rTisane: Externalizing Conceptual Models for Data Analysis Prompts Reconsideration of Domain Assumptions and Facilitates Statistical Modeling

Eunice Jun
emjun@cs.ucla.edu
University of California, Los Angeles
USA

Edward Misback
misback@cs.washington.edu
University of Washington
USA

Jeffrey Heer
jheer@cs.washington.edu
University of Washington
USA

René Just
rjust@cs.washington.edu
University of Washington
USA

ABSTRACT

Statistical models should accurately reflect analysts' domain knowledge about variables and their relationships. While recent tools let analysts express these assumptions and use them to produce a resulting statistical model, it remains unclear what analysts want to express and how externalization impacts statistical model quality. This paper addresses these gaps. We first conduct an exploratory study of analysts using a domain-specific language (DSL) to express *conceptual models*. We observe a preference for detailing *how* variables relate and a desire to allow, and then later resolve, ambiguity in their conceptual models. We leverage these findings to develop rTisane, a DSL for expressing conceptual models augmented with an interactive disambiguation process. In a controlled evaluation, we find that analysts reconsidered their assumptions, self-reported externalizing their assumptions accurately, and maintained analysis intent with rTisane. Additionally, rTisane enabled some analysts to author statistical models they were unable to specify manually. For others, rTisane resulted in models that better fit the data or enabled iterative improvement.

CCS CONCEPTS

• **Human-centered computing** → **User interface toolkits**; *User interface programming*; *Empirical studies in HCI*.

KEYWORDS

statistical analysis; linear modeling; end-user programming; end-user elicitation; domain-specific language

ACM Reference Format:
Eunice Jun, Edward Misback, Jeffrey Heer, and René Just. 2024. rTisane: Externalizing Conceptual Models for Data Analysis Prompts Reconsideration of Domain Assumptions and Facilitates Statistical Modeling. In *Proceedings of the CHI Conference on Human Factors in Computing Systems (CHI '24), May 11–16, 2024, Honolulu, HI, USA*. ACM, New York, NY, USA, 16 pages. **https://doi.org/10.1145/3613904.3642267**

CHI '24, May 11–16, 2024, Honolulu, HI, USA

ACM ISBN 979-8-4007-0330-0/24/05
https://doi.org/10.1145/3613904.3642267

1 INTRODUCTION

In order to answer research questions and test hypotheses, analysts must translate their research questions and hypotheses into statistical models. To do so accurately, analysts need to reflect on their implicit understanding of the domain and consider how to represent this conceptual knowledge in a statistical model. For example, consider a health policy researcher interested in accurately estimating the influence of insurance coverage on health outcomes. To formulate a statistical model, they consider prior work on how insurance coverage, race, education, and health outcomes relate to each other and other constructs. Then, they go to formulate a statistical model including or excluding covariates to account for confounding in these relationships [7].

A researcher who skips this process may overlook relevant conceptual relationships or implicit assumptions, resulting in statistical models (and conclusions) that are faulty or meaningless as answers to their motivating research question.

Key to this explanatory modeling process is analysts' domain knowledge, captured in *process models* [20] or *conceptual models* [13]. Conceptual models include variables and their relationships that are important to a domain. Figure 1 shows an example conceptual model from our exploratory study (Section 3). A number of software tools exist for building conceptual models. For example, Tisane [15], an open-source library for authoring generalized linear models with or without mixed effects, enables analysts to explicate their conceptual models and derives valid statistical models from them. Tisane has helped HCI researchers catch and fix analysis bugs prior to publication [4]. Other tools such as Dagitty [28] and DoWhy [25] also support analysts in externalizing conceptual models as causal graphs to reason through statistical modeling choices. These software tools support (i) conceptual model specification and (ii) statistical model formulation based on expressed conceptual models.

To benefit from these tools, analysts must be able to accurately externalize their implicit conceptual models (goal (i)). This goal presents two usability challenges. First, tools should make it easy for analysts to express their conceptual models. At the very least,

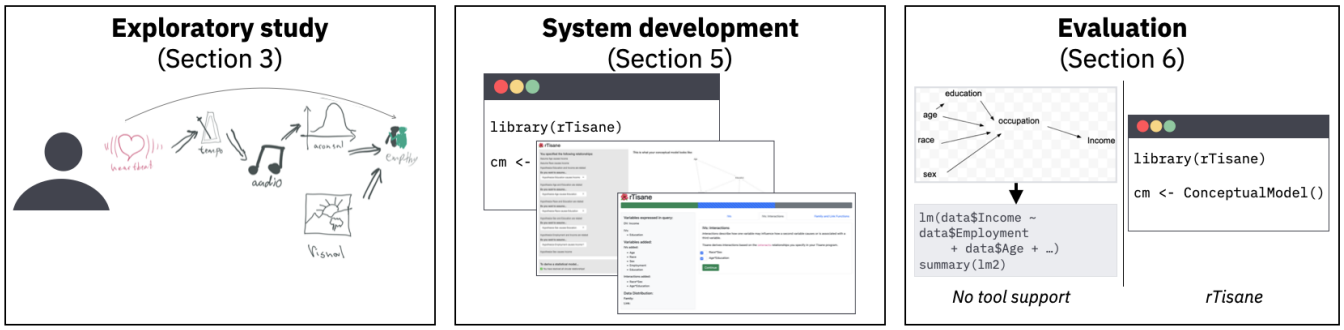

Figure 1: Visual overview of paper.
Through an exploratory study, we investigate how to better support statistical non-experts in specifying their conceptual models (Section 3). Based on findings, we develop rTisane, a system for specifying and refining conceptual models in order to derive statistical models (Section 5). We compare rTisane to a scaffolded workflow in a within-subjects controlled lab study (Section 6). We find that using rTisane to externalize conceptual models deepened consideration of implicit assumptions and helped maintain analysis intent. We also find that rTisane enabled a few analysts to author statistical models they were not able to author on their own. For others, rTisane's output statistical models fit the data better or facilitated iteration.

tools should not hinder specification. Second, analysts need guidance on which implicit assumptions are important to externalize. Addressing both challenges is particularly important for making these analysis tools usable for domain experts who have statistical experience but limited expertise (i.e., many researchers).

After analysts externalize conceptual models, tools must formulate statistical models (goal (ii)) in order to obtain high-quality statistical inferences. To ensure quality, there are two challenges to statistical model formulation: fidelity of the statistical model to the conceptual model and good statistical model fit to data. These criteria provide checks on one another. For instance, for any data set, an overfit statistical model can be found that satisfies the model fit criterion as well as possible without accurately representing the analyst's implicit conceptual model. As another example, a statistical model representing an unreasonable conceptual model may not fit real-world data well. We prioritize correspondence of conceptual models to statistical models and then, given this correspondence, consider statistical model fit.

This paper investigates how to support both accurate conceptual model specification and quality statistical model formulation. We focus on the design and implementation of a domain-specific language (DSL) for expressing conceptual models and using conceptual models to author statistical models. We focus on DSL design since end-users and graphical systems alike can benefit from DSLs. Our users are analysts who have domain expertise, experience with generalized linear modeling, and experience programming in R, but are not statistical experts. We refer to these end-users as *statistical non-experts*.

We start with an exploratory study to identify challenges statistical non-experts face when expressing their conceptual models. We find that analysts want to specify *how* variables relate causally (e.g., "more heartbeat alignment leads to more empathy") instead of stating *that* one causes another (e.g., "heartbeat alignment causes empathy"). Analysts also want to express ambiguity in their conceptual models, and, if necessary to derive statistical models, clarify any ambiguity in an interactive refinement step. Based on these findings, we develop rTisane, a system for externalizing conceptual models to author generalized linear models (GLMs). rTisane consists of (i) a DSL for expressing conceptual models and (ii) a two-phase interactive disambiguation process for refining conceptual models and then deriving statistical models. rTisane leverages an informative graphical user interface (GUI) for disambiguation. The result of this entire process is a script for fitting a statistical model that is guaranteed to reflect the expressed-then-refined conceptual model. To assess the impact of rTisane on conceptual model specification and statistical model formulation, we compare rTisane to a scaffolded workflow without tool support in a within-subjects lab study. We find that rTisane's DSL makes it easy for analysts to specify conceptual models and guides them to think more critically about their implicit assumptions. Furthermore, rTisane helps analysts focus on their analysis intents, and analysts are not surprised by rTisane's output statistical models. Of 13 analysts, three were only able to author a statistical model by using rTisane. Another six analysts were able to author statistical models that fit the data just as well, if not better, than statistical models they author without tool support. Figure 1 visually shows the three parts of this paper.

In summary, we contribute

- A study identifying how statistical non-experts want and are capable of expressing their implicit domain assumptions,
- The open-source rTisane system[1], which provides new language constructs for expressing conceptual models and a two-phase interactive disambiguation process for resolving ambiguity in conceptual models and deriving statistical models, and
- Evidence from a controlled lab study about how tool support for externalizing conceptual models to author statistical models leads to thorough conceptual model specification and quality statistical models.

[1] `https://rtisane.tisane-stats.org`

2 BACKGROUND AND RELATED WORK

We contextualize our work on rTisane in relation to empirical studies and theories of data analysis, tools for conceptual modeling, and tools for authoring statistical analyses.

2.1 Empirical studies and theories of data analysis

Data analysis is an iterative process of data discovery, wrangling, profiling, modeling, and reporting [16]. Exploratory data analysis helps analysts refine their data, analysis goals, and hypotheses [1, 3, 31]. Following exploration, analysts want to probe into relationships between variables in their data through statistical models. Statistical modeling involves considering numerous analysis decisions and choosing among a range of analysis alternatives. Liu, Althoff, and Heer [19] identified numerous decision points throughout the data lifecycle, which they call *end-to-end analysis*. They found that analysts often revisit key decisions during data collection, wrangling, modeling, and evaluation. Liu, Althoff, and Heer also found that researchers executed and selectively reported analyses that were already found in prior work and familiar to the research community. Furthermore, Liu, Boukhelifa, and Eagan [18] group analysis alternatives into cognitive (e.g., shifts in conceptual hypotheses), artifact (e.g., choice in statistical tools), and execution (e.g., computational tuning) *levels of abstraction. Cognitive* alternatives involve more conceptual shifts and changes (e.g., mental models, hypotheses). *Artifact* alternatives pertain to tooling (e.g., which software is used for analysis?), model (e.g., what is the general mathematical approach?), and data choices (e.g., which dataset is used?). *Execution* alternatives are closely related to artifact alternatives but are more fine-grained programmatic decisions (e.g., hyperparameter tuning).

Jun et al.'s conceptual framework of *hypothesis formalization* [12] encompasses all three levels of abstraction and describes more granularly how these levels cooperate with one another. Hypothesis formalization is the process by which analysts translate their research questions and hypotheses into statistical models. To craft statistical model programs, analysts incorporate and refine their domain knowledge, study design, statistical modeling choices, and computational instantiations of statistical models. Central to hypothesis formalization is the connection between implicit domain assumptions and a statistical model implementation. Implicit assumptions are encoded in informal *conceptual models*, or *process models* [20]. This paper focuses on how to provide tool support for analysts to externalize, iterate on, and formalize their implicit conceptual models. The resulting system, rTisane, facilitates one pass of hypothesis formalization in a potentially iterative modeling workflow (e.g., Bayesian Workflow [8]).

Furthermore, Grolemund and Wickham argue for statistical data analysis as a sensemaking activity [9]. Building upon the importance of *external representations* in Russell et al.'s theory of sensemaking [22], Grolemund and Wickham argue for the importance of representing and re-representing conceptual knowledge in a *schema*. Conceptual models are the external representations, or schema, this paper focuses on. We show how DSL primitives and interactive disambiguation can support conceptual modeling and how appropriate support ultimately facilitates sensemaking during and after statistical data analysis [9].

2.2 Tools for conceptual modeling

Despite the centrality of conceptual modeling to hypothesis formalization, few tools to support this step exist. Analysts can use general purpose text editing applications (e.g., Google Docs, Microsoft Word), whiteboards (e.g., manual or online), and diagramming software (e.g., Figma, Keynote) to document and share their implicit conceptual models. While usable, these software tools do not scaffold the conceptual modeling process so that it can lead to statistical models. On the other hand, tools such as Dagitty [28], CausalWizard [2], and DoWhy [25] help analysts specify causal diagrams and calculate causal estimands. Yet, these tools are designed for statistical experts who are comfortable expressing causal diagrams directly.

In this paper, we ask how we might design for both usability and rigor in expressing conceptual models. Through an iterative design process with statistical non-experts, we develop rTisane with the aim to ease conceptual modeling and reify the connection between conceptual and statistical models for both statistical non-experts and experts.

2.3 Tools for authoring statistical analyses

There is a vibrant ecosystem of tool support for statistical analysis. Libraries in programming languages such as Python, R, and Julia [5] support a wide range of analyses. Tools such as JMP [23], SAS [11], and SPSS [26] do not require programming and provide graphical user interfaces for selecting and executing statistical analysis approaches. However, existing software tools prioritize mathematical expressivity and computational control over explicit support for translating research questions and hypotheses into statistical analyses [13]. In fact, none elicit conceptual models to seed the statistical authoring process.

Researchers have proposed new DSLs and approaches that use explicit specifications of implicit conceptual assumptions to derive valid analyses. For instance, using Tea [14], analysts express hypotheses and study designs and rely on the system to automatically infer and execute a set of valid Null Hypothesis Significance Tests. Furthermore, Tisane [15] is a mixed-initiative system for authoring generalized linear models with or without mixed effects. Tisane provides a study design specification language for expressing conceptual and data relationships between variables and derives statistical models based on these. In this work, we use Tisane's open-source implementation[2] to design a study investigating challenges analysts face when expressing their implicit domain assumptions. We use Tisane because its implementation is publicly available, it is the first system to bridge conceptual and statistical modeling, and our focus is on how to best support conceptual modeling during analysis. Furthermore, while case studies of Tisane validated the feasibility and desirability of using conceptual models to author statistical models [15], the lab study in this paper delves deeper into how and why using conceptual models to author statistical analyses is beneficial. Finally, while the new DSL we design and evalute, rTisane, is scoped to output only generalized linear models,

[2] `https://github.com/emjun/tisane`

our findings generalize to primitives in Tisane and other systems (e.g., DoWhy [25]), that could result in more complex statistical models.

3 EXPLORATORY LAB STUDY

We aimed to understand the ways in which statistical non-experts want to articulate their implicit domain knowledge. We used an existing open-source library, Tisane [15], to probe into analysts' internal processes[3]. This approach helped us articulate design goals for developing rTisane (Section 4).

3.1 Method

We recruited participants through a graduate-level quantitative research methods course as a convenience sample. This allowed us to control recent exposure to statistical concepts. Five computer science PhD students volunteered to participate.

The study consisted of two parts: (i) a take-home assignment and (ii) an in-lab session. The take-home assignment asked participants to read a recently published CHI paper [30][4] and describe the paper's research questions and hypotheses, the authors' conceptual model, the study's design, and ways to analyze the data to answer the research questions. We designed the assignment to ensure that participants engaged with the paper's key ideas and internalized a common conceptual model before coming into the lab. In the lab, we could then interpret divergences in participants' expressed conceptual models as preferences and opportunities for designing new language constructs. The researcher reviewed each homework submission to prepare participant-specific questions for a semi-structured, think-aloud lab session.

At the start of the lab session, participants reviewed their homework submissions to remind themselves of the paper. The paper and participants' homework responses remained available for reference throughout the study. Then, participants completed three tasks: (i) declaring variables, (ii) specifying study designs, and (iii) expressing conceptual models. For each task, participants started with Tisane's language constructs to express their intent and discussed their confusions, how they understood each presented construct, and what they wanted to specify but could not (if applicable). The researcher repeatedly reminded participants that the constructs presented were prototype possibilities and that expressing their intentions was more important than using the constructs or getting the syntax correct. Throughout, the researcher paid particular attention to where Tisane broke down for participants and asked follow-up questions to probe deeper into why. The researcher considered such breakdowns as openings into semantic mismatches between the end-user and the DSL. The study materials are included as supplementary material.

[3]For the lab study, we re-implemented Tisane (originally in Python) in R due to R's widespread adoption in data science and use in the research methods course from which we recruited participants.

[4]We chose the specific paper because (i) we believed its topic (i.e., biosignals and empathy) would be broadly relatable, (ii) the statistical methods the authors used (i.e., generalized linear models) are aligned with our research goals, and (ii) students enrolled in the research methods course would be familiar with the paper's methods.

We iteratively coded homework submissions, audio transcripts from the lab sessions, and participants' artifacts from the lab studies. We also consulted the researcher's detailed notes from the lab sessions.

3.2 Key Observations

All participants demonstrated a working knowledge of the assigned paper's motivating research questions, study design, and general study procedure. We made the following four key observations about how statistical non-experts want to express their conceptual models. Based on these observations, we derived design goals for rTisane (Section 4).

3.2.1 Analysts want to express how variables relate to one another in detail. Analysts have an intuitive understanding of causality but bluntly stating that a variable causes another does not capture the richness or nuance of their implicit domain knowledge. Additional annotations about how a variable influences another are necessary.

When defining "causes," P2 described "*[Causes] is...like when we teach logic...it's like implication, right?....So I'm saying if we are observing an emotion and...emotion observed can lead to a change in emotional perspective.*" P0, P1, and P3 contrasted a bidirectional relationship between variables, encapsulated in the `associates_with` construct in Tisane, to their implicit understanding of "causes." For instance, P1 stated "*the most like, utilitarian definition by if A causes B, then by changing A, I can change B whereas* `associates_with` *means that...if I can turn dial A, B might not change.*" In addition to differentiating between causal and associative relationships, three participants [P0, P1, P3] provided statements of *specifically how* a variable influenced another in the conceptual models submitted as homework. For example, P0 wrote, "*Hearing a heartbeat that seems to be aligned with visual cues makes someone feel* more strongly *what another person is feeling*" (emphasis added), specifying a *positive influence* of "hearing a heartbeat" on empathy.

3.2.2 Analysts find moderation difficult to separate from bivariate relationships. Participants consistently found Tisane's `moderates` construct difficult to understand [P0, P1, P2, P3]. This construct is used to specify when one or more variables affect the strength or direction of the influence an independent variable has on a dependent variable. Participants expressed confusion about what moderation implied about the relationship between two variables. For example, P3 grappled with if `moderates` was shorthand for expressing associative relationships between each independent variable and the dependent variable, how moderation implies causal relationships, and if statistical and conceptual definitions of moderation differed from each other:

> "*[L]et's say there's two independent variables and one dependent variable. And each of the [independent] variables individually is not correlated with the outcome. But if you put them together, then the correlation appears....I mean, it's sort of a philosophical question of whether, like each of the ones individually causes [the dependent variable] in that case. But thinking from a...statistical perspective, I think that's a situation where you might be able to express...language and experience level together cause lines of code but individually they*

> *don't because no individual correlation would appear there.*"

Therefore, a clear delineation between bivariate relationships and partial statistical specifications of interaction terms is necessary.

3.2.3 Analysts distinguish between known and suspected relationships. Participants described relationships established in prior work as "assumptions" or "assertions" to check separately from the key research questions that tested "suspected" relationships. P0 described how

> "*maybe we have to differentiate as to like the* known *[relationships] are kind of the things you're* assuming *there's relationships between these things whereas the* suspected*...[are] the things kind of like your research questions are saying like, 'We* think *there's this relationship but...it's what we're testing for'*" (emphasis added).

Similarly, P4 suggested that Tisane should warn end-users when assumptions about known relationships are violated in a given data set:

> "*I would also say that it would be very handy to be able to say, kind of* assert *that language has no effect on the line of code. And be warned if it's not the case, like if your* assertion *is not...verified automatically with the DSL, but warned...that while your* assumption *is not holding there is actually an effect, which could be very handy on your study*" (emphasis added).

The inability to indicate relationships that are either known or suspected in Tisane may explain why analysts repeatedly preferred less technical verbs, such as "influences" [P0] or "leads to" [P3]. For instance, P0 explained how she preferred "influences" over "causes" because "*I guess it's like* a level of sureness *in it in which, like, 'cause' feels more confident in your answers than 'influences'*" (emphasis added). Providing a way to label conceptual relationships as assumptions or the focus of the present analysis could make conceptual modeling more approachable and lead to conceptual models that better capture analysts' implicit assumptions.

3.2.4 Analysts want to consider alternative conceptual structures. Participants grappled with what specific structures in a conceptual model meant. P1 and P3 described how a bidirectional relationship between two variables was really due to hidden, confounding variables causing both variables. P3 described how "*in the real world...when these bidirectional things happen, it means there's sort of this middleman complex system. Or some like underlying process of which [two variables are] both components...*" Another participant, P2, wondered aloud about how even what appears to be a direct relationship, may actually be a chain of indirect or mediated relationships at a lower granularity: "*It's like Google Maps. If you zoom out enough, that arrow becomes a direct arrow.*" These observations suggest that while participants can deeply reflect on what could be happening between variables conceptually, they need help exploring and figuring out which of these structures matches their implicit understanding. In other words, analysts need a way to indicate ambiguity about relationships they can then later re-consider with tool assistance.

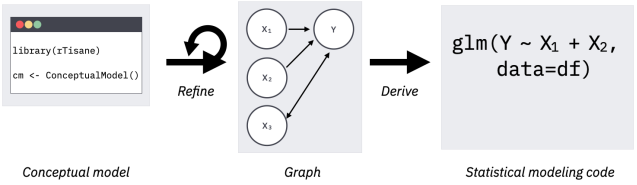

Figure 2: Overview of rTisane.
rTisane provides a DSL for specifying conceptual models (left box). Analysts validate and refine their conceptual models as the first step of a two-phase interactive disambiguation process (left arrow, see Figure 3). Interactive refinement updates the internal graph representation (middle box). rTisane traverses this graph to formulate possible statistical models (right arrow, see Figure 4). Analysts learn about rTisane's modeling decisions and can update them prior to getting a statistical modeling script as output (right box).

4 DESIGN GOALS

Based on our lab study observations, we derived four design goals to more accurately capture analysts' conceptual assumptions:

- *DG1 - Optional specificity*: Analysts should be able to provide optional details about how variables change in relation to each other (e.g., positive or negative changes in values) when describing conceptual relationships.
- *DG2 - Interactions as partial specifications*: Analysts should annotate conceptual models with interaction terms they want to include in an output statistical model.
- *DG3 - Distinction between assumed and hypothesized relationships*: Analysts should be able to distinguish between assumed and hypothesized relationships in their conceptual models.
- *DG4 - Consideration of possibilities*: Analysts should have support in expressing ambiguous relationships and then considering multiple possible conceptual structures.

We address these goals through new language constructs and a two-phase interactive disambiguation process in rTisane. We also update DSL constructs to more easily specify study design details (e.g., types of measures, syntactic sugar for specifying experimental conditions).

5 SYSTEM DESIGN AND IMPLEMENTATION

rTisane consists of (i) a DSL for analysts to express their conceptual models and (ii) interactive disambiguation steps to compile this high-level specification into a script for fitting a statistical model. A central tension in rTisane is how to design a usable DSL that allows statistical non-experts to express their assumptions in a way that is still amenable to rigorous, formal reasoning to derive statistical models. Figure 2 gives an overview of the rTisane system.

5.1 rTisane's Domain-Specific Language

rTisane provides language constructs for declaring variables, specifying a conceptual model, and querying for a statistical model.

```
1  library(rTisane)
2
3  # Declare variables
4  # Person: Observational unit
5  person <- Unit(name="person")
6  # Age: Continuous measure
7  age <- continuous(unit=person, "Age")
8  # Race, 5 categories:
9  # White, Black/African American, American Indian or
       Alaska Native, Asian or Pacific Islander, Mixed Race
10 race <- categories(unit=person, "Race", cardinality=5)
11 # Highest Education Completed, 5 ordered categories
12 edu <- categories(unit=person, "Education", order=list(
       "Grade 12","1 year of college","2 years of college",
       "4 years of college","5+ years of college"))
13 # Current Employment Status, 3 categories: Unemployed,
       Works for wage, Self-employed
14 employ <- categories(unit=person, "Employment",
       cardinality=3)
15 # Sex, 2 categories: Male, Female
16 sex <- categories(unit=person, "Sex", cardinality=2)
17 # Income: Continuous measure
18 income <- continuous(unit=person, "Income")
19
20 # Construct a conceptual model
21 cm <- ConceptualModel() %>%
22   assume(causes(age, income)) %>%
23   assume(causes(race, income)) %>%
24   hypothesize(relates(edu, income)) %>%
25   hypothesize(relates(age, edu)) %>%
26   hypothesize(relates(race, edu)) %>%
27   hypothesize(relates(sex, edu)) %>%
28   hypothesize(relates(employ, income)) %>%
29   hypothesize(causes(sex, income)) %>%
30   interacts(race, sex, dv=income) %>%
31   interacts(age, edu, dv=income)
32
33 # Query for a statistical model
34 query(conceptualModel=cm, iv=edu, dv=income)
```

Listing 1: Sample rTisane program adapted from P8 in the evaluation study. When declaring variables (lines 3-18), specifying cardinality is optional with data. Executing this program opens up the conceptual model disambiguation interface in Figure 3 .

5.1.1 Declaring variables. Analysts can express two types of variables: Units and Measures. Units represent observational or experimental units from which analysts collect data (see line 5 in Listing 1). A common unit is a participant in a study, so rTisane provides syntactic sugar for constructing a `Participant` unit directly. `Participant` is implemented as a wrapper for declaring a `Unit`.

Measures are attributes of Units collected in a dataset, so they are declared through a Unit. Measures can be one of four types: continuous, unordered categories (i.e., nominal), ordered categories (i.e., ordinal), and counts (see lines 6-18 in Listing 1). Analysts declare unordered and ordered categories through the `categories` function. Analysts can specify a variable is ordered by passing a list to the `order` parameter. Otherwise, the variable is considered unordered. Analysts can use `continuous` and `count` functions to declare continuous and count Measures. rTisane provides syntactic sugar for declaring `Conditions`, or discrete empirical interventions, as either unordered or ordered categories.

5.1.2 Specifying a conceptual model. Once analysts have constructed variables, they can specify how these variables relate conceptually. To do so, they construct a `ConceptualModel` and add variable relationships to it (lines 20-31 in Listing 1). The conceptual model is represented as a graph with variables as nodes and relationships as edges.

There are two types of relationships: `causes` and `relates`. `causes` indicates a unidirectional influence from a cause to an effect. `causes` introduces a directed edge from the cause node to the effect node. `relates` indicates that two variables are related but exactly how remains ambiguous. Analysts may be uncertain about the direction of influence. Therefore, `relates` introduces a bi-directional edge between two variables. During a disambiguation step, rTisane will walk analysts through possible graphical structures that a bi-directional edge could represent (*DG4 - Consideration of possibilities*). To derive a statistical model, rTisane requires an analyst to assume a direction of influence.

Furthermore, towards the design goal of *DG1 - Optional specificity*, rTisane allows analysts to optionally specify `when` and `then` parameters in the `causes` and `relates` functions. There are four comparisons analysts can specify in `when` and `then`: `increases` (for continuous, ordered categories, counts), `decreases` (for continuous, ordered categories, counts), `equals` (for any measure type), and `notEquals` (for any measure type). Supporting optional specificity is designed to make the rTisane program an accurate document of analysts' implicit assumptions.

To add relationships to the conceptual model, analysts must `assume` or `hypothesize` a relationship (*DG3 - Distinction between assumed and hypothesized relationships*). This distinction supports analysts in distinguishing between assumed, or strongly held, and hypothesized, or more uncertain, relationships. The distinction between `assume` and `hypothesize`, combined with the constructs for optional specificity, addresses analysts' inclination towards informal descriptions of variable relationships (e.g., "influences") observed in the exploratory study (Subsubsection 3.2.3).

Analysts can also specify interactions between two or more variables by adding `interacts` annotations to the `ConceptualModel` (lines 30-31 in Listing 1). Interactions provide additional information about existing relationships in the conceptual model (*DG2 - Interactions as partial specifications*). Interactions are not distinct relationships and so are added to the graph without `assume` or `hypothesize` statements.

5.1.3 Querying for a statistical model. Analysts `query` rTisane for a statistical model based on the input conceptual model (lines 33-34 in Listing 1). The query asks for a statistical model to accurately estimate the average causal effect (ACE) of the independent variable on the dependent variable. The querying process initiates the interactive disambiguation process, after which an `R` script specifying and fitting a generalized linear model is output.

5.2 Two-step Interactive Disambiguation

There are two phases to compiling a conceptual model to a statistical model: (i) conceptual model refinement and (ii) statistical model derivation.

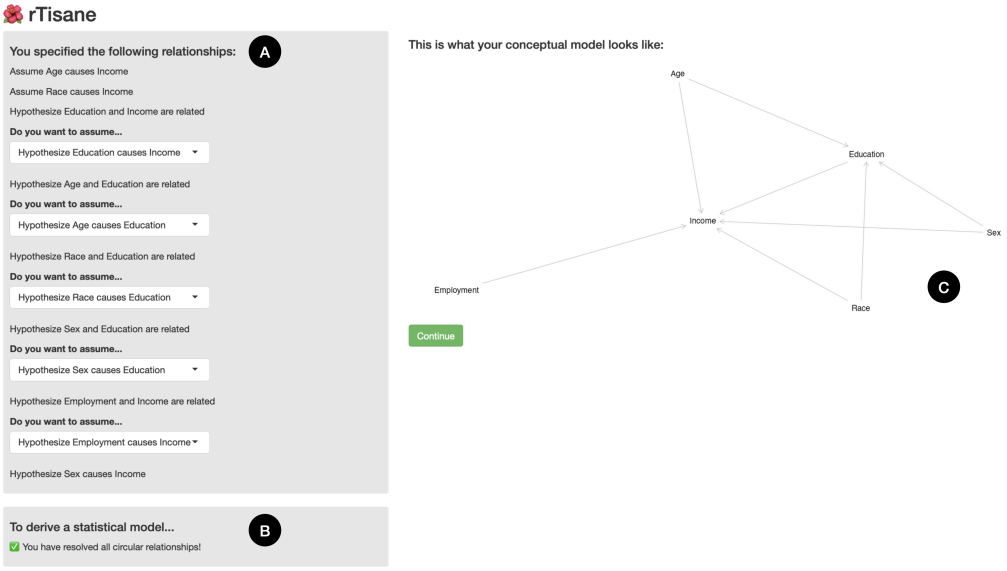

Figure 3: rTisane's conceptual model disambiguation interface.
Upon executing the example program in Listing 1, analysts see the above interface. To answer the query and derive a statistical model from a conceptual model, rTisane has analysts clarify and confirm their conceptual model. (A) The side panel shows options for resolving ambiguities in the conceptual model due to `relates` relationships (lines 24-28 in Listing 1). (B) rTisane checks and follows up with questions about breaking any cycles that hinder statistical model derivation. (C) The interface visualizes the underlying graph, updating as analysts resolve ambiguities and break cycles. Upon hitting the continue button, analysts see the statistical model disambiguation interface in Figure 4.

5.2.1 Conceptual Model Refinement. The goal of the conceptual model refinement step is to make analysts' expressed conceptual models precise enough to derive a statistical model. Conceptual model refinement involves breaking cycles in the conceptual model by (i) picking a direction for any `relates` relationships and/or (ii) removing edges. Cycles must be broken because they imply multiple different data generating processes that could lead to different statistical models. In this way, conceptual model refinement can help analysts reflect on and clarify their implicit assumptions.

To disambiguate conceptual models, rTisane uses a GUI. Figure 3 shows the conceptual model disambiguation interface for the input program in Listing 1. The GUI shows a graph representing analysts' conceptual models. If there are any `relates` relationships, rTisane suggests ways analysts could assume a direction of influence. Additionally, rTisane suggests ways to break any cycles in the conceptual model. rTisane finds cycles by iteratively searching for cycles of increasingly larger sizes up to the total number of nodes in the underlying graph representation. This algorithm takes exponential time and does not scale up to arbitrarily large graphs. rTisane suggests edges in the cycle to remove in no particular order. As analysts make changes, the graph visualization updates. The GUI also explains why these are necessary steps to derive a statistical model.

Once analysts have refined their conceptual models, rTisane updates the internal graph representation and derives a space of possible statistical models. To narrow this space of possible statistical models down to one output statistical model, rTisane asks additional follow-up disambiguating questions.

5.2.2 Statistical model derivation and disambiguation. To formulate possible statistical models, rTisane considers potential covariates to control for confounding, interactions, and family and link functions. rTisane is able to do this because it represents the conceptual model as a graph internally. rTisane treats these graphs as causal diagrams, allowing for formal reasoning about statistical model formulation.

Confounder selection. To determine confounders, rTisane uses recent recommendations from Cinelli, Forney, and Pearl [7][5]. Cinelli et al.'s recommendations are based on a meta-analysis of studies examining the impact of confounder selection from graphical structures on statistical modeling accuracy. By following Cinelli et al.'s recommendations, rTisane includes confounders that help assess the average causal effect of the query's independent variable on the dependent variable as accurately as possible.

Interaction term inclusion. Because interactions are treated as partial specifications (*DG2 - Interactions as partial specifications*), rTisane searches for interaction annotations in conceptual models. rTisane suggests any involving the query's dependent variable. Otherwise, rTisane does not consider any interactions.

Family and link function selection. rTisane determines family and link functions based on the query's dependent variable data type. For queries involving continuous dependent variables, rTisane considers Gaussian, Inverse Gaussian, and Gamma families. For counts, rTisane considers Poisson and Negative Binomial families. For ordered categories, rTisane considers Binomial, Multinomial,

[5]Tisane relied on Vanderweele's recommendations for confounder selection [29], but in rTisane we opt for more recent recommendations.

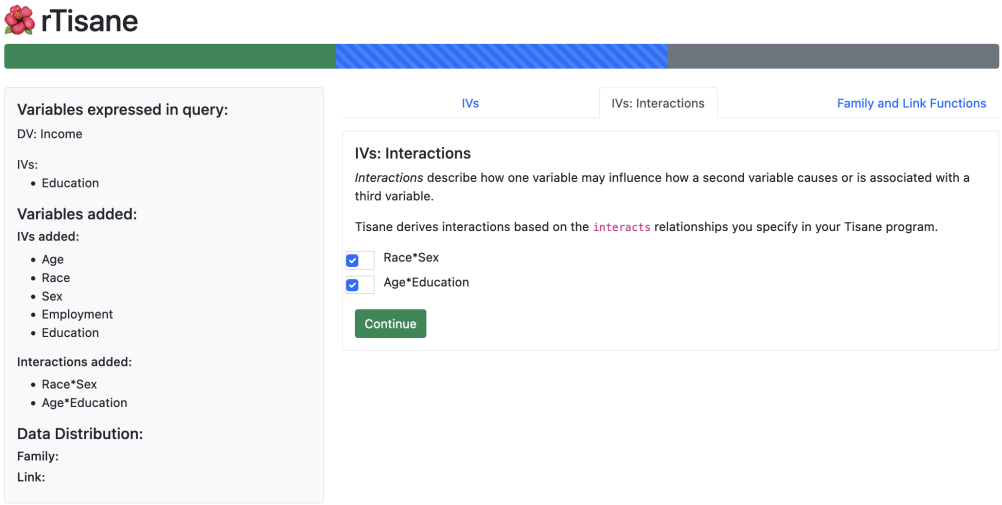

Figure 4: rTisane's statistical model disambiguation interface.
rTisane shows an interface explaining automatic statistical modeling decisions. rTisane also asks analysts questions to narrow the space of possible statistical models to a final one. Statistical model disambiguation occurs after conceptual model disambiguation (Figure 3).

Gaussian, Inverse Gaussian, and Gamma family functions. For unordered categories, rTisane considers Binomial and Multinomial family functions. rTisane outputs statistical models fit using the `lme4` package in `R`, so rTisane considers any family and link function combinations supported in `lme4`.

To inform analysts of statistical modeling choices, rTisane shows a GUI explaining confounder, interaction, and family and link function choices. In addition, for more skilled analysts, rTisane offers the opportunity to remove any confounders or interactions based on their domain knowledge or prior experience. Additionally, analysts must also pick a family and link function pair if multiple possibilities could apply. Figure 4 shows the GUI for statistical model disambiguation.

6 EVALUATION: CONTROLLED LAB STUDY

Two research questions motivated our evaluation of rTisane:

- **RQ1 - Conceptual model specification** What is the impact of rTisane on conceptual modeling? Specifically, do analysts find it easy to externalize their conceptual models with rTisane? Does rTisane help analysts determine what implicit assumptions to specify?
- **RQ2 - Statistical model quality** How does rTisane impact the statistical models analysts implement? Specifically, what are analysts' reactions to rTisane's output statistical models? How well do the statistical models analysts author on their own vs. with rTisane fit the data?

The core motivations of the paper are (i) to understand how to support conceptual model externalization and (ii) to assess the impact of tool support for externalizing conceptual models. Therefore, we designed our study to contrast rTisane—which provides a scaffolded workflow and tool support—with a scaffolded workflow. To our knowledge, no equivalent evaluation of Tisane has been performed, highlighting the significance of this study.

6.1 Study Design

We conducted a within-subjects (rTisane vs. no tool support) think-aloud lab study that consisted of four phases. All participants completed the phases in the following order:

- **Phase 1: Warm up** We presented participants with the following open-ended research question: "What aspects of an adult's background and demographics are associated with income?" We asked participants to specify a conceptual model including variables they thought influenced income. This warm-up exercise helped to externalize and keep track of participants' pre-conceived notions and assumptions prior to seeing a more restricted data schema.
- **Phase 2: Express conceptual models** We presented participants with a data schema describing a dataset from the U.S. Census Bureau. We then asked participants to specify a conceptual model using only the available variables. At the end, we asked participants about their experiences specifying their conceptual models in a brief survey and semi-structured interview.
- **Phase 3: Implement statistical models** We asked participants to implement "a statistical model that assesses the influence of variables [they] believe to be important (in the context of additional potentially influential factors) on income," relying on only their conceptual model. We then asked participants about their experiences implementing statistical models through a brief survey and semi-structured interview.
- **Phase 4: Exit interview** The study concluded with a survey and semi-structured interview where we asked participants about their experiences in the study, using rTisane, and connecting conceptual models to statistical models.

In order to assess the effect of tool support on conceptual models and the quality of statistical models, we counterbalanced the order

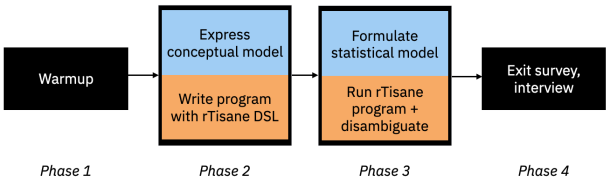

Figure 5: Evaluation phases and conditions.
We conducted a within-subjects controlled lab study where we compared rTisane to a scaffolded workflow without tool support (2 conditions). All participants completed four phases: warm-up, conceptual model specification, statistical model formulation, and exit survey with interview. For Phases 2 and 3, participants either completed the task (i) without tool support (blue) then with rTisane (orange) or (ii) with rTisane (orange) then without tool support (blue). Each participant saw the same condition order in Phases 2 and 3.

of tool support, or if participants completed each task with or without rTisane first. The order of tool use was the same for Phases 2 and 3. Specifically, within Phases 2 and 3, half the participants completed the task on their own and then with rTisane. The other half started with rTisane and then did the task on their own. Prior to using rTisane in Phases 2 and 3, participants followed a tutorial introducing the relevant language constructs for each task. Figure 5 summarizes the evaluation's study design.

In effect, the study compares rTisane to a scaffolded workflow. We chose this baseline for three reasons. First, we assume that conceptual modeling is a helpful strategy when specifying statistical models. Second, rTisane is designed to both scaffold a modeling process and provide tool support for externalizing conceptual models. Third, we wanted to isolate the effect of tool support for externalizing conceptual models rather than measure the effect of scaffolding plus tool support. Therefore, we anticipate that any impact of rTisane we observe will be more pronounced when comparing rTisane to an open-ended, unscaffolded analysis approach.

All the study materials are included as supplementary material.

Participants. We recruited 13 data analysts on Upwork. We screened for participants who reported having experience with authoring generalized linear models and using R at a three or higher on a five-point scale. Participants self-rated their data analysis experience at a median of eight out of ten (min: 5, max: 10). Table 1 summarizes the participants' backgrounds. All studies were conducted over Zoom. Participants used rTisane on a remote controlled computer, so they did not have to install it on their own. Each study lasted between two and three hours. Each participant was compensated $25 per hour. We recorded participants' screens, video, and audio throughout the study. We then transcribed the audio.

6.2 Analysis Approach

Our analysis procedure consisted of two parts: (i) a thematic analysis of lab notes, transcripts, and open-ended survey questions and (ii) an artifact analysis of conceptual models and statistical models authored with and without rTisane. For the conceptual models, we compared their form and content between tool support conditions. For the statistical models, we compared the overall statistical approach, specific statistical model formulations, rationale for analysis decisions, and two goodness of fit measures: AIC and BIC. The first two authors initially iterated on the thematic analysis and artifact analysis separately. Then, they jointly revisited both and interpreted emergent observations across the two analyses.

Table 1: Evaluation participants.

Participants came from a diversity of fields and job roles. All self-reported having familiarity with generalized linear models, experience programming in R, and significant data analysis experience.

ID	Field	Role
P1	Statistics	Data Scientist
P2	Mechanical Engineering	Graduate Student
P3	Data Science	Research Assistant
P4	Political Science	Data Science Educator
P5	Data Science	Professor
P6	Biology	Visiting Scientist
P7	Psychology	Quantitative User Researcher
P8	Bioinformatics	Researcher
P9	Data Analytics	Senior Operations Data Analyst
P10	Automotive Engineering	PhD Student
P11	Data Analysis	Research Analyst
P12	Data Analytics	Data Engineer
P13	Public Health	Data Scientist

One of the 13 participants, P1, dropped out part way through the study due to discomfort with programming in front of the researchers. P3 also stopped participation before obtaining a statistical model with rTisane. We analyzed the data we were able to collect from participants.

6.3 RQ1 Findings: rTisane's Impact on Conceptual Model Specification

Key takeaway: rTisane scaffolded and productively constrained how analysts expressed their conceptual models. As a result, analysts reflected on their implicit domain assumptions more deeply, considered new relationships, and felt they accurately externalized their implicit assumptions.

The conceptual models analysts expressed on their own were diverse in form, content, and complexity. The majority [P2, P4, P5, P8, P11, P13] invoked a graph-like structure. [P2, P4, P8 used rTisane second; P5, P11, P13 used rTisane first]. Figure 6 illustrates four example conceptual models from participants[6]. Participants also described their conceptual models verbally [P10], in natural language text [P6, P9], and as a timeline [P12]. As shown in Figure 6, P7, who used rTisane first, even jumped to expressing their conceptual model in a statistical model. P12's conceptual model was particularly creative. His timeline featured variables ordered starting on the left by how much an individual could intervene upon them (see Figure 6). P12's conceptual model reiterates our finding from the exploratory lab study that analysts want to capture nuances in a conceptual model. Furthermore, ten participants involved all five independent variables from the dataset in their

[6]An example conceptual model given in the task instructions may have biased analysts towards a graphical structure.

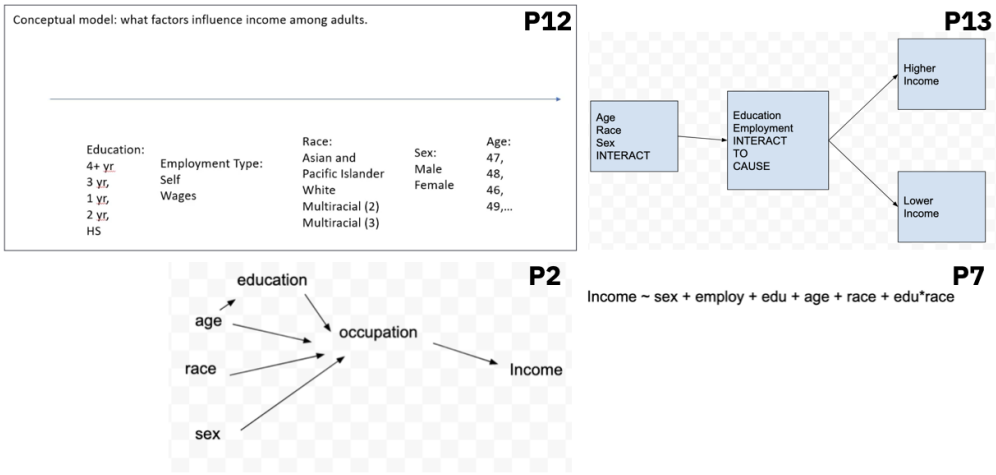

Figure 6: Evaluation: Example conceptual models without rTisane.
Participants expressed conceptual models without rTisane in a plurality of formats, including in natural language, a timeline [P12], graphs [P2, P13], and directly as a statistical model [P7]. Using rTisane, participants were able to express their conceptual models in a more structured way, which promoted deeper reflection on assumptions and consideration of additional relationships.

conceptual models [P2, P3, P4, P5, P7, P8, P9, P11, P12, P13]. Two participants [P7, P13] also included interactions between variables in their conceptual models. For instance, P13 specified a complex conceptual model where age, race, and sex interacted to cause an interaction between education and employment, which then causes income (see Figure 6).

6.3.1 Without rTisane, analysts find it difficult to fully express their assumptions. In a survey and interview about their conceptual modeling experiences, participants shared that they found it difficult to author conceptual models without tool support due to doubts about how to communicate nuances in relationships [P3, 13] and concerns about mis-specifying relationships beyond their domain knowledge [P5, P10]. P13 explained how they wanted to "*[i]dentify how I may weigh certain variables based on my general awareness and knowledge and overall weights of each variable of how one may affect income more or less in various circumstances.*" Similarly, P8 described the process of specifying their conceptual model as a general "*struggle*" because "*When doing it myself, there are so many possibilities [of expression].*" While rTisane is not designed to prevent mis-specifications due to limited domain knowledge, we found that rTisane's formalism removed the need for analysts to come up with how to express their domain knowledge. Instead, analysts could focus on expressing what they knew.

6.3.2 rTisane encourages analysts to think about and reconsider their domain assumptions. rTisane's DSL guided participants' thinking [P3, P4, P7, P8, P10, P12, P13], giving them, as P12 described, a structure to explore the "*boundaries of their domain knowledge.*" P3 explained how even after specifying conceptual models on her own, rTisane's four composable relationships (`assume`, `hypothesize x causes`, `relates`) facilitated re-consideration of each relationship and what she knew about each:

> "*Having to think about specifics like 'Do we know the direction of the relationship' or 'What happens when a category increases/decreases' actually helped me put my thoughts out more clearly. I was able to think about more possible scenarios that could conflict with my current assumption, which I was probably not doing [before without rTisane]...In conclusion, I want to say that looking at four possible ways to write a relationship made me think more about each one of them.*"

Similarly, P4 explained how the DSL's support for optional specificity "*encouraged [them] to think about the directionality of [their] hypothesized relationships and for categorical variables to think about the effect of each individual category.*"

Three participants expressed identical conceptual models with and without rTisane [P9, P11, P12]. Interestingly, for six participants, the conceptual models they authored with rTisane were subgraphs of conceptual models authored without rTisane [P2, P3, P4, P5, P7, P8]. For P2, P3, P4, and P8, who used rTisane second, rTisane appeared to help focus them on the specifics of variables and relationships of interest. P4 explained, "*coding made it [the conceptual model] more specific*". On the other hand, P5 and P7, both of whom used rTisane first, expanded upon conceptual models specified with rTisane when asked to subsequently express conceptual models on their own. For example, P7 authored a statistical model involving an interaction between variables in their rTisane conceptual model when asked to specify a conceptual model on their own. It seems that just conceptual modeling with rTisane helped P7 translate a conceptual model to a statistical model on his own. Taking these observations together, we see that rTisane's DSL can support both convergent and divergent creative thinking about analysts' domain knowledge.

6.3.3 rTisane provides structure to express conceptual models easily and accurately. Participants appreciated how rTisane structured their conceptual modeling process [P2, P4, P9, P10, P11, P12, P13]. Four participants said that rTisane generally made it easier for

them to specify their conceptual models [P4, P8, P10, P12]. P4 and P10 even believed that rTisane's "*formal structure made [conceptual modeling] more rigorous*" [P4] and "*more disciplined*" [P10]. P10 continued,

> "*My thinking was that before I didn't have much idea about how can I link my variable with the output [variable], and how this can interact. And so it may need some trial and error... using this API, there are predefined functions, they are translated in R language, cause or relates, it made my task easier. This translation was not on me anymore.*"

Participants relied on the conceptual disambiguation step to validate that what they expressed in code represented their implicit assumptions accurately [P2, P8, P12]. P2, who had drawn a conceptual model as a graph on his own prior to using rTisane, said, "*The interactive process was a good way to check that the graph came out the same way I was picturing it. It was helpful because it is easier to look at than code.*" rTisane's express-then-refine approach to specifying conceptual models helped analysts feel confident that the specified conceptual models represented their implicit assumptions accurately.

6.4 RQ2 Findings: rTisane's Impact on Statistical Model Quality

Key takeaway: rTisane focused participants on their analysis goals over low-level details that bogged them down without tool support. As a result, rTisane enabled analysts, who were not able to formulate statistical models on their own, to author statistical models. Using rTisane, analysts maintained their analysis intents and found the output statistical models to be consistent with what they would expect given their implicit domain assumptions. For other analysts, rTisane's statistical models had AIC/BIC scores that were identical to or better than those of statistical models authored without rTisane. An additional participant revised rTisane's output statistical model by log transforming the response variable to further improve the fit.

6.4.1 Without rTisane, analysts find statistical model formulation challenging. On their own, three participants were not able to author a statistical model due to unfamiliarity with statistical methods [P3], lack of time [P5], and reliance on visual analyses (e.g., heatmaps, scatterplots) [P12]. A fourth participant, P6, started to author a logistic regression model with Race and Income but stopped before binarizing Income. With rTisane, P5, P6, and P12 were able to successfully author statistical models, as Table 2 shows. P3 dropped out of the study before using rTisane.

Of the remaining eight participants who completed the study, six participants successfully authored linear regression models [P2, P4, P7, P8, P9, P10]. Two participants, both of whom had just finished authoring statistical models with rTisane, implemented GLMs [P11, P13]. P11 based their own statistical model (in the no tool support condition) on the rTisane output model script. After observing the model's "*AIC is large, the residual is large,*" P11 determined, "*I don't think this [rTisane output model] is the right fit.*" So, they log transformed the income variable and fit a new statistical model. P11's experience mirrors how we anticipate analysts will iterate on rTisane's output statistical models in the future.

Furthermore, participants reported formulating and evaluating statistical models [P2, P3, P5, P8, P12], programming [P6, P13], and preparing data [P7] as the major challenges to authoring statistical models without rTisane. For example, P3 explained how

> "*There are a number of statistical tests and it gets confusing if I don't practice it frequently. This is what happened today, I haven't worked on a hypothesis testing problem recently and while I knew what libraries to go to, I was not sure which test to implement.*"

Similarly, discussing the details of which covariates to include in a statistical model given a conceptual model, P8 explained how he was uncertain about which "*upstream relationships,*" or indirect causes, to include in a statistical model. Without rTisane, he described statistical model authoring as "*It immediately feels harder doing it directly [without rTisane] like this*" [P8].

6.4.2 Without rTisane, analysts change their analysis intent during statistical modeling. Without rTisane, participants [P2, P5, P6, P8, P10], adopted a more exploratory or data-focused approach, changing their analysis goals while authoring statistical models. This theme is best illustrated by P2, who started with a hypothesis that Occupation, or Employment, influenced Income. His conceptual model in rTisane had the variables Education, Age, Race, and Sex causing Occupation (Employment), which in turn, causes Income (see Figure 6).

He started authoring statistical models with the intent to assess this hypothesis. On his own, he first authored an ANOVA with Employment as the IV and Income as the DV. Once he saw that Employment had a statistically significant influence on Income, he changed his analysis goal to assessing if the variables causing Employment would "*be able to predict which occupation [employment]...And then...the income from the occupation [employment] just because that's how I like structured it [in the conceptual model] initially.*" However, P2 got stuck on how to author a model with Employment as the outcome variable because it was categorical, saying, "*But the way I structured it in like the diagram. I'm not sure exactly how to do that, because Occupation's [Employment's] like categorical. Um, so I'm not sure like exactly...how to model that.*" This roadblock led P2 to consider an alternative "*regression model with Income as like the output and then...all [the IVs] as terms and then just include the interactions between Occupation [Employment] and the terms that were pointing into it, and that would just be one model.*" In other words, P2 tried to author a single statistical model to assess if there was evidence for his conceptual model. However, he was unaware of three key things. First, given his conceptual model, he did not need to account for the other variables to estimate the influence of Employment on Income and assess his hypothesis. Second, adding interaction terms would not capture the dependencies in the conceptual model. Third, P2 likely needs a structural equation model to assess all the relationships in his conceptual model.

While it is well documented that statistical analysis is an iterative process [9, 13] and we saw evidence of this among participants [P5, P6, P10, P11, P12], what P2's experience exemplifies is how creative participants can be in convincing themselves that the statistical model they authored not only assessed a particular hypothesis but

Table 2: Evaluation: Comparing statistical models authored with and without rTisane.

Using rTisane, all participants were able to author statistical models. Some statistical models fit the data just as well or better than without rTisane. For each participant, the better AIC and BIC scores are in bold. AIC and BIC measure how well a statistical model fits data, with lower scores indicating better fit. Three analysts [P5, P6, P12] were only able to author statistical models with rTisane. P3 was also unable to author a statistical model on their own but stopped participation prior to obtaining a statistical model with rTisane. We omit P3 from the table below. P5, P7, P9, P11, and P13 authored statistical models with rTisane first, as indicated by [1]. P2's statistical model without tool support fits the data better in part because he prioritized data fit at the expense of maintaining analysis intent and fidelity to his conceptual model (see Subsubsection 6.4.2 for more details). For P7, P8, and P13, there are no bold scores because the statistical models with and without rTisane are identical. We did not observe a difference in statistical model quality depending on tool support order, except in the case of P11. When asked to author a statistical model without rTisane, P11 took the output model from rTisane, deemed poor model fit based on the AIC score, log transformed `Income`, and then fit the revised model as their own. To perform the log transform, P11 dropped observations where `Income=0`, explaining the marked difference in AIC/BIC scores between tool support conditions, as indicated by [a]. The supplementary material includes an additional table comparing the coefficient estimates of participants' variables of interest in models authored with and without rTisane.

ID	Tool	Statistical model	df	AIC	BIC
P2	None	`lm(data$Income ~ data$Employment + data$Age + data$Race + data$Education + data$Sex + data$Age*data$Employment + data$Race*data$Employment + data$Education*data$Employment + data$Sex*data$Employment)`	37	**60,327,741**	**60,328,211**
	rTisane	`glm(formula=Income ~ Employment, family=gaussian(link='identity'), data=data)`	4	60,781,341	60,781,392
P4	None	`lm(Income ~ Age + Education + Employment + Race + Sex, data=data)`	15	60,358,715	60,358,906
	rTisane	`glm(formula=Income ~ Education + Age + Education*Sex + Employment + Race + Sex, family=gaussian(link='identity'), data=data)`	19	**60,332,919**	**60,333,161**
P5[1]	None	-	-	-	-
	rTisane	`glm(formula=Income ~ Sex + Education + Employment, family=gaussian(link='identity'), data=data)`	9	**60,427,928**	**60,428,042**
P6	None	-	-	-	-
	rTisane	`glm(formula=Income ~ Race + Sex, family=gaussian(link='identity'), data=data)`	8	**60,794,763**	**60,794,865**
P7[1]	None	`lm(formula = Income ~ Age + Race + Education + Employment + Sex, data = data)`	15	60,358,715	60,358,906
	rTisane	`glm(formula=Income ~ Sex + Age + Employment + Race + Education, family=gaussian(link='identity'), data=data)`	15	60,358,715	60,358,906
P8	None	`lm(Income ~ Sex*Race + Employment + Education + Race*Sex + Age, data = data)`	20	60,354,038	60,354,292
	rTisane	`glm(formula=Income ~ Age + Race*Sex + Employment + Age*Education, family=gaussian(link='identity'), data=data)`	24	**60,351,454**	**60,351,759**
P9[1]	None	`smf.OLS("Income ~ Age + C(Race) + C(Education) + C(Employment) + C(Sex)", data=df)`	15	60,358,715	60,358,906
	rTisane	`glm(formula=Income ~ Employment + Race + Sex + Education + Age, family=gaussian(link='identity'), data=data)`	15	60,358,715	60,358,906
P10	None	`sm.OLS.from_formula("Income ~ Age", data=df)`	3	60,876,872	60,876,910
	rTisane	`glm(formula=Income ~ Employment + Sex + Education + Age + Sex*Education, family=gaussian(link='identity'), data=data)`	14	**60,339,137**	**60,339,315**
P11[1]	None	`glm(log_income ~ Employment + Race + Age + Education + Sex, family = "gaussian", data=data)`	15	**11,741,899[a]**	**11,742,089[a]**
	rTisane	`glm(formula=Income ~ Employment + Race + Sex + Education + Age, family=gaussian(link='identity'), data=data)`	15	60,358,715	60,358,906
P12	None	-	-	-	-
	rTisane	`glm(formula=Income ~ Age, family=gaussian(link='identity'), data=data)`	3	**60,876,872**	**60,876,910**
P13[1]	None	`glm(Income ~ Age*factor(Race)*factor(Sex) + factor(Education)*factor(Employment), family="gaussian", data)`	39	60,331,749	60,332,244
	rTisane	`glm(formula=Income ~ Employment + Age*Race*Sex + Education*Employment + Education, family=gaussian(link='identity'), data=data)`	39	60,331,749	60,332,244

could also arbitrate if their entire conceptual models were supported by data. Furthermore, this suggests an opportunity for rTisane to support a more iterative analysis process and help analysts author multiple models to assess an entire conceptual model, not just the influence of a single independent variable on a dependent variable, and idea we expand upon in Section 8.

6.4.3 Analysts validate that rTisane's output statistical models address their motivations for analysis and represent their domain assumptions. In contrast, participants reported that rTisane guided them to think about their domains more [P2, P12], lightened their burden in authoring statistical models [P10], and even promoted research transparency [P5] and reproducibility [P4]. Furthermore, rTisane reinforced prior knowledge about statistical methods [P6, P11] and helped participants learn more about GLMs [P4, P6, P7, P13]. P6, who had tried and failed to author a logistic regression

model on her own, explained how she could apply what she learned from using rTisane to future analyses: "*I like that a multivariate linear regression was used because this will inform any future data analysis.*" Additionally, participants reported feeling unsurprised at rTisane's output statistical models [P4, P6, P10, P11, P12, P13]. P10 remarked how rTisane's output statistical model "*was like my thinking*" and represented their conceptual model: "*What I notice is that rTisane formulated my thinking, but didn't add any other notions.*"

6.4.4 Statistical models authored with rTisane fit the data just as well or better than statistical models without rTisane. Of the eight participants who successfully authored linear regression models or GLMs on their own, three implemented identical models with or without rTisane [P7, P9, P13]. Notably, all three had authored the statistical model with rTisane first, suggesting that rTisane anchored their own modeling processes. Table 2 shows statistical models authored with and without rTisane and their AIC and BIC goodness of fit measures. For another three participants who used rTisane second [P4, P8, P10], their statistical models with rTisane had lower AIC and BIC scores than the statistical models without rTisane. Notably, P4, P8, and P10 did not rely on the statistical models they authored manually to author statistical models with rTisane, suggesting that rTisane is what helped them author statistical models that fit the data better. Thus, for six out of eight participants, rTisane's statistical models fit the data better or equally well. For P11, the statistical model they authored without rTisane dropped some observations, so the models are not directly comparable. For P2, the rTisane statistical model fit worse than his own statistical model in part due to an observed change in his motivation for analysis, discussed above (Subsubsection 6.4.2).

6.5 Opportunities to Improve rTisane

While participants found rTisane helpful, they suggested two areas of tool improvement: (i) family and link function selection and (ii) statistical model interpretation. Additionally, participants expressed wanting to use rTisane for scientific communication, not just statistical authoring. These ideas require future research and have the potential to help analysts engage with and understand their analyses more deeply.

Several participants had difficulty picking family and link functions [P2, P4, P5, P9, P10, P11]. P4 explained, "*I didn't understand the benefit or tradeoffs between different specifications. It wasn't obvious to me how to create a linear OLS regression, or why I would want to use a specification besides linear OLS.*" This problem arises from the stark contrast between rTisane's relatively high-level conceptual modeling abstractions, and rTisane's statistical disambiguation step that requires analysts to select specific family and link functions, a relatively low-level statistical modeling detail. Therefore, an important next step is to incorporate approaches to suggest a specific pair of family and link functions and interfaces that explain the "*tradeoffs*" in choices.

Because rTisane uses lme4 under the hood, the result of executing the output statistical modeling script is the output from lme4. However, analysts expected the outputs to at least relate back to their conceptual models, given that rTisane's DSL is focused on conceptual modeling. For example, P8 found the output from lme4 overwhelming, saying, "*Looking at the* `summary()` *in R was too much to look at.*" He suggested a simple way to tie the results back to his input conceptual model: "*Would be nice if you could have the same visual representation with p-values/coefficients!*" Future work should explore ways to make statistical modeling output more understandable for statistical non-experts.

Lastly, when asked how they might imagine using rTisane, participants described how experienced and novice analysts alike would benefit from using rTisane [P2, P4, P9, P10, P12]. Participants also suggested that conceptual models written using rTisane could help collaborations with less technical stakeholders [P8, P9]. For instance, P8 detailed how a conceptual model written using rTisane could be a communication tool, saying how the "*visual representation would play a role in a dialogue with the PI.*" P8 went on to imagine how he would like to use rTisane's conceptual model to generate process diagrams in scientific papers. We expand upon this possibility in Section 8.

7 DISCUSSION

rTisane structures a conceptual model specification process that prompts reconsideration of domain assumptions by providing both a usable DSL and an interactive disambiguation process for refining a conceptual model after initial specification. rTisane also guarantees fidelity between conceptual and statistical models by translating expressed conceptual models into causal diagrams to inform statistical model formulation. By distinguishing between conceptual models and statistical models, rTisane is a first step towards embodying the "blueprint for a 'causal inference engine'" described by Judea Pearl and Dana Mackenzie [21]. Specifically, rTisane's DSL captures the "inputs," and rTisane's interactive disambiguation process acts as the "inference engine" in Pearl's blueprint [21].

In the evaluative study, we find that rTisane enables analysts who otherwise cannot create statistical models to successfully author them. Analysts validate that rTisane's statistical models are consistent with their conceptual assumptions. Additionally, statistical models authored with rTisane at times fit real-world data better than statistical models authored without rTisane. In other cases, rTisane's output statistical models serve as the basis for further model tuning. In other words, this work has demonstrated how externalizing conceptual models (i) increases consideration of implicit domain assumptions and (ii) can facilitate authoring of quality statistical models. These findings demonstrate the benefits of formalism, designing for both usability and rigor in DSLs, and the potential for shared representations [10] to become boundary objects.

Formalism can facilitate reflection. While interfaces leveraging natural language, especially in the era of large language models and their applications (e.g., ChatGPT [6]), are enticing, we find that analysts in our evaluation preferred the structure provided by rTisane's DSL over open-ended specification without rTisane. rTisane focuses analysts on what to express about their implicit assumptions while also providing them with easy-to-learn syntax for doing so. Analysts use rTisane's DSL as a starting point to distinguish assumptions based on prior literature from their own hypotheses, reconsider their implicit assumptions, and consider new relationships. As a result, analysts report that the conceptual

models expressed with rTisane accurately represent their internalized knowledge. In other words, rTisane's formalism promotes what Donald Schön calls *reflection-in-action* [24] during data analysis. Therefore, HCI researchers should consider how formalisms and the process of specification using a formalism can facilitate a sensemaking process [22] that helps users attain their ultimate goal.

Usability and rigor as DSL design objectives. DSLs need to be both usable for people to write programs in them and rigorously designed for automation to accomplish specific tasks. Shared representations [10] may be key to attaining both. To ensure usability of rTisane's DSL, we iteratively design language constructs and disambiguating interactions. We use an existing DSL to probe into what and how analysts want to express their implicit knowledge, design rTisane, and evaluate it in a controlled lab study. We design for rigor in the compilation process from an input conceptual model specification to an output statistical model representing the conceptual model and analysis intent. In the evaluation, we find that analysts could easily translate their domain knowledge into conceptual models using the rTisane DSL (usability), which then generated statistical models that addressed their motivating research questions (rigor) and fit the data, sometimes better than hand-coded statistical models. In this way, rTisane exemplifies the synergy of usability and rigor by leveraging the conceptual model as a shared representation [10].

Conceptual models as potential boundary objects. In the evaluation, participants discuss the potential for using conceptual models to communicate assumptions and analyses with collaborators and the broader scientific community. Specifically, participants mention the value of conceptual models as a record of the analyst's thoughts for future analysts, as a way of summarizing the analysis for less technical collaborators, and as a way of generating process diagrams for scientific papers. In all of these applications, conceptual models serve as an *intermediate representation* that can be "compiled" to a number of "backends." Furthermore, these applications suggest that conceptual models are likely useful as boundary objects [27] for collaboration and communication, a future research direction worth pursuing. Indeed, scientists in the same discipline could use rTisane to author, share, debate, and build upon each other's conceptual models, independent of data collection or statistical modeling details. In this light, rTisane could serve as one instrumental tool in a larger effort to elevate scientific discourse and increase scientific and statistical literacy, transparency, and reproducibility.

8 LIMITATIONS AND FUTURE WORK

There are three promising avenues for future research building on this work.

DSL design and use in interactive systems for statistical analysis. First and foremost, our goal has been to (i) investigate how to support conceptual model externalization and (ii) assess the impact of conceptual modeling. To answer these questions, we focus on the design of a DSL because both analysts and analysis tool developers can leverage DSLs. Importantly, the current version of rTisane's DSL is one implementation of the design goals (Section 4) we identified from the exploratory study (Section 3). Alternative DSL designs (e.g., standalone vs. embedded in a host language) and syntaxes are likely to make different usability tradeoffs. These tradeoffs are worth exploring in order to support analysts with diverse statistical analysis and programming needs.

rTisane supports specification of initial conceptual models using text and refinement through an interactive GUI. In the lab evaluation, we found that the textual specification step structures analysts' thought processes and that participants want to share these programs with collaborators. We also found that the GUI during disambiguation helps analysts validate their specifications easily. It seems that by using both textual and graphical modalities, rTisane achieves simplicity in both specification and validation. In contrast, Dagitty's web interface [28] supports immediate drawing of causal diagrams through a GUI. While we suspect that rTisane's design is more approachable for statistical non-experts and that separating textual specification from graphical refinement is helpful in structuring analysts' thinking, ablation studies are necessary to isolate the impact of modalities and steps on analysts.

A related future direction is to investigate how to incorporate rTisane's primitives directly into tools like Dagitty [28]. For instance, could drawing-based tools provide analysts with drop-down menu options for labeling conceptual relationships as known or suspected? How would these designs affect the conceptual models analysts express? Could these designs make existing interactive tools for externalizing conceptual models more usable for statistical non-experts? Future work should address these questions.

Additional evaluations of rTisane. Second, the lab evaluation has three limitations: (i) the number and backgrounds of participants, (ii) the within-subjects design, and (iii) the measures used to evaluate statistical models.

Our sample size of 13 is limited. While we reached convergence and saturation of themes while analyzing transcripts and researcher notes, future evaluations with more participants are necessary to validate and expand upon our findings. Moreover, we recruited participants through the online freelance platform Upwork. As a result, our participants came from a variety of disciplines and were data analysis practitioners and educators (Table 1). We filtered for participants who self-reported familiarity with generalized linear modeling and R. However, some struggled with R syntax, suggesting that their self-reported skills were inflated. Therefore, it seems that rTisane is able to help even those with less expertise than we expected. A similar limitation exists for our exploratory study involving CS PhD students in a research methods course. Future work should focus on assessing the impact of rTisane on novice analysts from specific disciplines in order to reveal additional language constructs or interactions to help a wider range of users.

Additionally, we designed a within-subjects lab study because our priority was to capture and compare the qualitative differences between authoring analyses with and without rTisane. As a result, using rTisane first likely influenced, even changed, the analysis process without rTisane. Notably, P11 used the statistical model output from rTisane and tuned it when asked to author a statistical model on their own. This observation suggested how analysts are likely to incorporate rTisane into their worfklows and was only made possible by our within-subjects design. Additionally, we compared

AIC and BIC measures for statistical models authored with and without rTisane since they give a general sense of statistical model quality while controlling, to an extent, for overfitting to data. We also inspected the relative differences of effect estimates for analysts' variables of interest in the supplementary material. However, we cannot quantify the influence of rTisane on effect estimates since we used a real-world dataset without a ground truth causal diagram and allowed analysts to pick their variables of interest. In other words, in designing our study and measures, we prioritized ecological validity to make richer qualitative observations. Future evaluations of rTisane should consider alternative designs that further isolate and quantify the benefits of rTisane.

Support for statistical iteration. Third, we believe future tool support for statistical model iteration is crucial. Currently, rTisane allows analysts to iterate on their conceptual models by adding or removing variables and relationships. However, it lacks support for a larger iteration loop with the resulting statistical model. For instance, P11 described the rTisane output statistical model as "*an initial or baseline model but follow-up evaluation of the model is needed.*" They wanted to "*go back and tweak things a bit*" about their statistical model. Tools like rTisane should ensure that analysts maintain their analysis intents throughout iteration—or at least document conceptual shifts—while discouraging or even preventing analysts from questionable "data dredging" or HARKing [17] practices. A first step may be to support recommended workflows for statistical model development and refinement, such as Gelman et al.'s Bayesian Workflow [8].

Tool support for iterative modeling could foster new methods to handle ambiguity in conceptual models. For instance, analysts can already express ambiguous conceptual relationships in rTisane's DSL. What might leveraging this ambiguity look like? For example, what if rTisane could generate multiple statistical models corresponding to all conceptual models implied by the ambiguous specification (i.e., conduct a multiverse analysis)? Or, what if tools could guide analysts towards incrementally considering specific statistical models, their results, and their conceptual implications? For instance, future tools could enable analysts to fit a statistical model, revise their conceptual model based on results, and formulate follow-up queries until analysts arrive at a conceptual model supported by their data. To support these use cases, future research on how to judiciously guide exploration of conceptual and statistical alternatives is necessary.

9 CONCLUSION

rTisane provides a DSL with language constructs for expressing conceptual models and integrates a two-phase interactive disambiguation process for compiling conceptual knowledge into statistical analysis code. In a controlled lab study of rTisane, we find that the DSL is expressive enough to capture analysts' conceptual models, eases the burden of making their implicit assumptions explicit, and pushes analysts to think about and reconsider their domain assumptions. Using rTisane, analysts, including those who otherwise struggle with statistical model formulation, are able to author statistical models. The resulting statistical models fit the data just as well as, and sometimes better than, statistical models authored without rTisane and can even facilitate analyst-driven model tuning. Together, these results demonstrate how supporting externalization of conceptual models during data analysis enables analysts to author quality statistical models that they might struggle to author otherwise.

ACKNOWLEDGMENTS

The authors thank the members of the Interactive Data Lab and Eunice Jun's PhD committee (Emery Berger, Tyler McCormick, and Leilani Battle) for their support and advice throughout the project. Eunice thanks Alex Kale for early conversations about conceptual drift in data analysis. The authors also thank the anonymous reviewers for their thoughtful feedback. This work was supported by the NSF (Award Numbers III-1901386 Analysis Engineering and CCF-1942055) as well as by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, ComPort: Rigorous Testing Methods to Safeguard Software Porting (Award Number DE-SC0022081).

REFERENCES

[1] Sara Alspaugh, Nava Zokaei, Andrea Liu, Cindy Jin, and Marti A Hearst. 2018. Futzing and moseying: Interviews with professional data analysts on exploration practices. *IEEE transactions on visualization and computer graphics* 25, 1 (2018), 22–31.

[2] AI/ML Services (Australia). 2023. CausalWizard. **"https://causalwizard.app/"**

[3] Leilani Battle and Jeffrey Heer. 2019. Characterizing Exploratory Visual Analysis: A Literature Review and Evaluation of Analytic Provenance in Tableau. *Computer Graphics Forum (Proc. EuroVis)* (2019). **"http://idl.cs.washington.edu/papers/exploratory-visual-analysis"**

[4] Amanda Baughan, Mingrui Ray Zhang, Raveena Rao, Kai Lukoff, Anastasia Schaadhardt, Lisa D Butler, and Alexis Hiniker. 2022. "I Don't Even Remember What I Read": How Design Influences Dissociation on Social Media. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*. 1–13.

[5] Jeff Bezanson, Alan Edelman, Stefan Karpinski, and Viral B Shah. 2017. Julia: A fresh approach to numerical computing. *SIAM review* 59, 1 (2017), 65–98.

[6] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems* 33 (2020), 1877–1901.

[7] Carlos Cinelli, Andrew Forney, and Judea Pearl. 2020. A crash course in good and bad controls. *Sociological Methods & Research* (2020), 00491241221099552.

[8] Andrew Gelman, Aki Vehtari, Daniel Simpson, Charles C Margossian, Bob Carpenter, Yuling Yao, Lauren Kennedy, Jonah Gabry, Paul-Christian Bürkner, and Martin Modrák. 2020. Bayesian workflow. *arXiv preprint arXiv:2011.01808* (2020).

[9] Garrett Grolemund and Hadley Wickham. 2014. A cognitive interpretation of data analysis. *International Statistical Review* 82, 2 (2014), 184–204.

[10] Jeffrey Heer. 2019. Agency plus automation: Designing artificial intelligence into interactive systems. *Proceedings of the National Academy of Sciences* 116, 6 (2019), 1844–1850.

[11] SAS Institute Inc. 2021. SAS. **https://www.sas.com/**

[12] Eunice Jun, Melissa Birchfield, Nicole De Moura, Jeffrey Heer, and Rene Just. 2021. Hypothesis Formalization: Empirical Findings, Software Limitations, and Design Implications. In *ACM Transactions on Computer-Human Interaction (TOCHI)*, Vol. 29. Issue 1. **"https://arxiv.org/abs/2104.02712"**

[13] Eunice Jun, Melissa Birchfield, Nicole De Moura, Jeffrey Heer, and Rene Just. 2022. Hypothesis formalization: Empirical findings, software limitations, and design implications. *ACM Transactions on Computer-Human Interaction (TOCHI)* 29, 1 (2022), 1–28.

[14] Eunice Jun, Maureen Daum, Jared Roesch, Sarah E Chasins, Emery D Berger, Rene Just, and Katharina Reinecke. 2019. Tea: A High-level Language and Runtime System for Automating Statistical Analysis. In *Proceedings of the 32nd Annual Symposium on User Interface Software and Technology*. ACM.

[15] Eunice Jun, Audrey Seo, Jeffrey Heer, and René Just. 2022. Tisane: Authoring Statistical Models via Formal Reasoning from Conceptual and Data Relationships. *ACM CHI* (2022).

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

[17] Norbert L Kerr. 1998. HARKing: Hypothesizing after the results are known. *Personality and social psychology review* 2, 3 (1998), 196–217.

[18] Jiali Liu, Nadia Boukhelifa, and James R Eagan. 2019. Understanding the Role of Alternatives in Data Analysis Practices. *IEEE transactions on visualization and computer graphics* 26, 1 (2019), 66–76.
[19] Yang Liu, Tim Althoff, and Jeffrey Heer. 2019. Paths Explored, Paths Omitted, Paths Obscured: Decision Points & Selective Reporting in End-to-End Data Analysis. *arXiv preprint arXiv:1910.13602* (2019).
[20] Richard McElreath. 2020. *Statistical rethinking: A Bayesian course with examples in R and Stan.* CRC press.
[21] Judea Pearl and Dana Mackenzie. 2018. *The Book of Why: The New Science of Cause and Effect.* Basic books.
[22] Daniel M Russell, Mark J Stefik, Peter Pirolli, and Stuart K Card. 1993. The cost structure of sensemaking. In *Proceedings of the INTERACT'93 and CHI'93 conference on Human factors in computing systems*. ACM, 269–276.
[23] SAS. 2020. JMP. **"https://www.jmp.com/en_us/home.html"**
[24] Donald A Schön. 1987. *Educating the reflective practitioner: Toward a new design for teaching and learning in the professions.* Jossey-Bass.
[25] Amit Sharma and Emre Kiciman. 2020. DoWhy: An end-to-end library for causal inference. *arXiv preprint arXiv:2011.04216* (2020).
[26] IBM SPSS. 2021. SPSS Software. **https://www.ibm.com/analytics/spss-statistics-software**
[27] Susan Leigh Star and James R Griesemer. 1989. Institutional ecology,translations' and boundary objects: Amateurs and professionals in Berkeley's Museum of Vertebrate Zoology, 1907-39. *Social studies of science* 19, 3 (1989), 387–420.
[28] Johannes Textor, Juliane Hardt, and Sven Knüppel. 2011. DAGitty: a graphical tool for analyzing causal diagrams. *Epidemiology* 22, 5 (2011), 745.
[29] Tyler J VanderWeele. 2019. Principles of confounder selection. *European journal of epidemiology* 34, 3 (2019), 211–219.
[30] R Michael Winters, Bruce N Walker, and Grace Leslie. 2021. Can You Hear My Heartbeat?: Hearing an Expressive Biosignal Elicits Empathy. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–11.
[31] Kanit Wongsuphasawat, Yang Liu, and Jeffrey Heer. 2019. Goals, Process, and Challenges of Exploratory Data Analysis: An Interview Study. *arXiv preprint arXiv:1911.00568* (2019).