

Revealing Perceptual Proxies with Adversarial Examples

Brian D. Ondov, Fumeng Yang, Matthew Kay, Niklas Elmqvist, *Senior Member, IEEE*, and Steven Franconeri

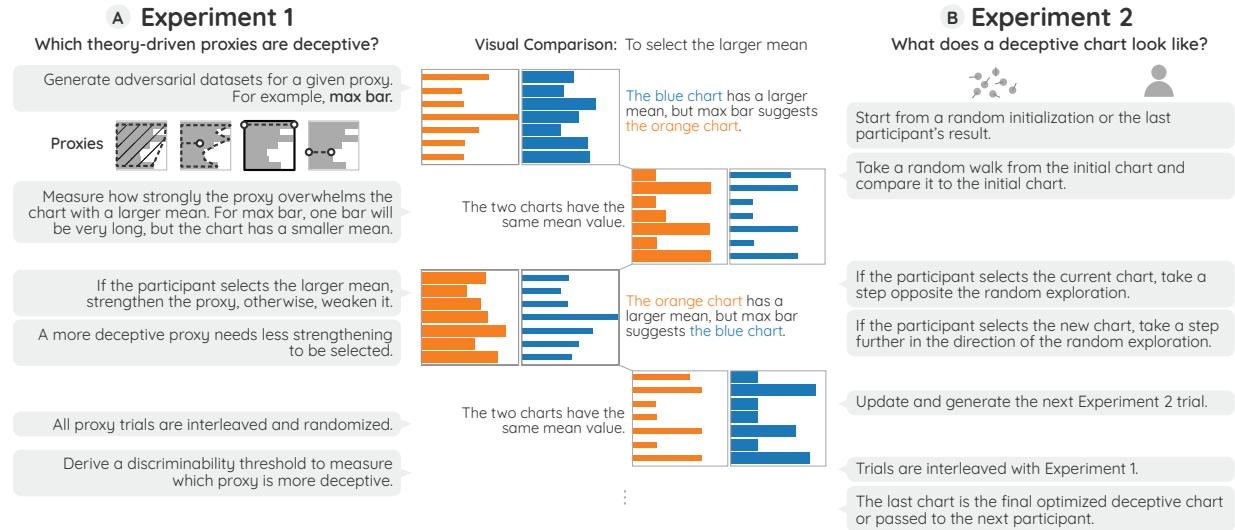


Fig. 1. We propose two approaches for uncovering how the visual system extracts statistics from a visualization, by pitting correct answers against adversarial models of candidate perceptual proxies. **A** In the “theory-driven” approach, we optimize charts to manipulate conjectured perceptual proxies, and test how powerfully they alter judgments. **B** In the “data-driven” approach, we seek to discover deceptive charts *de novo*, using human judgments as an objective function. The examples above present four real trials from the combined experiment (all annotation on bar charts are for illustrating purposes only).

Abstract—Data visualizations convert numbers into visual marks so that our visual system can extract data from an image instead of raw numbers. Clearly, the visual system does not compute these values as a computer would, as an arithmetic mean or a correlation. Instead, it extracts these patterns using *perceptual proxies*; heuristic shortcuts of the visual marks, such as a center of mass or a shape envelope. Understanding which proxies people use would lead to more effective visualizations. We present the results of a series of crowdsourced experiments that measure how powerfully a set of candidate proxies can explain human performance when comparing the mean and range of pairs of data series presented as bar charts. We generated datasets where the correct answer—the series with the larger arithmetic mean or range—was pitted against an “adversarial” series that should be seen as larger if the viewer uses a particular candidate proxy. We used both Bayesian logistic regression models and a robust Bayesian mixed-effects linear model to measure how strongly each adversarial proxy could drive viewers to answer incorrectly and whether different individuals may use different proxies. Finally, we attempt to construct adversarial datasets from scratch, using an iterative crowdsourcing procedure to perform black-box optimization.

Index Terms—Perceptual proxies, vision science, crowdsourced evaluation.

1 INTRODUCTION

Creating efficient data visualizations requires understanding how the human visual system processes data displays. A significant amount of

- Brian Ondov is with the National Institutes of Health in Bethesda, MD, USA and University of Maryland in College Park, MD, USA. E-mail: ondovb@umd.edu.
- Fumeng Yang is with Brown University in Providence, RI, USA. E-mail: fy@brown.edu.
- Matthew Kay is with University of Michigan in Ann Arbor, MI, USA. E-mail: mkay@umich.edu.
- Niklas Elmqvist is with University of Maryland in College Park, MD, USA. E-mail: elm@umd.edu.
- Steven Franconeri is with Northwestern University in Evanston, IL, USA. E-mail: franconeri@northwestern.edu.

Manuscript received xx xxx. 201x; accepted xx xxx. 201x. Date of Publication xx xxx. 201x; date of current version xx xxx. 201x. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org. Digital Object Identifier: [xx.xxxx/TVCG.201x.xxxxxxx](https://doi.org/10.1109/TVCG.201x.1xxxxxx)

work at the intersection of vision science and data visualization has been conducted over the years to determine the capabilities and limits of the visual system, as well as to create visualizations that leverage these findings. One way to think about human vision is that it is an *information processing system* capable of extracting vital information about the world from images, but also internally representing this information so that it can be efficiently used for decisions and action [30]. But if the visual system is a computational system, what are its programs?

The concept of *perceptual proxies* [19, 48, 50] has recently been proposed as a potential answer to this question. A perceptual proxy is a heuristic shortcut for how the visual system extracts data from images using simple features, such as a shape’s outline, center of mass, area, or color. The hypothesis is that, instead of computing statistics per se, the visual system relies on proxy computations across visual marks, when seeing trends in a line chart, finding maxima in a bar chart, or analyzing a distribution in a pie chart. However, while recent work has begun to uncover the perceptual proxies used for specific tasks, such as determining correlations in scatterplots [48] or making comparisons in bar charts [19], these studies are still only preliminary. In particular,

we have an insufficient understanding of how to measure which proxies are used, and how that use might change across tasks and individuals.

In this paper, we address this gap in the literature by evaluating the choice of perceptual proxy for two visual comparison tasks—highest mean and highest range—between two different bar charts.

We approach the problem in two complementary ways. In the first experiment, the datasets displayed on the two charts were carefully optimized using simulated annealing [25] so that the correct dataset (the one with the highest arithmetic mean or range) was juxtaposed against an “adversarial” dataset that would be perceived to be the correct answer if the viewer favored that particular perceptual proxy. For example, for a perceptual proxy of “longest bar”, our data generation algorithm would attempt to foil the correct perception by making the adversarial dataset (i.e., the wrong choice) to appear to have a higher mean by artificially elongating one of its bars more than the correct dataset while still having a smaller arithmetic mean or range. We then used a staircase design and Bayesian estimation to find the baseline at which each of four proxies (and one control) would consistently cause participants to choose the wrong answer. In other words, this procedure will progressively increase the difference between the larger mean or range for the correct dataset and the smaller adversarial dataset until the participant no longer is consistently fooled by that perceptual proxy.

If our first experiment was *theory-driven*—based on perceptual proxies, to be exact—then our follow-up experiment was instead *data-driven* with few preconceived assumptions. Instead of carefully tuning datasets, we randomly generated or perturbed datasets—all with the same statistical properties—and then showed two randomly selected datasets in a lineup. Participants were asked which of each pair they perceived had the largest range or mean, respectively. Using a black-box optimization method, where participant choices became the signal for which of two series were perceived to have the larger mean or range, we were able to “evolve” random data into series that should be increasingly deceptive for the task. In a post hoc analysis, we found that several perceptual proxies from the initial experiment are represented as motifs in the optimized datasets. However, a followup analysis was not able to confirm that these datasets are consistently chosen over random data.

We crowdsourced the data collection for these two experiments on Amazon Mechanical Turk with 65 participants for each. In both experiments, we investigated two primitive visual tasks: *MaxMean* (which chart has the larger mean?) and *MaxRange* (which chart has the larger range?). We first used Bayesian logistic regression models to derive participants’ discriminability thresholds for selecting the correct answer (and the measurement error associated with those thresholds). We then applied a robust Bayesian mixed-effects linear model to the thresholds and measurement errors to estimate the strength of each proxy and how individuals might use the proxies differently. We found that the most influential perceptual proxy for the *MaxMean* task was *centroid*, whereas it was *slope* for the *MaxRange* task; we also found evidence that individuals vary their usage of proxies and may select *against* a proxy. Our contributions are:

- Using an adversarial experimental design to sway performance, we give evidence of specific proxies for extracting mean and range; and
- Using black-box optimization, we create datasets that provide insights into previously theorized proxies or suggest new ones.

Our findings represent progress towards an operationalized model for how not just individuals, but also populations perform two specific visual comparison tasks using bar charts. Some of these ideas may generalize to additional tasks and visual representations. To facilitate these goals, we provide our experiment code, data collected for both experiments, and analysis scripts at <https://osf.io/2re7b/>.

2 BACKGROUND

Perceptual psychology has the potential to help uncover visual representations that can optimize pattern extraction requirements in data visualizations. Here we review relevant topics in both fields to provide the necessary background for our work.

2.1 Visual Comparison

Visual comparison of data across multiple visual representations is a common but complex task in visualization [1, 15, 49]; Amar et al. [1] call it a “compound” task in that it consists of multiple low-level tasks. Perhaps because of this complexity and the composite nature of the comparison task, it is only recently that visual comparison has been studied empirically in visualization research.

We base much of our work on the dual set of findings by Ondov et al. [33] as well as Jardine et al. [19], which both treat visual comparison as a relatively low-level perceptual task. Ondov et al. conducted a number of large-scale crowdsourcing experiments to measure the perceptual precision for comparing charts under different spatial arrangements. While the present tasks are different, we draw our experimental methodology of using a measurement called “tier” on this previous work, and use a similar simulated annealing [25] algorithm to generate our adversarial datasets.

In contrast, the follow-up paper by Jardine et al. [19] focuses on precisely the *MaxMean* and *MaxRange* tasks that we investigate here. However, while Jardine et al. continued to focus on comparing relative performance across spatial arrangements, our focus in the present paper is on modeling the perceptual proxies of visual comparison.

2.2 Perceptual Proxies

A relatively new concept in vision science [50], a *perceptual proxy* is a visual shortcut based on a spatial feature of a visualization that could conceivably explain how the human perceptual system interprets a scene and extracts data from it. Such proxies are a particularly useful reasoning tool for data visualizations, because understanding an individual’s—and a population’s—preferred proxies may suggest practical guidelines for how to optimize a visual representation to match these proxies. This, in turn, would enable us to minimize the perceptual error arising from a specific visualization. Furthermore, proxies can also easily be operationalized as small programs (or “bots”) that model that proxy, which would allow us to estimate how effectively a given visualization should show a given pattern to a viewer. Note that these proxies may be dependent on the visualization type, design, data arrangement, and the traits and task of the viewer, and therefore may need to be empirically evaluated on a case-by-case basis.

Perceptual proxies arise out of seminal findings on “elementary perceptual processes,” which were originally derived from a long history of empirical experiments in perceptual psychology [27, 40] and later summarized by Cleveland and McGill [7]. However, while these low-level processes can easily be applied to individual marks or groups of marks in a visualization, more composite tasks involving multiple values or general trends are more challenging to extract [6, 36]. In such situations, the visual system likely constructs proxies from these perceptual building blocks in order to support quick visual judgments.

Rensink and Baldridge explored the perceptual of correlation in scatterplots [37, 38], showing that just-noticeable-differences (JNDs) of correlations can be modeled Weber’s law [17]. Harrison et al. [16]—modified by Kay and Heer [23]—later extended this analysis to eight additional visual representations. Finally, Yang et al. [48] identified 49 proxies—which they called *visual features*—and empirically derive the small subset of features that people actually use when discriminating correlation in scatterplots. Obviously, our work in this paper directly builds on these ideas, but applies them to visual comparison in bar charts and different summary statistics.

The aforementioned work by Jardine et al. [19] was, to our knowledge, the first to study summary statistics in pairs of bar charts. However, their work was a *post hoc* analysis of existing data that were not been designed to be optimized, let alone adversarial. As a consequence, many of the proxies were highly correlated, both with each other and the true answer (see Appendix B). This made it difficult to distinguish proxy effects. Our work here differs in that we specifically create datasets in which the values of proxies diverge from the true summary statistic, with the goal of illuminating whether people use these proxies.

Also directly related to our work is the bar chart comparison study presented by Yuan et al. [50], where they asked participants to estimate averages in multi-value lineups of two side-by-side bar charts. Unlike

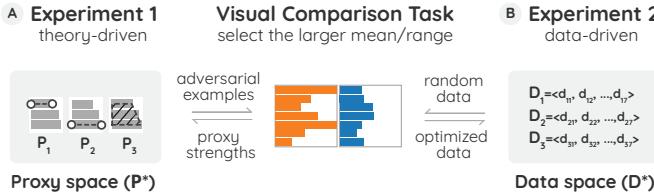


Fig. 2. **Overview of our methodological approach.** We took both theory-driven (in proxy space) and data-driven (in data space) approaches to investigate the effects of perceptual proxies.

our work where we keep the number of bars consistent for both charts, Yuan’s study varied the number of bars in the two charts. This enabled them to show that the summed area of the bars (which we call “amount of ink” in Section 6.2) is a likely perceptual proxy for the relative average value between two bar graphs. Thus, while our work shares methodological similarities, the purpose of this novel study is different.

2.3 Adversarial Visualization and Data

Central to our work is the idea to generate adversarial (or mostly adversarial; see below) tasks to derive datasets, visual representations, or visual appearances that can deceive the user’s perception. One first example of such an approach in data visualization was the work by Wickham et al. [46] on graphical inference in visualization. They propose both a “Rorschach” protocol, where participants are shown essentially random data in a visualization and asked to generate insights, as well as a lineup protocol, where multiple visualizations are shown of different datasets and the task is to identify the one dataset drawn from real data.

Pandey et al. [34] studied deception in visualization by asking participants in a crowdsourced study to interpret data presented using four different distortion techniques. For each distortion type, a deceptive version, which used the technique, and a control, which did not, was used. The dataset generation in the paper is idiosyncratic and done by hand. In contrast, our adversarial dataset generation is fully automated.

The notion of “adversarial” (or “black hat”) visualizations was first proposed by Correll and Heer [9], and used the language of computer security to survey the practice of “attacks” on data visualization. Their work is largely conceptual, and only one component of their model—data manipulation—is directly relevant to our study, but the overall tenor of these ideas are consistent with our methodology.

Correll et al. [10] created crowdsourced lineups where participants saw multiple visualizations of largely “innocent” datasets with one “flawed.” They generate these datasets using an iterative process based on three common data quality errors—spikes, gaps, and outliers—and at varying levels of data quality. In comparison, our approach optimizes both datasets in a lineup to exacerbate potentially deceptive proxies (Experiment 1), or uses a random walk optimization to identify deceptive datasets without the need for prescribing specific data quality errors (Experiment 2).

3 OVERVIEW

In this paper, we study perceptual proxies for comparing data series visualized using bar charts from two different directions (Fig. 2):

- **Theory-driven**, where we draw on the literature in vision science and visualization on perceptual proxies to generate “adversarial” datasets that optimize individual proxies to deceive a participant into selecting an incorrect choice (Experiment 1); and
- **Data-driven**, where we simply start from a set of randomly generated data series—with no preconceived notion of how they should be generated—and let participant choice for successive lineups between series guide a black box optimization to find increasingly more deceptive data (Experiment 2).

Common between both of these experiments is that they are based on the assumption that people use perceptual proxies as a short-hand when completing a comparison task. Similar to many classic experiments in perceptual psychology, such as JNDs [14], we employ “adversarial”

trials designed to elicit situations where specific perceptual phenomena compete. The difference between the experiment lies in how they generate the adversarial data series: while both are optimization methods, the former (theory-driven) is done off-line prior to running the experiment and relies on our belief that we understand the perceptual proxies at play, whereas the other one (data-driven) is done on-line in response to participant performance in real-time (making it a form of *human computation* [11, 28, 35]) and lets the driving phenomena behind perception of comparison for this task emerge from the data itself.

4 GENERATING ADVERSARIAL DATASETS

While our two experiments may on the surface appear to be very different—one where we carefully design datasets to optimize certain perceptual proxies (Experiment 1), the other where we perform a semi-random walk through the space of all possible datasets (Experiment 2)—the actual dataset generation for both experiments is actually quite similar. For both experiments, the goal is to perturb existing data to generate adversarial datasets that can deceive the viewer effectively. The difference is in how we determine whether a new dataset is better or worse than an existing one; we either attempt to (1) model human perception through the notion of perceptual proxies, or we simply (2) ask the participants, respectively.

We used two different execution modes. Optimizing datasets based on theorized perceptual proxies requires no participant intervention and can thus be done off-line prior to an evaluation. On the other hand, the optimization approach requires on-line execution, because participants’ preferences determine which dataset should be accepted and perturbed further, and which should be discarded.

4.1 Off-line Optimization with Simulated Annealing

The objective of our off-line optimization is to generate pairs of datasets where the adversarial series have specific differences in the proxy that we are measuring (e.g., longest bar), without affecting the summary statistic, which correlates with many proxies in random data. Meanwhile, the adversarial dataset should also have a lower mean or range (depending on the task); in other words, it should be the incorrect response. *How wrong it is*, e.g., how much smaller its mean or range, serves as the difficulty level of the task for Experiment 1.

We generate such datasets using Simulated Annealing [25], drawing inspiration from Matejka and Fitzmaurice [31]. Our objective is a pair of datasets with specified ratios for a proxy and a summary statistic. Deviation from this objective is formalized in a cost function as the sum of squared differences between the ratios in the objective and those of the dataset being considered.

4.2 On-line Optimization w/ Stochastic Gradient Descent

Rather than attempting to model the goodness of a specific dataset from the perspective of perceptual proxies, the alternative method we explore here is to simply ask people. More specifically, we consider the human perception of the summary statistic to be a black-box function [39] that we are seeking to optimize. Since we do not have access to a derivative of this function, we implement Dueling Bandit Gradient Descent [51], which stochastically estimates the gradient descent process using only pairwise rankings. In Experiment 2, we perform this search in an on-line manner, that is, in real-time as the experiment is proceeding. Our version of this algorithm, and its associated helper functions, is described in detail in Appendix A. We optimize in data, or “bar,” space, meaning we have 7 dimensions representing the lengths of each bar, in order from top to bottom. In our experiments, this procedure is both serialized, by performing multiple epochs (corresponding to unique participants), and parallelized, by starting from different random initializations (see Section 7.2).

5 METHOD

In this section, we discuss the aspects of both experiments that are common, including the two tasks (select the larger mean/range), visual representation, participants, apparatus, and procedure. Both experiments were run simultaneously, and with the same participants.

Table 1. **Perceptual proxies in Experiment 1.** All the example chart pairs have the same underlying datasets, and the blue chart on the right side has a **larger mean/range** (the correct answer). In real trials, the position of the correct answer is randomized and balanced. In *MaxMean*, charts randomly have skinny bars to decouple amount of ink from the mean (see Section 6.2). Slope proxies are based on the horizontal slope (the orientation of the dependent variable in data).

A <i>MaxMean</i>		B <i>MaxRange</i>	
Proxy	Description	Proxy	Description
<i>hull area</i>			The area of a convex hull around the bars, ignoring difference in bar thickness.
<i>centroid</i>			The centroid of the area occupied by the bars along just relevant x-axis
<i>max bar</i>			The length of the longest bar
<i>min bar</i>			The length of the shortest bar

5.1 Visual Representation

We used a simple horizontal bar chart where each bar had a uniform color and thickness (Fig. 1). The visual stimulus involved showing these bar charts in a lineup consisting of two charts arranged side by side. We used two diverging colors—orange (#ff7f0e) and blue (#1f77b4), respectively—for the two charts.

5.2 Tasks

Drawing from past work [19], we included two tasks on a per-chart level (i.e., they entailed choosing between two separate charts rather than choosing between individual data items in each chart):

- *MaxMean*: Determine the chart that has the larger mean value across all of its components.
- *MaxRange*: Determine the chart that has the larger range from its shortest to its longest components.

These tasks are global in scope, in that they require the participant to survey the entire visualization rather than individual items. They are motivated by the low-level analytic task taxonomy of Amar et al. [2], which describes both as building blocks for deeper tasks. As examples, they cite the mean being used to compare relative efficiencies of two categories of cars, or ranges being used to assess whether a data series could merit further analysis.

5.3 Procedure

After consenting, participants were shown a sequence of instructional screens followed by a set of practice trials. Practice trials gave feedback on whether or not the participant’s answer was correct; this was not the case for the timed trials. Participants were required to score three correct answers in a row to proceed past the practice phase. The purpose was to ensure that participants had correctly understood the task at hand.

Each individual trial started with a short countdown. Then the platform showed the lineup of two data series visualized as bar charts in a side-by-side arrangement (horizontal juxtaposition) as “impressions” for a short time period. Based on extensive piloting, we chose 1000ms impressions for *MaxMean* and 1500ms for *MaxRange*. After the impression time ended, the lineup was replaced by two colored (orange and blue) buttons to represent the bar charts had been shown. Answering the trial meant clicking on the button representing the bar chart that the participant had perceived as having the larger mean or range. Participants assigned to each task typically spent between 8 and 27 minutes to complete all the sessions ($\mu = 15.24$, $\sigma = 4.75$).

5.4 Participants

For each of the two tasks, we recruited 65 participants for the combined study from Amazon Mechanical Turk (MTurk). The *MaxMean* task had 22 female, 42 male, and 1 unspecified, and the *MaxRange* task

had 31 female and 34 male. We limited participation to the United States due to tax and compensation restrictions imposed by our IRB. We screened participants to ensure at least a working knowledge of English; this was required to follow the instructions in our testing platform. Participants could only perform the experiment once. All participants were compensated at a rate consistent with an hourly wage of at least \$10/hour (the U.S. federal minimum wage in 2020 is \$7.25).

5.5 Apparatus

All experiments were distributed through the participant’s web browser. Because of our crowdsourced setting, we were unable to control the specific computer equipment that the participants used. We required a screen resolution of at least 1280×800 pixels. During the experiment, we placed the participant’s device in full screen mode to maximize the visibility. The testing software was implemented in JavaScript and D3.js [4] with a server-side Perl and CGI backend.

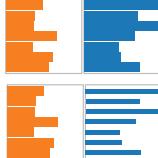
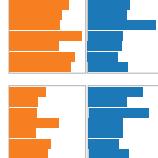
6 EXPERIMENT 1: TESTING SPECIFIC PROXIES

Experiment 1 follows a theory-driven approach: we start with a set of plausible perceptual proxies, generate datasets optimizing for them, and then test these datasets in human judgments. This experiment has two goals: first, to find evidence that participants could be using perceptual proxies in visual comparison tasks; second, to understand how participants used different proxies differently.

6.1 Selecting Proxies

We aimed to identify a set of proxies in the perceptual space that are likely used by participants and could be manifested by us to generate adversarial trials. We used the following heuristics and followed an iterative process. We primarily considered the proxies that best aligned with participants’ judgments in the studies by Jardine et al. [19]. We then eliminated proxies that were highly correlated with already-selected ones; this allows us to modulate each independently in the experiment. Last, we also considered a new proxy if it satisfies the above two constraints. As a result, we selected four proxies for the *MaxMean* task: *hull area*, *centroid*, *max bar*, and *min bar*; we also selected four other proxies for the *MaxRange* task: *hull area norm*, *slope*, *slope range*, and *slope neighbor*. Because of the high degree of correlation of many proposed proxies, the ones we have chosen can be thought of as representatives of broader classes. For example, the area of a chart’s convex hull is highly correlated with the horizontal position of that hull’s centroid. Evidence for any proxy we have chosen thus would thus imply that either that proxy or a similar proxy is at play. For each selected proxy, we show the description and an example in Table 1 and the correlation observed between these proxies in Jardine et al.’s data in Appendix B.

Table 2. The confounding proxies in the *MaxMean* and *MaxRange* tasks.

A <i>MaxMean</i>		B <i>MaxRange</i>	
Confounding Proxy	Description	Confounding Proxy	Description
<i>ink area</i>	 <p>If the bar thickness is the same, the bar chart with a larger mean will always have more ink. In this example, the blue chart has a larger mean and more ink. We therefore vary the thickness. This example shows that the blue chart with skinnier bars could have less or the same amount of ink.</p>	<i>min bar and max bar</i>	 <p>In the orange chart, if we make max bar longer to be deceptive (a smaller range), min bar has to be longer, too. The blue chart will always have a shorter min bar. We span the range of the deceptive chart across min bar or max bar of the other chart and balance all the cases. In this example, the blue chart could have a longer min bar or a shorter max bar.</p>

6.2 Eliminating Confounding Proxies

Besides the selected proxies, both tasks had other proxies directly related to the summary statistic itself. They could always indicate a correct answer (i.e., the larger mean or the larger range), and thus we attempted to eliminate their impact in our experiment.

For the *MaxMean* task, an *ink area* proxy—the total “amount of ink” [42] (i.e., the number of colored pixels on the screen)—could be used by participants to estimate mean when the number of bars is different [50]. If all the bars are of the same thickness, the *ink area* proxy reduces to the sum, and thus the arithmetic mean (see Table. 2a). We decoupled the *ink area* proxy from the mean by randomly choosing one of the two charts to have skinnier bars than the other. We choose a fixed skinniness such that the skinny-bar chart will always have the least amount ink, even for a large difference in mean. The *ink area* value thus cannot be used to determine the correct answer (see Table. 2a).

Similarly, for the *MaxRange* task, *min bar* and *max bar* are closely related to the range (see Table. 2b). Therefore, the other chart will always have a shorter *min bar*. The feedback from the pilot studies also supported this speculation, as some participants reported choosing the chart with the shortest bar as their strategy. We therefore manipulated the range values such that the smaller range spans either the minimum or maximum of the larger range. In this way, *min bar* or *max bar* only corresponds to the larger range 50% of the time and therefore is no longer correlated with the correct answer.

For each of these confounding proxies, we randomized and balanced the four cases: if the proxy is deceiving or not and if the correct response is on left or right.

6.3 Hypotheses

With our goal of understanding proxies and participants’ usage of proxies, we framed two research hypotheses for Experiment 1:

H_1 Adversarially manipulating perceptual proxies will mislead participants to be *worse* at making a visual comparison.

H_2 Individuals will be affected by such manipulations differently.

6.4 Experimental Design

For our hypotheses, we performed within-subjects factorization for the two tasks and the corresponding four proxies. We recruited different participants for each task due to concerns about practice [12], fatigue [44], and carryover effects. Each participant finished all four proxy conditions and a control condition where no specific proxy was manipulated. We designed this control condition to replicate the results from Jardine et al. [19] and also to provide a baseline for comparison. Each condition consisted of twenty trials. In each trial, we collected the participant’s response, the proxy manipulated, the two datasets presented, and the experiment parameters. The remaining details of experimental materials, framework, recruitment, procedure, and data collection were described above in Sections 4.1 and 5.

6.5 Measurement

To manifest specific proxies and quantify their effects on the *MaxMean* and *MaxRange* tasks, we followed the methodology of Jardine et al. [19] and Ondov et al. [33]. The core component in their methodology was to use a measurement called *titer* and a *staircase* method [8] called *titration* to adjust the *titer* value and to efficiently present stimuli [20].

Following this, we define the *titer* value for a pair of bar charts (left and right) as follows:

$$titer = \frac{\max(S_{\text{left}}, S_{\text{right}})}{\min(S_{\text{left}}, S_{\text{right}})} - 1, \quad S \in \{f_{\text{mean}}, f_{\text{range}}\} \quad (1)$$

where S is a summary statistic for the dataset, and it could be arithmetic mean (f_{mean}) or range (f_{range}). The *titer* value normalizes the difference of a summary statistic for the two side-by-side bar charts and scales task difficulty in different trials. For example, if a *titer* value is 0.1 in a *MaxMean* trial, one of the bar charts has a mean value 10% larger than the other one in homogeneous coordinates. In practice, a *titer* value of 0.5 is considered very large for participants to tell the larger mean or range (see Appendix C).

If participants need a large *titer* to correctly discriminate the summary statistic between the two bar charts (e.g., they need more differences in mean to select the larger mean), they are more likely to be deceived by the adversarial examples towards an incorrect answer, and therefore they likely used those proxies. Alternatively, if participants successfully select the correct answer even with a small *titer*, they may not be deceived by our manipulation of proxies.

We seek a *titer threshold* to summarize all the trials in an experimental condition and to describe participants’ performance for that condition. The *titer threshold* describes when participants could just discriminate the difference ratio of a summary statistic. This threshold concept is similar to the concept of *discrimination threshold*, like a *just noticeable difference* (JND) in psychophysics [14], but we use different ratio rather than absolute difference to normalize the stimuli. To measure a *titer threshold*, we started with titters of 0.25 and 0.40 for the *MaxMean* and *MaxRange* tasks, respectively, and approached the threshold using a staircase method [8]. The staircase method increased the *titer* value for an erroneous response (making the next trial easier) and decreased for a correct one (making the next trial more difficult) with two stages: in the first four trials, the increment and the decrement were both 0.03 for *MaxMean* and 0.06 for *MaxRange*; in the rest of the trials, the decrement was 0.01 for *MaxMean* and 0.02 for *MaxRange*. These mechanisms ensure that we efficiently present stimuli to participants and conceptually align with measuring 75% JND; that is, the minimum difference could be reliably discriminated 75% of the time [14, 38].

6.6 Prerequisites for Analysis

Data We planned to include all the participants and analyze all their data. We made only one exception where we excluded one participant from the *MaxRange* task due to an assignment error. As such, for the *MaxMean* task, we based our analysis upon 6,500 trials = 20 trials per condition \times (4 + 1) conditions \times 65 participants; and for the *MaxRange* task, we based our analysis on 6,400 trials = 20 trials per condition \times (4 + 1) conditions \times 64 participants.

Replication Our two control conditions were similar to the “adjacent” conditions in Jardine et al. [19], and the number of participants was also similar (65 cf. 50). To compare our results with theirs, we followed the same analysis method to calculate the average of *titer* values in the last ten trials and 95% confidence intervals (CIs) from a Student’s *t*-distribution. As a result, we had 0.19 [0.17, 0.21] for *MaxMean* and 0.46 [0.41, 0.51] for *MaxRange*, compared to 0.21 [0.19, 0.24] and 0.32 [0.30, 0.33] from Jardine et al. While our *MaxMean* results are similar to Jardine et al.’s, our *MaxRange* task appeared to be more difficult. This may be because we mixed the control condition with other

adversarial trials and trials from Experiment 2.

Bayesian inference For our own analyses, we followed a Bayesian estimation approach [24, 43]. We used weakly informative priors to incorporate constraints of the experimental design and to roughly capture theoretically possible values within two standard deviations. We used the R packages `brms` [5], `ggdist` [21], `tidybayes` [22], `rstan` [41], and `tidyverse` [45] for computing and presenting the results.

6.7 Analysis

Our analysis had two steps. First, we used separate Bayesian logistic regressions directly on participants' responses to estimate each participant's titer threshold for each proxy. From these models, we also derived the measurement error of participants' thresholds. Second, we used the titer thresholds and measurement errors in a robust Bayesian mixed-effects linear regression to estimate the effects of each proxy on participants' perception.

This two-step analysis protocol aligns with a common approach to aggregating repeated trials when analysing JNDs (e.g., [19, 38]), but also incorporates measurement error from the first models into the second to reduce variance. From the results of the second model, we compare different perceptual proxies (\mathcal{H}_1) and infer their various effects on different individuals (\mathcal{H}_2).

6.7.1 Step 1: Deriving Thresholds and Measurement Error

We illustrate how we derive titer thresholds and the associated measurement error in Fig. 3.

Logistic regression For each proxy \times participant, we built a Bayesian logistic regression model for that participant's 20 dichotomous responses on that proxy (1 if the participant correctly selected the chart with the larger mean/range, 0 otherwise) (Fig. 3a). The resulting logistic curves describe the relationship between titer values and the probability a participant makes a correct response (between 1 and 0). We used the inverse logistic function (logit) to calculate the corresponding titer value at which a participant has a 75% chance of getting the correct response; this value is the *tier threshold*. Similar approaches are common in psychophysics to calculate JNDs [14], and have recently been used in visualization [20, 48].

Measurement error Because we use two steps to our modeling (logistic regression to find titer thresholds followed by a linear model of thresholds), there is *measurement error* [3] associated with the titer thresholds that should be propagated from the first models to the second: the titer thresholds are uncertain, as they are estimated from data. In a Bayesian context, we can propagate this measurement error by replacing the point estimates of titer thresholds with probability distributions [32]. From the posterior distribution of each logistic regression model, we use robust estimates of location and scale—median and median absolute deviation (MAD) [29]—to derive a titer threshold (μ_{ij}) and the associated measurement error (σ_{ij}) for each participant $i \times$ proxy j (Fig. 3b). Then, in the linear regression (described below), instead of a response variable consisting only of point estimates (i.e., just the estimated titer thresholds, μ_{ij}), our response variables are distributions: $\text{Normal}(\mu_{ij}, \sigma_{ij}^2)$. This is a straightforward approach to measurement error in a Bayesian context [32].

6.7.2 Step 2: Modeling Thresholds

Mixed-effects linear regression We used a robust Bayesian mixed-effects linear regression to model the titer thresholds. We used a Student's *t* distribution instead of a Normal distribution as the likelihood to make the model more robust to outliers [26]. We followed a measurement error approach and specified our response variables as Normal distributions corresponding to titer threshold estimates and their measurement error (see Step 1 above). We specified *proxy* as a fixed effect, so that different proxies can have different titer thresholds on average. We then used a random intercept and random slopes for *proxy* dependent on *participant*. This allows each participant to have their own titer thresholds within each proxy in the model. In `brms`'s [5] extended

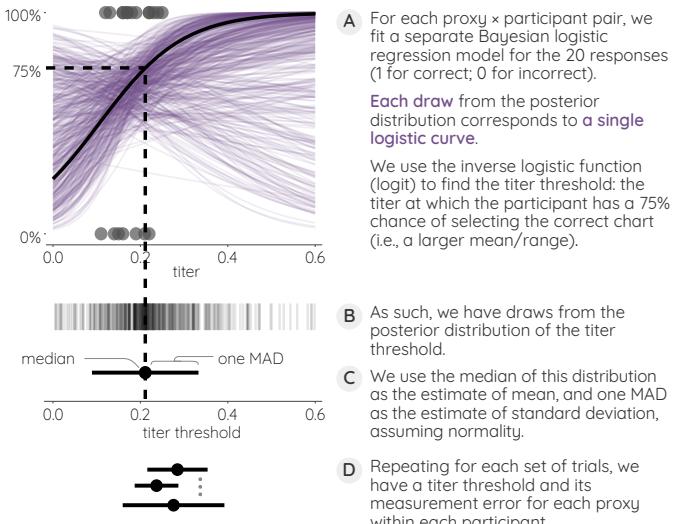


Fig. 3. Deriving titer thresholds and measurement error.

Wilkinson-Rogers [47] notation, this model is:

$$\text{tierThreshold} | \text{se}(\text{tierError}) \sim \text{proxy} + (\text{proxy} | \text{participant}) \quad (2)$$

Where *tierThreshold* is the estimated titer threshold (μ_{ij} above), *tierError* is the measurement error in the titer threshold (σ_{ij} above), and *proxy* and *participant* are categorical variables indicating the manipulated proxy and participant, respectively.

6.8 Results

We report medians, 50% and 95% quantile credible intervals (CIs; Bayesian analogs to confidence intervals) as estimates of mean effects, and present the medians of posterior predictive distributions to show individual differences, following the presenting style of Fernandes et al. [13] and Hullman et al. [18].

6.8.1 The Effects of Manipulating Perceptual Proxies

We report the mean effects for each proxy and comparisons with the control condition where no proxy was manipulated in Fig. 4. We found evidence to support \mathcal{H}_1 : participants are likely deceived by our manipulation and could use proxies differently.

MaxMean (Fig. 4a) The four proxies have posterior distributions surrounding and similar to the control condition. When looking at the posterior distributions of differences in titer threshold, weak evidence supports that manipulating *centroid* might lead to a *larger* average titer threshold, suggesting that an average participant might be *deceived* by the *centroid* proxy, and therefore might be using that proxy to estimate *MaxMean*. Manipulating *hull area*, *max bar*, or *min bar* are less likely to have a large effect on average, suggesting that an average participant is less likely to be deceived by those proxies.

MaxRange (Fig. 4b) We did not find strong evidence of an effect of either *hull area norm* or *slope neighbor* on titer threshold. The *slope neighbor* proxy is most likely to lead to larger titer thresholds, but neither the chance of this nor the associated size of the effect are large. We found *slope* and *slope range* are likely to yield smaller titer thresholds, suggesting that an average participant is more likely to *select against* these

Table 3. An example of participants selecting against a proxy.

Two conflicting proxies	Selecting against
	<p>Assume we are manipulating <i>slope</i>. The orange chart appears to be deceptive: it has a <i>larger slope</i> but a <i>smaller range</i>.</p> <p>The blue chart has a <i>larger slope neighbor</i>, which is negatively correlated with <i>slope</i>.</p> <p>If participants use <i>slope neighbor</i>, they would appear to select against <i>slope</i>: they always choose the incorrect chart of a <i>smaller slope</i> and <i>range</i>.</p>

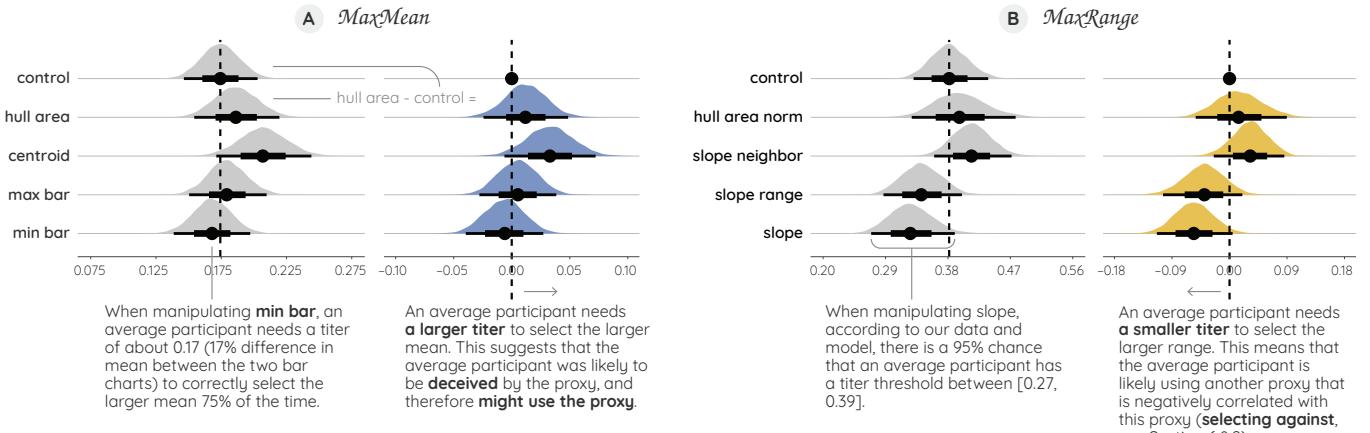


Fig. 4. The effects of manipulating perceptual proxies (\mathcal{H}_1). We show posterior distributions (▲), 50% and 95% CIs (—) of expected titer thresholds, and a comparison with the control condition for an average participant in both *MaxMean* and *MaxRange* tasks.

two proxies. As we explain below, this may suggest the presence of some other proxies, negatively correlated (conflicting) with *slope range* and *slope neighbor*, which an average participant might be using.

6.8.2 Interpreting Participants Selecting Against a Proxy

We found that *slope range* and *slope* might lead to smaller titer thresholds on average than the control condition. When this happens, we say that participants are *selecting against* a proxy. Consider two bar charts, *A* and *B* (see Table 3). *A* has the larger *slope* (our manipulated proxy) and the smaller range of the two; *B* has the smaller *slope* but the larger range. Say participants do not use *slope*, but do use some other proxy \mathcal{Y} that is negatively correlated with *slope* (e.g., *slope neighbor*), such that *B* has the larger value of \mathcal{Y} . Now *B* has both the larger value of \mathcal{Y} and the larger *slope*, so participants using proxy \mathcal{Y} will be more likely to correctly pick *B* at a smaller titer, leading *slope* to have a smaller titer threshold than the control. Thus, the smaller titer thresholds of *slope range* and *slope* suggest there may be some other proxy (negatively correlated with *slope range* or *slope*) that participants *were using*.

6.8.3 Individual Differences

To investigate individual differences, we report each participant's median of predicted expected titer threshold and a comparison to the control condition across different proxies in Fig. 5, assuming no measurement error. We found evidence supports that participants use proxies differently for our \mathcal{H}_2 .

MaxMean (Fig. 5a) We found that on average, most participants are consistent with themselves across all conditions (③): participants who have larger titer thresholds than others in the control condition are more likely to have larger titer thresholds in other conditions and vice versa. This is reasonable: if participants are good at selecting the larger mean between the two charts, they could have been good at the task across different conditions, and thus result in smaller titer thresholds in all the conditions. A large portion of participants behave similarly (④), but a small portion of participants have larger titer thresholds than the others.

We found that most participants seem to be deceived by the adversarial trials, suggesting that they might use the manipulated proxies or other proxies positively correlated with these. The exception is that in the *min bar* condition, participants seem to be consistently and slightly

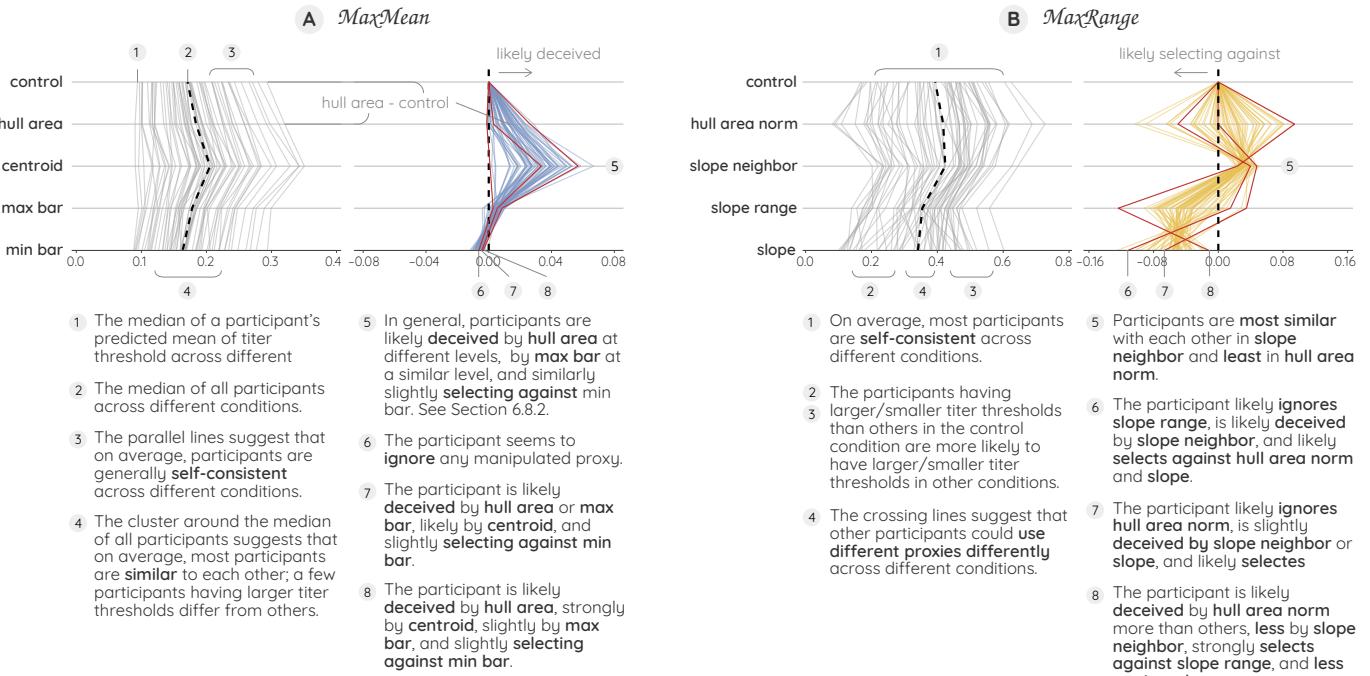


Fig. 5. The individual differences in different proxies conditions (\mathcal{H}_2). We show predicted median of titer thresholds, and a comparison with the control condition for each participant in both *MaxMean* and *MaxRange* tasks.

selecting against our manipulation, indicating that they might use other proxies negatively correlated with *min bar*. A handful of participants seem not to follow any manipulation (6); their titer thresholds are similar to those of the control condition. Different participants are likely to be deceived by different proxies to different extents (7). While the majority of participants seem to be deceived by *centroid* the most (8), *centroid* is also where participants' behavior deviate from each other the most. Last, participants are more similar across and within *min bar* and *max bar* conditions, meaning that in our procedure, if participants use the *max bar*, they are less likely to use *min bar*, consistent with our observations from Section 6.8.2.

MaxRange (Fig. 5b) We found that most participants appear to be self-consistent across all the proxy conditions (1), but less consistent than the *MaxMean* task (they were different participants). Participants who have larger titer thresholds than others in the control condition are more likely to have larger titer thresholds in other conditions (2) and vice versa (3). These two groups appear to have similar numbers of participants, and there are other participants who behave differently across different conditions (4).

We found evidence that supports participants might use different proxies differently across different conditions. Participants are most similar to each other in *slope neighbor* (5); but they are least similar in *hull area norm*. Some participants could be deceived by the manipulated proxy, while some are selecting against the proxy, and others are likely not to follow the manipulation; most participants are likely selecting against both *slope* and *slope range*. Different participants may ignore a manipulated proxy, be deceived by a second one, but select against another one (6-8).

7 EXPERIMENT 2: SEARCHING FOR ADVERSARIAL DATA

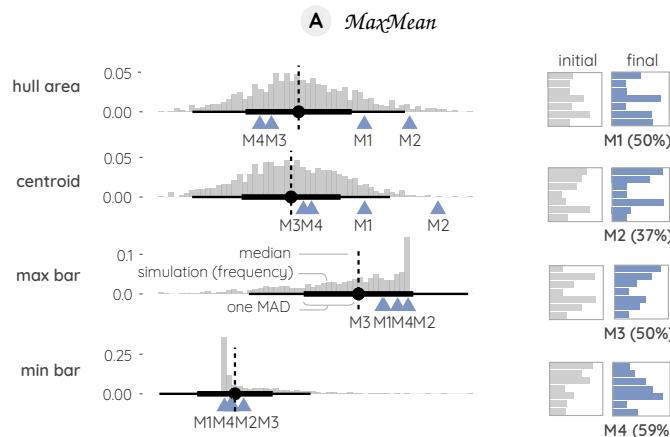
In Experiment 1, we started from the assumption that proxies may be at play and attempted to probe their effects. In this second experiment, we approach the question from the opposite direction, asking instead: what kinds of datasets *appear* to have a larger mean or range? We consider the human perception of the summary statistic to be a black-box function that we are seeking to optimize. We then stochastically estimate the gradient descent process using only pairwise ranking, and each pairwise ranking is the forced choice between two bar charts (see Section 4.2).

7.1 Hypotheses

We framed two hypotheses for Experiment 2:

H_3 Optimized charts will display identifiable characteristics corresponding to the proposed proxies.

H_4 Optimized charts will be adversarial, appearing to have larger summary statistics versus random charts with the same statistics.



7.2 Experimental Design

We performed optimization both within- and between-subjects to smooth out the noise in the measure of gradient. That is, we queried each participant multiple times and recruited multiple participants. We started four different random initializations in parallel (four epochs) and assigned each initialization to five sequential participants. Each of these participants completed 20 trials. The trials were seamlessly interleaved with those of Experiment 1 (see Section 5). However, different from Experiment 1, the two charts in each trial had the identical summary statistic—there is no *correct* answer (which amounts to the titer value being 0 for all trials). To participants, these trials would seem just like very difficult trials in the same experiment. Similar to Experiment 1, task-integral factors (the total amount of ink for *MaxMean*; *min bar* and *max bar* for *MaxRange*) were controlled and balanced between sides (see Section 6.2).

The last 22 or 23 participants (*MaxMean* and *MaxRange*, respectively) validated the optimized charts from the first 20 participants, instead of performing the optimization. They were shown the optimized charts along with other randomly initialized charts. Each of these participants saw all four optimized charts from the four epochs once and only once (for the task they were performing). In other words, each final optimized chart was subjected to 22 or 23 total validation trials. The remaining 16 trials used two randomly initialized charts.

7.3 Analysis

We performed both quantitative analysis and qualitative visual inspection for the optimization results. To see if the optimized charts reflected the properties of the proposed proxies, we computed the proposed proxies from the charts and compared them to a random guessing simulation. The simulation used the same algorithm as initialization, performed 1,000 times with 100 guessing trials (simulating 20 trials per participant \times 5 participants). We then computed median and MAD from the simulation for comparison. We also computed the ratio that a final optimized chart was selected by a participant for that task in the evaluation trials. Last, we plotted each of the optimized charts along with their random initializations in Fig. 6, denoted by \mathcal{M}_i and \mathcal{R}_j ($i, j \in \{1, \dots, 4\}$) for the two tasks, respectively.

7.4 Results

Our observations from Experiment 2 are as follows.

MaxMean We found that \mathcal{M}_1 and \mathcal{M}_2 are at least one MAD away from the median of the random guessing results for *centroid* and *hull area*, and half for *max bar*. In the validation trials, none of the final charts were selected by participants higher than chance (50%). In particular, in \mathcal{M}_1 and \mathcal{M}_2 , the bars have been pushed toward the extrema. We can conjecture that the prominence of the larger bars causes them to carry

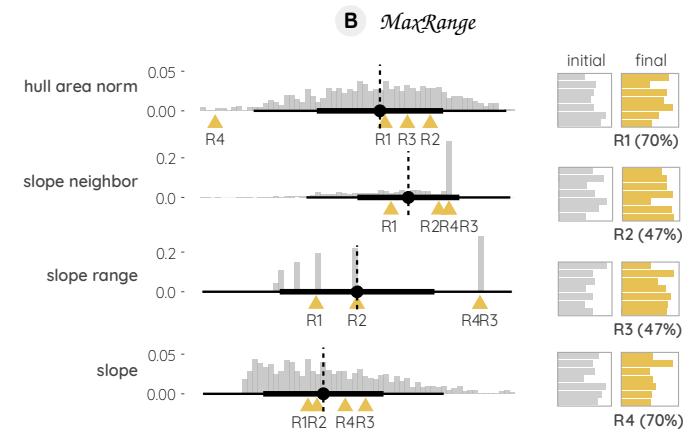


Fig. 6. The results of Experiment 2 (H_3 and H_4). We show the computed proxies in the final optimized charts along with those from a random guessing simulation, the initializations against the final charts (e.g., \mathcal{M}_1 and \mathcal{R}_1), and the win ratio against other random generated charts in the validation trials.

more weight, increasing the perceived mean. In \mathcal{M}_3 and \mathcal{M}_4 , there are staircase patterns, which may suggest a proxy related to slope.

MaxRange We found that \mathcal{R}_3 and \mathcal{R}_4 seem to suggest *slope range*. However, \mathcal{R}_4 is about two MADs from the median of *hull area norm* in the negative direction, slightly suggesting against this *hull area norm*; and \mathcal{R}_1 seems to suggest against most of the proposed proxies. In the validation trials, \mathcal{R}_1 and \mathcal{R}_4 are well above chance (50%), the very similar charts \mathcal{R}_2 and \mathcal{R}_3 are slightly below. This discrepancy could be an effect of individual differences. In \mathcal{R}_3 and \mathcal{R}_4 , there is an inverse of this motif: the maximum is flanked by either the minimum or bars close to it. We expect both of these motifs should correspond with *slope range* and *slope neighbor*. Turning to range, \mathcal{R}_1 and \mathcal{R}_2 appear to have “notches,” in which the shortest bar is flanked by bars near the maximum. We conjecture that this juxtaposition simplifies extraction of the range. This motif may not make the range appear larger, but easier to estimate, making it more attractive in a forced-choice task.

8 DISCUSSION

Here we discuss connections between the experiments, implications to visualization, and the limitations of our work.

8.1 Connection between the Two Experiments

The two experiments were approaching the same problem from two different directions; that is, we used both theory-driven (Experiment 1) and data-driven (Experiment 2) approaches to look for evidence that participants might have used perceptual proxies in the two tasks. Experiment 1 carefully optimized datasets based on pre-defined proxies, whereas Experiment 2 did away with all the preconceived notions of dataset characteristics and solves for deception through a black-box optimization. Experiment 1 constrained proxy and data space to optimize a few pre-defined proxies, while Experiment 2 explored a broader scope of both proxy and data space.

The meeting point of the two experiments is the evidence that participants might have used certain and the same proxies in both experiments. For example, for the *MaxMean* task, both experiments suggest that participants might have used *centroid* as a proxy.

8.2 Implications to Visualizations and Beyond

Our findings may suggest that when a visualization is precisely designed and applied for a specific task, it is possible that participants will be misled simply by virtue of the data. In a way, this is a corollary to Anscombe’s quartet where even a correct (even the “right”) visualization for a specific dataset can be misleading. This hints at some of the “black hat” visualization work discussed by Correll and Heer [9], where it is useful to start to think about visualization in the language of computer security, and where a particular visualization can be open to unintentional (or malicious) attacks even with the best of intentions.

However, our efforts to skew perception along these vectors for the sake of investigation have shown that, in practice, this is quite difficult, and likely to be subtle if successful. A malicious designer would thus have many paths of lower resistance [42].

Still, being aware of this problem is the first step towards addressing it. In the short term, establishing the preferred perceptual proxies for not just individuals, but also populations, may allow us to pinpoint situations where unfortunate (or intentional) configurations of data may lead to incorrect perceptions. In the longer term, the perceptual proxies we have proposed here may become the building blocks for perceptual frameworks that are capable of assessing any given visual representation and dataset, and report on the data loss inherent for different subsets of the population.

Finally, another aspect of our work that is useful to the visualization community is the methodological framework we have devised to test these phenomena. Our approach builds on the crowdsourced staircase titration design by Ondov et al. [33] and Jardine et al. [19], the perceptual framework scoring by Yuan et al. [50] and Jardine et al. [19], as well as the simulated dataset generation approach initially proposed by Matejka and Fitzmaurice [31]. We hope to see future studies in visualization use similar reactive testing frameworks such as ours to empirically derive increasingly more complex visual phenomena.

8.3 Limitations

While our approaches revealed some interesting findings, they are limited in many ways. For our theory-driven approach, though we added proxies to those used in previous studies, we still cannot claim to have anything approaching an exhaustive list, nor can we claim strong motivations for testing these particular proxies. Additionally, we intentionally omit some proxies that are either directly connected to their corresponding summary statistics or highly correlated with chosen proxies, which would be difficult to control independently, and thus to measure. While this necessarily limits the conclusions we can draw about specific proxies that lie within broader classes, we believe probing these few representatives is a necessary first step towards disentangling the myriad of proxies that have been proposed, and are yet to be conceived of.

Our data-driven approach is only scratching the surface of what we believe is possible with this paradigm. We ran relatively low numbers of iterations, and thus could see patterns better simply by obtaining more data. We also perform only rudimentary analyses, leaving probabilistic evidence of the potency of optimized charts for future study.

9 CONCLUSION AND FUTURE WORK

In this paper, we have reported on two experiments designed to probe the boundaries of human perception with the purpose of determining the low-level “programs” that the visual system executes in order to quickly extract data from a visualization. One experiment presupposes the use of posited features such as centroid or convex hull area, while the other attempts to optimize the deceptiveness in the space of raw data. For both examples, we study simple lineups of side-by-side bar charts with seven data points each, and use different techniques to generate adversarial datasets.

Combining the two experiments show interesting, if preliminary, signs that perceptual proxy theories from vision science can be observed in unstructured perceptual response data. For one thing, the datasets refined through black-box optimization from random data hint at specific spatial patterns that are reminiscent of perceptual proxies. Furthermore, while we cannot support the overall claim that the optimized datasets actually are adversarial, we did find several individual such datasets that show much better performance at being deceptive than random chance. However, more work is needed to investigate and confirm these phenomena in detail.

Many potential avenues for future research exists. Our study here has so far been constrained to the visual comparison task within side-by-side bar charts. Obviously, this represents only a single point in a large design space consisting of visualizations, tasks, and visual arrangements—the so-called “cube” proposed in Jardine et al. [19]. However, as was observed in the latter, continuing to explore this design space on a point-by-point basis is likely going to be an extremely time-consuming and ultimately inefficient approach. Just as Jardine et al. [19] proposed perceptual proxies as a reasoning framework to raise the abstraction level and explain all of these phenomena in one fell swoop, must we also endeavor to understand the relationship between different proxies for different visualizations, layouts, and tasks. Put differently, it is highly unlikely that the visual system has developed specialized “programs” (or proxies) for every conceivable visual representation. It is more likely that there are clear commonalities between the proxies used for different tasks, and moreover that specific individuals have specific affinities for various such proxies. In fact, our population analysis provides some support for this hypothesis. This would mean that a more fruitful gradient to optimize for future work would be to try to identify and generalize perceptual proxies across different visualizations and tasks.

ACKNOWLEDGMENTS

This work was supported partly by grant IIS-1901485 from the U.S. National Science Foundation and partly by the Intramural Research Program of the National Human Genome Research Institute, a part of the U.S. National Institutes of Health. Any opinions, findings, and conclusions or recommendations expressed here are those of the authors and do not necessarily reflect the views of the funding agencies.

REFERENCES

[1] R. Amar, J. Eagan, and J. Stasko. Low-level components of analytic activity in information visualization. In *Proceedings of the IEEE Symposium on Information Visualization*, pp. 111–117. IEEE, Piscataway, NJ, USA, 2005. doi: 10.1109/INFOVIS.2005.24

[2] R. A. Amar and J. T. Stasko. Knowledge precepts for design and evaluation of information visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 11(4):432–442, July 2005. doi: 10.1109/TVCG.2005.63

[3] J. M. Bland and D. G. Altman. Statistics notes: Measurement error. *British Medical Journal*, 312(7047):1654, 1996. doi: 10.1136/bmjj.312.7047.1654

[4] M. Bostock, V. Ogievetsky, and J. Heer. D3: Data-driven documents. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2301–2309, Dec. 2011. doi: 10.1109/TVCG.2011.185

[5] P.-C. Bürkner et al. brms: An R package for Bayesian multilevel models using Stan. *Journal of Statistical Software*, 80(1):1–28, 2017.

[6] C. M. Carswell. Choosing specifiers: an evaluation of the basic tasks model of graphical perception. *Human Factors*, 34(5):535–554, 1992. doi: 10.1177/001872089203400503

[7] W. S. Cleveland and R. McGill. Graphical perception: Theory, experimentation and application to the development of graphical methods. *Journal of the American Statistical Association*, 79(387):531–554, Sept. 1984. doi: 10.2307/2288400

[8] T. N. Cornsweet. The staircase-method in psychophysics. *The American Journal of Psychology*, 75(3):485–491, 1962. doi: 10.2307/1419876

[9] M. Correll and J. Heer. Black hat visualization. In *Proceedings of the IEEE VIS Workshop on Dealing with Cognitive Biases in Visualisations*, 2017.

[10] M. Correll, M. Li, G. Kindlmann, and C. Scheidegger. Looks good to me: Visualizations as sanity checks. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):830–839, 2018. doi: 10.1109/TVCG.2018.2864907

[11] R. Dawkins. *The Blind Watchmaker: Why the Evidence of Evolution Reveals a Universe without Design*. WW Norton & Company, 1986.

[12] M. G. Falletti, P. Maruff, A. Collie, and D. G. Darby. Practice effects associated with the repeated assessment of cognitive function using the CogState battery at 10-minute, one week and one month test-retest intervals. *Journal of Clinical and Experimental Neuropsychology*, 28(7):1095–1112, 2006. doi: 10.1080/13803390500205718

[13] 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 ACM Conference on Human Factors in Computing Systems*, pp. 144:1–144:12. ACM, New York, NY, USA, 2018.

[14] G. A. Gescheider. The classical psychophysical methods. *Psychophysics: the fundamentals*, pp. 45–72, 1997.

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

[16] L. Harrison, F. Yang, S. Franconeri, and R. Chang. Ranking visualizations of correlation using Weber's law. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):1943–1952, 2014. doi: 10.1109/TVCG.2014.2346979

[17] S. Hecht. The visual discrimination of intensity and the Weber-Fechner law. *The Journal of General Physiology*, 7(2):235–267, 1924. doi: 10.1085/jgp.7.2.235

[18] J. Hullman, M. Kay, Y.-S. Kim, and S. Shrestha. Imagining replications: Graphical prediction & discrete visualizations improve recall & estimation of effect uncertainty. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):446–456, 2018. doi: 10.1109/TVCG.2017.2743898

[19] N. Jardine, B. D. Ondov, N. Elmquist, and S. Franconeri. The perceptual proxies of visual comparison. *IEEE Transactions on Visualization and Computer Graphics*, 26(1):1012–1021, 2020. doi: 10.1109/tvcg.2019.2934786

[20] 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*, 25(1):892–902, 2018.

[21] M. Kay. *ggdist: Visualizations of Distributions and Uncertainty*, 2020. R package version 2.2.0.9000. doi: 10.5281/zenodo.3879620

[22] M. Kay. *tidybayes: Tidy Data and Geoms for Bayesian Models*, 2020. R package version 2.1.1.9000. doi: 10.5281/zenodo.1308151

[23] M. Kay and J. Heer. Beyond Weber's law: A second look at ranking visualizations of correlation. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):469–478, 2016. doi: 10.1109/TVCG.2015.2467671

[24] M. Kay, G. L. Nelson, and E. B. Hekler. Researcher-centered design of statistics: Why Bayesian statistics better fit the culture and incentives of HCI. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, pp. 4521–4532. ACM, New York, NY, USA, 2016. doi: 10.1145/2858036.2858465

[25] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983. doi: 10.1126/science.220.4598.671

[26] J. K. Kruschke. Bayesian estimation supersedes the t test. *Journal of Experimental Psychology: General*, 142(2):573, 2013. doi: 10.1037/a0029146

[27] J. H. Larkin and H. A. Simon. Why a diagram is (sometimes) worth ten thousand words. *Cognitive Science*, 11(1):65–100, 1987. doi: 10.1111/j.1551-6708.1987.tb00863.x

[28] E. Law and L. v. Ahn. Human computation. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 5(3):1–121, 2011.

[29] C. Leys, C. Ley, O. Klein, P. Bernard, and L. Licata. Detecting outliers: Do not use standard deviation around the mean, use absolute deviation around the median. *Journal of Experimental Social Psychology*, 49(4):764–766, 2013. doi: 10.1016/j.jesp.2013.03.013

[30] D. Marr. *Vision: A Computational Investigation Into the Human Representation and Processing of Visual Information*. W. H. Freeman and Company, San Francisco, CA, USA, 1982.

[31] J. Matejka and G. Fitzmaurice. Same stats, different graphs: Generating datasets with varied appearance and identical statistics through simulated annealing. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, pp. 1290–1294. ACM, New York, NY, USA, 2017. doi: 10.1145/3025453.3025912

[32] R. McElreath. *Statistical Rethinking: A Bayesian Course with Examples in R and Stan*. CRC Press, 2015.

[33] B. D. Ondov, N. Jardine, N. Elmquist, and S. Franconeri. Face to face: Evaluating visual comparison. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):861–871, 2019. doi: 10.1109/TVCG.2018.2864884

[34] A. V. Pandey, K. Rall, M. L. Satterthwaite, O. Nov, and E. Bertini. How deceptive are deceptive visualizations?: An empirical analysis of common distortion techniques. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, pp. 1469–1478. ACM, New York, NY, USA, 2015. doi: 10.1145/2702123.2702608

[35] A. J. Quinn and B. B. Bederson. Human computation: a survey and taxonomy of a growing field. pp. 1403–1412. ACM, New York, NY, USA, 2011. doi: 10.1145/1978942.1979148

[36] R. M. Ratwani, J. G. Trafton, and D. A. Boehm-Davis. Thinking graphically: Connecting vision and cognition during graph comprehension. *Journal of Experimental Psychology: Applied*, 14(1):36–49, 2008. doi: 10.1037/1076-898X.14.1.36

[37] R. A. Rensink. The nature of correlation perception in scatterplots. *Psychonomic Bulletin & Review*, pp. 1–22, 2016. doi: 10.3758/s13423-016-1174-7

[38] R. A. Rensink and G. Baldridge. The perception of correlation in scatterplots. *Computer Graphics Forum*, 29(3):1203–1210, 2010. doi: 10.1111/j.1467-8659.2009.01694.x

[39] L. M. Rios and N. V. Sahinidis. Derivative-free optimization: a review of algorithms and comparison of software implementations. *Journal of Global Optimization*, 56(3):1247–1293, 2013. doi: 10.1007/s10898-012-9951-y

[40] D. Simkin and R. Hastie. An information-processing analysis of graph perception. *Journal of the American Statistical Association*, 82(398):454–465, 1987. doi: 10.1080/01621459.1987.10478448

[41] Stan Development Team. RStan: the R interface to Stan, 2018. R package version 2.18.2.

[42] E. R. Tufte. *The Visual Display of Quantitative Information*. Graphics Press, Cheshire, CT, USA, 1986.

[43] E.-J. Wagenmakers, M. Marsman, T. Jamil, A. Ly, J. Verhagen, J. Love, R. Selker, Q. F. Gronau, M. Šimřá, S. Epskamp, et al. Bayesian inference for psychology. Part I: theoretical advantages and practical ramifications. *Psychonomic Bulletin & Review*, 25(1):35–57, 2018. doi: 10.3758/s13423-017-1343-3

[44] D. M. Webster, L. Richter, and A. W. Kruglanski. On leaping to conclude

sions when feeling tired: Mental fatigue effects on impressional primacy. *Journal of Experimental Social Psychology*, 32(2):181–195, 1996. doi: 10.1006/jesp.1996.0009

[45] H. Wickham, M. Averick, J. Bryan, W. Chang, L. D. McGowan, R. François, G. Grolemund, A. Hayes, L. Henry, J. Hester, M. Kuhn, T. L. Pedersen, E. Miller, S. M. Bache, K. Müller, J. Ooms, D. Robinson, D. P. Seidel, V. Spinu, K. Takahashi, D. Vaughan, C. Wilke, K. Woo, and H. Yutani. Welcome to the tidyverse. *Journal of Open Source Software*, 4(43):1686, 2019. doi: 10.21105/joss.01686

[46] H. Wickham, D. Cook, H. Hofmann, and A. Buja. Graphical inference for infovis. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):973–979, 2010. doi: 10.1109/TVCG.2010.161

[47] G. N. Wilkinson and C. E. Rogers. Symbolic description of factorial models for analysis of variance. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 22(3):392–399, 1973. doi: 10.2307/2346786

[48] F. Yang, L. Harrison, R. A. Rensink, S. Franconeri, and R. Chang. Correlation judgment and visualization features: A comparative study. *IEEE Transactions on Visualization and Computer Graphics*, 25(3):1474–1488, Mar. 2018. doi: 10.1109/TVCG.2018.2810918

[49] J. S. Yi, Y. ah Kang, J. T. Stasko, and J. A. Jacko. Toward a deeper understanding of the role of interaction in information visualization. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1224–1231, 2007. doi: 10.1109/TVCG.2007.70515

[50] L. Yuan, S. Haroz, and S. Franconeri. Perceptual proxies for extracting averages in data visualizations. *Psychonomic Bulletin & Review*, 26:669–676, 2019. doi: 10.3758/s13423-018-1525-7

[51] Y. Yue and T. Joachims. Interactively optimizing information retrieval systems as a dueling bandits problem. In *Proceedings of the Annual International Conference on Machine Learning*, pp. 1201–1208, 2009. doi: 10.1145/1553374.1553527

Revealing Perceptual Proxies with Adversarial Examples

Supplementary Materials

Brian D. Ondov, Fumeng Yang, Matthew Kay, Niklas Elmqvist, Steven Franconeri

A DUELING BANDIT GRADIENT DESCENT FOR EXPERIMENT 2

To optimize a chart based on how large a summary statistic appears, without actually changing it, we use an implementation of Dueling Bandit Gradient Descent [51]. Our version of the algorithm is depicted in Algorithm 1. This method requires a projection function \mathcal{P} to map points from euclidean space to the feasible set for a given optimization problem. In this case, the feasible set is all charts with the same mean μ (for *MaxMean* task) or the same range $[min, max]$ (for the *MaxRange* task). Taking steps along random vectors, as required by the algorithm, typical moves the query point outside of the feasible set (i.e. it changes the mean or range of the chart), and the projection allows the algorithm to stay as close as possible to the chosen point while satisfying the constraints. The estimation methods we use for these projections to a given mean (\mathcal{P}_m) or range (\mathcal{P}_r) are given in Algorithms 2 and 3, respectively. Note that the *data* space we explore lies between 0 and 1 in each dimension. However, since our charts have a minimum bar length of 0.25 times the chart width, data are mapped from $[0,1]$ to $[0.25,1]$ when converting to *bar* space.

Algorithm 1 The Dueling Bandit Gradient Descent algorithm. We deviate slightly from the algorithm as originally published in the case where the new point is not better than the current one: rather than doing nothing, we instead take a small step opposite the random exploration vector \mathbf{u} (line 15), both so that progress is made faster and so that participants do not see repeats of identical charts. This introduces a new parameter, η to denote the magnitude of this backward-step vector.

```

1:  $n :=$  dimensions (7)
2:  $\mathbf{d}_0 :=$  initial point,  $\in [0, 1]^n$ 
3:  $\delta :=$  exploration step size (0.5)
4:  $\gamma :=$  forward exploitation step size (0.9)
5:  $\eta :=$  backward exploitation step size (0.1)
6:  $k :=$  iterations (20)
7:  $\mathcal{P} :=$  project to feasible set
8: procedure DBGD( $\mathbf{d}_0, \delta, \gamma, \eta, k$ )
9:   for  $i = 1..k$  do
10:     $\mathbf{u} \leftarrow$  uniformRandomUnitVector()
11:     $\mathbf{d}' \leftarrow \mathcal{P}(\mathbf{d}_{i-1} + \delta \mathbf{u})$ 
12:    if  $\mathbf{d}' \succ \mathbf{d}_{i-1}$  then
13:       $\mathbf{d}_i \leftarrow \mathcal{P}(\mathbf{d}_{i-1} + \gamma \mathbf{u})$ 
14:    else
15:       $\mathbf{d}_i \leftarrow \mathcal{P}(\mathbf{d}_{i-1} - \eta \mathbf{u})$ 
16:   return  $\mathbf{d}_k$ 

```

Algorithm 2 Projection to mean. This function seeks to find the vector \mathbf{v}' s.t. the mean of \mathbf{v}' is a target value and the distance of \mathbf{v}' to the original vector \mathbf{v} is minimized. Without bounds, this could simply be accomplished by distributing the difference between the current and target means evenly among each dimension (line 10). However, this adjustment may move some elements of \mathbf{v} outside of the bounds $[0,1]$; imposing these bounds in turn changes the mean. An iterative optimization is this required. The algorithm terminates when the mean is within a tolerance or it has performed too many iterations.

```

1:  $n :=$  dimensions (7)
2:  $\mathbf{v} :=$  point to project,  $\in \mathbb{R}^n$ 
3:  $\mu^* :=$  target mean (0.4)
4:  $\epsilon :=$  error tolerance ( $1 \times 10^{-6}$ )
5:  $i_{max} :=$  max iterations (100)
6: procedure  $\mathcal{P}_m(\mathbf{v})$ 
7:    $\Delta_\mu \leftarrow \text{mean}(\mathbf{v}) - \mu^*$ 
8:   while  $|\Delta_\mu| > \epsilon$  and  $i < i_{max}$  do
9:     for  $j = 1..n$  do
10:       $\mathbf{v}_j \leftarrow \mathbf{v}_j - \frac{1}{n} \Delta_\mu$ 
11:       $\mathbf{v}_j \leftarrow \max(0, \min(1, \mathbf{v}_j))$ 
12:    $\Delta_\mu \leftarrow \text{mean}(\mathbf{v}) - \mu^*$ 
13:    $i \leftarrow i + 1$ 
14: return  $\mathbf{v}$ 

```

Algorithm 3 Projection to range. This function simply performs linear interpolation to map a vector from its original range to the interval [0,1].

```

1:  $n :=$  dimensions (7)
2:  $\mathbf{v} :=$  point to project,  $\in \mathbb{R}^n$ 
3: procedure  $\mathcal{P}_r(\mathbf{v})$ 
4:    $v_{min} \leftarrow \min(\mathbf{v})$ 
5:    $v_{max} \leftarrow \max(\mathbf{v})$ 
6:   for  $i = 1..n$  do
7:      $\mathbf{v}_i \leftarrow (\mathbf{v}_i - v_{min}) / (v_{max} - v_{min})$ 
8:   return  $\mathbf{v}$ 

```

B FULL CORRELATION MATRICES OF PROXIES

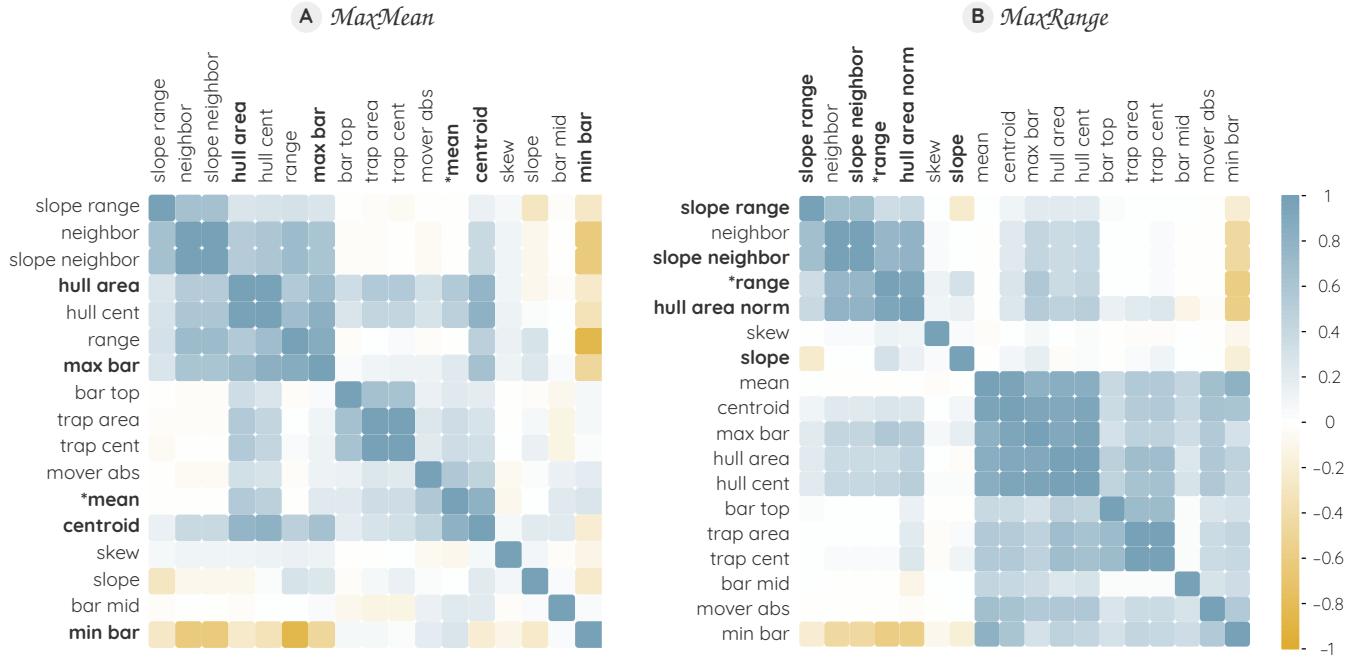


Fig. B.1. **Correlation matrices for all the proxies considered.** This figure illustrates the correlation between all the proxies considered for each task. The matrices were computed based on the data from Jardine et al. [19].

C EXAMPLES OF TITER VALUES AND PROXIES

Table C.1. *MaxMean* and *titer*. We show examples of bar charts for each condition, including the control condition. In each proxy condition, we optimize that particular proxy and make it more deceptive. **The bar charts on the left side** have roughly the same mean value around 0.4. **The bar charts on the right side** always have a mean value higher than 0.4, which are the correct answers. Note that to facilitate comparison, we use the same thickness for all the bars.

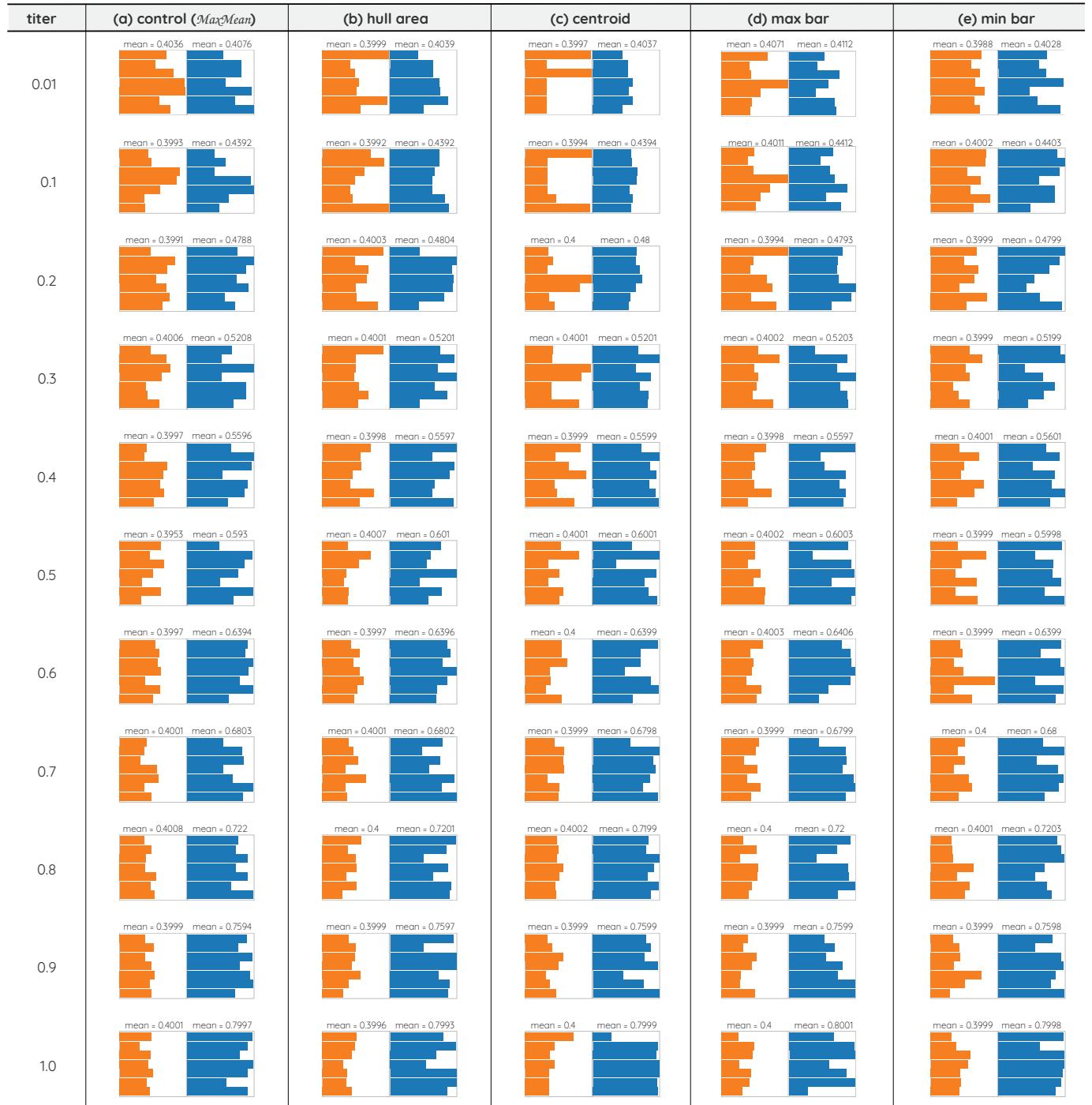
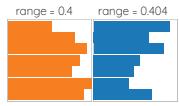
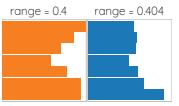
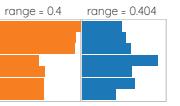
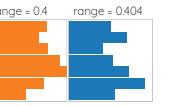
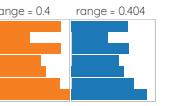
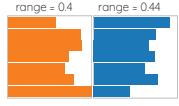
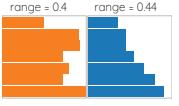
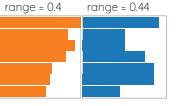
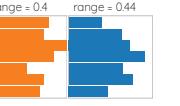
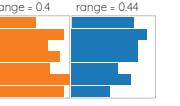
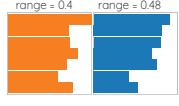
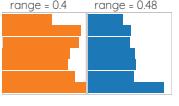
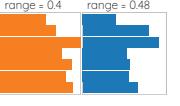
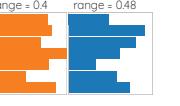
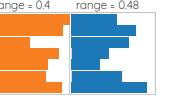
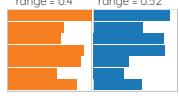
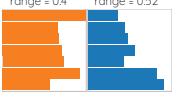
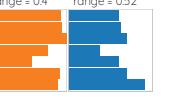
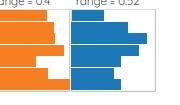
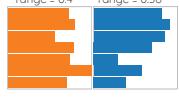
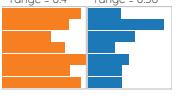
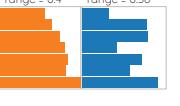
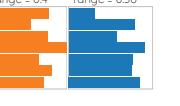
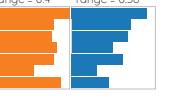
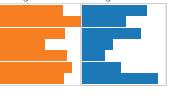
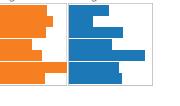
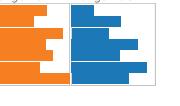
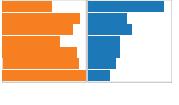
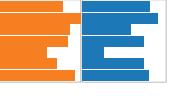
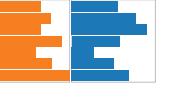
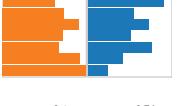
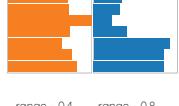
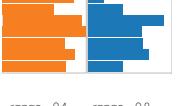
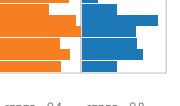
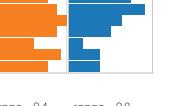
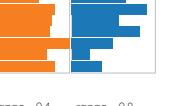
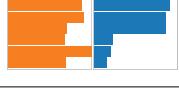
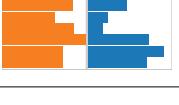


Table C.2. *MaxRange* and *titer*—We show examples of bar charts for each condition, including the control condition. In each proxy condition, we optimize that particular proxy and make it more deceptive. The bar charts on the left side have the same range 0.4. The bar charts on the right side always have a range value larger than 0.4, which are the correct answers.

titer	(a) control (<i>MaxRange</i>)	(b) hull area norm	(c) slope	(d) slope range	(e) slope neighbor
0.01	 range = 0.4 range = 0.404	 range = 0.4 range = 0.404	 range = 0.4 range = 0.404	 range = 0.4 range = 0.404	 range = 0.4 range = 0.404
0.1	 range = 0.4 range = 0.44	 range = 0.4 range = 0.44	 range = 0.4 range = 0.44	 range = 0.4 range = 0.44	 range = 0.4 range = 0.44
0.2	 range = 0.4 range = 0.48	 range = 0.4 range = 0.48	 range = 0.4 range = 0.48	 range = 0.4 range = 0.48	 range = 0.4 range = 0.48
0.3	 range = 0.4 range = 0.52	 range = 0.4 range = 0.52	 range = 0.4 range = 0.52	 range = 0.4 range = 0.52	 range = 0.4 range = 0.52
0.4	 range = 0.4 range = 0.56	 range = 0.4 range = 0.56	 range = 0.4 range = 0.56	 range = 0.4 range = 0.56	 range = 0.4 range = 0.56
0.5	 range = 0.4 range = 0.6	 range = 0.4 range = 0.6	 range = 0.4 range = 0.6	 range = 0.4 range = 0.6	 range = 0.4 range = 0.6
0.6	 range = 0.4 range = 0.64	 range = 0.4 range = 0.64	 range = 0.4 range = 0.64	 range = 0.4 range = 0.64	 range = 0.4 range = 0.64
0.7	 range = 0.4 range = 0.68	 range = 0.4 range = 0.68	 range = 0.4 range = 0.68	 range = 0.4 range = 0.68	 range = 0.4 range = 0.68
0.8	 range = 0.4 range = 0.72	 range = 0.4 range = 0.72	 range = 0.4 range = 0.72	 range = 0.4 range = 0.72	 range = 0.4 range = 0.72
0.9	 range = 0.4 range = 0.76	 range = 0.4 range = 0.76	 range = 0.4 range = 0.76	 range = 0.4 range = 0.76	 range = 0.4 range = 0.76
1.0	 range = 0.4 range = 0.8	 range = 0.4 range = 0.8	 range = 0.4 range = 0.8	 range = 0.4 range = 0.8	 range = 0.4 range = 0.8