


Charting EDA: Characterizing Interactive Visualization Use in Computational Notebooks with a Mixed-Methods Formalism

Dylan Wootton 

Amy Rae Fox 

Evan Peck 

Arvind Satyanarayan 

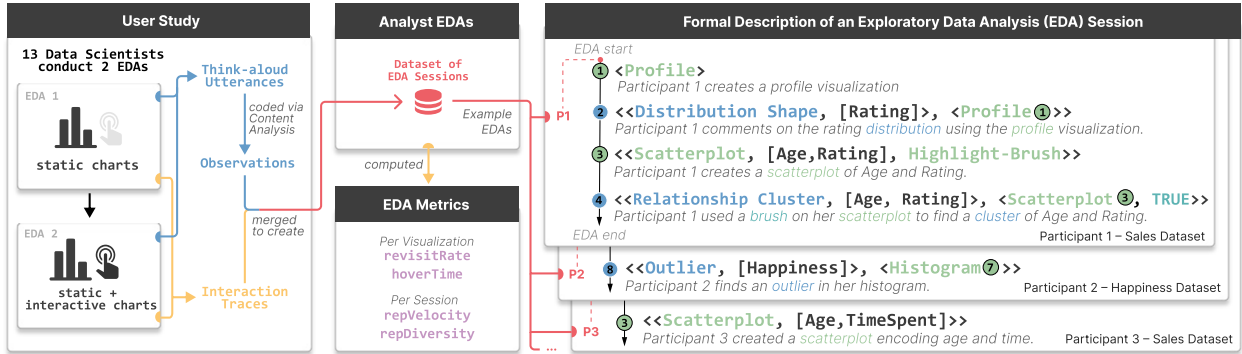


Fig. 1: This study examined EDA practices via mixed methods. Think-aloud and interaction data from EDA sessions were collected and quantified using metrics and formal descriptions. The resulting dataset facilitated analysis of EDA behaviors and strategies.

Abstract—Interactive visualizations are powerful tools for Exploratory Data Analysis (EDA), but how do they affect the observations analysts make about their data? We conducted a qualitative experiment with 13 professional data scientists analyzing two datasets with Jupyter notebooks, collecting a rich dataset of interaction traces and think-aloud utterances. By qualitatively coding participant utterances, we introduce a formalism that describes EDA as a sequence of analysis states, where each state is comprised of either a representation an analyst constructs (e.g., the output of a data frame, an interactive visualization, etc.) or an observation the analyst makes (e.g., about missing data, the relationship between variables, etc.). By applying our formalism to our dataset, we identify that interactive visualizations, on average, lead to earlier and more complex insights about relationships between dataset attributes compared to static visualizations. Moreover, by calculating metrics such as revisit count and representational diversity, we uncover that some representations serve more as “planning aids” during EDA rather than tools strictly for hypothesis-answering. We show how these measures help identify other patterns of analysis behavior, such as the “80-20 rule”, where a small subset of representations drove the majority of observations. Based on these findings, we offer design guidelines for interactive exploratory analysis tooling and reflect on future directions for studying the role that visualizations play in EDA.

Index Terms—Interaction Design, Methodologies, HumanQual, HumanQuant.

1 INTRODUCTION

The research literature widely considers interaction to play a central role in effective visualization for exploratory data analysis (EDA) [19, 51] because it supports a “dialogue between the analyst and the data” [50]. Recent empirical results, however, suggest a less clear picture. Studies have found no significant improvements in accuracy or error rates when using interactive visualizations for specific tasks such as bayesian reasoning or uncertainty communication [33, 49]. Furthermore, a contextual inquiry with professional data scientists revealed that interactive visualizations are primarily used for communicating results rather than as a medium for conducting the analysis itself [5]. These findings suggest a gap between the theoretical benefits of interactive visualizations and their practical application in EDA.

We hypothesize two diagnoses for these discordant bodies of results. First, much of the work demonstrating the value of interactive visualization in EDA is conducted within systems purpose-built to support

this activity (e.g., Tableau [6], Voyager [54], VisTrails [8], among others [24]). As a result, participants cannot “opt out” of the modality and conduct their analysis through other means (e.g., via code). Second, although existing approaches largely recognize that analysis is a *situated* activity—that is, it involves human analysts working in a particular context, making observations with various representations of data—thus far, these methods often focus on one aspect of this behavior rather than synthesizing across it. For instance, thematic analyses have been used to identify patterns of analytic behaviors [24], but it can be difficult to describe how these patterns manifest with particular interactive representations. On the other hand, quantitative approaches (e.g., interaction telemetry and provenance [36, 54]) capture detailed information about how analysts use particular representations. But without the context of qualitative insights, they can struggle to disambiguate observations. For instance, does hovering over a visualization indicate hesitation, gesticulation, or hypothesis testing? Recent “insight”-based approaches [7] have come perhaps the closest to capturing the richness of analytic activity, but are presently focused on a narrow band of activity: quantitative insights described as data transformations.

To study how choices of data representation (including interactive and static visualizations) affect EDA, we aim to understand not only the *what* of exploratory analysis (i.e., the insights gained) but also the *how* (i.e., the evolving process and the use of different representations). To this end, we pose two research questions:

RQ1: How do analysts’ observations evolve over an EDA session?

RQ2: How do interactive and static data representations influence the processes and outcomes of EDA?

- Dylan Wootton, Amy Rae Fox, and Arvind Satyanarayan are with MIT CSAIL. E-mails: {dwootton, amyfox, arvindsatya}@mit.edu,
- Evan Peck is with University of Colorado Boulder. E-mail evan.peck@colorado.edu

Manuscript received xx xxx. 201x; accepted xx xxx. 201x. Date of Publication xx xxx. 201x; date of current version xx xxx. 201x. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org. Digital Object Identifier: xx.xxx/TVCG.201x.xxxxxx

To address these questions, we conducted a qualitative experiment [42] involving 13 data science professionals using Jupyter notebooks. Participants were asked to complete two analysis tasks: the first with a lightweight library for authoring *static* visualizations, followed by a second with an extended library including *interactive* visualizations. Given their widespread use, Jupyter notebooks afford a more real-world context to study analytic behavior and, critically, do not *presuppose* the value of interactive visualization. Thus participants were free to forego visualization and interaction altogether, and simply author Python code using any third-party libraries they wished.

To capture the full spectrum of analytic behavior, we recorded participants' verbal utterances and telemetry, merging these data streams through a content analysis [21] to create a *unified dataset of analytic activity*. To analyze this dataset, we developed a novel *formalism* that models EDA sessions as a sequence of analysis states. Each analysis state is either the representation an analyst constructed (e.g., the output of a dataframe, or an interactive visualization), or an observation they made (i.e., an utterance about one or more representations).

To address RQ1, we leverage our formalism to code and track analyst observations over time. We identify 15 distinct types of utterances, grouped into four categories: utterances about *dataset* size or orientation, or whether there was any missing data; utterances about *variable* distribution or outliers; *relationship* utterances that expressed concepts including strength, directionality, and clustering; and *process* utterances that described intended analysis steps, or meta characteristics about a representation. Our analysis of these observations shows they follow distinct temporal patterns during EDA (§ 5). Analysts tend to address dataset-level metadata early on, while variable distributions and relationship insights occur throughout the analysis. Notably, interactive visualization accelerate relationship utterances, with these statements occurring 15% earlier than under the static condition.

To investigate RQ2, we leverage our formalism to combine representational telemetry with analyst observations, enabling us to explore the co-occurrence of representation use and analytical insights. We introduce a series of quantitative metrics including *revisit count*, or the total number of times a participant hovered over a representation; *output velocity*, or the number of representation instances created per unit time; and, *representational diversity*, or the number of unique representation types created during an analysis. We use these metrics to investigate patterns of exploration, revealing how some participants achieved broad coverage during their EDA (§ 6.4). Furthermore our formalism uncovers patterns in representation usage. Notably, we observe an 80-20 rule of representation use (§ 6.2.1) and the propensity to use all-attribute representations as aids to plan analyses (§ 6.2.2). Taken together, our work contributes to calls for "deepening [the] theoretical foundation" of exploratory data analysis [22].

2 RELATED WORK

Our work continues a tradition of studying EDA through technical and empirical approaches. In these section, we review these prior studies — organized by their methodological choices — and contrast their results with our objectives.

Attribute Methods: Attribute-based methods have provided valuable insights into how analysts explore data features during EDA. These approaches operationalize EDA by quantifying the number and combinations of attributes that analysts examine, using metrics such as attribute-set counts [2,44,54] or search trees structure [6]. These metrics facilitate comparing different analysis sessions, enabling researchers to assess how various interventions affect the breadth and depth of attribute exploration during EDA. Moreover, they reveal structural elements of the exploration process. For instance, Battle & Heer's study of analysts using Tableau identified key "analysis-states" — particular attribute combinations that played pivotal roles in participants' explorations [6]. Notably, their study finds that analysts using Tableau often prefer depth-oriented exploration, thoroughly investigating specific attribute relationships, rather than employing a breadth-oriented approach that surveys a wide range of different attribute sets. Our work extends these results by describing how particular representations shape attribute exploration. For example, we find analysts engage in *attribute*

addition when using interactive visualizations (§6.3) alongside other strategies used to broadly cover data attributes (§6.4).

Insight Methods: Insight methods focus on identifying and characterizing the analytical knowledge generated during EDA [39]. These methods typically employ think-aloud processes [9, 39] or elicit insights through open-ended responses [37]. Researchers then code these insights based on their semantic content, such as *Generalization* or *Hypothesis* [30], and analyze additional qualities like whether insights are broadening or deepening [44] or their factual correctness [58]. These coded utterances are often aggregated to compute metrics like time-to-first insight and total number of insights [9, 17, 30, 39, 58].

We differentiate our approach from previous insight methods through the use of qualitative content analysis to record both *what is said* and *what representations were used* to make such utterance. By explicitly linking the insight to the representation, our work investigates how different representations co-occur with particular insights. As a result, we compute aggregated information about insights during analysis conditions (§5) but also investigate how insights are formed using particular representations (§6.1). This approach lets us understand the impact of visualizations on the EDA process, such as analysts deriving 80% of their insights from just 20% of their representations (§6.2.1). Furthermore, our qualitative content analysis captures a wider range of insights, demonstrating how specific visualizations correspond to particular types of observations (Fig. 7).

Interaction Traces: Interaction traces provide rich quantitative data to describe analyst activity, offering insights into specific measurable behaviors during EDA. These traces range from simple actions like chart hovers [44, 54] to complex action sequences within interactive visualizations [36]. Researchers have leveraged these logs to create metrics assessing exploratory behavior and to reveal how user characteristics influence exploration patterns [13]. However, a key limitation of interaction traces is their inability to capture the meaning behind interactions. A hover over a chart could represent an insight being made or analyst confusion. To address this, researchers often combine interaction traces with other characterization strategies. In attribute-based methods, for example, they help demonstrate when a particular set of attributes is "considered," from hovering over visualizations [54] to creating them in Tableau [6]. In our work, we link interaction traces to utterances, revealing how specific interaction patterns can indicate different analysis strategies. For instance, we calculate a revisit count for each representation based on hover frequency, and used this metric to identify that a subset of highly revisited charts are frequently associated with analysis planning behaviors (§6.2.2).

Modeling Notebook Corpora: Recent research has explored modeling notebooks and their histories, primarily focusing on predicting future analyst actions given the current notebook state. For instance, Auto-Suggest [57] uses a recurrent neural network trained on notebook corpora to generate future data transformation operations. Similarly, EDA Assistant [29] ranks slices of programs from similar notebooks and provides frequently used next steps. Other approaches have focused on generating entire EDA sessions rather than snippets of code. For example, Bar et al. [4] formulate EDA as a control problem where they use a reward signal based on the novelty and diversity of insights to automatically generate entire EDA sessions. While these systems develop useful tools to facilitate EDA, they primarily aim to predict the analyst's next action rather than providing insights into broader patterns of analytic behavior during EDA. In contrast, our work seeks to understand the cognitive processes and decision-making patterns that underlie analysts' interactions. Future systems-building work could use the results of our analyses to better model analyst activity and recommend next steps.

Interviews and Surveys: Interview and survey studies provide crucial insights into the real-world practices of data scientists, shaping our understanding of EDA workflows. Kandel et al. conducted foundational work understanding the stages of data science work [25]. They interviewed data scientists across various enterprise organizations outlining five key job responsibilities: *discovery*, *profiling*, *data wrangling*, *modeling*, and *reporting*. These elements are central to data science activities. Further refining this understanding, Wongsuphasawat et al.

conducted interviews that revealed a more detailed set of 16 analytic behaviors, such as *converting data formats* and *examining bivariate plots* [53]. Interviews also enable researchers to investigate attitudes towards particular EDA tools, such as Batch et al.’s [5] work to understand the “Interactive Visualization Gap” in EDA. Furthermore, when conducting empirical studies, surveys are often administered following an exploratory analysis session [15, 44, 56]. Most commonly, surveys include questionnaires like the NASA-TLX [18] for understanding subjective workload during a task [15] or Likert scale questions to elicit preferences when using a tool [44, 56]. Our work builds on these findings by examining how analysts use different representations during EDA (§6.1), providing a more nuanced understanding of *when* and *why* certain visualizations are used — an approach that allows us to bridge the gap between reported practices and actual behavior.

Thematic Analysis: Thematic analysis seeks to identify occurrences of broad behavioral patterns or *themes* [9, 24, 41]. These approaches typically involve participants thinking aloud in order to interpret the meanings of behaviors given their context. For example, Kale et al. [24] investigated the effect of a tool that enables *model-checking* through a within subjects comparison. Using thematic analysis, they characterize how the patterns of analysis shifted when the model-checking functionality was introduced, revealing that this tool “structure[d] participants’ thinking around one or two long chains of operations”. In contrast to our study, thematic analysis does not seek to characterize the content of entire analysis session, choosing instead to focus on larger themes that were observed during exploration.

3 METHODS

Our research questions aim to describe the temporal progression of analysts’ observations and inferences (**RQ1**), while also comparing how these behaviors unfold with *static* vs. *interactive* visualizations (**RQ2**). These research questions are both *descriptive* and *comparative* in nature. To address these questions comprehensively, we adopted a hybrid design that combines task observation and semi-structured interviews within the framework of a repeated-measures experiment. This approach, described in the mixed methods literature as a *qualitative experiment* [42], allows us to capture rich, contextual data about analysts’ thought processes and actions while also enabling systematic comparisons between static and interactive visualization conditions.

3.1 Study Design, Procedure, and Participants

Our independent variable is **representation interactivity** with two levels: *static* and *interactive*. We use a repeated-measures (i.e. within-subjects) structure where we measure participant behavior in two tasks (static, interactive), and with two datasets that are counterbalanced in their assignment across the two tasks. Note that we *did not* counterbalance static/interactive task order because the interactive features necessarily built upon knowledge of the static visualizations. Participants engaged in a 90-minute video-conference divided into four parts: introductions/informed consent, two *EDA sessions*, and an interview.

Each *EDA Session* began with an introduction to the (static/interactive) features of the visualization library (*Features Intro*), followed by an opportunity for the participant to explore the new APIs via sample code (*Features Tutorial*). Next, participants were given a notebook with a dataset and scenario for an *Analysis Task*, and asked to complete an exploratory analysis in approximately 25 minutes while thinking aloud. Throughout this process, their interactions with the notebook — running code cells, brushing on charts, and scrolling — were recorded as interaction telemetry. The structure of the static task was identical, with the dataset counterbalanced across participants. Each session concluded with a semi-structured interview and debrief.

We recruited 16 participants through social media, personal networks, and crowdwork platforms. Two participants were involved in pilot studies to refine data collection procedures. Of the 16 participants who completed the study, three were excluded due to either incomprehensible think-aloud responses or an insufficient level of Python proficiency. Our resultant pool comprised 13 participants: 4 women, 8 men, and one person who identified as non-binary; participant ages ranged between 27 and 41 years (average age 31). All participants

regularly conducted EDA using Jupyter notebooks as part of their occupation. Their most common job title was Data Scientist (5), followed by PhD Candidate (3), Software Developer (2), Data Analyst (1), Economist (1), and Statistician (1).

3.2 Controlling for Library Expertise with Altair Express

To facilitate comparisons between participants’ behaviors, it was essential that they used the same visualization library. However, this introduces a confound: participants’ existing expertise with visualization packages. To control for this, we developed a new visualization package to establish a common baseline of relative novelty for all participants.

Our library, called *Altair Express* (ALX),¹ is a Python-based visualization package that offers a high-level declarative API for specifying interactive visualizations. In contrast to the composable approach of the existing Altair visualization package (and its underlying grammar Vega-Lite [45]), ALX instead provides a *typology* of visualizations and interaction techniques — an approach we chose to reduce specification friction analysts might face during EDA. We surveyed existing Python-based chart typologies (e.g., Plotly Express, Seaborn, etc.) and implemented the set of statistical charts we hypothesized to be most relevant to EDA including: *barplot*, *countplot*, *hist*, *jointplot*, *lineplot*, *heatmap*, *pairplot*, *profile*, *scatterplot*, and *stripplot*.

The interaction typology in ALX is defined by *effect-action* pairs: an *effect* is the change that occurs when a user performs an interaction (e.g., showing a tooltip, zooming into a region, etc.), and an *action* is the event that triggers the interaction (e.g., clicking, brushing, etc.). Thus, the typology comprises: *highlight_brush*, *filter_brush*, *tooltip_hover*, *pan_zoom*, *filter_slider*, *filter_type*, *highlight_color*, and *highlight_point*.

Using the `+` operator, visualization and interaction types can be composed together. For instance, `alx.highlight_brush() + alx.scatterplot(data, x='Weight', y='Horsepower')` produces a scatterplot of the Weight and Horsepower of cars; users can brush the scatterplot highlighting selected points in blue and dimming the rest to gray. Using `+`, users can add multiple interaction techniques to a single visualization, or concatenate multiple static and/or interactive visualizations together to produce a custom dashboard. ALX implements these interactive visualizations via Vega-Lite [45].

Finally, in addition to its specification language, ALX implements a handful of features designed to address limitations researchers have identified of using interactive visualizations in computational notebooks [5, 56]. For example, with ALX, analysts can use a “*copy-and-paste*” in order to extract an underlying data selection. When a selection is made — for instance, by clicking on a point, dragging a slider, or brushing — the analyst can press `control + c` to copy the pandas query necessary to select the data. This query can then be pasted into the subsequent cells in the notebook to filter down to the selected data for further investigation or charting.

3.3 Data Analysis Procedure

We applied an inductive content analysis [21, 32] to the rich stream of video and think-aloud data our participants produced. We split transcripts of the video recordings into discretized units of meaning we call *utterances*. And, using participants’ screenshare, mouse gestures, and linguistic prosody, we coded what representations participants used in the process of making a particular utterance. We limited the scope of our coding to only include the *Analysis Tasks* — thus, we excluded utterances participants made when they were familiarizing themselves with ALX’s features, debugging, or during the post-interview.

The first and second authors followed an inductive process consistent with the application of grounded theory in HCI [21, 34] to develop a codebook for categorizing participants’ utterances. This process involved eight iterations of independent coding centered on: (1) developing structure, (2) aligning criteria, and (3) reconciling discrepancies.

¹The name was chosen to mirror the relationship between Plotly and Plotly Express. That is, *Altair* : *Altair Express* :: *Plotly* : *Plotly Express*.

```

Session := State[]
State := <StateType, Timestamp>
StateType := Output | Observation
Output := <Representation, Telemetry>
Representation := Visualization | Dataframe | ValueCount |
                CodeCell | Column | Info | Describe
Telemetry := HoverWindow[] // could be extended
HoverWindow := <StartTime, StopTime>
Visualization := <Chart, Interaction{}>
Chart := <ChartType, Encodings{}>
ChartType := Scatterplot | Countplot | Profile | Pairplot |
            Barplot | Histogram | Lineplot | Stripplot |
            Heatmap | MultiView
Encodings := <Channel, Attribute>
Channel := x | y | color | ...
Interaction := <Effect, Action>
Effect := Highlight | Group | Filter | PanZoom | Tooltip
Action := Hover | Click | Drag | Type

Observation := <Utterance, RepresentationUsage[]?>
Utterance := <UtteranceType, Attribute[]?>
UtteranceType := Dataset | Variable | Relationship | Process
Dataset := Data Size | Missing Data | Data Orientation |
           Variable Metadata | Data Provenance
Variable := Range | Shape | Outlier
Relationship := Strength and Direction | Presence | Form |
              Subgroups | Outlier | Range Constriction
Process := Plan of Action | Representation Comment
Attribute := DataAttribute | DerivedAttribute

RepresentationUsage := <Output, InteractionUsed>
InteractionUsed := TRUE | FALSE

revisitRate(HoverWindow) := COUNT(HoverWindow)
hoverTime(HoverWindow) := SUM((StopTime - StartTime)[])
representationDiversity(Session) := UNIQUE(Representation)
representationVelocity(Session) := COUNT(Output)/Duration

```

Fig. 2: A formal definition of EDA sessions in terms of analysis states that comprise either a representation alone (e.g., a visualization, dataframe output, etc.) or an observation made with one or more representations. Italics indicates terminal symbols.

In the final round of reconciliation, the first and second authors independently coded a random sample of 100 utterances, to calculate an Inter-Rater Reliability (IRR) measure of Krippendorff’s $\alpha = 0.85$.²

4 A FORMAL DESCRIPTION OF EDA SESSIONS

We express the results of our mixed-methods analysis through the formal description shown in Figure 2. We find an EDA **Session** progresses through a sequence of analysis **States**. Each **State** can either be a standalone **Representation** (e.g., a **visualization**, dataframe printout, etc.) or be a verbal **Observation** that an analyst makes. For each representation, we collect a variety of **Telemetry** data, but our analysis focuses only on **HoverWindows** (i.e., time spans of when a participant hovered over a given representation)—we leave other abstractions that can be derived from telemetry data to future work.

Observations associate verbal **Utterances** with any **Representations** used to make them, as indicated through **RepresentationUsage**. We distinguish these observations into those made with interactive features (such as brushing or tooltips, coded as **InteractionUsed**) from those on interactive charts that did not utilize interactivity. We use the term **Utterance** rather than *insight* or *inference* to indicate that, even with the context of the participant’s screenshare, mouse gestures, and linguistic prosody, we cannot precisely determine the participant’s state of knowledge. Thus, we work to interpret as much of each utterance’s semantic content as possible via our qualitative coding procedure.

As Figure 3 shows, this procedure yielded 16 **UtteranceTypes** spread across four categories: utterances about the overall **Dataset**

²Krippendorff’s alpha is the recommended IRR metric for multi-code structures where more than one can be applied to one observation. Using a more generous alternative we calculate reliability of (*Observed Agreement*=0.87). In both cases our IRR passes normative thresholds of reliability [31].

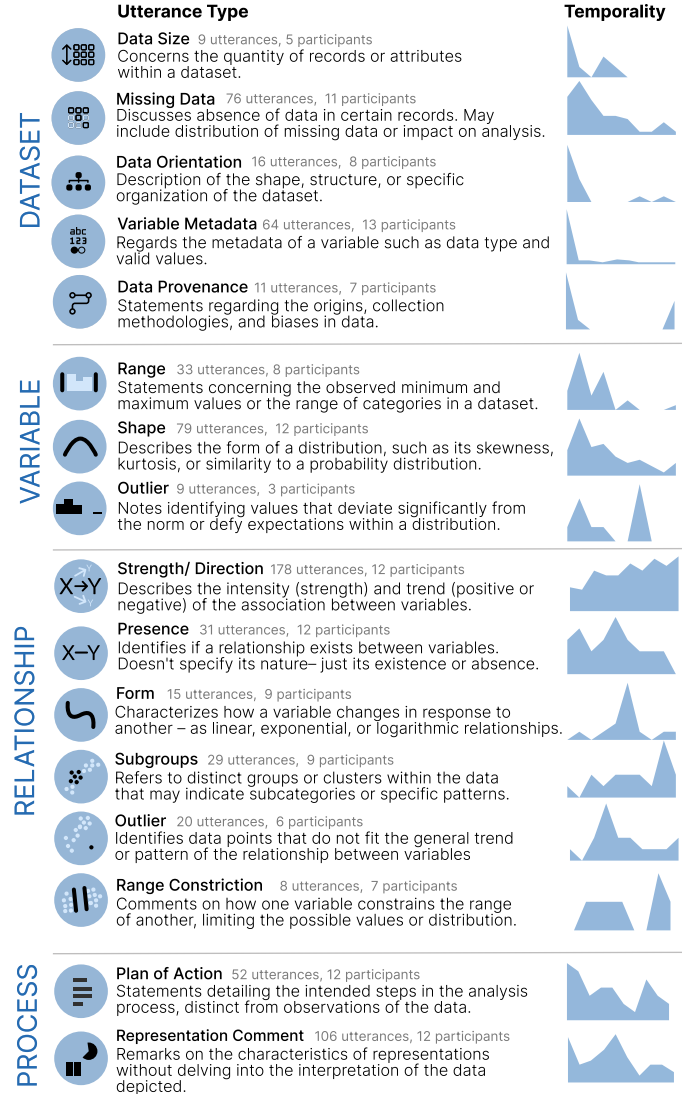


Fig. 3: Utterances are structured as a 2-level hierarchy, with the highest level codes (*Dataset*, *Variable*, *Relationship*, *Process*) describing the general topic of an utterance, and lower level detail codes delineating the utterance’s content more precisely.

including its size, orientation, quality, provenance, and metadata; utterances about individual **Variables** including about the distribution of data values (e.g., min, max, outliers) and the shape of this distribution; utterances about **Relationships** between variables including whether any relationship exists and, if so, what form, strength, and direction this relationship takes; and, finally, utterances about the overall analytic **Process** including statements about intended next steps or remarks about representations that are not about depicted data.

We find this formalism offers unique insights into EDA activity, illustrated by the following vignette inspired by participant behavior:

Ada, a data analyst at an e-commerce company, is tasked with investigating a customer purchase behavior dataset that includes *customer age*, *product categories*, *shipping speed*, and *customer satisfaction ratings*. Ada begins by creating a data profile ①, a multiview visualization with concatenated univariate histograms for each variable. While examining the distributions, she notices an unusual pattern in the *satisfaction ratings* ② - there’s a concerning spike at 1-star ratings, contrary to the company’s belief that customer satisfaction was generally high. Intrigued, Ada uses a crossfilter interaction to brush over the 1-star ratings, and observes a shift in the age distribution in the profile, noting that these dissatisfied customers tend to be younger ③.


```

analysis start
1 <Profile, Highlight-Brush>
2 <<Distribution Shape, [Rating], <Profile①>>
3 <<Relationship Strength, [Age, Rating], <Profile①, TRUE>>
4 <<Scatterplot, [Age, Rating], Highlight-Brush>
5 <<Relationship Cluster, [Age, Rating], <Scatterplot④>>
6 <Dataframe>
7 <<Relationship Strength, [Category, Speed, Age, Rating],
  <Dataframe⑥>, <Scatterplot④>, TRUE>>
analysis end

```

Fig. 4: Example of Ada’s analysis session encoded in our formalism. For clarity, we have omitted some levels of nesting for the formal description of this example. Colors are associated with the corresponding formalism construct: **Output** (non-visualization outputs), **Visualizations**, **Observations**, and **Representation Usage**.

To investigate further, Ada creates a scatterplot of *satisfaction ratings* vs. *customer age* ④. The scatterplot confirms a cluster of younger customers with low satisfaction ratings ⑤. Ada isolates this cluster using a brush selection tool and examines the associated customer details in a table view ⑥. Digging deeper into the table, Ada discovers that a significant portion of these customers’ purchases are from the “Gifts” category, and their *shipping speed* is often listed as “expedited” ⑦, suggesting young buyers might be using the platform primarily for last-minute gift purchases, resulting in higher stress and lower satisfaction when issues arise.

Using attribute-based metrics [6, 54], we might view Ada’s EDA as a three-step process: analyzing *all* attributes (with the profile); then analyzing *age* and *rating*; and finally, returning to *all* attributes (with the data table). This approach makes it difficult to identify that Ada did not ever actually analyze particular attributes (e.g., *purchase history*) despite their inclusion in certain representations (i.e., the profile and data table). Moreover, by being representation-agnostic, attribute-centric metrics treat the profile and data table as equivalent and, as a result, miss the different ways Ada used these two views — for instance, that she brushed the profile view to reveal a relationship between *age* and *satisfaction* versus examining the table in a more record-by-record fashion. These issues are compounded when applying attribute-centric metrics to analyze interactive visualization as the space of possible observations is greatly expanded [23].

Task and insight-based methods often do not account for representation either. As a result, they ignore analytic expressions that are not verbalized and instead latently conveyed via the representation — that is, the act of making a chart is intrinsically an inquiry, even if it is not used to make an observation out loud. Moreover, depending on the granularity of task/insight codes, these methods may miss important nuance in Ada’s activity. For instance, with the protocol followed by Zgraggen et al. [58], one might label Ada’s analysis as a *Distribution Shape* insight followed by two *Correlation* insights — a strategy that collapses insights about “clusters” and “correlations” together. More recent insight-based approaches, such as the formalism developed by Battle & Ottley [7], begin to address many of these shortcomings — for instance, they formalize an *AnalyticKnowledgeNode* to encompass data relationships and transformations. While this method would be able to capture much of Ada’s activity (e.g., interactive brushing as issuing a series of data queries), it is focused only on describing the *quantitative insights* a participant might make about a dataset.

In contrast, our formalism separately records the representations Ada constructed, the utterances she verbalized, and links the two together as a series of observations (Fig. 4). This description better reflects the situated nature of EDA — that observations occur *with* representations, and that non-verbalized representations can play important roles in an analysis session. In the subsequent sections, we demonstrate how to apply the formalism to investigate behaviors during EDA.

5 CHARACTERIZING ANALYST UTTERANCES

In this section, we analyze participant **Observations** to investigate the semantic content of analyst EDAs and how they evolve over time

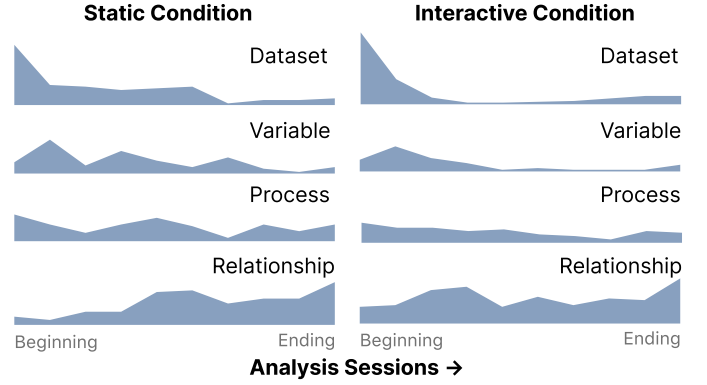


Fig. 5: Occurrence of utterances categories throughout analyses.

(RQ1). We examine the temporal patterns of different types of observations throughout EDA sessions (§5.1), comparing how these patterns manifest in static versus interactive conditions. Additionally, we explore the transitions between different types of observations, extending our understanding of exploratory behaviors beyond the previously identified touring motifs [24] (§5.2). This analysis provides insights into the structure of EDA processes and how they are influenced by the availability of interactive visualizations.

5.1 Temporal Patterns

As the area charts in Figure 5 show, we find that while analysts’ processes align *in aggregate* with traditional, linear EDA models (from individual variable analysis and then relationship exploration [25]), the analysis process is both more fluid and sensitive to interactivity than rigid interpretations of those models would suggest. To examine analyst processes, we calculated the median moment through the analysis session (expressed as a percentage) in which analysts made **Observations** across our four **UtteranceTypes**: **Dataset** (13.43%), **Variable** (25.60%), **Relationship** (56.86%), and **Process** (40.18%).

In particular, interactive EDA sessions prompted earlier observations about **Relationships** in the data (IQR 28%–75% through a session) compared to static EDAs (IQR 43%–85%). We hypothesize that the use of interactive profiles, featuring cross-filterable univariate histograms, encouraged analysts to explore relationships sooner. Our subsequent findings of analysts switching from static to interactive profilers (§ 6.3) support this: many participants shifted from **Variable** to **Relationship** utterances almost immediately upon encountering the interactive profile. This finding opens questions about whether the affordances (or *presence*) of interactive profiles enables bypassing distribution analysis, and whether we can articulate the tradeoffs of such process changes. More broadly, the presence of relationship utterances across both static and interactive EDA sessions suggests that analysts are willing, perhaps even eager, to explore **Relationships** before fully developing a mental model of individual **Variables**.

5.2 Sequential Transitions

During their analyses, participants made seven different types of utterances on average. Looking at the sequential transitions between utterances reveals a number of common analysis motifs [24].

Tour-Driven Exploration Fig 6 ①: Frequent self-transitions between similar utterance types (e.g., multiple consecutive utterances focused on **Relationship** strength) suggest that analysts often adopt a systematic “touring” approach during EDA. This finding aligns with concepts of univariate and bivariate tours [24, 27], where analysts methodically explore specific aspects of individual **Variables** and their **Relationships**. However, we observed self-transitions extending beyond **Relationship** analysis to include utterances about **Missing Data** and **Variable Metadata**. This suggests that “touring” behaviors are broader than previously described [24].

Column- vs. Row-Centric Missingness Fig 6 ②: The most common transition between utterance types was moving from **Missing**

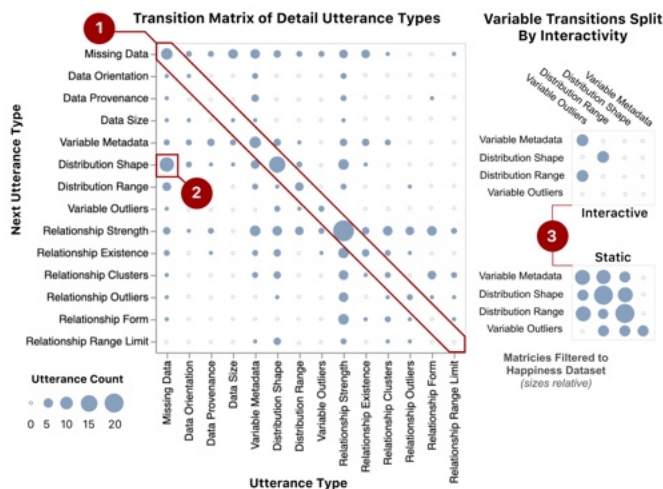


Fig. 6: (left) A transition matrix of sequential utterances. (right) The transition matrices showing the "Variable Gap" in transitions between Interactive and Static analyses on the happiness dataset.

Data to Distribution Shape. This often occurred early on in analyses through use of profile visualizations. The design of profile presents missing data alongside the column's distribution, subtly promoting a column-centric view of missingness. However, as a counter-example, P10 investigated missingness as a characteristic of individual data records (rows), skipping the profile entirely. Visualizing the missingness per record on a scatterplot, he commented "... most of the rows have no missing columns, and then they progressively have more and more. So I guess, depending on what the analysis we're gonna do is, we may or may not exclude data points." This approach highlights different potential causes for missingness and raises a design question: how can profile encourage analysis of column- and row-level missingness?

The "Variable Gap" and Interactive Profiles Fig 6 (3): In the happiness dataset, many participants skipped characterizing Variables altogether, instead immediately focusing on Relationships. This caused a Variable Gap between conditions, visible in the transition matrices (right). This shift often coincided with the use of an interactive profile—a tool comprising univariate distribution visualizations that supports cross-filtering. For example, participant P5 initially followed a variable-first pattern in her static analysis, narrating out 6 distributional utterances about her variables using the profile. Upon beginning her interactive analysis, she immediately began making relationship utterances by cross filtering on the profile view (see § 6.3 for more information).

6 CHARACTERIZING REPRESENTATIONS AND USAGE

Guided by RQ2, we explore the link between Representations and Observations. We find that analysts heavily rely on a small subset of representations for conducting their analyses (§6.2.1), and employ certain representations to plan and navigate subsequent steps of their analysis (§6.2.2). We also observe a shift in analysis content, with interaction drawing analysts towards relationship observations (§6.3). Additionally, we investigate the analysts who achieve the broadest coverage in their EDAs and describe the analysis strategies they employed to do so (§6.4).

6.1 Temporality, Diversity, and Velocity

Across all Sessions our participants constructed a total of 1169 Outputs, with an individual analyst averaging 44 outputs per analysis. Python code executions were most common, especially at the beginning and end of sessions, typically for checks on central tendencies. Visualizations began to dominate about 15% into each session, becoming the foundation for most subsequent observations (Fig. 7 (left)). Based on this data, we introduced two metrics: representationDiversity, the count of unique representations constructed during a session, and

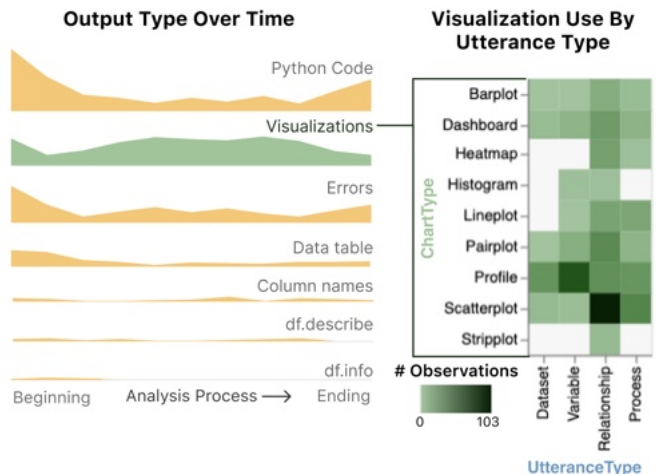


Fig. 7: (left) The count of representation created over time. (right) A heatmap of the number of times different Visualizations were used to make an Observation, according to UtteranceType.

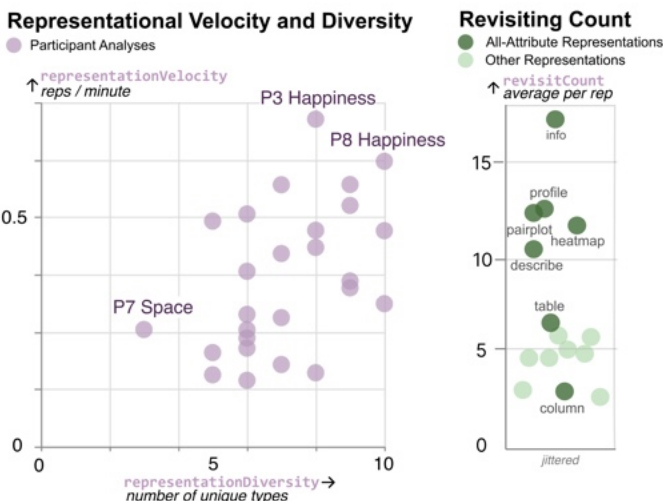


Fig. 8: (left) A scatterplot of representationDiversity and representationVelocity for each analysis session (§ 6.4). (right) A jittered strip plot showing average revisitCount and count of Plan of Action utterances by Representation. Representations are colored by whether or not it is an all-attribute representation. Representations to the bottom are typically one-off question-answering tools whereas representations to the top are frequently revisited when deciding analysis paths (§ 6.2)

representationVelocity, measuring the rate at which these representations were created. As Figure 7 shows, these metrics are moderately correlated (Pearson's $r = 0.47$); we discuss their role within analysis sessions in a subsequent section (§ 6.4).

Our analysis of the intersection of ChartTypes and Observations (Fig. 7 (right)) reveals both expected and surprising usage patterns. For example, unsurprisingly, scatterplots frequently facilitated Relationships utterances, while profile views were used in making Variable utterances. However, as Figure 7 shows, participants would frequently use charts beyond their intended purposes or in ways that break with best practice. For instance, Variable utterances constituted only 42% of observations made with profile views — even though, ostensibly, this is the core purpose of a columnar distribution of data values. Similarly, in contrast to visualization theory and recommender systems, which emphasize perceptual effectiveness, participant P9, a data science instructor, specifically created a representation she called a "spaghetti plot" — a line chart with 180 different series overplotted. Ahead of creating the chart she commented "It's going to be a bad idea", but persisted precisely because she wanted to ensure that the

plot itself *was ineffective*, as a gut check.

6.2 Hover Patterns and Observations

Hover patterns, captured through per-representation metrics such as `revisitCount` and `hoverTime`, indicate the frequency and duration of analysts' engagement with different representations. These metrics help uncover aspects of visualization usage and attention distribution that are not apparent from code execution histories alone. We combine these metrics with the `Observations` analysts made to reveal how telemetry correlates with analysis behavior.

6.2.1 The '80-20 Rule': Why Some Visualizations Matter More

Our analysis reveals a 80-20 pattern in how participants use representations during EDA. The top 20% of most frequently hovered representations (*top-20*) accounted for 79% of total `hoverTime` and 75% of observations. Representations in the *top-20* had hover durations of at least 30 seconds and an average of 2.8 Observations each, indicating deep engagement. In contrast, the bottom 80% of representations (*bottom-80*) saw significantly less use, with an average of just 0.2 observations per representation. We identify two key differences between these two sets that sheds light on analyst preferences: the ability to encode multiple attributes simultaneously, and the role of interactivity.

Representations displaying information about multiple variables simultaneously (e.g., profiles, correlation heatmaps, pairplots) were more common within the *top-20*. These *all-attribute representations* made up only 2% of the *bottom-80* but constituted 22% of the *top-20*, an 11-fold increase. Analysts frequently engaged with these visualizations a "touring" process, previously described in § 5.2. This involved systematically exploring the visualizations and commenting on different variable combinations approximately every 5-15 seconds. The prominence of this behavior is reflected in the extended average hover times for all-attribute visualizations, with profiles at 67 seconds, heatmaps at 75 seconds, and pairplots at 169 seconds. In contrast, we see a marked decrease in `hoverTime` with Code Cells used for quick statistical checks (from 48% of the *bottom-80* to 9% of *top-20*, averaging 4.9 seconds of hovering per representation).

Interactive visualizations were more prevalent within the *top-20* (24% of the *top-20* vs. 16% of the *bottom-80*). Analysts particularly favored the `highlight_brush` as it enabled cross-linking data subsets across multiple charts. This technique was used in over 56% of interactive representations in the *top-20*, compared to 37% in the *bottom-80*. Similarly, the `filter_brush` technique, which filters out all non-selected data marks from view, was used in 30% of the interactive scatterplots found within the *bottom-80*. However, `filter_brush` went to 2% in the *top-20*, a likely side effect of filtering obscuring important context in standalone charts.

Finally, `pan_zoom` interactions were prevalent in the *bottom-80* (31% of interactive representations) but declined to 18% in the *top-20*. Analysts consistently struggled to find effective use for pan-zoom interactions, suggesting a lack of intuition for its analytical value. Out of the 16 instances in which pan-zoom was used, we observed only one instance where it successfully uncovered an insight that would have been difficult to obtain otherwise. In this case, participant P10 zoomed into a dense, overplotted region of a scatterplot to gain more resolution, and was able to reveal a pattern in the depicted data. However, even this success story was marred by discomfort — P10 added pan-zoom to a set of horizontally arranged scatterplots that shared a common y-axis; thus, the coordinated scrolling of all scatterplots made him feel disoriented, prompting him to request "*can we turn that off?*"

6.2.2 All-Attribute Visualizations Aid Planning

Representations with high `revisitCounts` (over 10 times) often serve as process planning tools, helping analysts orient themselves and prepare their next actions (Fig. 8 (right)). A prime example of this is participant P6's use of a correlation heatmap. She created this visualization to identify the most strongly correlated attributes within her dataset and frequently returned to it as a guide for selecting specific attributes for further investigation. As she noted, "*let's look at the one that is most positively correlated, which seems to be log GDP per capita. So I'll start with that variable*". This led her to further

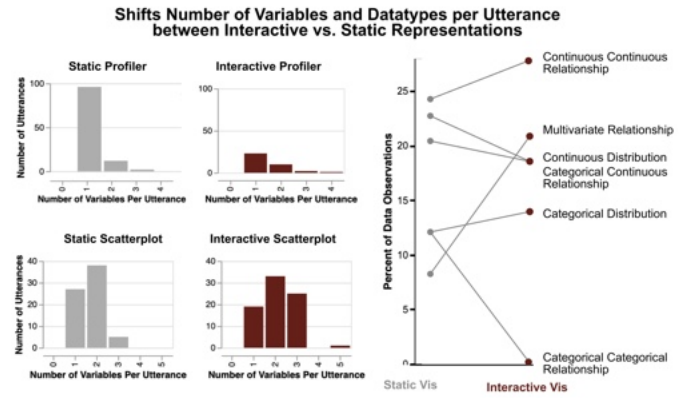


Fig. 9: (left) A barchart showing the number of utterances per attribute count, faceted by whether the utterance was made using static or interactive profiler and scatterplot visualizations. (right) A slope chart comparing utterance type counts between static and interactive visualizations.

investigate highly correlated variable sets through custom dashboards for deeper exploration, ultimately leading to an exceptional 23 observations (§ 6.4). Notably, heatmaps appeared to be particularly effective in this role, averaging 3 times as many Process utterances as other representations.

Such action-planning is not restricted to only visual all-attribute representations — participants frequently revisited data frame outputs (including `df.describe`, `df.info`, and the tabular output) to formulate their plans. For instance, P11 read through the individual values of a dataframe printout, commenting: "*Of course, we cannot say for the whole thing [based on just the shown rows]. So my strategy will be like going through each of the variables here, and do the summary statistic.*" Looking across all `Observations` tuples in our dataset, all-attribute representations are associated with Plan of Action utterances at a rate of 5 times higher than other representations.

6.3 An Interactive Draw Towards Complexity

We observed correlations between the use of interactive visualizations and changes in the types and number of attributes analysts considered. When using interactive visualizations, an *attribute addition* pattern emerged, where analysts' explorations moved from univariate distributions to bivariate relationships or multivariate analyses. For example, participant P6 used a static profile visualization to analyze the univariate distributions of her columns, making 6 utterances about their distributions. At the beginning of the interactive session, she created an interactive version of the profile, and immediately began using it to analyze relationships — brushing on the chart to examine a target population and generating 6 new utterances about that population's relationship to other variables. This pattern of behavior persisted across datasets for other participants (Fig. 9 (left)). Analysts consistently leveraged interactivity to deepen their exploration, sometimes even skipping over distributional analyses to instead analyze more complex data relationships (§ 5.2).

We also observed shifts in behavior prompted by filtering interactions in scatterplots (Fig. 9 (right)). Prior to the interactive session, we observed participants discussing bivariate relationships using scatterplots; however, when interaction was added, their utterances tended to focus on the multivariate relationships. Multiple participants used brushes to extract subsets from data clusters and pursued analysis paths to differentiate that cluster from the rest of the data. Another case of this was the use of the `filter_slider`, an interaction technique which filters the chart to only the data value present in a particular value on a slider query widget. The shift we observe between these interactive and static charts presents the allure of interactive representations, seemingly pulling analysts towards investigating more complicated relationships even when those interactions are not actively being used.

However, attribute addition behavior was not observed equally across data types. Our participants often used interactive visualizations for multivariate (frequently all continuous variables) and continuous x con-

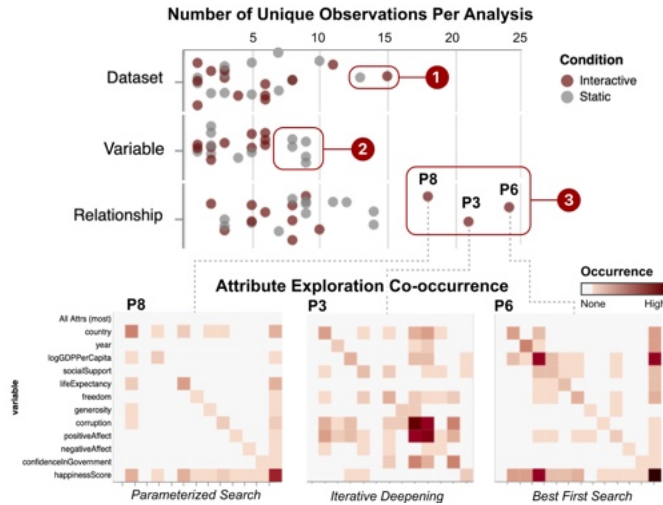


Fig. 10: (top) A striplot of percent of total unique **Observations** visited per analysis session, broken down by high level type and colored by Analysis Condition (Interactive or Static). (bottom) Heatmaps representing attribute co-occurrences when participants made observations about relationships between variables.

tinuous bivariate relationships (Fig. 9 (right)). However we note the overall patterns are most salient at the aggregate level and the participant level contains sparsity in the utterances made for a given data type. Thus while we chose to report the results to fully describe the behavior that we saw, such descriptions warrant additional investigations to understand the role that interaction may play in drawing analyst hypotheses towards more multivariate and complex relationships and if such patterns exist during longer EDA sessions.

6.4 Patterns of Broad Observation Space Exploration

Previous studies have characterized EDAs based on the number of attributes analysts considered [6, 54]. We build on this approach, applying it to our more comprehensive definition of **Observations**, which encompasses both what was learned (**UtteranceType**) and which data **Attributes** were considered. Adapting Battle et al.’s method [6], we created binary histograms representing whether participants made a specific utterance type on an attribute set (e.g., observed the *relationship* between *happiness* and *GDP*). By calculating the percentage of total possible states each participant explored, we can rank participants by their breadth of exploration and investigate the ways in which **Representations** changed the analysis **Session**. For example, participant P9, a data science instructor, made the most extensive **Dataset** observations across both static and interactive conditions (Fig. 10 ①). These observations occurred as P9 began each of her analysis sessions with a variable metadata tour: systematically going through each attribute in the data dictionary, spending time discussing what the variable meant and her opinions on its usefulness. Similarly, we observe the 5 participants who made the most **Variable** utterances (Fig. 10 ②) did so in the static condition using profile visualizations.

In contrast, approaches for exploring a broad set of **Relationship** observations (Fig. 10 ③) reveals a diverse set of strategies. To investigate these patterns of exploration, we created attribute co-occurrence heatmaps (Fig. 10) to “fingerprint” and explain these strategies:

P8: Parameterized Search. Driven by a clear goal and an aversion to “mindless” exploration, P8 adopted a systematic, iterative approach reminiscent of a parameterized search through **Representations** and **Encodings**. She cycled through which attributes were mapped to encodings (e.g., `scatterplot(y=happiness, x=column[index])`), methodically investigating potential relationships between each attribute and the outcome variable. When she encountered specific patterns of interest, she then modified her scatterplot, adding interactions such as brushes and tooltips to investigate outliers and subsets. The resultant fingerprint visualization depicts a focused analysis centered on the outcome variable, with some targeted off-diagonal probes

into the country, investigated using tooltips and brushes.

P3: Iterative Deepening. P3’s approach was guided by emergent patterns in the data, resembling an iterative deepening search. He generated scatterplots based on his intuition for interesting relationships, largely ignoring the outcome variable. This is reflected in his focus on variables other than *happinessScore* (bottom row and right column). Upon noticing clusters, he investigated their characteristics, iterating through interactions and encodings (adding tooltips, brushes and **color** encodings) to identify potential explanatory variables. This behavior is captured in his high **representationalVelocity** and **representationDiversity** as shown in Figure 8 (left), suggesting he wasn’t wedded to a single visualization type but explored various options to find insights. This iterative deepening process ultimately led to a scattered thumbprint reflecting his serendipitous journey through attribute space, driven by unexpected findings.

P6: Heuristic-Guided Best First Search. P6’s approach combined a methodical foundation with responsive, opportunistic elements characteristic of best-first search [40]. This strategy prioritizes exploring the most promising nodes within a search space based on a pre-defined heuristic. P6’s analysis mirrored this approach by selecting attributes to plot based on their correlation with her outcome variable. After analyzing these attribute sets in a custom dashboard, she would return to her correlation matrix to choose her next attribute set, effectively “touring” through her correlation matrix. She revisited this matrix 35 times during her analysis, demonstrating a high **revisitCount** for this visualization. This strategy produced a cohesive analysis that investigated both direct predictors and potential confounds of the outcome variable, evident in her targeted analysis along the bottom row and off-diagonal of her thumbprint visualization.

6.5 Thinking in the Language of Interaction

In interaction design, perceived affordances [38] signal the operations a user believes are possible within an interface. Well-designed affordances establish interaction dynamics — the rules governing how users interact with the interface. Our study revealed that data scientists reasoned about these dynamics to generate new analytical hypotheses. In other words, they translated “the language of interaction” into novel analytical questions. As participant P8 described: “*My thought of intersecting High GDP and High Life-Expectancy [countries] happened precisely because there was interaction... I was thinking, ‘Oh I wonder if multi-select works’... That is actually what led me to think, ‘Oh this would also be interesting on an analytical level.’*”. Later she commented that such an insight “*would not have occurred to me if not for the fact I was working with an interactive visualization.*”

Participant P6’s insights emerged from a similar process of experimentation. Having successfully used ALX’s copy-and-paste technique to paste filters between charts, he began to consider the broader possibilities this interaction technique offered. While browsing other charts, he stumbled upon a bar plot showing the count of records over time. Intrigued, he initially tested if the copy-and-paste would function in this context. However, a spark ignited: rather than a simple test of function, he realized it would be more insightful to filter on the most recent years of data. This act of guided experimentation, prompted by the affordances of an interaction design (rather than performing the interaction itself and observing any updates), led him to discover an unexpected trend in life expectancy over time.

These examples suggest that interactive features play a more generative role in analysis than typically acknowledged. While existing literature often focuses on interactions as tools for completing specific tasks, our observations reveal that the rules of the interaction design can inform emerging hypotheses and shape analytical reasoning. This insight has two key implications. First, there’s an opportunity to critically examine how we articulate and implement the constraints and rules of interaction dynamics. Different designs may substantially impact how analysts reason about these rules and, consequently, how they approach their analysis. Second, beyond investigating how visual cues influence interaction usage [10], future studies should explore how various cues shape analysts’ conceptualization and potential application of interaction techniques. By recognizing the interplay between interaction mechanics and analytical cognition, we can pave the way for tools that

more effectively partner with the analyst during the discovery process.

7 DISCUSSION AND FUTURE WORK

In this paper, we conducted a qualitative experiment to richly characterize the *situated* nature of EDA in computational notebooks. Through mixed-methods analysis of utterances and telemetry, we developed a formal description of EDA sessions and applied it to analyze 26 sessions by 13 data science professionals. In response to **RQ1**, we uncovered distinct temporal patterns in analysts’ **Observations**, revealing how different types of insights evolve throughout an EDA session. We identified phenomena such as *attribute-addition* and *reasoning in the language of interaction*, which shed light on the cognitive processes underlying EDA in computational notebooks. Addressing **RQ2**, our analysis uncovered substantial differences in how analysts use interactive versus static visualizations. Interactive visualizations often led to earlier discoveries of relationships between dataset attributes. Analysts also tended to rely heavily on a small subset of representations, with interactive visualizations comprising a sizeable portion of this subset. Finally, we introduce metrics such as **revisitCount**, **representationalDiversity**, and **representationalVelocity** to quantify broad coverage in EDA. Our work contributes to calls for investigating the theoretical foundation of EDA [22] and offers principles for designing more analyst-aligned EDA tools.

7.1 Limitations

Although our approach yielded useful insights about how data science professionals analyze data, we note that studying EDA in a laboratory context poses some inherent limitations. For example, think-aloud protocols may artificially structure thought processes that are more fluid in unobserved settings (e.g., participants may prioritize tasks that are easier to articulate) [11]. However, in comparison to post hoc reflections, thinking aloud provided *in situ* insights that captured important nuance, and aligns with approaches used in other studies [3].

Our study’s sample (N=13) may not fully represent the diversity of approaches to EDA. However, this size aligns with qualitative research practices that prioritize depth over breadth [1]. Thematic saturation observed in our data also suggests that the identified themes provide robust insights into the EDA process.

The 25-minute time limit per analysis may have also constrained the range of analyses participants engaged in. This time limit, consistent with prior visualization studies [6, 54, 56], balances the need to maintain participant engagement without requiring extended time commitments. Research shows that analysts often encounter time-sensitive tasks in their work [53], and in practice, we did not abruptly cut participants off. Thus, on average, participants took 29 minutes to complete an analysis.

Finally, using a new visualization library inevitably presents challenges to analysts and may introduce novelty effects, especially for those accustomed to static visualizations. We sought to mitigate these effects in two ways. First, we allocated 20-minutes to demonstrations and tutorials of the library. Second, ALX was intentionally designed as a visualization and interaction *typology* (as opposed to a more composable grammar) to minimize specification difficulty — with the terms of the two typologies designed to mirror common visualization and interaction design patterns. More importantly, introducing a new library allowed us to control for participant expertise, as analysts did not have prior tool-specific habits that could have confounded our comparison of analysis sessions. These sessions, therefore, reflect a “first-use study,” which is common in studies of EDA activity [24, 54, 56, 58].

7.2 Implications for EDA Tool Design

Our results suggest several opportunities for interactive visualization tooling to better support EDA. For instance, several of our participants engaged *touring* to systematically explore the data (§ 5.2). Yet, existing tools provide poor support for such activity, largely leaving analysts to drive interactions based on their priors and hypotheses they may wish to answer. Akin to visualization recommender systems [26], novel EDA tooling might instead leverage nascent grammars [48] to systematically enumerate the space of hypotheses that can be interactively reached with a given visualization, and proactively suggest particular analysis

paths. By leveraging information scent [52], such tools could help analysts think more deeply in the *language of interaction* (§ 6.5) — that is, even if an analyst did not adopt a suggestion for an interactive path, the suggestion itself may prompt them to think in different ways.

Relatedly, we found our participants’ use of visualizations as *action planning aids* (§ 6.2.2) striking. In computational notebooks, where visualizations are linearly presented, several participants were willing to pay a “scrolling tax” to reach these representations. While some research systems have explored mechanisms for making such representations more readily available (e.g., B2 stitches a visual analytics dashboard alongside a linear notebook [56]), our results suggest a wider opportunity. For instance, although research has identified the merit of overview+detail or focus+context techniques, few visualization libraries support them out-of-the-box. When they do, these techniques are supported in relatively limited ways (e.g., when panning/zooming a scatterplot or map). Our results suggest the need for more generalized support for wayfinding — especially to coordinate multiple separate visualizations. Here, we find the *interaction snapshots* [55] and *EDA assistant* [29] particularly promising for displaying the range of plausible next actions, and enabling quick probing of the analysis space.

The prevalence of **Process** utterances during analysis sessions illustrates that participants engage in a level of metacognition — thinking about their own thinking. How might visual analysis tools better support process reflections across visualization creation, interaction design, code, and statistical output? Drawing on research in distributed cognition [20], we envision that displays of analysis histories could foster valuable self-reflection. Systems like Lumos [29, 35] are already exploring this, highlighting a rich research space. For example, what marks a significant point in the analytical journey? While our formalism points to **Observations** and **Representation** creation as key moments, analysts may have different views when reflecting on their own activity.

7.3 Studying Interactive Analysis as Situated Activity

Our work was motivated by a desire to study interaction as *situated activity* — that is, involving human analysts working in a particular context, externalizing their cognition through visual representations, and interactively making observations with them. While valuable, we believe this paper takes only an initial step towards this approach. To complement recent work that looks to scale-up our ability to study interaction (e.g., through benchmarks [16] and novel systems [12, 36]), we advocate for methods that allow us to study it *more closely*.

We find methods from sociolinguistics and linguistic anthropology used to analyze interpersonal interaction particularly compelling. For instance, discourse and conversational analysis [43] involves a meticulous examination of conversation transcripts, and has been used by researchers to make fundamental linguistic discoveries such as turn-taking [43]. While visualization researchers are beginning to draw on such linguistic theories to inform interaction design guidelines [46, 47], we believe there is a ripe opportunity to adapt them for analyzing interactive behavior as well. For instance, the development of a specialized notation system was particularly crucial to the success of conversational analysis — allowing researchers to annotate linguistic features such as prosody, tone, pitch, pauses, and gaze. What would an equivalent notation for analyzing interaction look like? Similarly, systems for conversational analysis enable flexible definitions of analytic units and abstractions. In contrast, existing interaction provenance systems [28] largely follow a dichotomy of either low-level event logs (e.g., mouse movements, clicks, etc.) or high-level semantically meaningful events (e.g., filter, explore, etc.) — future systems must grapple with how to support more fluid analysis between these levels. Finally, as our study demonstrates, to “closely read” interactive behavior requires capturing a rich multimodal data streams. Simply concatenating and visually linking these streams together risks introducing ambiguities in understand the precise sequences and potential causal relationships between measures. Rather, akin to systems like ChronoViz [14], we envision future systems offering richer juxtapositions of this multimodal data.

8 ACKNOWLEDGEMENTS

We thank our study participants and anonymous reviewers for their thoughtful comments. This work was supported by NSF grants #1942659 and #1900991. This material is based upon work supported by the National Science Foundation under Grant No. 2141064.

REFERENCES

- [1] Using the TACT framework to learn the principles of rigour in qualitative research. 17. doi: 10.34190/JBRM.17.3.002 9
- [2] S. S. Alam and R. Jianu. Analyzing Eye-Tracking Information in Visualization and Data Space: From Where on the Screen to What on the Screen. *IEEE Transactions on Visualization and Computer Graphics*, 23(5):1492–1505, May 2017. Conference Name: IEEE Transactions on Visualization and Computer Graphics. doi: 10.1109/TVCG.2016.2535340 2
- [3] R. Arias-Hernandez, L. T. Kaastra, and B. Fisher. Joint Action Theory and Pair Analytics: In-vivo Studies of Cognition and Social Interaction in Collaborative Visual Analytics. *Proceedings of the 33rd Annual Conference of the Cognitive Science Society*, 2011. 9
- [4] O. Bar El, T. Milo, and A. Somech. Automatically Generating Data Exploration Sessions Using Deep Reinforcement Learning. In *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*, SIGMOD '20, pp. 1527–1537. Association for Computing Machinery, New York, NY, USA, May 2020. doi: 10.1145/3318464.3389779 2
- [5] A. Batch and N. Elmqvist. The Interactive Visualization Gap in Initial Exploratory Data Analysis. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):278–287, Jan. 2018. doi: 10.1109/TVCG.2017.2743990 1, 3
- [6] L. Battle and J. Heer. Characterizing Exploratory Visual Analysis: A Literature Review and Evaluation of Analytic Provenance in Tableau. *Computer Graphics Forum*, 38(3):145–159, June 2019. doi: 10.1111/cg.13678 1, 2, 5, 8, 9
- [7] L. Battle and A. Ottley. A Programmatic Definition of Visualization Insights, Objectives, and Tasks, Oct. 2022. arXiv:2206.04767 [cs]. 1, 5
- [8] L. Bavoil, S. Callahan, P. Crossno, J. Freire, C. Scheidegger, C. Silva, and H. Vo. VisTrails: Enabling Interactive Multiple-View Visualizations. In *VIS 05. IEEE Visualization*, 2005., pp. 135–142. IEEE, Minneapolis, MN, USA, 2005. doi: 10.1109/VISUAL.2005.1532788 1
- [9] A. Boggust, B. Carter, and A. Satyanarayan. Embedding Comparator: Visualizing Differences in Global Structure and Local Neighborhoods via Small Multiples. In *27th International Conference on Intelligent User Interfaces*, pp. 746–766. ACM, Helsinki Finland, Mar. 2022. doi: 10.1145/3490099.3511122 2, 3
- [10] J. Boy, L. Eveillard, F. Detienne, and J.-D. Fekete. Suggested Interactivity: Seeking Perceived Affordances for Information Visualization. *IEEE transactions on visualization and computer graphics*, 22(1):639–648, Jan. 2016. doi: 10.1109/TVCG.2015.2467201 8
- [11] S. Davies. The Cognitive Psychology of Planning. In *Planning and problem solving in well-defined domains*, p. 43. The Psychology Press, 2005. 9
- [12] D. Dotan, P. Pinheiro-Chagas, F. A. Roumi, and S. Dehaene. Track It to Crack It: Dissecting Processing Stages with Finger Tracking. *Trends in Cognitive Sciences*, 23(12):1058–1070, Dec. 2019. Publisher: Elsevier. doi: 10.1016/j.tics.2019.10.002 9
- [13] M. Feng, E. Peck, and L. Harrison. Patterns and Pace: Quantifying Diverse Exploration Behavior with Visualizations on the Web. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):501–511, Jan. 2019. Conference Name: IEEE Transactions on Visualization and Computer Graphics. doi: 10.1109/TVCG.2018.2865117 2
- [14] A. Fouse, N. Weibel, E. Hutchins, and J. D. Hollan. ChronoViz: a system for supporting navigation of time-coded data. In *CHI '11 Extended Abstracts on Human Factors in Computing Systems*, pp. 299–304. ACM, Vancouver BC Canada, May 2011. doi: 10.1145/1979742.1979706 9
- [15] K. Gadhave, Z. Cutler, and A. Lex. Persist: Persistent and Reusable Interactions in Computational Notebooks, Dec. 2023. doi: 10.31219/osf.io/9x8eq 3
- [16] S. Gathani, S. Monadjemi, A. Ottley, and L. Battle. A Grammar-Based Approach for Applying Visualization Taxonomies to Interaction Logs, Apr. 2022. arXiv:2201.03740 [cs]. 9
- [17] H. Guo, S. R. Gomez, C. Ziemkiewicz, and D. H. Laidlaw. A Case Study Using Visualization Interaction Logs and Insight Metrics to Understand How Analysts Arrive at Insights. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):51–60, Jan. 2016. doi: 10.1109/TVCG.2015.2467613 2
- [18] S. G. Hart and L. E. Staveland. Development of nasa-tlx (task load index): Results of empirical and theoretical research. *Human mental workload*, 1(3):139–183, 1988. 3
- [19] J. Heer and B. Shneiderman. Interactive dynamics for visual analysis. *Communications of the ACM*, 55(4):45–54, Apr. 2012. doi: 10.1145/2133806.2133821 1
- [20] W. C. Hill, J. D. Hollan, D. Wroblewski, and T. McCandless. Edit wear and read wear. In *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '92*, pp. 3–9. ACM Press, Monterey, California, United States, 1992. doi: 10.1145/142750.142751 9
- [21] H.-F. Hsieh and S. E. Shannon. Three approaches to qualitative content analysis. *Qualitative health research*, 15(9):1277–88, Nov. 2005. doi: 10.1177/1049732305276687 2, 3
- [22] J. Hullman and A. Gelman. Designing for Interactive Exploratory Data Analysis Requires Theories of Graphical Inference. *Harvard Data Science Review*, July 2021. doi: 10.1162/99608f92.3ab8a587 2, 9
- [23] E. Jun, M. Birchfield, N. De Moura, J. Heer, and R. Just. Hypothesis Formalization: Empirical Findings, Software Limitations, and Design Implications. *ACM Transactions on Computer-Human Interaction*, 29(1):1–28, Feb. 2022. doi: 10.1145/3476980 5
- [24] A. Kale, Z. Guo, X. L. Qiao, J. Heer, and J. Hullman. EVM: Incorporating Model Checking into Exploratory Visual Analysis, Aug. 2023. arXiv:2308.13024 [cs]. 1, 3, 5, 9
- [25] S. Kandel, A. Paepcke, J. M. Hellerstein, and J. Heer. Enterprise Data Analysis and Visualization: An Interview Study. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2917–2926, Dec. 2012. doi: 10.1109/TVCG.2012.219 2, 5
- [26] U. I. D. Lab. Draco: Representing, Applying & Learning Visualization Design Guidelines, Oct. 2018. Library Catalog: medium.com. 9
- [27] D. J.-L. Lee, D. Tang, K. Agarwal, T. Boonmark, C. Chen, J. Kang, U. Mukhopadhyay, J. Song, M. Yong, M. A. Hearst, and A. G. Parameswaran. Lux: Always-on Visualization Recommendations for Exploratory Dataframe Workflows, Dec. 2021. arXiv:2105.00121 [cs]. 5
- [28] A. Lex. Opportunities for Understanding Semantics of User Interactions. In *Workshop – Machine Learning from User Interactions*, Oct. 2021. 9
- [29] X. Li, Y. Zhang, J. Leung, C. Sun, and J. Zhao. EDAssistant: Supporting Exploratory Data Analysis in Computational Notebooks with In Situ Code Search and Recommendation. *ACM Transactions on Interactive Intelligent Systems*, 13(1):1:1–1:27, Mar. 2023. doi: 10.1145/3545995 2, 9
- [30] Z. Liu and J. Heer. The Effects of Interactive Latency on Exploratory Visual Analysis. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):2122–2131, Dec. 2014. doi: 10.1109/TVCG.2014.2346452 2
- [31] M. Lombard, J. Snyder-Duch, and C. Bracken. Practical resources for assessing and reporting intercoder reliability in content analysis research projects. (2002):1–18, 2004. 4
- [32] E. Marsh and M. White. Content analysis: A flexible methodology. *Library trends*, 55(1):22–45, 2006. doi: 10.1353/lib.2006.0053 3
- [33] A. Mosca, A. Ottley, and R. Chang. Does Interaction Improve Bayesian Reasoning with Visualization? In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pp. 1–14. ACM, Yokohama Japan, May 2021. doi: 10.1145/3411764.3445176 1
- [34] M. Muller. Curiosity, Creativity, and Surprise as Analytic Tools: Grounded Theory Method. In J. S. Olson and W. A. Kellogg, eds., *Ways of Knowing in HCI*, pp. 25–48. Springer, New York, NY, 2014. 3
- [35] A. Narechania, A. Coscia, E. Wall, and A. Endert. Lumos: Increasing Awareness of Analytic Behavior during Visual Data Analysis. *IEEE Transactions on Visualization and Computer Graphics*, 28(1):1009–1018, Jan. 2022. arXiv:2108.02909 [cs]. doi: 10.1109/TVCG.2021.3114827 9
- [36] C. Nobre, D. Wootton, Z. Cutler, L. Harrison, H. Pfister, and A. Lex. reVISit: Looking Under the Hood of Interactive Visualization Studies. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pp. 1–13. ACM, Yokohama Japan, May 2021. doi: 10.1145/3411764.3445382 1, 2, 9
- [37] C. Nobre, D. Wootton, L. Harrison, and A. Lex. Evaluating Multivariate Network Visualization Techniques Using a Validated Design and Crowdsourcing Approach. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pp. 1–12. ACM, Honolulu HI USA, Apr. 2020. doi: 10.1145/3313831.3376381 2
- [38] D. Norman. *The Design Of Everyday Things*. Basic Books, New York, New York, revised edition ed., Nov. 2013. 8

- [39] C. North. Toward measuring visualization insight. *IEEE Computer Graphics and Applications*, 26(3):6–9, May 2006. Conference Name: IEEE Computer Graphics and Applications. doi: 10.1109/MCG.2006.70 2
- [40] J. Pearl. *Heuristics: Intelligent Search Strategies for Computer Problem Solving*. Addison-Wesley, 1984. 8
- [41] X. Pu and M. Kay. How Data Analysts Use a Visualization Grammar in Practice. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, pp. 1–22. ACM, Hamburg Germany, Apr. 2023. doi: 10.1145/3544548.3580837 3
- [42] S. Robinson and A. L. Mendelson. A Qualitative Experiment: Research on Mediated Meaning Construction Using a Hybrid Approach. *Journal of Mixed Methods Research*, 6(4):332–347, Oct. 2012. doi: 10.1177/1558689812444789 2, 3
- [43] H. Sacks, E. A. Schegloff, and G. Jefferson. A Simplest Systematics for the Organization of Turn-Taking for Conversation. *Language*, 50(4):696–735, 1974. Publisher: Linguistic Society of America. doi: 10.2307/412243 9
- [44] A. Sarvghad, M. Tory, and N. Mahyar. Visualizing Dimension Coverage to Support Exploratory Analysis. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):21–30, Jan. 2017. doi: 10.1109/TVCG.2016.2598466 2, 3
- [45] A. Satyanarayan, D. Moritz, K. Wongsuphasawat, and J. Heer. Vega-Lite: A Grammar of Interactive Graphics. p. 10. 3
- [46] V. Setlur, M. Correll, A. Satyanarayan, and M. Tory. Heuristics for Supporting Cooperative Dashboard Design, Aug. 2023. arXiv:2308.04514 [cs]. 9
- [47] V. Setlur and M. Tory. How do you converse with an analytical chatbot? revisiting gricean maxims for designing analytical conversational behavior, 2022. 9
- [48] A. Suh, Y. Jiang, A. Mosca, E. Wu, and R. Chang. A Grammar for Hypothesis-Driven Visual Analysis, Apr. 2022. arXiv:2204.14267 [cs]. 9
- [49] S. Theis, C. Bröhl, M. Wille, P. Rasche, A. Mertens, E. Beauxis-Aussalet, L. Hardman, and C. M. Schlick. Ergonomic Considerations for the Design and the Evaluation of Uncertain Data Visualizations. In S. Yamamoto, ed., *Human Interface and the Management of Information: Information, Design and Interaction*, vol. 9734, pp. 191–202. Springer International Publishing, Cham, 2016. Series Title: Lecture Notes in Computer Science. doi: 10.1007/978-3-319-40349-6_19 1
- [50] J. Thomas and K. Cook. Illuminating the Path: Research and Development Agenda for Visual Analytics. Technical report, National Visualization and Analytics Center, 2005. 1
- [51] J. van Wijk. The value of visualization. In *VIS 05. IEEE Visualization, 2005.*, pp. 79–86, Oct. 2005. doi: 10.1109/VISUAL.2005.1532781 1
- [52] W. Willett, J. Heer, and M. Agrawala. Scented Widgets: Improving Navigation Cues with Embedded Visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1129–1136, Nov. 2007. doi: 10.1109/TVCG.2007.70589 9
- [53] K. Wongsuphasawat, Y. Liu, and J. Heer. Goals, Process, and Challenges of Exploratory Data Analysis: An Interview Study, Nov. 2019. arXiv:1911.00568 [cs]. 3, 9
- [54] K. Wongsuphasawat, Z. Qu, D. Moritz, R. Chang, F. Ouk, A. Anand, J. Mackinlay, B. Howe, and J. Heer. Voyager 2: Augmenting Visual Analysis with Partial View Specifications. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pp. 2648–2659. ACM, Denver Colorado USA, May 2017. doi: 10.1145/3025453.3025768 1, 2, 5, 8, 9
- [55] Y. Wu, R. Chang, J. M. Hellerstein, and E. Wu. Facilitating Exploration with Interaction Snapshots under High Latency, Sept. 2020. arXiv:1806.01499 [cs]. 9
- [56] Y. Wu, J. M. Hellerstein, and A. Satyanarayan. B2: Bridging Code and Interactive Visualization in Computational Notebooks. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*, pp. 152–165. ACM, Virtual Event USA, Oct. 2020. doi: 10.1145/3379337.3415851 3, 9
- [57] C. Yan and Y. He. Auto-Suggest: Learning-to-Recommend Data Preparation Steps Using Data Science Notebooks. In *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*, pp. 1539–1554. ACM, Portland OR USA, June 2020. doi: 10.1145/3318464.3389738 2
- [58] E. Zraggen, Z. Zhao, R. Zeleznik, and T. Kraska. Investigating the Effect of the Multiple Comparisons Problem in Visual Analysis. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, pp. 1–12. ACM, Montreal QC Canada, Apr. 2018. doi: 10.1145/3173574.