Impact of Cognitive Biases on Progressive Visualization

Marianne Procopio, Ab Mosca, Carlos Scheidegger, Eugene Wu, Remco Chang

Abstract—Progressive visualization is fast becoming a technique in the visualization community to help users interact with large amounts of data. With progressive visualization, users can examine intermediate results of complex or long running computations, without waiting for the computation to complete. While this has shown to be beneficial to users, recent research has identified potential risks. For example, users may misjudge the uncertainty in the intermediate results and draw incorrect conclusions or see patterns that are not present in the final results. In this paper, we conduct a comprehensive set of studies to quantify the advantages and limitations of progressive visualization. Based on a recent report by Micallef et al., we examine four types of cognitive biases that can occur with progressive visualization: *uncertainty bias*, *illusion bias*, *control bias*, and *anchoring bias*. The results of the studies suggest a cautious but promising use of progressive visualization – while there can be significant savings in task completion time, accuracy can be negatively affected in certain conditions. These findings confirm earlier reports of the benefits and drawbacks of progressive visualization and that continued research into mitigating the effects of cognitive biases is necessary.

Index Terms—Progressive visualization, Cognitive bias

1 INTRODUCTION

Progressive visualization is fast becoming a technique in the visualization community to help users interact with large amounts of data. Without the use of progressive techniques, queries or analyses over a large amount of data could take seconds, minutes, or even hours to complete. The wait time caused by the computation is not just a nuisance to the user [1], but has also been found to affect the user's analysis processes [2].

Research has shown that these long wait times are often unnecessary. For example, Fisher et al. report that while using non-progressive visual querying systems, users only realize their query mistakes after waiting for minutes or hours for the completed results [3]. These wait times can be avoided in progressive visualization systems as the user has the ability to catch any mistake within seconds of executing the query and correct accordingly.

Beyond time-saving, researchers have also identified a variety of other benefits of progressive visualization. A recent comprehensive report by Angelini et al. [4] categorizes and characterizes these progressive visualization systems. The result of this report suggests that, in addition to time saving, current progressive visualizations can also be used to help users gain understanding into the computation process [5], perform real time steering of the computation [6], [7], [8], [9] or query processing [10], [11], [12], and gain trust in the analysis results [8].

It is clear that progressive visualization offers a range of potential benefits. However, recent research has noted cases in which the use of progressive visualization can also lead to confusion and errors. For example, Moritz et al. [11] found that while early termination of an ongoing progressive query can save time, it can also mislead the user into believing false patterns observed early in the analysis that would not occur had the full dataset been analyzed. Similarly, Turkay at al. [6] found that users would make decisions based on patterns seen in early progressive visuals and thereby draw false conclusions about the data.

- *Marianne Procopio, Ab Mosca and Remco Chang are with Tufts University. E-mail: procopio@cs.tufts.edu, amosca01@cs.tufts.edu, remco@cs.tufts.edu.*
- *Carlos Scheidegger is with the University of Arizona. E-mail: cscheid@email.arizona.edu.*
- *Eugene Wu is with the Columbia University. E-mail: ewu@cs.columbia.edu.*

1.1 Cognitive Biases in Progressive Visualization

In a recent paper by Micallef et al. [13] the authors summarize that the benefits of progressive visualization – that is, incremental update of the visualization based on partial information – may also be the root of its usability problems. They outline four types of pitfalls that can occur during the use of progressive visualization systems and relate them to known cognitive biases. These pitfalls range from biases that stem from the perception of uncertainty (**Uncertainty Bias**) and incomplete information (**Illusion Bias**) in the data, to biases that arise from giving the user steering control to the progression (**Control Bias**) that could lead to the user's over-reliance of their prior beliefs (**Anchoring Bias**):

Uncertainty Bias: *"Misjudging the uncertainty of intermediate results"* (associated known cognitive biases: *ambiguity bias*, and *neglect of probability bias*): As observed by Fisher et al. [10], the first challenge of using progressive visualization lies in the user's ability to interpret the dynamic uncertainty information presented in the visualization. For example, in a comparison task to determine which of two gradient bars is taller, would the participants be able to determine and reason about their decisions? When using a progressive visualization, this challenge is made even more difficult because the participants have to make these decisions with *dynamic* visualizations, which results in an extra dimension of uncertainty that the users need to reconcile. The unanswered research question around this pitfall is therefore: how much does the amount of (dynamic) fluctuations in uncertainty affect the user's ability to use a progressive visualization?

Illusion Bias: *"Read something into incomplete results that is not there"* (associated known cognitive biases: *clustering illusion bias*, *illusory correlation bias*, *illusory truth effect*, and *attention bias*): Reported by Moritz et al. [11], users of progressive visualization can sometimes fall prey to "false patterns" during the progression of the visualization. In progressive query processing, false patterns can occur due to uneven sampling, particularly when data is stored in an organized data structure. For example, B+ tree is a common structure used in databases where data instances on the same node have similar values. In cases where sampling is unevenly distributed across nodes, the intermediary computational outcomes would be skewed, resulting in false patterns that do not represent the rest of the data. As Moritz et al. observed, when false patterns appear early in the progressive visualization, users could mistakenly believe they found the true pattern in the data and terminate the progression prematurely. The unanswered research question around this pitfall is therefore: how much do these false patterns affect the use of progressive visualization, and when does the occurrence of a false patterns have the highest negative impact?

Control Bias: *"Waiting for information irrelevant for the intended goal"* (associated known cognitive biases: *information bias* and *illusion of control bias*): Related to illusion bias, a broader type of cognitive bias that affects a user in using progressive visualization stems from letting the user *control* or steer the progression. While steerable progressive visualization allows the user to focus on areas of interests and thereby speed up the progressive computation [7], such flexibility can also lead to the user making false conclusions. For example, when using a steerable progressive visualization, a user can steer the progression towards certain bars in a bar chart, resulting in oversampling of those bars (and conversely, relative undersampling of the others). The uneven sampling could result in a user seeing a biased visualization that leads to an incorrect decision. The unanswered research question around this pitfall is therefore: at what frequency and extent do users of steerable progressive visualization make incorrect conclusions due to inappropriate steering?

Anchoring Bias: *"Over reliance on some information, often to the neglect of other relevant information"* [14] (associated known cognitive biases: *confirmation bias*, *belief bias*, and *exaggerated expectation bias*): Anchoring bias is a more specific form of control bias. Instead of the user steering the progression for a range of possible reasons, in anchoring bias a user is misled by the first piece of information they see (for a more complete explanation of biases in visualizations, see the work by Dimara et al. [15]). This can lead to confirmatory hypothesis testing based on the anchoring information [16]. When using an interactive visualization for data exploration, anchoring bias can cause users to preferentially weigh some information more than others unconsciously, resulting in poor analysis outcomes [14], [17]. The unanswered research question around this pitfall is therefore: do users of progressive visualization steer towards progressions unconsciously, either to what they want to see or been influenced to see? If so, how frequently do they occur and at what cost?

In summary, it appears that there are benefits and pitfalls to using progressive visualizations. On the positive side, progressive visualizations can save users a significant amount of time while also providing transparency and understanding to the underlying computation. However, the litany of potential biases increases the likelihood that users could unknowingly perceive the wrong information and ultimately make the wrong decisions. Given the pros and cons, how can we tell if progressive visualization is useful or harmful?

In this paper, we present a series of experiments aimed at evaluating the benefits and potential harms of progressive visualizations. Based on the four biases described above, we designed five studies (one preliminary study and four main studies) to assess the impact of these biases on the use of progressive visualization.

The results of our study suggest that the four biases we studied occur when using a progressive visualization and they have measurable effects on the participants' performance. However, the types of effects and their magnitudes can differ. Surprisingly, we found participants did not often suffer under *uncertainty bias*. The amount of uncertainty in the visualization had the least amount of impact on participant performance, suggesting that they were able to develop strategies to interpret uncertain information in order to make accurate decisions. Additionally, we found *illusion bias* affected the participants the most if the false patterns were introduced around the time of decision making. When making decisions prior to completion of the progressions, steering the progression had the potential to generate biased visualizations. *Control bias* can cause the participants to incorrectly steer the progression and take longer to make a decision, but unexpectedly the steering did not affect their accuracy and instead improved their confidence in their answers. And lastly, *anchoring bias* had similar effects as those exhibiting *control bias* in that participants exhibiting this bias took longer to make a decision, but also without affecting their accuracy.

Despite these pitfalls, across all experiments, we found that participants were able to complete tasks quickly and with limited error. These findings support claims that overall, progressive visualization can be an effective method for helping users understand their data and make decisions quickly, provided appropriate measures are taken to avoid biases.

2 Related Work

2.1 Progressive Visualization

Progressive visualization is closely related to a number of research areas: approximate query processing in database research (e.g. [18], [19], [20], [21]), progressive refinement in computer graphics (e.g. [22], [23], [24]), usability concerns in HCI (e.g. [10], [25]), and real time and streaming visualizations (e.g. [26], [27]). All of this research shares the goal of providing users with immediate feedback of computation.

However, what sets progressive visualization apart is the additional goal of helping users make decisions quickly while simultaneously providing understanding of the ongoing computation process [4], [28]. For example, Fisher et al. developed sampleAction [10], a progressive visualization based on the online aggregation technique in database query processing [18]. In sampleAction, a user can terminate a computationally expensive query quickly as a way of debugging. Stolper et al. extended the concept and applied it to data analysis in a system called Progressive Insight System [7]. In addition to allowing for early termination of unwanted analysis, their system supports dynamic steering of a pattern-mining algorithm in real time. Turkay et al. [6] and Badam et al. [8] adopted similar progressive strategies for high-dimensional data and Twitter data analysis, respectively.

Evaluations of these progressive visualization systems have been positive. Users of progressive visualization systems tend to prefer the immediate feedback [8] and have reported savings in analysis time via early termination of their query or analyses [10]. Further, when latency is high, the use of progressive visualization can lead users to discover more insights about the data faster when conducting an exploratory data analysis task [29] than ~~in~~the use of a traditional visual analytics system.

However, while it is generally believed that the use of progressive visualization is beneficial, there have been recent reports that warrant caution. For example, Moritz et al. [11] observed the challenge of asking analysts to make decisions based on dynamic and incomplete information. Similarly, Badam et al. [8] found that users can give partial answers and were often not confident about their decisions when using progressive visualizations. The source of low user confidence and increased decision difficulty can be attributed to the inherent challenge of making sense of uncertainty in visualization, the data, and the computation. Below, we introduce these challenges and existing work for addressing them.

2.2 Uncertainty Visualization and Data Sampling

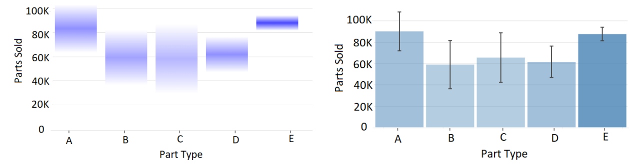
As recommended by Muhlbacher et al. [30], showing intermediate results from an ongoing computation proess is central to the design of progressive visualization. For progressive visualizations that utilize *data subsetting* [30] as the mechanism for showing intermediate results, the most common strategy is to perform data sampling and visualize the corresponding error bounds.

Existing progressive visualization systems have utilized different sampling methods. The most common approach is uniform sampling without replacement due to its simplicity in computing the error bound [10], [18]. However, uniform sampling is potentially very slow to converge, especially in cases where the sampling query is highly selective [18], [21]. As a result, other methods such as (weighted) sampling with replacement [12], [21], biased sampling [11], [31], stratified sampling [32], and sampling with ordering guarantees [33], [34] have all been adopted by progressive visualization systems. Cormode et al. provide a more complete survey of sampling techniques [35].

Given the differences in these sampling techniques, progressive visualizations will have different "behaviors" when representing error bounds, depending on the choice of the sampling technique. For example, techniques based on sampling without replacement will be guaranteed to converge because the error bound will eventually reach zero when all (pertinent) data points have been sampled. Conversely, while sampling with replacement techniques offer performance benefits [21], the error bounds will asymptotically approach but never reach zero.

In general, error bounds from sampling techniques will tend to decrease as more samples are drawn. However, in almost all cases, there is no guarantee that the decrease will be monotonic. Outliers, rare events, and "bad luck" in sampling can lead to fluctuations in error as well as in the values of the computed intermediary results. For example, error bounds based on Hoeffding's inequality are computed using the range of the randomly-sampled variable which also changes in the case of sampling an outlier [18].

Due to sampling's central role in progressive visualization, it is imperative that users are aware of the uncertainty that stems from the sampling process before they make decisions. However, uncertainty (information) visualization is still an active field of research with little consensus on the "best practices." Notable exceptions include work by Correll and Gleicher on improving the visualization of error bars using gradient plots [36], work by Ferreira et al. that proposes the use of annotations to assist users in reading visualizations with uncertainty [37], work by Fernandes et al. [38] that shows the benefits of cumulative distribution function (CDF) plots and low-density quantile dotplots helping users make real time transit decisions, and a comparative study by Gschwandtner et al. [39] on visualizations for temporal uncertainty. Additionally, Hullman summarizes the various difficulties in visualizing uncertainty despite its importance [40].



(a) (b)

Fig. 1: The gradient chart and bar chart used in our preliminary experiment. (a) The size of the bar represents the 95% confidence interval of the estimated value, shown at the midpoint of the bar. The color indicates the relative error of the estimate. The darker the blue, the more accurate the estimate is. (b) uses typical error bars to show 95% confidence interval, and the color indicates relative error of the estimate as with the gradient chart.

Differences in sampling methods, challenges in visualizing uncertainty, and the difficulties for users trying to make decisions given uncertain information can all be confounding factors for evaluating the benefits of progressive visualizations. In the section below, we describe a preliminary study for establishing the appropriate dynamic uncertainty visualization for our experiments.

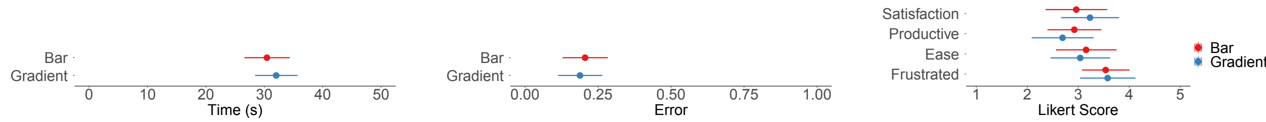
3 Preliminary Study: Dynamic Uncertainty Visualization Technique

Given the potential confounds described in the previous section, we first conduct a preliminary study to establish the visualization technique for displaying *dynamic* uncertainty information that we will use in our studies. As demonstrated by Correll and Gleicher [36], the traditional error bar for encoding uncertainty can mislead a user from interpreting statistical information correctly. A more effective alternative is the use of gradient plots (see Figure 1a). However, in the experiment by Correll and Gleicher, the comparison of the two techniques was done using static visualizations. In this preliminary study, we seek to determine if the finding holds when the visualizations are dynamic. The outcome of this preliminary study will determine the visualization technique that will be used in the remainder of the experiments[1].

Following the work by Correll and Gleicher, in this experiment we compare the use of gradient plots versus error bars for visualizing *dynamic* uncertainty visualization in a progressive manner. Using a similar experimental design described by Correll and Gleicher, we recruited 26 participants on Amazon's Mechanical Turk. Of them, 19 were between the ages of 25 and 39, 8 were female, 21 held a Bachelors or High School Diploma, and 10 rated themselves as intermediate visualization users. Participants were asked to complete 2 blocks of 3 tasks:

- Block 1:
  - T1) "How many parts were sold for Part D?"
  - T2) "How many parts were sold for Part A and B combined?"
  - T3) "Which part was sold the least?"
- Block 2:

1. The online study can be found at http://valt.cs.tufts.edu/studies/progvis/gradientBars/



(a) (b) (c)

Fig. 2: (a) Mean and 95% confidence interval of completion time, (b) error rate and (c) Likert scores for the preliminary study. We found no significant effects of interface type on time, error rate or any of the Likert questions

T4) "How many flights were flown with engine D?"
T5) "How many flights between engines A and B combined?"
T6) "Which engine had the least number of flights?"

Each block used either gradient charts (Figure 1a) or bar charts (Figure 1b). Participants were randomly assigned a presentation order and asked to rate Satisfaction, Ease of Use, Productivity and Frustration after each block on a 5 point Likert score.

We performed a within-subjects ANOVA to evaluate the error rates from using the gradient plot and the barchart with error bars. We found no significant effects of interface type on participants' error rate: $F(1,25) = 0.15, p = 0.70$ (Figure 2b).

Further, we ran a within-subjects ANOVA of completion time, and found no significant effect from the two uncertainty visualization techniques: $F(1,25) = 0.74, p = 0.39$ (Figure 2a). We also found no significant differences in Likert responses between the two using Pairwise Wilcoxon Rank Sum Tests (Figure 2c).

The results did not reveal a difference in performance using either error bars or gradient plots to visualize dynamic uncertainty. For the rest of our studies, we follow the convention established by Correll and Gleicher and use gradient plots.

4 Evaluating Bias Effects

We conduct four sets of experiments to evaluate the potential impact of the biases introduced by Micallef et al. [13] (*uncertainty bias*, *illusion bias*, *control bias* and *anchoring bias*). For our experiments, we group biases into two categories based on how the bias is evaluated: *data-oriented* and *action-oriented*.

Data-oriented experiments are experiments in which the conditions are based on altering the input data. For example, in evaluating the effects of *uncertainty bias* we modulate the amount of uncertainty in the data and measure how the different amounts of dynamic uncertainty affect the participants' decision-making when using a progressive visualization. Similarly, to evaluate the effects of *illusion bias*, we introduce "false patterns" in the data during the progression and record the participants' judgments.

In contrast, in the *action-oriented* experiments, we make use of an interactive, steerable progression visualization and infer whether the participants are under the influence of a bias by analyzing their interactions. For example, to measure *anchoring bias*, we observe and measure how much and how often a participant exhibits "anchoring" based on whether (or how often) the participant steers the progressive visualization towards a pattern in the visualization or a specific subset of the data. We use a similar method for evaluating *control bias*. We observe how often a participant deviates from the optimal configuration in a steerable progressive visualization and whether doing so results in a participant making the wrong decision.

In the following sections, we describe the experimental design for these two categories of experiments and state our hypotheses for each experiment.

4.1 Data-Oriented: Uncertainty Bias and Illusion Bias

Uncertainty Bias: Following the observation by Fisher et al. [10] (and our preliminary study) that the first challenge of using progressive visualization lies in the user's ability to interpret the dynamic uncertainty information presented in the visualization, our first experiment examines the impact of *uncertainty bias*. In this experiment, we hypothesize that increases in dynamic uncertainty when using a progressive visualization will negatively affect a participant's ability to make decisions. We categorize dynamic uncertainty in two ways: (1) variance in value (e.g. changes in the heights of the bars in a barchart), and (2) variance in error-bounds (e.g. changes in the sizes of the error bars in the barchart). Formally, we hypothesize that:

(H1) High variances in values and error-bounds in a progressive visualization will make it more difficult for a participant to read the visualization and make decisions. This will negatively affect their accuracy, speed, and confidence in their answers.

To evaluate our hypothesis, we conduct a *4 (levels of variances in values) x 4 (levels of variances in error-bounds)* factorial design study. We describe the experiment and the results in Section 6.

Illusion Bias: The amount of uncertainty is not the only factor that can affect a participant's ability to make decisions in a progressive visualization. As noted by Moritz et al. [11], during the progression if a random visual pattern happens to appear meaningful, a participant can be mislead into believing that the "false pattern" represents the real distribution in the data, thereby drawing the wrong conclusion.

In our second study, we design an experiment to examine the effects of *illusion bias* caused by these false patterns. In this experiment, a false pattern is introduced at five different predetermined times during the progressive visualization, our hypothesis, based on the finding by Moritz et al., is that:

(H2) If a false pattern appears early in the progression, a participant can be affected by *illusion bias* and will be more likely to make a decision based on this false pattern. As a result, participants will have reduced accuracy in their decisions with no effect on speed or confidence in their answers.

To detect the presence of *uncertainty bias* or *illusion bias*, we adopt the *Outcome-Oriented Operationalization* approach as proposed by Bedek et al. [41]. Using this approach, we alter the data progression and compare the decisions made by participants with the ground-truth solutions to the provided tasks. A bias is said to have occurred if there is significant statistical difference between the ground truth and the decisions by the participant.

4.2 Action-Oriented: Control Bias and Anchoring Bias

To detect whether someone is under the influence of *control bias* or *anchoring bias*, we adopt a technique proposed by Wall et al. [17] which measures the occurrence of cognitive bias as a deviation in the statistical distribution of a user's interactions with a visualization. For example, in detecting *control bias*, a participant is said to be under the influence of bias if they show an above-normal tendency towards steering the progression away from the correct or optimal sampling configuration. Similarly, in *anchoring bias*, the bias is found if a participant focuses on the steering towards a particular subset of the data (while ignoring others).

Control Bias: One danger of providing a user with control over a progressive visualization is that a user might not be aware of how the changes might affect the outcome of the sampling and the resulting visualization. When used properly, an interactive, steerable progressive visualization allows the user to focus the computation towards a specific area of interest, thereby saving time from performing unnecessary computations [7]. However, when used inappropriately, an interactive, steerable progressive visualization could result in the user obliviously under-sampling parts of the data and misjudging the final visualization.

Using the technique by Wall et al. [17], we detect whether a participant is under the influence of *control bias* using an observational experimental design. A participant is asked to use an interactive progressive visualization to complete a number of tasks. Each of these tasks has a known "ground truth" both in terms of the correct answer as well as the optimal configuration of the sampling. We observe and record participants' interactions and analyze how much they deviate from the optimal configuration and how often this results in incorrect answers. Our hypothesis is that:

(H3) When using an interactive steerable progressive visualization, some participants will succumb to *control bias* in that they cannot reason about the effect and outcome of an uneven sampling process. As a result, we will find some participants who incorrectly tune the parameters of the progression resulting in slower completion times and lower accuracy.

Anchoring Bias: *Anchoring bias* is a more specific form of *control bias*. Instead of testing whether a participant understands the control of uneven sampling, in evaluating anchoring bias we specifically test if a participant is misled by some initial piece of information that "anchors" the participant's behavior or decision-making process. This piece of information can be a false pattern in the visualization (similar to *illusion bias*), but can also be a prime with irrelevant information, visual cues, specific instructions or a user's previous experience with the data [14], [42], [43].

We follow the experimental designs of Wall et al. [14] and Wesslen et al. [42] to prime the participants with potential cues about the data. Both previous studies observe anchoring bias via interaction logs, examining if participants interacted with a subset of the data or particular views of the visualization more than others.

Additionally, the study by Wesslen et al. [44] shows that visuals shown in the training phase of an experiment can act as cognitive anchors to participants causing them to be over-reliant on information provided in training, and ultimately be over-confident, spend less time analyzing data, and make incorrect decisions. Based on the findings of Wesslen et al., we hypothesize that:

(H4) If primed prior to using a progressive visualization some participants will exhibit *anchoring bias* where they are more likely to make a decision influenced on the priming information. As a result, they will have reduced accuracy, spend less time when completing the tasks, and have an increased level of confidence in their answers.

5 Experiment Setup Overview

All experiments were conducted using Amazon Mechanical Turk. Each participant was limited to one of our experiments to prevent learning effects. During each experiment, participants were first shown a consent page, followed by explanations about how progressive visualization works including a breakdown of the different components of the visualization. They were then given a training task prior to beginning the main tasks in the study. After completion of the main tasks, participants were asked to complete a demographic survey.

As part of the instructions, participants were told to answer as accurately as possible. They were offered a base rate of $.75 for completing the tasks. Since a common concern with Mechanical Turk studies is that participants can rush through tasks with minimal effort to maximize the number of HITs they can complete [45], [46], we offered a bonus of $.25 for each correct answer. This can mitigate participants desire to complete the tasks as quickly as possible.

5.1 Visualization and Stimuli

The experiments were conducted using gradient plots, as shown in Figure 3. To carefully measure the participants' task completion time and maintain consistency of stimuli across participants, we break each of the main tasks into three separate stages. First, participants were shown the question they were tasked to answer and a blank visualization. Second, participants then clicked the "start button" when they were ready to start the task, at which point both the timer and the progression would start. The progressive visualization in the main task updated once a second and would converge to 1% error and stop updating at the 120-second mark. Participants did not know the visualization would stop updating after 120 seconds, and none of the participants in any of our studies waited the full duration before completing the task. The updated values were read from a pre-computed text file, thus ensuring all participants saw the same progressions. The progressions reflect real world behavior of sampling. The start of the progression can contain large jumps between updates, but as more "samples" are taken the variability between updates is reduced. Third, when the participants felt ready to answer, they clicked the "stop button" to stop the progression and the timer. The participants then typed their answer in a text box and selected how confident they were using a 4-point Likert-scale.

5.2 Datasets

Two datasets were manually generated for the studies. The first is a fictitious sales record of a manufacturing company of spare parts. The second dataset is loosely based on the flight dataset [47] and shows different types of airplane engines and their usage frequency.

5.3 Tasks

To evaluate how well participants can perform judgment with the "bars" in the gradient plots, we designed three types of tasks of increasing difficulty. These tasks correspond to three tasks in the analytic task taxonomy by Amar et al. [48]: *Retrieve Value* (read a value), *Derive Value* (compute the sum or differences of two values), and *Find Extremum* (compare all values to find the maximum or minimum). Together, success in these tasks reflect a participant's ability to read a progressive visualization, reason about the information in the progressive visualization, and ultimately make a decision. Specifically, questions used in our studies include:

1) *Read Value*: To complete these tasks, the participants only need to examine one bar in the gradient plot. Example questions include "how many units were sold for Part D?" and "how many flights were flown with engine D?"
2) *Derive Value*: To complete these tasks, the participants need to examine two bars in the gradient plot and perform an operation on the two values. For example, "how many parts were sold for Parts A and B combined?" and "how many flights between engine A and engine B combined?"
3) *Find Extremum*: To complete these tasks, participants need to examine all the bars in the gradient plot. For example, "which part was sold the least?" and "did manufacturer A sell the most items?" This task is perhaps the most difficult of the three in that a participant needs to read all the values in a gradient plot, compare them, and then make a decision.

5.4 Collected Data

Similar to the study by Wesslen et al. [44], we collected three pieces of data from each task that a participant completed: (1) completion time, (2) accuracy, and (3) the participant's confidence in their answers. In addition, through the demographic survey, participants provided information about their age, gender, education and proficiency in statistics and progressive visualization. Participants were also encouraged to provide feedback about their experience. Below we explain in detail our collected data.

Completion time is determined as elapsed time between the participant clicking the "start" and "stop" buttons, to remove any additional latency from typing or selecting their confidence level.

Accuracy is determined depending on the question types. Across the three tasks, there are three types of answers we collect: (1) quantitative value (e.g. "How many parts were sold for Part D?"), (2) categorical value (e.g. "Which part was sold the least?"), and (3) binary value (e.g. "Manufacturer A sold the most items in 2017. Did they sell the most in 2018?"). In each of the experiment sections below we describe how these values are used in statistical analyses in more detail.

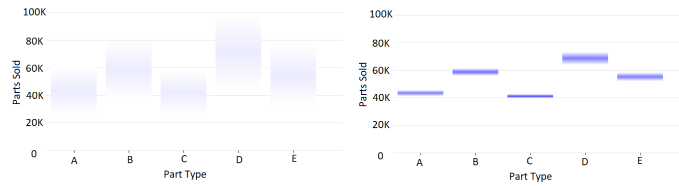
Confidence is based on a 4-point Likert scale. The participants choose between *Not at all Confident*, *Slightly Confident*, *Somewhat Confident* or *Extremely Confident* to reflect their confidence in each of their answers.

6 Experiment 1: Uncertainty Bias

The goal of this first experiment is to evaluate a participant's ability to *judge the uncertainty of intermediate results* in a progressive visualization. Dynamic uncertainty in a progressive gradient plot is controlled by modulating the variances in the values (heights of bars) and error-bounds (sizes of the error bar), defined below.

6.1 Data Generation

Different levels of variances are defined by changing both values and error-bounds by $\pm5\%$, $\pm10\%$, $\pm15\%$ and $\pm20\%$, respectively.



(a) (b)

Fig. 3: The gradient chart used in our experiments. The size of the bar represents the 95% confidence interval of the estimated value, shown at the midpoint of the bar. The color indicates the relative error of the estimate. The darker the blue, the more accurate the estimate is. (a) shows the chart early in the progression, where the error is between 70-80% for each bar. (b) shows the chart later in the progression when the values have started to converge with error rates between 10-15% for each bar.

For variances in values, each bar in a gradient plot is given a "truth" value which is represented as the actual value of a query. At each step in the progression, the "estimated" value would change to another value within the range of the variance. For example, if the "truth" value is 40,000, and the variance range is $\pm5\%$, the "estimated" value shown at each step in the progression could be somewhere between 38,000 and 42,000. The sequence of progression was generated prior to the study, so all participants in the same condition saw the same progression.

Generating multiple levels of error-bounds follows a similar paradigm. In a perfect progression, the error-bounds would decrease at each iteration, until convergence after 120 seconds. To simulate this, we create a base error-bound progression that starts at .80 and decreases by 1/100th at each step until reaching .01. An offset to the base error-bound is added at each step by the condition's percentage. For example, with error-bound condition of $\pm10\%$ and a base error-bound of .70 at that point in the progression, the error-bounds shown would be .70 plus an offset randomly chosen between -.1 and +.1. We limit the error-bounds to never be above .80 or below .01 at any time during the progression. Similar to the data generated for variances in values, the error-bound progression was randomly generated prior to the studies, so all participants in the same condition saw the same progression.

6.2 Methodology and Hypothesis

We conducted a between-subject *4 (levels of variances in the values) x 4 (levels of variances in the error-bounds)* study on Mechanical Turk[2] in which each participant was randomly assigned to one of the conditions. Following a training session, each participant was asked to complete 6 tasks:

- Using the Sales dataset (see Section 5.2):
  - T1) *Read Value*: "How many parts were sold for Part D?"
  - T2) *Derive Value*: "How many parts were sold for Part A and B combined?"
  - T3) *Find Extremum*: "Which part was sold the least?"
- Using the Flight dataset:
  - T4) *Read Value*: "How many flights were flown with engine D?"
  - T5) *Derive Value*: "How many flights between engines A and B combined?"
  - T6) *Find Extremum*: "Which engine had the least number of flights?"

Our hypothesis **(H1)** is that high variances in values and error-bounds in a progressive visualization will make it more difficult for participants to read the visualization and make decisions. This will negatively affect accuracy, speed, and confidence in their answers.

2. The online study can be found at http://valt.cs.tufts.edu/studies/progvis/p4/

6.3 Participants

We recruited 480 participants on Mechanical Turk. We removed participants who did not complete all 6 tasks and participants that did not follow instructions (e.g. they answered numerical-based answers with bar labels and vice versa), resulting in 352 participants. Of these, 45% were female, 80% had a Bachelor's or High School Diploma, 35% rated themselves as intermediate visualization experts and 44% rated themselves as novice progressive visualization experts.

6.4 Results

Based on our hypothesis, we examine the effects on completion time, accuracy, and confidence given the different levels of variances in values and error-bounds. Further, we analyze how these are affected by the task types

6.4.1 Completion Time x Variances

Figure 4a breaks down the time by task types. On average, participants completed all tasks at around the same time regardless of the conditions ($\sim$10 seconds).

We performed multiple linear regressions to analyze whether the amount of value and error-bound variances affected completion time based on the task type. Overall, with the exception of variances in values in the *Find Extremum* task, a participant's completion time does not seem to be affected by the levels of dynamic uncertainty in the progressive visualization. For the *Find Extremum* task, the higher the value variance, the more time it took to complete the task. Table 1 shows the results of the analysis.

(time)	Read Value	Derive Value	Find Extremum
overall	$F(2,349) = 0.75$ $R^2 = 0.0043$ $p = 0.47$	$F(2,701) = 0.12$ $R^2 = 0.00034$ $p = 0.89$	$F(2,701) = 5.35$ $R^2 = 0.015$ $p < 0.01$**
v_rate	$\beta = 83.21$ $p = 0.24$	$\beta = 20.81$ $p = 0.71$	$\beta = 107.03$ $p < 0.01$**
e_rate	$\beta = -27.04$ $p = 0.70$	$\beta = 16.76$ $p = 0.76$	$\beta = 59.33$ $p = 0.12$

Table 1: Multi-linear regression result on completion time versus variances in values (*v_rate*) and in error-bounds (*e_rate*). For the Find Extremum task, as the value variance increased, so did completion time.

6.4.2 Accuracy x Variances

Figure 4b shows the participants' accuracy of the different conditions and across task types. To further analyze the effect of variances in values and error-bounds on participants' accuracy in completing the tasks, we performed multiple linear regressions for *Read Value* and *Derive Value* tasks. For the *Find Extremum* tasks, we performed logistic regressions because the participants' answers were binary (either correct or incorrect). As shown in Table 2, we found no significant effects of variances in values nor in error-bounds on the participants' task accuracy.

6.4.3 Confidence x Variances

To analyze the effect of variances in values and error-bounds on participants' confidence in their answers, we performed linear regressions for all three tasks. We collected confidence data using a Likert scale, and treated the values as a continuous variable. As shown in Table 3 we found no significant effect of variances in values nor in error-bounds on the participants' confidence.

(error)	Read Value	Derive Value	Find Extremum
overall	$F(2,349) = 0.78$ $R^2 = 0.0045$ $p = 0.46$	$F(2,701) = 1.15$ $R^2 = 0.0033$ $p = 0.32$	$\chi^2(8) = 8.25$ $p = 0.41$
v_rate	$\beta = -0.0014$ $p = 0.58$	$\beta = 0.0019$ $p = 0.17$	$\beta = -0.0018$ $p = 0.93$
e_rate	$\beta = -0.0028$ $p = 0.27$	$\beta = -0.00094$ $p = 0.49$	$\beta = 0.016$ $p = 0.45$

Table 2: Regression results on accuracy (measured as amount of error) versus variances in values (*v_rate*) and error-bounds (*e_rate*). Varying value and error-bounds did not have an effect on accuracy for any of the tasks.

(confidence)	Read Value	Derive Value	Find Extremum
overall	$F(2,349) = 0.31$ $R^2 = -0.0039$ $p = 0.73$	$F(2,701) = 0.58$ $R^2 = -0.0012$ $p = 0.56$	$F(2,701) = 1.35$ $R^2 = -0.001$ $p = 0.26$
v_rate	$\beta = 0.0049$ $p = 0.50$	$\beta = -0.0017$ $p = 0.76$	$\beta = -0.0006$ $p = 0.91$
e_rate	$\beta = 0.0027$ $p = 0.71$	$\beta = -0.0056$ $p = 0.31$	$\beta = -0.0087$ $p = 0.10$

Table 3: Multi-linear regression results on participants' confidence versus variances in values (*v_rate*) and error-bounds (*e_rate*). Note that there is no significance in any of the conditions.

6.4.4 Time x Accuracy x Confidence

In addition to the analyses on fluctuations in values and error-bounds, we performed additional examinations on the relationships between completion time, accuracy (error), and confidence.

Accuracy vs. Time: We found statistically significant relationships between completion time and accuracy across all three task types in that the more time a participant spends on the task, the more accurate their answer was. Table 4 shows the result of linear and logistic regressions across the three tasks types.

(error)	Read Value	Derive Value	Find Extremum
overall	$F(1,350) = 22.67$ $R^2 = 0.061$ $p < 0.001$**	$F(1,702) = 66.38$ $R^2 = 0.086$ $p < 0.001$**	$\chi^2(8) = 74.51$ $p < 0.001$**
time	$\beta = -8.94e^{-6}$ $p < 0.001$**	$\beta = -7.21e^{-6}$ $p < 0.001$**	$\beta = -6.80e^{-5}$ $p < 0.01$**

Table 4: Regression results on accuracy (error) and completion time. Across all task types the participants were statistically more accurate (made fewer errors) when they spent more time on tasks.

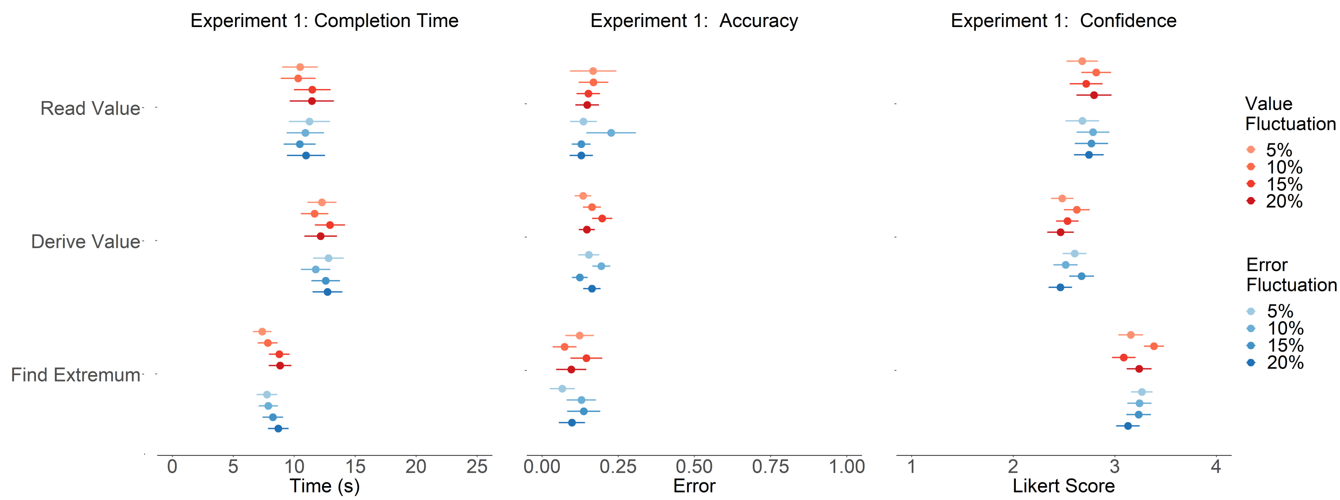
Confidence vs. Time: We performed additional linear regressions to determine if confidence correlated with completion time. As shown in Table 5, there is a significant effect for the *Derived Values* task and a possible effect for the *Read Value* task in that participants who took less time to complete their tasks were also more confident in their answers.

(confidence)	Read Value	Derive Value	Find Extremum
overall	$F(1,350) = 2.93$ $R^2 = 0.0083$ $p < 0.1$*	$F(1,702) = 18.57$ $R^2 = 0.26$ $p < 0.001$**	$F(1,702) = 1.87$ $R^2 = 0.78$ $p = 0.17$
time	$\beta = -9.43e^{-6}$ $p < 0.1$*	$\beta = -1.57e^{-5}$ $p < 0.001$**	$\beta = -7.22e^{-6}$ $p = 0.17$

Table 5: Logistic regression results on confidence and completion time. Note the significant effect for the *Derived Values* task and a possible effect for the *Read Value* task, indicating that participants who spend less time completing their tasks are more confident in their answers.

6.5 Findings

We examine our original hypothesis **(H1)** which states that "increased variance in values and error-bounds in the progressive



(a) (b) (c)

Fig. 4: (a) Mean and 95% confidence interval of completion time, (b) accuracy and (c) confidence for participants to answer each task type for each condition of variance in value and error-bounds. Lower error means the participants answered the task questions correctly more often. Higher Likert scores mean participants were more confident in their answers.

visualization will make it more difficult for participants to read the visualization and make decisions. This will negatively affect their accuracy, speed, and confidence in their answers." Our analysis results do not find evidence to support our hypothesis. As a result, we **reject (H1)**. To be more specific:

Completion Time: The analyses find that variances in values affect the participants' completion time, but only for *Find Extremum* tasks. Variances in error-bounds have no effect.

Accuracy: The analyses find that neither variances in values nor error-bounds affect the participants' accuracy.

Confidence: The analyses find that neither variances in values nor error-bounds affect the participants' confidence.

6.6 Discussion

These results are surprising because prior work in uncertainty visualization (e.g. [36]) has shown that *static* uncertainty visualizations are difficult to read for participants. For *dynamic* uncertainty visualizations, such as progressive visualizations, conventional wisdom would suggest that the additional uncertainty (stemming from the dynamic nature of the visualization) would make the task more difficult.

We speculate that the reason participants were adept in using progressive visualization with high amount of dynamic uncertainty is because the participants adopted an *estimation strategy* when observing the dynamic visualizations, similar to the behavior when using *Hypothetical Outcome Plots (HOPs)* as reported by Hullman et al. [49]. HOPs animate probability distributions and have been shown to increase accuracy in uncertainty estimates by people with no special training or statistical expertise. HOPs also allow for easier estimating trends in sampled data, using the animation to draw inferences on the underlying likelihood of the data distribution [50].

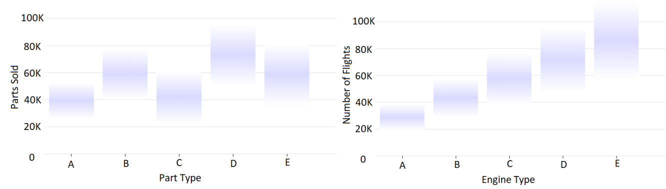
Since the progression updates can be perceived as an animation, participant strategies using progressive visualization can be similar to that when using HOPs. In particular, as the heights of the error bars fluctuated, the participants would observe the range of values and estimate the "average" (over time) of these dynamic gradient plots. Similarly, as the error-bounds fluctuated, the participants adopted the same strategy in that they largely ignored the length of the gradient (i.e. the amount of error) and simply used the center of the gradient plot as a proxy.

As a result, the participants' accuracy in their answers are only affected by their ability to determine the trend of the data and not in the fluctuations of the value or the error themselves. This finding is supported by the analysis result of the *Find Extremum* tasks (e.g "which bar is the tallest") where there is a significant effect between variances in values and completion time. We posit that the reason for this phenomenon is because the participants would require more time to keep track of the movements and mentally compute the averages of all the bars in the gradient plot. With high fluctuations in the heights of the bars, this can be difficult to do and therefore would require more time to complete.

Lastly, we find that participants who took longer to answer questions had higher accuracy in all three types of tasks but had lower confidence. The higher accuracy finding is not surprising because with the use of progressive visualization, the longer a participant waits, the more accurately they can guess the "average" points. On the other hand, the decreased confidence can be counterintuitive. We posit that this finding reflects the correlation that participants who are not confident in using progressive visualizations tend to take longer time to complete the tasks (instead of a causal relation that suggests participants become less confident in their answers with more time). However, further experiments will be needed to confirm our speculation.

7 Experiment 2: Illusion Bias

While the results of the previous experiment show that high variances in values and error-bounds do not have an immediate effect on a participant's ability to use a progressive visualization, these high variances can cause a secondary effect that has not been evaluated. As reported in prior studies [6], [11], fluctuations in a progressive visualization can occasionally result in an (unintentionally) meaningful pattern. These "false patterns" are often the result of uneven sampling or "rare events" in the data, in which the inclusion of a single outlier value in the sampling can significantly



(a) (b)

Fig. 5: In Experiment 2, participants were shown the baseline progression without false patterns first (a), then shown the "false pattern" illusion (b) at different time periods during the progression.

alter the estimation [11]. Regardless of the cause, false patterns can lead users to mistakenly assume that they have found the answer to their query and terminate the progression prematurely.

The aim of this experiment is to evaluate the effects of the this bias by deliberately introducing "false patterns" in a progressive visualization and measuring the participants' task completion time, accuracy, and confidence in their answers.

7.1 Methodology and Hypothesis

The "false pattern" used in this experiment is shown in Figure 5b. To evaluate how the false pattern might have the strongest effect on the participants, we conducted a between-subject experiment with *5 (false-pattern)* conditions. The five false-pattern conditions are: (1) no false pattern (baseline), (2) false pattern shown from 0-5 seconds, (3) from 5-10 seconds, (4) from 10-15 seconds, and (5) from 15-20 seconds[3]. Using the Sales dataset and following the same experimental design from Experiment 1, each participant was asked to complete three tasks:

T1) *Read Value*: "How many parts were sold for Part D?"
T2) *Derive Value*: "Did Part B sell more than Part A?"
T3) *Find Extremum*: "Which part was sold the least?"

Our hypothesis **(H2)** follows observations made by prior studies [6], [11] in that we hypothesize that if a false pattern appears early in the progression, a participant can be affected by *illusion bias* and will be more likely to make a decision based on this false pattern. As a result, participants will have reduced accuracy but with no effect on speed or confidence in their answers.

7.2 Participants

We recruited 216 participants on Mechanical Turk. For the bias conditions we had: (baseline) 40, (0-5 second) 38, (5-10 second) 47, (10-15 second) 44, (15-20 second) 47 participants. Of all participants, 39% were female, 77% had Bachelor's or High School Diploma, 32% rated themselves as intermediate visualization experts and 47% as novice progressive visualization experts.

7.3 Results

Based on our hypothesis, we examine the effects of the introduction of false patterns on completion time, accuracy, and confidence. Further, we analyze how these are affected by the task types (*Read Value*, *Derive Value*, and *Find Extremum*).

3. The study material can be found here:
0-5 seconds: http://valt.cs.tufts.edu/studies/progvis/p2a
5-10 seconds: http://valt.cs.tufts.edu/studies/progvis/p2_5_10
10-15 seconds: http://valt.cs.tufts.edu/studies/progvis/p2_10_15
15-20 seconds: http://valt.cs.tufts.edu/studies/progvis/p2_15_20

7.3.1 Completion Time

We looked to determine if the time that the false pattern was introduced affected participants' completion time. We found the completion time data was not normally distributed, so we used the Kruskall-Wallis rank sum tests across all task types:

- *Retrieve Value:* $\chi^2(4) = 1.12, p = 0.89$
- *Derive Value:* $\chi^2(4) = 6.57, p = 0.16$
- *Find Extremum:* $\chi^2(4) = 15.31, p < 0.005$**

The results indicate a significant effect for the *Find Extremum* task. A post-hoc analysis using the Dunn test with Benjamini-Hochberg adjustment finds significance between two conditions:

- Baseline vs. 5-10 second delay ($p < 0.005$**)
- 5-10 second delay vs. 15-20 second delay ($p < 0.05$**)

Together, the results suggest that participants were faster in the baseline (no false pattern) condition when completing the *Find Extremum* task. The completion times for all other conditions are higher in other conditions in which a false-pattern is introduced (see Figure 6a).

7.3.2 Accuracy

We examined the timing of false patterns and whether it had an impact on accuracy. Similar to completion time, we found that the data was not normally distributed, so we performed the Kruskall-Wallis rank sum test for the *Retrieve Value* task (because the answer participants provided were integers and thus were treated as a continuous variable), and chi-squared tests for the *Derive Value* and *Find Extremum* tasks because the participants' responses were measured in binary (correct or incorrect). Our results are as follows:

- *Read Value:* $\chi^2(4) = 22.49, p < 0.001$**
- *Derive Value:* $\chi^2(4) = 21.51, p < 0.001$**
- *Find Extremum:* $\chi^2(4) = 23.93, p < 0.001$**

In sum, we found strong evidence that the bias conditions had an effect on error in all three tasks. Specifically, as shown in Figure 6b, the introduction of a false-pattern in the progression could reduce the participants' accuracy. However, this effect tapered off in conditions where the false pattern was introduced later in the progression.

7.3.3 Confidence

To examine if the timing of false patterns affected participants' confidence, we again ran the Kruskall-Wallis rank sum test:

- *Read Value:* $\chi^2(4) = 2.46, p = 0.65$
- *Derive Value:* $\chi^2(4) = 13.62, p < 0.01$**
- *Find Extremum:* $\chi^2(4) = 28.20, p < 0.001$**

We found strong evidence that bias had an effect on confidence for the *Derive Value* and *Find Extremum* tasks, but did not have an effect on the *Read Value* tasks. Similar to the previous result, this effect was more pronounced if the false pattern was introduced earlier (i.e. in the 5-10 and 10-15 second delay conditions, see Figure 6c). The effect tapered off if the false pattern was introduced later in the progression.

7.4 Findings

We examine our hypothesis **(H2)** that "if a false pattern appears early in the progression, a participant can be affected by illusion bias and will be more likely to make a decision based on this false pattern. As a result, participants will have reduced accuracy but with no effect on speed or confidence in their answers." Our

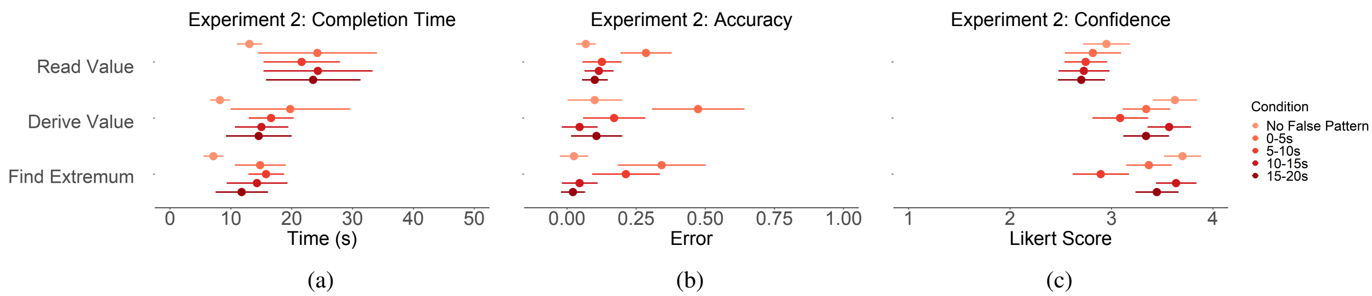


Fig. 6: Experiment 2 (a) Mean and 95% confidence interval of completion time, (b) accuracy and (c) confidence for participants to answer each task type for each condition of variance in value and error-bounds. Lower error means the participants answered the task questions correctly more often. Higher Likert scores mean participants were more confident in their answers. *Derive Value* and *Find Extremum* tasks showed strong evidence of the bias affecting participants' confidence in their answers.

analysis results find evidence that supports this hypothesis. As a result, we **confirm (H2)** on the basis that:

Completion Time: The analyses find that the introduction of a false pattern during the progression affects the participants' completion time, but only for *Find Extremum* tasks.
Accuracy: The analyses find that the introduction of a false pattern decreases the participants' accuracy. The effect is more pronounced if the false pattern is introduced earlier in the progression.
Confidence: The analyses find the introduction of a false pattern decreases the participants' confidence in the *Derive Value* and *Find Extremum* tasks. Similar to the analysis on accuracy, this effect is more pronounced if the false pattern is introduced earlier in the progression.

7.5 Discussion

Overall, our results confirm the previous reports by Moritz et al. [11] and Turkay et al. [6] in which the authors observed that users of a progressive visualization could be negatively influenced by seeing false patterns during the progression.

However, digging deeper into our experimental results suggest that the finding is a little more nuanced. First, we observe that task difficulty plays an important role when considering the impact of false patterns in a progressive visualization. For the most difficult task, *Find Extremum*, we find that the participants' completion time, accuracy, and confidence are all negatively impacted. Conversely, for the easiest task, *Read Value*, only the their accuracy is affected.

Secondly, the timing of *when* the false pattern appears can be a crucial factor. We observe that an early introduction of a false pattern can significantly reduce the participant's performance, most notably their accuracy (see Figure 6b). However, this effect tapers off if the false pattern appears later in the progression. While one simple reason for this finding is that participants may have completed the task before the false pattern appeared (which is least likely to happen in the 0-5 second condition). However, we note that on average participants completed their task between the 10-20 second mark (see Figure 6a). So this simple explanation does not explain the differences between the 5-10 and 10-15 conditions in which the negative impacts on completion time, accuracy, and confidence are more pronounced in the 5-10 condition.

8 Experiment 3: Control Bias

In the previous two experiments, we evaluate the biases that affect the use of progressive visualization when used as a "passive" mechanism for delivering information incrementally. In practice,

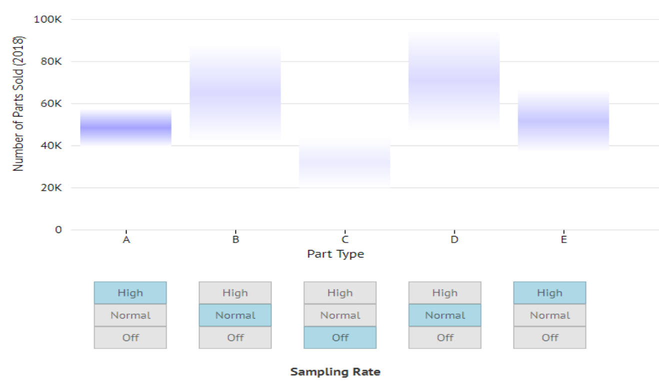


Fig. 7: Buttons below each bar allowed participants to steer the progression. Setting a bar to "high" caused the progression of that bar to increase at a faster rate, presenting a higher confidence estimate faster to the user. Setting a bar to "Off" caused the progression to stop for that bar.

most modern progressive visualization systems allow some amount of user control. Referred to as "steerable" progressive visualization, these interactive progressive visualizations allow the user to control the amount of progression and steer the computation or sampling. As a result, the user can reduce computation time by focusing on the parts of data that are relevant to their analysis [6], [7], [8], [9], gain better awareness and control of the system [5], and in turn build trust in the system [8].

However, given the results of the previous experiment which show that users of progressive visualization can be susceptible to the *illusion bias*, it is plausible to hypothesize that the additional control over the computation could further mislead the user. In this experiment, we examine this potential bias and its effects. In particular, we examine whether users of an interactive progressive visualization can effectively utilize *steering* or succumb to *control bias* where the users perform either non-optimal or incorrect steering that lead to wrong decisions.

8.1 Methodology and Hypothesis

We conducted an observational study on Mechanical Turk where we tracked participants' interactions using a progressive visualization. In our experiment, the participants were asked to use a steerable progressive visualization (as shown in Figure 7) to complete a number of tasks. Each of these tasks had a known ground truth both in terms of the correct answer as well as the optimal configuration of the sampling[4]. We recorded the participants' interactions and

4. http://valt.cs.tufts.edu/studies/progvis/p4/

analyzed how much they deviated from the optimal configuration and how often the participants came to incorrect decisions due to steering. Participants had the same experiment setup as described in Section 5, with the addition of information on steering and explanation of controls. Their training task included using the steering controls to answer a sample question.

This experimental approach is similar to that of the work by Wall et al. [14], [17] in that we measure statistical deviation from the control group to identify biased behaviors. However, contrary to the work by Wall et al. where participants are expected to explore data evenly if they are not under the influence of cognitive biases, in our experiment we expect the opposite behavior in that the participants should explore a specific subset of the data if they are not under the influence of *control bias*. This difference is due to the nature of the tasks used in these two sets of experiments. In the experiments by Wall et al., participants performed exploratory data analysis where data coverage is a key indicator of success in the task, whereas our study is more akin to a confirmatory data analysis where a participant is looking for a specific piece of data to complete the task.

While the goals of these experiments differ, the statistical approach to detecting bias is the same. Specifically, in our experimental setup, the tasks that the participants were asked to complete were the same as those used in Experiment 1 but with a steering progressive visualization. Initially, all the sampling rates for all bars were set to "Normal." For the *Read Value* tasks, the optimal configuration was to set the sampling rate for the bars specified in the task to "High" and the remaining sampling rates to "Off." For example, to answer Q1 ("How many parts were sold for Part D?"), the participant should turn off sampling for all bars except for Part D. For Part D, the setting for sampling should be "High"[5]. For the *Derive Value* tasks, optimal was to set two of the bars to "High" and the rest to "Off." Finally, for the *Find Extremum* tasks, the optimal strategy was not to interact with the steering interface at all, but instead leave all the sampling rates to "Normal."

Our hypothesis **(H3)** is that when using an interactive steerable progressive visualization, some participants will succumb to *control bias* in that they cannot reason about the effect and outcome of an uneven sampling process. As a result, we will find a number of participants who incorrectly tune the parameters of the progression resulting in slower completion times and lower accuracy.

8.2 Participants

We recruited 59 participants on Mechanical Turk. We dropped the results of participants who did not answer all 6 questions or clearly did not follow instructions. This includes participants who did not interact with at least one question or completed the task in less than 2 seconds, which was too short of a time to see an update to the progression. This left us with 33 participants. Of these, 45% were female, 66% had a Bachelor's or High School Diploma, 39% rated themselves as intermediate visualization experts and 48% as novice progressive visualization experts.

8.3 Results

First, we analyzed the percentages of the participants who interacted with the steering interface. Although all the participants had learned to interact with the steering interface during the training phase of the experiment (see Section 5, Experiment Setup), we observed that not all the participants interacted with the steerable interface on all tasks.

For the remainder of the participants (who interacted with the steering interface), we computed a *penalty score* for each of their task performance based on how different the participant's configurations were from the optimal configuration, and how long they were in the non-optimal configuration. We normalized each penalty score based on the total time the participant took to complete the task. This allowed us to compare penalty scores between tasks and between participants that took varying amounts of time. Specifically, our penalty score is defined as:

$$S = \left(t \sum_{i}^{n} |L_i - W_i| \right) / T_{total} \quad (1)$$

In this equation, t is the time in seconds spent in a configuration, n is the number of steerable bars in the visualization, L_i is the ideal configuration for bar i and W_i is the configuration set by the participant for bar i. This score is normalized by T_{total}, which is the total amount of time (in seconds) elapsed for completing a task. L_i and W_i will be 0 (Off), 1 (Normal) or 2 (High).

The final score S will be between 0 and $2n$ where 0 means the participant had the ideal configuration for the duration of the task while $2n$ mean the configuration was the furthest possible from the ideal for the duration of the task. Since the visualization used in our experiment had 5 bars ($n = 5$), the worst possible score is 10.

8.3.1 Number of Participants who Used Steering

We first calculated how many of the participants interacted with the steerable progressive visualization interface. The percentages of trials where participants used steering were:

- *Read Value*: 80%
- *Derive Value*: 83%
- *Find Extremum*: 45%

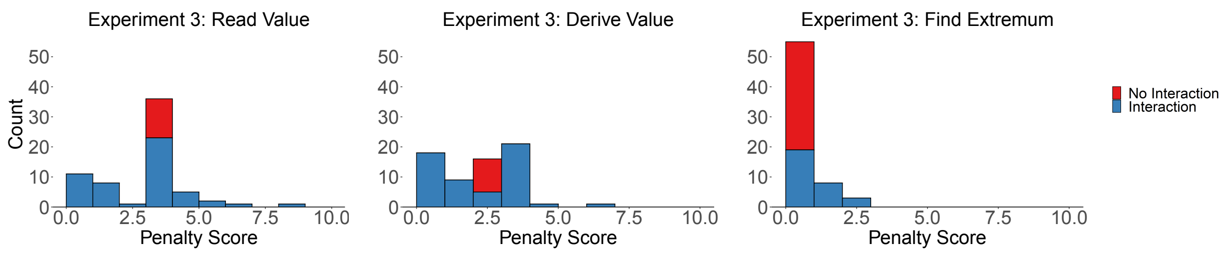
Since *Read Value* and *Derive Value* tasks require turning off sampling for multiple bars, it naturally follows that most participants that understood the interface would interact with the steering controls. What is surprising is that nearly half of the participants **did not interact** during the *Find Extremum* tasks. Recall that the ideal strategy for the *Find Extremum* tasks is to not interact with the visualization. The fact that about half the participants utilized the optimal strategy suggests that our participants on Mechanical Turk were able to understand the concept of progressive visualization and utilize it effectively. This is especially true when contrasted with the *Read Value* and *Derive Value* tasks in which 80% and 83% of the participants (correctly) interacted with the visualization, respectively. The significantly lower percentage in the *Find Extremum* tasks (45%) is an indicator that the participants understood the task and were able to use the progressive visualization correctly.

8.3.2 Penalty Score

We analyzed how well the participants were able to use the steerable progressive visualization optimally. Figure 8 shows the distributions of scores for each task type, and whether or not the participant steered the progression at all for the task. For all tasks, the mean penalty was 2.02. Per task means were:

- *Read Value:* 3.32 (baseline: 4.0)
- *Derive Value:* 2.25 (baseline: 3.0)

5. Setting sampling for Part D to be "Normal" would have the same result because all computing resources will be devoted to sampling for Part D. However, we do not consider that as optimal because participants would not be aware of this system-level optimization.



(a) (b) (c)

Fig. 8: Distribution of penalty score by task type for Experiment 3. The stack breaks the histograms down by participants that interacted with the steering interface during the task (Interaction) and those that didn't interact with the steering interface at all (No Interaction).

- *Find Extremum:* 0.48 (baseline: 0.0)

Where 'baseline" is the penalty if none of the controls were touched, and therefore the starting configuration did not change.

Since the penalty score is based on the configuration of the steering weights, the further the configuration is from optimal at the beginning of the task, the more likely the penalty score will be higher. The *Read Value* task starts the furthest from optimal, requiring the participants to turn off sampling from 4 bars. The *Derive Value* task only requires the participants to turn off 3 bars. A penalty score can accumulate if the participant does not set the optimal configuration at the start of the progression. The *Find Extremum* tasks start in the ideal configuration, so no penalty is present until the participant interacts with the controls.

8.3.3 Completion Time

To evaluate if non-optimal steering affected completion time, we performed multiple linear regressions on penalty score and participants' time in completing the tasks. Figure 9 shows the data and the regressions. When analyzing all task types together, an increase in penalty score correlated with an increase in completion time, however this effect was not found on any of the individual task types:

- *All Tasks:* $F(1,196) = 3.83, R^2 = 0.02, p = 0.05$**
- *Read Value:*$F(1,64) = 0.20, R^2 = 0.003, p = 0.66$
- *Derive Value:* $F(1,64) = 0.98, R^2 = 0.015, p = 0.32$
- *Find Extremum:* $F(1,64) = 2.98, R^2 = 0.044, p = 0.08$

However, we analyzed the effect of those that steered the progression vs. those that did not interact with the interface on a per task basis. We noted that those that did not interact with the interface completed the tasks faster than those that did, for all tasks:

- *All Tasks:* $F(1,196) = 40.2, R^2 = 0.17, p < 0.001$**
- *Read Value:*$F(1,64) = 8.34, R^2 = 0.115, p = 0.005$**
- *Derive Value:* $F(1,64) = 7.88, R^2 = 0.109, p = 0.007$**
- *Find Extremum:* $F(1,64) = 9.035, R^2 = .124, p = 0.004$**

8.3.4 Accuracy

We look to evaluate if non-optimal steering affected accuracy. To analyze the effect of the penalty score on the participants' accuracy in completing the tasks, we performed multiple linear regressions across all task types as well as for *Read Value* and *Derive Value* tasks. For the *Find Extremum* tasks, we performed logistic regressions because the participants' answers were either correct or incorrect. Figure 10 shows the data and the regressions.

- *All Tasks:* $F(1,196) = 2.97, R^2 = 0.015, p = 0.08$
- *Read Value:* $F(1,64) < 0.001, R^2 < .001, p = 0.97$
- *Derive Value:* $F(1,64) < 0.001, R^2 < 0.001, p = 0.99$
- *Find Extremum:* $\chi^2(8) < 0.001, p = 1$

To analyze the effect of interaction vs. no interaction on the participants' accuracy in completing the tasks, we similarly performed multiple linear regressions across all task types and for *Read Value* and *Derive Value* tasks. For the *Find Extremum* tasks, we again performed logistic regressions.

- *All Tasks:* $F(1,196) = 0.02, R^2 < 0.001, p = 0.87$
- *Read Value:* $F(1,64) = 6.952, R^2 = 0.098, p = 0.01$**
- *Derive Value:* $F(1,64) = 1.211, R^2 = 0.019, p = 0.28$
- *Find Extremum:* $\chi^2(8) =< 0.001, p = 1$

The significance of interaction on a *Read Value* task is that those who interacted had higher accuracy than those who did not. Although *Read Value* tasks have statistical significance, this is likely due to a handful of extreme outliers, as shown in Figure 10b. The general trend however is consistent across all tasks.

8.3.5 Confidence

We found a significant effect of increased penalty score resulting in higher confidence across all task types, but found no significant effect on a per task type basis. Figure 11 shows the multiple linear regression on the participants' confidence in their answers on a per task basis relative to their penalty score.

- *All Tasks:* $F(1,196) = 40.97, R^2 = 0.17, p < 0.001$**
- *Read Value:* $F(1,64) = 3.07, R^2 = 0.045, p = 0.08$
- *Derive Value:* $F(1,64) = 0.61, R^2 = 0.009, p = 0.43$
- *Find Extremum:* $F(1,64) = 0.75, R^2 = 0.011, p = 0.39$

Similarly, we found the same significant effects on confidence relative to whether participants interacted or not with the steering interface:

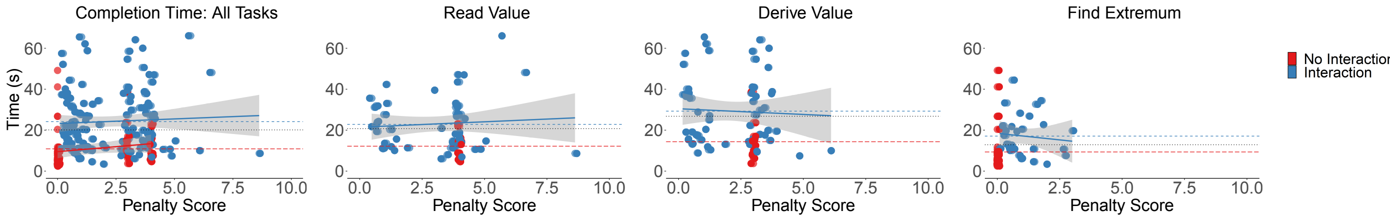
- *All Tasks:* $F(1,196) = 14.47, R^2 = 0.69, p < 0.001$**
- *Read Value:* $F(1,64) = 2.67, R^2 = 0.04, p = 0.11$
- *Derive Value:* $F(1,64) = 0.29, R^2 = 0.0004, p = 0.87$
- *Find Extremum:* $F(1,64) = 0.71, R^2 = 0.011, p = 0.40$

Those who interacted with the steering controls had a slightly higher confidence on average (3.2 for those that interacted vs 3.1 for those that did not). However, we did see a trend that as participants progressed through the tasks, they became more confident in their responses, indicating they may have become more comfortable using the progressive interface as time went on, as shown in Figure 12.

- $F(1,196) = 27.54, R^2 = 0.123, p < 0.01$ **

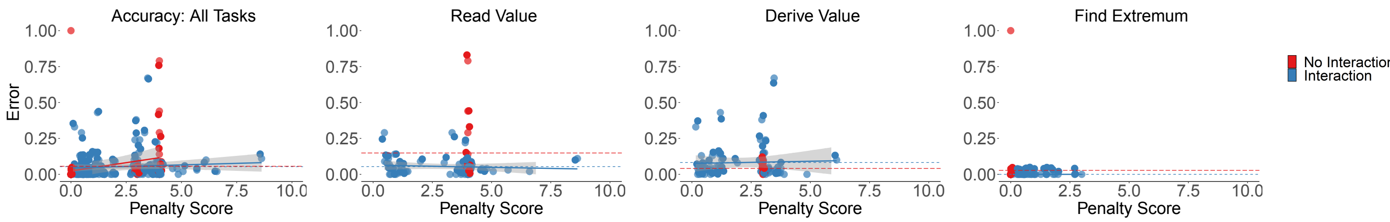
8.4 Findings

We examine our hypothesis **(H3)** that "some participants will succumb to *control bias* when using a progressive visualization.



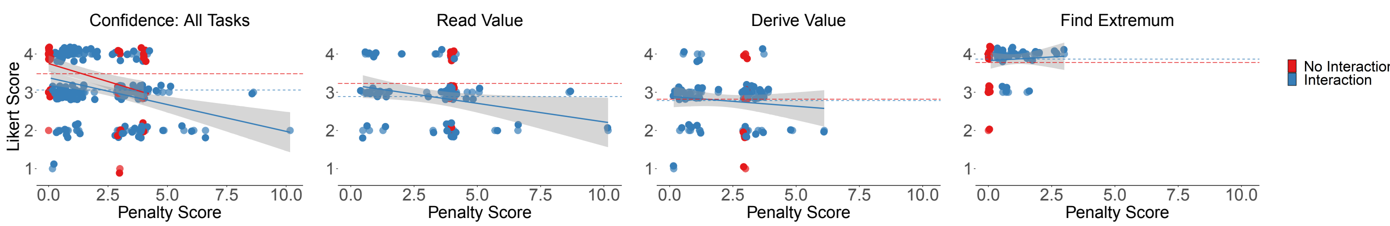
(a) (b) (c) (d)

Fig. 9: Regression charts for *control bias* showing how completion time is affect by the penalty score for each task type. Dashed line is the mean value. Those that exhibited *control bias* took longer to complete tasks than those who did not. Note that points in the scatterplot have been *jittered* to reduce overplotting (*Plot jitter width=0.1, height=0.1)*



(a) (b) (c) (d)

Fig. 10: Regression charts for *control bias* showing how accuracy is affected by penalty score for each task type. Dashed line is the mean value. Unnecessary interactions did not affect accuracy in our study. Note that points in the scatterplot have been *jittered* to reduce overplotting (*Plot jitter width=0.1, height=0.1)*



(a) (b) (c) (d)

Fig. 11: Regression charts for *control bias* showing how confidence is affected by penalty score for each task type. Dashed line is the mean value. Those that steered the progression had higher confidence than those who did not. Note that points in the scatterplot have been *jittered* to reduce overplotting (*Plot jitter width=0.1, height=0.1)*

As a result, these participants will incorrectly tune the parameters of the progression resulting in slower completion time and lower accuracy." Our analysis results do not find evidence that support this hypothesis. As a result, we **partially reject (H3)** due to:

Completion Time: Analyses show that participants who succumbed to *control bias* (i.e. interacted with the progressive visualization unnecessarily) took more time to complete the tasks.
Accuracy: Analysis did not show that the unneeded interactions affected the participants' accuracy in completing tasks.
Confidence: Analyses show participants who steered the progressive visualization had higher confidence than those who did not. However, we note that whether this is due to the longer completion times and therefore an increased sense of familiarity or due to an improved sense of control provided by a steerable visualization would require further analysis.

8.5 Discussion

The outcome of this experiment is surprising. As hypothesized, when given a steerable progressive visualization, many participants were not able to reason about how to steer the visualization in an optimal manner and succumbed to *control bias*. As a result, these participants interacted with the visualization unnecessarily and took longer to complete the tasks.

However, what is unexpected is that the non-optimal steering did not affect the participants' ability to analyze the results and come to the right decisions. Further, although the non-optimal steering increased task completion times, it provided the benefit that participants had higher confidence in their decisions. Since there is no difference in accuracy, the increased confidence can be considered a positive feature of steerable progressive visualization.

We also found evidence that novices in visualization can make sense of steering the progression, making steerable progressive visualization easy to understand and use (recall that 48% of our participants rated themselves as novice progressive visualization experts). It has been speculated that because of the combined complexities of steering and progression, interactive progressive visualization is not well suited for novice users [4]. However, our experiment results which found about half of the participants correctly **did not interact** in the *Find Extremum* task suggest otherwise. Although a definitive evaluation is still needed to confirm our observation, this finding gives credence to the possibility that steerable progressive visualizations can be used in a wider range of contexts than those requiring expert users such as scientific

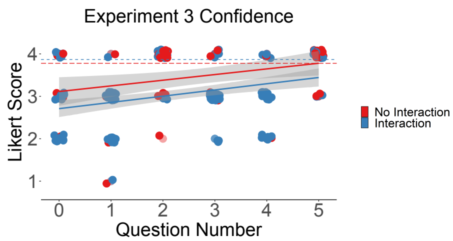


Fig. 12: Regression chart for *control bias* confidence for all questions. Dashed line is the mean value. Confidence increased as participants progressed through the tasks. Note that points in the scatterplot have been *jittered* to reduce overplotting (*Plot jitter width=0.1, height=0.1)*

computing [6], [9], debugging database queries [10], [11], etc.

9 Experiment 4: Anchoring Bias

The outcome of the previous study on *control bias* found that participants under the influence of the bias were slower but no less accurate in task completion using a progressive visualization. In this experiment, we further investigate whether priming can have unintentional negative effects on participants. Commonly known as *anchoring bias*, people primed with unrelated information or given specific instructions have been known to have significant effect on their exploration of data when using a visualization and their decision making. Previous work in the visualization community has examined *anchoring bias* by priming subjects with visual anchors [42], [51], [52] or with different descriptions of the task [14]. These found that using a visualization containing a subset of the data can lead to an over reliance on that subset, often neglecting other related information when making a decision.

However, little is known about how *anchoring bias* might affect the use of progressive visualizations; or conversely, whether the use of progressive visualizations may mitigate its effects (similar to our findings from the *control bias* experiment). We detect *anchoring bias* similar to previous work in visualization, by priming participants with information, and using interaction logs to determine if participants focus on a subset of the data when completing tasks.

9.1 Methodology and Hypothesis

The goal of this experiment is to answer the question, does anchoring bias have an effect on participants' use of a progressive visualization? Following a similar experimental setup as Experiment 3, participants recruited from Amazon's Mechanical Turk were asked to use a steerable progressive visualization to complete a set of tasks (see Figure 7). However, unlike Experiment 3, each of the tasks now includes priming participants with information[6].

We modeled our study on that used by Wesslen et al. [44], providing a cue and measuring speed, accuracy and confidence to gauge participant performance. For example, instead of "which part was sold the fewest?" the new task says "Part A sold the fewest in 2017. Did it in 2018?" Participants who steer the progressive visualization towards increasing the sampling for Part A are considered to exhibit behaviors that indicate they may be under influence of *anchoring bias*.

Formally, we define that a participant is under the influence of *anchoring bias* if: (1) the participant increased the sampling towards the Part mentioned in the prior but none of the other Parts, or (2) if they decreased the sampling of all Parts except the one mentioned in the prior.

Given the nature of the prime, all tasks in this experiment are *Find Extremum* tasks:

T1) "Part A was sold the fewest in 2017. Did it in 2018?"
T2) "Manufacturer A sold the fewest number of items in 2017. Did they sell the fewest in 2018?"
T3) "Engine Type A had the fewest flights in 2017. Did it in 2018?"
T4) "Part D was sold the most in 2017. Did it sell the most in 2018?"
T5) "Manufacturer A sold the most items in 2017. Did they in 2018?"
T6) "Engine Type D had the most flights in 2017. Did it in 2018?"

When primed about the data, our hypothesis **(H4)** is that when using a progressive visualization some participants will exhibit *anchoring bias* where they are more likely to make a decision that conforms to the priming information. As a result, they will have reduced accuracy and spend less time completing the tasks, while having an increased level of confidence in their answers.

9.2 Participants

We recruited 146 participants on Mechanical Turk. After removing participants who did not complete or answer all 6 questions, we were left with 133 participants. Of these, 35% were female, 73% had a Bachelor's or High School Diploma, 33% rated themselves as intermediate visualization experts and 40% rated themselves as novice progressive visualization experts.

9.3 Results

We examined the results of the experiment by first categorizing the participants' trials into three groups:

1) **No-Interact**: In these trials, the participants did not interact with the progressive visualization. Note that because the tasks are *Find Extremum* tasks, the optimal strategy is to *not* interact with the progressive visualization. Out of the 798 trials, participants did not interact in 525 of them (66%).
2) **Interact with (Anchoring) Bias**: In these trials, the participants interacted with the progressive visualization. Further, their interactions exhibit *anchoring bias* based on our definition (see Section 9.1). Out of 798 trials, in 49 of them participants exhibited *anchoring bias* (6%).
3) **Interact without Bias**: In these trials, participants interacted with the progressive visualization but did not exhibit *anchoring bias*. Of the 798 trials, 224 fall into this category (28%).

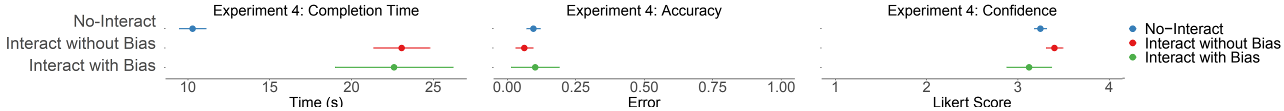
In the sections below, we analyze the participants' speed, accuracy, and confidence using these three groups.

9.3.1 Completion Time

Mean completion time for a task was ($M = 14.64s, SD = 12.41$). Mean completion time when anchoring bias occurs (*Interact with Bias*) was ($M = 22.6s, SD = 12.57$), non bias interaction (*Interact without Bias*) was ($M = 23.06s, SD = 13.12$) and ($M = 10.28s, SD = 9.53$) when no interaction was taken (*No-Interact*), as shown in Figure 13a.

We found that completion time was strongly affected by the conditions via Kruskal-Wallis Rank Sum Test ($\chi^2(2) = 201.97, p < 0.001$). To isolate which conditions had an effect, we ran the Dunn test for multiple comparison with uneven sample sizes using the Benjamini-Hochberg adjustment. We found that completion time was strongly affected by whether or not a participant interacted

6. The online study can be found at http://valt.cs.tufts.edu/studies/progvis/p1a/



(a) (b) (c)

Fig. 13: Experiment 4 mean and 95% confidence interval of (a) completion time, (b) accuracy and (c) confidence per condition. Those that showed a bias or interacted with the controls were slower in completing tasks than those not interacting, however accuracy was not affected by condition. Participants exhibiting anchoring bias potentially had lower confidence than those in the Interact without Bias condition.

with the steering controls, with those interacting taking longer than those that did not. We did not find an effect between biased and non-biased interaction types.

- *No-Interact vs. Interact without Bias*: $p < .001$**
- *No-Interact vs. Interact with Bias*: $p < .001$**
- *Interact without Bias vs. Interact with Bias*: $p = 0.97$

Therefore we cannot conclude that the bias itself had an effect on completion time, but we can confirm our results in the *control bias* study that those who interacted were slower than those who did not.

9.3.2 Accuracy

Overall, tasks were answered correctly 91.3% of the time. When *anchoring bias* occurred, tasks were answered correctly 89.8% of the time (*Interact with Bias*), and when a non-biased interaction approach was taken, tasks were answered correctly 93.7% of the time (*Interact without Bias*). The non interaction approach answered correctly 90.4% of the time (*No-Interact*).

Because all tasks required answers that were either strictly right or wrong, we performed chi-squared tests to test if the interaction conditions had an effect on error. We found that error was not affected by bias or interaction ($\chi^2(2) = 2.35, p = 0.31$).

We further examined the case where the information included in the priming happened to be the correct answer, which was the case for Tasks 3 and 4 (so the correct answer was "yes"). For Tasks 1, 2, 5, and 6, the priming information did not match the correct answer, so the correct response was "no":

- *Prior Matched Task Answer:* $\chi^2(2) = 1.19, p = 0.55$
- *Prior Did Not Match Task Answer:* $\chi^2(2) = 01.24, p = 0.54$

This result suggest that error was not affected by whether the priming information matched the correct answer. When the two matched, participants were correct 87.5% of the time; when it did not matched, they were correct 93.2% of the time.

9.3.3 Confidence

Mean confidence for a task was ($M = 3.30, SD = 0.77$). Mean confidence when anchoring bias occurred (*Interact with Bias*) was ($M = 3.12, SD = 0.85$), when non-biased interaction (*Interact without Bias*) was taken was ($M = 3.4, SD = 0.70$) and ($M = 3.25, SD = 0.78$) when no interactions took place (*No-Interact*). Distribution of Likert scores are shown in Figure 13c.

Because the confidence data was not normally distributed, we performed Kruskal-Wallis rank sum tests to determine if the bias condition had an effect on user confidence. We found that bias could have a significant effect on confidence ($\chi^2(2) = 7.47, p = 0.02$**).

To isolate which conditions had an effect, we ran the Dunn test for multiple comparison with uneven sample sizes using the Benjamini-Hochberg adjustment. We found significance that those who *Interact without Bias* had higher confidence than those who did not interact.

- *No Interact vs. Interact without Bias* $p = 0.05$**
- *No Interact vs. Interact with Bias:* $p = 0.37$
- *Interact with Bias vs. Interact without Bias:* $p = 0.06$

9.4 Findings

We examine our original hypothesis **(H4)** stating that "when using a progressive visualization some participants will exhibit *anchoring bias* where they are more likely to make a decision that conforms to the priming information. As a result, they will have reduced accuracy and spend less time completing the tasks while having an increased level of confidence in their answers." The analyses results **reject H4** in that although in some small percentage of the trials the participants exhibit *anchoring bias* (6%), these participants did not make any more mistakes than those who did not. They also did not spend more time solving the tasks (not more so than those in the *Interact without Bias* condition), nor did they demonstrate a change in confidence. More specifically:

Completion Time: We found that those exhibiting *anchoring bias* (*Interact with Bias*) took longer to complete tasks than those who did not interact with the steering controls (*No-Interact*). However, there is no statistical difference between those who exhibited *anchoring bias* from those who interacted with the visualization but did not exhibit *anchoring bias* (*Interact without Bias*).
Accuracy: Accuracy was not affected whether or not the participant exhibited *anchoring bias*.
Confidence: Confidence was not affected whether or not the participant exhibited *anchoring bias*.

9.5 Discussion

Overall, the results of this experiment demonstrate that participants could be primed to exhibit *anchoring bias*. Although the success of the priming is low (6%), we were able to evaluate their behaviors against the two conditions in Experiment 3, namely: (a) the baseline condition where the participants (correctly) did not interact with the progressive visualization (*No-Interact*), and (b) the participants who incorrectly interacted with the visualization but did not exhibit *anchoring bias* (*Interact without Bias*).

Unexpectedly, we did not find a significant difference between participants exhibiting *anchoring bias* from those in the *Interact without Bias* condition in terms of accuracy and speed. This is contrary to our original hypothesis that those under the influence of *anchoring bias* would hone in on the wrong answer more quickly. We were also incorrect in hypothesizing that participants exhibiting *anchoring bias* would have higher confidence in their answers.

10 Progressive Visualization, Helpful or Harmful?

The results of these experiments confirm some of the existing suspicions about the benefits and pitfalls of progression visualization while revealing some new findings. On the positive side, across these experiments, participants completed the tasks fairly quickly ($M = 13.85 seconds, SD = 15.38$). Since the progressions in our experiments require 120 seconds to fully converge, this represents a 88.35% in time saving. While the percentage in time saving can be an artifact of our experimental setup, we can still assume that for computational tasks that take minutes or hours to complete, participants using a progressive visualization can make decisions in a shorter amount of time.

However, the savings in time can come with the cost of reduced accuracy. Specifically, for *Retrieve Value* tasks, the average error across all experiments is ($M = 11.42\%, SD = 25.24$), for *Derive Value* tasks it is ($M = 13.58\%, SD = 22.29$), and for *Find Extremum* tasks answers were incorrect 9.83% of the time.

Digging deeper into the cause of the reduced accuracy, we find that the cognitive biases suggested by Micallef et al. can indeed play a role. Of the two data-oriented biases we evaluated (*uncertainty bias* and *illusion bias*), we find that *uncertainty bias* in general does not affect the participants' performance – when given dynamic and uncertain information, the participants were able to develop heuristics that were unexpectedly effective and accurate.

We find that *illusion bias* can have significant and detrimental effect on a user, especially if the false-patterns appear early in the progressive visualization. This effect tapers off if the false-pattern appears later in the progression. In cases where this bias was present, results showed longer time spent (on the *Find Extremum* tasks), and reduced confidence in decisions (in *Derive Value* and *Find Extremum* tasks).

When evaluating action-oriented biases (*control bias* and *anchoring bias*), we find participants exhibited *control bias* about 70% of the time, steering the progression in non-optimal or incorrect ways. We further find that *anchoring bias* was much less common in our study, with participants only exhibiting the bias 6% of the time.

For the participants who exhibited these biases, we find that they took longer to complete tasks, but did not suffer a loss of accuracy and sometimes increased confidence in their answers. This can suggest that allowing participants to steer the progression can let them become more familiar with the interface and data, providing an overall benefit. This is similar to the findings reported by Wall et al. [53] which found that biases when viewed as filters and preconceptions can be both beneficial and detrimental depending on the circumstances.

11 Limitations

Given our experiments, there are some limitations to the conclusions we can draw. Our experiments were done with simulated progressions, although modeled to reflect real world sampling. By limiting the duration to 120 seconds, any time savings discussed are relative to our experiment conditions. Further study with real world progressive systems and expert users are needed to gauge the potential time savings when calculations can last for hours or days. Additionally, biases that are present during the progression can be negated if the cost of waiting for the progression to complete is small. Therefore even when exhibiting biased steering, a user can wait for the final (correct) answer and bias will not affect accuracy.

We did not explicitly test for participant's understanding of error bounds and statistical sampling. Although progressive visualization users are often analysts with a statistical background, for this study the details of error bounds and confidence intervals can be largely ignored. Participants needed to primarily understand the mean and be able to follow the trends of the progression, and the actual visualization of uncertainty may not have played a large role in decision making. As shown in other Mechanical Turk studies assessing understanding of uncertainty via animations [49], [50], participants were able to interpret and make reasonably accurate estimates without special training.

With regards to steering, we did not explicitly test if participants fully understood the action of steering the sampling. What we can note however, is that nearly half of those in the *Control Bias* study and nearly 2/3 of those in the *Anchoring Bias* study did perform optimal steering (or abstained from interacting when appropriate). This leads us to believe that steering can be used and understood by non-experts.

12 Conclusion

In summary, we conducted four experiments to evaluate the benefits and drawbacks of progressive visualization based on the four cognitive biases suggested by Micallef et al. [13]: *uncertainty bias*, *illusion bias*, *control bias* and *anchoring bias*. Our results suggest a cautious but promising use of progressive visualization for data analysis:

- **Ease of Use**: Although progressive visualization is often thought to be an "advanced" visualization technique [4], we found that participants recruited from Amazon's Mechanical Turk had little trouble using the technique (over all of our experiments 43% of participants rated themselves as progression visualization novices). This is evident from the low error rates in task completion (in all four experiments) and similar to other Turk-based studies on uncertainty [49], [50]. Further, many of the participants were able to use steering correctly and chose *not* to interact with the progressive visualization in *Find Extremum* tasks in Experiments 3 and 4, which indicate a sophisticated understanding of the caveats when using progressive visualization. Together, these results suggest the potential for a wider application of the progressive visualization technique beyond expert users.
- **Savings in Time**: Across all four experiments, we found that the participants were able to complete their tasks very quickly (between 10 to 15 seconds) relative to the time it took for the progression to complete (120 seconds). Although we cannot directly extend this time reduction to other scenarios, our results indicate the potential progressive visualization has in time savings. Even in cases where the participants were under the influence of *control bias* or *anchoring bias* and took twice as much time as those not under a bias, our results indicate the use of a progressive visualization could result in savings in time.
- **Tradeoff in Accuracy**: Savings in time come with the tradeoff of possible reduction in accuracy. Across the three types of tasks (*Read Value*, *Derive Value*, and *Find Extremum*), we observed a reduction of approximately 10-15% in accuracy. Of the four biases, we found that the bias that contributed the most to the reduction in accuracy was the *illusion bias* in which a false-pattern was introduced during the progression. In some cases, the error rates in the *illusion bias* experiment could be as high as 40 to 50%. This finding is consistent with prior reports of the danger

of using a progressive visualization [6], [11] where participants made the wrong decisions after observing unintentional patterns during the progression.

- **Interaction Improves Confidence**: In Experiments 1 and 2 where participants did not interact with the visualization, the participants' confidence in their answers correlated with the accuracy of their responses. However, in Experiments 3 and 4, we found that interacting with the progressive visualization increased the participants' confidence, even if the interactions resulted in non-optimal steering that lead to longer task-completion times.

We believe these results encourage the use of progressive visualization while accounting for potential biases, as well as lay the foundation for future study of additional biases in progressive visualization systems.

ACKNOWLEDGMENT

We would like to thank Shannon Robinson and Brian Montambault for their help in preparing this manuscript, as well as the organizers and participants of Dagstuhl Seminar 18411 "Progressive Data Analysis and Visualization" whose discussions laid the foundation for this work. This work was supported in part by NSF 1527765, 1564049, 1513651, 1452977, 1845638 and DARPA FA8750-17-2-0107.

REFERENCES

[1] B. Shneiderman, "Response time and display rate in human performance with computers," *ACM Computing Surveys (CSUR)*, vol. 16, no. 3, pp. 265–285, 1984.

[2] Z. Liu and J. Heer, "The effects of interactive latency on exploratory visual analysis," *IEEE transactions on visualization and computer graphics*, vol. 20, no. 12, pp. 2122–2131, 2014.

[3] D. Fisher, S. M. Drucker, and A. C. König, "Exploratory visualization involving incremental, approximate database queries and uncertainty," *IEEE computer graphics and applications*, vol. 32, no. 4, pp. 55–62, 2012.

[4] M. Angelini, G. Santucci, H. Schumann, and H.-J. Schulz, "A review and characterization of progressive visual analytics," in *Informatics*, vol. 5, no. 3. Multidisciplinary Digital Publishing Institute, 2018, p. 31.

[5] R. Rosenbaum and H. Schumann, "Progressive refinement: more than a means to overcome limited bandwidth," in *Visualization and Data Analysis 2009*, vol. 7243. International Society for Optics and Photonics, 2009, p. 72430I.

[6] C. Turkay, E. Kaya, S. Balcisoy, and H. Hauser, "Designing progressive and interactive analytics processes for high-dimensional data analysis," *IEEE transactions on visualization and computer graphics*, vol. 23, no. 1, pp. 131–140, 2017.

[7] C. D. Stolper, A. Perer, and D. Gotz, "Progressive visual analytics: User-driven visual exploration of in-progress analytics," *IEEE Transactions on Visualization and Computer Graphics*, vol. 20, no. 12, pp. 1653–1662, 2014.

[8] S. K. Badam, N. Elmqvist, and J.-D. Fekete, "Steering the craft: Ui elements and visualizations for supporting progressive visual analytics," in *Computer Graphics Forum*, vol. 36, no. 3. Wiley Onl. Lib, 2017, pp. 491–502.

[9] N. Pezzotti, T. Höllt, J. Van Gemert, B. P. Lelieveldt, E. Eisemann, and A. Vilanova, "Deepeyes: Progressive visual analytics for designing deep neural networks," *IEEE transactions on visualization and computer graphics*, vol. 24, no. 1, pp. 98–108, 2018.

[10] D. Fisher, I. Popov, S. Drucker *et al.*, "Trust me, i'm partially right: incremental visualization lets analysts explore large datasets faster," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2012, pp. 1673–1682.

[11] D. Moritz, D. Fisher, B. Ding, and C. Wang, "Trust, but verify: Optimistic visualizations of approximate queries for exploring big data," in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. ACM, 2017, pp. 2904–2915.

[12] M. Procopio, C. Scheidegger, E. Wu, and R. Chang, "Selective wander join: Fast progressive visualizations for data joins," in *Informatics*. Multidisciplinary Digital Publishing Institute, 2019.

[13] L. Micallef, H.-J. Schulz, M. Angelini, M. Aupetit, R. Chang, J. Kohlhammer, A. Perer, and G. Santucci, "The human user in progressive visual analytics," in *21st EG/VGTC Conference on Visualization*, 2019.

[14] E. Wall, L. Blaha, C. Paul, and A. Endert, "A formative study of interactive bias metrics in visual analytics using anchoring bias," in *IFIP Conference on Human-Computer Interaction*. Springer, 2019, pp. 555–575.

[15] E. Dimara, S. Franconeri, C. Plaisant, A. Bezerianos, and P. Dragicevic, "A task-based taxonomy of cognitive biases for information visualization," *IEEE transactions on visualization and computer graphics*, 2018.

[16] A. R. Smith and P. D. Windschitl, "Biased calculations: Numeric anchors influence answers to math equations," *Judgment and Decision Making*, vol. 6, no. 2, p. 139, 2011.

[17] E. Wall, L. M. Blaha, L. Franklin, and A. Endert, "Warning, bias may occur: A proposed approach to detecting cognitive bias in interactive visual analytics," in *2017 IEEE Conference on Visual Analytics Science and Technology (VAST)*. IEEE, 2017, pp. 104–115.

[18] J. M. Hellerstein, P. J. Haas, and H. J. Wang, "Online aggregation," in *Acm Sigmod Record*, vol. 26, no. 2. ACM, 1997, pp. 171–182.

[19] P. J. Haas and J. M. Hellerstein, "Ripple joins for online aggregation," *ACM SIGMOD Record*, vol. 28, no. 2, pp. 287–298, 1999.

[20] C. Jermaine, S. Arumugam, A. Pol, and A. Dobra, "Scalable approximate query processing with the dbo engine," *ACM Transactions on Database Systems (TODS)*, vol. 33, no. 4, p. 23, 2008.

[21] F. Li, B. Wu, K. Yi, and Z. Zhao, "Wander join: Online aggregation via random walks," in *Proceedings of the 2016 International Conference on Management of Data*. ACM, 2016, pp. 615–629.

[22] D. Laur and P. Hanrahan, "Hierarchical splatting: A progressive refinement algorithm for volume rendering," in *ACM SIGGRAPH Computer Graphics*, vol. 25, no. 4. ACM, 1991, pp. 285–288.

[23] M. F. Cohen, S. E. Chen, J. R. Wallace, and D. P. Greenberg, "A progressive refinement approach to fast radiosity image generation," in *ACM SIGGRAPH computer graphics*, vol. 22, no. 4. ACM, 1988, pp. 75–84.

[24] H. Hoppe, "View-dependent refinement of progressive meshes," in *Siggraph*, vol. 97. Citeseer, 1997, pp. 189–198.

[25] M. Glueck, A. Khan, and D. J. Wigdor, "Dive in!: Enabling progressive loading for real-time navigation of data visualizations," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2014, pp. 561–570.

[26] E. G. Hetzler, V. L. Crow, D. A. Payne, and A. E. Turner, "Turning the bucket of text into a pipe," in *IEEE Symposium on Information Visualization, 2005. INFOVIS 2005.* IEEE, 2005, pp. 89–94.

[27] R. J. Crouser, L. Franklin, and K. Cook, "Rethinking visual analytics for streaming data applications," *IEEE Internet Computing*, vol. 21, no. 4, pp. 72–76, 2017.

[28] H.-J. Schulz, M. Angelini, G. Santucci, and H. Schumann, "An enhanced visualization process model for incremental visualization," *IEEE transactions on visualization and computer graphics*, vol. 22, no. 7, pp. 1830–1842, 2016.

[29] E. Zgraggen, A. Galakatos, A. Crotty, J.-D. Fekete, and T. Kraska, "How progressive visualizations affect exploratory analysis," *IEEE transactions on visualization and computer graphics*, vol. 23, no. 8, pp. 1977–1987, 2017.

[30] T. Mühlbacher, H. Piringer, S. Gratzl, M. Sedlmair, and M. Streit, "Opening the black box: Strategies for increased user involvement in existing algorithm implementations," *IEEE transactions on visualization and computer graphics*, vol. 20, no. 12, pp. 1643–1652, 2014.

[31] B. Ding, S. Huang, S. Chaudhuri, K. Chakrabarti, and C. Wang, "Sample+ seek: Approximating aggregates with distribution precision guarantee," in *Proceedings of the 2016 International Conference on Management of Data*. ACM, 2016, pp. 679–694.

[32] S. Agarwal, B. Mozafari, A. Panda, H. Milner, S. Madden, and I. Stoica, "Blinkdb: queries with bounded errors and bounded response times on very large data," in *Proceedings of the 8th ACM European Conference on Computer Systems*. ACM, 2013, pp. 29–42.

[33] A. Kim, E. Blais, A. Parameswaran, P. Indyk, S. Madden, and R. Rubinfeld, "Rapid sampling for visualizations with ordering guarantees," *Proceedings of the VLDB Endowment*, vol. 8, no. 5, pp. 521–532, 2015.

[34] S. Rahman, M. Aliakbarpour, H. K. Kong, E. Blais, K. Karahalios, A. Parameswaran, and R. Rubinfield, "I've seen enough: incrementally improving visualizations to support rapid decision making," *Proceedings of the VLDB Endowment*, vol. 10, no. 11, pp. 1262–1273, 2017.

[35] G. Cormode, M. Garofalakis, P. J. Haas, C. Jermaine *et al.*, "Synopses for massive data: Samples, histograms, wavelets, sketches," *Foundations and Trends® in Databases*, vol. 4, no. 1–3, pp. 1–294, 2011.

[36] M. Correll and M. Gleicher, “Error bars considered harmful: Exploring alternate encodings for mean and error,” *IEEE transactions on visualization and computer graphics*, vol. 20, no. 12, pp. 2142–2151, 2014.

[37] N. Ferreira, D. Fisher, and A. C. Konig, “Sample-oriented task-driven visualizations: allowing users to make better, more confident decisions,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2014, pp. 571–580.

[38] M. Fernandes, L. Walls, S. Munson, J. Hullman, and M. Kay, “Uncertainty displays using quantile dotplots or cdfs improve transit decision-making,” in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. ACM, 2018, p. 144.

[39] T. Gschwandtner, M. Bögl, P. Federico, and S. Miksch, “Visual encodings of temporal uncertainty: A comparative user study,” *IEEE transactions on visualization and computer graphics*, vol. 22, no. 1, pp. 539–548, 2016.

[40] J. Hullman, “Why authors don’t visualize uncertainty,” *IEEE transactions on visualization and computer graphics*, vol. 26, no. 1, pp. 130–139, 2019.

[41] M. A. Bedek, A. Nussbaumer, L. Huszar, and D. Albert, “Methods for discovering cognitive biases in a visual analytics environment,” in *Cognitive Biases in Visualizations*. Springer, 2018, pp. 61–73.

[42] R. Wesslen, S. Santhanam, A. Karduni, I. Cho, S. Shaikh, and W. Dou, “Anchored in a data storm: How anchoring bias can affect user strategy, confidence, and decisions in visual analytics,” *arXiv preprint arXiv:1806.02720*, 2018.

[43] C. Xiong, L. van Weelden, and S. Franconeri, “The curse of knowledge in visual data communication,” *IEEE Transactions on Visualization and Computer Graphics*, 2019.

[44] R. Wesslen, S. Santhanam, A. Karduni, I. Cho, S. Shaikh, and W. Dou, “Investigating effects of visual anchors on decision-making about misinformation,” in *Computer Graphics Forum*, vol. 38, no. 3. Wiley Online Library, 2019, pp. 161–171.

[45] D. R. Johnson and L. A. Borden, “Participants at your fingertips: Using amazons mechanical turk to increase student–faculty collaborative research,” *Teaching of Psychology*, vol. 39, no. 4, pp. 245–251, 2012.

[46] B. McInnis, D. Cosley, C. Nam, and G. Leshed, “Taking a hit: Designing around rejection, mistrust, risk, and workers’ experiences in amazon mechanical turk,” in *Proceedings of the 2016 CHI conference on human factors in computing systems*, 2016, pp. 2271–2282.

[47] H. Wickham, “Asa 2009 data expo,” *Journal of Computational and Graphical Statistics*, pp. 281–283, 2011.

[48] R. Amar, J. Eagan, and J. Stasko, “Low-level components of analytic activity in information visualization,” in *IEEE Symposium on Information Visualization, 2005. INFOVIS 2005.* IEEE, 2005, pp. 111–117.

[49] J. Hullman, P. Resnick, and E. Adar, “Hypothetical outcome plots outperform error bars and violin plots for inferences about reliability of variable ordering,” *PloS one*, vol. 10, no. 11, p. e0142444, 2015.

[50] A. Kale, F. Nguyen, M. Kay, and J. Hullman, “Hypothetical outcome plots help untrained observers judge trends in ambiguous data,” *IEEE transactions on visualization and computer graphics*, vol. 25, no. 1, pp. 892–902, 2018.

[51] A. C. Valdez, M. Ziefle, and M. Sedlmair, “Priming and anchoring effects in visualization,” *IEEE transactions on visualization and computer graphics*, vol. 24, no. 1, pp. 584–594, 2017.

[52] I. Cho, R. Wesslen, A. Karduni, S. Santhanam, S. Shaikh, and W. Dou, “The anchoring effect in decision-making with visual analytics,” in *2017 IEEE Conference on Visual Analytics Science and Technology (VAST)*. IEEE, 2017, pp. 116–126.

[53] E. Wall, L. M. Blaha, C. L. Paul, K. Cook, and A. Endert, “Four perspectives on human bias in visual analytics,” in *Cognitive biases in visualizations*. Springer, 2018, pp. 29–42.

Marianne Procopio received her BA and MA degrees in Computer Science from Boston University. She is currently working towards her PhD at Tufts University. Her research spans the fields of data visualization and database systems by examining the different strategies that integrate data storage and retrieval techniques with front-end visualizations to enable interactive exploratory analysis of large scale data.

Ab Mosca received their MS in Computer Science from Tufts University and their BA in Mathematics from Smith College. They are currently working towards their PhD at Tufts University. Ab’s research spans the fields of information visualization and human computer interaction with a focus on leveraging visualization and human-centered design to lower the barrier between decision makers and advanced analytics.

Carlos Scheidegger received his PhD from the University of Utah, where he worked on software infrastructure for scientific collaboration. He is an assistant professor in the Department of Computer Science, University of Arizona. His research interests include large-scale data analysis, information visualization, and more broadly, what happens when people meet data.

Eugene Wu received his Ph.D. from MIT. He is an assistant professor at Columbia University. His research interests span the areas of core database optimization, stream processing systems, crowd sourcing, data visualization, data cleaning, and HCI.

Remco Chang received his PhD in computer science from the University of North Carolina Charlotte. He is an associate professor in computer science with Tufts University. His research interests include visual analytics, information visualization, human computer interaction, and databases.