Supporting Reliable Data Analysis by Evaluating All Reasonable Analytic Decisions

Yang Liu

A dissertation

submitted in partial fulfillment of the

requirements for the degree of

Doctor of Philosophy

University of Washington

2022

Reading Committee:

Jeffrey Heer, Chair

Christopher Althoff

Katharina Reinecke

Program Authorized to Offer Degree:

Paul G. Allen School of Computer Science & Engineering

University of Washington

Abstract

Supporting Reliable Data Analysis by Evaluating All Reasonable Analytic Decisions

Yang Liu

Chair of the Supervisory Committee:
Jeffrey Heer
Paul G. Allen School of Computer Science & Engineering

Analysts make many, sometimes arbitrary, decisions throughout the data analysis pipeline, yet different choices can lead to divergent conclusions. The flexibility of making analytic decisions can inflate false positive rates and lead to non-replicable findings. In this dissertation, we first characterize how researchers make analytic decisions in their analysis pipeline. We confirm that researchers may experiment with choices in search of desirable results, but also identify other reasons why researchers explore alternatives yet omit findings.

A promising approach to address decision flexibility is multiverse analysis – rather than fixating on a single analytic path, a multiverse analysis evaluates all "reasonable" analytic decisions in parallel and interprets results collectively. We introduce tools and techniques that lower the barriers for analysts to author, run, and interpret multiverse analyses. We present the Boba DSL, a domain-specific language that represents the structure of the decision space, providing critical context for subsequent system components. We introduce the Boba Monitor, a

dashboard that leverages approximation algorithms under the hood to enable monitoring progress and diagnosing issues while the multiverse is still running. We contribute the Boba Visualizer, a visual analysis system that aids users in interpreting the outcomes of all analytic paths, with judicious design choices that push users towards reducing rather than suppressing uncertainty. Finally, we discuss case studies where model quality issues change what one can reasonably take away from the multiverse and justify an iterative workflow. We hope that our findings will help inspire the design of both improved analysis tools and community standards.

Contents

List of Figures

List of Tables

Acknowledgments

First of all, I would like to thank my PhD advisor, Jeffrey Heer, for his consistent support throughout the entirety of my PhD career. Jeff is extremely insightful and provided numerous invaluable comments on the work within and outside this dissertation. He is kind and caring, consistently offering help and encouragement while being sensitive in his constructive criticism. He fostered a harmonious group culture where we treated each other with mutual respect. Most importantly, he supported me through a devastating period with great understanding, patience, and empathy. For that, I am immensely grateful.

I would also like to thank my many incredible collaborators and mentors. I would like to give special thanks to Tim Althoff for his mentorship throughout the last 3 years of my PhD. I am thankful to Kanit "Ham" Wongsuphasawat for guiding me through a qualitative research project, which equipped me with crucial knowledge to finish the interview study later. I am grateful to Alex Kale for his insightful feedback on the design of the Boba system and his contribution to the statistical aspects of the project. I would like to thank Zhicheng "Leo" Liu, Fan Du, and other members of Adobe Research for an invaluable internship experience. I am lucky to have collaborated with Eric Luis Uhlmann, Martin Schweinsberg, Marcel A.L.M. van Assen, Zainab Mohamed, and Robbie C.M. van Aert, Mike Merrill, Ge Zhang, Pierre Dragicevic, Yvonne Jansen, Matthew Kay, Brian Hall, and Fanny Chevalier on projects related to robust data science. Finally, I would like to thank Katharina Reinecke and Jevin West for serving on my dissertation committee, whose insights and critiques are important for strengthening this dissertation.

Besides research collaboration, another important piece of my overall grad school experience was the community. I would therefore like to thank my friends and colleagues at the Interactive Data Lab and the UW Allen School who have provided feedback on my papers, talks, and projects. I especially appreciate the numerous discussions I had with Eunice Jun, which gave me much needed inspiration on my research.

Finally, I would like to thank my friends and family who encouraged and supported me along my PhD journey. In particular, I owe my thanks to Xingfan Huang and Qisheng Li for taking care of me when I was ill.

Thank you!

Chapter 1

Introduction

Producing reliable analysis outcomes is challenging. In a series of "many analysts" studies [8, 46, 92, 93], well-intentioned experts independently analyze the same dataset to answer the same research question, yet they produce largely different analyses and conclusions. The variations in the results are not explained by prior beliefs, expertise, or peer-reviewed quality of the analysis [93].

Why do these well-intentioned experts produce conflicting analytic conclusions? A key contributing factor is the so-called *researcher degrees of freedom* – the flexibility in making analytic decisions. Analysts typically make many decisions throughout the analysis pipeline: Which data values are considered outliers? Which variables should be included as covariates? What model family and parameterization are appropriate? At each decision point, more than one defensible option can exist. Different combinations of choices might lead to diverging results and conflicting conclusions, akin to a *garden of forking paths* [32]. When analysts explore the garden of forking paths and selectively report the best findings, they face the risk of inflated false-positive rates [94]. This practice is believed to be one cause for the replication crisis affecting various scientific fields [3, 32, 33]. To eliminate undisclosed flexibility in decision making, an increasingly popular approach is *pre-registration* [12], where analysts commit all analytic choices to a verifiable registry before collecting any data. However, even by committing to a single analytic path, the conclusion might still be less rigorous, as multiple justifiable paths might exist and different paths might produce diverging conclusions.

One promising approach to address decision flexibility is *multiverse analysis*. Rather than fixate on a single analytic path, analysts of multiverse analyses outline all "reasonable" alternatives a-priori, exhaust all possible combinations between them, execute the end-to-end analysis per combination, and interpret the outcomes collectively [95, 97, 78, 113]. With multiverse analyses, analysts can gauge whether their conclusions are robust to sometimes arbitrary decisions, and identify decisions that have a large impact on results. By reporting the full range of possible outcomes, not just those that fit a particular hypothesis or narrative, the transparency of the study is also improved [88].

However, multiverse analyses are challenging to *author*, *run*, and *interpret*, due to having myriad forking paths. Authoring a multiverse is tedious, as researchers are

no longer dealing with a single analysis, but hundreds of forking paths resulting from possible combinations of analytic decisions. Without proper scaffolding, researchers might resort to multiple, largely redundant analysis scripts [61], or rely on intricate control flow structure including nested for-loops and if-statements. Running a multiverse analysis can introduce long delays before assessing results because a multiverse can lead to a combinatorial explosion of analyses to compute. Interpreting the outcomes of a vast number of analyses is also challenging. Besides gauging the overall robustness of the findings, researchers often seek to understand what decisions are critical in obtaining particular outcomes (*e.g.,* [95, 97, 113]). As multiple decisions might interact, understanding the nuances in how decisions affect robustness will require a comprehensive exploration, suggesting a need for an interactive interface.

This dissertation aims to support analysts in conducting reliable data analysis by evaluating all reasonable analytic decisions. Across the work described in this dissertation, I demonstrate the idea that:

> Thesis Statement
> Multiverse analyses can be made more effective by a specification design that captures critical decision points, an iterative workflow that takes model quality into account, and guidance based on statistical principles that pushes users towards reducing rather than suppressing uncertainty.

There are three important requirements to the design of multiverse software that arise from the thesis statement. First, the multiverse specification should represent the structure of the decision space, as this information is crucial for contextualizing subsequent exploration. Second, the multiverse workflow should be iterative, where users explore the results from *a priori* reasonable paths, determine the reasonableness *post hoc* based on model quality and other considerations, then go back to improve the multiverse specification. Third, software supporting multiverse analysis will benefit from a principled reduction of uncertainty that guide users toward statistical best practices through judicious design choices. We will describe how we design and build a software system to instantiate these processes and requirements.

1.1 Contributions

This dissertation makes the following contributions:

1. **Characterization of researchers' decision-making practices during their analysis process.**
   To better understand how researchers make analytic decisions across phases of data collection, wrangling, modeling, and evaluation, we pore over 9 published research studies and conduct semi-structured interviews with their authors. We characterize recurring rationales for analytic decisions, along with conflicts and implicit trade-offs among options. We observe that researchers often base their decisions on methodological or theoretical concerns, but subject to constraints arising from the data, expertise, or perceived interpretability. We also try to

understand the motivations for exercising the researcher degrees of freedom and selectively reporting results. We confirm that researchers may experiment with choices in search of desirable results, but also identify other reasons why researchers explore alternatives yet omit findings. Based on the interview results, we identify design opportunities for strengthening end-to-end analysis. One major opportunity is a meta-analysis of multiple decision paths via a multiverse analysis. The rest of this dissertation addresses this opportunity by introducing a system for authoring, running, and interpreting multiverse analyses.

2. **The design of a system for authoring, running and interpreting multiverse analyses.** We introduce *Boba*, an integrated system for supporting multiverse analysis. Boba supports a workflow where analysts author the specification, review the results, refine the analysis based on model quality, and commit the final choices in the step of making inference. Boba consists of three components:

   (a) The *Boba DSL* is a domain-specific language (DSL) for *authoring* multiverse analyses. The DSL formally models an analysis decision space, providing critical structure that other system components later leverage. With the DSL, users only need to specify the shared portion of the analysis code once, alongside local variations defining alternative analysis decisions. The compiler enumerates all compatible combinations of decisions and synthesizes individual analysis scripts for each path. As a meta-language, the Boba DSL is agnostic to the underlying programming language of the analysis script (*e.g.,* Python or R), thereby supporting a wide range of data science use cases.

   (b) The *Boba Monitor* is a dashboard that allows users to control multiverse *execution* on the fly, while monitoring progress and diagnosing issues while the multiverse analysis is running. The Boba Monitor leverages sampling-based approximation algorithms under the hood to reduce the time until decision sensitivity is estimated accurately. Using synthetic and real multiverses, we empirically evaluate how quickly the approximation algorithms converge to accurately estimate and rank sensitive decisions. Round robin and sketching approaches are 5 times faster on average compared to no sampling, and up to 2 times faster than uniform sampling.

   (c) The *Boba Visualizer* is a visual analysis system for facilitating the *interpretation* of the output of all analysis paths. The Boba Visualizer first provides linked views of both analysis results and the multiverse decision space to enable a systematic exploration of how decisions do (or do not) impact outcomes. Besides decision sensitivity, we enable users to take into account sampling uncertainty and model fit. We also provide facilities for principled pruning of "unreasonable" specifications, and support inference to assess effect reliability. The Boba Visualizer seeks to guide users towards best practices based on statistical principles, for example promoting visual predictive checks.

3. **The evaluation of the system using three case studies.**
   We evaluate the Boba system through three case studies. The first two case studies replicate multiverses from prior work, whereas the third case study builds upon a crowdsourced data analysis initiative, capturing the various analytic choices made by crowd analysts as a meta-analysis of the results. We show how the Boba Visualizer affords multiverse interpretation, enabling a richer understanding of robustness, decision patterns, and model fit quality via visual inspection. We also demonstrate that by running a small subset of a multiverse using the Boba Monitor, we can arrive at the same conclusion about decision sensitivity and model quality as in the full multiverse. In all case studies, model fit visualizations surface previously overlooked issues and change what one can reasonably take away from these multiverses. We discuss the implications of these issues, along with other challenges in multiverse analysis, in our design reflection.

1.2 Outline

Chapter 2 surveys prior empirical studies and software systems for data analysis. The chapter also discusses recognized pitfalls in data analysis and improved practices, with an emphasis on multiverse analysis.

Chapter 3 describes an interview study for characterizing how researchers make analytic decisions across phases of data collection, wrangling, modeling, and evaluation. The chapter also presents a visualization design for representing analytic decisions in the analysis process.

Chapter 4 presents the design and evaluation of the Boba DSL, which aid users in authoring a multiverse analysis.

Chapter 5 introduces three sampling-based approximation algorithms for estimating decision sensitivity quickly, along with an empirical evaluation of the algorithms. The chapter also presents the design of the Boba Monitor which uses the approximation algorithms under the hood to reduce the latency before assessing results.

Chapter 6 presents the design of the Boba Visualizer, which facilitates users in interpreting the outcomes of all analysis paths.

Chapter 7 describes three case studies to evaluate the Boba system.

Finally, chapter 8 summarizes this dissertation and outlines new research directions for supporting multiverse analyses.

1.3 Prior Publications and Authorship

While I am the primary author of the research in this dissertation, the research is also the result of years of collaboration with my advisor, Jeffrey Heer, as well as my mentors and colleagues at the University of Washington, especially Tim Althoff. The interview study on analytic decision making (Chapter 3) was published at CHI 2020 [66] in collaboration with Jeffrey Heer and Tim Althoff. The Boba system was

published at VIS 2020 [67] in collaboration with Alex Kale, Tim Althoff and Jeffrey Heer. The approximation algorithms underlying the Boba Monitor (Chapter 5) were inspired by discussion with Tim Althoff and Kevin Jamieson. Uri Simonsohn provided a dataset for us to replicate the multiverse about female hurricanes and fatalities (Chapter 7). The case study on crowdsourcing data analysis (Chapter 7) was done in collaboration with Martin Schweinsberg, Eric Luis Uhlmann, Marcel A.L.M. van Assen, Zainab Mohamed, and Robbie C.M. van Aert. Jeffrey Heer provided invaluable feedback and comments on all aspects of the work contained in this dissertation.

To reflect my collaborators' contributions, I will use the first-person plural to describe the work throughout this dissertation.

Chapter 2

Related Work

We draw on prior empirical studies and software systems for data analysis. We also discuss recognized pitfalls in data analysis and improved practices, with an emphasis on multiverse analysis.

2.1 Understanding and Supporting Data Analysis

There are numerous empirical studies on understanding data analysis practices and many software systems for supporting data analysis. This section does not attempt to give a comprehensive overview of the landscape, but instead focus on a much narrower subset of literature that our work builds on.

2.1.1 Empirical Studies of Data Analysis

Many prior studies characterize high-level tasks in the data analysis processes, often using qualitative methods to elicit experiences from analysts. Some focus on specific groups of users or types of activities (*e.g.,* [24, 50, 60]). Others seek to model the data analysis process in general, conceptualizing the pipeline as a sequence of iteratively visited stages [1, 55, 110]. The process model from Wongsuphasawat *et al.* [110], for example, consists of acquisition, wrangling, exploration, modeling, and reporting, with iterative feedback loops between almost every pair of stages (Figure 2-1). In later sections, we refer to this whole data lifecycle as *end-to-end data analysis*.

Some studies examine analytic decision-making in particular [53, 65], a topic that our work in Chapter 3 also focuses on. Our work corroborates Kale's findings on analytic decision-making strategies [53] and Liu's observations on motivations for pursuing alternatives [65]. By richly diagramming our participants' analyses, we further observe recurring patterns in analysis processes, such as feedback loops and fixations. In addition, by closely examining specific, published analyses, we identify conflicts between decision rationales and opportunism. While our study asks participants to recall their decision rationales after the analysis, a subsequent study [92] collects rationales and considered alternatives to executed code blocks while the participants are authoring the analysis script. This later study further examines why analytic

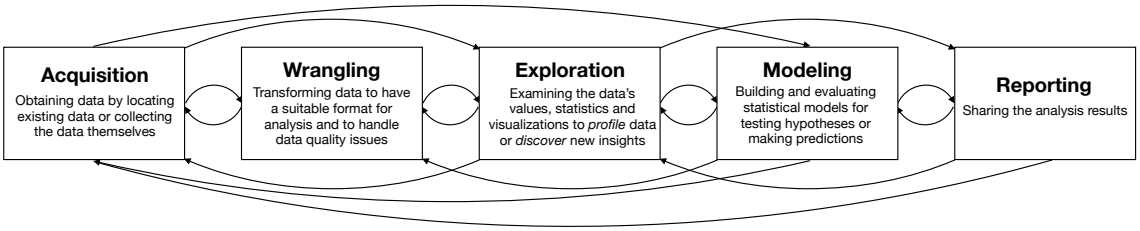


Figure 2-1: The analysis process model from Wongsuphasawat *et al.* [110]. The process model consists of many stages including acquisition, wrangling, exploration, modeling, and reporting, with many iterative feedback loops between stages.

choices contribute to variations in analysis results and identifies various ambiguity in characterizing reasonable decisions.

Across different studies, researchers observe the non-linear nature of the analysis process. A number of studies note the "back-tracking" behavior where analysts revisit an earlier point in their process after meeting a dead end [61, 79, 107]. Related studies characterize the analysis process as a bidirectional search, with both data-focused (bottom-up) and goal-focused (top-down) components [4, 50]. Rule *et al.* [89] describe the tension between experimenting iteratively and explaining insights in a linear narrative, and highlight the need to "support non-linear narrative". Analysts often manage alternatives from exploratory work by duplicating code snippets and files, but these ad-hoc variants can be messy and difficult to keep track of [61, 36]. We also observe many instances of iterative refinement in our interview study (Chapter 3), and our system for supporting multiverse analyses provides a way to author non-linear analyses (Chapter 4).

2.1.2 Software Systems for Data Analysis

Observations in the previous subsection call for tools to support authoring alternative programs. Some systems allow analysts to author multiple variants simultaneously, for example via linked-editing [39] or "forked" interpreter sessions [107]. Another approach cleans up the messy analysis code *after* exploratory programming to produce minimal code for a single analysis path [40]. A related branch of work concerns preserving the provenance of non-linear workflows. Variolite [61] allows analysts to version small chunks of code and easily compare between versions of such local alternatives. While these software systems primarily support ad-hoc exploration of a small number of variants, our system (Chapter 4) allows a more systematic approach to test out all combinations of reasonable alternatives. In future work, inspirations might be taken from approaches for authoring alternative designs. For example, techniques like subjunctive interfaces [68, 69] and Parallel Pies [100] embed and visualize multiple design variants in the same space, and Parallel Pies allows users to edit multiple variants in parallel.

Some systems support statistical analysis in general, often aiming to lower the barriers for less experience users. These systems target various steps along the end-to-

end analysis pipeline, including *a priori* power analysis [106], experimental design [23, 70], data wrangling [54], exploration [111], and statistical modeling [51, 81, 104]. To support less experienced users, a strategy is to represent analysis goals in higher-level abstractions and synthesize appropriate analysis methods from these goals [51]. Our systems aim to lower the barriers for researchers to conduct multiverse analyses, a specific type of statistical approach for increasing robustness and transparency.

2.2 Recognized Pitfalls and Improved Practices

Faulty analyses led to a widespread "replication crisis" [3], with replication studies failing to validate prior results [5, 6, 76, 83]. In Biology, two laboratories ventured to validate published "landmark" studies, but were successful in replicating the original results in only 11% and 25% of projects, respectively [5, 83]. In Psychology, the Open Science Collaboration replicated 100 published studies using high-powered designs and original materials, but found that on average, "replication effects were half the magnitude of original effects" [76].

Replicability concerns have prompted scientists to re-examine how data analysis practices might lead to spurious findings. A simple explanation for why most published studies are underpowered yet yield significant results is "file drawer effect", as studies that fail to find significant results are sent to the file drawer. But scholars argue that in some cases, non-significant results are not *missing*, but masked as significant results. Simmons *et al.* [94] describe how *researcher degrees of freedom* – the flexibility in making analytic decisions – might inflate false-positive rates (*i.e., p-hacking* [75]), allowing one to find significant results in almost any experiment. Machine learning researchers note similar issues, for example tuning random seeds can drastically alter results [42]. Gelman & Loken [32, 33] argue that p-hacking need not be intentional, as *implicit* decisions present similar threats. They use a metaphor of a *garden of forking paths*, with each path potentially leading to different outcomes. Failing to address this flexibility gives rise to issues such as multiple comparison problem (MCP) [25, 26, 114], hypothesizing after the results are known (*HARKing*) [58], and overfitting [84]. As indicated by a survey of 2,000 psychologists [49], p-hacking was unfortunately prevalent. While prior studies base their arguments on simulations and surveys, we conduct semi-structure interviews with researchers to gain a nuanced understanding of potential pitfalls in their analyses.

In response, scholars have endorsed a number of practices, including pre-registration [12, 102, 105], using estimation instead of dichotomous testing [2, 16, 20], adopting Bayesian statistics [30, 57], and increasing transparency in reporting [24, 73, 75, 96]. Preregistration, in particular, is widely advocated in methodological reforms. To preregister, researchers commit all analytic choices to a verifiable registry before collecting any data. The method seeks to eliminate undisclosed flexibility in decision making and promotes "purely confirmatory research". However, some scholars argue that the clear distinction of confirmatory and exploratory analyses is not well-motivated [99, 45]. In addition, fixating on a single analytic path may be less conclusive, as multiple justifiable paths might exist and choosing one path would be

arbitrary. In crowdsourced data analysis studies, well-intentioned experts independently analyze the same dataset to test the same hypothesis, but their analyses and conclusions differ considerably [8, 46, 92, 93]. The variations in the results are not explained by prior beliefs, expertise, or peer-reviewed quality of the analysis [93]. For more comprehensive assessments, researchers have proposed multiverse analysis [97], which we will discuss in detail in the next section.

2.2.1 Multiverse Analysis

Analysts begin a multiverse analysis by identifying reasonable analytic decisions *a-priori* [95, 97, 78]. Prior work defines reasonable decisions as those with firm theoretical and statistical support [95], however, not all decisions might be equally defensible [19]. Decision points might arise in different phases of the analytic process, including data collection [38], wrangling [97], and modeling [95]. While general guidelines such as a decision checklist [108] exist, defining what decisions are reasonable still involves a high degree of researcher subjectivity.

The next step in multiverse analyses is to exhaust all compatible decision combinations and execute the analysis variants (we call a variant a *universe*). Despite the growing interest in performing multiverse analysis (*e.g.,* [48, 77, 13, 6, 87]), few tools currently exist to aid authoring. Young and Holsteen [113] developed a STATA module that simplifies multimodel analysis into a single command, but it only works for simple variable substitution. *Rdfanalysis* [27], an R package, supports more complex alternative scenarios beyond simple value substitution, but the architecture assumes a linear sequential relationship between decisions. *Multiverse* [90] is another R package that allows analysts to iteratively author multiverse analyses in their preferred workflow inside RMarkdown computational notebooks. It handles procedural dependencies between decisions, allowing analysts to more easily exclude unreasonable paths. Our DSL (Chapter 4) similarly provides scaffolding for specifying a multiverse, but it extends to other languages beyond R.

After running all universes, the next task is to interpret results collectively (see [37] for a survey). Some prior studies visualize results from individual universes by either juxtaposition [95, 97, 85] or animation [21]. Visualizations in other studies apply aggregation [18, 82], for example showing a histogram of effect sizes. The primary issue with juxtaposing or animating individual outcomes is that they cannot scale beyond hundreds of universes, though this might be circumvented by sampling [95]. Our visualizer (Chapter 6) shows individual outcomes, but overlays or aggregates outcomes in larger multiverses to provide scalability.

Besides the overall robustness, many studies also investigate which analytic decisions are most consequential. The simplest approach is a table [97, 11, 85, 17] where rows and columns map to decisions, and cells represents outcomes from individual universes. Simonsohn et al. [95] extend this idea, visualizing the decision space as a matrix beneath a plot of sorted effect sizes. These solutions might not scale beyond hundreds of universes as they juxtapose individual outcomes, and the patterns of how outcomes vary might be difficult to identify depending on the spatial arrangements of rows and columns. Another approach [82] slices the aggregated distribution of out-

comes along a decision dimension to create a trellis plot (*a.k.a.* small multiples [101]). The trellis plot shows how results vary given a decision, but does not convey what decisions are prominent given certain results. Our visualizer uses trellis plots and supplements it with brushing to show how decisions contribute to particular results.

Finally, prior work relies on various strategies to infer whether a hypothesized effect occurs given a multiverse. The simplest approach is counting the fraction of universes having a significant p-value [97, 11] and/or an effect with the same sign [18]. Young and Holsteen [113] calculate a robustness ratio analogous to the t-statistic. Simonsohn et al. [95] compare the actual multiverse results to a null distribution obtained from randomly shuffling the variable of interest. We build upon Simonsohn's approach and use weighted model averaging based on model fit quality [112] to aggregate uncertainty across universes.

Chapter 3

Decision Points & Selective Reporting in End-to-End Data Analysis: An Interview Study

As discussed in Chapter 1, drawing reliable inferences from data involves many, sometimes arbitrary, decisions. As different choices can lead to diverging conclusions, understanding how researchers make analytic decisions is important for supporting robust and replicable analysis. In this chapter, we investigate decision making within *end-to-end quantitative analysis*: the full lifecycle of quantitative data analysis including phases of data collection, wrangling, modeling, and evaluation. We conduct semi-structured interviews with authors of nine published studies in HCI and other scientific domains. We pore over participants' manuscripts and analysis scripts to assess their decisions, and ask them to recall, brainstorm, and compare alternatives in every analytic step.

In this chapter, we contribute the results and analysis of these interviews. We present a visualization design for representing analytical decisions, both to communicate our interview results and as a tool for mapping future studies. We identify recurring rationales for analytic decisions, highlighting conflicts and implicit trade-offs among options. Next we examine the motivations for carrying out alternative analyses, a practice that exercises freedom in analytic decisions. We subsequently discuss how participants choose what to include in research reports if they have explored multiple paths. Finally, based on our observations, we identify design opportunities for strengthening end-to-end analysis, for instance via tracking and meta-analysis of multiple decision paths. Given the HCI community's demonstrated interest in quantitative empirical research, we hope our findings will help inspire the design of both improved analysis tools and community standards.

3.1 Methods

To better understand decision-making in end-to-end quantitative data analysis, we conducted semi-structured interviews with authors of nine published studies. We

Theme	Category	Description	Representative Quote	%
Decision rationales	Methodology	Participants defend the decision with methodological concerns, including statistical validity, study design and research scope.	I mainly used t-test for hypothesis testing because my data was parametric.	25
	Prior work	Participants support the analytic decision using previous studies, "standard practice" and/or internalized knowledge.	We adapt the method from a previous paper and we follow the same process to do the analysis.	33
	Data	Participants mention data constraints, including data availability, data size and data quality.	The reason I combined them together is because more data has less variation.	21
	Expertise	Participants feel limited by expertise.	I don't know how to do this really.	12
	Communication	Participants prefer an alternative that is easier to communicate.	Because they were actually very hard to write up.	7
	Sensitivity	Participants believe that the decision has little impact on the results and provide no further rationales.	In my quick mental calculation, it seemed like it wouldn't actually make a big difference.	3
Executing alternatives	Opportunism	Participants willingly explore new alternatives to look for desired results.	I tried three different settings for those parameters and the chosen ones looked slightly better.	45
	Systematicity	Participants outline all reasonable alternatives, implement them, and choose the winning alternative based on an objective metric.	We performed a sensitivity analysis to identify the best combination.	9
	Robustness	Participants implement additional alternatives after making a decision, in order to gauge the robustness of their conclusions.	That is just for robustness, to say, "even if you look at [another option], you see the same thing."	16
	Contingency	Participants have to deviate from their original plans because the planned analysis turned out to be erroneous and/or infeasible.	This [filter] produced anomalous results and we went back [to apply] a more stringent filtering.	30
Selective reporting	Desired results	Participants only report the desired results and omit findings that are non-significant, uninteresting, or incoherent to their theory.	It felt stronger to say five out of seven, rather than four out of six, was one reason to keep it.	29
	Similar results	Participants claim that the results are similar and thus omit interchangeable alternative analyses.	But it didn't make a huge difference so we just kind of went with [the current option].	10
	Correctness	Participants apply rationales, primarily methodology and prior work, to remove analytic approaches they consider incorrect.	I was concerned about whether I had a strong hypothesis to see those interaction effects or not.	31
	Social constraints	Social constraints and communication concerns prevent participants from reporting some findings.	I'm a second author and many decisions made in the manuscript writing were against my wishes.	31

Table 3.1: Themes and categories that emerged from open coding of the interview data. The rightmost column lists the prevalence of a category within a theme.

first inspected the papers and analysis scripts, then engaged researchers in discussion about their decision rationales and possible alternatives.

3.1.1 Participants

We interviewed 9 academic scientists (3 females, 6 males, age 24–72), including 6 Ph.D. students, 2 research scientists and 1 tenured professor. Our interviewee's research fields include Human-Computer Interaction (5), Proteomics (2), Marine Biology (1), and Geography (1). Participants' analyses cover a spectrum from directed question-answering to open-ended exploration. P1-5 conducted confirmatory analyses: they designed controlled experiments to answer predefined research questions. P6 explored their data to develop a biological assay. P7 and P8 performed exploratory data analyses (EDA). P9 gathered insights from EDA to form a hypothesis for a subsequent confirmatory experiment.

We recruited interviewees by advertising in multiple HCI and data science mailing lists. We also identified 15 local authors from the CHI 2018 proceedings and

emailed them directly, netting three participants. Regardless of recruitment method, all interested participants filled out a survey to provide a publication and the accompanying analysis scripts. We recruited every respondent whose publication involved quantitative data analysis and had been published in a peer-reviewed venue.

3.1.2 Interview Procedure

We interviewed one researcher at a time for 60–90 minutes. We began each interview with an introduction describing the purpose of the interview: to understand decision making during data analysis and to collect use case examples for developing prototype tools for robust data analysis. We then proceeded with our discussion protocol, which consisted of three phases. The discussion focused specifically on the analysis project provided to us by the participant in the signup survey. Afterwards, all participants were compensated with a $20.00 gift card.

Phase 1: Recall. We first asked participants to freely propose different, yet justifiable analytic decisions. We encouraged participants to recall alternative paths they had considered and executed, and those raised by reviewers. We did this prior to other phases to elicit responses without biasing participants.

Phase 2: Brainstorm. We asked participants to brainstorm alternatives using a checklist based loosely on the work of Wicherts *et al.* [108]. The checklist contains common analytic decisions across stages of a typical data analysis pipeline, from data collection and wrangling to modeling and inference. We used the checklist to help participants systematically examine all steps in the end-to-end pipeline.

Phase 3: Compare. To raise options overlooked by participants in the previous phases, we discussed additional decisions we had prepared before the interview. We generated alternative analytic proposals by perusing the paper, appendix, and analysis scripts, while consulting the checklist to ensure a comprehensive coverage of different phases of the analysis.

3.1.3 Analysis of Interview Data

All interviews were audio recorded and transcribed verbatim. The first author analyzed the data, with iterative feedback from other authors throughout the analysis process. As our findings might put participants in a vulnerable position, we have replaced identifiable information in figures and quotes. For example, we might replace an identifiable variable name (*autophagy substrate*) with a generic name (IV).

We first sought to understand the overall analysis process. From the interview data, we extracted analytic steps and their relationships to re-construct both decision points and data flow. We drew graphs to aid interpretation, and soon realized that the graphs had greater utility beyond summarizing interview results. We thus conducted a dedicated design exercise by outlining design goals, iterating over visual encodings, and producing visualizations, as detailed in the next section.

Next we investigated how participants made analytic decisions. We began by using open coding [14] as a preliminary step to identify recurring themes. Three themes emerged: participants provided *rationales* for decisions, described their experiences in

executing alternative analyses and subsequently *selective reporting* of the results. We integrated raw codes within each theme to extract common concepts and patterns. Table 3.1 summarizes the themes and categories, along with example quotes (the quotes were edited for brevity and clarity; full quotes and relevant contexts are in later sections). The table also lists the prevalence of each category, computed as the ratio of unique instances within each theme. We discuss our empirical findings in the section *Interview Results*.

3.1.4 Limitations

One limitation is our convenience sampling approach, which introduces potential bias. For example, our sample is mostly composed of HCI and junior researchers. To be clear, our research goal is to characterize the space of analytic processes and decisions, *not* to quantify the prevalence of any specific activity. Also, while our study reaches saturation in some regards [34] as the last two participants did not surface new categories, our convenience sample might miss known phenomena. Some practices currently gaining adherents, such as pre-registration followed by exploratory analysis on collected data and planned analysis based on simulated data, are not observed. A future taxonomy might better delineate the distinction between a-priori and a-posteriori decisions.

We note violations of methodological validity when perusing participants' analyses to flag potentially problematic practices, but our judgments are subjective. Some methods we endorse are not universally accepted, such as multiple comparison correction [15]. Other than methodological validity, we interpret from the perspectives of the participants as much as we could.

Participants might withhold information on potentially problematic practices. Where possible, we complement the transcripts with what we found from participants' analysis scripts (*e.g.,* evidence of implementing multiple model formulae in R code), but not all scripts retain the full history. Thus, there are likely additional explorations of alternatives that we are unable to observe. In addition, all accounts of analytic decisions were given post-hoc. Future studies are needed to inspect researchers' decision making process during the analysis event.

3.2 Analytic Decision Graphs

To represent participants' process and decisions, we created visualizations that we call *Analytic Decision Graphs* (ADGs). We developed ADGs in conjunction with our analysis of the transcribed interviews. We present the design of ADGs here first, so that we can refer to them in our later discussions.

3.2.1 Design Goals

ADGs aim to visualize analytic decisions in the context of end-to-end analysis pipelines. We expect ADGs to afford two utilities. First, with ADGs as visual illustrations,

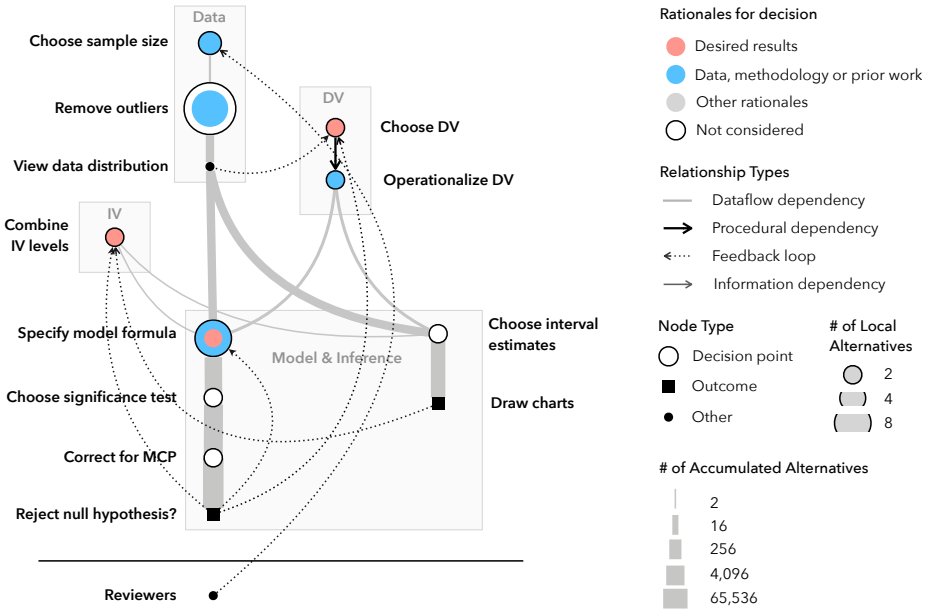


Figure 3-1: Analytic Decision Graph for P1, representing a controlled experiment to investigate the impact of web design on reading performance. At several steps, P1 revised her analytic decisions based on end results and reviewer feedback, for instance merging two levels of an IV because effect sizes were similar. While she examined model specification options thoroughly, she appeared to place less emphasis on inference decisions such as choosing which significance test to use.

authors should be able to communicate their decisions and processes more easily. Second, ADGs might prompt reflection on decisions, potentially encouraging consideration of further alternatives.

To review an analysis decision process, users will need to perform at least the following tasks:

- Gain an *overview* of the high-level analytic components.
- Understand the analytic *steps* and their relationships.
- Examine and evaluate the *decisions* made in each step.

From these tasks we can distill some design requirements:

- *Represent the input and the outcomes.* To provide context, ADGs should include inputs such as data sources and outcomes such as deliverables supporting the conclusion.
- *Display granularity of analysis components.* ADGs should visualize both high-level modules and individual decisions.
- *Represent relationships between the steps.* ADGs should capture various types of relationships, such as order and dependency, to organize steps into a coherent process.
- *Visualize the rationales and the ramifications of a decision.* Visualizing rationales might help authors identify weak spots and help readers gauge the validity.

3.2.2 Visual Encodings

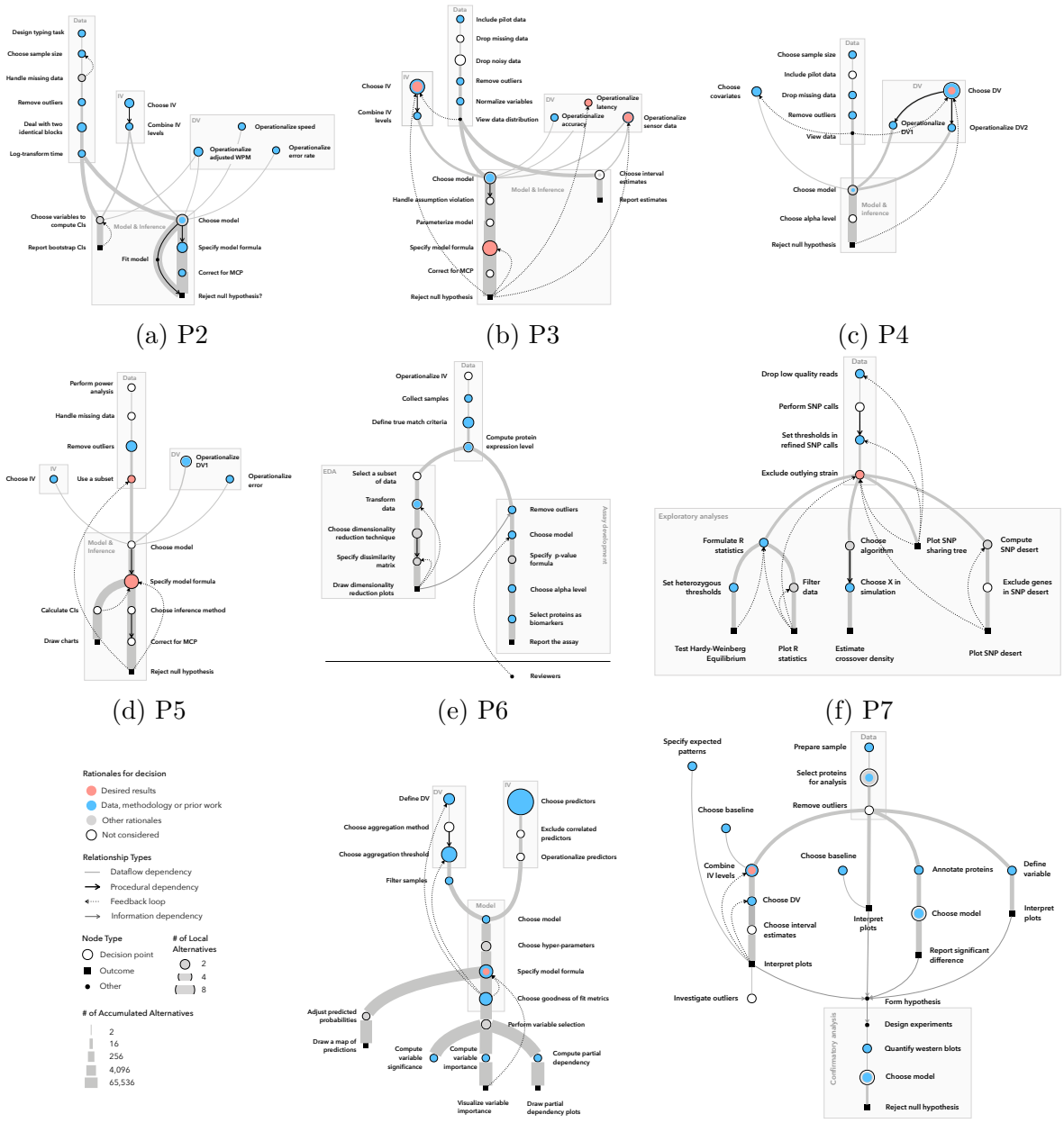
To meet these requirements we iterated over several designs. We discuss the tradeoffs made and present the final design.

As ADGs should visualize both steps and their relationships, a graph is a natural representation. We use a *node* (○) to encode a *decision point* and an *edge* to encode the *relationship* between two decision points. We further include auxiliary nodes with distinct shapes: rectangles (■) represent analysis outcomes, whereas solid dots (●) are "dummy" nodes. In an earlier design, we visualized all potential alternative choices one could make in addition to the decision point, but the graph soon grew cluttered. We thus omit individual alternatives.

Various types of relationships exist between two decision points. The first type is a *dataflow dependency* (——), where the output of one node is the input to another. The second type is a *procedural dependency* (⟶), where the downstream decision would not exist if some alternative in the upstream decision were chosen. For example, if a researcher had chosen a frequentist model instead of a Bayesian model, she would not need to decide among different priors. The third type is an *information dependency* (⟶), where one decision informs another. For example, insights from exploratory analysis might inform the hypothesis of a subsequent confirmatory experiment. We also have *feedback loops* (⇠), as researchers revise an upstream decision based on the results from a downstream step. All of these relationships appear as edges of different textures. We further arrange the nodes vertically according to their order in the dataflow, with the top being the start. Yet another type of relationship exists – *temporal order* – as some decisions are made earlier than others. We overload the vertical axis to represent temporal order when it does not conflict with dataflow dependency.

We use a categorical color palette to represent type of decision rationale. To reduce visual complexity, we simplify the categories of Table 3.1 to three groups. We use a red color for *desired results* (●) to call out potentially problematic practice; this is when researchers made the decision by weighing end results, for example discarding options that produced non-significant results. We assign blue to *data, methodology, and prior work* (●), which are relatively primary concerns. The rest of the rationales, denoted *other rationales* (●), receive a desaturated gray color. Finally, as we (the interviewers) might propose alternatives that the participant had not thought of, we use white (○) to indicate additional decisions not considered by the participants at the time of analysis.

The size of a node corresponds to the number of enumerated alternatives for the decision point. The thickness of a dataflow edge conveys the number of accumulated alternatives, namely all possible combinations of alternatives of previous decisions leading to that point. Since the accumulated total grows exponentially, we use a logarithmic scale for edge thickness.



(a) P2 (b) P3 (c) P4

(d) P5 (e) P6 (f) P7

(g) Legends (h) P8 (i) P9

Figure 3-2: Analytic Decision Graphs for P2–P9.

3.3 Analysis of Analytic Decision Graphs

We created an ADG for each participant, as shown in Figure 3-2. We first describe P1's ADG (Figure 3-1) in detail, then summarize recurring patterns drawn from the ADGs for all participants. As comparing unrelated studies is not a design goal of ADGs, care should be taken when interpreting apparent differences between graphs. Some visual properties (*e.g.,* those described below) are meaningful to compare, but other visual differences (*e.g.,* horizontal position of nodes, edge curvature) are not.

3.3.1 ADG Walkthrough for P1

P1 designed a controlled experiment to investigate the impact of web design on reading performance. She followed a typical confirmatory pipeline: she operationalized (*i.e.,* defined the measurements of) the variables germane to her research questions, collected and processed the data, built a statistical model, and interpreted the results, ultimately producing a bar chart of effect sizes with uncertainty intervals and several p-values.

The dataflow edges funnel into two linear paths leading to the end results, as opposed to a typical exploratory analysis (*e.g.,* Figure 3-2f) where the dataflow forks into multiple branches. Still, P1's analysis has many feedback loops: P1 revised her analytic decisions at several steps, based on observed data, end results, and reviewer feedback. Despite being a relatively simple pipeline with 9 decision points, P1's analysis gives rise to over 5,000 possible ways to compute the final p-values, as indicated by the width of the dataflow edge into the final node *reject null hypothesis*. Judging by the size and color of decision nodes, P1 examined model specification options thoroughly (indicated by the size of the *specify model formula* node), but she appeared to place less emphasis on *inference* decisions (indicated by empty nodes in the inference section).

3.3.2 Summary of ADG Patterns

Using the interpretation approach above, we analyzed ADGs for all participants. Here are a few recurring observations.

Feedback loops are present in all analysis processes of our participants, regardless of whether the analysis is confirmatory or exploratory (Figure 3-2, dotted edges). We further examine these iterative fine-tuning behaviors in the next section.

Participants often fixate on a few prominent steps while ignoring decisions in the end-to-end pipeline. Among our participants, we observe that data and inference decisions are often neglected (Figure 3-2, empty nodes). When prompted by the interview checklist or the interviewer, participants revealed that they did not recognize these steps as decision points and implicitly chose a single viable option. On the other hand, *choosing variables*, *choosing models* and *specifying model formula* are often considered thoroughly (Figure 3-2, large nodes).

Procedural branches are rare among our participants. P1's process includes one procedural edge and no procedural branches (Figure 3-1, thick black edges); she could

have considered ways to operationalize other candidate dependent variables. The lack of such branches implies a relatively linear process where decisions were made in order, one step at a time.

Across participants, the "multiverse" size ranges from 16 to over 25,000,000 (median 1,632; see Figure 3-2, thickness of dataflow edges immediately before rectangular nodes). We revisit issues related to scale in the discussion section.

3.4 Interview Results

We now describe the patterns that emerged from the qualitative analysis of our interview data, following the organization of themes and categories in Table 3.1.

3.4.1 Rationales for Analytic Decisions

When participants recognized an analytic step as a decision point, they might *reason* about it, identifying and evaluating options before selecting a path along which to proceed. From 190 such instances, we identified six categories of rationales for analytic decision making.

Methodology

Methodological concerns comprised a major set of rationales (48 instances, 9/9 participants). These arguments typically involved statistical validity, study design, and research scope.

Many methodological concerns (19 instances, 9/9 participants) were rooted in statistical validity. Meeting model assumptions was a concern for seven participants, as they chose the statistical model best suited for the data distribution, or wrangled the inputs to satisfy model assumptions. Participants used various strategies for the later approach: they might balance the datasets, normalize the inputs, log-transform a variable, or remove collinear variables. Besides model assumptions, five participants supported their decision with logical arguments, pointing out mathematical properties or explaining the intuitions behind customized methods. As a simpler example, P6 explained why she used a less common log-transformation, $log(x + 1)$, to process the data: "*because I have a lot of zeros.*" (Figure 3-2e, transform data).

Validity concerns also stem from study design (21 instances, 9/9 participants). Five participants argued that confounders were controlled for and thus were excluded in model specifications. Four participants stressed that variables in their models strictly followed the factors, levels, and measures in their experimental design. Participants followed a preselected plan akin to pre-registration [12, 102, 105], though none of the studies was officially pre-registered.

Other than validity, a few rationales (8 instances, 2/9 participants) are rooted in scope, as researchers discarded alternatives outside the scope of their current research questions. As P2 argued (Figure 3-2a, design typing task): "*the intention of this research is to evaluate text entry, the real-life text entry. So we'll not type random text.*"

Prior Work

Another group of rationales were anchored in prior work (62 instances, 9/9 participants). Here, we use *prior work* to refer to prior studies, standard practices, and internalized knowledge.

All participants cited prior studies to support their decisions. Besides utilizing knowledge from prior studies to inform decisions, three participants mimicked configurations from a prior work. While this enables direct comparison with previous findings, participants might admit that alternatives warranting further considerations might exist. For example, P8 stated (Figure 3-2h, choose goodness of fit metrics):

> *"...[the chosen method] is what multiple other papers have used. But there would be alternatives and we have a whole host of other model performance metrics."*

Without citing specific sources, seven participants drew on knowledge that likely resulted from a combination of prior studies, consensus, and training. A participant referred to field consensus in outlier removal: "*We did not remove any outliers. Because in the autism field, why it's called Autism Spectrum Disorder, because other Autism are considered outliers.*"

Six participants in 22 instances honored "*standard practice*", "*tradition*", and "*convention*", sometimes without questioning its validity. For instance, a participant followed a "*rule-of-thumb*" of recruiting ∼20 participants for an experiment, though the study might be under-powered and so fail to resolve effects of smaller size (Figure 3-2a, choose sample size). A participant chose to "*start with a t-test, because it's standard*" (Figure 3-2b, choose model), though the data violated normality assumptions. Two participants admitted that standard practices might not be best practices, but they were concerned about social aspects. They believed that readers would accept standard practice more readily and "*reviewers would have asked for it.*"

Five participants expressed how the lack of theory prevented them from choosing statistically valid alternatives. Two participants avoided interaction patterns that they "*didn't have a strong hypothesis to include*" (Figure 3-2d, specify model formula). P2 explained how tweaking alpha, a parameter in a metric to operationalize a variable, might allow one to obtain desirable outcomes, and argued against such practices because "*there's no reasonable theory or rationale underlying that alpha.*"(Figure 3-2a, operationalize adjusted WPM).

Data

Data constraints represented another major group of rationales (39 instances, 8/9 participants). Researchers were constrained by data availability, quality, and size.

Some data constraints were hard constraints. Unavailable data might prevent participants from investigating additional variables. P8 originally identified 23 relevant predictors from prior work, but later dropped 7 of them for which he was unable to obtain sufficient data (Figure 3-2h, choose predictors). Three participants stated that collecting more data was too costly or infeasible, as P4 complained: "*we set the target beforehand, but we couldn't achieve the target group. We just tried to recruit*

as many groups as possible." (Figure 3-2c, choose sample size). Sticking with a small sample size, two participants noted that certain modeling approaches, for instance time series analysis, were infeasible.

On the other hand, some data issues allowed more room for flexibility. What constituted clean data might be subjective, but three participants excluded noisy data at the expense of study design, for instance dropping an entire variable. Similarly, three participants altered study designs, such as pooling variable levels, in order to achieve a larger sample size.

Expertise

Researchers also felt limited by expertise (23 instances, 8/9 participants). They might not know what alternatives were possible, as P9 commented (Figure 3-2i, choose model):

> *"And there is almost certainly some other way to do that, but I'm not sure that I would know what it is."*

When researchers had a rough notion of viable alternatives, they opted not to pursue an unfamiliar method. P4 echoed sentiments of three participants about Bayesian analysis: "*I heard something about Bayesian statistics, but I don't have any background to try more than that.*" (Figure 3-2c, choose model). Two researchers deferred a decision to a statistician, who they believed had better authority over the subject.

Communication

Sometimes researchers preferred an alternative that was easier to communicate (13 instances, 6/9 participants), quoting a variety of values. Two participants preferred an "*interpretable*" method over "*methods that merely produce black-box predictions*" (Figure 3-2h, choose model). A participant attempted to be "*consistent*" with the methods he used, because "*otherwise the readers will be confused*" (Figure 3-2b, choose model). Another participant aimed for higher *generalizability* by targeting for practical use cases (Figure 3-2a, design typing task). Finally, a participant just wanted to keep things *simple*, avoiding "*more complex*" options (Figure 3-2a, choose IV). Communication concerns can come at the expense of validity. P3 chose a statistical model suboptimal for their data distribution because "*to make the analysis consistent across the whole study, we just stick with one statistical test.*" (Figure 3-2b, choose model).

Sensitivity

Finally, researchers sometimes claimed that choosing another alternative would have little impact on the results (5 instances, 4/9 participants). Two researchers supported the claim with logic. P8 said: "*in my quick mental calculation, it seemed like it wouldn't actually make a big difference.*" (Figure 3-2h, adjust predicted probabilities). Others recalled from past experience that two methods tended to produce similar results. As they did not evaluate their current situation, perceived sensitivity might differ from actual sensitivity.

3.4.2 Interactions of Rationales

We observed an interplay between decision rationales, particularly in terms of which rationales tended to dominate others. Both our own interviews and previous studies [95, 102] identify *methodology* and *prior work* as dominant rationales that researchers primarily rely upon. A bottom-up, exploratory approach might include *data* as a dominant rationale category, as researchers develop tentative theories to account for observed phenomena. However, in practical situations, the analysis plan supported by the dominant rationales nevertheless accommodates various constraints concerning *data*, *expertise* and *communication*. *Sensitivity* ignores other rationales by focusing instead on the impacts of the decision.

These decision rationales interact with each other, often creating conflicts. The previous section described two ways in which the dominant categories, *methodology* and *prior work*, are contradictory. First, standard practices are not always best practices; by adhering to conventions, participants might adopt a statistically faulty method. Second, a statistically valid approach might lack theoretical support, as five participants described how they avoided such situations. The previous section also contains ample evidence of how secondary rationales constrain and override dominant concerns. *Data*, *expertise* and *communication* all limit the viable methods researchers choose from, as researchers prefer a method that is familiar, easy to communicate, and feasible for the current data size. *Data*-related issues also impact study design, for instance researchers might drop a noisy variable or combine multiple levels within a variable to increase sample size.

3.4.3 Motivations for Executing Alternative Analyses

While some researchers *reasoned* about alternatives, ruled out options, and implemented a single final decision, others *executed* alternative analyses. What spurred researchers to actualize possibilities and travel multiple analytic paths? We found 44 instances in which participants explicitly described, or we could reasonably infer, their motivations to pursue alternatives. We then identified four categories of motivations.

Opportunism

When being opportunistic, researchers willingly explored new alternatives, searching for desired results in the garden of forking paths (20 instances, 7/9 participants). Such exploratory behavior comes in two forms: one might search for patterns without a hypothesis to defend, or one might actively search for a confirmation of existing hypothesis. The first form is sensible as long as the exploratory nature is clearly acknowledged in the publications [102, 105]. In fact, exploratory data analysis (EDA) literature often advocates an open mindset and a comprehensive exploration before focusing on pre-defined questions [1, 4]. Participants doing EDA all demonstrated an opportunistic attitude, as P8 described:

> *"It was like a little experiment . . . It wasn't to test any hypothesis, but it was to explore the data in a more complete way where we could actually investigate the*

effects that we were interested in."

However, we also observed opportunism among participants who reported strictly confirmatory findings (Figure 3-1 & 3-2a-d, feedback loops into red nodes). Participants tried multiple analytic options and selected a path leading to desired results. Such endeavors might happen in the data wrangling phase, when participants qualitatively explored data distributions and avoided analytic options unlikely to produce desired outcomes. P1 discarded a dependent variable because it failed to yield differential results across conditions (Figure 3-1, choose DV):

> *"The distributions of accuracy are similar across questions. So, instead of looking at how different conditions affect it, we use [accuracy] as another exclusion criteria."*

Others adopted a deliberate and structured search. P3 tried "*all the different combinations*" of independent variables in a model specification (Figure 3-2b, specify model formula):

> *"You can think of it as a cross product, we did all of them, right? ... we have ANOVA to test the difference of accuracy with and without considering age, and with and without considering gender, and with considering both gender and age. We did all of them."*

After an exhaustive search for patterns, he selectively reported "*interesting findings.*" These examples of opportunism in confirmatory analysis might increase the chance of false discovery and lead to non-replicable conclusions [32, 75, 94].

Systematicity

Voluntary exploration was not always driven by a desire to find interesting results. Researchers could *systematically* enumerate reasonable alternatives, implement them, and evaluate the outcomes based on an objective metric (4 instances, 3/9 participants). The key evidence to help us distinguish *systematicity* from *opportunism* was that the evaluation metric did not hinge on anticipated conclusions; the metric was not the end result. Two participants enumerated model specifications and chose the best one based on the goodness of fit. P8 also ran a local multiverse analysis and used the goodness of fit to choose the best combination of two decisions (Figure 3-2h).

Robustness

In another type of voluntary exploration, researchers tested alternatives *after* making a decision to gauge the *robustness* of the outcomes (7 instances, 4/9 participants). After the model yielded expected results, P2 implemented two redundant tests "*to gain an inner confidence of the metric*" (Figure 3-2a, choose model). P9 applied two protein annotation methods to corroborate the same conclusion (Figure 3-2i, annotate proteins):

> *"That is just for robustness, to say, 'Hey, even if you look at orthologs of proteins that in mammals and so on are EV proteins, you see the same thing."'*

Contingency

In the case of contingency, researchers had no choice. They had to deviate from their original plans because the planned path proved to be erroneous or infeasible (13 instances, 5/9 participants). Contingency might arise internally, as five participants ran into a dead end and retracted to an upstream analytic step. At a filtering step, P7 initially set loose thresholds because "*having more data was probably better*", but the decision backfired (Figure 3-2f, drop low quality reads):

> *"But two years into the project, it was realized that this [filter] produced some very anomalous results, and we went back, and for some of the subsequent analysis we went through a more stringent filtering of the data which removed some of these anomalies."*

External contingency came from reviewers, who urged researchers to revise the analysis. P6 switched to a Fisher's exact test from a t-test: "*well, the reviewer made me do it, but I'm not sure it's the best choice.*" (Figure 3-2e, choose model).

3.4.4 Motivations for Selective Reporting

After researchers executed alternative analytic paths and observed multiple outcomes, they must choose which analyses to include in publications. We observed 52 instances in which researchers did not report all analytic paths taken. Why did researchers report some findings but omit others? We identified four categories of motivations underlying selective reporting.

Desired Results

Evaluating multiple options allowed researchers to view and weigh the outcomes. Unsurprisingly, the quality of the outcome was a major criterion in selecting which alternative to report. In opportunistic exploration, researchers searched the garden of forking paths for desired results; consequently, they typically only reported the desired results and omitted findings that were non-significant, uninteresting, or incoherent to the theory they intended to support (15 instances, 7/9 participants).

A majority of participants conducting confirmatory analysis (4/5) omitted statistically non-significant results. When multiple results proved significant, participants selected the option with stronger implications for their intended theory. P5 tested two ways to filter the data and both produced significant results, so she chose the larger subset such that she could argue for a greater impact of the proposed mechanism (Figure 3-2d, use a subset). Two participants included non-significant results and devised further criteria for "interesting" findings worthy of reporting. To P3, interesting findings meant all significant results plus unexpected null results "*which we thought it might be significant but it turns out not.*" He truthfully documented initially plausible hypotheses that failed an empirical test, yet his reporting strategy also includes any hypothesis that seemed plausible post-hoc – which is a form of HARKing [58].

Two participants conducting EDA also omitted explored analysis paths that did not corroborate the conclusions. Only one participant comprehensively documented alternative analyses they had performed during exploration.

Similar Results

In a few cases (5 instances, 3/9 participants), researchers relied on analytic outcomes, but argued that the outcomes were similar in terms of both the actual results and their implications. Thus, reporting one of the alternatives was deemed sufficient. Participants did not elaborate any criteria for selecting among similar options, implying that sensitivity alone was the reason for suppressing interchangeable analysis alternatives.

Correctness

Despite having access to the analytic outcomes, sometimes participants did not utilize this information. Instead, they fell back to using rationales described in the *decision rationales* theme, most frequently *methodology* and *prior work*, to remove analytic approaches they considered incorrect (16 instances, 7/9 participants). Such practices might ensue from an exploration out of contingency or robustness. For example, researchers switched to an alternative method requested by reviewers, omitting the original, presumably flawed, method. However, sometimes the motivation for exploring alternatives was unclear and we do not know whether the correctness argument was formed before or after seeing the results. The latter scenario, namely coming up with post-hoc explanations for *desired results*, is precisely HARKing [58].

Social Constraints

Finally, social constraints could prevent participants from reporting certain findings (16 instances, 7/9 participants). Colleagues and reviewers might disapprove of particular analysis methods. P2 did not report his experimental code on Bayesian analysis because his "*colleagues don't seem to favor that*" (Figure 3-2a, choose model). P8 similarly complained that he did not have full control over reporting:

> *"I'm a second author and many decisions made in the publication, in the manuscript writing, and figure making were decisions against my wishes."*

Two participants mentioned that reporting every detail would exceed the page limit. In response, P3 deleted the alternative taking up more space and P2 removed a finding perceived by the authors to be "*not of interest.*"

Researchers might voluntarily cater to communicative concerns to make figures and manuscripts easier to understand. Two participants applied additional filtering to a visualization to reduce over-plotting; they omitted the original plot and parameters. Another two participants removed analysis methods unfamiliar to the audience. P2 stated that describing Bayesian analysis in an accessible way would be too much work, and P9 simply claimed that a method would confuse readers.

3.5 Design Opportunity

Based on the interview results, we identify design opportunities for supporting users in making and communicating analytic decisions.

3.5.1 Analysis Diagramming & Provenance Tracking

In many instances our respondents were limited in coming up with alternatives: they might fail to recognize an analytic step as a decision (*e.g.,* following default settings), adopt a single option without considering alternatives (*e.g.,* making the same decision as a previous study), or overlook possible alternatives due to *expertise*. A corresponding avenue for future research concerns analysis *linters* or *recommenders*, in which tools flag potentially problematic practices (such as the feedback loops observed in our interviews), recommend alternative methods, or even automatically suggest a preferred method based on statistical validity [9, 51, 104]. One strategy for such tools is to enable higher-level specifications of analysis goals (*e.g.,* specifying annotated model inputs and outputs rather than explicit test types or formulae), from which appropriate analysis methods might be synthesized in conjunction with the data [51]. Another strategy is to leverage the abundance of online analysis code [89] to mine patterns of decisions and alternatives, which might be useful for building automatic recommenders.

In some cases, our respondents evaluated multiple alternatives and then engaged in selective reporting. Integrating diagramming methods with provenance tracking could provide some level of automated documentation, for example by analyzing executed code paths to model and visualize the various alternatives that were explored (*c.f.* [59]). Similar elicitation and tracking strategies have also been suggested for reducing false discovery during exploratory visualization [114].

Even with complete documentation of analysis history, hindsight bias might lead researchers to unintentionally misremember *post hoc* explanations developed after conducting analysis as motivating *a priori* hypotheses [58]. Tools for mapping analysis decisions might promote more comprehensive assessment *a priori*. By instantiating decision points and providing analytic checklists [108], analysis tools might do more to promote *planning*, not just *implementation*. For example, an analysis team might manually author, annotate, and debate an analytic decision graph and corresponding rationales *a priori*. The results could then document and aid communication of decision points and rationales. Overviews of the end-to-end analysis process could also guide implementation work, for example with decision graph nodes linked to corresponding analysis code snippets (*i.e.,* cells in a computational notebook).

3.5.2 Multiverse Specification & Analysis

While the above methods focus on documenting decisions and selecting a preferred path, many "reasonable" alternatives may exist. Proponents of *multiverse analysis* [95, 97] have argued for preserving such decisions and evaluating them collectively. However, the design and evaluation of tools for both specifying and evaluating

multiverse analyses remains an open challenge.

Authoring a multiverse analysis may be tedious, as analysts have to write scripts to manually execute all possible combinations of reasonable alternatives. Future tools could provide better scaffolding for defining decision points and procedural branches without devolving into a morass of multiple, largely redundant analysis scripts [35]. We will discuss how we support multiverse authoring in Chapter 4.

Second, multiverse analysis poses a number of underlying systems challenges. How might one optimize multiverse evaluation, for example by efficiently reusing shared computation across "universes," or by using adaptive sampling methods to more efficiently explore a parameter space? We will discuss a sampling-based approximation method to reduce latency before assessing results in Chapter 5.

Finally, interpreting the outcomes of a vast number of analyses is difficult. Visualizations that juxtapose or animate individual outcomes [21] may not scale, and may fail to accurately convey the relative sensitivity of decision points. In addition, some of our participants bypassed decision making if they perceived the *sensitivity* to be low; they did not always verify if the decision indeed had limited influence on the results. Future tools might aggregate subsets of outcomes, and quantify the end-to-end statistical variance via a meta-analysis of multiverse results [78, 113]. Multiverse analysis tools might assess sensitivity across decision points and identify high-impact decisions for further consideration. We will introduce a visual analysis system for interpreting multiverse analyses in Chapter 6.

3.5.3 Sociotechnical Concerns

While new analysis tools might help improve systematic consideration and communication of analysis alternatives, they must operate within an accepting social environment. We are hardly the first to note that the urges to "tell a good story," sidestep unfamiliar methods, and appease reviewers can undermine a full and accurate accounting of one's research [58], and our interviews confirm their persistence. If publication incentives and reviewer criteria remain unchanged, a provenance tracking tool that reveals problematic choices, or multiverse tools that produce more comprehensive yet more complex and unfamiliar outputs, may be abandoned in favor of the status quo. Accordingly, improving the reliability of end-to-end analysis must also be a community priority, ranging from the standards and practices of peer review to how we educate researchers, new and old. We hope that the decision making and selective reporting rationales identified in our interview analysis provide useful insights for the design of both improved analysis tools *and* community processes.

3.6 Conclusion

In this chapter, we pored over nine published studies and interviewed the authors to discuss analytic decisions in the end-to-end quantitative data analysis. We presented common rationales for analytic decisions and discussed how researchers trade off between options. We observed various reasons for exploring alternatives and selectively

reporting results. We also introduced Analytic Decision Graphs and discussed recurring patterns along analysis processes. Based on the interview results, we identified design opportunities for strengthening end-to-end analysis. One major opportunity is a meta-analysis of multiple decision paths via a multiverse analysis. Chapter 4-6 address this opportunity by introducing a system for authoring, running, and interpreting multiverse analyses.

Chapter 4

The Boba DSL: Authoring Multiverse Analyses

Multiverse analysis is an approach to data analysis in which all "reasonable" analytic decisions are evaluated in parallel and interpreted collectively, in order to foster robustness and transparency. However, specifying a multiverse is demanding because analysts must manage myriad variants from a cross-product of analytic decisions. Without proper scaffolding, researchers might resort to multiple, largely redundant analysis scripts [61], or rely on intricate control flow structure including nested for-loops and if-statements.

In this chapter, we introduce Boba DSL, a domain-specific language for multiverse authoring. Rather than managing myriad analysis versions in parallel, the Boba DSL allows users to specify the shared portion of the analysis code only once, alongside local variations defining alternative analysis decisions. The compiler enumerates all compatible combinations of decisions and synthesizes individual analysis scripts for each path. As a meta-language, the Boba DSL is agnostic to the underlying programming language of the analysis script (*e.g.,* Python or R), thereby supporting a wide range of data science use cases.

In this chapter, we first describe the design requirements for the DSL that we distill from prior work and our own experiences authoring multiverse analyses. Then, we introduce the design of the Boba DSL, including its language constructs, compilation, and runtime. We evaluate the Boba DSL in a code comparison example, which demonstrates how the DSL eliminates custom control-flows when implementing a real-world multiverse of considerable complexity.

4.1 Design Requirements

Our overarching goal is to make it easier for researchers to specify multiverse analyses. From prior literature and our past experiences, we identify barriers in authoring a multiverse, and subsequently identify features that our tool should support.

As noted in prior work [21, 66], specifying a multiverse is tedious. This is primarily because a multiverse is composed of many forking paths, yet non-linear program

structures are not well supported in conventional tools [89]. One could use a separate script per analytic path, such that it is easy to reason with an individual variant, but these variants are redundant and difficult to maintain [61]. Alternatively, one could rely on control flows in a single script to simulate the nonlinear execution, but it is hard to selectively inspect and rerun a single path, and deeply nested control flows are thought to be a software development anti-pattern [71]. Instead, a tool should eliminate the need to write redundant code and custom control flows, while allowing analysts to simultaneously update variants and reason with a single variant.

Compared to arbitrary non-linear paths from an iterative exploratory analysis, the forking paths in multiverses are usually highly systematic. We take advantage of this characteristic, and account for other scenarios common in existing multiverse analyses. We distill the following design requirements:

R1: **Multiplexing**. Users should be able to specify a multiverse by writing the shared portion of the analysis source code along with analytic decisions, while the tool creates the forking paths for them. Users should also be able to reason about a single universe and update all universes simultaneously.

R2: **Decision Complexity**. Decisions come in varying degrees of complexity, from simple value replacements (*e.g.,* cutoffs for excluding outliers) to elaborate logic requiring multiple lines of code to implement. The tool should allow succinct ways to express simple value replacements while at the same time support more complex decisions.

R3: **Procedural Dependency**. Existing multiverses [97, 13] contain *procedural dependencies* [66], in which a downstream decision only exists if a particular upstream choice is made. For example, researchers do not need to choose priors if using a Frequentist model instead of a Bayesian model. The tool should support procedural dependencies.

R4: **Linked Decisions**. Due to idiosyncrasies in implementation, the same conceptual decision can manifest in multiple forms. For example, the same set of parameters can appear in different formats to comply with different function APIs. Users should be able to specify different implementations of a high-level decision.

R5: **Language Agnostic**. Users should be able to author their analysis in any programming languages, as potential users are from various disciplines adopting different workflows and programming languages.

4.2 The Boba DSL

We design a domain-specific language to aid the authoring of multiverse analyses. The DSL formally models an analysis decision space, providing critical structure that the visual analysis system later leverages. With the DSL, users annotate the source code of their analysis to indicate decision points and alternatives, and provide additional information for procedural dependencies between decisions. The specification is then compiled to a set of universe scripts, each containing the code to execute one analytic path in the multiverse. An example Boba specification for a small multiverse is shown in Figure 4-1.

(a) input.R

```
# --- (A)
df = read_csv("data.csv") %>%
     filter(speed > {{cutoff=10, 200}})

# --- (M) frequentist
model = lm(log_y ~ x, data = df)

# --- (M) bayesian
model = brm(y ~ x, data = df,
        family = {{brm_family="binomial", "lognormal"}}())
```

(b) output files

File	cutoff	brm_family	M
1.R	10		frequentist
2.R	200		frequentist
3.R	10	binomial	bayesian
4.R	10	lognormal	bayesian
5.R	200	binomial	bayesian
6.R	200	lognormal	bayesian

(c) 1.R

```
df = read_csv("data.csv") %>%
     filter(speed > 10))
model = lm(log_y ~ x, data = df)
```

(d) 4.R

```
df = read_csv("data.csv") %>%
     filter(speed > 10))
model = brm(y ~ x, data = df,
        family = lognormal())
```

Figure 4-1: An example Boba specification. The user annotates an R script (a) with two placeholder variables (blue) and three code blocks (pink). The compiler synthesizes six files (b). In the example output files (c) and (d), placeholder variables are replaced by their possible values, and only one version of the decision block M is present.

4.2.1 Language Constructs

The basic language primitives in the Boba DSL consist of source code, placeholder variables, blocks, constraints, and code graphs.

Source Code

The most basic ingredient of an annotated script is the source code (Figure 4-1a, black text). The compiler treats the source code as a string of text, which according to further language rules will be synthesized into text in the output files. As the compiler is agnostic about the semantics of the source code, users are free to write the source code in any programming language (R5).

Placeholder Variables

Placeholder variables are useful to specify decisions points consisting of simple value substitution (R2). To define a placeholder variable, users provide an identifier and a set of possible alternative values that the variable can take up (Figure 4-1a, blue text). To use the variable, users insert the identifier into any position in the source code. During synthesis, the compiler removes the identifier and replaces it with one of its alternative values. Variable definition may occur at the same place as its usage

(a) input.R

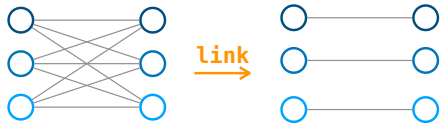
```
# --- (BOBA_CONFIG)
{
  "decisions": [
    {"var": "formula", "options": [
      "female * damage",
      "female * damage + zwin + female:zwin",
      "female + z3 + female:year"
    ]},
    {"var": "variables", "options": [
      "female, damage",
      "female, damage, zwin",
      "female, z3, year"
    ]}
  ],
  "constraints": [
    {"link": ["formula", "variables"]}
  ]
}
# --- (END)

df <- read_csv('../data.csv')

model <- lm(log_death~{{formula}}, data = df)

uncertainty = df %>%
  data_grid({{variables}}) %>%
  augment(model, newdata = .)
```

(b) combinations



(c) outputs

File	formula	variables
1.R	female * damage	female, damage
2.R	female * damage + zwin + female:zwin	female, damage, zwin
3.R	female + z3 + female:year	female, z3, year

(d) 3.R

```
df <- read_csv('../data.csv')

model <- lm(log_death~female + z3
   + female: year, data = df)

uncertainty = df %>%
  data_grid(female, z3, year) %>%
  augment(model, newdata = .)
```

Figure 4-2: A Boba specification illustrating linked decisions. Here, the same conceptual decision (what variables are in the model) has multiple implementations (one function expects a formula, while another expects a list). The user defines two placeholder variables and links them (a). Linked decisions have a one-to-one mapping (b) such that the ith alternative is always chosen together (c). An example output file is shown in (d).

(Figure 4-1a) or ahead of time inside the config block (Figure 4-2a).

Code Blocks

Code blocks (Figure 4-1a, pink text) divide the source code into multiple snippets of one or more lines of code, akin to cells in a computational notebook. A block can be a *normal block* (Figure 4-1a, block A), or a *decision block* (Figure 4-1a, block M) with multiple versions. The content of a normal block will be shared by all universes, whereas only one version of the decision block will appear in a universe. Decision blocks allow users to specify alternatives that require more elaborate logic to define (R2). In the remainder of Chapter 4, *decision points* refer to placeholder variables and decision blocks.

With the constructs introduced so far, a natural way to express procedural dependency (R3) is to insert a placeholder variable in some, but not all versions of a decision block. For example, in Figure 4-1, the variable `brm_family` only exists when `bayesian` of block M is chosen.

Constraints

By default, Boba assumes all combinations between decision points are valid. Constraints allow users to express dependencies between decision points, for example infeasible combinations, which will restrict the universes to a smaller set. Boba supports two types of constraints: procedural dependencies (R3) and linked decisions (R4).

A procedural dependency constraint is attached to a decision point or one of its alternatives, and has a conditional expression to determine when the decision/alternative should exist (Figure 4-3b, orange text). Variables within the scope of the conditional expression are declared decision points, and the values are the alternatives that the decision points have taken up. For example, the first constraint in Figure 4-3b indicates that `ECL computed` is not compatible with `NMO reported`.

The second type of constraint allows users to *link* multiple decision points, indicating that these decision points are different manifestations of a single conceptual decision (R4, Figure 4-2). Linked decisions have one-to-one mappings between their alternatives, such that the i-th alternatives are chosen together in the same universe. One-to-one mappings can also be expressed using multiple procedural dependencies, but linked decisions make them easier to specify.

Code Graph

Users may further specify the execution order between code blocks as a directed acyclic graph (DAG), where a parent block executes before its child. To create a universe, the compiler selects a linear path from the start to the end, and concatenates the source code of blocks along the path. Branches in the graph represent alternative paths that appear in different universes. Users can flexibly express complex dependencies between blocks with the graph, including procedural dependencies (R3). For example, to indicate that block `prior` should only appear after block `bayesian` but not block `frequentist`, the user simply makes `prior` a descendant of `bayesian` but not `frequentist`.

4.2.2 Compilation and Runtime

The compiler parses the input script, computes compatible combinations between decisions, and generates output scripts. During parsing, the compiler extracts the corresponding language primitives from the input file. To enumerate compatible combinations, the compiler proceeds in the following steps. First, it obtains the DAG specifying the execution relationships between code blocks. If users omit the DAG, the compiler creates a default graph which is a linear path of all code blocks depending on their order in the input script. The compiler modifies the DAG to incorporate decision blocks and constraints. Then, it computes all possible paths from any source node with no input edges to any destination node with no output edges. For each path, the compiler further expands placeholder variables (*i.e.,* it enumerates all possible combinations between them). Finally, for each path and

(a)

```
for (i in 1:no.nmo){          # for each NMO option
  for (j in 1:no.f){          # for each F option
    for (k in 1:no.r){        # for each R option
      for (l in 1:no.ecl){    # for each ECL option
        for (m in 1:no.ec){   # for each EC option
          # preprocessing code
          [...]
          if (i == 1) {
            [...]  # code for the first NMO option
          } else if (i == 2) {
            [...]  # code for the second NMO option
          } else if (i == 3) {
            [...]  # code for the third NMO option
          }

          # fertility options
          bounds = c(7,8,9,8,9)
          df$fertility[df$cycle > bounds[j]] = 'High'
          [...]

          if (l == 1) {
            [...]  # code for the first ECL option
          } else if (l == 2) {
            if (i == 2) {
              next
            }
            [...]  # code for the second ECL option
          } else if (l == 3) {
            if (i == 1) {
              next
            }
            [...]  # code for the third ECL option
          }
          # two more decisions are omitted
          [...]
        }
      }
    }
  }
}
```

(b)

```
# preprocessing code
[...]

# --- (NMO) computed
[...]  # code for the 1st NMO option

# --- (NMO) reported
[...]  # code for the second NMO option

# --- (NMO) estimate
[...]  # code for the third NMO option

# --- (F)
df$fertility[df$cycle > {{bound=7,8,9,8,9}}] = 'High'
[...]

# --- (ECL) none
[...]  # code for the first ECL option

# --- (ECL) computed @if NMO != reported
[...]  # code for the second ECL option

# --- (ECL) reported @if NMO != computed
[...]  # code for the third ECL option

# two more decisions are omitted
[...]
```

Figure 4-3: Specification of a real-world multiverse analysis [97] with five decisions and a procedural dependency. (a) Markup of the R code written by original authors, with custom control flow (nested for-loops and if-statements) highlighted. (b) Markup of the Boba DSL specification.

variable combination, the compiler concatenates source code along the path, replaces placeholder variables with corresponding values, and outputs a universe script. It also outputs a summary table that keeps track of all the decisions made in each universe, along with other intermediate data that can be ingested into the visual analysis components of the Boba system.

Boba infers the language of the input script based on its file extension and uses the same extension for output scripts. These output scripts might be run with the corresponding script language interpreter. Universe scripts log the results into separate files, which will be merged together after all scripts finish execution. Each universe must output a point estimate, along with other optional data such as a p-value, a model quality metric, or a set of sampled estimates to represent uncertainty. As the universe scripts are responsible for computations such as extracting point estimates and computing uncertainty, we provide language-specific utilities for a common set of model types to generate these visualizer-friendly outputs. We also provide a command-line tool for users to (1) invoke the compiler, (2) execute the generated universe scripts, (3) merge the universe outputs, and (4) invoke the visualizer as a local server reading the intermediate output files.

4.3 Example: Replicating a Real-World Multiverse

We use a real-world multiverse example [97] to illustrate how the Boba DSL eliminates the need for custom control flows otherwise required for authoring a multiverse in a

single script. The multiverse, originally proposed by Steegen *et al.* [97], contains five decisions and a procedural dependency. Figure 4-3a shows a markup of the R code implemented by the original authors (we modified the lines in purple to avoid computing infeasible paths). The script starts with five nested for-loops (yellow highlight) to repeat the analysis for every possible combination of the five decisions. Then, depending on the indices of the current decisions, the authors either index into an array, or use if-statements to define alternative program behaviors (blue highlight). Finally, to implement a procedural dependency, it is necessary to skip the current iteration when incompatible combinations occur (purple highlight).

The resulting script has multiple issues. First, the useful snippets defining multiverse behavior start at five levels of nesting at minimum. Such deeply nested code is often considered to be hard to read [71]. Second, nested control flows are not easily amenable to parallel execution. Third, without additional error-handling mechanisms, an error in the middle will terminate the program before any results are saved.

The corresponding specification in the Boba DSL is shown in Figure 4-3b. The script starts directly with the preprossessing code shared by all universes. It then uses decision code blocks to define alternative snippets in decision `NMO` and `ECL`, and uses a placeholder variable to simulate the value array for a simpler decision `F`. It additionally specifies constraints (orange text) to signal incompatible paths. Compared to Figure 4-3a, this script reduces the amount of boilerplate code needed for control-flows and does not require any level of nesting. The script compiles to 120 separate files. Errors in one universe no longer affect the completion of others due to distributed execution, it is trivial to execute universes in parallel, and users can selectively review and debug a single analysis.

4.4 Conclusion

In this chapter, we introduced a domain-specific language to aid the authoring of multiverse analyses. With the DSL, users annotate their analysis script to insert local variations, from which the compiler synthesizes executable script variants corresponding to all compatible analysis paths. The Boba DSL is agnostic to the underlying programming language of the analysis script (*e.g.,* Python or R), thereby supporting a wide range of data science use cases. In a code comparison example, we demonstrated how the Boba DSL eliminates custom control-flows when implementing a real-world multiverse of considerable complexity.

The Boba DSL formally models an analysis decision space, providing critical structure that other system components later leverages. Chapter 5-6 discuss how the visual analysis components of the Boba system automatically generate Analytic Decision Graphs using information provided by the Boba DSL.

The Boba DSL aids users in the first step of the multiverse analysis pipeline: authoring the multiverse specification. Chapter 5-6 introduce tools for supporting subsequent steps, namely multiverse execution and interpretation.

Chapter 5

Approximation Algorithms and the Boba Monitor: Running Multiverse Analyses

The Boba DSL compiles user specification into a collection of executable script variants, but this collection may contain a combinatorial explosion of analyses to compute. Combined with the need to evaluate each script end-to-end, a large collection of computationally demanding analyses (*e.g.,* Bayesian models) might take hours or even days to run. Furthermore, multiverse analysis workflows can be *iterative*. It can be challenging to construct an *a priori* "reasonable" decision space based solely on theoretical and methodological concerns, as certain validity issues only arise during runtime, such as model convergence and goodness of fit (Chapter 7). As a result, analysts might revise the multiverse specification according to what seems reasonable post-hoc, run the multiverse again, and repeat the process if necessary. Being able to assess and revise multiverse results efficiently is thus important.

To address these challenges, we first investigate approximation algorithms based on sampling: given a decision space, might we find a subset of universes that provide a good estimate of key results? Since the primary goal of multiverse analysis is to gauge the robustness of outcomes to analytic decisions, we would like the sampling algorithms to accurately estimate *decision sensitivity*: to what extent outcomes vary across different options within a decision. We also ensure that the algorithms estimate the *mean effect size* with low bias. By estimating these two types of multiverse results quickly, the algorithms reduce the time until results are "good enough" to move forward with additional analysis. Sampling is also compatible with parallelism (*e.g.,* running universes across multiple processes), a common method for reducing latency. As a secondary contribution, we evaluate four methods for quantifying the sensitivity of individual decisions and recommend a metric based on the K-samples Anderson-Darling test [91].

Next, we propose a monitoring dashboard for analysts to track progress, identify issues, and control execution of the multiverse – leveraging sampling-based approximation algorithms under the hood. To help analysts determine when the sampling has achieved reasonable performance, the interface displays sampling estimates in real

time with confidence intervals. More importantly, the dashboard enables analysts to reflect on the decision space structure, review intermediate results, and identify issues including runtime warnings and model diagnostics, before the multiverse evaluation completes. When issues are identified early, users can stop the execution, refine the multiverse specification, and then resume, thus preventing wasted effort in running a problematic multiverse specification to completion.

In this chapter, we first discuss the goals for approximation algorithms and design requirements for the monitoring dashboard. Next, we describe three sampling-based approximation algorithms, four methods for quantifying decision sensitivity, and a technique for correcting for bias in mean estimation. We then present an empirical evaluation of the sampling algorithms using synthetic and real multiverses. Finally, we introduce the design of the Boba Monitor.

5.1 Requirements

We now outline our goals for approximation algorithms and design requirements for monitoring visualizations.

5.1.1 Requirements: Approximation Algorithms

Online Approach: We would like to adopt an online approach such that intermediate results can be displayed progressively in a user interface. The approximation algorithms need to support continuous update and refinement of results.

Accurate Estimation of Key Aspects: We would like important aspects of the multiverse analysis to be approximated accurately. Among prior works that report multiverse analysis results, the primary task is to understand the robustness of results across all reasonable specifications. In particular, do certain decisions lead to large variations in outcomes? If so, which decisions are the most sensitive? Thus, accurate estimation of **decision sensitivity** is an important goal. In addition, the average of all end-to-end analysis outcomes (*e.g.,* effect size) suggest the overall direction and magnitude of effect. Therefore, the algorithms should also estimate the **mean outcome** accurately.

Confidence Intervals: When viewing intermediate results, analysts should be able to gauge the uncertainty of the sensitivity or mean estimates. The algorithms should produce confidence intervals (CI) that can help analysts decide when the results are "good enough".

5.1.2 Requirements: Monitoring Dashboard

We would like a monitoring dashboard for analysts to inspect the multiverse structure and intermediate outcomes without waiting for the entire multiverse to finish running. With the dashboard, analysts should be able to diagnose issues early and stop the multiverse execution, in order to iterate on the multiverse specification. In addition, users should be able to gauge when the sampling has achieved reasonable convergence,

and dive into the next phase of the multiverse analysis workflow in an optimistic manner.

We identify the following tasks that analysts need to perform:

- T1. **Review Decisions**: The tool should visualize the decision space for analysts to reflect on the multiverse structure, answering questions like "are possible decisions missing" and "does a decision dominate the space"?
- T2. **Control Execution**: Analysts should be able to start, stop, and resume execution. Controlling execution is useful in various scenarios, including pausing to reclaim computational resources temporarily, or to sanity check errors and potential issues.
- T3. **Assess Progress**: Analysts should be able to track the progress of multiverse execution, in particular mean effect size and decision sensitivity over time with confidence intervals. Viewing progress allows analysts to gauge when the sampling results are stable enough to afford an optimistic analysis.
- T4. **Find Issues**: Analysts should be able to observe potential issues and decide whether to stop early and refine the multiverse. Potential issues include an abnormal range of estimates, program errors and warning messages, and poor model quality.
- T5. **Identify Causes**: When issues arise, the tool should facilitate analysts in identifying the decisions that lead to the issues. Finding the contributing factors allows analysts to discard invalid options or narrow down the subset of universes to debug further.

5.2 Sampling-Based Approximation Algorithms

5.2.1 Sampling Algorithms

We investigate three approximation algorithms based on random sampling. Sampling lends itself naturally to progressive display as universes are drawn one at a time from the multiverse. We choose two algorithms that might have an advantage in estimating decision sensitivity quickly, and a third algorithm – uniform sampling – to serve as a baseline.

Notation. A multiverse consists of m decisions $\{A_i, i = 1...m\}$. Each decision A_i has k_i discrete options, forming the set $\mathcal{D}_i = \{a_{il}, l = 1...k_i\}$. The decision space is initially the product space $\Theta = \mathcal{D}_1 \times ... \times \mathcal{D}_m$, but some combinations may be invalid (*e.g.,* frequentist models and priors). All valid combinations give rise to n universes, $n \leq |\Theta|$. All decisions in a given universe are summarized in vector s_j with $s_j(A_i) \in \mathcal{D}_i$. Executing the end-to-end script of the j-th universe produces an outcome value (*e.g.,* effect size estimate), denoted as $y_j \in \mathbb{R}$.

Sketching. The first sampling method is a sketching algorithm in linear experimental design [86], with the goal of selecting a subset of input data points such that the underlying linear model is estimated accurately. To apply the sketching algorithm, we first assume that a linear model describes the relationship between the input

decision combination and the corresponding multiverse outcome value. The model is most likely "wrong", but it is useful because a large model coefficient may indicate that the corresponding decision variable has a large impact on the outcome. Since the decisions are categorical variables, we apply one-hot encoding enc: $s \to \mathbb{R}^d$ to obtain the feature vector $\text{enc}(s_i) = x_i$. Then, we define a linear model between x_i and y_i, with $\epsilon_i \sim N(0, \sigma^2)$ accounting for the model mismatch:

$$y_i = x_i^T \beta + \epsilon_i \quad i = 1...n \tag{5.1}$$

The sketching algorithm aims to find a small number of (x_i, y_i) such that the estimated β from the reduced version of the data is not too much different from the original Least Square estimator [86]. It relies on statistical leverage scores, where each leverage score corresponds to a feature vector x_i. A higher leverage score indicates that the universe is more influential on the estimation. If we stack x_i as rows in matrix $X \in \mathbb{R}^{n \times d}$, the leverage score of the i-th universe is:

$$l_i = (UU^T)_{ii}$$

where $X = U\Lambda V^T$ is the singular value decomposition for X. The algorithm then samples universes according to an independent distribution $(p_i)_{i=1}^n$, where $p_i = \frac{l_i}{d}$.

Round Robin. The second algorithm is adapted from round robin stratified sampling, a widely-used sampling strategy in online aggregation of database systems (*e.g.,* [41, 44]). In each round, the algorithm goes over each decision and subsequently every option of the decision, and samples a universe uniformly at random from all remaining universes adopting the option. This procedure ensures that for each decision option we will eventually have at least one multiverse covering that decision. The pseudocode is shown in Algorithm 1.

Algorithm 1 Round Robin

```
Initialize T ← ∅
while |T| < n do
    for each A_i ∈ {A_1, ..., A_m} do
        for each a_j ∈ D_i do
            S ← {s_k | s_k(A_i) = a_j and s_k ∉ T}
            if S ≠ ∅ then
                Draw a sample s uniformly at random from S
                T ← T ∪ {s}
```

Uniform Sampling. As a baseline method, we also apply uniform sampling that draws each universe with equal probability. Specifically, the algorithm samples universes from a discrete uniform distribution, where the probability of drawing the i-th universe is $p_i = \frac{1}{n}$.

5.2.2 Correcting for Bias in Mean Estimation

Intuitively, when the decision space is not a Cartesian product of all decisions, sketching and round robin algorithms over-sample certain regions of the decision space. Sketching assigns higher leverage scores to data points that have a larger influence on the estimation of the linear model. Round robin ensures that at least one sample is drawn from each option in a round, including rare options that would have a very low probability of being drawn in uniform sampling. As a results, the sampling is biased. This bias is intended, as some outcomes y_i are important to know for gauging sensitivity, yet unlikely to be sampled. We would like to correct for the bias such that we are not changing the final estimator of the mean outcome.

We apply *importance sampling*, where the goal is to estimate properties of a distribution with only samples generated from a different, "biased" distribution. Importance sampling corrects for the use of the biased distribution by applying weights given by the likelihood ratio:

$$\bar{y} = \frac{1}{|T|} \sum_{i \in T} \frac{y_i f(y_i)}{g(y_i)} \tag{5.2}$$

where f is the probability density function of the target distribution and g is the probability density function of the biased distribution. In our case, the target distribution is the discrete uniform distribution, $f(y_i) = \frac{1}{n}$. The sampling distribution of the sketching algorithm, as described in the previous section, is $g(y_i) = \frac{l_i}{d}$.

We compute $g(y_i)$ of the round robin algorithm as follows. Recall that in each round, the algorithm goes over every decision A_j, and a universe would have a matching option for each of the decisions. Thus, the universe might be selected from any of the decisions, but the universe can appear in the sample only once. With independent sampling between decisions, the inclusion probability for the i-th universe is

$$g(y_i) = \sum_j q_{ij} - \sum_{j<k} \sum q_{ij} q_{ik} + \sum_{j<k<l} \sum \sum q_{ij} q_{ik} q_{il} \cdots$$

where q_{ij} is the probability that the i-th universe is selected from the j-th decision. Basically, the equation calculates the probability that the i-th universe is selected by the first decision or the second decision or the third ... As we sample uniformly from each stratum, q_{ij} is simply the inverse of the size of the stratum that the universe belongs to.

5.2.3 Quantifying Decision Sensitivity

Because decision sensitivity is an important goal of our approximation algorithms, we need a method to quantify it. To gain a deeper understanding of different sensitivity metrics, we conduct an evaluation on four candidate methods using synthetic datasets. Here, we briefly introduce the methods and take-away results.

Metrics

We compare the following methods:

- The *F-test* quantifies how much a decision shifts the mean of outcomes compared to the variance. If some options produce very different means than others, the decision may be highly sensitive.
- The *Kolmogorov–Smirnov (K–S) test* is a non-parametric method to quantify the difference between two distributions. Using the test, we measure how different the outcome distributions are across different options of a decision. If certain options lead to very different distributions, the decision may be highly sensitive.
- The *K-samples Anderson–Darling (AD) test* is another non-parametric method for establishing differences in two or more distributions [91]. Intuitively, the AD test is similar to the K–S test, but it compares k options simultaneously.
- *Linear regression (LR)* models a multiverse outcome as a linear combination of decision values (Equation 5.1). We would expect highly sensitive decisions to be associated with large coefficients.

Summary of Sensitivity Evaluation

False Positives: Could a decision appear more sensitive simply because it has more options? We construct a synthetic multiverse with only a non-sensitive decision (using the same general scheme as in Section 5.3.1), then vary the number of options within this decision. We find that the sensitivity score of this non-sensitive decision in K–S test and LR increase with decision cardinality. In other words, K–S test and LR may consider a decision to be more sensitive simply because the decision has more options.

Non-Normality: We then check how well the sensitivity tests handle non-normal distributions. We simulate decisions where the options are from different distributions (*e.g.,* normal versus Poisson), but have the same mean and standard deviations. Being parametric methods, F-test and LR cannot reliably distinguish these different distribution functions. For example, a normally-distributed option and another lognormal-distributed option appear the same to F-test and LR.

Sensitivity Ranking: Finally, we systematically construct a series of synthetic multiverses with multiple decisions, then assess how well the metrics estimate the correct ranking of sensitive decisions. F-test and AD test correctly recover the ranking in nearly all datasets, while K–S test and LR fail to do so in over half of the datasets.

Based on these take-aways, we find the AD test to be the most effective for quantifying sensitivity. We use it in all subsequent experiments and recommend its use over the F-test and K–S metrics in our visual analysis interfaces.

5.2.4 Confidence Intervals

We must calculate 95% confidence intervals (CI) for both decision sensitivity and outcome mean estimates. A widely-used method in database online aggregation literature is using Hoeffding's inequality [43] to approximate the size of the confidence

intervals, but we find the CIs to be too loose for the common size of multiverse analysis. Instead, we use bootstrapping: sampling with replacement within the universes drawn so far, we compute the decision sensitivity or bias-corrected mean to construct a resampling distribution, and obtain a CI from this resampling distribution. To account for potential skew in the resampling distribution, we use the **bias-corrected and accelerated bootstrap** [22] to derive CIs.

5.3 Evaluation of Approximation Algorithms

We perform empirical validation of the sampling algorithms in a suite of synthetic and real multiverse analyses. We first measure how quickly the algorithms *approximate* and *rank* sensitive decisions. Then, we evaluate the method for correcting bias in mean estimation.

5.3.1 Datasets

We design five synthetic datasets that are inspired by characteristics of real multiverse analyses. Building these synthetic datasets allow us to control the data generating process and tease apart interesting properties of the multiverse to study each property in isolation. We also run the benchmark on a multiverse analysis in the wild [92] to demonstrate the utility of the sampling algorithms in real-world scenarios.

We now describe the general scheme for constructing the synthetic dataset. We would like the synthetic data to contain both signal and noise to better simulate real-world scenarios, yet we need to have control over how much a decision influences the outcome. Thus, we model each option a_i within a decision as a normal random variable $N(\mu_{a_i}, \sigma^2)$, with a larger difference in μ between options indicating a more sensitive decision. The outcome of a universe is then the sum over the contributions from all decisions. Specifically, given an input vector of the i-th universe s_i, the outcome is

$$y_i = \sum_{j=1}^{m} Z_j, \quad Z_j \sim N(\mu_{s_i(A_j)}, \sigma^2)$$

Multiverses may contain decisions that are not important at all (*e.g.,* [113]). To model these, we first define a *baseline option* by setting the mean to zero, such that the option contributes nothing but random noise to the outcome. We then construct *non-sensitive decisions* by setting every option to a baseline option. Multiverses may also contain certain *rare* conditions – the number of universes adopting a particular option is smaller compared to the number of universes adopting other options (*e.g.,* [92]). We capture these by simulating procedural dependencies, which exclude invalid combinations from the Cartesian product decision space. In building synthetic datasets, we also seek to make the total size, the number of decisions, and the cardinality of decisions as realistic as possible (Chapter 3). The sizes of the synthetic multiverses range from 200 to 1,552, with 4 to 8 decisions and 2 to 10 options

per decision, unless stated otherwise. We now introduce the characteristics of each dataset.

D1: Simplest. We design this synthetic multiverse to be the simplest scenario where we would expect the sampling algorithms to perform differently. The multiverse contains four non-sensitive decisions and one sensitive decision with some baseline zero-mean options, as well as an influential option with a large mean $\mu_{a_i} = 6\sigma$. This influential option is relatively rare due to invalid combinations. We expect both sketching and round robin to have a different probability of including the influential option compared to uniform sampling.

D2: Interaction. Decisions may interact in real-world multiverses, where the impact of one decision on the outcome depends on the option chosen by another decision. We construct a synthetic multiverse with eight binary decisions, six of which are non-sensitive. The remaining two sensitive decisions interact, and one of the interacting combination is relatively rare.

D3: High Cardinality. This dataset aims to evaluate how well sampling algorithms perform on a decision with a large number of options. The multiverse has one sensitive decision with 50 options, including 45 baseline options and 5 options with non-zero means. Half of the options, including the 5 impactful options, are relatively rare. The other decisions are non-sensitive.

D4: Distractor Decisions. Building upon the simplest scenario D1, we would like to "distract" the sampling algorithms in finding the rare, sensitive decision by including other sensitive decisions. The multiverse has three sensitive decisions and four non-sensitive ones. Similar to D1, a sensitive decision consists of a rare influential option among other baseline options. The other two sensitive decisions do not involve procedural dependencies, serving as the "distractors".

D5: Distractor Options. Building again on the simplest scenario D1, we now seek to distract the sampling algorithms by making other options rare. The synthetic multiverse has the same composition as D1, except that four baseline options of the sensitive decisions are relatively rare, in addition to the influential option.

D6: Real Multiverse. The last dataset is a real-world multiverse analysis, taken from a study on crowdsourced data analysis [92]. The multiverse has seven decisions, including different ways to operationalize dependent and independent variables, choose the model family, and pick the set of covariates. We choose this dataset because the decisions have complex dependencies and the multiverse is large in size with 2,977 universes.

5.3.2 Evaluation: Decision Sensitivity

Using the six datasets, we first evaluate how well the sampling algorithms estimate decision sensitivity.

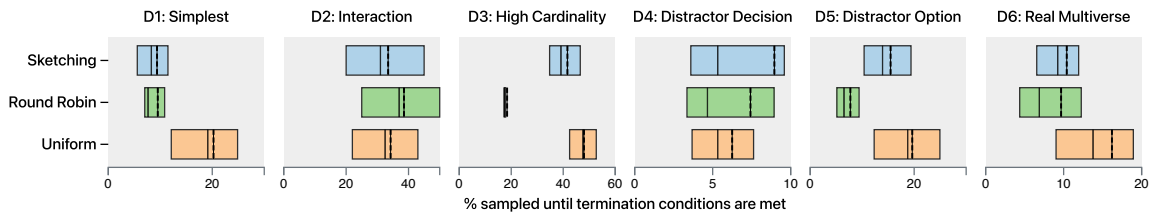


Figure 5-1: Empirical evaluation of the approximation algorithms on six datasets. The x-axis indicates the percentage of the full multiverse sampled until the algorithms accurately estimate and rank sensitive decisions (lower is better). The box plot shows the median and IQR across 200 runs using different random seeds, and the dashed line represents the mean.

Methods

We assess how quickly the algorithms *approximate* and *rank* sensitive decisions. To this end, we define a set of termination conditions, sample universes progressively, and record the percentage of the full multiverse drawn when the termination conditions are met. The termination conditions consist of three requirements.

First, the estimated sensitivity must closely match the ground truth. We quantify the sensitivity of individual decisions using the standardized test statistics of the k-samples AD test, which produces a list of sensitivity scores. We then compute the Pearson correlation of the sensitivity scores between the sample and the full data. The first condition requires the Pearson correlation coefficient to be larger than 0.95.

Second, the sensitive decisions must be ranked correctly. Here, we discard non-sensitive decisions, because non-sensitive decisions do not have a clear ranking, yet may have slightly different sensitivity scores due to noise in the data generation process. To assess ranking, we calculate the Spearman correlation between the sensitivity scores of sensitive decisions in the sample and the full data. The Spearman correlation must be 1 for the second condition to be met.

Third, each option must have at least three samples. This condition is a prerequisite for the k-samples AD test to work reasonably well [91].

Because the samples drawn by all three sampling algorithms vary due to randomness, we repeat the benchmark 200 times using different random seeds and report the median performance.

While the method above summarizes performance into a single number, it relies on several cutoff values. To further characterize performance, we examine Pearson and Spearman correlation (if applicable) as more samples are drawn. We average the correlations across 200 runs. The correlation is null when the options do not have sufficient sample size for the AD test. Because a sample producing a null correlation is useful information, we impute these null values using 0, as if there is no relationship between the sample sensitivity and the ground truth.

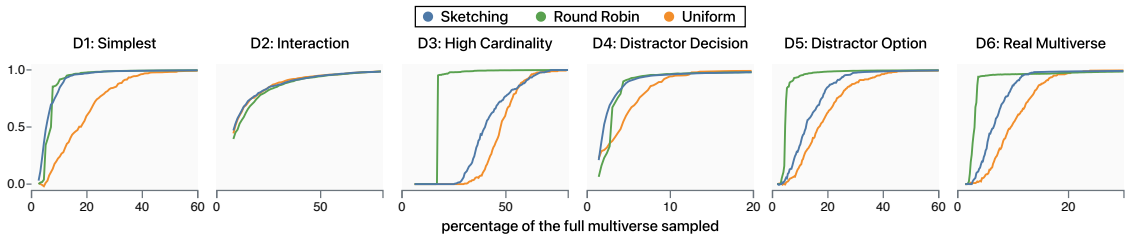


Figure 5-2: Pearson correlation between sample sensitivity and the ground truth over time. The x-axis encodes the percentage of the full multiverse sampled (lower is better) and the y-axis encodes the correlation coefficient (higher is better).

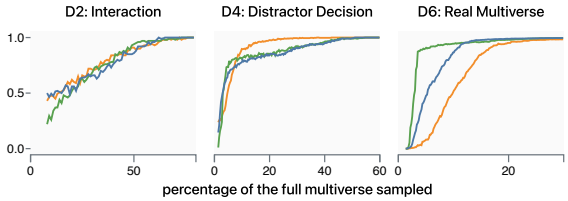


Figure 5-3: Spearman's rank correlation between sample sensitivity and the ground truth over time. The x-axis encodes the percentage of the full multiverse sampled and the y-axis encodes the correlation coefficient. Datasets with only one sensitive decision are omitted.

Results

Figure 5-1 shows the mean, median, and IQR of the proportion of the full multiverse drawn until the three termination conditions are met, across 200 runs. In the simplest scenario D1, the sketching and round robin algorithms take 9.4% and 9.5% on average to meet the termination goals, respectively. Both algorithms are over 2 times faster than uniform sampling (mean=20.2%). Compared to no approximation at all, using an approximation algorithm, even the baseline, is 5 times faster.

In two of the more complex scenarios, round robin outperforms the other algorithms by at least 2 times, while sketching is slightly better than the baseline. On a multiverse with a high cardinality decision (D3), round robin is relatively fast (mean=18.4%) in including all options needed for an accurate estimation, due to the structure imposed in the sampling procedure. On the contrary, the random nature takes sketching (mean=41.6%) and uniform (mean=48.0%) much longer to gather sufficient samples across all options. When distractions from other rare options are present (D5), round robin maintains a similar performance to D1 (mean=7.7%), but sketching is considerably worse (mean=15.5%). In the remaining two synthetic datasets, sketching and round robin do not provide a performance gain over the baseline.

In D2, sketching offers no advantage in detecting interactions between decisions, since its underlying linear model does not include interaction terms. As a follow-up exploratory analysis, we add all possible two-way interactions in the linear model,

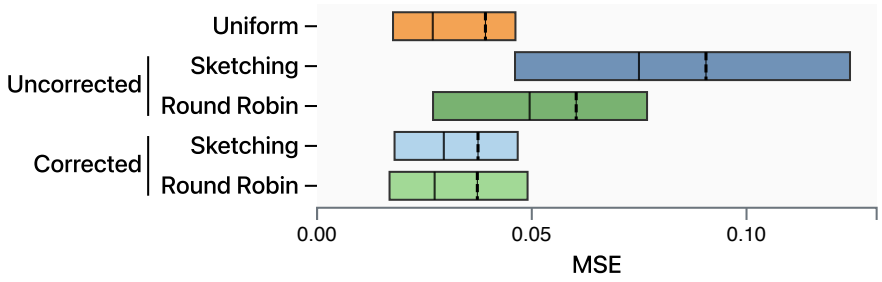


Figure 5-4: Empirical evaluation of bias correction in mean estimation. The x-axis shows the MSE between the estimated and actual mean (lower is better). The box plot shows the median and IQR across 200 runs, and the dashed line represents the mean.

and sketching improves slightly by using 3% fewer samples.

In D4, other sensitive decisions might indeed distract sketching and round robin from correctly ranking sensitivity. Let A_1 be the rare sensitive decision and A_2 be the distractor decision. In the full multiverse, A_1 is less influential than A_2. As both algorithms over-sample A_1, the larger sample size with more "evidence" of impact leads both algorithms to believe A_1 to be more sensitive. Figure 5-3-D4 shows how the ranking of sensitive decisions is disrupted along the course, while overall sensitivity scores of all decisions are already very close to the full data (Figure 5-2-D4).

Finally, both sketching (mean=10.4%) and round robin (mean=9.7%) are considerably faster on the real multiverse compared to the baseline (mean=16.2%). The trends of decision similarity (Figure 5-2-D6) and ranking (Figure 5-3-D6) over time show that round robin outperforms sketching, while sketching outperforms uniform sampling.

5.3.3 Evaluation: Bias Correction in Mean Estimation

We also evaluate the methods for correcting bias in the estimation of mean multiverse outcome, as described in Section 5.2.2. Recall that uniform sampling gives an unbiased estimation of the mean, whereas sketching and round robin may shift the outcome by oversampling certain regions of the decision space. The empirical evaluation seeks to validate that the two algorithms produce a mean estimate as accurate as uniform sampling, after bias correction.

In the experiment, we draw samples progressively until all universes are included. We then calculate the mean squared error between estimated and actual mean:

$$\text{MSE} = \frac{1}{n-b}\sum_{i=b}^{n}(\bar{y}_i - \mu)^2$$

where $\bar{y}_i$ is the estimated mean (Equation 5.2), μ is the true mean, and b is a parameter for discarding large errors in very small samples. We set b to be the sum of the cardinality of all decisions. To offset randomness, we repeat the experiment 200

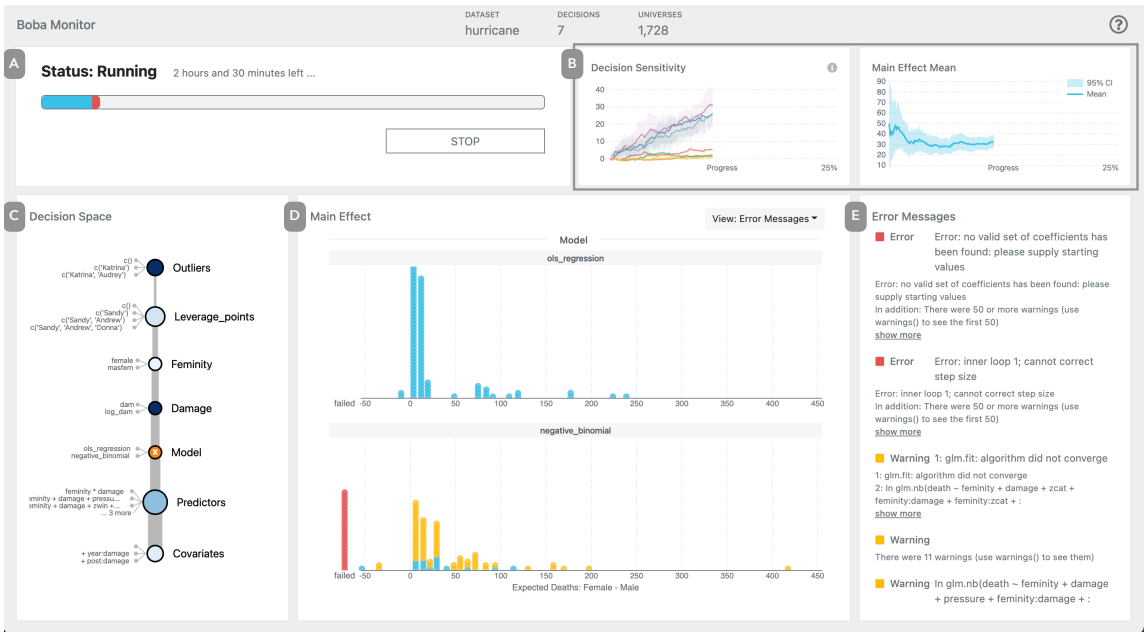


Figure 5-5: Leveraging underlying approximation algorithms, the Boba Monitor enables analysts to monitor progress and diagnose issues while a multiverse analysis is running. Analysts can control the execution on the fly (a), observe progressive estimates of decision sensitivity and effect size (b), and gauge when the approximation has achieved reasonable convergence. To assess validity, analysts may reflect on the decision space structure (c), examine the range of effect size estimates (d), and review runtime errors and warnings (e). Clicking a decision node (c) allows users to compare between options and identify which option(s) lead to specific issues (d).

times using different random seeds.

Figure 5-4 shows the MSE of 200 runs on D4. The result confirms that the arithmetic mean of sketching and round robin does differ considerably from the actual mean, but the bias correction method successfully brings MSE to a value similar to uniform sampling.

5.4 Progressive Visualization: The Boba Monitor

Next, we introduce the visual design and interactions of the Boba Monitor, a component of the Boba system. The Boba Monitor is a dashboard for users to control and monitor multiverse execution. It serves as a bridge between authoring the multiverse specification using the Boba DSL (Chapter 4), and analyzing the multiverse results using the Boba Visualizer (Chapter 6). Before opening the dashboard, users first need to author the multiverse specification, including the shared portion of the analysis script and corresponding decisions. After using the dashboard, users either go back to iterate on the multiverse specification, or proceed to the next stage of multiverse analysis using the Visualizer.

5.4.1 Monitoring Progress

The upper part of the dashboard is primarily for controlling the execution and observing the progress as the multiverse continues to run.

Control Panel

The control panel (Figure 5-5a) allows users to directly control the multiverse runtime from within the dashboard. It supports starting, stopping and resuming the execution (T2). The panel also displays a progress bar to indicate the proportion of universes completed so far, along with an estimated completion time. The progress bar and estimated completion time are updated in real time as the multiverse runs in the background, allowing users to continuously track the basic status of execution (T3). From a user experience perspective, a progress indicator alone can improve the perceived speed of the system [74].

Before invoking the multiverse runtime, the system applies one of the approximation algorithms to calculate the sampling order and executes universe scripts according to the order. By default, we use the round robin algorithm, but users can override the default in configurations. As opposed to running the multiverse sequentially, apply a sampling algorithm is crucial because it not only reduces the time until decision sensitivity is estimated accurately, but also ensures that the intermediate results are not misleading.

Progress View

The progress view (Figure 5-5b) displays the trends of sampling estimates over time, enabling users to continuously observe progress and determine when the estimates are good enough (T3). The view includes two line charts: one visualizes the time-series of the average multiverse outcome, and the other show a collection of time-series of decision sensitivity score, one per decision. Around each line, an area band indicates the 95% confidence interval of the corresponding estimate. Similar to the control panel, these charts are updated continuously, but with a lower frequency as bootstrapped confidence intervals can be costly to compute.

The primary purpose of this view is giving users a sense about when to dive in optimistically to the next analytic phase. Users might decide that results are "good enough" to proceed when the mean effect size chart shows a stable trend with a narrow confidence interval that does not overlap with the critical value (*e.g.,* zero). Similarly, in the decision sensitivity chart, the sensitivity scores of important decisions might need to be well-separated, with non-overlapping confidence intervals.

5.4.2 Diagnosing Issues

The bottom part of the dashboard consists of multiple coordinated views to facilitate users in identifying potential issues in the multiverse.

Decision Space View

The decision space view (Figure 5-5c) visualizes the decisions and their relationship to support reflection on the decision space structure (T1) and to contextualize subsequent tasks. The visualization is present before any universes are run, giving users a chance to review the decisions and potentially go back to revise the multiverse specification early on.

We visualize the decision space using an adapted version of the Analytic Decision Graphs(Section 3.2), as it effectively depicts decisions in the context of the analysis process. In the graph, nodes represent decisions, with a larger size indicating a higher cardinality and a darker color indicating a higher sensitivity. Edges depict relationships between decisions in the analysis process, including order (light gray edge) and procedural dependency (black arrow). Options of a decision are shown besides the decision node. The underlying data for the graph is extracted automatically from the multiverse specification written by the author in the Boba DSL (Chapter 4). Decision sensitivity is not available before runtime, but is gradually updated as the multiverse executes.

Main Effect View

The main effect view (Figure 5-5d) visualizes outcomes from individual universes as a dot plot, with one-to-one mapping between a dot and a universe. The view serves three main purposes. First, it conveys the range of possible outcomes for users to spot potential issues of abnormal outcome values (T4). Second, it supports an overview of other type of issues, including errors and model quality, by encoding the corresponding aspect of a universe using color (T4). Third, brushing and linking interactions with other views enable users to connect decisions to a particular issue and gain a better understanding of the responsible decisions (T6).

In the dot plot, the x-axis represents the magnitude of the outcome value and the y-axis represents count. Outlying point estimates might be observed at the left and right extremes of the x-axis. To bring attention to universes that do not produce an outcome (often due to errors causing the execution to fail), we show them in a special bin at the left end of the x-axis. The color scheme of the dots encodes either error information or model quality, and can be toggled in a dropdown menu (Figure 5-6c). A categorical scheme (Figure 5-5d) encodes error information, indicating if a universe stopped with errors (red), completed with warnings (yellow), or completed without warnings (blue). Alternatively, a sequential color scheme (Figure 5-6a) maps to a quantitative model quality metric, such as R-squared, with a lighter blue indicating poorer model quality. The color encoding gives an overview of the issue and invites direct manipulation interactions to inspect details.

This view supports two types of brushing and linking interactions with other views. First, brushing the main effects view filters the diagnostics to the selected universes. The error message view and the model quality view are updated accordingly to detailed diagnostics restricted to the selected results. Second, clicking a decision node in the decision view (Figure 5-5c) splits the main effects into a small multiples plot,

with each subplot mapping to a different option of the decision (Figure 5-5d). The small multiples enable analysts to compare between options and identify options that potentially lead to a particular issue. For example, an option might exclusively produce outlying main effect estimates, runtime errors, or poor-fitting models. Finding the responsible decision gives users actionable information to improve the multiverse. Users might realize that certain options are unreasonable and exclude them from the multiverse, or otherwise narrow the issue down to a smaller subset of the multiverse for further diagnosis.

Error Message View

The error message view (Figure 5-5e) compiles a list of distinct error and warning messages from the runtime to give users more details on implementation issues (T4). The messages are aggregated since the same runtime error often gets repeated across multiple universe scripts. This is because universe scripts are highly similar, for instance two adjacent universes that differ in only one decision may share the same source code except for the value of a variable.

To extract the error messages, the system first collects the outputs of universe scripts in the standard error stream. These outputs are then simplified and aggregated using heuristics that look for specific keywords such as *warning*. The heuristics take advantage of the shared logging format of a particular programming language, for example a Python program typically prints a stack trace that starts with the word *Traceback*. We find the heuristics work well in practice, though a more generalizable method could apply sentence embeddings [10] to compute the similarity between error messages and look for repeated messages.

In the interface, we assign a categorical color scheme according to the type of the message: red for errors that terminate the program and produce a non-zero exit code, and yellow for warnings. The messages are ordered first by their type (*i.e.,* errors before warnings) and then by the number of universes that share the issue, to prioritize more important messages at the top. As described before, brushing the main effect view filters messages to be those that occur in the selected universes. In addition, clicking a message updates the main effect view to highlight universes with that specific message.

Model Quality View

The model quality view (Figure 5-6b) supports an in-depth assessment of the model quality of individual universes using visual predictive checks (T4). A dropdown menu (Figure 5-6c) toggles between the model quality view and the error message view. The predictive check allows users to detect systematic discrepancies between the model and observed data in order to assess the fit of the model to the data [29]. It compares the replicated dataset from predictions of the fitted model to the observed dataset. We facilitate the qualitative comparison by plotting the predicted and observed data distributions side-by-side as two violin plots. We also overlay a representative subset of individual data points to convey further details. This subset is derived by sam-

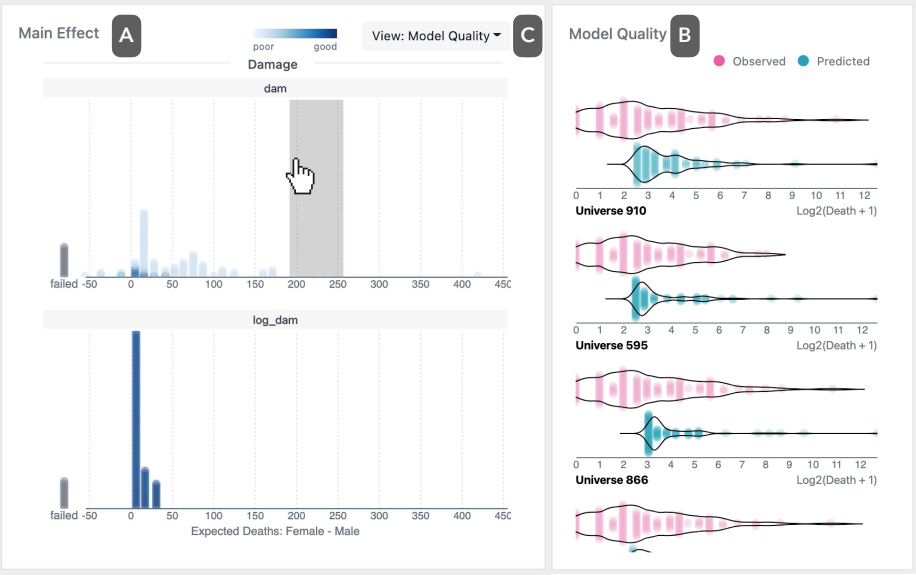


Figure 5-6: To enable an overview of model quality, universes are colored according to a quantitative model quality metric, with a lighter blue indicating a poorer fit (a). Brushing the main effect view populates the model quality view (b) with visual predictive checks that compare predicted data (blue) with observed data (pink). Users toggle between error information and model quality using a dropdown menu (c).

pling in each percentile of the corresponding distribution and visualized as a centered quantile dot plot [56].

5.5 Conclusion

This chapter presents methods for reducing the delay from having a multiverse specification to checking intermediate results and potential issues. We adopt an online approach and investigate three sampling-based approximation algorithms. Empirical evaluation on synthetic and real multiverses show that round robin and sketching approaches are 5 times faster on average to estimate decision sensitivity accurately compared to no sampling, and up to 2 times faster than uniform sampling. Next, we develop a monitoring dashboard that leverages sampling algorithms under the hood. The interface facilitates users in choosing a "good enough" snapshot for early interpretation of multiverse results, as well as diagnosing runtime errors and model quality issues.

Chapter 6

The Boba Visualizer: Interpreting Multiverse Analyses

After running the multiverse analysis, analysts face additional challenges in interpreting the outcomes of a vast number of universes. Besides gauging the overall robustness of the findings, researchers often seek to understand what decisions are critical in obtaining particular outcomes. The simplest approach is a table [97, 11, 85, 17] or a matrix [95] where rows and columns map to decisions, and cells represents outcomes from individual universes. But the patterns of how outcomes vary might be difficult to identify depending on the spatial arrangements of rows and columns. As multiple decisions might interact, understanding the nuances in how decisions affect robustness will require a comprehensive exploration, suggesting a need for an interactive interface.

In this chapter, we introduce a visual analysis system called the Boba Visualizer for reviewing multiverse analyses (Figure 6-1). The Boba Visualizer facilitates assessment of the output of all analysis paths. The system first provides linked views of both analysis results and the multiverse decision space to enable a systematic exploration of how decisions do (or do not) impact outcomes. Besides decision sensitivity, we enable users to take into account sampling uncertainty and model fit by comparing observed data with model predictions [28]. After viewing the results, users can exclude models poorly suited for inference by adjusting a model fit threshold, or adopt a principled approach based on model averaging to incorporate model fit in inference.

6.1 Requirements

We identify tasks that our tool should support from prior literature and our past experiences conducting multiverse analyses.

The primary task in prior work (Section 2.2.1) is understanding the robustness of results across all reasonable specifications. If robustness checks indicate conflicting conclusions, a natural follow-up task is to identify what decisions are critical to reaching a particular conclusion or what decisions produce large variations in results.

We also propose new tasks to cover potential blind spots in prior work. First,

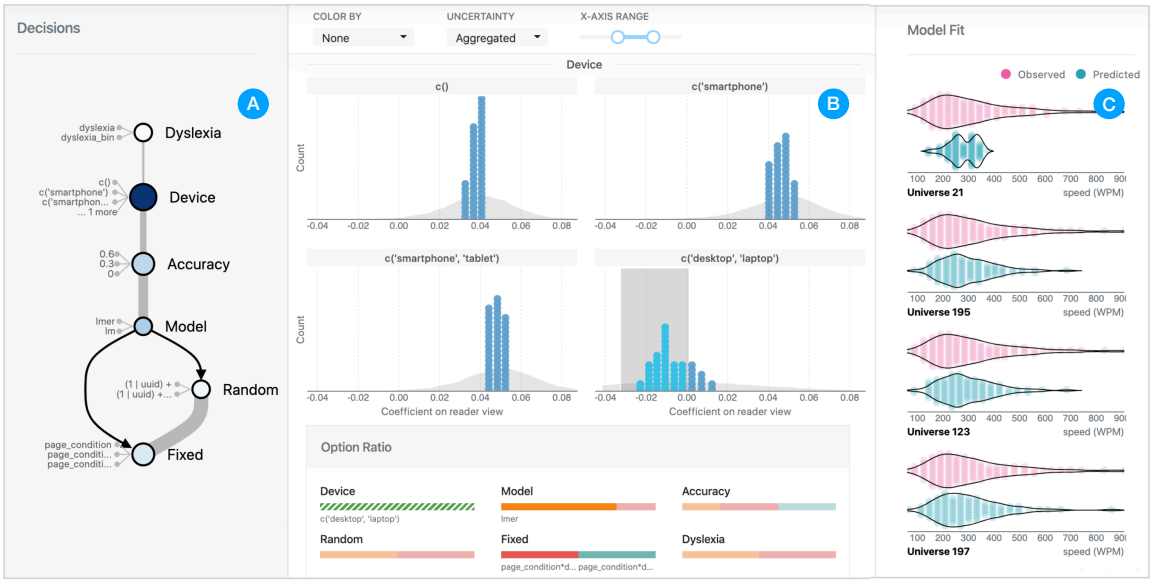


Figure 6-1: Visualizing multiverse analyses with Boba. Users start with a graph of analytic decisions (a), where sensitive decisions are highlighted in darker blues. Clicking a decision node allows users to compare point estimates (b, blue dots) and uncertainty distributions (b, gray area) between different alternatives. Users may further drill down to assess the fit quality of individual models (c) by comparing observed data (pink) with model predictions (teal).

besides point estimates, a tool should convey appropriate uncertainty information to help users gauge the end-to-end uncertainty caused by both sampling and decision variations, and compare the variance between conditions. Second, it is important to assess the model fit quality to distinguish trustworthy models from the ones producing questionable estimates. Uncertainty information and fit issues become particularly important during statistical inference. Users should be able to propagate uncertainty in the multiverse to support judgments about the overall reliability of effects, and they should be able to refine the multiverse to exclude models with fit issues before making inferences.

We identify six tasks that our visual analysis system should support:

- T1: *Decision Overview* – gain an overview of the decision space to understand the multiverse and contextualize subsequent tasks.
- T2: *Robustness Overview* – gauge the overall robustness of findings obtained through all reasonable specifications.
- T3: *Decision Impacts* – identify what combinations of decisions lead to large variations in outcomes, and what combinations of decisions are critical in obtaining specific outcomes.
- T4: *Uncertainty* – assess the end-to-end uncertainty as well as uncertainty associated with individual universes.
- T5: *Model Fit* – assess the model fit quality of individual universes to distinguish trustworthy models from questionable ones.

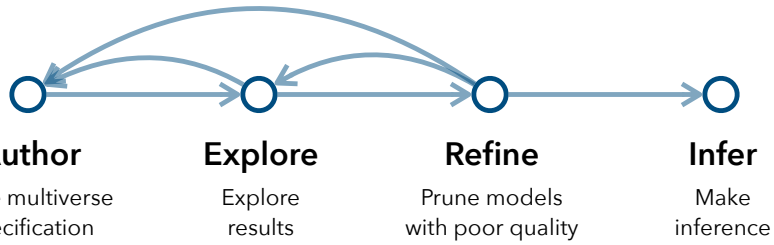


Figure 6-2: The intended workflow for multiverse analysis in Boba.

- T6: *Inference* – perform statistical inference to judge how reliable the hypothesized effect is, while accounting for model quality.

Besides the tasks, our system should also support the following data characteristics (S1) and types of statistical analyses (S2). First, our visual encoding should be scalable to large multiverses and large input datasets. This is because the multiverse size increases exponentially with the number of decisions, with the median size in practice being in the thousands [66]. The input datasets might also have arbitrarily many observations. Second, we should support common simple statistical tests in HCI research [81], including ANOVA and linear regressions.

6.1.1 Workflow

We propose a general workflow for multiverse analysis with four stages (Figure 6-2). In this workflow, users *author* the multiverse specification, *explore* the results, *refine* the multiverse by pruning universes with poor model quality, and make *inference*. Users should be free to cycle between the first three stages, because upon exploring the results, users might discover previously overlooked alternatives, or notice that certain decisions are poorly suited for inference. In this case, they might iterate on their multiverse specification to include only decisions resulting in universes that seem "reasonable". However, once users proceed to the *inference* stage, they should not return to any of the prior stages.

6.2 System Walkthrough

We present the system features and design choices in a fictional usage scenario where Emma, an HCI researcher, uses the visualizer to explore a multiverse on data collected in her experiment. We construct the multiverse based on how the authors of a published research article [63] might analyze their data, but the name "Emma" and her workflow are fictional.

Emma runs an experiment to understand whether "Reader View" – a modified web page layout – improves reading speed for individuals with dyslexia. She assigns participants to use Reader View or standard websites, measures their reading speed, and collects other variables such as accuracy, device, and demographic information. She plans to build a model with reading speed as the dependent variable. To check

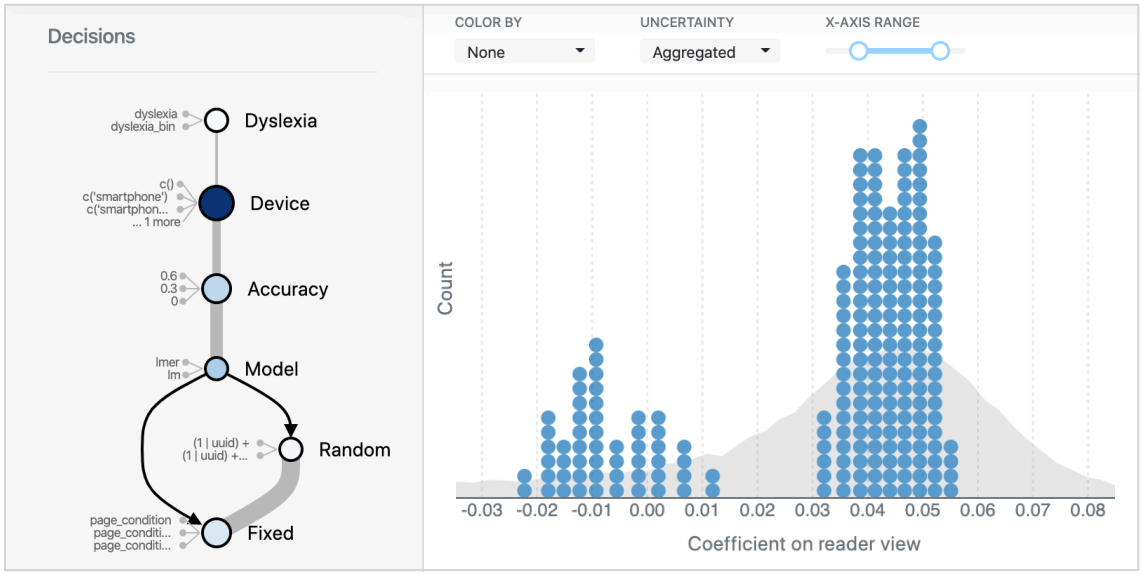


Figure 6-3: **Decision view and outcome view**. (a) The decision view shows analytic decisions as a graph with order and dependencies between them, and highlights more sensitive decisions in darker colors. (b) The outcome view visualizes outputs from all analyses, including individual point estimates and aggregated uncertainty.

whether her conclusion depends on idiosyncratic specifications, Emma identifies six analytic decisions, including the `device` type and `accuracy` cutoff used to filter participants, ways to operationalize `dyslexia`, the statistical `model`, and its `random` and `fixed` terms. She then writes a multiverse specification in the Boba DSL, compiles it to 216 analysis scripts, and runs all scripts to obtain a set of effect sizes. She loads these outputs into the Boba Visualizer.

6.2.1 Outcome View

On system start-up, Emma sees an overview distribution of point estimates from all analyses (Figure 6-3b). The majority of the coefficients are positive, but a smaller peak around zero suggests no effect.

The outcome view visualizes the final results of the multiverse, including point estimates (*e.g.,* model coefficient of `reader view`, the independent variable encoding experimental conditions) and uncertainty information. By default, the chart contains outcomes from all universes in order to show the overall robustness of the conclusion (T2).

Boba visualizes one point estimate from each universe using a density dot plot [109] (Figure 6-3b, blue dots). The x-axis encodes the magnitude of the estimate; dots in the same bin are stacked along the y-axis. To accommodate large multiverses (S1), we allow dots to overlap along the y-axis, which represents count. Density dot plots more accurately depict gaps and outliers in data than histograms [109]. One-to-one

mapping between dots and universes affords direct manipulation interactions such as brushing and details-on-demand, as we introduce later.

Boba visualizes end-to-end uncertainty from both sampling and decision variations (T4) as a background area chart (Figure 6-3b, gray area). When the uncertainty introduced by sampling variations is negligible, the area chart follows the dot plot distribution closely. In contrast, the mismatch of the two distributions in Figure 6-3b indicates considerable sampling uncertainty. We compute the end-to-end uncertainty by aggregating over modeling uncertainty from all universes. Specifically, we first calculate $\hat{f}(x) = \sum_{i=1}^{N} f_i(x)$, where $f_i(x)$ is the sampling distribution of the i-th universe, and N is the overall multiverse size. Then, we scale the height of the area chart such that the total area under $\hat{f}(x)$ is approximately the same as the total area of dots in the dot plot.

Besides aggregated uncertainty, Boba allows users to examine uncertainty from individual universes (Figure 6-5). In a dropdown menu, users can switch to view the probability density functions (PDFs) or cumulative distribution functions (CDFs) of all universes. A PDF is a function that maps the value of a random variable to its likelihood, whereas a CDF gives the area under the PDF. In both views, we draw a cubic basis spline for the PDF or CDF per universe, and reduce the opacity of the curves to visually "merge" the curves within the same space. There is again a one-to-one mapping between a visual element and a universe to afford interactions. To help connect point estimates and uncertainty, we draw a strip plot of point estimates beneath each PDFs/CDFs chart (Figure 6-5, blue dashes), and show the corresponding sampling distribution PDF when users mouse over a universe in the dot plot.

6.2.2 Decision View

As the overall outcome distribution suggests conflicting conclusions, Emma wants to investigate what decisions lead to changes in results. She first familiarizes herself with the available decisions.

The decision view shows a graph of analytic decisions in the multiverse, along with their order and dependencies (Figure 6-3a), helping users understand the decision space and inviting further exploration (T1).

We adapt the design of Analytic Decision Graphs (Section 3.2) to show decisions in the context of the analysis process. Nodes represent decisions and edges represent the relationships between decisions: light gray edges encode *temporal order* (the order that decisions appear in analysis scripts) and black edges encode *procedural dependencies*. To avoid visual clutter, we aggregate the information about alternatives, using the size of a node to represent the number of alternatives and listing a few example alternative values besides a node. Compared to viewing decisions in isolation, this design additionally conveys the analysis pipeline to help users better reason with the ramifications of a decision.

The underlying data structure for the graph is inferred from the Boba DSL specification. We infer decision names from variable identifiers. We extract temporal order as the order that decision points are first used in the specification, and detect procedural dependencies from user-specified constraints and code graph structure. After

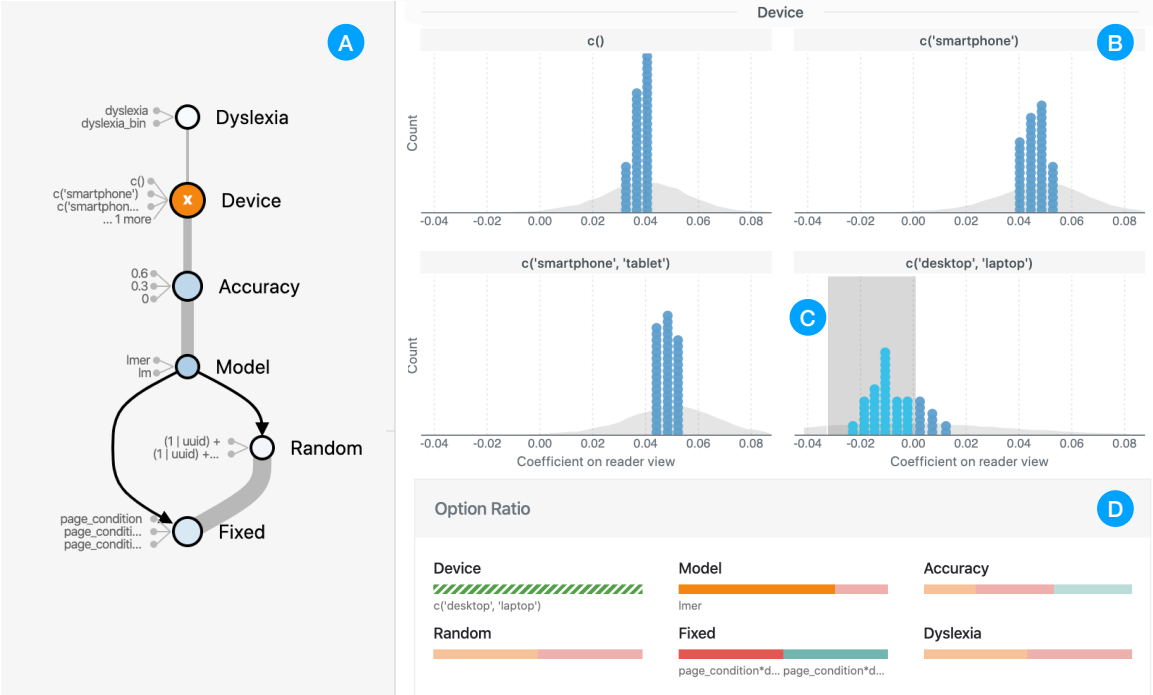


Figure 6-4: **Facet and Brushing**. Clicking a node in the decision view (a) divides the outcome view into a trellis plot (b), answering questions like "does the decision lead to large variations in effect size?" Brushing a region in the outcome view (c) reveals dominant alternatives in the option ratio view (d), answering questions like "what causes negative results?"

we extract the data structure, we apply a Sugiyama-style [98] flow layout algorithm, as implemented in Dagre [80], to compute the graph layout.

Sensitivity

When viewing the decision graph, Emma notes a sensitive decision "Device" which is highlighted in a darker color (Figure 6-3a).

To highlight decisions that lead to large changes in analysis outcomes (T3), we compute the marginal sensitivity per decision and color the nodes using a sequential color scale. The color encoding helps draw the user's attention to consequential decisions to aid initial exploration.

As discussed in Section 5.2.3, we evaluated four candidate methods for estimating sensitivity, and recommended the use of K-samples Anderson–Darling (AD) test. Boba uses AD test by default, while users can override the default in the config file.

6.2.3 Facet and Brushing

Seeing that the decision "Device" has a large impact, Emma clicks on the node to further examine how results vary (Figure 6-4a). She finds that one condition

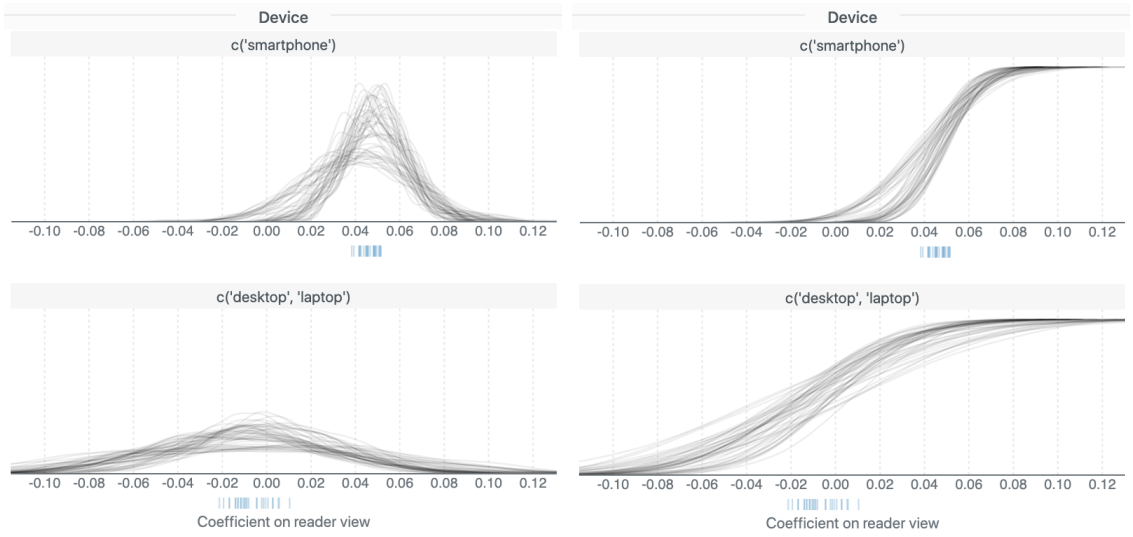


Figure 6-5: **PDFs** (a) and **CDFs** (b) **views** visualize sampling distributions from individual universes. Toggling these views in a trellis plot allows users to compare the variance between conditions.

exclusively produces point estimates around zero (Figure 6-4b) and it also has a much larger variance (Figure 6-5).

Clicking a node in the decision graph facets the outcome distribution into a trellis plot, grouping subsets of universes by shared decision alternatives. This allows users to systematically examine the trends and patterns caused by a decision (T3). Akin to the overall outcome distribution, users can toggle between point estimates and uncertainty views to compare the variance between conditions. The trellis plot can be further divided on a second decision by shift-clicking a second node to show the interaction between two decisions. With faceting, users may comprehensively explore the data by viewing all univariate and bivariate plots. Sensitive decisions are automatically highlighted, so users might quickly find and examine consequential decisions as well.

What decisions lead to negative estimates? Emma brushes negative estimates in a subplot (Figure 6-4c) and inspects option ratios (Figure 6-4d).

Brushing a region in the outcome view updates the option ratio view. The option ratio view shows percentages of decision options to reveal dominating alternatives that produce specific results (T3).

The option ratio view visualizes each decision as a stacked bar chart, where bar segment length encodes the percentage of results coming from an alternative. When the user brushes a range of results, the bars are updated accordingly to reflect changes, and dominating alternatives (those having a higher percentage than default) are highlighted. For example, Emma notices that the `lmer` model (*i.e.,* linear mixed-effect

model in R) and two sets of fixed effects are particularly responsible for the negative outcomes in Figure 6-4c. We color the bar segments using a categorical color scale to make bars visually distinguishable. Decisions used to divide a trellis plot are marked by a striped texture, as each trellis subplot only contains one alternative by definition.

6.2.4 Model Fit View

Now that Emma understands what decisions lead to null effects, she wonders if these results are from trustworthy models. She changes the color-by field to get an overview of model fit quality (Figure 6-6a) and sees that the universes around zero have a poorer fit. She then uses a slider to remove universes that fail to meet a quality threshold (Figure 6-6b).

Boba enables an overview of model fit quality across all universes (T5) by coloring the outcome view with a model quality metric (Figure 6-6a). We use normalized root mean squared error (NRMSE) to measure model quality and map NRMSE to a single-hue colormap of blue shades where a darker blue indicates a better fit.

To obtain NRMSE, we first compute the overall mean squared prediction error (MSE) from a k-fold cross validation:

$$MSE = \frac{1}{k}\sum_{j=1}^{k}\frac{1}{n_j}\sum_{i=1}^{n_j}(y_i - \hat{y}_i)^2$$

where k is the number of folds (we set $k = 5$ in all examples), n_j is the size of the test set in the j-th fold, y_i is the observed value, and $\hat{y}_i$ is the predicted value. We then normalize the MSE by the span of the maximum y_{max} and minimum y_{min} values of the observed variable:

$$NRMSE = \sqrt{MSE}/(y_{max} - y_{min})$$

We use k-fold cross validation [103] because metrics such as Akaike Information Criterion cannot be used to compare model fit across classes of models (e.g., hierarchical vs. linear) [31]. Prior work shows that cross validation performs better in estimating predictive density for a new dataset than information criteria [103], suggesting that it is a better approximation of out-of-sample predictive validity.

To further investigate model quality, Emma drills down to individual universes by clicking a dot in the outcome view. She sees in the model fit view (Figure 6-1c) that a model gives largely mismatched predictions.

Clicking a result in the outcome view populates the model fit view with visual predictive checks, which show how well predictions from a given model replicate the empirical distribution of observed data [28], allowing users to further assess model quality (T5). The model fit visualization juxtaposes violin plots of the observed data and model predictions to facilitate comparison of the two distributions (see Figure 6-1c). Within the violin plots, we overlay observed and predicted data points as centered density dot plots to help reveal discrepancies in approximation due to kernel density estimation. When the number of observations is large (S1), we plot a representative subset of data, sampled at evenly spaced percentiles, as centered

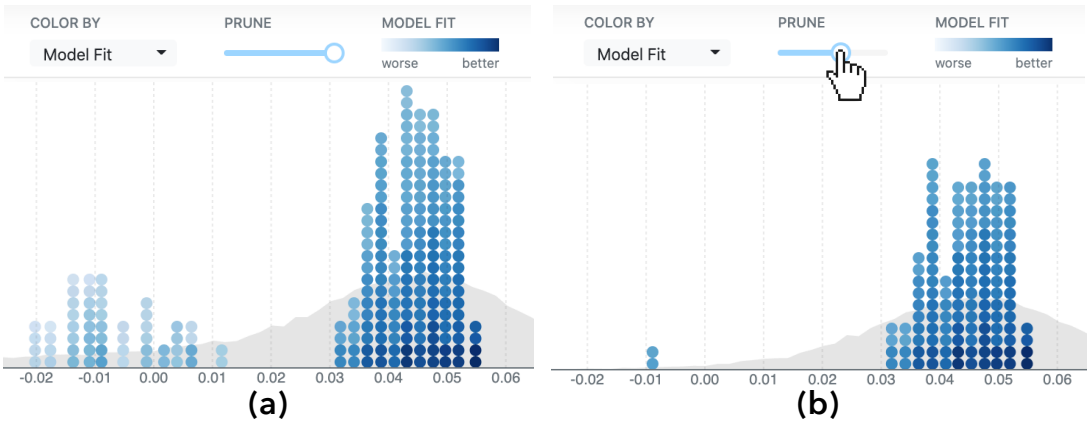


Figure 6-6: (a) Coloring the universes according to their model fit quality. (b) Removing universes that fail to meet a model quality threshold.

quantile dotplots [56]. As clicking individual universes can be tedious, the model fit view suggests additional universes that have similar point estimates to the selected universe.

6.2.5 Inference

After an in-depth exploration, Emma proceeds to the final step, asking "given the multiverse, how reliable is the effect?" She confirms a warning dialog to arrive at the inference view (Figure 6-7).

To support users in making inference and judging how reliable the hypothesized effect is (T6), Boba provides an inference view at the end of the analysis workflow, after users have engaged in exploration. Once in the inference view, all earlier views and interactions are inaccessible to avoid multiple comparison problems [114] arising from repeated inference. The inference view contains different plots depending on the outputs from the authoring step, so that users can choose between robust yet computationally-expensive methods and simpler ones.

A more robust inference utilizes the null distribution – the expected distribution of outcomes when the null hypothesis of no effect is true. In this case, the inference view shows an aggregate plot followed by a detailed plot (Figure 6-7ab). The aggregate plot (Figure 6-7a) compares the null distribution (red) to possible outcomes of the actual multiverse (blue) across sampling and decision variations. The detailed plot (Figure 6-7b) shows point estimates (colored dots) against 95% confidence intervals representing null distributions (gray lines) for each universe. Each point estimate is orange if it is outside the range, or blue otherwise. Underneath both plots, we provide descriptions (Figure 7-4) to guide users in interpretation: For the aggregate plot, we prompt users to compare the distance between the averages of the two densities to the spread. For the detailed plot, we count the number of universes with the point estimate outside its corresponding range. If the null distribution is unavailable, Boba shows a simpler aggregate plot (Figure 6-7c) where the expected effect size under the null hypothesis is marked with a red line.

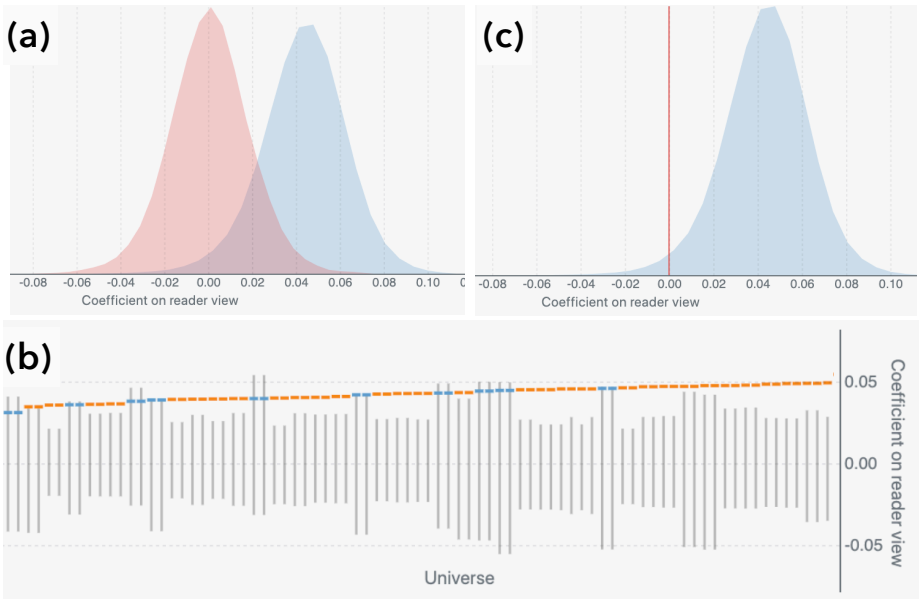


Figure 6-7: **Inference views.** (a) Aggregate plot comparing the possible outcomes of the actual multiverse (blue) and the null distribution (red). (b) Detailed plot showing the individual point estimates and the range between the 2.5th and 97.5th percentile in the null distribution (gray line). Point estimates outside the range are colored in orange. (c) Alternative aggregate plot where a red line marks the expected null effect.

To compute the null distribution, we permute the data with random assignment [95]. Specifically, we shuffle the column with the independent variable (`reader view` in this case) N times, run the multiverse of size M on each of the shuffled datasets, and obtain $N \times M$ point estimates. As `reader view` and `speed` are independent in the shuffled datasets, these $N \times M$ point estimates constitute the null distribution.

In addition, Boba enables users to propagate concerns in model fit quality to the inference view in two possible ways. The first way employs a model averaging technique called *stacking* [112] to take a weighted combination of the universes according to their model fit quality. The technique learns a simplex of weights, one for each universe model, via optimization that maximizes the log-posterior-density of the held-out data points in a k-fold cross validation. Boba then takes a weighted combination of the universe distributions to create the aggregate plot. While stacking provides a principled way to approach model quality, it can be computationally expensive. As an alternative, Boba excludes the universes below the model quality cutoff users provide in Section 6.2.4. The decisions of the cutoff and whether to omit the universes are made before a user enters the inference view.

6.3 Conclusion

This chapter presents the Boba Visualizer, a visual analysis system for interpreting multiverse analyses. The visual analysis system provides linked views between analytic decisions and model estimates to facilitate systematic exploration of how decisions impact robustness, along with views for sampling uncertainty and model fit. We also provide facilities for principled pruning of "unreasonable" specifications, and support inference to assess effect reliability. Boba is available as open-source software at `https://github.com/uwdata/boba`.

Chapter 7

Case Studies

We evaluate Boba in three case studies. The first two case studies use multiverses replicated from prior work [95, 113]. We first show how the Boba Visualizer affords multiverse interpretation, enabling a richer understanding of robustness, decision patterns, and model fit quality via visual inspection. In both case studies, model fit visualizations surface previously overlooked issues and change what one can reasonably take away from these multiverses. Then, we use the first case study to demonstrate that by running a small subset of a multiverse, we can arrive at the same conclusion about decision sensitivity and model quality as in the full multiverse.

The third case study builds upon a crowdsourced data analysis initiative, where 29 analyst teams independently analyzed the same datasets to answer two research questions. We capture the various analytic decisions made by crowdsourced analysts into two multiverse specifications, in order to disentangle which analytic decisions are most responsible for variability of the resulting effect sizes. We implement the multiverses using the Boba DSL, diagnose errors and iterate on the specification using the Boba Monitor, and interpret the final results using the Boba Visualizer.

Finally, we discuss the implications of model quality issues in our design reflections.

7.1 Case Study: Mortgage Analysis

Young *et al.* [113] propose a multimodel analysis approach to gauge whether model estimates are robust to alternative model specifications. Akin to the philosophy of multiverse analysis, the approach takes all combinations of possible control variables in a statistical model. The outputs are multiple summary statistics, including (1) an overall *robustness ratio*, (2) *uncertainty* measures for sampling and modeling variations, and (3) metrics reflecting the *sensitivity* of each variable.

As an example, the authors present a case study on mortgage lending, asking "are female applicants more likely to be approved for a mortgage?" They use a dataset of publicly disclosed loan-level information about mortgage, and fit a linear regression model with mortgage application acceptance rate as the dependent variable and female as one independent variable. In their multimodel analysis, they test different control variables capturing demographic and financial information. The

Figure 7-1: A case study on how model estimates are robust to control variables in a mortgage lending dataset. (a) Decision view shows that *black* and *married* are two consequential decisions. (b) Overall outcome distribution follows a multimodal distribution with three peaks. (c) Trellis plot of *black* and *married* indicates the source of the peaks. (d) Model fit plots show that models produce numeric predictions while observed data is categorical. (e) PDFs of individual sampling distributions show significant overlap of the three peaks.

resulting summary statistics indicate that the estimate is not robust to modeling decisions with large end-to-end uncertainty, and two control variables, *married* and *black*, are highly influential. These summary statistics provide a powerful synopsis, but may fail to convey more nuanced patterns in result distributions. The authors manually create additional visualizations to convey interesting trends in data, for instance the estimates follow a multimodal distribution. These visualizations, though necessary to provide a richer understanding of model robustness, are ad-hoc and not included in the software package.

We replicate the analysis in Boba. The Boba DSL specification simply consists of eight placeholder variables, each indicating whether to include a control variable in the model formula. The specification compiles to 256 scripts.

7.1.1 Evaluating the Boba Visualizer

We run all scripts to completion and start the Boba Visualizer. We demonstrate how analysts might quickly arrive at insights provided by summary statistics in prior work, while at the same time gaining a richer understanding of robustness patterns. We also show that by incorporating uncertainty and model fit checks, Boba surfaces potential issues that prior work might have neglected.

We first demonstrate that the default views in the Boba Visualizer afford similar insights on uncertainty, robustness, and decision sensitivity. Upon launching the

visualizer, we see a decision graph and an overall outcome distribution (Figure 7-1). The decision view (Figure 7-1a) highlights two sensitive decisions, *black* and *married*. The outcome view (Figure 7-1b) shows that the point estimates are highly varied with conflicting implications. The aggregated uncertainty in the outcome view (Figure 7-1b, background gray area) has a wide spread, suggesting that the possible outcomes are even more varied when taking both sampling and decision variability into account. These observations agree with the summary metrics in previous work, though Boba uses a different, non-parametric method to quantify decision sensitivity, as well as a different method to aggregate end-to-end uncertainty.

The patterns revealed by ad-hoc visualizations in previous work are also readily available in the Boba Visualizer, either in the default views or with two clicks guided by prominent visual cues. The default outcome view (Figure 7-1b) shows that the point estimates follow a multimodal distribution with three separate peaks. Clicking the two highlighted (most sensitive) nodes in the decision view (Figure 7-1a) produces a trellis plot (Figure 7-1c), where each subplot contains only one cluster. From the trellis plot, it is evident that the leftmost and rightmost peaks in the overall distribution come from two particular combinations of the influential variables. Alternatively, users might arrive at similar insights by brushing individual clusters in the default outcome view.

Finally, the uncertainty and model fit visualizations in Boba surface potential issues that previous work might have overlooked. First, though the point estimates in Figure 7-1b fall into three distinct clusters, the aggregated uncertainty distribution appears unimodal despite a wider spread. The PDF plot (Figure 7-1e) shows that sampling distribution from one analysis typically spans the range of multiple peaks, thus explaining why the aggregated uncertainty is unimodal. These observations suggest that the multimodal patterns exhibited by point estimates are not robust when we take sampling variations into account. Second, we assess model fit quality by clicking a dot in the outcome view and examining the model fit view (Figure 7-1d). As shown in Figure 7-1d, while the observed data only takes two possible values, the linear regression model produces a continuous range of predictions. It is clear from this visual check that an alternative model, for example logistic regression, is more appropriate than the original linear regression models, and we should probably interpret the results with skepticism given the model fit issues. These observations support our arguments in Section 6.1 that uncertainty and model fit are potential blind spots in prior literature.

7.1.2 Evaluating the Boba Monitor

Next, we demonstrate that by running a small subset of a multiverse, we can arrive at the same conclusion about decision sensitivity and model quality as above.

We load the same multiverse specification into the Boba Monitor. After starting the runtime, we observe the change in the estimates of decision sensitivity and mean effect size, and pause the runtime when these estimates seem to reasonably converge. Figure 7-2 shows the snapshot when we pause, with 46 universes completed so far. The mean effect size estimate (Figure 7-2b) fluctuates slightly at the beginning, but

Figure 7-2: A case study on mortgage analysis. With less than one fifth of the multiverse completed, the sampling estimates converge reasonably: sensitive decisions are distinct from non-sensitive ones (a), and the mean effect size estimate remains stable (b). The two sensitive decisions (c), *black* and *married*, agree with prior work [113]. However, model quality checks show an unsatisfactory quality throughout (d) and indicate a large mismatch between observed and predicted data (e).

soon remains stable around the value of 2.2, with a reasonably narrow confidence interval spanning roughly 1.6 to 2.6. The decision sensitivity estimates (Figure 7-2a) show that two decisions are sensitive while others are not, and the confidence intervals of the sensitive decisions do not overlap with those of the non-sensitive decisions. The decision graph confirms that the sensitive decisions are *black* and *married* (Figure 7-2c), which agree with prior work. We may also identify the sensitive decisions by hovering over the time series in the progress view. This exercise shows that an optimistic analysis at 18% of the full multiverse size will not lead to a qualitatively different interpretation of decision sensitivity than running every universe.

Before proceeding to further analysis and running the multiverse to completion, we sanity-check possible validity issues. The main effect view shows that all model coefficients are within a sensible range. The error message view shows that no runtime errors or warnings have occurred so far. We then switch to examine the model quality, which we quantify using adjusted R^2 as all universes use a linear model. As indicated by the light color (Figure 7-2d), the model quality of all completed analyses is unanimously bad. The visual predictive checks (Figure 7-2e) show that the observed data is binary while the model prediction is continuous, suggesting that we need to revise the multiverse specification with a more appropriate model such as

Figure 7-3: A case study on whether hurricanes with more feminine names have caused more deaths. (a) The majority of point estimates suggest a small, positive effect, but there are considerable variations. (b) Faceting and brushing reveal decision combinations that produce large estimates. Coloring by model quality shows that large estimates are from questionable models, and predictive checks (c) confirms model fit issues. (d) Inference view shows that the observed and null distributions are different in terms of mode and shape, yet with highly overlapping estimates.

logistic regression. Importantly, we are able to identify the model quality issue before one fifth of the multiverse finished running (or as early as the first few universes, if we check earlier), reducing the time to iteration.

7.2 Case Study: Female Hurricanes Caused More Deaths?

We replicate another multiverse where Simonsohn *et al.* [95] challenged a previous study [52]. The original study [52] reports that hurricanes with female names have caused more deaths, presumably because female names are perceived as less threatening and lead to less preparation. The study used archival data on hurricane fatalities and regressed death count on femininity. However, the study led to a heated debate on proper ways to conduct the data analysis. To understand if the conclusion is robust to alternative specifications, Simonsohn *et al.* identified seven analytic decisions, including alternative ways to exclude outliers, operationalize femininity, select

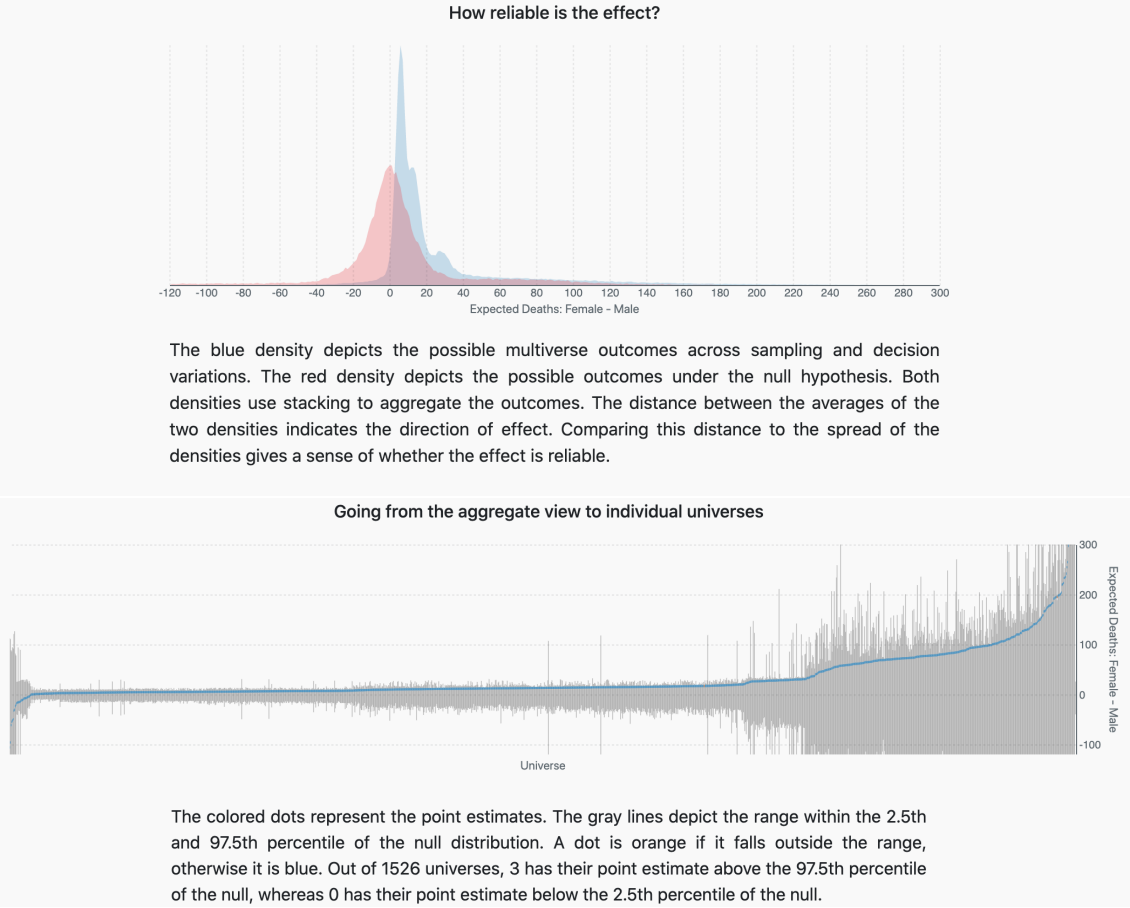


Figure 7-4: The inference view of the hurricane case study. The top plot shows that the two distributions are different in terms of mode and shape, yet they are highly overlapping, which suggests the effect is not reliable. The bottom plot shows that the vast majority of the universes have the point estimate within the 2.5th and 97.5th percentile of the corresponding null distribution.

the model type, and choose covariates. They then conducted a multiverse analysis and interpreted the results in a visualization called a *specification curve*.

We build the same multiverse using these seven analytic decisions in Boba. In the Boba DSL specification, we use a decision block to specify two alternative model types: negative binomial regression versus linear regression with log-transformed deaths as the dependent variable. The rest of the analytic decisions are placeholder variables that can be expressed as straightforward value substitutions. However, the two different model types lead to further differences in extracting model estimates. For example, we must invert the log-transformation in the linear model to obtain predictions in the original units. We create additional placeholder variables for implementation differences related to model types and link them with the model decision block. The specification compiles to 1,728 individual scripts.

We then interpret the results using the Boba Visualizer. As shown in the overview distribution (Figure 7-3a), the majority of point estimates support a small, positive

effect (female hurricanes lead to more deaths, and the extra deaths are less than 20), while some estimates suggest a larger effect. A small fraction of results have the opposite sign.

What analytic decisions are responsible for the variations in the estimates? The decision view indicates that multiple analytic decisions might be influential (Figure 7-3a). We click on the relatively sensitive decisions, *outliers*, *damage* and *model*, to examine their impact. In the corresponding univariate trellis plots (Figure 7-8), certain choices tend to produce larger estimates, such as not excluding any outliers, using raw damage instead of log damage, and using negative binomial regression. However, in each of these conditions, a considerable number of universes still support a smaller effect, suggesting that it is not a single analytic decision that leads to large estimates.

Next, we click on two influential decisions to examine their interaction. In the trellis plot of *model* and *damage* (Figure 7-3b), one combination (choosing both log damage and negative binomial model) produces mostly varied estimates without a dominating peak next to zero. Brushing the large estimates in another combination (raw damage and linear model) indicates that these results are coming from specifications that additionally exclude no outliers. Removing these decision combinations will eliminate the possibility of obtaining a large effect.

But do we have evidence that certain outcomes are less trustworthy? We toggle the color-by drop-down menu so that each universe is colored by its model quality metric (Figure 7-3b). The large estimates are almost exclusively coming from models with a poor fit. We further verify the model fit quality by picking example universes and examining the model fit view (Figure 7-3c). The visual predictive checks confirm issues in model fit, for example the models fail to generate predictions smaller than 3 deaths, while the observed data contains plenty such cases.

Now that we have reasons to be skeptical of the large estimates, the remaining universes still support a small, positive effect. How reliable is the effect? We proceed to the inference view to compare the possible outcomes in the observed multiverse and the expected distribution under the null hypothesis (Figure 7-3d). The two distributions are different in terms of mode and shape, yet they are highly overlapping, which suggests the effect is not reliable. The detail plot depicting individual universes (Figure 7-4) further confirms this observation. Out of the entire multiverse, only 3 universes have point estimates outside the 2.5th and 97.5th percentile of the corresponding null distribution.

7.3 Case Study: Gender and Professional Status in Scholarly Debates

Schweinsberg *et al.* [92] conduct a crowdsourced data analysis study where independent analysts use the same dataset to test two hypotheses. Expanding beyond the first crowdsourcing initiative [93], they allow analysts to not only choose their statistical approach, but also how they would operationalize key variables. The analysts

Figure 7-5: A case study illustrating error diagnostics of a multiverse on scientific debate. (a) The Poisson regression model is responsible for all errors, warnings, and abnormal point estimates. (b) Adding random effects greatly improves model quality and removes outlier estimates. (c) A warning of predictors having different scales only occurs in one set of covariates, suggesting the fix of rescaling these covariates.

produce even more dispersed analytic outcomes than the first study, however, it is unclear whether the variability in estimates is due to unconstrained operationalization of variables. To help answer this question, the researchers turn to the multiverse analysis as a meta-analysis of the analysts' scripts. The multiverse contains all reasonable decision combinations, which in this case includes and expands beyond the paths taken by the crowd analysts. Decision sensitivity of the multiverse then indicates which analytic choice plays the largest role on the dispersion of results.

The dataset used in the crowdsourced study contains words and metadata of scientific debates from an invitation-only online forum for scholars. Different teams of analysts are tasked with analyzing the data to answer two questions. The first question (H1) asks "does womens' tendency to participate actively in a conversation increase with more woman participants?" The second question (H2) is "are higher status participants more verbose than are lower status participants?" We build a multiverse for each hypothesis. Analysts make different choices in operationalizing the dependent and independent variable (*e.g.,* whether to measure status with academic citations or job rank), determining the unit of analysis (*e.g.,* commentators vs. conversations), transforming the data, choosing covariates, and specifying the statistical model. The two multiverses contain major categories of all these choices, crossing as many choice as possible and is reasonable.

Before writing the Boba specification, we need to map out the decision space and remove analytic choices that did not make sense in conjunction with one another (*e.g.,* apply logistic regression analysis to a continuous dependent variable). Examining each potential combination of all choices is infeasible due to the sheer number of paths. To work around this issue and make it easy to reason about incompatible decision combinations, we examine each pair of decisions in isolation. For each pair, we use a table where rows and columns map to options in a decision, and decide if any cells correspond to an invalid combination. We translate the tables into constraints in the Boba DSL specification.

Synthesizing the Boba DSL specification from analysis scripts authored by in-

dividual analysts is also challenging, because the scripts are not created with the goal of being combined in mind. We first reduce the scripts from analysts to the simplest version possible that still reproduces the analysis outcomes. These analysis scripts follow a general workflow of wrangling the input dataset, creating the unit of analysis, fitting a model, and extracting the model parameters. We divide the Boba specification into code blocks according to the higher-level steps in the workflow, and then distill smaller variations as placeholder variables inside code blocks. We do so incrementally, starting with a simple multiverse and then building in additional complexity, while checking for issues along the way. In the next subsection, we show how we diagnose issues that arise from the early stage of developing a multiverse analysis.

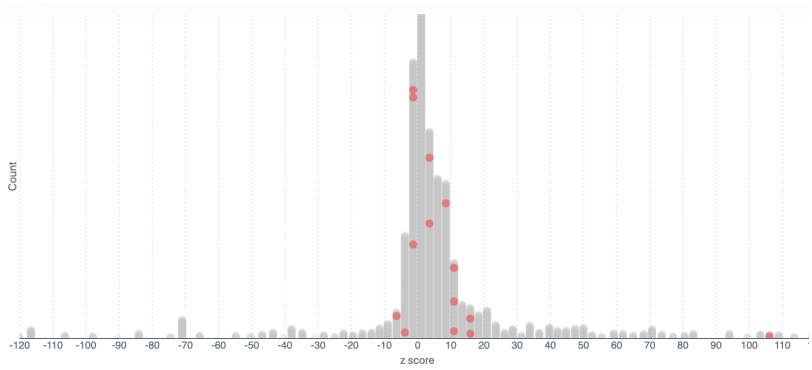
7.3.1 Diagnosing Issues Using the Boba Monitor

We now illustrate in more detail how the Boba Monitor facilitates early error diagnostics. This partial multiverse starts with a subset of the decisions and takes all compatible combinations between them, which results in 468 universes.

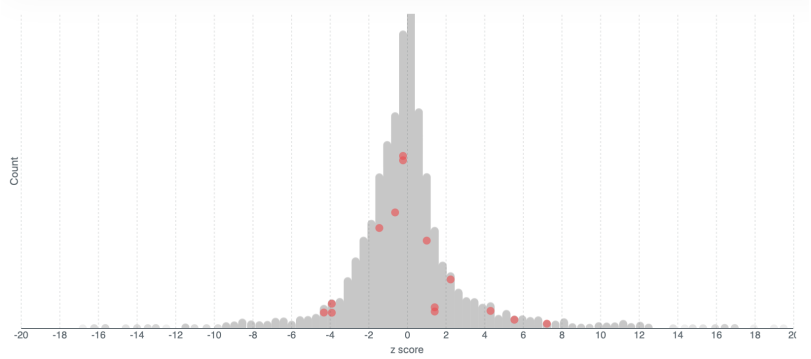
After writing the specification for this partial multiverse, we run it in the Boba Monitor. We observe plenty of warnings while the sampling estimates have not yet converged, and pause the runtime to investigate. Figure 7-5 shows the snapshot of the multiverse when 79 universes are completed. We first look for potential issues from the overview in the main effect view. The point estimate of this multiverse is the *z-score*, which measures how many standard deviations away the raw estimate is from the mean. From the range of z-scores, we can see disproportionate estimates that are more than a hundred standard deviations from the mean. In addition, almost half of the universes output an error or warning message. Switching to viewing the model quality, we observe that many models are of unsatisfactory quality. These observations suggest a need to revise the multiverse specification.

To gain a better understanding of the directions of improvement, we gauge the validity of decisions by clicking on the decision nodes and comparing the options. As shown in Figure 7-5a, the Poisson regression model is responsible for all errors, warnings, and abnormal z-score estimates. Figure 7-5b shows that adding random effects greatly improves model quality and eliminates outlying point estimates, which suggests that a single-level regression model is probably insufficient to describe the data. We might consider removing the option of single-level models from the decision space in subsequent iterations.

Next, we take a closer look at the runtime errors and warnings in the error message view. We first click on the *show more* button to view each message in full. Some errors are programming oversights such as a null reference error, requiring a simple fix. Some warnings are related to model convergence and might be improved by increasing the number of possible iterations or switching to a different optimizer. However, fixes for other issues may require knowing the accountable decisions. For example, the warning "*some predictor variables are on very different scales*" indicates problems with predictors, yet the multiverse has many combinations of independent and control variables. We click on the message to highlight related universes in the main effect view, and click the *IV* and *Covariates* decision nodes. The warning



(a) z-scores for Hypothesis 1



(b) z-scores for Hypothesis 2

Figure 7-6: Z-scores for Hypotheses 1 and 2. Outcomes from the crowd analysts are highlighted in red and represent only a subset of the multiverse of possible analyses.

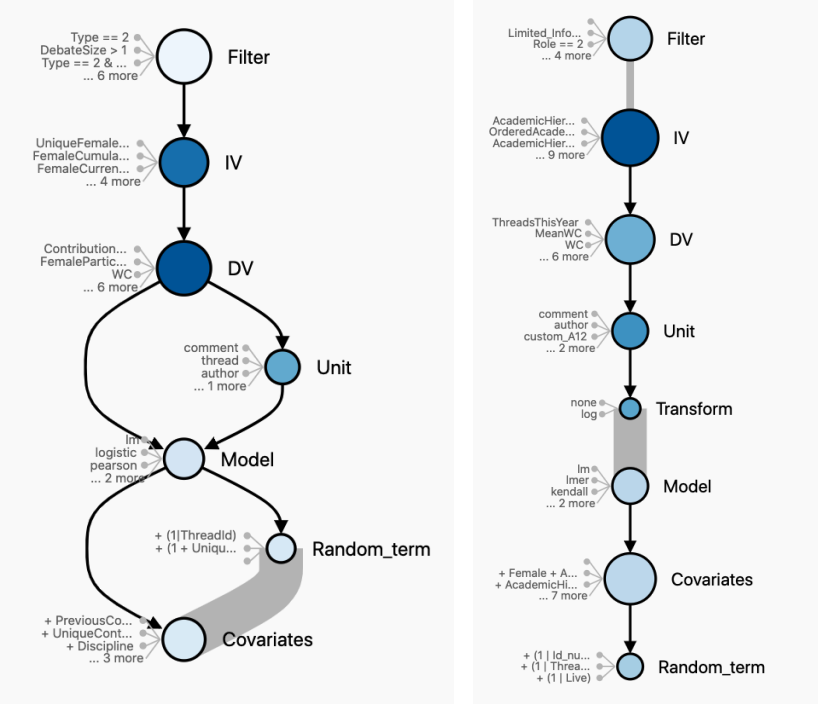
occurs in both independent variable options, but only appears in one set of covariates (Figure 7-5c), which suggests that rescaling the variables in this set of covariates should fix the issue.

7.3.2 Interpreting the Final Results

The full cross-product of all analytic choices give rise to over 5,000,000 and over 13,000,000 universes for H1 and H2, respectively. After excluding invalid combinations, only 2,984 and 15,257 universes remain. Out of these universes, some produce runtime errors related to optimization, which requires tuning individual scripts. Due to the challenges in debugging these optimization errors, we exclude these failing scripts (7 universes for H1 and 422 universes for H2) from the final multiverses.

We load the results into the Boba Visualizer, and switched to a color scheme where outcomes from the crowd analysts are highlighted in red. Figure 7-6 shows the overall distribution of point estimates, in this case z-scores, of the two multiverses. The majority of z-scores are positive for H1, suggesting an overall positive effect. In contrast, H2 seems to be quite symmetrical around zero, suggesting no effect or a tiny effect.

Figure 7-7 shows the Analytic Decision Graph for the two multiverses. For H1, DV and IV operationalizations lead to the most varied distributions in z-score, and for H2, alternative IV operationalizations have the most differing z-score distributions. These



(a) H1 (b) H2

Figure 7-7: Analytic decision graphs for Hypotheses 1 and 2.

observations suggest that unconstrained operationalization does indeed contribute to the large variation in analysis results.

7.4 Conclusion

In this chapter, we presented three case studies to demonstrate the usefulness and feasibility of the Boba system. In two multiverses replicated from prior work, we showed that we gained a rich understanding of how decisions affect results, and found issues around uncertainty and model fit that changed what we could reasonably take away from these multiverses. We also showed that by running only one fifth of the multiverse, we could arrive at the same conclusion about decision sensitivity and model quality as the full multiverse. In a multiverse consisting of analytic choices made by crowd analysts, we showed how Boba afforded early error diagnostics and enabled us to conclude whether unconstrained operationalization led to variations in analysis results.

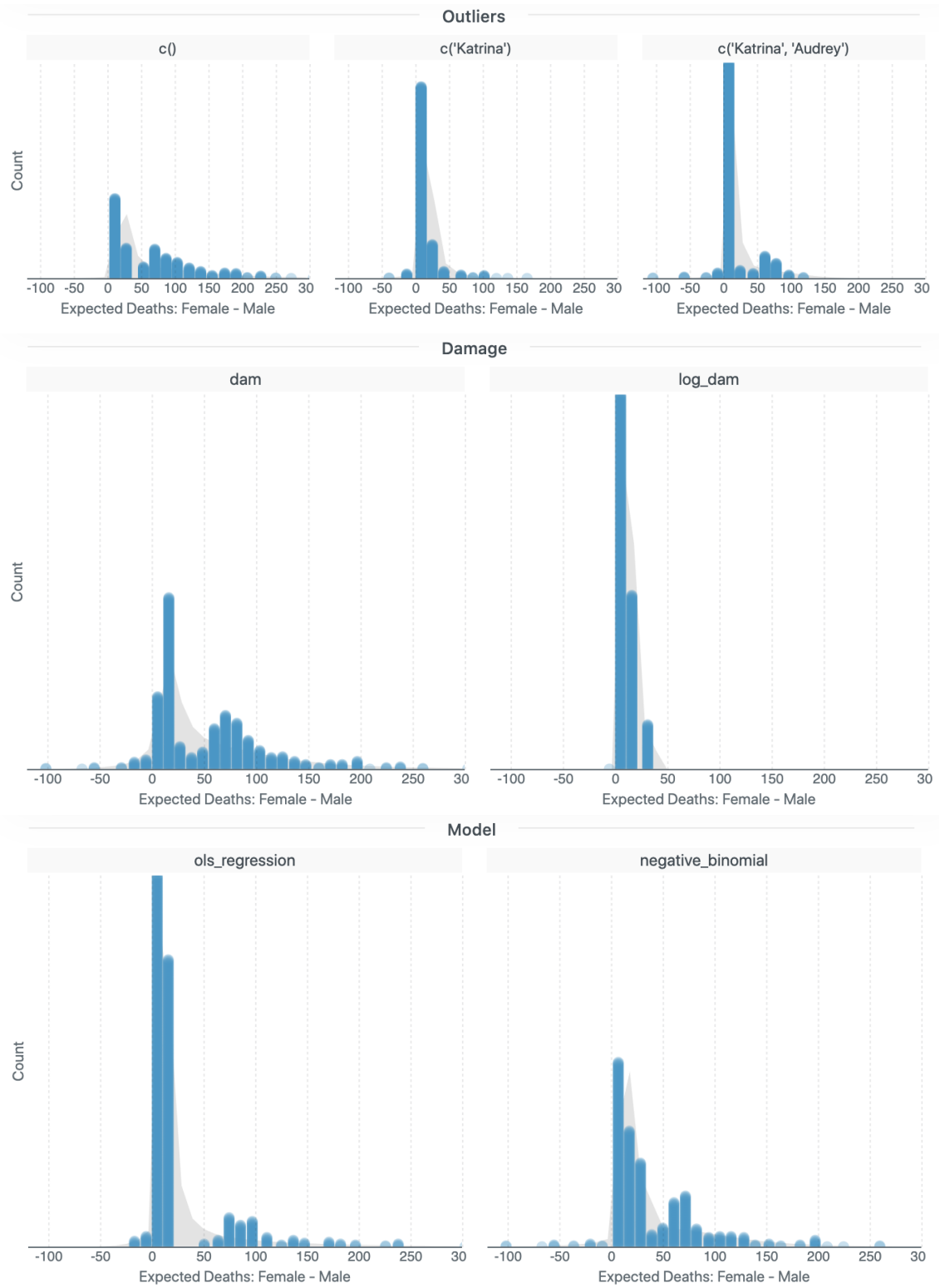


Figure 7-8: The univariate trellis plots of the relatively sensitive decisions, *outliers*, *damage* and *model* in the hurricane case study. While certain conditions tend to produce large estimates, these conditions still contain a peak around zero, suggesting that it is not a single decision that leads to large estimates.

Chapter 8

Conclusion

Producing reliable data analysis outcomes is challenging, partly due to the flexibility in making analytic decisions. Drawing inferences from data involves many decisions throughout phases of data collection, wrangling, modeling and evaluation. Since different combination of choices may lead to diverging results, and multiple reasonable options might exist at each decision point, fixating on one analytic path is less robust. In contrast, a *multiverse analysis* evaluates all compatible combinations of reasonable analytic choices together and interprets them collectively. However, multiverse analyses are challenging to author, run, and interpret, due to having myriad forking paths. Non-linear program structures are not well-supported by conventional tools, the sheer number of possible paths makes execution expensive, and analysts need to make sense of many overlapping workflows and results. To address these challenges, this dissertation contributes the design of a system that lowers the barrier for users to conduct multiverse analysis.

8.1 Review of Contributions

In Chapter 3, we conducted an interview study with 9 researchers to better understand how they made analytic decisions in their end-to-end analysis process. Corroborating prior findings on analytic decision making strategies [53], we observed that researchers often base their decisions on methodological or theoretical concerns, but subject to constraints arising from the data, expertise, or perceived interpretability. Similar to prior work documenting the reasons to value alternatives [65], we brought additional evidence on the motivations for exploring the garden of forking paths. We confirmed that researchers may experiment with choices in search of desirable results, but may also walk down alternative paths to gauge robustness or backtrack from contingency. In some cases, researchers selectively report desired results, but they may also omit findings due to similarity, correctness, or social constraints. We also contributed novel visualization designs for communicating decision processes throughout an analysis.

Based on the interview results, one major opportunity to strengthen data analysis is simultaneously analyzing multiple decision paths via a multiverse analysis. The

rest of this dissertation addressed this opportunity by introducing a system for authoring, running, and interpreting multiverse analyses. Specifically, this dissertation presented a domain-specific language for authoring multiverse analyses, approximation algorithms and monitoring visualizations for running multiverse analyses, and a visual analysis system for interpreting multiverse outcomes.

The Boba DSL (Chapter 4) formally models the decision space, providing critical structures that other system components later leverage. With the DSL, users annotate their analysis script to insert local variations, from which the compiler synthesizes executable script variants corresponding to all compatible analysis paths. The Boba DSL is agnostic to the underlying programming language of the analysis script (*e.g.,* Python or R), thereby supporting a wide range of data science use cases.

Next, we presented approximation algorithms for estimating multiverse sensitivity and monitoring visualizations for assessing progress and controlling execution on the fly (Chapter 5). In an empirical evaluation using synthetic and real multiverses, we show that round robin and sketching approaches are 5 times faster on average to estimate decision sensitivity accurately compared to no sampling, and up to 2 times faster than uniform sampling. The Boba Monitor interface facilitates users in choosing a "good enough" snapshot for early interpretation of multiverse results, as well as diagnosing runtime errors and model quality issues.

The Boba Visualizer (Chapter 6) aids users in interpreting the outcomes of all analytic paths. The Boba Visualizer first provides linked views of both analysis results and the multiverse decision space to enable a systematic exploration of how decisions do (or do not) impact outcomes. Besides decisions sensitivity, we enable users to take into account sampling uncertainty and model fit. In two multiverses replicated from prior work (Chapter 7), we demonstrate that model fit visualizations surface previously overlooked issues and change what one can reasonably take away from these multiverses. These issues imply that the multiverse workflow is iterative and a collection of *a-priori* reasonable paths may not be reasonable *post-hoc*.

8.2 Discussion and Future Directions

Through the process of designing, building, and using Boba, we gain insights into challenges that multiverse analysis poses for software designers and users. We now reflect on these challenges and additional design opportunities for supporting multiverse analysis.

Previous approaches to multiverse analysis have largely overlooked the quality of model fit, focusing instead on how to enumerate analysis decisions and display the results from the entire multiverse. We visualize model fit in two ways: we use color to encode the NRMSE from a k-fold cross validation in the outcome view, and use predictive checks to compare observed data with model predictions in the model fit view. Together these views show that a cross-product of analytic decisions can produce many universes with poor model fits, and many of the results that prior studies include in their overviews may not provide a sound base for subsequent inferences. The prevalence of fit issues, which are immediately apparent in the Boba

Visualizer, calls into question the idea that a multiverse analysis should consist of a cross-product of all *a-priori* "reasonable" decisions.

We propose adding a step to the multiverse workflow where analysts must distinguish between what seems reasonable *a-priori* vs. *post-hoc*. Boba supports this step in two ways: (1) the Boba Monitor allows users to diagnose errors early and iterate on the specification efficiently, (2) in the inference view of the Boba Visualizer, we can use model averaging to produce a weighted combination of universes based on model fit, or we can simply omit universes below a certain model fit threshold chosen by the users. The model fit threshold relies on analysts making a post-hoc subjective decision and might be susceptible to p-hacking. However, one can pre-register a model quality threshold to eliminate this flexibility. Should we enable more elaborate and interactive ways to give users control over pruning? If so, how do we prevent analysts from unintentionally biasing the results? These questions remain future work.

Indeed, a core tension in multiverse analysis is balancing the imperative of transparency with the *need for principled reduction of uncertainty*. Prior work on researcher degrees of freedom in analysis workflows [53] identifies strategies that analysts use to make decisions (see also [64, 7]), including two which are relevant here: *reducing* uncertainty in the analysis process by following systematic procedures, and *suppressing* uncertainty by arbitrarily limiting the space of possible analysis paths. In the context of Boba, design choices which direct the user's attention toward important information (e.g., highlighting models with good fit and decisions with a large influence on outcomes) and guide the user toward best practices (e.g., visual predictive checks) serve to push the user toward reducing rather than suppressing uncertainty. Allowing users to interact with results as individual dots in the outcome view while showing aggregated uncertainty in the background reduces the amount of information that the user needs to engage with in order to begin exploring universes, while also maintaining a sense of the range of possible outcomes. We believe that guiding users' attention and workflow based on statistical principles is critical.

We also reflect on how our findings and artifacts may have implications for analysis more generally. The Boba system might be useful in educating people about the sensitivity of outcomes to analytic choices. The awareness of such potential sensitivity is an important consideration even when people are not performing multiverse analyses. The design of the Boba system may be relevant for simulation studies that quantify the effect of simulation parameters on outcomes. Boba may also support an augmented form of meta analysis that covers not only the analyses performed, but also the potential analyses resulting from the combinations of analytic choices.

Building on the work presented in this dissertation and its limitations, we see a number of future directions for better supporting multiverse analyses.

8.2.1 Mapping Out the Decision Space

A multiverse analysis allows researchers to simultaneously consider results from all valid combinations of reasonable analytic decisions. Since each combination produces an outcome, and these outcomes are treated equally in the interpretation of the multiverse, the composition of the decision space is critical for the conclusion one can

draw from the multiverse. For example, an analyst may fill the decision space with redundant, non-sensitive variants and conclude that the multiverse is robust, while neglecting other meaningful, sensitive decisions. The Boba system reduces the gulf of execution [47] for analysts to perform multiverse analyses once they identify the set of reasonable decisions. However, there is little guidance among the literature and our work on how to arrive at the decision space in the first place.

We anticipate that in many cases, whether an alternative is reasonable is a subjective judgment. Past analyses, expertise, and personal agenda may play a huge role in how one characterizes reasonable decisions: a researcher may only feel comfortable using a method they are familiar with, and a Bayesian advocate may be reluctant to view frequentist methods as justifiable. Other decision rationales we observed in the interview study (Chapter 3) also involve a certain amount of ambiguity. Whether a method is easy to interpret or requires too much work is subjective. Methodological arguments can be subjective as well, for example, analysts may be unsure how restrictive they should be about statistical assumptions. We anticipate that analysts are willing to include some types of uncertain alternatives in their multiverses (*e.g.,* a method that mildly violates statistical assumptions) but not others (*e.g.,* an unfamiliar method). In other words, a multiverse analysis can iron out some idiosyncrasies but not others.

The question, then, is how much subjectivity can a multiverse analysis mitigate? Furthermore, what types of specification variations do analysts deem appropriate or inappropriate to include? Answers to these questions will allow us to scope the limitations of multiverse analyses. If most people, including those with similar backgrounds, come up with non-overlapping decision spaces, one person performing a multiverse analysis is not enough to obtain reliable outcomes. Collaboration support may be important future work. In addition, if subjectivity is unavoidable, it might be hard to create a ground truth multiverse for a given dataset. The lack of a ground truth multiverse to compare to might make it difficult to quantitatively evaluate the quality of tools supporting the generation of decision spaces.

For the types of specification variations that people tend to include in multiverse analyses, future tools may aid analysts in a more comprehensive exploration of such variations. As we observe in the interview study (Chapter 3), analysts tend to fixate on methods that they are familiar with. Due to analogical reasoning, analysts may follow the same steps that worked in the past, or dismiss choices that did not appear sensitive on previous datasets. As a result, analysts may not explore the possible decision space as comprehensively as they ideally could. One direction is to build decision recommenders that leverage the abundance of open-source analysis code [89] to automatically suggest reasonable decision points and alternatives [72]. The analyst's domain knowledge and statistical expertise are essential in ultimately determining whether the suggested alternatives are reasonable or not, suggesting the need for a mixed-initiative approach.

8.2.2 Debugging the Multiverse

As a new programming tool, Boba requires additional support to increase its usability, including code editor plugins, debugging tools, documentation, and community help. In Chapter 7 we assess the feasibility of Boba, with the understanding that its usability will need to be subsequently evaluated. Currently, Boba specifications are compiled into scripts in a specific programming language, so users can leverage existing debugging tools for the corresponding language.

However, debugging analysis scripts becomes difficult at the scale of a multiverse because a change that fixes a bug in one script might not fix bugs in others. When we attempt to run a multiverse of Bayesian regression models, for example, models in multiple universes do not converge for a variety of reasons including problems with identifiability and difficulties sampling parameter spaces with complex geometries. These issues are common in Bayesian modeling workflows and must be resolved by adjusting settings, changing priors, or reparameterizing models entirely. At the scale of multiverse analysis, debugging this kind of model fit issue is particularly difficult because existing tools for diagnostics and model checks (e.g., trace and pairs plots) are designed to assess one model at a time. While this points to a need for better debugging and model diagnostic tools in general, it also suggests that these tools must be built with a multiplexing workflow in mind if they are going to facilitate multiverse analysis.

Because the multiverse workflow is iterative, tools might audit the decision space in between iterations to highlight paths overlooked and paths that should not be included. For paths overlooked, a checklist method may be useful in prompting users about potential areas of decision points to consider. For example, users may fixate on modeling decisions, while a checklist reminds users to consider data cleaning decisions. For paths that should not be included, model quality could be taken into account to prune the decision space.

8.2.3 Target Users

The intended users for the Boba system are experts, who we assume would have a high level of statistical expertise in constructing, diagnosing, and interpreting the multiverse analyses. How to further lower the barriers for less experienced analysts is an open question. One direction is to present users with high-level abstractions that represent expert statistical knowledge, and synthesize appropriate specifications from high-level analysis goals [51]. Another is to use Boba as a design probe to both field-test the prototype and understand the challenges that less experienced users face when conducting multiverse analysis.

While the Boba DSL uses a language-agnostic design, it requires users to adopt a completely new workflow and possibly switch out of their preferred programming environment and tools when authoring the multiverse specification. The `multiverse` package [90] more tightly integrates into the existing workflows of data science workers by supporting an iterative authoring process in RMarkdown computational notebooks, but by doing so restricts itself to a single programming language. Boba might

leverage an existing programming environment by supporting linked editing [39], where users make changes in one concrete universe and have the changes automatically propagate to the Boba specification.

Our current studies focus on the *authors* of the multiverse analysis, while future work should investigate the needs of multiverse *readers*. Certain views in the Boba Visualizer might be suitable for communicating multiverse analysis results in paper manuscripts, including the outcome view showing individual point estimates and aggregated uncertainty, as well as the inference view comparing possible outcomes of the actual multiverse to the null distribution. A possible venue for future work is to assess how people interpret multiverse results, for instance comparing Boba with the animation approach in explorable multiverse analysis reports [21]. Another strategy is to leverage the ongoing research on effective uncertainty visualizations to better communicate multiverse results to a more general audience.

8.2.4 Reducing Latency

To reduce delay in viewing multiverse results, we apply approximation algorithms, but this does not preclude other methods in latency reduction and runtime optimization. A simple extension is parallel computing, with universes executed on separate processes simultaneously. Sampling is amenable to parallel computing as along as the sampling order is preserved. More involved techniques might attempt to optimize the runtime directly, as universe scripts are often highly similar with a majority of redundant source code. Future work could investigate dynamic program analysis and caching to reduce repeated computation.

The Boba Monitor largely adopts an online approach to display sampling estimates and confidence intervals progressively, and relies on analysts to decide when to view optimistic visualizations on a "good enough" snapshot. We imagine a system that automates this decision, for example recommending a snapshot when the ranking of decision sensitivity is likely to be correct with high probability [62]. The system can also do more to communicate the discrepancies between interpretations of optimistic visualizations and the ultimate precise results.

8.2.5 Model Expansion

In practical data analyses, it is recommended to start with a simple model, perform checks and diagnostics once the model is fit, and expand the model with further complexity incrementally [29]. A general method for model diagnostics is to visually gauge systematic discrepancies from posterior predictive checks. Then, one could design a *discrepancy variable* to quantify the discrepancies and perform goodness-of-fit tests [29]. By visualizing posterior predictive checks of a partial multiverse, and coloring universes based on a model quality metric (*e.g.,* discrepancy variable), users of the Boba system might adopt a similar workflow to improve a multiverse based on model diagnostics. However, it is unclear how to integrate the model expansion and multiverse analysis workflow: does the analyst incrementally build a series of models and construct the multiverse out of all reasonable ones, or start with an

a priori decision space and prune away models that fail the diagnostics, or both? Furthermore, while posterior predictive checks suggest directions to improve a model, it is difficult to do so across many different models. As a starting point, future tool might automatically cluster similar patterns in the predictive checks such that analysts might formulate refinement strategies for a set of universes instead of one at a time.

8.3 Concluding Remarks

In this dissertation, we present a characterization of researchers' decision-making practices during their analysis process, and a system for authoring, running, and interpreting multiverse analyses. We hope this dissertation will help inspire the design of both improved analysis tools and community standards. The system presented in this dissertation is available as open-source software at `https://github.com/uwdata/boba`.

Bibliography

[1] Sara Alspaugh, Nava Zokaei, Andrea Liu, Cindy Jin, and Marti A. Hearst. Futzing and moseying: Interviews with professional data analysts on exploration practices. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):22–31, 2019.

[2] David R. Anderson, William A. Link, Douglas H. Johnson, and Kenneth P. Burnham. Suggestions for presenting the results of data analyses. *The Journal of Wildlife Management*, 65(3):373–378, 2001.

[3] Monya Baker. 1,500 scientists lift the lid on reproducibility. *Nature*, 533(7604):452–454, 2016.

[4] Leilani Battle and Jeffrey Heer. Characterizing exploratory visual analysis: A literature review and evaluation of analytic provenance in tableau. *Computer Graphics Forum (Proc. EuroVis)*, 2019.

[5] C. Glenn Begley and Lee M. Ellis. Raise standards for preclinical cancer research. *Nature*, 483(7391):531–533, 2012.

[6] Richard Border, Emma C Johnson, Luke M Evans, Andrew Smolen, Noah Berley, Patrick F Sullivan, and Matthew C Keller. No support for historical candidate gene or candidate gene-by-interaction hypotheses for major depression across multiple large samples. *American Journal of Psychiatry*, 176(5):376–387, 2019.

[7] Nadia Boukhelifa, Marc-Emmanuel Perrin, Samuel Huron, and James Eagan. How Data Workers Cope with Uncertainty : A Task Characterisation Study. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2017.

[8] Nate Breznau, Eike Mark Rinke, Alexander Wuttke, Muna Adem, Jule Adriaans, Amalia Alvarez-Benjumea, Henrik Kenneth Andersen, Daniel Auer, Flavio Azevedo, and Oke Bahnsen. Observing many researchers using the same data and hypothesis reveals a hidden universe of data analysis. MetaArXiv cd5j9, Center for Open Science, 2021.

[9] Dylan Cashman, Shah Rukh Humayoun, Florian Heimerl, Kendall Park, Subhajit Das, John R. Thompson, Bahador Saket, Abigail Mosca, John Stasko,

Alex Endert, Michael Gleicher, and Remco Chang. A user-based visual analytics workflow for exploratory model analysis. *Computer Graphics Forum (Proc. EuroVis)*, 2019.

[10] Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Brian Strope, and Ray Kurzweil. Universal sentence encoder for English. In *Proc. Empirical Methods in Natural Language Processing*, pages 169–174, Brussels, Belgium, 2018. Association for Computational Linguistics.

[11] Joseph Cesario, David J Johnson, and William Terrill. Is there evidence of racial disparity in police use of deadly force? analyses of officer-involved fatal shootings in 2015–2016. *Social psychological and personality science*, 10(5):586–595, 2019.

[12] Andy Cockburn, Carl Gutwin, and Alan Dix. HARK no more: On the preregistration of CHI experiments. In *Proc. ACM Human Factors in Computing Systems*, pages 141:1–141:12, 2018.

[13] Marcus Credé and Leigh A Phillips. Revisiting the power pose effect: How robust are the results reported by carney, cuddy, and yap (2010) to data analytic decisions? *Social Psychological and Personality Science*, 8(5):493–499, 2017.

[14] John W. Creswell and Cheryl N. Poth. *Qualitative inquiry and research design: Choosing among five approaches*. SAGE publications, 2018.

[15] Robert A. Cribbie. Multiplicity control, school uniforms, and other perplexing debates. *Canadian Journal of Behavioural Science*, 49(3):159, 2017.

[16] Geoff Cumming and Robert Calin-Jageman. *Introduction to the new statistics: Estimation, open science, and beyond*. Routledge, 1 edition, 2016.

[17] Egon Dejonckheere, Elise K Kalokerinos, Brock Bastian, and Peter Kuppens. Poor emotion regulation ability mediates the link between depressive symptoms and affective bipolarity. *Cognition and Emotion*, 33(5):1076–1083, 2019.

[18] Egon Dejonckheere, Merijn Mestdagh, Marlies Houben, Yasemin Erbas, Madeline Pe, Peter Koval, Annette Brose, Brock Bastian, and Peter Kuppens. The bipolarity of affect and depressive symptoms. *Journal of personality and social psychology*, 114(2):323, 2018.

[19] Marco Del Giudice and Steven W Gangestad. A traveler's guide to the multiverse: Promises, pitfalls, and a framework for the evaluation of analytic decisions. *Advances in Methods and Practices in Psychological Science*, 4(1):2515245920954925, 2021.

[20] Pierre Dragicevic. Fair statistical communication in HCI. In *Modern Statistical Methods for HCI*, pages 291–330. Springer, 2016.

[21] Pierre Dragicevic, Yvonne Jansen, Abhraneel Sarma, Matthew Kay, and Fanny Chevalier. Increasing the transparency of research papers with explorable multiverse analyses. In *Proc. ACM Human Factors in Computing Systems*, pages 65:1–65:15, 2019.

[22] Bradley Efron. Better bootstrap confidence intervals. *Journal of the American statistical Association*, 82(397):171–185, 1987.

[23] Alexander Eiselmayer, Chat Wacharamanotham, Michel Beaudouin-Lafon, and Wendy E. Mackay. Touchstone2: An interactive environment for exploring trade-offs in HCI experiment design. In *Proc. ACM Human Factors in Computing Systems*, pages 217:1–217:11, 2019.

[24] Sebastian S. Feger, Sünje Dallmeier-Tiessen, Albrecht Schmidt, and Paweł W. Woźniak. Designing for reproducibility: A qualitative study of challenges and opportunities in high energy physics. In *Proc. ACM Human Factors in Computing Systems*, pages 455:1–455:14, 2019.

[25] Wolfgang Forstmeier and Holger Schielzeth. Cryptic multiple hypotheses testing in linear models: Overestimated effect sizes and the winner's curse. *Behavioral Ecology and Sociobiology*, 65(1):47–55, 2011.

[26] Wolfgang Forstmeier, Eric-Jan Wagenmakers, and Timothy H. Parker. Detecting and avoiding likely false-positive findings — a practical guide. *Biological Reviews*, 92(4):1941–1968, 2017.

[27] Joachim Gassen. A package to explore and document your degrees of freedom. https://github.com/joachim-gassen/rdfanalysis, 2019.

[28] Andrew Gelman. A Bayesian Formulation of Exploratory Data Analysis and Goodness-of-Fit Testing. *International Statistical Review*, 2003.

[29] Andrew Gelman and Jennifer Hill. *Data analysis using regression and multilevel/hierarchical models*. Cambridge university press, 2006.

[30] Andrew Gelman, Jennifer Hill, and Masanao Yajima. Why we (usually) don't have to worry about multiple comparisons. *Journal of Research on Educational Effectiveness*, 5(2):189–211, 2012.

[31] Andrew Gelman, Jessica Hwang, and Aki Vehtari. Understanding predictive information criteria for Bayesian models. *Statistics and Computing*, 24(6):997–1016, 2014.

[32] Andrew Gelman and Eric Loken. The garden of forking paths: Why multiple comparisons can be a problem, even when there is no "fishing expedition" or "p-hacking" and the research hypothesis was posited ahead of time. *Department of Statistics, Columbia University*, 2013.

[33] Andrew Gelman and Eric Loken. The statistical crisis in science. *American Scientist*, 102(6):460, 2014.

[34] Greg Guest, Arwen Bunce, and Laura Johnson. How many interviews are enough? an experiment with data saturation and variability. *Field methods*, 18(1):59–82, 2006.

[35] Philip J Guo and Margo I Seltzer. BURRITO: Wrapping your lab notebook in computational infrastructure. In *USENIX Workshop on the Theory and Practice of Provenance*, 2012.

[36] Philip Jia Guo. *Software tools to facilitate research programming*. PhD thesis, Stanford University, 2012.

[37] Brian D Hall, Yang Liu, Yvonne Jansen, Pierre Dragicevic, Fanny Chevalier, and Matthew Kay. A survey of tasks and visualizations in multiverse analysis reports. In *Computer Graphics Forum*. Wiley Online Library, 2022.

[38] Jenna A Harder. The multiverse of methods: Extending the multiverse analysis to address data-collection decisions. *Perspectives on Psychological Science*, 15(5):1158–1177, 2020.

[39] Björn Hartmann, Loren Yu, Abel Allison, Yeonsoo Yang, and Scott R. Klemmer. Design as exploration: Creating interface alternatives through parallel authoring and runtime tuning. In *Proc. ACM User Interface Software and Technology*, pages 91–100, 2008.

[40] Andrew Head, Fred Hohman, Titus Barik, Steven M. Drucker, and Robert DeLine. Managing messes in computational notebooks. In *Proc. ACM Human Factors in Computing Systems*, pages 270:1–270:12, 2019.

[41] Joseph M Hellerstein, Peter J Haas, and Helen J Wang. Online aggregation. In *Proceedings of the 1997 ACM SIGMOD international conference on Management of data*, pages 171–182, 1997.

[42] Peter Henderson, Riashat Islam, Philip Bachman, Joelle Pineau, Doina Precup, and David Meger. Deep reinforcement learning that matters. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018.

[43] Wassily Hoeffding. Probability inequalities for sums of bounded random variables. In *The Collected Works of Wassily Hoeffding*, pages 409–426. Springer, 1994.

[44] Wen-Chi Hou, Gultekin Ozsoyoglu, and Baldeo K Taneja. Statistical estimators for relational algebra expressions. In *Proceedings of the seventh ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems*, pages 276–287, 1988.

[45] Jessica Hullman and Andrew Gelman. To design interfaces for exploratory data analysis, we need theories of graphical inference. *arXiv preprint arXiv:2104.02015*, 2021.

[46] Nick Huntington-Klein, Andreu Arenas, Emily Beam, Marco Bertoni, Jeffrey R. Bloem, Pralhad Burli, Naibin Chen, Paul Grieco, Godwin Ekpe, Todd Pugatch, Martin Saavedra, and Yaniv Stopnitzky. The influence of hidden researcher decisions in applied microeconomics. *Economic Inquiry*, pages 1–17.

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

[48] Zubin Jelveh, Bruce Kogut, and Suresh Naidu. Political language in economics. *Columbia Business School Research Paper*, (14-57), 2018.

[49] Leslie K. John, George Loewenstein, and Drazen Prelec. Measuring the prevalence of questionable research practices with incentives for truth telling. *Psychological Science*, 23(5):524–532, 2012.

[50] Eunice Jun, Melissa Birchfield, Nicole de Moura, Jeffrey Heer, and Rene Just. Hypothesis formalization: Empirical findings, software limitations, and design implications. *arXiv preprint arXiv:2104.02712*, 2021.

[51] Eunice Jun, Maureen Daum, Jared Roesch, Sarah Chasins, Emery Berger, Rene Just, and Katharina Reinecke. Tea: A high-level language and runtime system for automating statistical analysis. In *Proc. ACM User Interface Software and Technology*, pages 591–603, 2019.

[52] Kiju Jung, Sharon Shavitt, Madhu Viswanathan, and Joseph M Hilbe. Female hurricanes are deadlier than male hurricanes. *Proceedings of the National Academy of Sciences*, 111(24):8782–8787, 2014.

[53] Alex Kale, Matthew Kay, and Jessica Hullman. Decision-making under uncertainty in research synthesis: Designing for the garden of forking paths. In *Proc. ACM Human Factors in Computing Systems*, pages 202:1–202:14, 2019.

[54] Sean Kandel, Andreas Paepcke, Joseph Hellerstein, and Jeffrey Heer. Wrangler: Interactive visual specification of data transformation scripts. In *Proc, ACM Human Factors in Computing Systems*, page 3363–3372, New York, NY, USA, 2011. ACM.

[55] Sean Kandel, Andreas Paepcke, Joseph M Hellerstein, and Jeffrey Heer. Enterprise data analysis and visualization: An interview study. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2917–2926, 2012.

[56] Matthew Kay, Tara Kola, Jessica R Hullman, and Sean A Munson. When (ish) is my bus? user-centered visualizations of uncertainty in everyday, mobile predictive systems. In *Proc. ACM Human Factors in Computing Systems*, pages 5092–5103, 2016.

[57] Matthew Kay, Gregory L. Nelson, and Eric B. Hekler. Researcher-centered design of statistics: Why bayesian statistics better fit the culture and incentives of HCI. In *Proc. ACM Human Factors in Computing Systems*, pages 4521–4532, 2016.

[58] Norbert L. Kerr. HARKing: Hypothesizing after the results are known. *Personality and Social Psychology Review*, 2(3):196–217, 1998.

[59] Mary B. Kery, Bonnie E. John, Patrick O'Flaherty, Amber Horvath, and Brad A. Myers. Towards effective foraging by data scientists to find past analysis choices. In *Proc. ACM Human Factors in Computing Systems*, pages 92:1–92:13, 2019.

[60] Mary B. Kery and Brad A. Myers. Interactions for untangling messy history in a computational notebook. In *2018 IEEE Symposium on Visual Languages and Human-Centric Computing*, pages 147–155, 2018.

[61] Mary Beth Kery, Amber Horvath, and Brad Myers. Variolite: Supporting exploratory programming by data scientists. In *Proc. ACM Human Factors in Computing Systems*, pages 1265–1276, 2017.

[62] Albert Kim, Eric Blais, Aditya Parameswaran, Piotr Indyk, Sam Madden, and Ronitt Rubinfeld. Rapid sampling for visualizations with ordering guarantees. *Proceedings of the VLDB Endowment International Conference on Very Large Data Bases*, 8(5):521, 2015.

[63] Qisheng Li, Meredith Ringel Morris, Adam Fourney, Kevin Larson, and Katharina Reinecke. The impact of web browser reader views on reading speed and user experience. In *Proc. ACM Human Factors in Computing Systems*, pages 524:1–524:12, 2019.

[64] Raanan Lipshitz and Orna Strauss. Coping with Uncertainty: A Naturalistic Decision-Making Analysis. *Organizational Behavior and Human Decision Processes*, 69(2):149–163, 1997.

[65] Jiali Liu, Nadia Boukhelifa, and James R. Eagan. Understanding the role of alternatives in data analysis practices. *IEEE Transactions on Visualization and Computer Graphics*, pages 1–1, 2019.

[66] Yang Liu, Tim Althoff, and Jeffrey Heer. Paths explored, paths omitted, paths obscured: Decision points & selective reporting in end-to-end data analysis. In *Proc. ACM Human Factors in Computing Systems*, pages 406:1–406:14, 2020.

[67] Yang Liu, Alex Kale, Tim Althoff, and Jeffrey Heer. Boba: Authoring and visualizing multiverse analyses. *IEEE Transactions on Visualization and Computer Graphics*, 27(2):1753–1763, 2021.

[68] Aran Lunzer. Towards the subjunctive interface: General support for parameter exploration by overlaying alternative application states. In *Late Breaking Hot Topics, IEEE Visualization*, volume 98, pages 45–48, 1998.

[69] Aran Lunzer. Choice and comparison where the user wants them: Subjunctive interfaces for computer-supported exploration. In *Proceedings of INTERACT*, pages 474–482, 1999.

[70] Wendy E. Mackay, Caroline Appert, Michel Beaudouin-Lafon, Olivier Chapuis, Yangzhou Du, Jean-Daniel Fekete, and Yves Guiard. Touchstone: Exploratory design of experiments. In *Proc. ACM Human Factors in Computing Systems*, pages 1425–1434, 2007.

[71] Steve McConnell. *Code complete*. Microsoft Press, 2 edition, 2004.

[72] Mike A Merrill, Ge Zhang, and Tim Althoff. Multiverse: Mining collective data science knowledge from code on the web to suggest alternative analysis approaches. In *Proceedings of ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 1212–1222, 2021.

[73] Marcus R. Munafò, Brian A. Nosek, Dorothy V. M. Bishop, Katherine S. Button, Christopher D. Chambers, Nathalie Percie Du Sert, Uri Simonsohn, Eric-Jan Wagenmakers, Jennifer J. Ware, and John P. A. Ioannidis. A manifesto for reproducible science. *Nature Human Behaviour*, 1:0021, 2017.

[74] Brad A Myers. The importance of percent-done progress indicators for computer-human interfaces. *ACM SIGCHI Bulletin*, 16(4):11–17, 1985.

[75] Leif D. Nelson, Joseph Simmons, and Uri Simonsohn. Psychology's renaissance. *Annual Review of Psychology*, 69(1):511–534, 2018.

[76] Open Science Collaboration. Estimating the reproducibility of psychological science. *Science*, 349(6251), 2015.

[77] Amy Orben and Andrew K Przybylski. The association between adolescent well-being and digital technology use. *Nature Human Behaviour*, 3(2):173, 2019.

[78] Chirag J. Patel, Belinda Burford, and John P. A. Ioannidis. Assessment of vibration of effects due to model specification can demonstrate the instability of observational associations. *Journal of Clinical Epidemiology*, 68(9):1046–1058, 2015.

[79] Kayur Patel, James Fogarty, James A. Landay, and Beverly Harrison. Investigating statistical machine learning as a tool for software development. In *Proc. ACM Human Factors in Computing Systems*, pages 667–676, 2008.

[80] Chris Pettitt. Dagre. https://github.com/dagrejs/dagre, 2015.

[81] Chanda Phelan, Jessica Hullman, Matthew Kay, and Paul Resnick. Some prior(s) experience necessary: Templates for getting started with bayesian analysis. In *Proc. ACM Human Factors in Computing Systems*, pages 479:1–479:12, 2019.

[82] Gregory J Poarch, Jan Vanhove, and Raphael Berthele. The effect of bidialectalism on executive function. *International Journal of Bilingualism*, 23(2):612–628, 2019.

[83] Florian Prinz, Thomas Schlange, and Khusru Asadullah. Believe it or not: How much can we rely on published data on potential drug targets? *Nature Reviews Drug Discovery*, 10(9):712, 2011.

[84] Xiaoying Pu and Matthew Kay. The garden of forking paths in visualization: A design space for reliable exploratory visual analytics. In *2018 IEEE Evaluation and Beyond - Methodological Approaches for Visualization (BELIV)*, pages 37–45, 2018.

[85] James R. Rae, Selin Gülgöz, Lily Durwood, Madeleine DeMeules, Riley Lowe, Gabrielle Lindquist, and Kristina R. Olson. Predicting early-childhood gender transitions. *Psychological Science*, 2019.

[86] Garvesh Raskutti and Michael W Mahoney. A statistical perspective on randomized sketching for ordinary least-squares. *The Journal of Machine Learning Research*, 17(1):7508–7538, 2016.

[87] Julia M Rohrer, Boris Egloff, and Stefan C Schmukle. Probing birth-order effects on narrow traits using specification-curve analysis. *Psychological Science*, 28(12):1821–1832, 2017.

[88] Mark Rubin. Do p values lose their meaning in exploratory analyses? it depends how you define the familywise error rate. *Review of General Psychology*, 21(3):269–275, 2017.

[89] Adam Rule, Aurélien Tabard, and James D. Hollan. Exploration and explanation in computational notebooks. In *Proc. ACM Human Factors in Computing Systems*, page 32, 2018.

[90] Abhraneel Sarma, Alex Kale, Michael J Moon, Nathan Taback, Fanny Chevalier, Jessica Hullman, and Matthew Kay. multiverse: Multiplexing alternative data analyses in R notebooks, Apr 2021.

[91] Fritz W Scholz and Michael A Stephens. K-sample anderson–darling tests. *Journal of the American Statistical Association*, 82(399):918–924, 1987.

[92] Martin Schweinsberg, Michael Feldman, Nicola Staub, Olmo R van den Akker, Robbie van Aert, Marcel ALM Van Assen, Yang Liu, Tim Althoff, Jeffrey Heer,

Alex Kale, et al. Same data, different conclusions: Radical dispersion in empirical results when independent analysts operationalize and test the same hypothesis. *Organizational Behavior and Human Decision Processes*, 2021.

[93] Raphael Silberzahn, Eric Luis Uhlmann, Dan Martin, Pasquale Anselmi, Frederik Aust, Eli C Awtrey, Štěpán Bahník, Feng Bai, Colin Bannard, Evelina Bonnier, R. Carlsson, F. Cheung, G. Christensen, R. Clay, M. A. Craig, A. Dalla Rosa, L. Dam, M. H. Evans, I. Flores Cervantes, N. Fong, M. Gamez-Djokic, A. Glenz, S. Gordon-McKeon, T. J. Heaton, K. Hederos, M. Heene, A. J. Hofelich Mohr, F. Högden, K. Hui, M. Johannesson, J. Kalodimos, E. Kaszubowski, D. M. Kennedy, R. Lei, T. A. Lindsay, S. Liverani, C. R. Madan, D. Molden, E. Molleman, R. D. Morey, L. B. Mulder, B. R. Nijstad, N. G. Pope, B. Pope, J. M. Prenoveau, F. Rink, E. Robusto, H. Roderique, A. Sandberg, E. Schlüter, F. D. Schönbrodt, M. F. Sherman, S. A. Sommer, K. Sotak, S. Spain, C. Spörlein, T. Stafford, L. Stefanutti, S. Tauber, J. Ullrich, M. Vianello, E.-J. Wagenmakers, M. Witkowiak, S. Yoon, and B. A. Nosek. Many analysts, one data set: Making transparent how variations in analytic choices affect results. *Advances in Methods and Practices in Psychological Science*, 1(3):337–356, 2018.

[94] Joseph P. Simmons, Leif D. Nelson, and Uri Simonsohn. False-positive psychology: Undisclosed flexibility in data collection and analysis allows presenting anything as significant. *Psychological Science*, 22(11):1359–1366, 2011.

[95] Uri Simonsohn, Joseph P Simmons, and Leif D Nelson. Specification curve: Descriptive and inferential statistics on all reasonable specifications. *Available at SSRN 2694998*, 2015.

[96] Transparent statistics in Human–Computer Interaction working group. Transparent statistics guidelines, Feb 2019. (Available at https://transparentstats.github.io/guidelines).

[97] Sara Steegen, Francis Tuerlinckx, Andrew Gelman, and Wolf Vanpaemel. Increasing transparency through a multiverse analysis. *Perspectives on Psychological Science*, 11(5):702–712, 2016.

[98] Kozo Sugiyama, Shojiro Tagawa, and Mitsuhiko Toda. Methods for visual understanding of hierarchical system structures. *IEEE Transactions on Systems, Man, and Cybernetics*, 11(2):109–125, 1981.

[99] Aba Szollosi, David Kellen, Danielle J Navarro, Richard Shiffrin, Iris van Rooij, Trisha Van Zandt, and Chris Donkin. Is preregistration worthwhile? *Trends in cognitive sciences*, 24(2):94–95, 2020.

[100] Michael Terry, Elizabeth D. Mynatt, Kumiyo Nakakoji, and Yasuhiro Yamamoto. Variation in element and action: Supporting simultaneous development of alternative solutions. In *Proc. ACM Human Factors in Computing Systems*, pages 711–718, 2004.

[101] Edward R Tufte, Nora Hillman Goeler, and Richard Benson. *Envisioning information*. Graphics Press, 1990.

[102] Anna E. van't Veer and Roger Giner-Sorolla. Pre-registration in social psychology – a discussion and suggested template. *Journal of Experimental Social Psychology*, 67:2–12, 2016.

[103] Aki Vehtari, Andrew Gelman, and Jonah Gabry. Practical Bayesian model evaluation using leave-one-out cross-validation and Estimating out-of-sample pointwise predictive accuracy using posterior simulations. *J Stat Comput*, 27(5):1413–1432, 2017.

[104] Chat Wacharamanotham, Krishna Subramanian, Sarah Theres Völkel, and Jan Borchers. Statsplorer: Guiding novices in statistical analysis. In *Proc. ACM Human Factors in Computing Systems*, pages 2693–2702, 2015.

[105] Eric-Jan Wagenmakers, Ruud Wetzels, Denny Borsboom, Han L. J. van der Maas, and Rogier A. Kievit. An agenda for purely confirmatory research. *Perspectives on Psychological Science*, 7(6):632–638, 2012.

[106] Xiaoyi Wang, Alexander Eiselmayer, Wendy E. Mackay, Kasper Hornbaek, and Chat Wacharamanotham. Argus: Interactive a priori power analysis. *IEEE Transactions on Visualization and Computer Graphics*, 27(2):432–442, 2021.

[107] Nathaniel Weinman, Titus Barik, Steven M Drucker, and Rob DeLine. Fork it: Supporting stateful alternatives in computational notebooks. In *Proc. ACM Human Factors in Computing Systems*, pages 0–0, 2021.

[108] Jelte M. Wicherts, Coosje L. S. Veldkamp, Hilde E. M. Augusteijn, Marjan Bakker, Robbie C. M. van Aert, and Marcel A. L. M. van Assen. Degrees of freedom in planning, running, analyzing, and reporting psychological studies: A checklist to avoid p-hacking. *Frontiers in Psychology*, 7:1832, 2016.

[109] Leland Wilkinson. Dot plots. *The American Statistician*, 53(3):276–281, 1999.

[110] Kanit Wongsuphasawat, Yang Liu, and Jeffrey Heer. Goals, process, and challenges of exploratory data analysis: An interview study. *arXiv preprint arXiv:1911.00568*, 2019.

[111] Kanit Wongsuphasawat, Dominik Moritz, Anushka Anand, Jock Mackinlay, Bill Howe, and Jeffrey Heer. Voyager: Exploratory analysis via faceted browsing of visualization recommendations. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):649–658, 2016.

[112] Yuling Yao, Aki Vehtari, Daniel Simpson, and Andrew Gelman. Using stacking to average bayesian predictive distributions (with discussion). *Bayesian Analysis*, 13(3):917–1007, 2018.

[113] Cristobal Young and Katherine Holsteen. Model uncertainty and robustness: A computational framework for multimodel analysis. *Sociological Methods & Research*, 46(1):3–40, 2017.

[114] Emanuel Zgraggen, Zheguang Zhao, Robert Zeleznik, and Tim Kraska. Investigating the effect of the multiple comparisons problem in visual analysis. In *Proc. ACM Human Factors in Computing Systems*, pages 479:1–479:12, 2018.