From Data to Hypotheses: Automating Hypothesis Generation for Visual Data Analysis

Category: Research

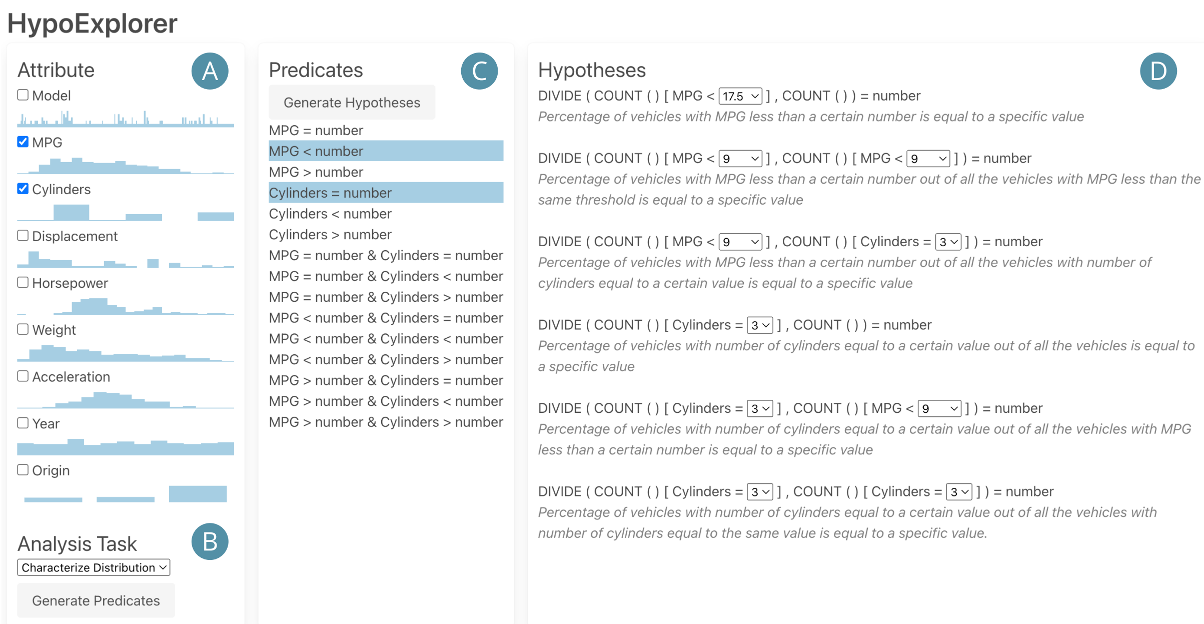

Figure 1: An example of using our proposed HypoExplorer interface with the *cars* dataset. HypoExplorer is built on top of a grammar of hypotheses to facilitate automatic hypothesis generation through three levels of specifications. (A) Users select data attributes based on their relevance to the analysis goal. (B) Users select a pre-defined analysis task (e.g., *characterize distribution*, *find anomalies*, *identify correlations*, *find extremum*) as well as (C) predicates (i.e. filters) over data to compare specific subgroups of data. Finally, (D) users are shown relevant hypotheses for data, given their specifications, in the form of grammar and natural language statements. Dropdowns are provided so finer granularity of hypotheses can be evaluated. By automatically providing relevant hypotheses, HypoExplorer can better support hypothesis discovery to inform future analyses.

Abstract

The ability to support hypothesis generation is vital to the success of a visual analytics tool, as it empowers users in uncovering insights and interesting patterns in data. However, practical techniques are still needed to guide users in formulating, refining, and presenting hypotheses within visual analytic systems. To address this missing gap, we propose HypoExplorer, an interactive visual interface that uses a grammar of hypotheses to enable users to generate hypotheses by specifying their analysis tasks and data attributes of interest. We demonstrate the effectiveness of HypoExplorer by comparing it to an existing visualization tool for generating data facts, Voder. Our case study shows that HypoExplorer can generate hypotheses similar to Voder's data facts and discover novel hypotheses beyond Voder's capabilities. The results of our case study suggest that HypoExplorer can efficiently formalize the hypothesis generation process, allowing users to generate hypotheses relevant to their data analysis goals.

Index Terms: Human-centered computing—Visualization;

1 Introduction

Hypothesis formation is a crucial component of visual analytics, enabling users to generate insights and make informed decisions [8, 14, 15]. By formulating and refining hypotheses, users can explore complex data and uncover patterns that might otherwise be hidden [32, 33]. However, despite the importance of hypothesis formation, there is still a practical gap in how to effectively operationalize this process within visual analytic workflows. While various analysis methodologies have been proposed to describe the high-level processes of hypothesis formation and validation [18, 26, 28], there still exists a practical gap in visual analytic systems for guiding users in the formulation, refinement, and presentation of hypotheses.

In this paper, we seek to address this gap with **HypoExplorer**, an interactive visual interface that supports automatic hypothesis generation given a user's data of interest, as well as their proposed analysis task. HypoExplorer is the first hypothesis-driven visual analysis interface that operationalizes the process of generating hypotheses, which can enhance the user's ability to form new insights. HypoExplorer uses a grammar-based approach (specifically, using a grammar of hypotheses [31]), such that users can perform partial (i.e. high-level) specifications – which in turn generates relevant hypotheses represented by both the grammar and with natural language. This approach offers an alternative way to analyze data, leveraging both the user's domain expertise and the grammar's ability to automatically express testable hypotheses.

HypoExplorer is designed to guide users through the partial specification process by helping them express: (1) the data attributes they are interested in gaining insights about, (2) the analysis tasks they would like support of, and (3) the regions of data (subgroups) they would like to directly compare. From these specified inputs, HypoExplorer generates a customized list of hypotheses that align with the user's overarching analysis interests, providing a starting point for further analysis and discovery. Unlike previous visual analysis systems that do not directly consider a user's analysis task, instead focusing on supporting users in openly exploring their data (e.g., [35, 36]), HypoExplorer explicitly guides hypothesis generation given a user's proposed analysis task, e.g., [2].

From HypoExplorer, we describe a concrete, four-stage workflow that can be used to operationalize hypothesis formation in future visual analysis systems by building on top of our proposed approach. As a result, visual analytic systems that employ a similar approach to HypoExplorer can more meaningfully support users in generating hypotheses that can be used or tested for downstream tasks – thereby allowing users to discover interesting insights or patterns in data without the onus of open-ended, unguided data exploration [33].

To illustrate the efficacy of HypoExplorer in generating relevant hypotheses, we provide a case study comparing to Voder [30], an interactive visualization tool that generates "data facts" (derived from Amar et al.'s analysis tasks [2]) when users upload and create visualizations for data. We demonstrate that HypoExplorer is able to provide hypotheses that model data facts generated by Voder, and can also generate additional hypotheses beyond those supported in Voder as data facts. Furthermore, we showcase the richness of HypoExplorer in allowing users to guide the hypotheses provided by the system, rather than relying on the heuristics described by Voder. In summary, by offering a concrete and effective approach to operationalizing hypothesis generation, HypoExplorer provides a demonstration of how visual analytic systems can better guide users in formulating, refining, and presenting hypotheses.

2 Motivation & Related Work

Visual analytics aims to support users in forming, refining, and validating hypotheses [14, 15], as evidenced in models such as Sensemaking Process [26] and the Knowledge Generation Model [28]. Hypotheses are viewed as essential for exploring data and constructing mental models [25]. Despite recognizing the importance of supporting users in forming hypotheses, few practical approaches exist for integrating them directly in visual data analysis.

Instead, visualization and visual analytics systems today (e.g., Tableau [35]) primarily rely on supporting users in *exploratory data analysis* (EDA) [33]. In the context of visual analysis, EDA involves using visualization to gain new insights about data [5]. With EDA, users typically conduct open-ended, unstructured, and unguided searches through data via various chart creations to identify interesting patterns and gain new knowledge about data [15, 23].

The flexibility of EDA has inspired several significant innovations in visualization. However, recent work has underscored potential hazards of relying too heavily on EDA while using visual analysis tools. For example, Zgraggen et al. found that over 60% of insights reported by users with EDA-based visualization tools were spurious or false [37]. To address potential pitfalls of EDA, *hypothesis-first* analysis methodologies [12] have become increasingly popular as an approach in HCI and visualization. Hypothesis-first analysis has inspired work in belief elicitation [1, 21], knowledge elicitation through natural language interfaces [3, 11, 22, 24], and preregistration [7, 17]. In hypothesis-first analysis, users' assumptions for data are elicited before analysis can take place.

Empirical research into the effects of hypothesis-first analysis shows promise in mitigating the shortcomings of EDA [6, 20]. Users have higher engagement with visualizations [10], they conduct more deliberate analyses [19], and have more accurate recall of data [13, 16]. However, current methods for hypothesis-first analysis largely rely on "Wizard of Oz" experiments or think-aloud protocols (e.g., [6, 20]), resulting in theoretical findings and concepts that are difficult to operationalize in real-world settings.

This paper introduces HypoExplorer to address this missing gap. HypoExplorer implements the grammar of hypotheses proposed by Suh et al. [31], enabling a formal, grammar-based representation of hypotheses. Grammars have aided the visualization research community in facilitating users through custom visualization creation with formal languages [29, 34]. We apply a comparable approach to operationalize hypothesis generation for visual data analysis. With HypoExplorer, users express partial analysis goals and tasks, and the system generates hypotheses that reflect the user's analysis interests, informing future data exploration and discovery.

3 Grammar of Hypotheses

To automatically generate hypotheses, HypoExplorer implements a grammar of hypotheses [31] in its backend. The grammar of hypotheses is based on the definition of *scientific hypotheses*, where a single hypothesis statement (that can be parsed by the grammar) represents a precise explanation about a relationship between two or more variables [27]. At the highest level, a hypothesis from the grammar is expressed as `expr op expr [pred]`, where `expr` is an expression describing: a value, a data attribute, or a qualifier over an attribute. The operator `op` compares values, and the predicate `pred` applies a filter over data.

For example, the scientific hypothesis "*as a lightbulb's voltage increases, its brightness similarly increases*" can be represented as `CORR (voltage, brightness)> 0.9` using the grammar of hypotheses, which tests whether the data attributes (`attr`) voltage and brightness have a strong, positive (`0.9`) correlation (`CORR`). Importantly, the grammar of hypotheses expresses statements that result in true or false statements (as per the constraints of scientific hypotheses being both testable and falsifiable [27]). Therefore, when HypoExplorer generates hypotheses from the grammar, the statements are those that can only be evaluated to true or false, given a tabular dataset with columns (data attributes) and rows (instances).

As part of the grammar of hypotheses, qualifiers can be added over data attributes using pre-registered functions (`func`), such as `AVG` (average), `MAX` (maximum), or `CORR` (correlation), and can include those built-in to data systems, such as relational databases utilizing SQL. Hypothesis statements can also target specific regions of data by applying predicates (`pred`), i.e. row-based filters over instances of data (e.g., `value > 0.5`). Multiple data attributes (`attr`) and values (`const`) can be compared using standard operators (`op`), such as `>`, `!=`, and so on. The full hypothesis grammar, using PEG notation [9], is:

```
hyp   :- expr op expr ("["pred"]")? ("\&" hyp)?
expr  :- func "(" (expr ("," expr)? )? ")" | var
var   :- attr ("[" pred "]")? | const
pred  :- attr op const ("\&" pred)?
func  :- str
op    :- = | < | > | = | != | ...
attr  :- str
const :- number | str | datetime | array | ...
```

For a more detailed write-up of the grammar of hypotheses, including its limitations and capabilities, we refer to the original paper [31]. Beyond the grammar's ability to express scientific hypotheses, another essential affordance is its ability to allow for partial specification in generating statements, which we discuss below.

4 HypoExplorer

We introduce HypoExplorer, a hypothesis-driven visual interface that automatically generates hypotheses about a given dataset. HypoExplorer is designed to integrate with a user's workflow by leveraging their interactions with data to generate the most relevant hypotheses. To make the underlying grammar more accessible, HypoExplorer offers an intuitive interface that allows users to partially specify components of the grammar, and uses OpenAI's GPT [4] to provide natural language explanations of the hypotheses expressed with the grammar. This approach allows users to bypass the complexity of the grammar and instead more easily generate hypotheses that align with their analytical interests.

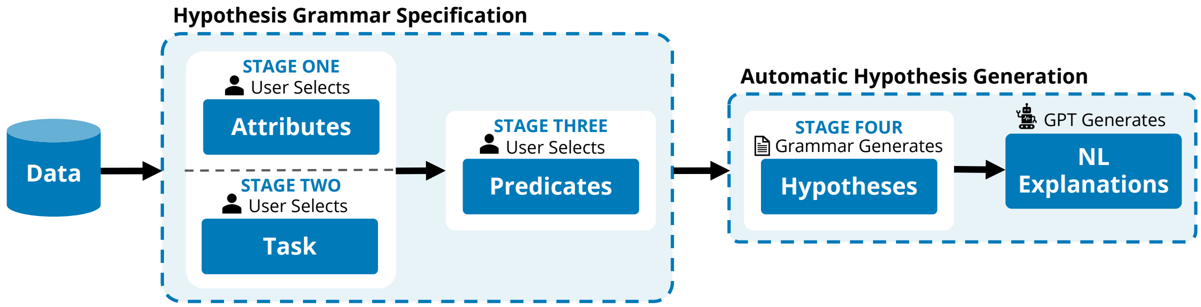

Figure 2: The workflow for HypoExplorer, consisting of four stages. From a dataset, users can select their desired (1) data attributes, (2) analysis task, and (3) predicates which are used as specifications to the hypothesis grammar in the backend. Finally, in (4) the user is shown relevant hypotheses automatically generated from the grammar. We discuss the workflow in detail in Section 4.2.

4.1 Implementing the Grammar of Hypotheses

The grammar of hypotheses implemented by HypoExplorer is a context free grammar, where each non-terminal symbol is expressed as a production rule. The process of generating a hypothesis using this grammar can be considered as iteratively expanding the root production rule until no further expansion is possible.

However, since there are many options available when expanding the production rules, there is a vast number of possible hypotheses that can be generated from the full grammar (shown in Section 3). Unfortunately, the majority of these hypotheses, while being syntactically valid, are not semantically meaningful or useful for the user's analysis goals. To address this issue, HypoExplorer supports the user by allowing *partial specification* of the grammar. Provided user-specified data attributes and a task of interest, the system can guide the expansion of the production rules, leading to the generation of hypotheses that are more relevant to the user's analysis goal.

4.2 Workflow

A user's partial specifications of their analysis interests and goals follows the workflow shown in Figure 2. HypoExplorer's workflow consists of multiple stages, where a user's interactions with the interface helps narrow down the search space of hypotheses, resulting in a smaller set of relevant and meaningful hypotheses.

Stage 1: Data Attribute Selection: In this stage, the user selects the data attributes that are relevant to their analysis goal. These attributes are grounded to the grammar of hypotheses, which specifies the `attr` production rule in the grammar (see [31] for more detail).

Stage 2: Analysis Task Selection: In this stage, the user selects an analysis task of interest. In HypoExplorer, analysis tasks are expressed using the grammar of hypotheses. In Table 1, we provide examples of encoding the low-level analysis tasks by Amar et al. [2] with the grammar. This step further narrows down the search space of hypotheses by providing more context and constraints on the hypotheses to be generated.

Stage 3: Predicate Selection: In addition to using the data attributes and analysis task as a means to reduce the search space of hypotheses, HypoExplorer also uses these specifications to specify predicates (`pred`). Predicates are used in reducing the number of "rows" in the search space, complementing the reduction of "columns" achieved by the previous stages. In this stage, the user selects the predicates that best represent the subset of rows for data that are of interest, given a generated list of possible predicates.

Stage 4: Hypothesis Generation: In this stage, HypoExplorer performs type-checking and generates all possible hypotheses based on the user's selections in the previous three stages by iteratively expanding the partially-specified production rules. The partially-specified production rules guide the search towards generating hypotheses that are relevant to the user's analysis goal.

4.3 Design Considerations

HypoExplorer is designed to provide an intuitive and easy-to-use interface for generating hypotheses, without requiring users to have in-depth knowledge about the underlying grammar of hypotheses. Towards this goal, HypoExplorer includes the following design elements and considerations:

Natural Language Translation with GPT: Hypotheses generated by HypoExplorer can be complex statements. To make HypoExplorer more accessible, we leverage the capabilities of OpenAI's GPT [4] to translate each hypothesis into a natural language statement. While the quality of the translations vary due to the stochastic nature of GPT, we have observed promising results with GPT-4, which can produce human-like explanations, as demonstrated by the translation of the hypothesis about weight and displacement:

```
CORR (Weight, Displ) [Displ < 70 & Weight < 1600] > 0.8
```

to the natural language statement: *There is a strong positive correlation (greater than 0.8) between Weight and Displacement for vehicles with Displacement less than 70 and Weight less than 1600. This suggests that, within this specific subset of vehicles, as the Displacement increases, the Weight tends to increase as well.*

"No-Code" Interface for Partial Grammar Specification: Users may similarly find it challenging to use the grammar of hypotheses for partial specification. To alleviate this burden, HypoExplorer automates the process of generating predicates and hypotheses while providing the user with the ability to customize the final hypotheses through visualization (see the dropdowns in Figure 1(D)). This enables users to tailor the hypotheses to their specific analysis tasks without requiring in-depth knowledge of the grammar.

Histogram Visualizations: HypoExplorer provides an overview of the data characteristics with a histogram for each attribute (see Figure 1(A)). This offers intuitive insights about the data, assisting the user in guiding their hypothesis specification process.

5 Case Study

To demonstrate the efficacy of HypoExplorer, we compare the hypotheses produced by HypoExplorer with the data facts of Srinivasan et al.'s visualization tool, Voder [30]. We show that HypoExplorer can generate hypotheses that would lead to all of the data facts presented by Voder. Further, we show that HypoExplorer can generate potentially interesting hypotheses that are not supported by Voder.

5.1 Voder System

Voder is a tool built around "interactive data facts" in which a list of facts computed from data is presented to users. The user has the ability to view alternative facts as well as the visualizations that can be used to confirm the data facts. Each data fact can be mapped to an analysis task from the task taxonomy by Amar et al. [2].

Analysis Task	Voder's Data Fact	HypoExplorer's Corresponding Hypothesis Statement
Characterize Distributions	Most values for Horsepower are in the range 75-125	DIVIDE (COUNT ( ) [Horsepower $>$ 75 & Horsepower $<$ 125], COUNT ( )) $>$ number
	Most cars in US have below 8 Cylinders	DIVIDE (COUNT ( ) [Cylinder < 8 & Origin = US], COUNT ( ) [Origin = US]) > number
Correlate	Cars with Origin:Japan exhibit a strong correlation between Displacement and Weight	CORR (Displacement, Weight) [Origin = Japan] > number
Find Anomalies	Pontiac Catalina appears to be an outlier	MIN (attribute [Model = Pontiac Catalina]) < Q1 - 1.5 * IQR & MAX (attribute [Model = Pontiac Catalina]) > Q3 + 1.5 * IQR
Find Extremum	Pontiac Grand Prix has highest value for Horsepower	MAX (Horsepower) [Model = Pontiac Grand Prix] = MAX (Horsepower)
	Datsun 1200 with lowest value for Weight has Cylinders:4 and Origin:Japan	MIN (Weight) [Model = Datsun 1200 & Cylinders = 4 & Origin = Japan] = MIN (Weight) [Model = Datsun 1200]

Table 1: A subset of data facts as presented in Voder [30] and the corresponding hypotheses from HypoExplorer for tasks *Characterizing Distributions*, *Correlate*, *Finding Anomalies*, and *Finding Extremum*. The full list is provided as supplemental material.

Tasks: For this case study, HypoExplorer implements the four tasks contributed by Amar et al. [2] that were used in Voder. These tasks are Characterize Distributions, Correlate, Find Anomalies, and Find Extremum; however, other analysis tasks could also be supported by HypoExplorer beyond these four (as illustrated in [31]).

Dataset: We use the same *cars* dataset used to evaluate the Voder system [30], which includes information about various models of cars, such as their MPG, Horsepower, Origin, Acceleration, etc.

5.2 Modeling Data Facts as Hypotheses

All of the data facts presented by Voder can be mapped to a corresponding hypothesis in HypoExplorer. We present a few example data facts for each of our four analysis tasks and the corresponding hypotheses produced by HypoExplorer for each in Table 1. The full comparison is provided as supplemental material.

5.3 Hypotheses Beyond Voder's Data Facts

HypoExplorer can provide users with additional hypotheses that are not demonstrable by Voder. We characterize these additional hypotheses as: *Missing or Nonexistent Data* and *False Hypotheses*.

Missing or Nonexistent Data: HypoExplorer can generate hypotheses to verify whether a subgroup of data exists in the dataset, given a particular condition. This is related to finding missing data, a prominent step in data analysis. Data facts related to missing data are not included in Voder's system.

One such hypothesis produced by HypoExplorer is:

```
DIVIDE (COUNT() [Cylinders > 6 & MPG > 32], COUNT()) = num
```

When "num" is 0, this leads to the data fact "There are no cars with more than six cylinders and MPG $>$ 32." These hypotheses can help users assess whether an absence of data is a broader characteristic of their dataset, or is the result of incomplete data.

False Hypotheses: Unlike Voder, which only presents data facts that are factually true, HypoExplorer can produce hypotheses that do not have supporting evidence from the given data, i.e. false hypotheses. For instance, the hypothesis "All cars with 6 cylinders have worse MPG than cars with 4 cylinders" can be produced as:

```
MAX (MPG [Cylinders = 4]) < MIN (MPG [Cylinders = 6])
```

However, this will not produce a data fact, since there are some cars with four cylinders with worse MPG than some with six cylinders. Evaluating this hypothesis (to false) could lead users to question a prior belief that more cylinders means less fuel efficiency.

5.4 Takeaways

HypoExplorer can produce all of the data facts presented by Voder as well as produce additional hypotheses that can lead to interesting data facts that can spark further analysis questions. These include hypotheses regarding the existence of data satisfying specific conditions, and false hypotheses. In addition, another limitation of Voder is its assumption that a user must have trust in the system to generate correct and relevant data facts (see Section 5.2 in [30]). In contrast, HypoExplorer generates hypotheses that the user can choose from and verify themselves, promoting better transparency for users.

6 Limitations & Future Work

Currently, HypoExplorer is restricted to tabular data and does not include any data transformation or pre-processing capabilities. Therefore, the data must be transformed and cleaned prior to using HypoExplorer to generate hypotheses. Moreover, HypoExplorer does not implement the full hypothesis grammar and instead enforces the depth (i.e. recursion) level to a fixed number based on the task selected by the user. Similarly, HypoExplorer does not order (i.e. rank) or further reduce the number of hypotheses beyond showing those that are based on the user's specification of data, task, and predicates. In future work, we plan to incorporate additional techniques (e.g., heuristics, statistics, databases) to resolve this limitation. We also plan to investigate the use of natural language techniques to help a user better elaborate their analysis goals.

While previous hypothesis-driven tools (e.g., PredictMe [19]) elicit the beliefs or assumptions of users from shown visualizations, HypoExplorer instead focuses on eliciting the user's analysis interests to supply hypotheses that can be used for downstream tasks. HypoExplorer also does not include any visualization creation capabilities; therefore, future systems could seek to include the capabilities of HypoExplorer to better support hypothesis-driven analysis, e.g., by providing or suggesting relevant hypotheses *during* visualization creation.

7 Conclusion

In this paper, we introduce HypoExplorer, an interactive visual interface that supports users in generating relevant hypotheses based on their analysis task, data attributes, and predicates of interest. We demonstrate the operationalization of hypothesis formation using a grammar-based approach, specifically, a grammar of hypotheses. By adopting this approach, HypoExplorer can systematically provide users with hypotheses and insights that can lead to further discoveries and more informed analyses. This paper distills a four-stage workflow that can be integrated into future visual analysis tools, as well as a case study comparing HypoExplorer to Voder. The results show that HypoExplorer is capable of generating all data facts from Voder, as well as additional interesting hypotheses. Finally, we highlight the limitations of HypoExplorer, which can inform future work in hypothesis-driven visual analysis tools and practices.

References

[1] G. Aisch, A. Cox, and K. Quealy. You draw it: How family income predicts children's college chances. *The New York Times*, 28, 2015.

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

[3] J. Aurisano. Articulate 2 : Toward a conversational interface for visual data exploration. In *IEEE Visualization*, 2016.

[4] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[5] R. Chang, C. Ziemkiewicz, T. M. Green, and W. Ribarsky. Defining insight for visual analytics. *IEEE Computer Graphics and Applications*, 29(2):14–17, 2009.

[6] I. K. Choi, N. K. Raveendranath, J. Westerfield, and K. Reda. Visual (dis)confirmation: Validating models and hypotheses with visualizations. In *2019 23rd International Conference in Information Visualization – Part II*, pp. 116–121, 2019. doi: 10.1109/IV-2.2019.00032

[7] A. Cockburn, C. Gutwin, and A. Dix. *HARK No More: On the Preregistration of CHI Experiments*, p. 1–12. Association for Computing Machinery, New York, NY, USA, 2018.

[8] K. A. Cook and J. J. Thomas. *Illuminating the Path: The Research and Development Agenda for Visual Analytics*. National Visualization and Analytics Ctr, 2005.

[9] B. Ford. Parsing expression grammars: a recognition-based syntactic foundation. In *Proceedings of the 31st ACM SIGPLAN-SIGACT symposium on Principles of programming languages*, pp. 111–122, 2004.

[10] F. Hohman, M. Conlen, J. Heer, and D. H. P. Chau. Communicating with interactive articles. *Distill*, 2020. https://distill.pub/2020/communicating-with-interactive-articles. doi: 10.23915/distill.00028

[11] J. Huang, Y. Xi, J. Hu, and J. Tao. Flownl: Asking the flow data in natural languages. *IEEE Transactions on Visualization and Computer Graphics*, 29(1):1200–1210, 2022.

[12] J. Hullman and A. Gelman. Designing for interactive exploratory data analysis requires theories of graphical inference. *Harvard Data Science Review*, 3(3), 7 2021. https://hdsr.mitpress.mit.edu/pub/w075glo6. doi: 10.1162/99608f92.3ab8a587

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

[14] D. Keim, J. Kohlhammer, G. Ellis, and F. Mansmann. *Mastering the information age: solving problems with visual analytics*. Goslar: Eurographics Association, 2010.

[15] D. A. Keim, F. Mansmann, J. Schneidewind, and H. Ziegler. Challenges in visual data analysis. In *Tenth International Conference on Information Visualisation (IV'06)*, pp. 9–16. IEEE, 2006.

[16] Y.-S. Kim, K. Reinecke, and J. Hullman. Explaining the gap: Visualizing one's predictions improves recall and comprehension of data. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pp. 1375–1386, 2017.

[17] Y.-S. Kim, L. A. Walls, P. Krafft, and J. Hullman. A bayesian cognition approach to improve data visualization. In *Proceedings of the 2019 chi conference on human factors in computing systems*, CHI '19, p. 1–14. Association for Computing Machinery, New York, NY, USA, 2019. doi: 10.1145/3290605.3300912

[18] G. Klein, J. K. Phillips, E. L. Rall, and D. A. Peluso. A data–frame theory of sensemaking. In *Expertise out of context*, pp. 118–160. Psychology Press, 2007.

[19] R. Koonchanok, P. Baser, A. Sikharam, N. K. Raveendranath, and K. Reda. Data prophecy: Exploring the effects of belief elicitation in visual analytics. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI '21. Association for Computing Machinery, New York, NY, USA, 2021. doi: 10.1145/3411764.3445798

[20] R. Koonchanok, G. Y. Tawde, G. R. Narayanasamy, S. Walimbe, and K. Reda. Visual belief elicitation reduces the incidence of false discovery. *arXiv preprint arXiv:2301.12512*, 2023.

[21] S. Mahajan, B. Chen, A. Karduni, Y.-S. Kim, and E. Wall. Vibe: A design space for visual belief elicitation in data journalism. *Computer Graphics Forum*, 41(3):477–488, 2022. doi: 10.1111/cgf.14556

[22] R. Mitra, A. Narechania, A. Endert, and J. Stasko. Facilitating conversational interaction in natural language interfaces for visualization. In *2022 IEEE Visualization and Visual Analytics (VIS)*, pp. 6–10. IEEE, 2022.

[23] A. Mosca, S. Robinson, M. Clarke, R. Redelmeier, S. Coates, D. Cashman, and R. Chang. Defining an Analysis: A Study of Client-Facing Data Scientists. In J. Johansson, F. Sadlo, and G. E. Marai, eds., *EuroVis 2019 - Short Papers*. The Eurographics Association, 2019. doi: 10.2312/evs.20191173

[24] A. Narechania, A. Srinivasan, and J. T. Stasko. Nl4dv: A toolkit for generating analytic specifications for data visualization from natural language queries. *IEEE Transactions on Visualization and Computer Graphics*, 27:369–379, 2020.

[25] W. A. Pike, J. Stasko, R. Chang, and T. A. O'connell. The science of interaction. *Information visualization*, 8(4):263–274, 2009.

[26] P. Pirolli and S. Card. The sensemaking process and leverage points for analyst technology as identified through cognitive task analysis. In *Proceedings of international conference on intelligence analysis*, vol. 5, pp. 2–4. McLean, VA, USA, 2005.

[27] K. Popper. *The logic of scientific discovery*. Routledge, 2005.

[28] D. Sacha, A. Stoffel, F. Stoffel, B. C. Kwon, G. Ellis, and D. A. Keim. Knowledge generation model for visual analytics. *IEEE transactions on visualization and computer graphics*, 20(12):1604–1613, 2014.

[29] A. Satyanarayan, D. Moritz, K. Wongsuphasawat, and J. Heer. Vega-lite: A grammar of interactive graphics. *IEEE transactions on visualization and computer graphics*, 23(1):341–350, 2016.

[30] A. Srinivasan, S. Drucker, A. Endert, and J. Stasko. Augmenting visualizations with interactive data facts to facilitate interpretation and communication. *IEEE Transactions on Visualization and Computer Graphics*, PP:1–1, 08 2018. doi: 10.1109/TVCG.2018.2865145

[31] A. Suh, A. Mosca, E. Wu, and R. Chang. A grammar of hypotheses for visualization, data, and analysis, 2023.

[32] J. W. Tukey. We need both exploratory and confirmatory. *The American Statistician*, 34(1):23–25, 1980.

[33] J. W. Tukey et al. *Exploratory data analysis*, vol. 2. Reading, MA, 1977.

[34] L. Wilkinson. *The grammar of graphics*. Springer, 2012.

[35] K. Wongsuphasawat, D. Moritz, A. Anand, J. Mackinlay, B. Howe, and J. Heer. Voyager: Exploratory analysis via faceted browsing of visualization recommendations. *IEEE Trans. Visualization & Comp. Graphics (Proc. InfoVis)*, 2016.

[36] K. Wongsuphasawat, Z. Qu, D. Moritz, R. Chang, F. Ouk, A. Anand, J. Mackinlay, B. Howe, and J. Heer. Voyager 2: Augmenting visual analysis with partial view specifications. In *ACM Human Factors in Computing Systems (CHI)*, 2017.

[37] E. Zgraggen, Z. Zhao, R. Zeleznik, and T. Kraska. Investigating the effect of the multiple comparisons problem in visual analysis. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, pp. 1–12, 2018.