Cover Sheet

Title: CAREER: Effective Interaction Design in Data Visualization

PI: Arvind Satyanarayan, NBX Career Development Assistant Professor, MIT
E-Mail: arvindsatya@mit.edu

CAREER: Effective Interaction Design for Data Visualization

Society's broad adoption of data visualization has been driven, in part, by decades of research developing theories of effective visual encoding, and instantiating them in software systems to lower the threshold for authoring visualizations. However, there has been little analogous theory-building for interactivity — a feature widely thought to be critical for effective visualization as it enables a tighter feedback loop between generating and answering hypotheses. Comparative empirical studies of interactivity are rare, and have been conducted in an ad hoc fashion; and, interaction taxonomies do not provide design guidelines such as how to pick between different techniques for a given task. For instance, does the data distribution affect whether zooming should occur via buttons, continuous scrolling, or brush-to-zoom? Similarly, how should interactive filters be depicted (e.g., highlighting selected points, dimming unselected points, or removing them from the chart altogether) to more easily perceive trends in the data? And, critically, how do these interaction design choices affect dataset coverage, the rate of insights, and people's confidence?

This lack of theory has also impeded support for interaction design in visualization systems. While recent work has explored higher-level abstractions for authoring interactivity, users must still manually invoke and wire the necessary components together, with little of the guidance and support that accompanies the visual encoding process. Moreover, with no established conventions for interaction design, authors and consumers must contend with inconsistent and unreliable experiences — for instance, by default, dragging may pan the chart, highlight brushed points, or zoom into the selected region depending on the tool used.

The proposed work will begin to develop a theory of effective interaction design for data visualization, and will leverage it to enable new methods of specifying interactive behaviors. We propose to systematically study the interaction design space using the PI's Vega-Lite visualization grammar to enumerate candidate points. By conducting a mix of large-scale crowdsourced experiments and in-lab studies, we will evaluate the lower-level features of interaction design (e.g., usability) in context (e.g., data science notebooks, interactive articles, etc.) and tie interaction design choices to higher-level cognition such as hypothesis generation. Study results will be codified in computational models (i.e., an interaction recommender system) to enable exploration of new automated and mixed-initiative interfaces for authoring interactivity.

Intellectual Merit: The proposed work will contribute the first systematic, empirically-validated effectiveness rankings for interaction techniques in data visualization. Critically, these rankings will be conditioned on dataset characteristics, analytic task, and context, and will account for both low-level properties (e.g., accuracy and time-taken) and higher-level features such as query formulation and learnability (e.g., as determined by interaction traces). We will codify these rankings in a recommender system to suggest effective interaction techniques for a given dataset and task, as well as unexplored interactive states. Finally, we will explore how the recommender system allows us to operationalize our effective rankings by developing new methods for authoring interactivity, and designing novel perceived affordancces for interactive visualization.

Broader Impacts: If successful, the proposed work will establish best practices for interaction design in data visualization, and significantly lower the barrier to entry for authoring interactive data visualizations. To support real-world adoption of our results, we will open source our software contributions, integrate them with popular data science environments (e.g., Jupyter and Observable), and host workshops and tutorials for practitioners at appropriate venues (e.g., OpenVis Conf and ODSC). We will incorporate our research results into classes the PI teaches at MIT, developing new material to teach interaction design through faded worked examples and rapid prototyping exercises. Moreover, we will prioritize providing research opportunities for women, underrepresented populations, and undergraduate students.

Keywords: visualization; interaction; empirical studies; design; data science; computational notebooks

Project Description

1 Introduction

Data visualization has gone mainstream — from business intelligence to data-driven journalism, society has embraced visualization as a medium for recording, analyzing, and communicating data. Although visualization has been practiced for centuries[1], the success of modern visualization rests, in part, on decades of research developing theories of effective visual encoding. The need for this theory was prominently articulated by statisticians in the 1970s, including William Kruskal who noted that *"in choosing, constructing, and comparing graphical methods we have little to go on but intuition, rule of thumb, and a kind of master-to-apprentice passing along of information"* [53]. In response, in a seminal 1984 paper, statisticians William Cleveland and Robert McGill began to define a theory of visual encoding by identifying a set of elementary perceptual tasks people perform when reading charts, and then conducting experiments to order the tasks based on how accurately participants performed them [22].

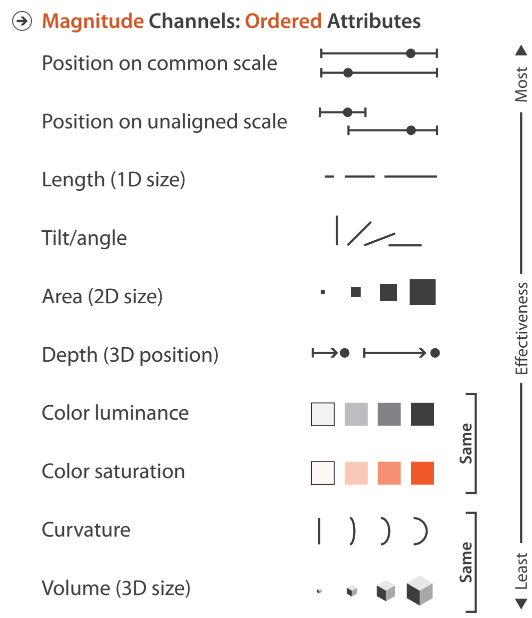


Figure 1: An excerpt of rankings of visual encoding effectiveness [62], first empirically validated by Cleveland & McGill [22].

Cleveland & McGill's theory-building has had a transformative effect on the field. As they note, *"a theory is testable"* and the subsequent body of graphical perception studies has refined and expanded their initial rankings [32,33,49,87], as well as questioned expert wisdom on avoiding pie charts [52] and "chart junk" [9, 17]. And, instantiating this theory in software systems has lowered the barrier to entry for visualization design — people no longer need to be experts in, nor have an intuition for, visualization design but can rather rely on recommender systems to guide them towards effective design choices [58, 61, 93].

Although the research community has long articulated the value of interactivity in supporting a *"dialogue between the analyst and the data"* [89], current visualization theory has focused almost exclusively on visual encodings [27, 28, 45, 90, 96]. Comparative empirical studies of interaction techniques are rare [48, 83]; those that exist have been largely conducted in a piecemeal and ad hoc fashion, making it difficult to gain a broad overview of interaction effectiveness. Taxonomies of analysis tasks, interaction techniques, and cognitive costs have been proposed [4, 19, 34, 54, 77, 96] but have never been empirically validated to produce effectiveness rankings. For instance, are the different means of depicting interactive filters (e.g., highlighting selected points, dimming unselected points, or removing the latter from the visualization altogether) equally usable, and do they have any impact on higher-level goals such as identifying trends or formulating hypotheses? Without analogous theory on effectiveness, current interaction design practice suffers from many of the concerns of early visual encoding: there is a high barrier to entry for authoring interactivity, and it is driven by largely unscientific and unstructured intuition.

This project proposes to begin developing **a theory of effective interaction design for data visualization**, and exploring how it can be operationalized to enable **new forms of authoring interactivity that lower the barrier to entry**. In particular, we expect the project to unfold across three phases:

[1] The first bar charts and line charts are credited to William Playfair, appearing in his 1786 work *Commercial and Political Atlas*.

(1) Empirically Derive Effectiveness Criteria. We will design and conduct controlled studies to develop measures for comparing alternative interactive designs conditioned on data, task, and context. Akin to graphical perception studies, we will use large-scale crowdsourced experiments to study the "lower-level" features of an interaction technique including usability, task completion time, and accuracy. Critically, we will complement these experiments with in-lab studies to evaluate interactivity in context (including within data science notebooks and interactive articles) and understand how it impacts "higher-level" cognition such as hypothesis generation and downstream consumption of data. For instance, how does an interactive visualization change subsequent analysis in a notebook, or does data comprehension increase for readers who have interacted with a visualization embedded in an article?

(2) Codifying Effectiveness Criteria. To study the implications of effectiveness rules for interaction design, we will first codify the results from the previous phase of research into a new interaction recommender system. This system should not only recommend effective interaction techniques for a given dataset, analysis task, and visual encodings, but should also suggest unexplored interactive states. To do so, we will design a novel task-based representation of interaction design, express our effectiveness criteria through a system of logical facts and formal constraints, and incorporate machine learning methods to mine interaction traces.

(3) Operationalizing Effectiveness Criteria. We will explore how our effectiveness criteria, through our recommender system, enable novel visualization interfaces and techniques. For instance, we will expose recommended interactive states through new minimap displays, and adapt information scent techniques to design new perceived affordances for interactive visualization. And, with recommended interaction techniques, we will develop new approaches for authoring interaction through demonstration and automatically augmenting static visualizations with interactivity by inferring analysis tasks in data science notebooks.

1.1 Intellectual Merit

This project will contribute the first *testable theory* of interaction design in data visualization. This theory will comprise measures to rank alternative interaction techniques conditioned on dataset characteristics, analytic task, and overall context, and account for both lower-level usability concerns as well as higher-level implications on user goals and workflows. By building an interaction recommender system, this project will also demonstrate how to *operationalize this theory*. In particular, we will design new perceived affordances for interactivity in data visualization, as well as new approaches for authoring interactive behaviors including through mixed-initiative interfaces and methods that automatically augment static visualizations with interactivity by inferring task context and user goals. If successful, this project will serve as the foundation for future work that systematically studies the interaction design space, mirroring the success of graphical perception studies in the visual encoding space.

1.2 Broader Impacts

Through this work, we aim to shift interaction design practice away from being guided purely by intuition and towards being grounded in empirically-validated principles. And, in doing so, we seek to make authoring *interactive* visualizations more broadly accessible, particularly for non-technical and non-expert audiences. To support these goals, we will (1) open source our software contributions; (2) engage in collaborations (letters attached) to integrate them with popular data science environments including Jupyter and Observable; (3) host workshops and tutorials at practitioner-oriented venues (e.g., OpenVis Conf, ODSC, and Information+); (4) integrate both theory and systems results into data visualization and human-computer interaction classes the PI teaches at MIT (detailed plans are in § 4.1); and (5) as described in § 4.2, we will prioritize providing research opportunities for women, underrepresented populations, and undergraduates.

1.3 PI Qualifications and Career Objectives

The PI is poised to conduct this research based on his award-winning prior work developing toolkits [37, 80–82] and design systems [78, 79] for visualization with a focus on abstractions for interaction design. This work will be central to achieving both the intellectual merits and broader impacts of this project: the Vega-Lite grammar [80] provides the first representation of interactive visualization that can systematically enumerate the interaction design space — a critical property for phase one of the project; the Lyra visualization design environment [79] offers a platform for developing mixed-initiative approaches for interaction design (phase three); and the thriving community around these tools (including wide adoption in the Jupyter data science community, on Wikipedia, and at companies such as Apple and Google) provides not only a large potential participant pool for phase one, but also an avenue for more immediate uptake of our results.

Moreover, the PI has an established track record of outreach to underrepresented and non-academic populations. In his first year at MIT, a majority of his 13-member group are women and/or people of color. The PI has served on the diversity committees for several conferences, and is a diversity co-chair for IEEE VIS 2019, responsible for the diversity scholarship. He has a track record of speaking at practitioner venues (including repeated appearances at OpenVis Conf and OSDC East) and has also served in these communities (on the OpenVis Conf 2017 program committee, and as conference co-chair in 2018).

An Early Career Award will enable the PI to expand his research agenda to cover new categories of visualization contributions[2]: while his doctoral work focused exclusively on visualization systems, this project makes contributions in empirical studies, theories & models, and techniques & algorithms. This project, and the collaborations it spurs, also lays a foundation for the PI to generalize insights on formally modeling and empirically evaluating interactivity from visualization to the broader human-computer interaction space.

2 Background and Related Work

2.1 Graphical Perception Studies

In his seminal 1967 work *Semiologie Graphique*, French cartographer Jacques Bertin laid the foundation for a theory of data visualization by proposing that visualization is a process of mapping data variables to the *visual variables* (or *channels*) of graphical primitives called *marks*, and that these channels had an effectiveness ordering [15]. By conducting human-subjects experiments, statisticians Cleveland & McGill were the first to empirically validate these rankings (their elementary perceptual tasks roughly map to Bertin's visual variables), and Heer & Bostock later replicated and extended this work with crowdsourced participants [33]. In this work, effectiveness is defined as how accurately participants perceive encoded proportions by data type (nominal, ordinal, or quantitative). More recent work has expanded this definition to include *ensemble* encodings [88] (i.e., encodings of more than one visual object where participants perceive averages [2, 30] or correlation [32, 44]). Thus, effectiveness rankings are now conditioned on task and dataset characteristics [50, 77]. *However, this body of work has only considered static visualizations, and a similar systematic and rigorous approach has not yet been applied to determine the effectiveness of interactive behaviors.*

2.2 Studies of Interaction Design

In the data visualization literature, interaction techniques have typically been evaluated at the point of their development. Researchers have primarily focused on validating internal design decisions [24, 60, 86] and if comparative studies are conducted, they focus on at most a handful of related techniques [48, 51]. As a result, *it is difficult to gain a broad and systematic overview of interactivity in data visualization*. The

[2] `http://ieeevis.org/year/2019/info/call-participation/infovis-paper-types`

human-computer interaction (HCI) literature provides exemplars of comparative evaluations of interactive behaviors, with a rich body of work comparing techniques for pointing [3, 5, 8, 31, 64], scrolling [36, 70, 98, 99], and selecting [14, 23]. However, this work has primarily focused on low-level properties of interaction design — namely, the speed of performing an interaction and often through the lens of Fitt's Law [57] — and has *rarely studied the implications of interaction on higher-level cognition*. For instance, does the scrolling mechanism affect how likely a participant is to complete reading a document? Similarly, do particular pointing techniques spur collaboration in multi-user settings? It has been difficult to make progress on this dimension in HCI as the design space is insufficiently structured or constrained — techniques such as pointing, scrolling, and selecting can be instantiated in myriad, diverse interfaces. By contrast, *data visualization provides a more structured design space to evaluate both lower-level properties and higher-level implications of interactivity* as researchers have developed taxonomies of analysis tasks and interaction techniques at varying levels of abstraction [4, 19, 34, 77, 96] as well as their associated costs [54].

2.3 Models of Interaction Design

HCI pioneer Michel Beaudouin-Lafon has proposed [11, 12] that interaction models should provide three properties: (1) *descriptive power*, "to describe a significant range of existing interfaces"; (2) *evaluative power*, "to help assess multiple design alternatives"; and (3) *generative power*, "to help designers create new designs". This lens is useful for understanding why it has been difficult to build a theory of effective interaction design: generative power is critical for isolating specific properties and systematically enumerating alternative designs to test. With visual encodings, Bertin's visual variables and Cleveland & McGill's elementary perceptual tasks provided this power: experimental conditions could be systematically generated by varying one channel and holding the others constant. Thus far, we have lacked an equivalent generative representation of interaction design, which has typically occurred through low-level imperative event handling programming [63]. As a result, interaction techniques have been implemented in isolation and researchers have had to reverse engineer [70] and reimplement them [83] in order to conduct ad hoc comparative evaluations. *The PI's past work with the Vega-Lite visualization grammar [80] provides the first generative representation of interaction design.* For a constant set of visual encodings, Vega-Lite allows us to not only systematically enumerate interaction techniques (e.g., filtering, panning, zooming) but also vary their constituent properties (e.g., what events trigger the interaction, or how is it depicted on the visualization).

3 Research Plan and Intellectual Merit

To begin developing a theory of effective interaction design in data visualization, this project proposes three phases of research. First, we will conduct empirical comparative studies to evaluate the low-level usability of different alternative designs for common interaction techniques (e.g., panning, zooming, and filtering) and to study the implications of interactivity on higher-level cognition such as query formulation and insight generation. To study their implications, we will codify these results in an interaction recommender system that is able to suggest effective techniques for a given dataset and task, and is also able to recommend unexplored interactive states. Finally, we will explore how our theory, via the recommender system, enables new visualization techniques and interfaces that help lower the threshold for interaction design.

3.1 Phase One: Empirical Studies of Interaction Design

The goal of phase one is to begin to do for interaction design what Cleveland & McGill's early studies, and the subsequent rich body of graphical perception studies, have done for visual encoding: develop effectiveness rankings. We define the effectiveness of an interaction technique by drawing on cognitive science theory: Hutchins, Hollan, & Norman identify that an interactive behavior is most successful when it bridges the *gulf of execution*, or how easily can a user operate a given technique, and the *gulf of evaluation*, or how well does the technique accomplish the user's goals [41]. Heidi Lam proposes that an additional *gulf of formulation* is necessary to capture the cost of reacting to output and formulating subsequent new goals during interactive data analysis and discovery [54]. In this phase, we will empirically evaluate these three gulfs through a series of controlled studies.

To evaluate the gulfs of execution and evaluation, we will isolate individual interaction techniques (e.g., zooming or filtering) and vary their constituent properties (e.g., what input events trigger them, how are they displayed on the visualization, etc.) for a constant set of visual encodings. We will measure the accuracy, time taken, and number of interactions (i.e., input events) needed to complete tasks from Amar et al.'s taxonomy [4] of primitive analysis tasks (e.g., retrieve value or characterize distribution). As we are assessing the lower-level *mechanics* of interactivity, we will conduct these studies by crowdsourcing participants, akin to graphical perception studies [33].

By contrast, to evaluate the gulf of formulation and impact on subsequent analysis goals, we must study interactivity *in context* and under a more open-ended process. Thus, we will conduct in-lab experiments with participants covering a broad range of data analysis expertise, and will ask them to engage with instrumented visualizations embedded in data science notebooks (e.g., Jupyter and Observable) and interactive articles. Participants will be asked to conduct tasks, such as explore the dataset, drawn from higher-level taxonomies [34, 96]. In addition to capturing quantitative measures (e.g., usability metrics, dataset coverage, etc.), we will ask participants to think aloud, and bookmark and caption interesting interactive states that generate insights or spur further questions — a protocol used to evaluate visual analytics systems [94, 95]. Through a qualitative coding procedure, we will characterize these insights using existing taxonomies to identify their complexity, depth, and relevance [67, 97]. Finally, we will record interaction traces [25, 35, 42] to determine patterns between interactive operations, higher-level cognition, and participant personality [20].

Recent work by Kim et al. has shown that visual encoding effectiveness is conditioned on dataset characteristics [50], and we expect the same to hold true for interaction effectiveness. Thus, following their setup, we expect to use a mixed-design: a within-subjects treatment for interaction techniques, and between-subjects treatments for task and data distributions balanced across subjects. To ensure ecological validity, we build on **the PI's preliminary work with VizNet**: a large-scale corpus of over 31 million real-world datasets, with which the Kim et al. results were replicated [40]. All study conditions will be constructed using **the PI's Vega-Lite visualization grammar** [80], which provides the first generative model of interaction design in data visualization: for constant visual encodings, we can enumerate interaction techniques and vary their constituent design elements systematically (see Figs. 2&3). Moreover, Vega-Lite provides us with a thriving user community which we can tap into for recruiting participants; doing so further promotes ecological validity by ensuring our studies account for real-world data analysis and visualization design expertise.

3.1.1 Example Empirical Studies

Here, we describe some specific studies we plan to conduct. We note that *this is not an exhaustive list*, but rather serves to provide concrete examples of the broader guiding principles discussed above.

(A) Pan & Zoom. Recent work by Schwab et al. [83] provides one of the rare examples of empirical evaluation of interaction techniques in data visualization: a comparative study of pan & zoom techniques in one-dimensional timelines. To do so, however, the authors had to re-implement the techniques they sought to compare in a custom toolkit. As Figure 2 depicts, a number of the design variations they consider, as well as several they did not, naturally fall out of the Vega-Lite interaction grammar. Thus, a first study we intend to run will *replicate* this prior study and *extend* it to evaluate additional designs (e.g., two-dimensional panning & zooming, multi-view techniques such as overview+detail, etc.) and the effect of varying dataset distributions. As in the original study, participants will be asked to *locate* a particular point (a task the authors

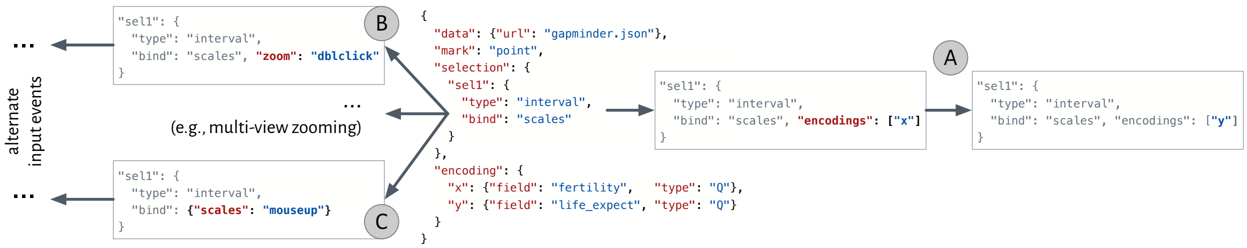


Figure 2: A systematic enumeration of alternate designs for interactive zooming using Vega-Lite [80] including (a) uni-dimensional zooming; (b) via alternate input events as described by the event stream selector syntax [82]; (c) brush-to-zoom functionality; and (not depicted) multi-view mechanisms such as overview+detail.

drew from Brehemer & Munzner's taxonomy [19], which roughly maps to *retrieve value* from Amar et al.) and we will measure usability metrics to assess the gulf of execution these techniques present.

With additional rounds of studies, we can also study the implications on the gulf of formation. In particular, by instantiating these techniques in context, and posing a broader more open-ended task for participants (e.g., explore the dataset), we can ask questions including how does the panning & zooming style affect dataset coverage and the rate participants make observations about the data? Or, might particular techniques be preferred within data science notebooks versus interactive articles? With this latter question, for instance, we might expect data scientists to prefer the precision of brush-to-zoom to explore a small cluster, but article readers may favor continuous zooming via the scroll wheel for the low activation energy it presents [91].

(B) Interactive Filtering. Dynamic query widgets (e.g., dropdown menus, range sliders, etc.) are frequently used to interactively filter a dataset [1], and Figure 3 illustrates a number of design variations that Vega-Lite yields for depicting their effect: driving conditional visual encoding for selected or unselected points, removing unselected points from the visualization, or showing selected points in a secondary view. Users perform the same action (manipulating input widgets) across all variants, but what they are attending to likely varies. As a result, how filters are depicted on the visualization may affect the difficulty of performing particular analysis tasks (i.e., the design variants may present different gulfs of evaluation). To evaluate this question, we will employ a mixed design with a within-subjects treatment for the design alternatives, and between-subjects assignments for different data distributions and across 5 tasks from Amar et al.: retrieve value, find extremum, determine range, characterize distribution, and correlate.

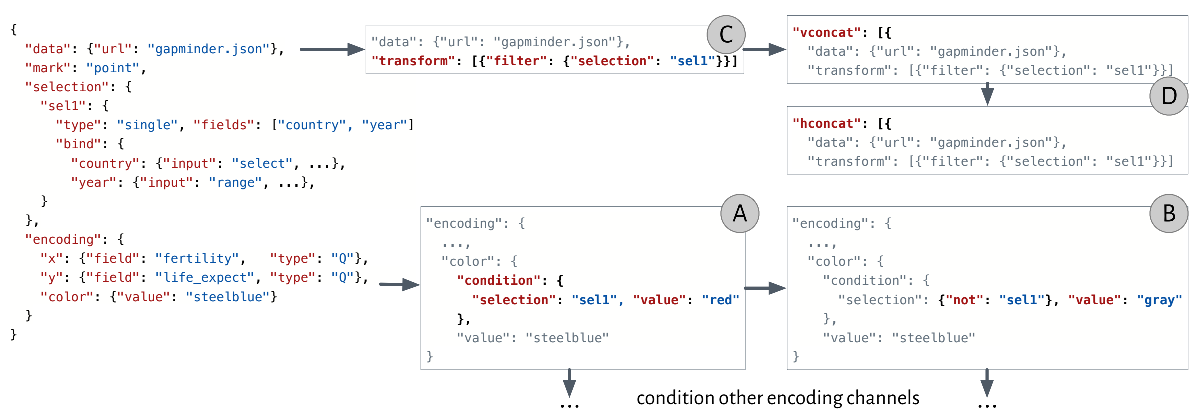


Figure 3: A systematic enumeration of alternate designs for depicting interactive filters using Vega-Lite [80] including (a) highlighting selected points; (b) dimming unselected points; varying (a) or (b) using other encoding channels (e.g., opacity, shape, size, etc.); (c) displaying only selected points; or, (d) displaying selected points in a secondary view.

(C) Multiplexing Space vs. Time. Comparing different subsets of data is a common analysis task [96], and a natural question to ask is whether comparisons should occur visually or temporally. Thus far, researchers have focused on a narrow portion of this problem space: comparing visual juxtaposition (or small multiples) with animation for graph data [6] or temporal trends [72]. The role of interactivity has been underexplored and applying our experimental framework yields a rich area of inquiry. For instance, how do the gulfs of execution and evaluation compare for static juxtapositioning, interactive juxtapositioning (i.e., brushing & linking [13]), and interactive superpositioning (i.e., visual highlights or dimming, informed by study B) on analysis tasks from Amar et al.? How does the type of variable (i.e., nominal, ordinal, temporal, etc.) used to partition the data impact these gulfs? And, critically, how do these variants affect the gulf of formation: does interactivity increase data coverage, the number of insights generated, or the number of hypotheses posed? Finally, by also testing animated variants — which will look identical to their interactive counterparts, but provide a more passive experience — we can empirically validate popular claims of interactivity allowing people to engage data in dialogue [89, 90], and the degree to which this is important.

3.1.2 Potential Risks and Alternative Approaches

In this phase of the project, there is a risk posed by the studies to assess higher-level cognition. Namely, whether there will be interaction patterns that generalize across participants, or whether each participant takes an idiosyncratic flow. To mitigate this risk, we adopt protocols successfully used by prior visual analysis systems [94, 95] and will begin with pilot studies to evaluate their feasibility for our use. Moreover, if developing new protocols is necessary, we will seek guidance from Drs. Michel Beaudoin-Lafon and Wendy Mackay, experts in conducting qualitative studies in human-computer interaction (letters attached).

3.2 Phase Two: Codifying Effectiveness Criteria

In this phase, we will investigate how to codify our empirically-derived effectiveness rankings in the design of an **interaction recommender system.** This system will provide the foundation for phase three (operationalizing effectiveness) by helping bridge all three gulfs: execution, evaluation, and formation. To do so, the system should be capable of not only **recommending specific interaction techniques** for a given dataset and visual encodings (thus narrowing the gulfs of execution and evaluation) but also **suggesting interactive states yet to be explored** to further a particular analysis goal (thus bridging the gulf of formation). We expect to break the design of the interaction recommender system down into two research questions.

3.2.1 How is the recommender system invoked?

Recommender systems suggest alternative methods of resolving ambiguities in the input they receive. As existing work shows, specifications of visual encodings can be extended in a fairly straightforward fashion to introduce the necessary ambiguity: users leave blank one or both sides of a data-visual mapping [59, 93, 95].

Ambiguous specification of interaction techniques, however, presents a more challenging problem. Vega-Lite decomposes interaction specification into roughly two components: what triggers the interaction (e.g., input events, HTML widgets, etc.) and what effect does it have (e.g., data transformation, conditional visual encoding, etc.). If we followed an approach akin to visual encoding, we would allow users to leave one of these components un- or under-specified and have the recommender system suggest possible completions. However, it is unclear whether tractable progress can be made here as this model of interactivity is focused purely on the *mechanics* of an interaction technique, and does not capture its *purpose* (i.e., the analysis task a user hopes to accomplish, and a critical conditioning variable for our effectiveness criteria).

Thus, a key first step will be **designing extensions to Vega-Lite to support specification of analysis tasks.** Critically, doing so **should not sacrifice its descriptive and generative powers** (§ 2.3) — e.g., how do lower-level tasks compose together into higher-level tasks? We will then **extend CompassQL** [93], a querying language that augments Vega-Lite with wildcards to denote ambiguities (or "holes") that a recommender should resolve. Figure 4 depicts two potential extensions, but careful and thorough exploration of the wildcard design space is necessary to **study the tradeoffs** between expressivity (i.e., the space of ambiguities), viscosity (i.e., the difficulty of navigating this space), and the tractability of making recommendations.

3.2.2 How does the system perform the recommendation?

To answer this question, we will **extend Draco** [61], a system that expresses effectiveness rankings for visual encodings via a system of constraints. Draco takes CompassQL specifications as input, compiles them to a set of logical facts, and then solves the constraints to return a set of suggested effective visualization designs. Moreover, Draco is capable of using results from empirical studies to tune weights associated with constraints; and thus, it provides a prime platform to codify our effectiveness rankings for interactivity.

A first consideration will be **how to represent interaction design as a set of logical facts.** Some of these facts may map directly to statements in our extended CompassQL or Vega-Lite (e.g., codifying analysis task composition) but, in other instances, these facts may express concerns not articulated by other representations. For example, a fact distinguishing discrete from continuous interactions (e.g., clicks or drags, respectively) may be a key property for particular analysis tasks but does not have an analogous statement in CompassQL or Vega-Lite. Once these facts are in place, we will **design novel constraints to express our effectiveness criteria**, and use Draco's existing facilities to tune their weights from our empirical results.

With these steps, for a given dataset and visual encodings, Draco will be capable of recommending effective interaction techniques for particular analysis tasks. Thus, the remaining challenge will be **how to model interaction traces** (captured in all phase one experiments) **to recommend interactive *states*.** Given their high-dimensional nature, we expect that constraints will not be the right paradigm to generalize over interaction traces — for instance, it is unclear how we would use constraints to discretize continuous interactions like brushing, panning, or zooming, to be able to suggest next states a user should visit. Instead, we expect to **adapt machine learning approaches** which have been shown to be able to predict personality traits [20] and future interaction for pre-fetching data [10] via support vector machines and Markov chains respectively. Critically, these models should not only work alongside Draco's constraints — such that, with a single ambiguous specification, Draco would recommend interaction techniques and ways to begin interacting — but they should **update their suggestions in near real-time** as a user interacts with the visualization.

A

```
{
  "data": {"url": "gapminder.json"},
  "mark": "point",
  "task": ["explore", "filter"],
  "selection": "?",
  "encoding": {
    "x": {"field": "fertility",   "type": "Q"},
    "y": {"field": "life_expect", "type": "Q"},
    "color": {"value": "steelblue"}
  }
}
```

B

```
{
  "data": {"url": "gapminder.json"},
  "mark": "point",
  "task": "?",
  "selection": {"type": "?", "on": [recording of input events]}
  "encoding": {
    "x": {"field": "fertility",   "type": "Q"},
    "y": {"field": "life_expect", "type": "Q"},
    "color": {"value": "steelblue"}
  }
}
```

Figure 4: Two possible extensions to CompassQL [93] (which implicitly depict potential extensions to Vega-Lite to incorporate analysis task specification). (a) Recommending effective interaction techniques for given analysis tasks; (b) Inferring possible tasks and techniques for a given demonstration (§ 3.3.2).

3.2.3 Evaluation, Potential Risks and Alternative Approaches

Thorough evaluation of the interaction recommender will occur through the interfaces and techniques we develop in phase three. Nevertheless, following Draco's original approach, we will conduct **benchmark studies** to understand the implications of recommending interaction techniques and states as compared to simply visual encodings. The primary potential risk in this phase is that a system of logical facts and constraints will prove to be too fine-grained to express effectiveness criteria for interaction design. We mitigate this risk by also considering machine learning approaches validated in prior work, and by drawing on the expertise of Draco's lead author, Dominik Moritz (letter of collaboration attached). In the event that alternate approaches are necessary, we will consider simpler generate-and-test methods [73] used by classic automated visualization systems including APT [58] and SAGE [74].

3.3 Phase Three: Operationalizing Effectiveness Criteria

In the final phase of this project, we will explore novel visualization techniques and interfaces that are enabled by the interaction recommender system developed in phase two.

3.3.1 From a Minimap of Interactive States to Perceived Affordances

Consider Figure 5: if you saw this visualization embedded on a web page, would you expect it to be static or interactive? If the latter, what interactions would you imagine were possible? In user interface design, the set of interface actions a user understands to be readily possible (which may be a subset of all possible actions) are called *perceived affordances* [65,66]. And, though perceived affordances have been well-studied in the human-computer interaction (HCI) literature, they have been relatively neglected in data visualization. To our knowledge, only a single recent data visualization paper has studied this issue: coining the term *suggested interactivity*, Boy et al. use icons and animation to indicate a visualization can be interacted with, but human-subject evaluations yield largely inconclusive results about whether participants do indeed interact unprompted [18]. By applying our theoretical lens from phase one (and particularly study C), we hypothesize that animation transforms interactivity from an active experience into a passive one without relating to a user's higher-level goals, and thus fails to bridge the gulf of formation.

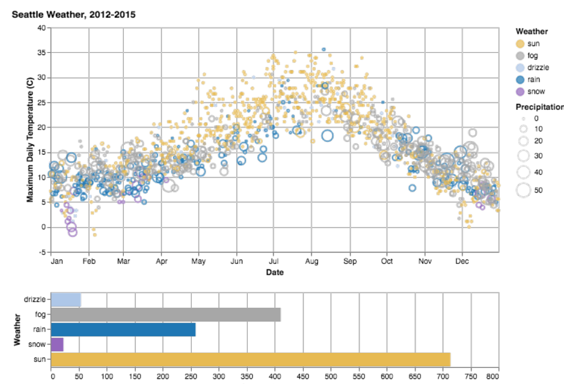


Figure 5: Is this an example of a static or interactive visualization? Without any perceived affordances [65, 66], users have no way of knowing that they can click on bars to filter points, or brush in the scatterplot to reaggregate the histogram.

Our interaction recommender system allows us to **test this hypothesis by exploring the design space of perceived affordances for data visualization.** We will start by exploring the more straightforward design a recommender system enables: displaying a gallery of suggestions. We dub this approach a "minimap" of interactive states as it resembles similar views found in video games and text editors. For a given interactive visualization, the recommender would suggest a high-level task or goal our studies found the technique to be effective at answering. This task, along with the number of interactive states left to explore, would label a button which reveals the minimap on click. Thumbnails represent clusters of states: hovering over one sets the the visualization's state, and clicking drills down into the state space suggestions.

Minimaps are just one point in a larger design space of using interaction recommendation to drive perceived affordances for data visualization. As suggestions will update in near real-time, based on how a user interacts with the visualization, we will explore adapting **information scent techniques** [68, 92] to depict a distribution of suggested states directly on the visualization itself — an approach more in-keeping with the minimalist spirit of perceived affordances. For example, how can brush extents be overlaid to suggest dragging or resizing it to explore particular portions of the data? Similarly, how can axes be augmented to encourage a user to pan or zoom to particular locations? Can this information be displayed "just-in-time"?

Evaluation, Potential Risks and Alternative Approaches. We will conduct comparative human-subjects evaluations of design variants we explore, measuring whether they increase a user's propensity to interact with a visualization, and how frequently users explore the specific states suggested. Moreover, the Boy et al. work, and our phase one experiments, provide useful baselines for additional comparative studies. For instance, how do interaction traces differ with and without perceived affordances? How does higher-level cognition change (e.g., do participants pose more questions of the data)? Like the Boy et al. work, there is a risk that our perceived affordance designs will be similarly inconclusive. However, we mitigate this risk by grounding our designs in the results of our empirical studies via the interaction recommender system. Moreover, our minimap design mimics displays of recommended visualizations in exploratory visual analysis systems that have been found to yield superior analysis outcomes [94, 95]. In the unlikely event that none of our designs yield meaningfully more or different participant engagement with interactive visualizations, we believe our work would still provide a useful negative result to motivate further research.

3.3.2 Interaction Design by Demonstration

There has been a recent resurgence in research on graphical interfaces for visualization design. Systems like Data Illustrator [55] and Charticulator [71] explore how the threshold for authoring visualizations can be lowered by employing direct manipulation interaction including drawing shapes to represent marks, and using drag-and-drop to bind data fields to visual properties (e.g., position, color, size, etc.). Though these systems have a large expressive gamut, they focus exclusively on creating *static* visualizations.

We will explore how graphical and direct manipulation interfaces can be extended to support the design of interactive behaviors. In particular, we will work with the PI's Lyra system [79] which provides a graphical frontend to the Vega and Vega-Lite visualization grammars. In Lyra, direct manipulation interactions (e.g., dragging a data field to drop zones that overlay the visualization canvas) generate statements in Vega-Lite. Lyra compiles these statements, and merges them into the full Vega specification that is its backing data model. Visual property inspectors (e.g., HTML widgets, color pickers, etc.) provide fine-grained control over the latter. As a result, Lyra unifies these two levels of abstraction into a single cohesive environment: via direct manipulation, users can rapidly create recognizable output; and with the visual inspectors, they can manually tweak low-level details. Users never need to explicitly select a level of abstraction to work with, but rather work across them seamlessly without experiencing a sharp *complexity cliff* [16].

In this context, a natural analogy to draw would be for interactive behaviors to be designed via demonstration. Heuristics, however, will not be sufficiently expressive to capture the myriad possible interaction techniques a single demonstration may map to. Thus, we will instead **use the interaction recommender to infer the most likely behavior the user intended** for the given dataset and visual encodings. For instance, the recommender may suggest "brush-to-zoom" when the user drags over a cluster of points in a scatterplot, but "pan" when dragging over a geographic map. However, instantiating the recommender in the context of an authoring environment raises several new research questions, including:

1. **How are recommendations exposed to the user?** While we anticipate being able to reuse aspects of the minimap design (§ 3.3.1), it would be insufficiently expressive to merely recommend *complete* interaction techniques in an authoring context. In particular, a user must be able to move across levels

of abstraction for interaction design as seamlessly as they do for visual encoding. Thus, an immediate challenge will be the design of **new interface elements to visually represent the constituent components of an interaction technique.** Critically, these interface elements should promote a design process with *low viscosity* [16] — through additional demonstrations or interface manipulations, users should be able to fluidly compose and recompose whole interaction techniques and their components.

2. **Why are these recommendations being made?** Demonstration interfaces can present a wide gulf of evaluation as the system may disambiguate a user's input in unexpected ways. Visually representing the components of a recommended technique is a first step to bridging this gulf, and we will explore two further strategies. First, while prior systems [76] only surface recommendations after a demonstration is complete, we will investigate **how to generate and update recommendations during a demonstration**. Second, using information scent techniques [68, 92], the demonstration interface will communicate to the user how they need to vary their actions to generate alternate recommendations. Together, these approaches should narrow the gulf of evaluation by giving users a *continuous representation* into the system state, with *rapid, incremental, and reversible operations* [84].

After constructing an interaction technique via demonstration, users will want to test and verify that it behaves as they want it to. Thus, a final research question we will study here is **the design of visual debugging techniques for interactive behaviors.** We have begun to investigate this question in preliminary work [38,39] that has developed a timeline view for recording, replaying, and inspecting interactive state as well as layered annotations for visualizing interactions on the chart. However, these methods operate over Vega primitives [81] within a text editor, and exploring them within the context of Lyra opens new research directions. For instance, how do we narrow the gap between debugging and design? If a user notices an error through the debugging interface, can they identify it, demonstrate the desired behavior, and have the interaction recommender synthesize the necessary bug fix?

Evaluation, Potential Risks and Alternative Approaches. We will conduct human-subjects studies to evaluate the usability of interaction design by demonstration. And, through within-subjects studies that compare demonstration with editing Vega and Vega-Lite specifications, we will study its impact on the barrier to entry for authoring interactivity. The primary risk here is that our demonstration interface will exhibit gulfs of execution and evaluation too wide for the user to cross. Recent results, however, suggest there is a role for demonstration in interactive data analysis [75], and this project would study its impact on data visualization design. Moreover, we look to mitigate this risk by coupling demonstration with more traditional user interface elements to depict the construction of interaction techniques.

3.3.3 Automated Interaction Design in Data Science Notebooks

Interaction design by demonstration requires significant user intervention: users must first have the express intent to create an interaction technique, and then work to disambiguate system suggestions. For data scientists, however, visualizations are often an intermediary artifacts: they want to be able to rapidly create them, interact with them to unearth insights, and then proceed with their subsequent analysis. Thus, it is infeasible to devote this level of attention to crafting interactive visualizations in the midst of an analysis process. While the PI's prior work with Vega-Lite formulated concise, high-level abstractions for interactive visualization, and has seen broad adoption in the Jupyter and Observable data science communities, **informal, preliminary interviews** with representative users suggests that manually specifying interaction mechanics introduces sufficient friction into the analysis process (without a guarantee of insight commensurate with the effort) that many users eschew it in favor of simply static visualization. Thus, we will explore **the minimal amount of user intervention necessary to synthesize an effective interactive visualization.**

We will work in Jupyter and Observable notebooks as the literate programming that occurs within them — mixing code, comments, and other artifacts — offers a rich source of metadata to mine for analytic intent. We will begin by looking narrowly at code cells, and using static analysis techniques to identify expressions of low-level tasks from Amar et al. [4] or high-level tasks from Yi et al. [97]. For instance, `data.loc[data['year'] === 2000]` *filters* for records from the year 2000 while `sales.join(stores, ...)` *connects* two datasets together. When a task match is found, the interaction recommender is invoked and its top suggestion is displayed in situ, and the dataflow is rewritten such that output is received from the interactive visualization rather than through code execution[3]. With our previous examples, we might imagine the recommender suggests a histogram where the user can click bars to interactively filter data in subsequent cells, and the `join` produces a matrix view where users can not only rapidly visualize the result of the operation but interactively reorder the rows and columns to identify patterns in the data.

Evaluation, Potential Risks and Alternate Approaches. We will run within-subjects user studies to assess how an analyst's process is affected by automated interactive visualization. For instance, do analysts more broadly explore their datasets, do they pose and answer more hypotheses, or are they more confident in their findings? To promote ecological validity, we will recruit real-world data scientists. A risk posed here is analysis code is often a form of *exploratory programming* [46], and thus may prove too hetereogeneous to mine reliably. Recent work by Kery et al. [47], however, suggests that analysis code does indeed provide sufficient structure for static analysis. Moreover, newer analysis languages (e.g., Tea [43] or Touchstone [26, 56]) will be simpler to mine as they provide more declarative and domain-specific abstractions.

4 Education Plan

The research goals of this project align closely with the PI's pedagogical goals: to engage students in research early in their academic careers, and to diversify the computer science study body using data visualization, human-computer interaction (HCI), and design thinking. In this section, we detail the PI's existing work along these two axes and describe how this project would further these goals.

Rating Scale: 1=Strongly Disagree, 4=Neutral, 7=Strongly Agree, N/A=Not Applicable (7 is best)

Quality of Teaching	***AVG***	***1 2 3 4 5 6 7***	***MEDIAN***	***STDEV***
Stimulated interest	**6.9**		7.0	0.35
Displayed thorough knowledge of subject material	**7.0**		7.0	0.0
Helped me learn	**6.8**		7.0	0.61
Overall rating	**6.9**		7.0	0.35

Comments on teaching (strengths, areas for improvement)

Student 7533 - Arvind displays a lot of knowledge on the subject matter of the class through his lectures and answering questions. It's clear from his teaching and response to students (both in person and on piazza) that he cares about the course, the subject matter, and the students.

Student 39994 - Very refreshing lecture style. The lecture slides are worth their weight in gold (very useful and I will refer to them later on). I found the in class tutorials for Tableau and other software very useful. As were the way the smaller projects leading up to Final Project was organized. Hope this is offered in future semesters.

Student 40609 - Arvind is really passionate about the material in this class and teaching it to his students. He makes lecture interesting and interactive, so it's really easy to stay engaged. He is really supportive of student projects, and it's obvious he wants us all to be successful.

Student 48999 - Definitely one of the best lecturers I've had at MIT over the course of 5 years. His lectures are engaging, informative, and fly by.

Figure 6: An excerpt of end-of-semester evaluations for the PI's new class 6.894: Interactive Data Visualization.

4.1 Curriculum Development

In Spring 2019, the PI introduced a new course at MIT, *6.894: Interactive Data Visualization*. Despite being a brand new, graduate-level computer science class, **over 50 students** enrolled including a sizable number of **undergraduates and students from other departments** including architecture, design, urban studies & planning, and the Sloan business school. In end-of-semester evaluations, students rated the class **6.7 out of 7** with a **6.9 out of 7** for the PI's instruction — both are **among the department's top scores**.

As Fig. 6 shows, students found the class particularly engaging and interactive as the PI, drawing on his participation in a CRA/NSF-funded teaching workshop for new computer science faculty [69], mixed traditional lectures with **studio and active learning techniques** that have been empirically-validated to improve student performance [29]. For example, every class session included several **think-pair-share** activities to

[3] In Jupyter, this step involves coordinating the JavaScript frontend and Python kernel — an engineering task we will work on with our Jupyter collaborators (see letter). Observable, as a reactive environment, natively supports this coordination.

encourage students to actively and critically engage in new material [85]. Throughout the semester, students were exposed to example visualizations, and conducted verbal **design critiques** to understand the strengths and weaknesses of these examples [21]. To ensure students were not mechanically applying design principles, we employed **faded worked examples across multiple modalities** [7] including annotating suboptimal visualization designs and sketching a redesign on paper, and exploring effective and ineffective visualization designs via Vega-Lite in Jupyter notebooks. And, students conducted several **peer evaluations** for each other's assignments and final projects — in their final reports, many students noted how this process helped improve their work, not only due to the feedback they received, but also as they came to more critically assess their own work. The vast majority of these final projects focused on **civic and social issues** (e.g., gerrymandering, workforce diversity, income inequality, social mobility, etc.) as the the PI had carefully curated the final project suggestions through outreach to researchers actively working on these issues — where possible, teams were partnered with these researchers to engage in a user-centric design process.

In addition to *6.894*, the PI co-instructs the undergraduate course *6.170: Software Studio on Web Development*, which also incorporates many of the same techniques including think-pair-share, faded worked examples, and peer evaluation. **If this project is successful, the research outcomes will positively impact both classes** as interaction design is a major component of both classes, but our ability to teach it effectively is hampered by a lack of theory and corresponding design principles. At best, we are only able to help students build an intuition for good interaction design by exposing them to myriad positive and negative examples — an unstructured and ad hoc process that is particularly problematic when a student's intuition does not align with that of grading staff. Moreover, though research has successfully been raising the level of abstraction for authoring interactive behaviors, the primary mechanism to do so remains textual specifications — a representation that is often difficult for non-computer science students to work with.

To better integrate this project's research and education goals, the PI has begun initial discussions with MIT's **Teaching & Learning Lab** (T&LL, letter of collaboration attached). As phase one of this project unfolds, we plan to integrate our findings through new faded worked examples of *interactive* visualizations — for instance annotating screenshots of interactive states with design principles, and then sketching (or implementing) redesigns. Phase two will further facilitate this activity: the interaction recommender can be used to first enumerate interactive states for annotation, and the alternate techniques it suggests can be used by students to double check their sketches. And, we will use the interaction design by demonstration system (built in phase three) to engage students in a rapid interaction prototyping exercise. These exercises will be analogous to ones we currently run for visual encodings, shown in Figure 7. Throughout the duration of this project, T&LL will conduct **classroom observations** to evaluate and improve the PI's instruction as well as administer **pre- and post-assessments to measure student learning in design thinking and data literacy**.

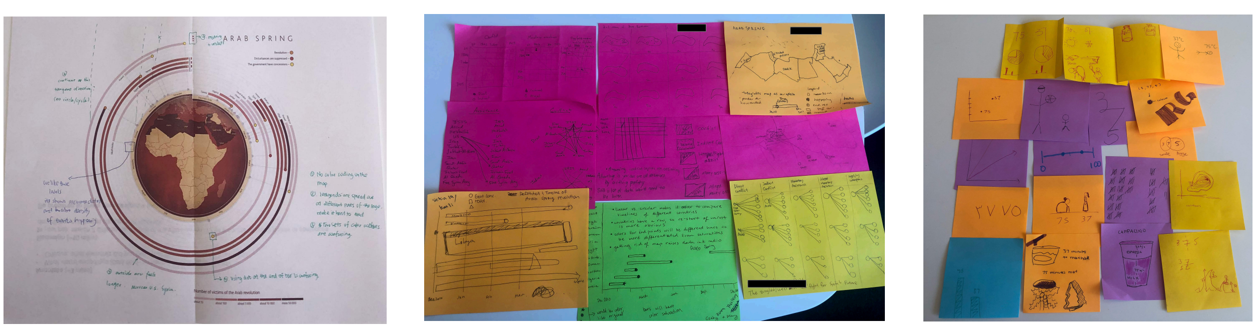


Figure 7: Photos from in-class activities in 6.894. (left) A faded worked example, where students annotated strengths and weaknesses of the design using theory on visual encoding effectiveness. (center) Sketches from a re-design exercise. (right) Sketches from a rapid prototyping exercise. In years 2 – 4, we will build similar exercises for interactivity.

4.2 Diverse Participation in Computer Science Research

The PI is committed to, and has a track record of, working to increase diverse participation in computer science research **across career levels, demographics, and disciplinary boundaries**. As a PhD student, he mentored 8 undergraduate students (4 women). At MIT, the PI advises a female postdoc, 4 PhD students (2 women including one of color), 3 Master's students (including 2 women of color), and 5 undergraduates (including 2 women, one of color). 3 of these undergraduates, including both women, began working with the PI after taking *6.894*, and have submitted their final projects as posters or workshop papers to IEEE VIS and ACM KDD. Several of these students, across all levels, come from **outside of computer science** including science & technology studies, architecture, urban studies & planning, and integrated design & management. Moreover, the PI has worked to further this goal through external service including on conference diversity committees (e.g., Information+ and IEEE VIS 2018) and as the diversity co-chair for IEEE VIS 2019, where he is responsible for administering the diversity scholarship.

This project budgets support for one PhD student, but we expect that approximately 1 – 2 undergraduate students will participate every year (with funding support provided by the department's UROP [4] and SuperUROP[5] programs as well REU supplements that the PI will apply for).

5 Broader Impacts

This project will make the following broader impacts:

(1) Open source software contributions. The PI has a track record of not only releasing his research artifacts as open source software, but supporting and maintaining them on a continuous basis. All software developed as part of this project (including the interaction recommender, and the demonstration and automated design systems) will be released as open source, or merged back into existing projects. We will also collaborate with members of the Python/Jupyter data science community and Observable to ensure smooth integration with these platforms. **2 months have been allocated in each years 3 – 5 towards this goal.**

(2) Broad dissemination of results. Going hand-in-hand with the prior goal, the PI has a similar history of facilitating broad adoption of his research. Research results from this project will not only be published through papers at premier academic venues, but will be accompanied by an article that describes the work in an accessible style published on *Multiple Views*[6], a high-profile Medium blog about explaining data visualization research to lay audiences. We will also partner with the MIT CSAIL press team (letter attached) to promote research results in popular science & technology media.

(3) Advance discovery and understanding while promoting teaching, training, and learning. As described in § 4.1, theoretical results will be incorporated into new curriculum material. Every software release will include thorough documentation and tutorials. And, as we have done in the past, we will host training workshops at practitioner-oriented venues (e.g., OpenVis Conf and ODSC) to facilitate adoption by real-world users, and we have reserved specific travel funds for this purpose during every year of this project. All educational and training material will be released under broad, permissive licenses.

(4) Broaden participation of under-represented groups. The PI has a track record of including diverse student populations in the research process (see § 4.2) and this goal will be a priority for this project.

[4] http://uaap.mit.edu/research-exploration/urop

[5] https://superurop.mit.edu

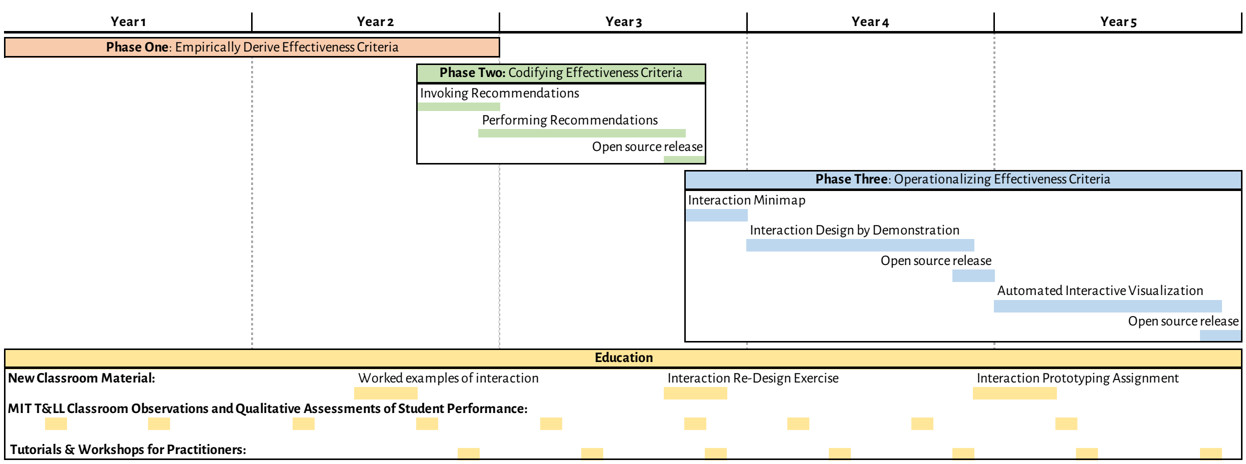
[6] https://medium.com/multiple-views-visualization-research-explained

6 Advisory Committee

The PI has assembled an advisory board of recognized experts to support the research and education work described in this proposal. **Dr. Brian Granger** is the co-founder and co-director of Project Jupyter, and will advise on the use of Jupyter's telemetry system for the phase one studies, as well as provide feedback on the automated interaction design tools (§ 3.3.3). **Mike Bostock** is the CEO of Observable, the creator of D3.js, and formerly a visual data journalist at the New York Times. His advice will help us craft ecologically valid experimental conditions in phase one, and he will also help us conduct experiments within Observable notebooks. **Dr. Michel Beaudouin-Lafon** and **Dr. Wendy Mackay** are pioneers of human-computer interaction, and have conducted decades of research into interaction models (§ 2.3) and quantitative and qualitative studies of interaction techniques. We will draw on their expertise to inform the design of studies in phase one, particularly if developing new qualitative methodologies becomes necessary. **Dominik Moritz** will be an Assistant Professor at Carnie Mellon University in Fall 2020, and as the lead author of Draco, this project will benefit from his feedback in phase two. **Dr. Loudes Alemán** is the Associate Director of MIT's Teaching & Learning Laboratory, and will work with us to develop new educational materials, conduct classroom observations, and help administer pre- and post-completion surveys to assess student learning.

7 Project Timeline

The following visualization depicts how the research and education components of this project will unfold:



8 Results From Prior NSF Support

The PI has no previous NSF support.

References Cited

[1] Christopher Ahlberg, Christopher Williamson, and Ben Shneiderman. Dynamic queries for information exploration: An implementation and evaluation. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 619–626. ACM, 1992.

[2] Danielle Albers, Michael Correll, and Michael Gleicher. Task-driven evaluation of aggregation in time series visualization. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 551–560. ACM, 2014.

[3] Pär-Anders Albinsson and Shumin Zhai. High precision touch screen interaction. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 105–112. ACM, 2003.

[4] Robert Amar, James Eagan, and John Stasko. Low-level components of analytic activity in information visualization. In *IEEE Symposium on Information Visualization, 2005. INFOVIS 2005.*, pages 111–117. IEEE, 2005.

[5] Caroline Appert, Olivier Chapuis, and Michel Beaudouin-Lafon. Evaluation of pointing performance on screen edges. In *Proceedings of the working conference on Advanced visual interfaces*, pages 119–126. ACM, 2008.

[6] Daniel Archambault, Helen Purchase, and Bruno Pinaud. Animation, small multiples, and the effect of mental map preservation in dynamic graphs. *IEEE Transactions on Visualization and Computer Graphics*, 17(4):539–552, 2010.

[7] Robert K Atkinson, Sharon J Derry, Alexander Renkl, and Donald Wortham. Learning from examples: Instructional principles from the worked examples research. *Review of educational research*, 70(2):181–214, 2000.

[8] Ravin Balakrishnan. beating fitts law: virtual enhancements for pointing facilitation. *International Journal of Human-Computer Studies*, 61(6):857–874, 2004.

[9] Scott Bateman, Regan L Mandryk, Carl Gutwin, Aaron Genest, David McDine, and Christopher Brooks. Useful junk?: the effects of visual embellishment on comprehension and memorability of charts. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2573–2582. ACM, 2010.

[10] Leilani Battle, Remco Chang, and Michael Stonebraker. Dynamic prefetching of data tiles for interactive visualization. In *Proceedings of the 2016 International Conference on Management of Data*, pages 1363–1375. ACM, 2016.

[11] Michel Beaudouin-Lafon. Instrumental interaction: an interaction model for designing post-wimp user interfaces. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, pages 446–453. ACM, 2000.

[12] Michel Beaudouin-Lafon. Designing interaction, not interfaces. In *Proceedings of the working conference on Advanced visual interfaces*, pages 15–22. ACM, 2004.

[13] Richard A Becker, William S Cleveland, and Ming-Jen Shyu. The visual design and control of trellis display. *Journal of computational and Graphical Statistics*, 5(2):123–155, 1996.

[14] Hrvoje Benko, Andrew D Wilson, and Patrick Baudisch. Precise selection techniques for multi-touch screens. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, pages 1263–1272. ACM, 2006.

[15] Jacques Bertin, William J Berg, and Howard Wainer. *Semiology of graphics: diagrams, networks, maps*, volume 1. University of Wisconsin press Madison, 1983.

[16] A F Blackwell, C Britton, A Cox, T R G Green, C Gurr, G Kadoda, M S Kutar, M Loomes, C L Nehaniv, M Petre, C Roast, C Roe, A Wong, and Young R M. Cognitive dimensions of notations: Design tools for cognitive technology. In *Cognitive technology: instruments of mind*, pages 325–341. Springer, 2001.

[17] Michelle A Borkin, Azalea A Vo, Zoya Bylinskii, Phillip Isola, Shashank Sunkavalli, Aude Oliva, and Hanspeter Pfister. What makes a visualization memorable? *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2306–2315, 2013.

[18] Jeremy Boy, Louis Eveillard, Françoise Detienne, and Jean-Daniel Fekete. Suggested interactivity: Seeking perceived affordances for information visualization. *IEEE transactions on visualization and computer graphics*, 22(1):639–648, 2015.

[19] Matthew Brehmer and Tamara Munzner. A multi-level typology of abstract visualization tasks. *IEEE transactions on visualization and computer graphics*, 19(12):2376–2385, 2013.

[20] Eli T Brown, Alvitta Ottley, Helen Zhao, Quan Lin, Richard Souvenir, Alex Endert, and Remco Chang. Finding waldo: Learning about users from their interactions. *IEEE Transactions on visualization and computer graphics*, 20(12):1663–1672, 2014.

[21] Michelene TH Chi, Miriam Bassok, Matthew W Lewis, Peter Reimann, and Robert Glaser. Self-explanations: How students study and use examples in learning to solve problems. *Cognitive science*, 13(2):145–182, 1989.

[22] William S Cleveland and Robert McGill. Graphical perception: Theory, experimentation, and application to the development of graphical methods. *Journal of the American statistical association*, 79(387):531–554, 1984.

[23] Andy Cockburn, David Ahlström, and Carl Gutwin. Understanding performance in touch selections: Tap, drag and radial pointing drag with finger, stylus and mouse. *International Journal of Human-Computer Studies*, 70(3):218–233, 2012.

[24] Aurélie Cohé, Bastien Liutkus, Gilles Bailly, James Eagan, and Eric Lecolinet. Schemelens: A content-aware vector-based fisheye technique for navigating large systems diagrams. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):330–338, 2015.

[25] Matt Conlen, Alex Kale, and Jeffrey Heer. Capture & analysis of active reading behaviors for interactive articles on the web. *Computer Graphics Forum (Proc. EuroVis)*, 2019.

[26] Alexander Eiselmayer, Chat Wacharamanotham, Michel Beaudouin-Lafon, and Wendy E Mackay. Touchstone2: An interactive environment for exploring trade-offs in hci experiment design. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, page 217. ACM, 2019.

[27] Niklas Elmqvist, Andrew Vande Moere, Hans-Christian Jetter, Daniel Cernea, Harald Reiterer, and TJ Jankun-Kelly. Fluid interaction for information visualization. *Information Visualization*, 10(4):327–340, 2011.

[28] Jean-Daniel Fekete. Advanced interaction for information visualization. In *PacificVis*, 2010.

[29] Scott Freeman, Sarah L Eddy, Miles McDonough, Michelle K Smith, Nnadozie Okoroafor, Hannah Jordt, and Mary Pat Wenderoth. Active learning increases student performance in science, engineering, and mathematics. *Proceedings of the National Academy of Sciences*, 111(23):8410–8415, 2014.

[30] Michael Gleicher, Michael Correll, Christine Nothelfer, and Steven Franconeri. Perception of average value in multiclass scatterplots. *IEEE transactions on visualization and computer graphics*, 19(12):2316–2325, 2013.

[31] Yves Guiard, Michel Beaudouin-Lafon, and Denis Mottet. Navigation as multiscale pointing: extending fitts' model to very high precision tasks. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, pages 450–457. ACM, 1999.

[32] Lane Harrison, Fumeng Yang, Steven Franconeri, and Remco Chang. Ranking visualizations of correlation using weber's law. *IEEE transactions on visualization and computer graphics*, 20(12):1943–1952, 2014.

[33] Jeffrey Heer and Michael Bostock. Crowdsourcing graphical perception: using mechanical turk to assess visualization design. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 203–212. ACM, 2010.

[34] Jeffrey Heer and Ben Shneiderman. Interactive dynamics for visual analysis. *Queue*, 10(2):30, 2012.

[35] William C Hill, James D Hollan, Dave Wroblewski, and Tim McCandless. Edit wear and read wear. In *Chi*, volume 92, pages 3–7, 1992.

[36] Ken Hinckley, Edward Cutrell, Steve Bathiche, and Tim Muss. Quantitative analysis of scrolling techniques. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 65–72. ACM, 2002.

[37] Jane Hoffswell, Arvind Satyanarayan, and Jeffrey Heer. Visual debugging techniques for reactive data visualization. *Computer Graphics Forum (Proc. EuroVis)*, 2016.

[38] Jane Hoffswell, Arvind Satyanarayan, and Jeffrey Heer. Visual debugging techniques for reactive data visualization. *Computer Graphics Forum (Proc. EuroVis)*, 2016.

[39] Jane Hoffswell, Arvind Satyanarayan, and Jeffrey Heer. Augmenting code with in situ visualizations to aid program understanding. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, page 532. ACM, 2018.

[40] Kevin Hu, Snehalkumar'Neil'S Gaikwad, Madelon Hulsebos, Michiel A Bakker, Emanuel Zgraggen, César Hidalgo, Tim Kraska, Guoliang Li, Arvind Satyanarayan, and Çağatay Demiralp. Viznet: Towards a large-scale visualization learning and benchmarking repository. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, page 662. ACM, 2019.

[41] Edwin L Hutchins, James D Hollan, and Donald A Norman. Direct manipulation interfaces. *Human-computer interaction*, 1(4):311–338, 1985.

[42] TJ Jankun-Kelly, Kwan-Liu Ma, and Michael Gertz. A model and framework for visualization exploration. *IEEE Transactions on Visualization and Computer Graphics*, 13(2):357–369, 2007.

[43] Eunice Jun, Maureen Daum, Jared Roesch, Sarah E Chasins, Emery D Berger, Rene Just, and Katharina Reinecke. Tea: A high-level language and runtime system for automating statistical analysis. *arXiv preprint arXiv:1904.05387*, 2019.

[44] Matthew Kay and Jeffrey Heer. Beyond weber's law: A second look at ranking visualizations of correlation. *IEEE transactions on visualization and computer graphics*, 22(1):469–478, 2015.

[45] Daniel F Keefe. Integrating visualization and interaction research to improve scientific workflows. *IEEE Computer Graphics and Applications*, 30(2):8–13, 2010.

[46] M Beth Kery and B A Myers. Exploring exploratory programming. In *2017 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*, pages 25–29, October 2017.

[47] Mary Beth Kery, Bonnie E John, Patrick O'Flaherty, Amber Horvath, and Brad A Myers. Towards effective foraging by data scientists to find past analysis choices. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, page 92. ACM, 2019.

[48] Yea-Seul Kim, Katharina Reinecke, and Jessica Hullman. Explaining the gap: Visualizing one's predictions improves recall and comprehension of data. In *ACM Human Factors in Computing Systems (CHI)*, 2017.

[49] Younghoon Kim and Jeffrey Heer. Assessing effects of task and data distribution on the effectiveness of visual encodings. *Computer Graphics Forum (Proc. EuroVis)*, 2018.

[50] Younghoon Kim and Jeffrey Heer. Assessing effects of task and data distribution on the effectiveness of visual encodings. *Computer Graphics Forum*, 37(3):157–167, 2018.

[51] Brittany Kondo and Christopher Collins. Dimpvis: Exploring time-varying information visualizations by direct manipulation. *IEEE transactions on visualization and computer graphics*, 20(12):2003–2012, 2014.

[52] Robert Kosara. An empire built on sand: Reexamining what we think we know about visualization. In *Proceedings of the Sixth Workshop on Beyond Time and Errors on Novel Evaluation Methods for Visualization*, pages 162–168. ACM, 2016.

[53] William Kruskal. Visions of maps and graphs. In *Proceedings of the International Symposium on Computer-Assisted Cartography, Auto-Carto II*, pages 27–36. Citeseer, 1975.

[54] Heidi Lam. A framework of interaction costs in information visualization. *IEEE transactions on visualization and computer graphics*, 14(6):1149–1156, 2008.

[55] Zhicheng Liu, John Thompson, Alan Wilson, Mira Dontcheva, James Delorey, Sam Grigg, Bernard Kerr, and John Stasko. Data illustrator: Augmenting vector design tools with lazy data binding for expressive visualization authoring. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, page 123. ACM, 2018.

[56] Wendy E Mackay, Caroline Appert, Michel Beaudouin-Lafon, Olivier Chapuis, Yangzhou Du, Jean-Daniel Fekete, and Yves Guiard. Touchstone: exploratory design of experiments. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 1425–1434. ACM, 2007.

[57] I Scott MacKenzie. Fitts' law as a research and design tool in human-computer interaction. *Human-computer interaction*, 7(1):91–139, 1992.

[58] Jock Mackinlay. Automating the design of graphical presentations of relational information. *Acm Transactions On Graphics (Tog)*, 5(2):110–141, 1986.

[59] Jock Mackinlay, Pat Hanrahan, and Chris Stolte. Show me: Automatic presentation for visual analysis. *IEEE transactions on visualization and computer graphics*, 13(6):1137–1144, 2007.

[60] Miro Mannino and Azza Abouzied. Expressive time series querying with hand-drawn scale-free sketches. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, page 388. ACM, 2018.

[61] Dominik Moritz, Chenglong Wang, Gregory Nelson, Halden Lin, Adam M. Smith, Bill Howe, and Jeffrey Heer. Formalizing visualization design knowledge as constraints: Actionable and extensible models in draco. *IEEE Trans. Visualization & Comp. Graphics (Proc. InfoVis)*, 2019.

[62] Tamara Munzner. *Visualization Analysis and Design*. AK Peters Visualization Series. CRC Press, 2015.

[63] Brad A Myers. Separating application code from toolkits: eliminating the spaghetti of call-backs. In *UIST*, volume 91, pages 211–220. Citeseer, 1991.

[64] Mathieu Nancel, Emmanuel Pietriga, Olivier Chapuis, and Michel Beaudouin-Lafon. Mid-air pointing on ultra-walls. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 22(5):21, 2015.

[65] Don Norman. *The design of everyday things: Revised and expanded edition*. Basic books, 2013.

[66] Donald A Norman. Affordance, conventions, and design. *interactions*, 6(3):38–43, 1999.

[67] Chris North. Toward measuring visualization insight. *IEEE computer graphics and applications*, 26(3):6–9, 2006.

[68] Peter Pirolli and Stuart Card. Information foraging. *Psychological review*, 106(4):643, 1999.

[69] Leo Porter, Cynthia Lee, Beth Simon, and Mark Guzdial. Preparing tomorrow's faculty to address challenges in teaching computer science, May 2017.

[70] Philip Quinn, Sylvain Malacria, and Andy Cockburn. Touch scrolling transfer functions. In *Proceedings of the 26th annual ACM symposium on User interface software and technology*, pages 61–70. ACM, 2013.

[71] Donghao Ren, Bongshin Lee, and Matthew Brehmer. Charticulator: Interactive construction of bespoke chart layouts. *IEEE transactions on visualization and computer graphics*, 25(1):789–799, 2018.

[72] George Robertson, Roland Fernandez, Danyel Fisher, Bongshin Lee, and John Stasko. Effectiveness of animation in trend visualization. *IEEE transactions on visualization and computer graphics*, 14(6):1325–1332, 2008.

[73] Steven F Roth and Joe Mattis. Data characterization for intelligent graphics presentation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 193–200. ACM, 1990.

[74] Steven F Roth, Joe Mattis, and Xavier Mesnar. Graphics and natural language as components of automatic explanation. *ACM SIGCHI Bulletin*, 20(1):76, 1988.

[75] B Saket and A Endert. Investigating the manual view specification and visualization by demonstration paradigms for visualization construction. In *Computer graphics forum*, 2019.

[76] Bahador Saket, Hannah Kim, Eli T Brown, and Alex Endert. Visualization by demonstration: An interaction paradigm for visual data exploration. *IEEE transactions on visualization and computer graphics*, 23(1):331–340, 2016.

[77] Alper Sarikaya and Michael Gleicher. Scatterplots: Tasks, data, and designs. *IEEE transactions on visualization and computer graphics*, 24(1):402–412, 2017.

[78] Arvind Satyanarayan and Jeffrey Heer. Authoring narrative visualizations with ellipsis. *Computer Graphics Forum (Proc. EuroVis)*, 2014.

[79] Arvind Satyanarayan and Jeffrey Heer. Lyra: An interactive visualization design environment. *Computer Graphics Forum (Proc. EuroVis)*, 2014.

[80] Arvind Satyanarayan, Dominik Moritz, Kanit Wongsuphasawat, and Jeffrey Heer. Vega-Lite: A grammar of interactive graphics. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):341–350, 2017.

[81] Arvind Satyanarayan, Ryan Russell, Jane Hoffswell, and Jeffrey Heer. Reactive Vega: A streaming dataflow architecture for declarative interactive visualization. *IEEE transactions on visualization and computer graphics*, 22(1):659–668, 2016.

[82] Arvind Satyanarayan, Kanit Wongsuphasawat, and Jeffrey Heer. Declarative interaction design for data visualization. In *ACM User Interface Software & Technology (UIST)*, 2014.

[83] Michail Schwab, Sicheng Hao, Olga Vitek, James Tompkin, Jeff Huang, and Michelle A Borkin. Evaluating pan and zoom timelines and sliders. 2019.

[84] Ben Shneiderman. Direct manipulation: a step beyond programming languages. *Sparks of innovation in human-computer interaction*, 17:1993, 1993.

[85] Michelle K Smith, William B Wood, Wendy K Adams, Carl Wieman, Jennifer K Knight, Nancy Guild, and Tin Tin Su. Why peer discussion improves student performance on in-class concept questions. *Science*, 323(5910):122–124, 2009.

[86] Ji Soo Yi, Rachel Melton, John Stasko, and Julie A Jacko. Dust & magnet: multivariate information visualization using a magnet metaphor. *Information visualization*, 4(4):239–256, 2005.

[87] Danielle Albers Szafir. Modeling color difference for visualization design. *IEEE transactions on visualization and computer graphics*, 24(1):392–401, 2017.

[88] Danielle Albers Szafir, Steve Haroz, Michael Gleicher, and Steven Franconeri. Four types of ensemble coding in data visualizations. *Journal of vision*, 16(5):11–11, 2016.

[89] James J Thomas. *Illuminating the path:[the research and development agenda for visual analytics]*. IEEE Computer Society, 2005.

[90] Christian Tominski. Interaction for visualization. *Synthesis Lectures on Visualization*, 3(1):1–107, 2015.

[91] Archie Tse. Why We Are Doing Fewer Interactives. `https://github.com/archietse/malofiej-2016/blob/master/tse-malofiej-2016-slides.pdf`, 2016. Malofiej Graphics.

[92] Wesley Willett, Jeffrey Heer, and Maneesh Agrawala. Scented widgets: Improving navigation cues with embedded visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1129–1136, 2007.

[93] Kanit Wongsuphasawat, Dominik Moritz, Anushka Anand, Jock Mackinlay, Bill Howe, and Jeffrey Heer. Towards a general-purpose query language for visualization recommendation. In *Proceedings of the Workshop on Human-In-the-Loop Data Analytics*, page 4. ACM, 2016.

[94] Kanit Wongsuphasawat, Dominik Moritz, Anushka Anand, Jock Mackinlay, Bill Howe, and Jeffrey Heer. Voyager: Exploratory analysis via faceted browsing of visualization recommendations. *IEEE Transactions on Visualization & Computer Graphics*, 22(1):1–1, 2016.

[95] Kanit Wongsuphasawat, Zening Qu, Dominik Moritz, Riley Chang, Felix Ouk, Anushka Anand, Jock Mackinlay, Bill Howe, and Jeffrey Heer. Voyager 2: Augmenting visual analysis with partial view specifications. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pages 2648–2659. ACM, 2017.

[96] Ji Soo Yi, Youn ah Kang, and John Stasko. Toward a deeper understanding of the role of interaction in information visualization. *IEEE transactions on visualization and computer graphics*, 13(6):1224–1231, 2007.

[97] Ji Soo Yi, Youn-ah Kang, John T Stasko, and Julie A Jacko. Understanding and characterizing insights: how do people gain insights using information visualization? In *Proceedings of the 2008 Workshop on BEyond time and errors: novel evaLuation methods for Information Visualization*, page 4. ACM, 2008.

[98] Shumin Zhai, Barton A Smith, and Ted Selker. Improving browsing performance: A study of four input devices for scrolling and pointing tasks. In *Human-Computer Interaction INTERACT97*, pages 286–293. Springer, 1997.

[99] Jian Zhao, R William Soukoreff, Xiangshi Ren, and Ravin Balakrishnan. A model of scrolling on touch-sensitive displays. *International Journal of Human-Computer Studies*, 72(12):805–821, 2014.

Data Management Plan

Software Sharing Plan

All software produced as part of this project will be released under a BSD-like open source license, that permits the use, modification, and packaging of our contributions in both commercial and non-commercial contexts. Software will be released on completion of each phase or component of this project, typically alongside a corresponding publication. All open source code will be made available on GitHub.com, a public open source code repository.

Contributions to existing projects: In addition to making our software available independently, we plan to merge pieces of our contributions back into existing open source projects (e.g., Vega-Lite, CompassQL, Draco, and Jupyter) whenever possible and as appropriate. In doing so, we hope to extend the longevity of our work, promote large-scale usage, and ensure long-term maintenance through further development. The PI has a long history of open source contributions, and has been involved in several high-impact, mainstream open source projects. The PI plans to continue this track of open source impact in this project.

Data Sharing Plan

For our phase one studies, we will use real-world datasets that are freely available in the public domain. Data from our user studies will require more careful curation, as described below.

Data from User Studies: Data will be collected with the explicit permission and consent of participants, and in strict accordance with IRB protocols. Outside of IRB-mandated contact information, no personally identifiable information will be collected. Study participants will be assigned a pseudorandom identifier for their data and a code linking them to the data will be kept on an encrypted external disk stored in a secure cabinet in the laboratory.

Policies for Archiving, Sharing and Re-distribution: All public / synthetic data, code, and publications (results, papers, etc.) will be made available to the general public by hosting them on the research group's website. Moreover, to ensure broad dissemination and access, a copy of all non-sensitive data will be hosted on open access data and research repositories (e.g., arXiv and OSF).

Compliance: No personally identifiable information will be collected. All experiments will be approved by IRB processes and conducted according to IRB guidelines.

Data, Metadata and Sharing Formats

To foster re-use, development and extension of research products, all research data releases will be made available in universally accepted formats such as CSV and JSON, with sufficient documentation.