Avoiding Drill-down Fallacy with STORYBOARD: Assisted and Accelerated Data Exploration Through Data Subsets

ABSTRACT

As datasets continue to grow in size and complexity, exploring multi-dimensional datasets remain challenging for analysts. A common operation during this exploration is drill-down—understanding the behavior of data subsets by adding filters. While widely used, in the absence of careful attention towards confounding factors, drill-downs could lead to inductive fallacies. Specifically, an analyst may end up being "deceived" into thinking that a deviation in trend is attributable to a local change, when in fact it is more generalized phenomenon; we call this the drill-down fallacy. A naive solution to prevent the drill-down fallacy is to explore all sub-paths of a drill-down path, which quickly becomes infeasible on large and complex datasets with many attributes. In this paper, we present STORYBOARD, an accelerated visual data exploration tool that guides the analyst to the key insights in a dataset avoiding drill-down fallacies. Our user study results show that our tool can help analysts discover interesting visualizations, understand attribute importance, and predict unseen visualizations better than other summarization baselines.

KEYWORDS

exploratory data analysis, visualization recommendation.

1 INTRODUCTION

Visual data exploration is the *de facto* first step in understanding a multi-dimensional dataset. This exploration serves a variety of purposes: identifying trends and patterns, spotting outliers and anomalies, and verifying hypotheses. However, as datasets grow in size and complexity, visual data exploration ends up becoming challenging. For example, to identify patterns that merit further investigation, an analyst may need to explore different subsets of the data to determine when and where certain patterns occur. Generating and examining visualizations for this space of data subsets (which grows exponentially in number of attributes) is a major bottleneck in exploration.

One way of navigating this combinatorial space is to perform drill-downs on the lattice of data subsets. For example, a campaign manager who is interested in understanding the voting patterns across different demographics (say, race, gender, social class) using the 2016 US election exit polls [**?**] may first generate a bar chart for the entire population, where the x-axis shows the election candidates and the y-axis the percentage of votes for each of these candidates. In Figure 1, the visualization at the top of the lattice corresponds to this overall population. They may then drill down to specific demographics of interest, say gender-based demographics, by generating bar charts for female voters (as shown in the second visualization at the second row of Figure 1).

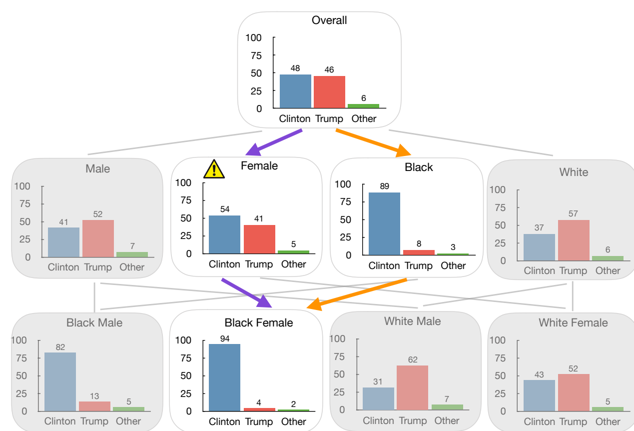

Figure 1: Example data subset lattice illustrating the drill-down fallacy along the purple path as opposed to the informative orange path.

There are three challenges associated with performing manual drill downs in this manner. First, it is often not clear which attributes to drill-down on. Analysts may use their intuition for this, but such arbitrary exploration may lead to large swaths of the lattice being ignored. Second, the path an analyst takes may lead to behavior that is not very surprising (insightful). For example, an analyst going from `Black` visualization to `Black Female` visualization in Figure 1 may find the two distributions to be similar, and therefore not surprising. This may well end up being wasted effort on the part of the analyst. Third, an analyst may encounter the "drill-down fallacy": as shown in Figure 1, an analyst can arrive at the `Black Female` visualization by going through the purple or orange drill-down path. An analyst who followed the purple path may be surprised at how drastically the `Black Female` voting behavior differs from that of `Female`. This behavior is no longer surprising if the analyst had gone down the orange path, where the proper reference (i.e., the vote distribution for `Black`) explains the vote distribution for `Black Female`. That is, even though the vote distribution for `Black Female` is very different from that of `Female`, it can be explained by a more general phenomenon or "root cause"—the vote distribution for `Black` community. Attributing an overspecific cause to an effect, while ignoring the actual cause, not only leads to less interpretable explanations for the observed visualizations, but can also be detrimental to decision-making (in this case, it could lead to a misallocation of campaign funds).

The aforementioned example demonstrates the *drill-down fallacy*—incomplete insights that result from potentially confounding factors not explored along a drill-down path. In particular, while performing a drill-down on a randomly selected path, an analyst may find a "local difference" in trends, without being aware of the more "general phenomenon" that could explain the trend of interest. In this case, they lack the proper parent reference (visualization) that explains the behavior of the visualization of interest. Thus, they are at risk of falling prey to the drill-down fallacy. A naive solution to avoid this fallacy is to explore all potential drill-down paths. Unfortunately, this approach does not scale with the increasing number of factors in the drill-down path.

In this paper, we present a visual data exploration tool that addresses the aforementioned challenges of exploration through three principles: (i) **Safety.** ensure that proper (informative) references are present to avoid the drill-down fallacy, (ii) **Saliency.** identify interesting visualizations that convey new information (insights), and (iii) **Summarization.** succinctly convey the key insights. To facilitate safety, we develop a notion of *informativeness*—the capability of a reference visualizations to explain the visualization of interest. To facilitate saliency, we characterize the notion of *interestingness*—the difference between a visualization and its informative reference in terms of underlying data distribution. Finally, to facilitate summarization, we embrace *collectiveness*—a connected network of visualizations that collectively offer informative insights. Based on these three principles, our tool, STORYBOARD, automatically identifies a network of visualizations that succinctly conveys the key informative insights in a dataset. Our user study results demonstrate that our tool can guide an analyst towards meaningful insights for a variety of tasks. The contribution of this paper include:

- Introducing the novel concept of *informativeness* that helps users identify meaningful insights that arise from something *actually interesting* about the data (instead of confounding variables),
- Designing a system that automatically identifies a network of visualizations that succinctly conveys the key informative insights in a dataset,
- Demonstrating the efficacy of our system through a user study evaluation on how well users can retrieve interesting insights, judge the importance of attributes, and predict unseen visualizations against two other summarization baselines.

2 TOWARDS INFORMATIVE EXPLORATION

In this section, we first describe how analysts manually explore the space of data subsets using drill-downs. We then introduce the three design principles aimed at addressing the current challenges of manual exploration, and automatically guide analysts to the key informative insights.

Manual Exploration via Drill-Downs

During visual data exploration, an analyst may need to explore different subsets of the data, which form a combinatorial *lattice*. Figure 1 shows a partial lattice for the 2016 US election polls. The lattice contains the overall visualization with no filter at the first level, all visualizations with a single filter at the second level, all visualizations with two filters at third level, and so on. Analysts explore such a combinatorial lattice hierarchically, by generating and examining visualizations with increasing levels of aggregation. In particular, analysts perform drill-downs to access data subsets at lower levels by adding one filter at a time, and for each such data subset visualize their metrics of interest. Further, as analysts perform drill-downs, they use the most recent visualization in the drill-down path (known as the 'parent') as a reference to establish what they expect to see in the new visualization (known as the 'child'). For example in Figure 1, the visualizations `Female` and `Black` are the *parents* of the `Black Female` visualization, explored along the purple and orange path respectively.

As exemplified by the purple path in Figure 1, during drill-downs analysts may be misguided by improper references that exhibits high deviation locally, in particular when other potential parents (i.e., parents not explored in the drill-down path) that could explain the more general phenomenon are overlooked. We refer to this misinterpretation as *drill-down fallacy*, since the fallacy arises from the inductive nature of the drill-down operation.

Three Elements of Informative Exploration

Our goal is to enable users to discover the key informative insights in a dataset avoiding drill-down fallacies. We argue in favor of three essential principles for finding such meaningful insights—namely safety, saliency, and summarization. We adopt these principles to develop a visual exploration tool that automatically selects visualizations that collectively convey the key informative insights of a multidimensional dataset.

Safety. To prevent the drill-down fallacy, we concentrate on *safety*—using informative references for discovering insights. We identify informative references in a drill-down context by modeling the *informativeness* of an observed parent in characterizing the child visualization. An observed parent is *informative* if its data distribution closely follows the child visualization's data distribution, since the parent serves as a proper reference that helps analysts form an accurate mental picture of what to expect from the child visualization. Specifically, we formulate the informativeness of an observed parent V_i^j for a visualization V_i as the similarity between their data distributions measured using a distance function $D(V_i, V_i^j)$. The most informative parents V_i^* for visualization V_i are the ones whose data distributions are most similar to V_i.

$$V_i^* = \underset{V_i^j}{argmin}\, D(V_i, V_i^j) \tag{1}$$

We regard a visualization as informative if its distance falls within a user-defined threshold $\theta\%$ close to its most informative parent:

$$V_i^{*,\theta} = \{V_i^j : \frac{D(V_i, V_i^*)}{D(V_i, V_i^j)} \geq \theta\} \tag{2}$$

For example in Figure 1, while both visualization `Black` and `Female` visualizations are considered parents of the `Black Female` visualization, only the `Black` visualization are considered the informative parent of the `Black Female` population, for any values of $\theta \geq 11\%$ via the Euclidean distance metricAditya: CITE?. Note that, our proposed system can work with different distance metrics such as cosine similarity and earth mover's distanceAditya: CITE. Without loss of generality, we chose to use Euclidean distance metric for the remainder of our paper.

Saliency. To discover insights, we emphasize *saliency*— identifying interesting visualizations that convey new information. In general, a visualization is interesting if its underlying data distribution differs from that of its parents, and thus offers new information or unexpected insights. The notion of such interestingness have been explored in past work [7, 12, 18], particularly through the usage of distance-based metrics. However, unlike past works, we concentrate on *informative interestingness*, where the goal is to identify interesting visualizations in presence of informative references. Specifically, to model the interestingness of an visualization V_i in the context of its *informative* parent V_i^j, we characterize the deviation between their data distributions using a distance function $D(V_i, V_i^j)$. Further, to address the effect of subpopulation size, we multiply the distance $D(V_i, V_i^j)$ between an informative parent V_i^j and a child visualization V_i by the ratio of their sizes $U(V_i, V_i^j) = \frac{|V_i|}{|V_i^j|} \cdot D(V_i, V_i^j)$.

Summarization. To succinctly convey insights, we concentrate on *summarization*—identifying a group of visualizations that collectively contain informative insights. Since our aim is to identify a unified narrative, instead of discrete insights, we enforce that any selected visualization must have its proper, informative parent be present in the dashboard. Specifically, we identify a set of k connected visualizations that collectively maximize the proposed utility $U(V_i, V_i^j)$, and thus succinctly convey informative insights. This problem of finding a connected subgraph in the lattice that has the maximum combined edge utility is known as the maximum-weight connected subgraph problem [3] and is known to be NP-Complete [15]. We design several sub-optimal algorithms to solve this problem efficiently.

3 SYSTEM

In this section, we present our system, STORYBOARD, by first providing a high-level overview of the underlying algorithms, and then describing the user interaction mechanisms.

Algorithms

We discuss the algorithms used for generating the visualization lattice, and then present a high-level overview of our algorithms that traverse the lattice to select the k-connected maximum-weighted subgraph.

Lattice Generation: Our system supports two variants of traversal algorithms based on the lattice generation procedure—offline variants that first generate the complete lattice and then work towards identifying the maximum utility solution, and online variants that incrementally generate the lattice and simultaneously identify the solution. The offline variants are appropriate for datasets with a small number of low-cardinality attributes, where we can generate the entire lattice in a reasonable time; whereas the online variants are appropriate for datasets with large number of high-cardinality attributes, where we incrementally generate a partial lattice.

Lattice Traversal: Given the materialized lattice, the objective of the traversal algorithm is to find the connected subgraph in the lattice that has maximum combined edge utility.

The frontier greedy algorithm obtains a list of candidate nodes known as the *frontier* nodes, which encompasses all neighbors of nodes in the existing subgraph solution. Any of the nodes in the frontier can be added to the current solution since their informative parent is present in the solution. To obtain the frontier nodes, the algorithm scans and adds all children of leaf nodes of the current dashboard as part of the frontier. In the online version, it additionally checks for each child whether its informative parent is present in the current dashboard. At each step, our algorithm greedily picks the node with the maximum utility amongst the frontier nodes to add to the current solution, and updates the frontier accordingly.

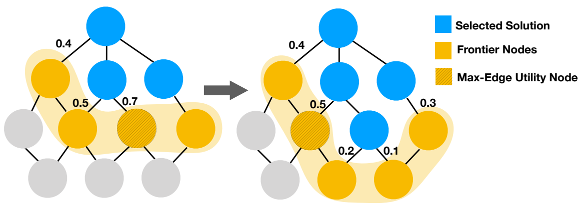

Figure 2: Example illustrating how the frontier greedy algorithm incrementally build up the solution by selecting the maximal-edge utility node from the frontier at every step.

Algorithm 1 Frontier Greedy Algorithm

procedure PICKVISUALIZATIONS(k,lattice)
    dashboard ← { $V_{overall}$ }
    while |dashboard| < k **do**
        frontier ← getFrontier(dashboard,lattice)
        maxNode ← getMaxUtilityNode(frontier)
        dashboard ← dashboard ∪ {maxNode}
    return dashboard

User Interaction

Given the selected visualizations, we render them in a dashboard, as shown in Figure 3. Users can inspect the visualization dashboard through panning and zooming with navigation buttons, mouse clicks, and keyboard bindings. Users can also select the x and y axes of interest, aggregation function, and optional system parameter settings to generate a dashboard.

After browsing through visualizations in the dashboard, users may be interested in getting more information about the visualization at a particular node. STORYBOARD allows users to request a new dashboard centered on a chosen visualization of interest as the new starting point (or equivalently, the root of the lattice) for analysis. As shown in Figure 3a, the analyst starts with a 7-visualization dashboard on the Police Stop dataset [16]. The dataset contains records of vehicle and pedestrian stops from law enforcement departments in Connecticut, dated from 2013 to 2015. The analyst learns that for the drivers who had contraband found in the vehicle, the arrest rate for drivers who are 60 and over is surprisingly higher than usual, whereas for Asian drivers the arrest rate is lower. In addition, she is also interested in learning more about the other factor that contribute to high arrest rate: duration=30+min. In Figure 3b, she clicks on the corresponding visualization and requests for 2 additional visualizations. Upon seeing the updated dashboard in Figure 3c, she learns that any visualization that involves the duration=30+min filter is likely to result in high ticketing and arrest rates. This implies that if a police stop lasts more than 30 minutes, the outcome would more or less be the same, independent of other factors such as driver's race or age. To generate the expanded dashboard, STORYBOARD uses the same models and algorithms as before, except the root node is now set as the selected visualization, rather than the overall visualization. This node expansion capability is motivated by the idea of *iterative view refinement* in other visual analytics systems, which is essential for users to iterate on and explore different hypotheses [9, 20].

4 USER STUDY EVALUATION

Methods

We evaluate the utility of STORYBOARD by performing a between-subject user study focusing on addressing the following research questions:

- RQ1: How *interesting* are the visualizations in the dashboard perceived subjectively by the users?
- RQ2: How well do the dashboards *summarize* the relative importance of different attributes within a given dataset?
- RQ3: How *informative* are the visualizations in the dashboard at providing users with an accurate understanding for unseen child visualizations?

We recruited 18 participants with prior experience in working with data. Participants included undergraduate and graduate students, researchers, and data scientists, with 1 to 14 years of data analysis experience (average = 5.61). There were 8 female and 10 male participants. No participants reported prior experience in working with the two datasets used in the study. Participants were randomly assigned two of the three dashboards with k=10 visualizations generated by following conditions.

STORYBOARD: The dashboards for this condition are generated by the frontier greedy algorithm (described in Section 3) and displayed in a hierarchical layout (as seen in Figure 3). In order to establish a fair comparison with the two other conditions, we deactivated the interactive node expansion capabilities.

BFS: Starting from the visualization of the entire population, *k* visualizations is selected level-wise, traversing down the subset lattice, adding the visualizations at the first level with 1-filter combination one at a time, proceeding with the 2-, 3-, and so on, until *k* visualizations have been added to the dashboard. This baseline is designed to simulate the dashboard generated by a meticulous analyst who exhaustively inspects all possible visualization combinations from the top-down. The chosen visualizations are displayed in a 5x2 table layout in the traversed order.

CLUSTER: K-Means clustering is performed on the dataset with *k* clusters, corresponding to *k*, the number of visualizations to be shown in the dashboard. For each representative cluster, we select the visualization with the least number of filter conditions for interpretability[1] and display them in a 5x2

[1] Due to this requirement, the overall visualization is guaranteed to be picked as one of the displayed visualizations.

Figure 3: a) Overview of the STORYBOARD interface for the Police Stop dataset. Users can select x and y axes of interest, as well as a choice of an aggregation function. Default values are set for system related parameters such as the number of visualizations to show in the dashboard (k), iceberg condition for pruning (δ), and informative parent criterion (θ), which can be adjusted by the users via the sliders if needed. b) User clicks on the duration=30+min visualization to request 2 additional visualization. c) A preview of the added portion of the resulting dashboard is shown.

table layout. This baseline is designed to showcase a diverse set of pattern distributions within the dataset.

Each participant were assigned two different conditions on two different datasets. The ordering of each condition was randomized to prevent confounding learning effects.The study began with a 5 minute tutorial using dashboards generated from the Titanic dataset [1] for each condition. To prevent bias across conditions, participants were not provided an explanation of how the dashboard was generated and why the visualizations were arranged in a particular way. Then, participants proceeded onto the Police Stop dataset, as described in Section 3.We generated dashboards of bar chart visualizations with x-axis as the stop outcome (i.e., whether the police stop resulted in a ticket, warning, or arrest) and y-axis as the percentage of police stops that led to each outcome.

The second dataset in the study is the Autism dataset [8], which includes the result of autism spectrum disorder screening for 704 adults. The attributes in the dataset are binary responses to 10 diagnostic questions that are part of the screening process. This dataset serves as a data-agnostic condition, since there was no descriptions of the questions or answer labels provided to the user. We generate dashboard visualizations based on whether the participant is diagnosed with autism or not.

Participants were given some time to read through a worksheet containing the descriptions of the data attributes. Then, they were given an attention check question where they were given a verbal description of the visualization filter and asked about the distributions for the corresponding visualization in the dashboard. After understanding the dataset and chart schema, participants were asked to accomplish the following tasks in the prescribed order below:

Retrieval: Participants were asked to talk aloud as they interpreted the visualizations in the dashboard and mark each visualization as either interesting, not interesting, or leave it as unselected. This task was intended to measure how interesting are the selected visualizations to participants (RQ1).

Attribute Ranking: Participants were given a sheet of paper with all the attributes listed and asked to rank the attributes in order of importance in contributing to a particular outcome (e.g., factors leading to an arrest or autism diagnosis). Participants were allowed to assign equal ranks to more than one attributes or skip attributes that they were unable to infer their importance for. Attribute ranking tasks are common in feature selection and other data science tasks. The goal of this task is to measure how well participants understand the relative importance of each attribute in contributing towards an outcome (RQ2).

Prediction: Participants were given a separate worksheet and asked to sketch an estimate for a visualization that is not present in the dashboard. For every condition, the visualization to be estimated contained 2 filter combinations, with exactly one parent present in the given dashboard. After making the prediction, participants were shown the actual data distribution and asked to rate on a Likert scale of 10 how surprising the result was (where 1 is not surprising and 10 is very surprising). The prediction task measures how accurate participants are at predicting an unseen visualization, estimating how well they understood key informative insights that influences other distributions from the dataset(RQ3).

We repeat the same study procedure described above for the Autism dataset. At the end of the study, we asked two open-ended questions regarding the stories and insights that they have learned and what they like or dislike about each dashboard. On average, the study lasted around 48 minutes.

Quantitative Results

Retrieval (RQ1): Using the click-stream data logged from the user study, we recorded whether a participant marked a visualization in the dashboard as interesting, not interesting, or left the visualization unselected. Table 1 summarizes counts of visualizations marked as interesting or not interesting aggregated across conditions. We also normalize the interesting count by the total number of selected visualizations to account for variations in how some participants select more visualizations than others. The result indicate that participants who used STORYBOARD found more visualizations that they found interesting compared to the BFS and CLUSTER condition.

Condition	Storyboard	BFS	Cluster
Interesting	66	61	51
Not Interesting	10	20	22
Interesting (Normalized)	0.87	0.75	0.7

Table 1: Total count of visualizations marked as interesting or not interesting across the different algorithms. STORYBOARD leads to more visualizations marked as interesting and fewer visualizations marked as uninteresting.

Attribute Ranking (RQ2): To determine the attribute importance for a dataset, we computed the Cramer's V statistics between attributes to be ranked and the attributes of interest. Cramer's V test makes use of the chi-square statistics to determine the strength of association between attributes [**?**]. Using the ground truth based on ranks determined by Cramer's V statistics, we compute the normalized discounted cumulative gain (NDCG@k) of each participant's ranking average over all tasks[2], as detailed in Table 2. We see that STORYBOARD performs better than clustering in both cases. Since clustering seeks visualizations that exhibit diversity in the shape of the data distribution, it results in visualizations with many filter combinations, which is hard to interpret without appropriate context to compare against. BFS performs better than STORYBOARD in the Police dataset, but not in the Autism dataset. BFS may have performed better than STORYBOARD in the Police dataset for a combination of two reasons: 1) since BFS exhaustively displays all attributes sequentially, for the Police dataset it happened to select several of the important attributes (related to contraband and search) to display in the first 10 visualizations and 2) some participants had priors on the data attribute which influenced their ranking. However, with a budget of k=10, only visualizations regarding diagnostic questions 1-5 fit in the dashboard for the Autism dataset, so the poor ranking behavior comes from the fact that the BFS generated dashboard failed to display the important attributes (questions 6 and 9) given the limited budget. This demonstrates BFS's lack of providing a guarantee especially when exhaustive exploration has a limit (e.g., time or attention of analyst). In general, our results indicate that using STORYBOARD, users gain a better understanding of attribute influence and importance.

Dataset	STORYBOARD	BFS	CLUSTER
Police	0.63	0.84	0.45
Autism	0.50	0.24	0.30

Table 2: NDCG@10 scores for the attribute ranking task.

Prediction (RQ3): In order to compare how accurate analysts' mental model for various —drill down paths for investigating RQ3, we use the prediction tasks as a proxy for how informative the selected visualizations in the dashboard are. Informativeness is measured by how accurate participants are at predicting an unseen visualization. In order to measure how accurate participants' decisions are, we computed the Euclidean distance between their predicted distributions and ground truth data distributions. As shown in Figure 4, the predictions made by using STORYBOARD is closer to the actual distribution compared to the baselines. This indicates that users are able to more accurately reason about how unseen data would behave with STORYBOARD. Aditya: Why was cluster better in one setting but not the other?

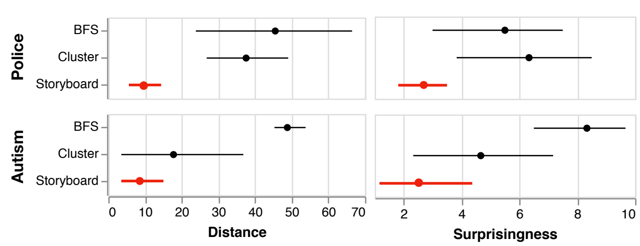

Figure 4: Left: Euclidean distance between predicted and ground truth. In general, predictions made using STORYBOARD is closer to the ground truth. Right: Surprisingness rating reported by users after seeing the actual visualizations on a Likert scale of 10. STORYBOARD participants had a more accurate mental model of the unseen visualization and therefore reported less surprise than compared to the baseline.

[2]Since participants are asked to examine all attributes, the k for NDCG@k corresponds to total number of attributes in that dataset.

We also compute the variance of participants' predictions across the same task. In this case, low variance implies that any user who reads the dashboard is able to provide consistent predictions, whereas high variance implies that the dashboard did not convey a clear data-driven story that could guide their predictions. So instead, participants relied on different priors or guessing to form their prediction. These trends can be observed in Figure 5, where the prediction variance amongst participants who used STORYBOARD is generally lower than the variance from the baselines.

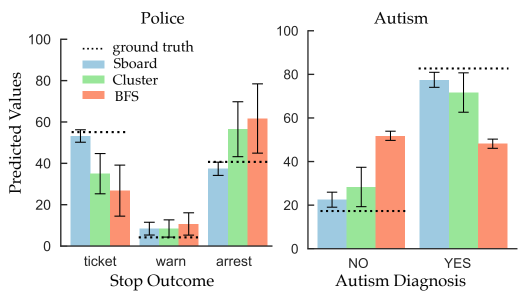

Figure 5: Mean and variance of predicted values. Predictions based on STORYBOARD exhibits lower variance (as indicated by the error bars) and great proximity to the ground truth values (dotted).

5 DISCUSSION

To understand the usefulness of our recommended visualizations, we analyzed the user study transcriptions through an open coding process by two of the authors. For each task in our study, we assigned a binary-valued code to indicate whether or not a participant engaged in that particular task (action or thought process). Table 4 highlights results from thematic coding, discussed in this section. We will use the notation [Participant.DatasetAlgorithm] to refer to a participant engaging with a dashboard created by an algorithm={1,2,3}={STORYBOARD, CLUSTER, BFS} on a dataset ={A,B}={Police, Autism}.

The Choice of Contextual References

As discussed earlier, analysts often make use of related visualizations as *contextual references* to form their expectations on unseen visualizations. The choices of a proper informative parent is essential for ensuring the *safety* of insights derived through drill-downs. To understand how 'safe' dashboards generated from each condition were, we examined the types of visualizations that participants utilized and compared against to form their expectations regarding how other visualizations should look like. In particular, we thematically encoded participant's use of contextual parents based on the verbal explanations that they provided to justify their prediction task responses. Participants can (and often do) make comparisons against more than one type of contextual parent to obtain their prediction. We uncovered four main classes of contextual references, described below using the example visualization `gender=F, race=White, age=21-30` (in order of most to least similar) and illustrated graphically in Figure 6:

(1) Parent : Comparison against a visualization with one filter criterion removed (e.g. `gender=F, race=White`)
(2) Siblings : Comparison against a visualization that share the same parent. In other words, the filter types are the same, but with one criterion that inherit a different value. (e.g. `gender=M, race=White, age=21-30`)
(3) Relatives : Comparison against a visualization that share some common ancestor (excluding overall), but not necessarily the same parent. In other words, these visualizations share at least one common filter type, but with more than one criterion that inherit a different value. (e.g. `gender=F, race=White, age=60+, search conducted=T`)
(4) Overall : Comparison against the distribution that describes the overall population (no filters applied).

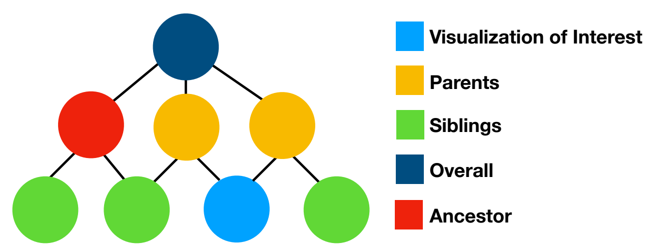

Figure 6: Illustrative example of the different types of contextual reference for a given visualization of interest.

As shown in Table 3, in general, we find that participants make more comparisons in total using STORYBOARD than compared to CLUSTER and BFS. Studying participants' use of contextual references reveals inherent challenges that arise from using the dashboards generated by BFS and CLUSTER. For CLUSTER, participants mainly compared against relatives and the overall. Since CLUSTER optimizes the diversity of shape (distributions) amongst the visualizations, the selected visualizations had up to 4 filters and were disconnected from each other. For this reason, in many cases participants could only rely on relatives and the overall as contextual references. For example, P4.A2 pointed at a 4-filter visualization with extreme values (100% for warning; 0% for arrest and ticket) and indicated how "*a lot of [the visualizations] are far too specific. This is not very helpful. You can't really hypothesize that all people are going to be warned, because it is such a specific category, it might just be one person*". He further explained how he "*would not want to see the intersections* *[(visualizations with many filters)] at first and would want to see all the bases [(univariate summaries)] then dig in from there.*" The lack of informative contextual references in the CLUSTER dashboard is also reflected in how analysts exhibited high variance and deviation in their prediction responses.

Algorithm	Parent	Sibling	Relative	Overall	Total
STORYBD	12	8	0	11	31
CLUSTER	4	0	7	8	19
BFS	0	5	1	8	14

Table 3: Out of 12 participants, the number of participants who made use of each contextual parents across the two datasets. Participant behavior shows a similar trend in individual datasets. STORYBOARD participants made more comparisons in general and against parents compared to the baseline.

For BFS, most comparisons were based on the overall and siblings. Due to the sequential level-wise picking approach, for the BFS dashboards, the overall corresponded to the immediate parent, so they are not explicitly recorded as a parent. While the overall and sibling comparisons can be informative, due to the limited number of visualizations (k), not all first-level visualizations were displayed in the dashboard. These incomplete comparisons can result in flawed reasoning, as observed in the Autism shallow prediction task described earlier. In contrast, for STORYBOARD, almost all users compared against the overall and parents, while some also exploited sibling comparison information to make weaker guesses for less-frequently observed attributes (e.g., using a 2-filter sibling visualization involving `driver_age` to infer another 2-filter visualization `driver_age` with a different parent.)

The Danger of Improper References

While comparisons are essential for data understanding, choosing the wrong contextual reference for comparison could lead to misleading insights. In particular, when a visualization composed of multiple filter conditions is shown in a dashboard created using CLUSTER, 25% of participants had trouble interpreting the meaning of the filter for at least one of the datasets. In contrast, as shown in Figure 4, this confusion only happened once for BFS and none for STORYBOARD. This is due to the fact that CLUSTER dashboards are seemingly random to the users, whereas BFS and STORYBOARD both have a more natural, interpretable ordering. In addition, when examining visualizations with many filters along with extremely-skewed values in one or more bars (bars with 100% or 0%), 4 CLUSTER participants did not realize that charts with multiple filters may have a smaller subpopulation size, echoing our previous concern regarding the danger of small subpopulation sizes. This issue stems from the fact that the contextual reference used for comparison was the overall population, however the unseen parent subpopulation may have behaved very differently. This subpopulation-size fallacy was observed to be more severe for the Autism dataset, where participants had less intuition on the expected attribute behavior. In contrast, 6 of the participants using STORYBOARD explicitly noted that while these extreme-valued visualizations may be interesting, they were less certain due to the unknown subpopulation size and should be investigated further. For example, P1.A1 noted that a visualization with warning=100% caught her eye, "*but I don't know what the N is, maybe it's one person, this makes me a little skeptical, that makes me want to go back to the raw data and look at what is the N and what drives something so drastic?*" Since BFS dashboards only displayed first-level visualizations, participants for BFS did not see such visualizations during the study session, so none of the BFS participants exhibited signs of this fallacy.

Interpretability of Hierarchical Layouts

In the post-study interviews, participants cited hierarchical layout as one of the key reasons why it was easier to follow contextual reference in STORYBOARD. Users were able to easily interpret the meaning of the dashboard through STORYBOARD's hierarchical layout, even though they were never explicitly told what the edge connections between the visualizations meant. For example, P1.A1 stated that "*the hierarchical nature [is] a very natural flow...so when you are comparing, you don't have to be making those comparisons in your head, visually that is very pleasing and easy to follow.*" Likewise, P9 described how the hierarchical layout she saw for the Autism dataset was a lot easier to follow than the Police dataset shown in the table layout for CLUSTER:

> If I had to look at this dataset in the format of the other one, this would be much more difficult. It was pretty hard for me to tell in the other one how to organize the tree, if there was even a tree to be organized. I like this layout much better, I think this layout allows me to approach it in a more meaningful way. I can decide, what do I think matters more: the overall trend? or the super detailed trends? and I know where to look to start, in the other one, every time I go back to it, I would say, where's the top level, where's the second level? I mentally did this. Like when you asked me that first question, it took much longer to find it, because I literally have to put every chart in a space in my head and that took a lot longer than knowing how to look at it.

At the end of the study, some participants who saw table layouts sketched and explained how they would like the layout of the visualizations to be done. Participants expressed that they wanted "groupings" or layouts that arranged visualizations with the same attribute together. Other participants advocate for isolating the overall visualization outside of the dashboard table for facilitating easier comparisons. Both of these provides further motivation for our hierarchical layout and the idea of the collapsed visualizations as described in Section 3.

Since we did not inform participants how the dashboards were generated, it was also interesting to note that some participants thought that the dashboards were hand-picked by a human analyst and described what this person's intentions were (e.g. "*It seems like the researcher who created this dashboard was specifically looking at people of Asian descent and people who are 60 or older.*" [P7.A1]). We encoded this phenomenon by looking at instances where a participant either explicitly referring to a person who picked out the dashboard or implicitly described their intentions through personal pronouns. A total of 5 different participants referred to the dashboard generated by STORYBOARD as generated by a human, whereas there was only 1 participant for CLUSTER and none for BFS made such remarks. At the end of the study, many were surprised to learn that the STORYBOARD dashboard was actually picked out by an algorithm, indicating that STORYBOARD could automatically generate convincing dashboard stories similar to a dashboard that was authored with human intention.

Limitations of STORYBOARD

As described earlier, since the details of how the dashboard was obtained was not explained to the users during the study, some users expressed that they were initially confused by STORYBOARD since not all variables were present in the dashboard. Others also found it confusing that the addition of filters did not always correspond to the same variables. For example, P2.A1 criticized how the dashboard was intentionally selected to be biased:

> I feel like this one, not all the data is here, so we are already telling a story, you are trying to steer the viewer to look at certain things. And the focus seems to be on where the arrest rate is high. You probably could have found other things that led to ticket being high, but you didn't pull those out. You are trying to see if there are other factors that leads to more arrests.

This sentiment is related to participants' desire to perform their own ad-hoc querying alongside the dashboard to inspect other related visualizations for verifying their hypothesis. For example, P7.A1 wanted to inspect all other first-level visualizations for driver's race to assess its influence. P7.A1 expressed that while he had learned many insights from the dashboard, "*the only thing I don't like is I cannot control the types of filter, which is fixed.*" Outside the context of the user study, it is essential to explain how STORYBOARD are picking the visualizations in a easy and interpretable manner to establish a sense of summarization guarantee for the users and help them make better inferences with the dashboard.

As discussed earlier, subpopulation size is important in establishing the significance of a trend observed in a visualization. While subpopulation size is taken into account implicitly in our objective, we should design interfaces that can convey the notion of subpopulation size in our dashboard. Examples include Sankey-like flow diagrams indicating the percentage of the parent population broken down into individual subpopulations or subpopulation size explicitly specified via edge labels.

	STORYBD	CLUSTER	BFS
Difficulty with Interpretation	0	3	1
Misjudged Significance	0	4	1
"Human-like" Dashboard	5	1	0
Number of Insights (Police)	11	8	9
Number of Insights (Autism)	16	6	11

Table 4: Summary of qualitative insights from thematic coding.

6 RELATED WORK

Our work draws from, and improves upon, past research in multidimensional data exploration, fallacies in visual analytics, and visualization storytelling.

Guided Exploration of Multidimensional Data

Given a dataset, tools such as Tableau supports automatic generation of visualizations based on perceptual graphical presentation rules [14, 20]. A more recent body of work automatically selects visualizations based on statistical measures, such as scagnostics and deviation. Given a scatterplot, Anand et al. [4] applies randomized permutation tests to select partitioning variables that reveals interesting small multiples using scagnostics. Given a bar chart, Vartak et al. [18] finds other interesting bar charts that deviate form the input chart using a deviation-based measure. Our work extends the deviation-based measure to formulate user expectation. However, unlike the existing works, we concentrate on informativeness, which enables our systems to prevent the drill-down fallacy.

Preventing Biases and Statistical Fallacies

Visualizations are powerful representations for discovering trends and patterns in a dataset; however, cognitive biases and statistical fallacies could be mislead analysts' interpretation of those patterns [2, 19, 22]. Wall et al. [19] presents six metrics to systematically detect and quantify bias from user interactions in visual analytic systems. These metrics are based on coverage and distribution, which focus on the assessment of the process by which users sample the data space. Alipourfard et al. [2] presents a statistical method to automatically identify Simpson's paradox by comparing statistical trends in the aggregate data to those in the disaggregated subgroups. Zgraggen et al. [22] presents a method to detect presence of the multiple comparisons problem in visual analysis. In this paper, we concentrate on a novel type of fallacy during drill-down exploration that have not been addressed by past work.

Storytelling with Visualization Sequences

Visualizations are often arranged in a sequence to narrate a data-driven story. Existing work on visualization sequences and storytelling have studied the structures of narrative visualizations [11, 17], effects of augmenting exploratory information visualizations with narration [6] and, more recently, ways to automate the creation of visualization sequences [10, 13]. Most of these work have adopted a linear layout (motivated by slidedecks) to present the visualization sequences. Hullman et al. [11] found that most people prefer visualization sequences structured hierarchically based on shared data properties such as levels of aggregation. Kim et al. [13] models relationships between charts by empirically estimating transition (edge) cost between moving from one visualization (node) to another. They find that participants preferred "*starting from the entire data and introducing increasing levels of summarization*". Our work is the first to automatically organize visualizations in a hierarchical layout for summarizing the space of data subsets.

REFERENCES

[1] [n. d.]. Titanic: Machine Learning from Disaster. Kaggle.

[2] Nazanin Alipourfard, Peter G. Fennell, and Kristina Lerman. 2018. Can You Trust the Trend?: Discovering Simpson's Paradoxes in Social Data. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining (WSDM '18)*. ACM, New York, NY, USA, 19–27. https://doi.org/10.1145/3159652.3159684

[3] Ernst Althaus, Markus Blumenstock, Alexej Disterhoft, Andreas Hildebrandt, and Markus Krupp. 2009. Algorithms for the Maximum Weight Connected k-Induced Subgraph Problem. 5573 (2009), 313–321. https://doi.org/10.1007/978-3-642-02026-1

[4] Anushka Anand and Justin Talbot. 2015. Automatic Selection of Partitioning Variables for Small Multiple Displays. 2626, c (2015). https://doi.org/10.1109/TVCG.2015.2467323

[5] Carsten Binnig and Lorenzo De Stefani. 2017. Towards Sustainable Insights or why polygamy is bad for you. *8th Biennial Conference on Innovative Data Systems Research (CIDR '17)* (2017).

[6] Jeremy Boy, Francoise Detinenne, and Jean-Daniel Fekete. 2015. Storytelling in Information Visualizations : Does it Engage Users to Explore Data? *CHI 2015* (2015), 1449–1458.

[7] Michael Correll and Jeffrey Heer. 2016. Surprise! Bayesian Weighting for De-Biasing Thematic Maps. *IEEE Transactions on Visualization and Computer Graphics* 2626, c (2016), 1–1. https://doi.org/10.1109/TVCG.2016.2598618

[8] Fadi Fayez Thabtah. 2017. Autism Screening Adult Data Set. UCI Machine Learning Repository.

[9] Enamul Hoque, Vidya Setlur, Melanie Tory, and Isaac Dykeman. 2017. Applying Pragmatics Principles for Interaction with Visual Analytics. *IEEE Transactions on Visualization and Computer Graphics* c (2017). https://doi.org/10.1109/TVCG.2017.2744684

[10] Jessica Hullman, Steven Drucker, Nathalie Henry Riche, Bongshin Lee, Danyel Fisher, and Eytan Adar. 2013. A deeper understanding of sequence in narrative visualization. *IEEE Transactions on Visualization and Computer Graphics* 19, 12 (2013), 2406–2415. https://doi.org/10.1109/TVCG.2013.119

[11] Jessica Hullman, Robert Kosara, and Heidi Lam. 2017. Finding a Clear Path : Structuring Strategies for Visualization Sequences. 36, 3 (2017).

[12] Laurent Itti and Pierre Baldi. 2009. Bayesian surprise attracts human attention. *Vision Research* 49, 10 (19 May 2009), 1295–1306. https://doi.org/10.1016/j.visres.2008.09.007

[13] Younghoon Kim, Kanit Wongsuphasawat, Jessica Hullman, and Jeffrey Heer. 2017. GraphScape: A Model for Automated Reasoning about Visualization Similarity and Sequencing. *Proc. of ACM CHI 2017* (2017). https://doi.org/10.1145/3025453.3025866

[14] Jock D. Mackinlay, Pat Hanrahan, and Chris Stolte. 2007. Show Me: Automatic presentation for visual analysis. *IEEE Transactions on Visualization and Computer Graphics* 13, 6 (2007), 1137–1144. https://doi.org/10.1109/TVCG.2007.70594

[15] Aditya G. Parameswaran, Hector Garcia-Molina, and Jeffrey D. Ullman. 2010. Evaluating, combining and generalizing recommendations with prerequisites. *Proceedings of the 19th ACM international conference on Information and knowledge management - CIKM '10* (2010), 919. https://doi.org/10.1145/1871437.1871555

[16] E. Pierson, C. Simoiu, J. Overgoor, S. Corbett-Davies, V. Ramachandran, C. Phillips, and S. Goel. 2017. A large-scale analysis of racial disparities in police stops across the United States. https://openpolicing.stanford.edu/data/

[17] Edward Segel and Jeffrey Heer. 2010. Narrative visualization: Telling stories with data. *IEEE Transactions on Visualization and Computer Graphics* 16, 6 (2010), 1139–1148. https://doi.org/10.1109/TVCG.2010.179

[18] Manasi Vartak, Sajjadur Rahman, Samuel Madden, Aditya Parameswaran, and Neoklis Polyzotis. 2015. SEEDB : Efficient Data-Driven Visualization Recommendations to Support Visual Analytics. (2015).

[19] Emily Wall, Leslie M Blaha, Lyndsey Franklin, and Alex Endert. 2017. 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)* (2017). https://www.cc.gatech.edu/{~}ewall9/media/papers/BiasVAST17.pdf

[20] Kanit Wongsuphasawat, Dominik Moritz, Anushka Anand, Jock Mackinlay, Bill Howe, and Jeffrey Heer. 2016. Voyager: Exploratory Analysis via Faceted Browsing of Visualization Recommendations. *IEEE Transactions on Visualization and Computer Graphics* 22, 1 (2016), 649–658. https://doi.org/10.1109/TVCG.2015.2467191

[21] Dong Xin, Jiawei Han, Xiaolei Li, Zheng Shao, and Benjamin W. Wah. 2007. Computing iceberg cubes by top-down and bottom-up integration: The starcubing approach. *IEEE Transactions on Knowledge and Data Engineering* 19, 1 (2007), 111–126. https://doi.org/10.1109/TKDE.2007.250589

[22] Emanuel Zgraggen, Zheguang Zhao, Robert Zeleznik, and Tim Kraska. 2018. Investigating the Effect of the Multiple Comparisons Problem in Visual Analysis. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, Article 479, 12 pages. https://doi.org/10.1145/3173574.3174053