle plot, but the overall hierarchical structure is less apparent.

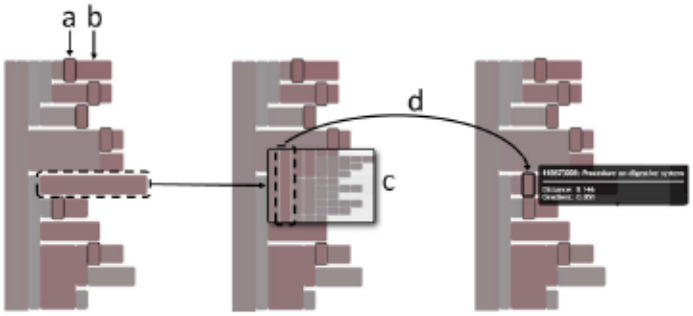

Fig. 11. Split icicle plots with hierarchical aggregation. (a) Salient nodes are drawn with black outlines. (b) Merged groups are drawn with reduced heights. A grey-red color map indicates drift. For groups, the maximum drift in the group is used. Placing the cursor over a group will cause it to expand (c), showing a standard icicle plot of the nodes in that group. The user can select a nodes in the expanded group, and the layout will be updated with that node manually defined as a salient node (d).

6.5 List View

In addition to the hierarchical visualization techniques described above, we also provide a list view that shows a basic table of dimensions in descending order by drift. The table includes a bar chart representation of the drift values as shown in Figure 1c. This view ignores the hierarchical relationships in the data, but has been provided as a simple, easy-to-use format that is familiar to users.

6.6 Variable Distribution

The in-depth cohort comparison visualizations listed above enable the user to see which dimensions in the hierarchy have drifted, and the magnitudes of those drifts. However, they do not convey *how* the dimensions differ between the two cohorts. For example, filtering for patients with a SNOMED-CT code for *Procedure on hip* reveals a large drift in distribution for the ICD-10-CM code for *Pain in hip*. In this case, it can be assumed that the cohort with a hip procedure also has a higher incidence of hip pain, but such relationships may not always be so obvious. We therefore enable the user to select any attribute or event to see a detailed visualization of the distributions of the selected dimension for each cohort.

Each of the detailed cohort comparison visualizations (split icicle plot, hierarchical dot plot, list) enable users to select any dimension from within the visualization by direct selection, or via explicit search with a search box. The dimension selection is linked across all views, enabling users to pivot between the different visualizations.

Moreover, once a dimension has been selected, a data-type-dependent visualization of the distributions for that dimension in both the baseline and focus cohorts is displayed at the bottom of the cohort comparison panel. As shown in Figure 1e, this distribution comparison view supports bar charts for categorical data, a histogram for numeric data, and horizontal binary bar charts for binary variables. In all cases, the proportion of each value is shown to enable comparison between cohorts of different sizes. Interactive highlighting shows contextual information on individual data values, such as the specific number of patients. These visualizations further R4 by enabling detailed distribution comparisons for individual variables.

7 Example Use Case and Domain Expert Interviews

To demonstrate the benefits of the approaches outlined in this paper, we present an example use case for the selection bias tracking capabilities within *Cadence*. We also report a thematic analysis of qualitative feedback gathered through interviews with medical domain experts.

7.1 Example Use Case

Figure 12 shows a series of screenshots from *Cadence* during an analysis of a cohort of 1,732 patients who were discharged from a hospital after having been diagnosed with some form of pain. The queried data includes one year's worth of medical event data (SNOMED-CT and ICD-10-CM codes) prior to the pain diagnosis, as well as data from the period between the pain diagnosis and the hospitalization.

Using the temporal event analysis portion of *Cadence* (faded in Figure 12), the user has created a number of cohorts based on filters that reference various attributes and events surfaced by the interface. Examining the cohort tree, the user notices that the current focus cohort of 227 patients, filtered by *Obesity and other hyperalimentation*, seems to have drifted considerably farther than other cohorts from the baseline (in this case the initial query result). The high drift is indicated by the red color of the path to the cohort and the cohort glyph itself in Figure 12a. The user decides to investigate the differences between the focus and baseline cohorts. After adjusting the aggregation level for the split icicle plot (Figure 12b), the user identifies areas of the dimension hierarchy that are contributing the most to the high drift. As expected, the constrained *Obesity...* variable, indicated by a ♦ symbol, has drifted the most. Moving on, the user highlights the ICD-10-CM event with the highest drift in a different branch of the hierarchy and finds it to be *Sleep apnea*. Suspecting that patients in the *Obesity...* cohort probably have a higher prevalence of *Sleep apnea*, the user checks this assumption with the selected variable distribution visualization (Figure 12c). This reveals that the focus *Obesity...* cohort contains a 59% incidence rate (133 of 227 patients) of *Sleep apnea*, whereas the baseline cohort contains just 29% (496 of 1,732 patients). Without selection bias tracking and detailed cohort comparison, the user would likely not notice that *Sleep apnea* is more prevalent in the focus dataset than in the baseline. Having access to this information, however, enables the user to incorporate this information into any subsequent analyses of the data. For example, they could control for sleep disorders when studying the effectiveness of a particular medication for obese patients to ensure analytical validity for the target population.

Looking back at the split icicle plot, the user notices that some descendant nodes of the constrained variable also have large drift values (red), whereas others have a very low drift value (grey). The user selects the constrained variable and switches to the hierarchical dot plot view (Figure 12d) to inspect further. The user notices that four descendants maintains relatively high drift, whereas the drift for five other descendants drop precipitously. This pattern may indicate different subgroups of obesity warranting additional study.

7.2 Domain Expert Interviews

To better understand the strengths and weaknesses of the selection bias tracking capabilities outlined in this paper, we conducted a set of semi-structured interviews with three medical experts.

All three participants were health-focused researchers with data analysis experience. One was a medical doctor with both clinical and research responsibilities, whereas both of the others were PhD-level researchers. All three participants were full-time university employees with experience using the i2b2 system [32], an NIH-funded cohort selection platform that has been deployed to support data-driven health research at a large number of research universities.

Each participant met individually with two study moderators for a one hour session. Participants were first introduced to *Cadence* and provided a demonstration of cohort selection tasks using the system. After the demonstration, participants were asked questions in a semi-structured interview format. The interviews triggered additional interactions with the visualization system.

7.2.1 Thematic Analysis of Interview Findings

The three domain experts provided feedback about many aspects of the *Cadence* visual analytics system. We summarize their comments with a thematic analysis of the interview findings, focusing on the feedback that is most relevant to the selection bias tracking and cohort comparison capabilities.

General feedback. General comments about the approach were typically very positive, and participants agreed that the system provided many clear benefits. One participant commented that it enables you to "instantly generate insights" and is a powerful "hypothesis generating application." Other feedback included, "I really like your design," this

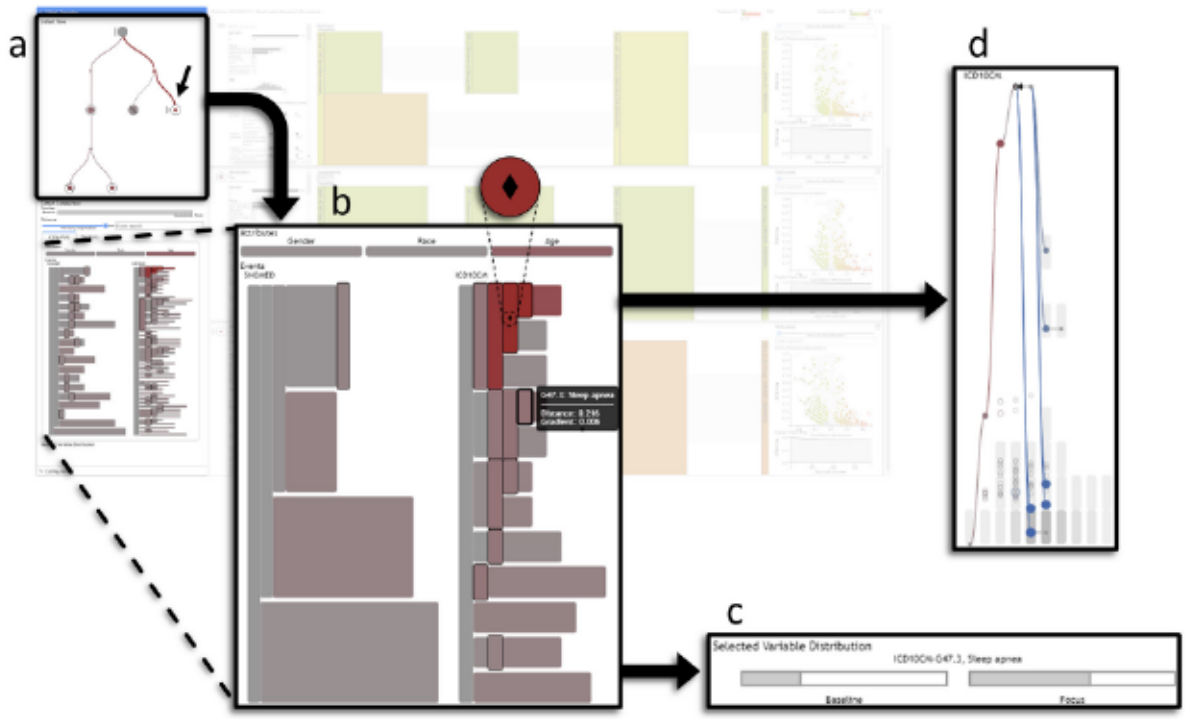

Fig. 12. The example use case described in Section 7.1. The temporal event sequence visualization, which is not the focus of this paper, is faded.

is a "really powerful analytical tool," and "this is very useful...for cohort studies." Although feedback was positive, the participants did acknowledge the complexity of the system, and the need for training. One participant noted that it seemed complicated at first glance, but that "it made sense" after seeing it demonstrated. Another participant noted that some degree of complexity and training should be expected: "i2b2 was complicated at first," but that such tools "are for skilled users" and require training. To sum up, one participant stated that this is "very cool, but there is a learning curve."

Benefits of cohort tree and selection bias tracking. Participants agreed that selection bias, and other issues related to the validity of analytical findings, such as "fishing expeditions," are important, and germane to their work, e.g. "that's something I'm really concerned about." However they did raise the issue of how to indicate "how much [selection bias] is too much?" Participants also agreed that the cohort tree is a useful method for keeping track of the provenance of created cohorts. One participant stated that "it helps you not get lost. It's breadcrumbs." Another stated that "if I get lost here, I can go back [to the cohort tree] to orient myself." Some suggestions for improvements included providing the ability to toggle baseline and focus cohort tooltips–instead of showing them only on mouseover–to serve as a reminder when interacting with the other parts of the visualization. This feature has since been implemented.

Benefits of detailed cohort comparison. Participants saw benefits and drawbacks of all three detailed cohort comparison methods. Regarding the hierarchical visualizations, one participant remarked, "this is good, but takes extra effort to see what it means," due to the need to mouseover to obtain detailed information on the event type. Another participant stated that "All those views are very useful." Regarding a particular selected variable distribution visualization, one participant highlighted a discovery during the interview session: when we "saw the spike in the age distribution, [I asked] why? The system lets you look into it right away." One participant wished to "make the comparison view larger," suggesting that it would be useful to be able to switch between cohort selection and cohort analysis modes.

Alternative visualization designs. More specific feedback on the alternative visualization designs for the detailed cohort comparison visualizations included one participant's view that the icicle plot "was good for comparison" and that they intuitively understood the use of color. However, for the dot plot, the same participant suggested that "when it is crowded, hard to see." In general, participants found that the hierarchical visualizations were "good for exploring," whereas the list was "good for finding specifics."

8 Conclusion

This paper presents a new selection bias tracking and high-dimensional comparison system for exploratory cohort selection. It overcomes limitations in prior work by (1) using a tree-based cohort provenance system and (2) leveraging hierarchical relationships between data dimensions to both better communicate areas of drift in variable distributions between cohorts, and enable user-controlled aggregation for improved visualization.

The approach is demonstrated by integration with a medical temporal event sequence visual analytics tool, and implemented via: (1) a cohort provenance tree visualization that enables the user to keep track of created cohorts and indicates when selection bias may have occurred, (2) a novel split icicle plot that improves the ability to order nodes by value in a hierarchical visualization, highlights areas of potential selection bias, and provides user-controlled aggregation, and (3) a set of complementary visualizations, including a hierarchical dot plot, list view, and detailed variable distribution visualizations, to examine the differences between two cohorts in detail.

Feedback from domain expert interviews indicated that selection bias tracking in visual analytics tools is an important capability, and that further work in this vein is warranted. Specific areas for future work include: (1) providing a reference for potentially "dangerous" amounts of selection bias that is comparable across particular analyses, (2) exploring techniques to correct for selection bias implemented within the system, and (3) investigating the selection bias tracking and cohort comparison techniques for applications beyond the medical domain. In addition, more in-depth evaluations of the *Cadence* system, and its individual components, are planned.

Acknowledgments

The research reported in this article was supported in part by a grant from the National Science Foundation (#1704018).

REFERENCES

[1] ICD - ICD-10-CM - International Classification of Diseases, Tenth Revision, Clinical Modification, July 2018.

[2] D. Baur, B. Lee, and S. Carpendale. TouchWave: Kinetic Multi-touch Manipulation for Hierarchical Stacked Graphs. In *Proceedings of the 2012 ACM International Conference on Interactive Tabletops and Surfaces*, ITS '12, pp. 255–264. ACM, New York, NY, USA, 2012. event-place: Cambridge, Massachusetts, USA. doi: 10.1145/2396636.2396675

[3] D. Borland, W. Wang, and D. Gotz. Contextual Visualization. *IEEE Computer Graphics and Applications*, 38:17–23, 2018. doi: 10.1109/MCG.2018.2874782

[4] S. Bremm, T. v. Landesberger, M. Heß, T. Schreck, P. Weil, and K. Hamacherk. Interactive visual comparison of multiple trees. In *2011 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pp. 31–40, Oct. 2011. doi: 10.1109/VAST.2011.6102439

[5] I. Cho, R. Wesslen, A. Karduni, S. Santhanam, S. Shaikh, and W. Dou. The Anchoring Effect in Decision-Making with Visual Analytics. In *2017 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pp. 116–126, Oct. 2017. doi: 10.1109/VAST.2017.8585665

[6] E. Cuenca, A. Sallaberry, F. Y. Wang, and P. Poncelet. MultiStream: A Multiresolution Streamgraph Approach to Explore Hierarchical Time Series. *IEEE Transactions on Visualization and Computer Graphics*, 24(12):3160–3173, Dec. 2018. doi: 10.1109/TVCG.2018.2796591

[7] W. Cui, S. Liu, Z. Wu, and H. Wei. How Hierarchical Topics Evolve in Large Text Corpora. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):2281–2290, Dec. 2014. doi: 10.1109/TVCG.2014.2346433

[8] E. Dimara, S. Franconeri, C. Plaisant, A. Bezerianos, and P. Dragicevic. A Task-based Taxonomy of Cognitive Biases for Information Visualization. *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–1, 2018. doi: 10.1109/TVCG.2018.2872577

[9] G. Ellis, ed. *Cognitive Biases in Visualizations*. Springer International Publishing, 2018.

[10] N. Elmqvist and J.-D. Fekete. Hierarchical Aggregation for Information Visualization: Overview, Techniques, and Design Guidelines. *IEEE Transactions on Visualization and Computer Graphics*, 16(3):439–454, May 2010. doi: 10.1109/TVCG.2009.84

[11] U. Fayyad, G. G. Grinstein, and A. Wierse. *Information Visualization in Data Mining and Knowledge Discovery*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1st ed., 2001.

[12] G. Gigerenzer. Why Heuristics Work. *Perspectives on Psychological Science*, 3(1):20–29, Jan. 2008. doi: 10.1111/j.1745-6916.2008.00058.x

[13] M. Gleicher, D. Albers, R. Walker, I. Jusufi, C. D. Hansen, and J. C. Roberts. Visual comparison for information visualization. *Information Visualization*, 10(4):289–309, Oct. 2011. doi: 10.1177/1473871611416549

[14] M. Glueck, A. Gvozdik, F. Chevalier, A. Khan, M. Brudno, and D. Wigdor. PhenoStacks: Cross-Sectional Cohort Phenotype Comparison Visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):191–200, Jan. 2017. doi: 10.1109/TVCG.2016.2598469

[15] D. Gotz and H. Stavropoulos. DecisionFlow: Visual Analytics for High-Dimensional Temporal Event Sequence Data. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):1783–1792, 2014. doi: 10.1109/TVCG.2014.2346682

[16] D. Gotz, S. Sun, and N. Cao. Adaptive Contextualization: Combating Bias During High-Dimensional Visualization and Data Selection. In *IUI*, 2016. doi: 10.1145/2856767.2856779

[17] D. Gotz, S. Sun, N. Cao, R. Kundu, and A.-M. Meyer. Adaptive Contextualization Methods for Combating Selection Bias During High-Dimensional Visualization. *ACM Trans. Interact. Intell. Syst.*, 7(4):17:1–17:23, Nov. 2017. doi: 10.1145/3009973

[18] D. Gotz, J. Zhang, W. Wang, J. Shrestha, and D. Borland. Visual Analysis of High-Dimensional Event Sequence Data via Dynamic Hierarchical Aggregation. *IEEE Transactions on Visualization and Computer Graphics (Proceedings of Visual Analytics Science and Technology 2019)*, 26(1), 2020.

[19] S. Hahn, J. Trümper, D. Moritz, and J. Döllner. Visualization of varying hierarchies by stable layout of voronoi treemaps. In *2014 International Conference on Information Visualization Theory and Applications (IVAPP)*, pp. 50–58, Jan. 2014.

[20] D. R. Harris and D. W. Henderson. i2b2t2: Unlocking Visualization for Clinical Research. *AMIA Joint Summits on Translational Science proceedings. AMIA Joint Summits on Translational Science*, 2016:98–104, 2016.

[21] R. v. Hees and J. Hage. Stable Voronoi-based visualizations for software quality monitoring. In *2015 IEEE 3rd Working Conference on Software Visualization (VISSOFT)*, pp. 6–15, Sept. 2015. doi: 10.1109/VISSOFT.2015.7332410

[22] M. A. Hernán, S. Hernández-Díaz, and J. M. Robins. A structural approach to selection bias. *Epidemiology (Cambridge, Mass.)*, 15(5):615–625, Sept. 2004.

[23] P. Isenberg and S. Carpendale. Interactive Tree Comparison for Co-located Collaborative Information Visualization. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1232–1239, Nov. 2007. doi: 10.1109/TVCG.2007.70568

[24] B. Johnson and B. Shneiderman. Tree-maps: a space-filling approach to the visualization of hierarchical information structures. In *Proceeding Visualization '91*, pp. 284–291, Oct. 1991. doi: 10.1109/VISUAL.1991.175815

[25] D. Keim, H. Qu, and K.-L. Ma. Big-Data Visualization. *IEEE Comput. Graph. Appl.*, 33(4):20–21, July 2013. doi: 10.1109/MCG.2013.54

[26] W. Köpp and T. Weinkauf. Temporal Treemaps: Static Visualization of Evolving Trees. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):534–543, Jan. 2019. doi: 10.1109/TVCG.2018.2865265

[27] J. B. Kruskal and J. M. Landwehr. Icicle Plots: Better Displays for Hierarchical Clustering. *The American Statistician*, 37(2):162–168, 1983. doi: 10.2307/2685881

[28] S. Liu, D. Maljovec, B. Wang, P. Bremer, and V. Pascucci. Visualizing High-Dimensional Data: Advances in the Past Decade. *IEEE Transactions on Visualization and Computer Graphics*, 23(3):1249–1268, Mar. 2017. doi: 10.1109/TVCG.2016.2640960

[29] J. Lukasczyk, G. Weber, R. Maciejewski, C. Garth, and H. Leitte. Nested Tracking Graphs. *Comput. Graph. Forum*, 36(3):12–22, June 2017. doi: 10.1111/cgf.13164

[30] S. Malik, F. Du, M. Monroe, E. Onukwugha, C. Plaisant, and B. Shneiderman. Cohort Comparison of Event Sequences with Balanced Integration of Visual Analytics and Statistics. In *Proceedings of the 20th International Conference on Intelligent User Interfaces - IUI '15*, pp. 38–49. ACM Press, Atlanta, Georgia, USA, 2015. doi: 10.1145/2678025.2701407

[31] D. Moher, S. Hopewell, K. F. Schulz, V. Montori, P. C. Gøtzsche, P. J. Devereaux, D. Elbourne, M. Egger, and D. G. Altman. CONSORT 2010 Explanation and Elaboration: updated guidelines for reporting parallel group randomised trials. *BMJ*, 340:c869, Mar. 2010. doi: 10.1136/bmj.c869

[32] S. N. Murphy, G. Weber, M. Mendis, V. Gainer, H. C. Chueh, S. Churchill, and I. Kohane. Serving the enterprise and beyond with informatics for integrating biology and the bedside (i2b2). *Journal of the American Medical Informatics Association*, 17(2):124–130, Mar. 2010. doi: 10.1136/jamia.2009.000893

[33] E. A. Nohr and Z. Liew. How to investigate and adjust for selection bias in cohort studies. *Acta Obstetricia Et Gynecologica Scandinavica*, 97(4):407–416, Apr. 2018. doi: 10.1111/aogs.13319

[34] D. Pollard. *A User's Guide to Measure Theoretic Probability*. Cambridge University Press, 2002.

[35] E. M. Reingold and J. S. Tilford. Tidier Drawings of Trees. *IEEE Transactions on Software Engineering*, SE-7(2):223–228, Mar. 1981. doi: 10.1109/TSE.1981.234519

[36] A. Rind, T. D. Wang, W. Aigner, S. Miksch, K. Wongsuphasawat, C. Plaisant, and B. Shneiderman. Interactive information visualization to explore and query electronic health records. *Foundations and Trends® in Human–Computer Interaction*, 5(3):207–298, Feb. 2013. doi: 10.1561/1100000039

[37] L. Rokach and O. Maimon. Clustering methods. In O. Maimon and L. Rokach, eds., *Data Mining and Knowledge Discovery Handbook*, pp. 321–352. Springer, Boston, MA, 2005.

[38] H. Schulz, S. Hadlak, and H. Schumann. The Design Space of Implicit Hierarchy Visualization: A Survey. *IEEE Transactions on Visualization and Computer Graphics*, 17(4):393–411, Apr. 2011. doi: 10.1109/TVCG.2010.79

[39] K. F. Schulz, D. G. Altman, and D. Moher. CONSORT 2010 Statement: updated guidelines for reporting parallel group randomised trials. *BMJ*, 340:c332, Mar. 2010. doi: 10.1136/bmj.c332

[40] K. F. Schulz, I. Chalmers, R. J. Hayes, and D. G. Altman. Empirical evidence of bias. Dimensions of methodological quality associated with estimates of treatment effects in controlled trials. *JAMA*, 273(5):408–412, Feb. 1995.

[41] B. Shneiderman. Tree visualization with tree-maps: 2-d space-filling approach. *ACM Transactions on Graphics*, 11(1):92–99, Jan. 1992. doi: 10.1145/102377.115768

[42] B. Shneiderman. The eyes have it: a task by data type taxonomy for information visualizations. In , *IEEE Symposium on Visual Languages, 1996. Proceedings*, pp. 336–343, Sept. 1996. doi: 10.1109/VL.1996.545307

[43] D. G. Simpson. Minimum Hellinger Distance Estimation for the Analysis of Count Data. *Journal of the American Statistical Association*, 82(399):802–807, 1987. doi: 10.2307/2288789

[44] M. Sondag, B. Speckmann, and K. Verbeek. Stable Treemaps via Local Moves. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):729–738, Jan. 2018. doi: 10.1109/TVCG.2017.2745140

[45] K. A. Spackman, K. E. Campbell, and R. A. Côté. SNOMED RT: a reference terminology for health care. *Proceedings of the AMIA Annual Fall Symposium*, pp. 640–644, 1997.

[46] A. Sud, D. Fisher, and H. Lee. Fast Dynamic Voronoi Treemaps. In *2010 International Symposium on Voronoi Diagrams in Science and Engineering*, pp. 85–94, June 2010. doi: 10.1109/ISVD.2010.16

[47] D. L. Swofford, G. J. Olsen, P. J. Waddell, and D. M. Hillis. Phylogenetic inference. In D. M. Hillis, C. Moritz, and B. K. Mable, eds., *Molecular Systematics*, pp. 407–514. Sinauer, Sunderland, MA, 2nd ed., 1996.

[48] S. Tak and A. Cockburn. Enhanced Spatial Stability with Hilbert and Moore Treemaps. *IEEE Transactions on Visualization and Computer Graphics*, 19(1):141–148, Jan. 2013. doi: 10.1109/TVCG.2012.108

[49] J. J. Thomas and K. A. Cook, eds. *Illuminating the Path: The Research and Development Agenda for Visual Analytics*. National Visualization and Analytics Center, 2005.

[50] A. Törner, P. Dickman, A.-S. Duberg, S. Kristinsson, O. Landgren, M. Björkholm, and Å. Svensson. A method to visualize and adjust for selection bias in prevalent cohort studies. *American Journal of Epidemiology*, 174(8):969–976, Oct. 2011. doi: 10.1093/aje/kwr211

[51] A. Törner, A.-S. Duberg, P. Dickman, and A. Svensson. A proposed method to adjust for selection bias in cohort studies. *American Journal of Epidemiology*, 171(5):602–608, Mar. 2010. doi: 10.1093/aje/kwp432

[52] Y. Tu and H. Shen. Visualizing Changes of Hierarchical Data using Treemaps. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1286–1293, Nov. 2007. doi: 10.1109/TVCG.2007.70529

[53] A. C. Valdez, M. Ziefle, and M. Sedlmair. Priming and Anchoring Effects in Visualization. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):584–594, Jan. 2018. doi: 10.1109/TVCG.2017.2744138

[54] E. Wall, L. M. Blaha, L. Franklin, and A. Endert. Warning, Bias May Occur: A Proposed Approach to Detecting Cognitive Bias in Interactive Visual Analytics. *2017 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pp. 104–115, 2017. doi: 10.1109/VAST.2017.8585669

[55] E. Wall, L. M. Blaha, C. L. Paul, K. Cook, and A. Endert. Four Perspectives on Human Bias in Visual Analytics. In G. Ellis, ed., *Cognitive Biases in Visualizations*, pp. 29–42. Springer International Publishing, Cham, 2018. doi: 10.1007/978-3-319-95831-6_3

[56] M. Wattenberg and J. Kriss. Designing for social data analysis. *IEEE Transactions on Visualization and Computer Graphics*, 12(4):549–557, July 2006. doi: 10.1109/TVCG.2006.65

[57] M. Zinsmaier, U. Brandes, O. Deussen, and H. Strobelt. Interactive Level-of-Detail Rendering of Large Graphs. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2486–2495, Dec. 2012. doi: 10.1109/TVCG.2012.238