Exploring the Potential of 2D Freeform Computational Notebooks with SAGECells

Jesse Harden*
Radford University
Radford, Virginia, USA
jharden@radford.edu

John Wenskovitch*
Virginia Tech
Blacksburg, Virginia, USA
jw87@vt.edu

Luc Renambot
University of Illinois, Chicago
Chicago, Illinois, USA
renambot@uic.edu

Nurit Kirshenbaum
University of Hawaii at Manoa
Honolulu, Hawaii, USA
nuritk@hawaii.edu

Jason Leigh
University of Hawaii at Manoa
Honolulu, Hawaii, USA
leighj@hawaii.edu

Chris North
Virginia Tech
Blacksburg, Virginia, USA
north@vt.edu

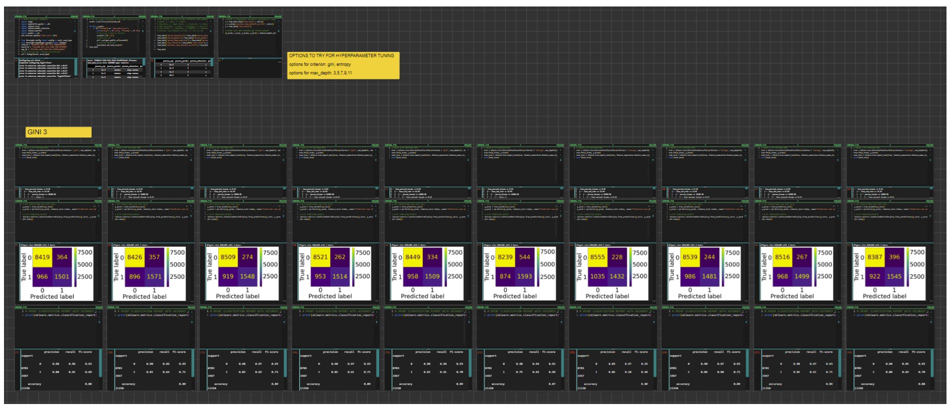

Figure 1: Example Freeform 2D Layout showing a row of cells and a row of columns of cells in SAGE3.

Abstract

Computational notebooks are popular for the iterative, exploratory, and narrative work of data science, but their one-dimensional (1D), top-down organization may clash with nonlinearity prevalent in data science work. While evaluations of rigid two-dimensional (2D) notebooks suggest enabling structured nonlinearity has benefits, users may prefer to organize cells in 2D without rigid constraints; freeform 2D notebooks thus warrant research. This work contributes a sanity check by comparing freeform 2D (SAGECells) to standard 1D (Jupyter) on coding and comparing semantically similar sections of code (*Branching Code Paths*). Freeform 2D with nonlinear cell organizations saved users approximately 30% in time taken to revisit their analysis, and users preferred freeform 2D for all surveyed data science tasks. The results suggest that further research investigating freeform 2D notebooks is warranted.

*Both authors contributed equally to this research.

Conference acronym 'XX, Woodstock, NY

ACM ISBN 978-1-4503-XXXX-X/2018/06
https://doi.org/XXXXXXX.XXXXXXX

CCS Concepts

• **Human-centered computing** → **Human computer interaction (HCI)**; **User studies**; **Graphical user interfaces**; *Information visualization*.

Keywords

computational notebooks, data science, design, evaluation, 2D

ACM Reference Format:
Jesse Harden, John Wenskovitch, Luc Renambot, Nurit Kirshenbaum, Jason Leigh, and Chris North. 2026. Exploring the Potential of 2D Freeform Computational Notebooks with SAGECells. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation email (Conference acronym 'XX)*. ACM, New York, NY, USA, 13 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

Computational notebooks (e.g. Jupyter [43, 53]) are often used in data science to construct and present computational narratives [64, 72, 78]. However, their one-dimensional (1D), top-down code cell organization impedes users in representing nonlinear analysis and narrative patterns [37, 72], using larger widescreen displays well [36, 37], and may affect issues like messiness [25, 38, 47, 56, 57, 72]. One common nonlinear pattern in data science involves semantically parallel sections of code, or "Branching Code Paths"; this can happen when a user copies and pastes (or overwrites) code to see performance changes, compares performance of several different machine learning (ML) algorithms on a single dataset, or does similar actions. This pattern serves as context for this work.

Recent studies suggest rigid, software-imposed two-dimensional (2D) code cell organizations help with issues like those above [36, 86, 87], but there is movement away from such rigidity towards user-imposed ("freeform") organizations (e.g. Einblick.ai [31, 81], Observable Canvas [11]). These design decisions impact user behaviors, perceptions, and performance, yet the authors did not find works testing if "freeform" 2D notebooks may benefit users over 1D as rigid 2D seems to; thus, the authors provide a sanity check.

This paper contributes to computational notebook research via a detailed literature review, a brief design, and a thorough evaluation of SAGECells, a freeform 2D computational notebook interface, within SAGE3 [80, 82]. We focus on the research questions below:

- **RQ1:** When comparing linear and nonlinear strategies for code cells, which supports more efficient and accurate coding and comparison for Branching Code Paths?
- **RQ2:** What strategies do users utilize in 1D and freeform 2D notebooks to deal with Branching Code Paths?
- **RQ3:** How do users organize cells in a freeform 2D notebook to deal with Branching Code Paths?
- **RQ4:** Are freeform 2D computational notebooks preferred over 1D computational notebooks, and why?

To answer these questions, we designed and implemented SAGECells, an application in SAGE3 [80] with freely organizable cells, compared it to 1D Jupyter Notebooks with a user study on a large display, and surveyed participants for qualitative data. We found, similar to our earlier work [37], that more than 67% of our participants made a nonlinear organization of cells (nonlinear strategy) when working nearly from scratch; these users tended to nest rows and columns, creating Multi-Column and Grouped Combinations patterns identified in our previous work cited above. When used with a nonlinear strategy, SAGECells saved users approximately 30% in time over Jupyter Notebooks for revisiting an analysis, but was not statistically different for setting up and running an initial analysis. Still, users tended to prefer SAGECells for all surveyed data science tasks, with the median preference for any task being no lower than Slightly Prefer 2D and more often being Prefer 2D. We discuss these results and their implications, including usability benefits, design challenges, and opportunities for further improving and investigating 2D computational notebooks.

2 Background and Related Works

2.1 Nonlinearity in Data Science and Sensemaking

While often shown as a linear progression from data collection to modeling and deployment [62], data science in practice is an iterative, exploratory process of transforming raw data into actionable knowledge [8, 67]. Discovering answers through data exploration leads to new questions arising, and goal shifting in this exploration further leads to the fluidity of data science [13, 69, 83]. Model results can uncover data quality or sample bias issues, requiring more data foraging and cleaning, as well as potentially having further impacts on modeling and analysis code [9, 99].

Many methodologies and structured frameworks developed to guide data science projects include iteration at their core; examples include Jurney's Agile Data Science Lifecycle ("*iterate, iterate, iterate*") [44], the Domino DS Lifecycle ("*expect and embrace iteration*") [39], and Guo's Data Science Workflow ("*a repeated iteration cycle*") [34]. Similarly, analysis phases exploring data, parameters, and workflows appear in methodologies like CRISP-DM [77], Byrne's Development Workflow for Data Scientists [14], and the Dutta and Bose framework for managing big data analytics projects [28].

Such actions in data science are related to the repeated information foraging and synthesis processes seen in sensemaking research [54, 66] and in cognitive processes associated with learning and decision-making [19, 73]. As new information is found and incorporated into current knowledge, existing beliefs may be revised, mental models may be transformed, and the foundations of hypotheses may need recalibration [18]. All these revisions can have both upstream and downstream impacts on already-performed analyses, requiring further iteration and realignment of prior work [42].

Given the nonlinearity in data science and sensemaking, this research helps explore a possible way to better support nonlinear sensemaking and data analysis for computational notebook users.

2.2 Limitations and Pain Points of Computational Notebooks

Computational notebooks are seen as useful for data science research because the separately-executed cell structure supports incremental and iterative analysis during exploration [45] and because the interleaving of code, visualizations, and explanatory text naturally leads to the formation of computational narratives [63]. However, the nonlinear analysis inherent to these sensemaking and exploratory processes also results in numerous problem areas for computational notebooks [16, 48, 72]. For example, interviews conducted by Kery et al. demonstrated that the incremental formation of ideas and associated code branching processes led data scientists to create complex, informal versioning behaviors to simultaneously maintain older analyses and develop new results [46]. The additional burden of tracking previous code versions and data states can be mitigated with refactoring and other cleaning actions, but these steps have their own usability challenges [25, 56].

Letting unused code remain can also lead to navigational challenges like tracing dependencies between cells, particularly in longer notebooks [37]. For example, the structure of cell dependencies and the intended cell execution order are clearer to the notebook creator than to a notebook reader, as the development and maintenance of these structures are more aligned with the mental model and information discovery processes of the creator [42, 68]. These remnants of prior analyses can further impact both collaborative work and reproducibility of results by other researchers [10, 38, 65]. Researchers and developers have implemented extensions to notebooks, as well as separate tools that complement notebooks, to work on resolving these navigation and structural issues with the addition of hyperlinked lists of section headers [32], dependency graph visualizations [89], and version control [47].

Noting that a subset of the issues associated with navigation and structure are connected to the linear structure inherent to computational notebooks, recent research is investigating the use of two-dimensional spaces and large displays [35, 86, 87]. These works have explored the potential of rigidly structured two-dimensional spatial computational notebooks, where users are mostly locked into a given organization of cells using two-dimensional space, to improve the state of computational notebooks. Fork-It [87] introduced temporary two-dimensional space usage through the creation of forks, or split columns, in a standard linear notebook; this helped nonlinear analyses, but the temporary nature of the forks meant that nonlinear narratives were not accommodated. Stickyland [86] enabled users to "stick" cells to a persistently viewable dock at the top of the notebook interface. 2D Jupyter enabled organizing code cells into a row of columns, or a multi-column pattern [37], but did not accommodate other, more complex organizations of code cells.

While such rigidly structured two-dimensional notebooks have proven beneficial, the authors are unaware of explorations involving completely freeform two-dimensional spatial computational notebooks, where users can organize code cells however they wish. This study explores the usage of this type of flexible environment.

2.3 Benefits and Limitations of Structure

Work in distributed cognition by Attfield et al. [5] and Wright et al. [95] suggests there are advantages to adding structure to an interactive sensemaking space, particularly when collaborating in the space. Structure and organization can create or extend a shared visual language, enabling participants to recognize the meaning of and relationships between on-screen artifacts, facilitating common ground in the process. The shared knowledge contained within these artifacts has been called community evidence [7]. More than just adding facts to a mental store, shared knowledge is embedded in these contextual frames [74]. Segmenting this knowledge into cells of a computational notebook can be viewed as a manifestation of a frame from Klein's data/frame sensemaking model [52], presenting a shared cognitive artifact available to all collaborators that will grow and morph as exploration and analysis unfolds.

However, there can be drawbacks to having too much structure, particularly early in an exploratory investigation. As Wong notes, an analyst typically follows a fluid process during the early stages of an investigation [93]. The intermediate analytic thoughts of users can be captured from this sensemaking process [58], demonstrating tight personal coupling between the mental model of a user and the artifacts and structures that they create. Individual users need freedom to explore the data and various hypotheses, and the structures created by these users play an important role in their personal sensemaking process [98]. A highly structured space might hamper the individual's process both physically and cognitively, as users with distinct personality types and cognitive abilities demonstrate observable differences in task-solving patterns and their related behaviors when interacting with a system [1, 94].

Prior work, including our own, has shown 2D computational notebooks with rigid layout structures benefit users [35, 86, 87]. If freeform 2D notebooks show similar benefits, as this work investigates, comparing them with rigid 2D notebooks may be worthwhile.

2.4 Data Analysis Actions in Large Displays

Interactive sensemaking tools provide a workspace for users to externalize their sensemaking process. Early versions of these spaces include the "Planning Space" tool that Wang and Haake [85] introduce within the COWFISH system, as well as the organizational workspace presented by Hsieh and Shipman [40] in the VITE system. These early experiments led to more advanced sensemaking exploration spaces, such as "The Sandbox" from Wright et al. [96] and "Space to Think" from Andrews et al. [2].

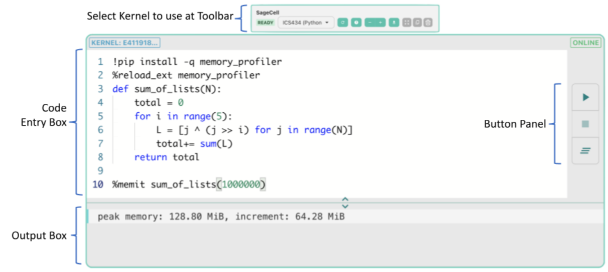

Figure 2: SAGECell Application containing the code section, the output section, the kernel and font size selection, and the execution control on the side.

Such systems provide a digital environment for knowledge externalization [61], assisting users by storing information and connections for later access [84], and anchoring the external structures to the user's cognition [6]. They have been noted as particularly helpful for the iterative nature of exploratory data analysis [49], as well as enhancing goal-directed thinking [30]. Further benefits associated with these spaces include the freeing of cognitive resources to address other analysis tasks [15, 50], providing shared artifacts to benefit communication and collaboration [51], and broadly improving comprehension and recall [17, 79].

Similar research in embodied cognition [92] highlights the personal mental meaning associated with structures and locations in these spaces [26, 71, 90], mapping semantics to position and enhancing understanding within the workspace [3]. The formation of groups within these discrete regions can connect the meanings of individual locations via similarity relationships demonstrated through spatial proximity [20, 27, 70]. Such groups can also help to reduce clutter in a workspace, compressing similar observations into a group that requires less physical or screen space [59], further supporting improved memory and recall by providing a simplified method of understanding the data [22].

Use of these grouping and spatialization sensemaking techniques is an area of research interest, including tools and techniques [24, 55, 60, 97], studies [12, 75], and surveys [29, 88]. However, the behavior of users when performing these tasks in unstructured 2D computational notebooks has not been explored. This work presents such a study.

3 Overview of SAGECells in SAGE3

SAGECells are an application developed for SAGE3 [80, 82], an online collaborative data science canvas, that act as computational notebook cells. They can be moved, resized, have their font size changed, and more; their language kernels (e.g. Python) are managed by the SAGE3 Kernel Dashboard with a JupyterLab backend; Users can create several kernels and choose which kernel to run a SAGECell on at any time using an application menu. Multiple SAGECells, in the form of a single row or column, can be run in order by selecting them via a click-and-drag maneuver; this uses an algorithm to detect whether the overall shape of the SAGECell selection is wider or taller. If wider, the SAGECells run left-to-right. If taller, the SAGECells top-to-bottom. Also, a single SAGECell or a selection of SAGECells can be duplicated (code and output) into new SAGECells to the right of the selected one(s). That being said, at the time of the study SAGECells lacked functionality to create and use directed acyclic graphs (DAGs) for connecting code cells.

Some SAGECell features not considered in this study include:

- Images in SAGECell outputs can be dragged onto the board.
- SAGECell code can be downloaded as a Python script file.
- SAGECells can be created from Python script files uploaded to SAGE3.

4 Study Methodology

We ran a controlled study with a pre-screening questionnaire, two user study sessions in 1D (Jupyter) and freeform 2D (SAGECells) respectively, and a post-study survey. We compared performance and usability of the two different environments (1D, 2D) while also considering user strategies, and we included counterbalancing; about half the participants did 1D First and the others did 2D First.

4.1 Recruitment and Screening

We recruited 65 potential participants via academic listservs of students and faculty from a large state university; they did an online screening questionnaire to ensure experience with computational notebooks and Python. 52 had experience in both Python and computational notebooks; 19 of them took part in the study, with 9 doing 1D First and 10 doing 2D First. Participants were randomly assigned such that the groups were evenly balanced.

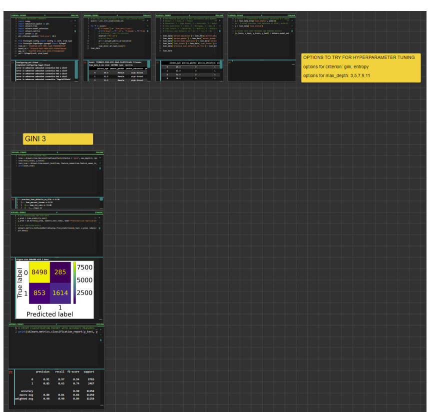

Figure 3: Starting Layout for 2D SAGECells condition. Top row is data pre-processing. The column is the first branch.

4.2 Hardware for User Study

For the user study sessions, participants used an computer and a 28" high by 48" wide monitor with a resolution of 3840x2160 pixels. The computer had a corded mouse and keyboard. The monitor was wide enough to display six to eight columns of computational notebook cells at a time.

4.3 Task Design and Rationale

We designed the user study task to emulate Branching Code Paths, a pattern of nonlinearity that can cause or exacerbate user issues with standard 1D computational notebooks. Branching Code Paths means having sections of code that are semantically parallel; results of such sections are often compared against each other. Copying and pasting cells with few code changes for comparing performance on a task (e.g. hyperparameter tuning) and testing varied machine learning algorithms on one dataset are examples of this pattern.

We created two nearly identical computational notebooks for this study, one with SAGECells and one in a Jupyter Notebook. Each notebook had a hyperparameter tuning task with a loan classification dataset using a decision tree where users would tune a combination of criterion (Gini or Entropy) and max_depth; max_depth options were 2, 4, 6, 8, and 10 for the 1D notebook and 3, 5, 7, 9, and 11 for the 2D notebook. In addition, we split the dataset into two subsets; each notebook used a different subset. These differences prevented memorization of past results from being a viable strategy while keeping the tasks practically identical; participants were not otherwise limited. Finally, both notebooks' starting states included data pre-processing and one branch of hyperparameter tuning done; participants did not need to interact with data pre-processing cells for this study. The 2D notebook's starting layout, as seen in Figure 3, included a row and a column of cells for demonstrating SAGECell features in SAGE3 as part of a brief training on the system.

To compare performance in 1D Jupyter Notebooks vs. 2D SAGECells, we measured the time it took to answer the survey questions necessary for task completion (selecting the best (highest) value for number of loans correctly approved, overall model accuracy, precision for loans approved, and F1-score for loans denied) and press the "Next" button on the survey as "Time to Completion." Accuracy was measured in terms of whether the user selected all the right answers and whether the user selected any wrong answers. Each participant took at most 1 hour and 15 minutes for each study session, with one study session for the 1D Juypter Notebook and one for the 2D SAGECells in SAGE3.

4.3.1 Hyperparameter Tuning Task.

A common data science task is tuning hyperparameters of a machine learning model to maximize performance. This study's code used a Decision Tree algorithm to classify loan applications, and involved two task passes. For the **1st Pass (Initial Pass)**, we instructed participants to code and compare the 10 branches using whatever strategy they wanted, and we warned them that they would have to re-run their analysis later so as to minimize the effects of repeated tasks between sessions on the participants. We designed the 1st Pass to see how participants tackle Branching Code Paths and measure user performance (time to completion, accuracy) in each layout setting. Participants could move cells/change cell order, write new code (e.g. loops), duplicate cells, and more. For the **2nd Pass (Revisit Pass)**, participants could use their prior work to re-run the analysis after the column the decision tree tended to split on first was dropped, or they could adopt a new strategy. We designed the 2nd Pass to emulate revisiting an analysis with different data. User performance measurements in each layout setting for this pass were gathered as separate columns from the measurements for the Initial Pass; all measurements were correctly linked to the participant whose performance generated them. In addition, we added the time taken in each pass by user into a final column of measurements, **Both Passes**. With two conditions (2D Freeform in SAGECells, 1D in Jupyter) and three passes with time to completion measurements (1st Pass, 2nd Pass, Both Passes) for each condition, we gathered a total of 6 columns of measurements.

4.3.2 Branching Code Paths in Data Science. The study task is a simple example of Branching Code Paths, chosen so all participants could complete the task with limited time. Still, this example could emulate more complex examples of Branching Code Paths in how users approach them and how they might organize branches in 2D.

4.3.3 SAGECells Training. The participants lacked experience with SAGECells, so we did 10 minutes of demonstration on navigation, running individual cells and groups of cells, and duplicating cells. Users then spent 5 to 10 minutes practicing to become more comfortable with SAGECells in SAGE3.

4.4 Survey Questions Design

We designed surveys to evaluate user perceptions of the usability of 2D SAGECells versus 1D Jupyter Notebooks to answer **RQ4**. We divide our discussion of survey questions into two areas: Session Surveys and Post-Study Survey.

4.4.1 Session Surveys. The 1D and 2D user study session surveys had identical questions except for the layout referenced. After each task pass, users rated and explained their confidence level and explained their strategies. After completing both task passes, users answered Likert-scale questions about layout usability in relation to navigation, finding information, coding and comparing Branching Code Paths, revisiting the analysis, and large display usage.

4.4.2 Post-Study Survey. After completing both sessions, users reported their perceptions of the usability of 1D and 2D computational notebooks in the Post-Study survey, starting with a 7-point Likert-scale matrix (from Strongly Prefer 2D to Strongly Prefer 1D) asking which notebook type participants would prefer for tasks such as navigation, dealing with Branching Code Paths, and data exploration and preparation. Next, users filled out a 7-point Likert-scale matrix (from Strongly Agree to Strongly Disagree) with 6 statements; half were pro-1D and the other half were pro-2D, with each kind interleaved in the statement order. Users could then elaborate on their answers to the first two Likert-scale matrix questions before proceeding. On the next page users filled out a 7-point Likert-scale matrix (from from Strongly Prefer 2D to Strongly Prefer 1D) on which notebook type they preferred for data science and analysis projects of varying length and type; they could then elaborate on their answers to this matrix question. On the final page were three qualitative answer questions focusing on differences in strategies for 1D and 2D notebooks, perceptions of the usefulness or uselessness of 2D notebooks, and what features users would like to see added to SAGECells and SAGE3, followed by a final opportunity to share any remaining thoughts.

4.5 Data Analysis Process

We divide the data analysis into three areas: Strategy Labeling, Performance and Usage Metrics, and Survey Questions.

4.5.1 Strategy Labeling. To help answer **RQ1** and to answer **RQ2**, we first labeled each strategy used by participants with SAGECells in SAGE3 as nonlinear or linear. Nonlinear strategies organize cells in 2D space such that no single straight line in the 2D plane can intersect all relevant cells; if such a single line exists, then the cell organization used is a linear strategy, and thus should be effectively replicable in a 1D, top-down notebook. To determine whether a user's strategy was linear or nonlinear, we noted the following after the Initial Pass was complete:

(1) We identified which cells the participant used to complete the task, which we call *relevant cells*.
(2) We checked if the *relevant cells* formed a single line (row, column, diagonal) or if there was only one such cell.
   - If true, we labeled the layout strategy as linear.
   - If false, we labeled the layout strategy as nonlinear.

Since switching from a linear strategy in the Initial Pass to a nonlinear strategy in the Revisit Pass required steps unrelated to revisiting the analysis that users who had already implemented a nonlinear strategy would not need to do, we labeled user strategies based on their cell organization at the time of completing the Initial Pass. Also, we only considered relevant cells to avoid classifying users who dabbled briefly with 2D space usage only to use a more familiar linear strategy (e.g. loop in a single cell) as nonlinear. Finally, we did not require users to make a nonlinear strategy in order to gauge approximately what percentage of users might not make use of 2D space, and how those who made use of 2D space did so, to answer **RQ3** and to compare this to our previous exploratory work [37].

We labeled strategies used in 1D by users after their Initial Pass, to answer **RQ2**, into two mutually-exclusive groups: Overwrite, and Looping. **Overwrite** users overwrote relevant hyperparameters and re-ran code cells without adding functionally new code; **Looping** users wrote loops to show multiple branches' results in each loop code cell's output. We also noted whether users duplicated code into new cells. Finally, we tallied how many users took notes outside the analysis screen (e.g. on paper) for both 1D and 2D.

4.5.2 Performance Metrics. For *Time to Completion*, we used counterbalancing to minimize the effects of order. Since completion time is often not normally distributed (as noted in our earlier work [35]), we took several steps to strengthen the validity of our analysis. First, we calculated the mean and standard deviation for each condition for the 1st Pass, 2nd Pass, and Both Passes. Next, we tested the differences between paired results for statistical significance with both the parametric Paired T-test and the non-parametric Wilcoxon Signed-Rank Test for Paired Samples [91] to better evaluate any potentially significant results and gather potentially useful data like Cohen's d for effect size and Cohen's d' corrected for power analysis usage [21]. Statistically significant differences were then tested for normality of distribution with the Shapiro-Wilk Test [76] and for outliers with the Grubbs Test [33]. Finally, an appropriate power analysis was conducted for significant results.

For *Accuracy*, participants earned 1 point for selecting all correct answers and an additional point for not selecting any wrong answers; with 8 points per pass, each condition (1D, 2D) had 16 possible points. A Paired T-test was used to test differences.

4.5.3 Survey Questions. For the Post-1D and Post-2D questions, we created and analyzed a boxplot (see Fig. 5) of the ratings by layout (1D or 2D). As in our previous work [35], we tested the statistical significance of differences in ratings using a Paired T-test and the Wilcoxon test per De Winter's and Doduo's work on Likert-scale analysis [23]; we then calculated mean and median differences for statistically significant results. For the Post-Study questions, we created and analyzed heatmaps of ratings and analyzed qualitative answers for themes using open coding by the first author. After the initial coding, the first author got feedback from other authors on the themes to help refine them; the first author then recoded quotes and grouped all quotes into the chosen themes.

5 Results

We divide our results into three areas: Strategies and Layouts Used, Performance Metrics, and Survey Questions.

5.1 Strategies and Layouts Used

We divide results on strategies and layouts used into two sections: Strategies and Layouts in 2D, and Strategies in 1D.

5.1.1 Strategies and Layouts in 2D. Per Table 1, most participants (68%) used a nonlinear strategy and layout (**RQ2**); these participants all also used the duplication feature of SAGECells to quickly create layouts that helped them compare the results of different branches, and one of them combined code from the starting analysis cells into one cell before duplication (**RQ3**). Review of layouts showed these participants organized code cells into two main design patterns found in our prior work [37]: Multi-Column, a single row of columns like in Fig. 1, and Grouped Combinations, a nesting of rows and columns to create more complex structures like in Fig. 4. The Grouped Combinations layouts avoided scrolling for the analysis task since they fit everything needed onto the screen; the Multi-Column layouts required some scrolling. No users made a Directed Graph layout, likely due to current limitations with SAGECells.

Table 1: Strategies and Layouts Used in 2D Notebook

Item: Subgroup	1D First	2D First	Total
2D Initial Pass: Linear Strategy	2	4	6
2D Initial Pass: Nonlinear Strategy	7	6	13
2D Duplication Feature: Used	7	7	14
2D Duplication Feature: Unused	2	3	5
2D Layout: Linear	2	4	6
2D Layout: Multi-Column	2	4	6
2D Layout: Grouped Combinations	5	2	7

Of the "Linear Strategists", four used a single column, one used a single row, and one wrote a loop in one cell to show all outputs. In the 2nd Pass in 2D, one linear strategist switched to using 2D space with a Grouped Combinations (nested rows and columns) layout, while the remaining five stuck with a Linear layout [37] and strategy (**RQ2**); this change is not reflected in Table 1, as it focuses on the initial pass. Given that 2/3 of the Linear Strategists started with 2D, it may be that some users struggle to decide how to use 2D space even after being briefly trained (**RQ3**). Finally, 9 of the 19 users took external notes in the freeform 2D condition.

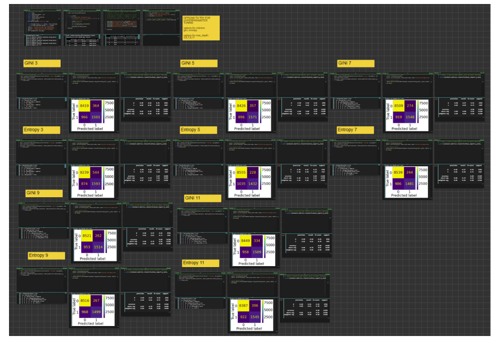

Figure 4: Example Grouped Combinations layout.

Table 2: Strategies and Layouts Used in 1D Notebook

Item: Subgroup	1D First	2D First	Total
1D Initial Pass: Overwrite Strategy	8	6	14
1D Initial Pass: Looping Strategy	1	4	5
1D Revisit Pass: Overwrite Strategy	7	6	13
1D Revisit Pass: Looping Strategy	2	4	6

5.1.2 Strategies in 1D. After the 1st Pass using a Jupyter Notebook, we found that 14 of the 19 participants used the Overwrite and Re-Run strategy; the other five used the Looping strategy (**RQ2**). One participant created a custom function to generate all outputs for each branch given the parameters and then used it in 10 cells, one for each branch. In the 2nd Pass in 1D, two of the Overwrite and Re-Run users switched to the Looping strategy. Finally, 16 of the 19 users took external notes in the 1D condition.

5.2 Performance Metrics

We divide results on performance metrics into two sections: Accuracy and Efficiency.

5.2.1 Efficiency. As seen in Table 3, when considering all participants regardless of strategy, there was not a statistically significant difference in efficiency (**RQ1**). After excluding participants who used a linear strategy in 2D, we found that the mean time to completion for 2D was lower in the 1st Pass, 2nd Pass, and Both Passes, as seen in Table 4; howwever, only the 2nd Pass had a statistically significant difference with a Cohen's d effect size of approximately 69.0%. The lack of significance for the 1st Pass may be due to time taken to duplicate and arrange code cells in 2D space, rather than starting the analysis right away as in 1D.

Our testing for normality and outliers on the differences between 1D and 2D conditions for the 2nd Pass post-filtering indicated that these paired t-test assumptions were likely met. The p-value in the Shapiro-Wilk Test for normality was approximately 0.433, well above the usual 0.05 threshold, and the G value in the Grubbs Test for outliers was approximately 1.90, less than the G-crit value of 2.33. Thus, we conducted a power analysis of the 1-tailed paired

Table 3: Pre-Filtering Time to Completion Results

Value	1st Pass	2nd Pass	Both Passes
1D Mean	919.21	504.84	1424.05
1D STD	317.98	208.48	506.84
2D Mean	1012.95	429.68	1442.63
2D STD	383.87	190.47	527.59
1-Tailed t-test p-value	0.211	0.093	0.455
1-Tailed Wilcoxon p-norm	0.266	0.089	0.476

Table 4: Post-Filtering Time to Completion Results

Value	1st Pass	2nd Pass	Both Passes
1D Mean	909.08	480.00	1389.08
1D STD	221.40	157.14	349.75
2D Mean	874.15	338.31	1212.46
2D STD	279.76	125.62	376.34
1-Tailed t-test p-value	0.289	**0.014**	0.067
1-Tailed Wilcoxon p-norm	0.390	**0.015**	0.104

Table 5: One-Tail Paired T-Test on Accuracy Results

Item	1D Mean	2D Mean	p-value
Pre-Filtering	14.94	14.11	0.092
Post-Filtering	15.08	14.58	0.194

t-test on this subset of the data using the corrected Cohen's d' (approximately 67.6%) [21]; the resulting 1-tailed paired t-test power was approximately 74.2%. If we had at least 15 participants, this test would have, by our calculations, at least 80% power.

5.2.2 Accuracy. Per Table 5, the accuracy was slightly lower on average in 2D in both pre- and post-filtering, contrary to our previous results from 2D Jupyter [36], but neither case was statistically significant. Still, this slight drop in accuracy might suggest increased cognitive load or some other challenge in 2D (**RQ1**).

5.3 Survey Questions

We divide survey results into three areas: Post-1D and Post-2D Questions, Post-Study Questions, and Qualitative Comments.

5.3.1 Post-1D and Post-2D Questions. Per Fig. 5, freeform 2D (SAGECells) seems similar to 1D in ease of navigation and quickly finding info; 1D seems worse in ease of coding branches, comparing branches, revisiting and relearning, and large screen use. However, in our study only differences in comparing branches, revisiting and relearning, and large screen use were statistically significant, with 2D being preferred per the positive values of the mean in Table 6 (**RQ4**). Of note is that (Ease of) Revisit and Relearn had the smallest difference, with a mean less than 1 and a median of 0.

5.3.2 Post-Study Questions. Per the heatmaps in Figs. 6, 7, and 8, the vast majority of participants had a positive view of the freeform 2D notebook environment. Per Fig. 6, the median results found 2D at least slightly preferable for each common data science task mentioned; no participants preferred 1D for comparing results, while six users preferred navigation in 1D (**RQ2**). Per Fig. 7, the median

Table 6: Post-2D - Post-1D *p-values* and Differences

Item	*Paired T-Test*	*Wilcoxon*	**Mean**	**Median**
Navigation	0.607	0.593	-	-
Quickly Find Info	0.285	0.334	-	-
Coding Branches	0.097	0.098	-	-
Comparing Branches	0.008	0.013	1.421	1.000
Revisit and Relearn	0.039	0.045	0.789	0.000
Use of Large Screen	0.035	0.043	1.368	1.000

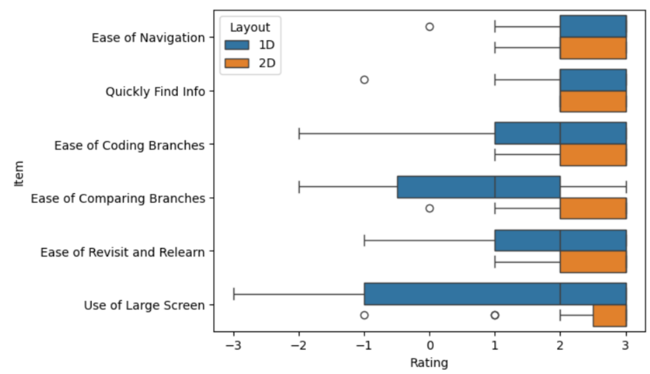

Figure 5: Boxplots of subjective usability results by layout.

results never found 1D better than 2D, even with the pro-1D statements. Notably, participants found having more notebook cells on the screen beneficial, setup time in 2D worthwhile, and 2D notebooks more preferable than 1D notebooks. Still, seven participants found the simplicity of 1D notebooks beneficial. Per Fig. 8, participants largely preferred 2D for larger, more complex projects and 1D for short, small projects; only one participant preferred 1D over 2D for any of the three larger, more complex project types (**RQ4**).

5.3.3 Qualitative Responses. Of the 19 participants, all gave some qualitative feedback; 17 participants expressed positive aspects of the 2D notebook environment, eight expressed challenges of the 2D notebook environment, 10 expressed drawbacks of the 1D notebook, and only one out of all 19 expressed a preference for 1D over 2D. Several comments point out how the 2D notebook environment with the large display effectively enables more externalized memory and spatial encoding of meaning, both key features of Space to Think [2], along with the increased efficiency from physical navigation [4] as opposed to the required virtual navigation in 1D (**RQ2**). The results are summarized in Table 7 with the major themes found, a sample quote for each theme, and the number of participants who gave a comment matching the theme.

6 Discussion

We focus discussion on 4 areas: How Affordances Affect Layouts, Efficiency in Linear vs. Nonlinear Notebooks, Usability Benefits of Nonlinear Notebooks, and Design Challenges and Opportunities. We also note limitations in our study, as well as future research directions inspired by this work.

TASK	Strongly Prefer 2D	Prefer 2D	Slightly Prefer 2D	Neutral	Slightly Prefer 1D	Prefer 1D	Strongly Prefer 1D	Median
Navigating through the notebook	4	3	4	2	4	2	0	Slightly Prefer 2D
Locating items (code, results) in the notebook	6	5	7	0	0	1	0	Prefer 2D
Dealing with Branching Code Paths	6	8	1	2	0	2	0	Prefer 2D
Organizing/cleaning a notebook	7	4	2	3	2	1	0	Prefer 2D
Presenting a notebook	7	5	2	3	0	1	1	Prefer 2D
Data exploration and preparation	4	5	2	5	2	1	0	Slightly Prefer 2D
Performing analysis and development	6	3	5	3	2	0	0	Slightly Prefer 2D
Comparing results	9	9	1	0	0	0	0	Prefer 2D
Collaborating on a shared notebook	5	4	2	6	0	1	1	Slightly Prefer 2D

Figure 6: Heatmap of Preferred Layout for Various Data Science Tasks.

CONDENSED STATEMENT	Strongly Agree	Agree	Agree a little	Neutral	Disagree a little	Disagree	Strongly Disagree	Median
1D Simplicity improved performance over 2D Complexity	1	2	4	1	4	4	3	Disagree a little
1D used available screen space better than 2D	0	2	0	3	5	5	4	Disagree a little
1D increased my confidence in my answers over 2D	1	1	1	9	2	3	2	Neutral
2D showing more code cells improved performance over 1D	8	7	3	1	0	0	0	Agree
2D setup time is worth it for benefits provided	7	7	4	1	0	0	0	Agree
2D notebooks are what would I would use instead of 1D	6	6	6	1	0	0	0	Agree

Figure 7: Heatmap of Agreement Level with Statements (Pro-1D statements first and then Pro-2D statements).

CONDENSED STATEMENT	Strongly Prefer 2D	Prefer 2D	Slightly Prefer 2D	Neutral	Slightly Prefer 1D	Prefer 1D	Strongly Prefer 1D	Median
Short, small data analysis project (few branches, iterations)	1	0	2	4	2	7	3	Prefer 1D
Longer-term data analysis project (many branches, iterations)	4	10	2	3	0	0	0	Prefer 2D
Project with Hyperparameter tuning	5	8	4	2	0	0	0	Prefer 2D
Project with comparing several different algorithms	7	8	2	1	1	0	0	Prefer 2D

Figure 8: Heatmap of Preferred Layout for Various Data Science Project Types.

6.1 How Affordances Affect Layouts

Of note is that 37% of our users made a Grouped Combinations layout, while only 24% made such a layout in our earlier exploratory work [37]. One should note that the set of software affordances available in the freeform nonlinear notebook of SAGECells differs from the affordances suggested by the use of Miro boards as a scratch space to explore nonlinear arrangements; SAGECells, at the time of this study, could not make directed graphs for run order, and the screen space provided could not show all 10 columns at once while maintaining legibility for code and results. Given the available screen space and features, especially the ability to run a single column or row of cells in order, it makes sense that more participants made a Grouped Combinations layout than a Multi-Column or Linear layout, separately; after all, Grouped Combinations layouts could fit everything on the screen in a way that makes sense to users and makes the most of the available affordances (**RQ3**).

Fork-It by Weinman et al. [87] introduces an interesting affordance: forking, or the temporary creation of a multi-column portion, or fork, within a linear notebook. They found users appreciated forking for exploration and even containing messes. However, the ephemeral nature of forking limits its applicability for revisiting and/or reusing explored alternatives, as well as for presenting narratives outside the main story. Users in our study, on average, reported preferring 2D notebooks for presenting and slightly preferring 2D notebooks for exploring data and performing analyses; this suggests that preserving explored alternatives for potential later use, a capability that SAGECells enable, is valuable (**RQ4**). Still, there may be value in permanently discarding some explored alternatives, perhaps to avoid spatialized messiness.

6.2 Efficiency in Linear vs. Nonlinear Notebooks

In our previous work on 2D Jupyter [36], we found efficiency gains on common comparative analysis tasks with rigidly structured 2D notebooks. However, their analysis did not include setup time, as the notebooks were pre-made, limiting the efficiency gains to the comparative analyses alone. This work did include setup time in the Initial Pass, as the notebooks only had data preparation code and one branch of the analysis included in their starting form, as seen in Fig. 3. While we did not see a statistically significant difference in efficiency between the 1D and 2D notebooks on the Initial Pass and Both Passes combined for the task we laid out, we did see such a difference on the Revisit Pass (**RQ1**). Given how comparative analyses and revisiting and reusing analyses, one of which has been demonstrated to benefit from 2D notebooks [36] and the latter which we demonstrate in this paper, are so critical in data science work, the authors contend that the nature of data science work leads to

Table 7: Qualitative Themes in Study Surveys

Theme	*Sample Quote*	*# Participants*
2D Benefits		**17**
Easier Comparison vs. 1D	"[2D] is very useful for data analysis where we need to compare between multiple things."	11
External Memory	"The ability to look at multiple data simultaneously is the best usefulness of the 2D computational notebook."	4
Spatial Encoding Meaning	"I think 2D computational notebooks are very useful for data science problems because I feel like the way you can organize information is better than the way you would with a 1D notebook."	3
Physical Navigation Good	"The rapidness of moving your head side to side to compare data or moving your eyes up and down is much easier [compared to] having to scroll down a page."	3
Easier Navigation	"A 2D Notebook was much easier to navigate and perform data analysis as compared to 1D notebook."	3
Good Use of Display	"With the 2D notebook, I could arrange the cells in different groups and fully utilize the screen space."	3
2D Challenges		**8**
Training Needed	"2D is helpful, however the time required to understand what to do could enhance the barrier to entry."	5
More Complex Navigation	"I think if I was more comfortable with the controls, I would be able to navigate around more easily."	5
Setup Time Needed	"There is more time to set up in 2D versus opening up a 1D notebook and start typing."	2
1D Challenges		**10**
Virtual Navigation Needed	"For 1D notebook, I had to repeatedly scroll up and down..."	6
External Notes Needed	"I had to note down all the values [in 1D], lot of manual effort required for this."	4
Layout Preferences		**5**
Prefer 2D	"SAGE is a really amazing platform to perform data analysis... I think this really has a good potential to improve developer productivity and CS education."	4
Prefer 1D	"I preferred 1D, I think it's cleaner and easier to navigate and I also think it has the same flexibility as 2D. I could take notes on paper, so this would make up for the shortcomings when using 1D."	1

solutions that can effectively utilize 2D space. While the Initial Pass did not have a statistically significant difference, this could be due to the lack of sufficient practice with SAGECells, as well as the potential for more upfront cognitive load due to having to decide how to organize cells. The freeform nature of SAGECells, in enabling user-imposed structure as opposed to software-imposed structure, does require users to make organizational decisions; while this setup time comes at a cost, our participants agreed that the cost seemed worth it, per Fig. 7, for the benefits provided.

6.3 Usability Benefits of Nonlinear Notebooks

SAGECells, as a freeform, nonlinear 2D computational notebook, may provide benefits over rigid 1D notebooks. Per Fig. 6, our users preferred locating items, comparing results, presenting a notebook, and dealing with Branching Code Paths in a 2D notebook (**RQ2**). These potential benefits, per Table 7 and past literature [2, 4], may be due to how nonlinear notebooks, when combined with a large display, can enable Space to Think [2]. Four participants noted the value of externalizing memory on the screen, and three noted the ability to organize information spatially, which are the key components of Space to Think [2]. Add in the ability to use physical navigation to (mostly) replace virtual navigation, which according to the quoted participant makes navigation "much easier", and performing tasks can become easier and more efficient [4].

6.4 Design Challenges and Opportunities

While freeform nonlinear 2D notebooks have benefits, they also have challenges like cognitive load and more complex navigation.

SAGECells, as a completely freeform 2D notebook, does not impose a particular structure on users; instead, users can create their own structures that may use their screen space and mental models better. However, having to organize code cells seems to have a cognitive cost (**RQ4**). With how approximately 25% of our participants felt more training and practice with SAGECells would improve their performance, it becomes clear there is a cognitive hurdle that thoughtful solutions could help address. Potential solutions for this issue include the creation and use of organizational templates, having an AI agent suggest, upon request, organizational patterns, and giving users more tools (e.g. creating a column- or row-based cell group) with which to impose their own structure upon the 2D canvas. Whether such tools would be effective enough and how effective they would be is an open area for future research.

On another note, while three users expressed that navigation was easier in SAGE3 with SAGECells, five users' sentiments suggest freeform 2D notebooks require more complex navigation than 1D notebooks (**RQ4**). Although this issue for freeform 2D notebooks could be addressed through more practice and experience, it is important for designers to try to minimize the cost of increased complexity in virtual navigation while maximizing the usefulness of physical navigation. After all, large physical displays are not always accessible, especially when traveling, so freeform 2D computational notebook designers should strive to make virtual navigation easy and intuitive while also empowering physical navigation.

6.5 Limitations

Our work may have limitations such as demand bias, large display as confound, ecological validity, and small sample size.

6.5.1 Demand Bias. Demand bias occurs when participants know, or think they know, the research agenda and thus change their actions; our results may be affected by this. Still, given the qualitative feedback, this potential effect might not be a major concern. For example, some of the expressed benefits of freeform 2D notebooks relate to prior research on large displays for sensemaking (Space to Think [2] and physical navigation [4] especially), and the participants were likely generally unaware of this prior research; thus, the authors suggest genuine benefits have been uncovered.

6.5.2 Large Display as Confound. This study used a single large high-resolution display with greater width than height; such a screen may benefit the 2D study condition without providing similar benefits for the 1D study condition, introducing a confounding factor. Similarly, the size of the large display used is atypical for most data science practitioners. Still, it is possible that freeform 2D notebooks could still benefit such users even with a more moderately-sized screen; after all, our work on 2D Jupyter [36] used a mere 24-inch monitor and still found statistically significant benefits. Furthermore, as larger, higher-resolution displays continue to become more affordable, the atypicality of larger screens may change.

6.5.3 Ecological Validity. The ecological validity of our experiment is bounded by the artificial constraints imposed by the study protocol. Participants performed a predefined analytic task within a standardized interface using supplied data and standardized analysis processes. This contrasts with real-world data science practice, where problem framing, data acquisition, iterative hypothesis generation, and collaborative code sharing evolve organically over longer time scales. Furthermore, the study task is a simple example of Branching Code Paths and may benefit more from some SAGECells features (e.g. duplicating cells with code) than other, more complex instances of the pattern. While additional features and user adaptions could help address these issues, the degree to which our findings translate to everyday analytical work remains uncertain.

6.5.4 Small Sample Size. Our study had a relatively small cohort of participants. While this enabled intensive measurements, it may have limited statistical power and increased vulnerability to other confounds (e.g. individual differences in spatial cognition). Additionally, the modest sample size restricts the precision of effect size estimates and precludes robust subgroup analysis (e.g., by age or domain expertise). Our further filtering of participants for additional analysis extended this limitation. While we worked to address the concern of generalizability with our power analysis, the potential for additional confounds to have affected our results remains.

6.6 Future Work

This study compared a rigid 1D notebook with a freeform 2D notebook to test, as a sanity check, for differences in performance and perceptions; it also explored user behavior in an implemented freeform 2D notebook. Future freeform 2D notebook research could study usage "in the wild" and run further controlled experiments.

"In the wild" testing may more clearly elucidate benefits and challenges of using freeform 2D notebooks to better inform future designs; such works could provide more ecologically valid observations on user behaviors and perceptions, and the effects of variables like screen size, task types, other nonlinear patterns, design decisions (e.g. rigid vs. freeform), and more. Further controlled experiments could build on such "in the wild" works to measure and statistically test effects of variables like those above. Of particular interest are how screen size differences affect behaviors, perceptions, and outcomes with freeform 2D notebooks, how rigid 2D compares to freeform 2D, and what causes such differences. Comparing 2D notebooks with 3D notebooks (e.g. In et al [41]) may also prove valuable. These research directions may help inform the next generation of spatialized notebooks.

7 Conclusion

Computational notebooks are a versatile, popular tool for creating and presenting computational narratives, but their linear, 1D, top-down organization may contribute to user issues, such as using larger displays effectively and dealing with Branching Code Paths. Rigid 2D computational notebooks showed benefits, but users may prefer more control; thus, we designed and evaluated SAGECells as a sanity check on the potential of freeform 2D notebooks.

The freeform nonlinear 2D computational notebook environment of SAGECells in SAGE3 [80], when used with a nonlinear strategy, provides some benefits in efficiency as it relates to revisiting and reusing analyses, as well as comparative analyses, by enabling physical navigation and Space to Think [2]. Most users utilized the 2D space to create nonlinear layouts with tools like duplication of code cells, especially in the Multi-Column and Grouped Combinations patterns found in our previous work [37]. While a freeform, user-imposed approach to generating structures, like those mentioned above, does require additional work by users, its benefits appear to be worth the investment according to our study participants. Overall, freeform 2D computational notebooks have the potential to improve upon the current, rigid state of computational notebook organization through enabling innovative spatial patterns and Space to Think.

Acknowledgments

This work was supported in part by the National Science Foundation awards CSSI-2004014, 2003800 and 2003387.

References

[1] Icek Ajzen. 2005. *Attitudes, Personality and Behaviour.* McGraw-Hill Education. https://books.google.com/books?id=dmJ9EGEy0ZYC

[2] Christopher Andrews, Alex Endert, and Chris North. 2010. Space to think: large high-resolution displays for sensemaking. In *Proceedings of the 28th International Conference on Human Factors in Computing Systems, CHI 2010, April 10-15, 2010*, Elizabeth D. Mynatt, Don Schoner, Geraldine Fitzpatrick, Scott E. Hudson, W. Keith Edwards, and Tom Rodden (Eds.). ACM, Atlanta, Georgia, USA, 55–64. doi:10.1145/1753326.1753336

[3] Christopher Andrews and Chris North. 2012. Analyst's Workspace: An embodied sensemaking environment for large, high-resolution displays. In *2012 IEEE Conference on Visual Analytics Science and Technology (VAST)*. 123–131. doi:10.1109/VAST.2012.6400559

[4] Christopher Andrews and Chris North. 2013. The Impact of Physical Navigation on Spatial Organization for Sensemaking. *IEEE Transactions on Visualization and Computer Graphics* 19, 12 (Dec 2013), 2207–2216. doi:10.1109/TVCG.2013.205

[5] Simon Attfield, Bob Fields, and Chris Baber. 2018. A Resources Model for Distributed Sensemaking. *Cognition, Technology & Work* 20, 4 (2018), 651–664. doi:10.1007/s10111-018-0529-4

[6] David Paul Ausubel. 2012. *The Acquisition and Retention of Knowledge: A Cognitive View.* Springer Science & Business Media. doi:10.1007/978-94-015-9454-7

[7] C Baber, K Cook, S Attfield, LM Blaha, A Endert, and L Franklin. 2018. A Conceptual Model for Mixed-Initiative Sensemaking. In *ACM SIGCHI 2018 "Sensemaking in a Senseless World" Workshop.* https://drive.google.com/file/d/1MKgG84BipPpCcbb6WgtpaHtd3AwaveMj/view

[8] Sumon Biswas, Mohammad Wardat, and Hridesh Rajan. 2022. The art and practice of data science pipelines: A comprehensive study of data science pipelines in theory, in-the-small, and in-the-large. In *Proceedings of the 44th International Conference on Software Engineering* (Pittsburgh, Pennsylvania) *(ICSE '22)*. Association for Computing Machinery, New York, NY, USA, 2091–2103. doi:10.1145/3510003.3510057

[9] Tobias Bleifuß, Leon Bornemann, Theodore Johnson, Dmitri V. Kalashnikov, Felix Naumann, and Divesh Srivastava. 2018. Exploring change: a new dimension of data analytics. *Proc. VLDB Endow.* 12, 2 (Oct. 2018), 85–98. doi:10.14778/3282495.3282496

[10] Marcel Borowski, Johannes Zagermann, Clemens N. Klokmose, Harald Reiterer, and Roman Rädle. 2020. Exploring the Benefits and Barriers of Using Computational Notebooks for Collaborative Programming Assignments. In *Proceedings of the 51st ACM Technical Symposium on Computer Science Education* (Portland, OR, USA) *(SIGCSE '20)*. Association for Computing Machinery, New York, NY, USA, 468–474. doi:10.1145/3328778.3366887

[11] Mike Bostock. 2025. Introducing Observable Canvases | Observable. https://observablehq.com/blog/introducing-canvases-early-access

[12] Matthew Brehmer, Michael Sedlmair, Stephen Ingram, and Tamara Munzner. 2014. Visualizing Dimensionally-Reduced Data: Interviews with Analysts and a Characterization of Task Sequences. In *Proceedings of the Fifth Workshop on Beyond Time and Errors: Novel Evaluation Methods for Visualization* (Paris, France) *(BELIV '14)*. ACM, New York, NY, USA, 1–8. doi:10.1145/2669557.2669559

[13] Marc Bron, Jasmijn Van Gorp, and Maarten de Rijke. 2016. Media studies research in the data-driven age: How research questions evolve. *Journal of the Association for Information Science and Technology* 67, 7 (2016), 1535–1554. arXiv:https://asistdl.onlinelibrary.wiley.com/doi/pdf/10.1002/asi.23458 doi:10.1002/asi.23458

[14] Ciara Byrne. 2017. *Development Workflows for Data Scientists.* O'Reilly Media.

[15] Stuart K Card, Jock Mackinlay, and Ben Shneiderman. 1999. *Readings in Information Visualization: Using Vision to Think.* Morgan Kaufmann.

[16] Souti Chattopadhyay, Ishita Prasad, Austin Z Henley, Anita Sarma, and Titus Barik. 2020. What's Wrong with Computational Notebooks? Pain Points, Needs, and Design Opportunities. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems.* ACM, Honolulu, HI, USA, 1–12. doi:10.1145/3313831.3376729

[17] Yang Chen, Scott Barlowe, and Jing Yang. 2010. Click2Annotate: Automated Insight Externalization with rich semantics. In *2010 IEEE Symposium on Visual Analytics Science and Technology.* 155–162. doi:10.1109/VAST.2010.5652885

[18] Michelene TH Chi. 2008. Three Types of Conceptual Change: Belief Revision, Mental Model Transformation, and Categorical Shift. *International Handbook of Research on Conceptual Change* 61 (2008), 82. doi:10.4324/9780203874813

[19] C.W. Choo. 1996. The Knowing Organization: How Organizations Use Information to Construct Meaning, Create Knowledge, and Make Decisions. *International Journal of Information Management* 16, 5 (1996), 329 – 340. doi:10.1016/0268-4012(96)00020-5

[20] C Chrea, Dominique Valentin, Claire Sulmont-Rossé, D Hoang Nguyen, and Hervé Abdi. 2005. Semantic, typicality and odor representation: A cross-cultural study. *Chemical senses* 30, 1 (2005), 37–49. doi:10.1093/chemse/bjh255

[21] Jacob Cohen. 2013. *Statistical power analysis for the behavioral sciences.* Routledge. doi:10.4324/9780203771587

[22] Jacqueline M Curiel and Gabriel A Radvansky. 1998. Mental Organization of Maps. *Journal of Experimental Psychology: Learning, Memory, and Cognition* 24, 1 (1998), 202. doi:10.1037/0278-7393.24.1.202

[23] Joost CF de Winter and Dimitra Dodou. 2010. Five-point likert items: t test versus Mann-Whitney-Wilcoxon (Addendum added October 2012). *Practical Assessment, Research, and Evaluation* 15, 1 (2010). doi:10.7275/bj1p-ts64

[24] Chris Ding and Tao Li. 2007. Adaptive Dimension Reduction Using Discriminant Analysis and K-Means Clustering. In *Proceedings of the 24th International Conference on Machine Learning* (Corvalis, Oregon, USA) *(ICML '07)*. ACM, New York, NY, USA, 521–528. doi:10.1145/1273496.1273562

[25] Helen Dong, Shurui Zhou, Jin L.C. Guo, and Christian Kästner. 2021. Splitting, Renaming, Removing: A Study of Common Cleaning Activities in Jupyter Notebooks. In *2021 36th IEEE/ACM International Conference on Automated Software Engineering Workshops (ASEW)*. 114–119. doi:10.1109/ASEW52652.2021.00032

[26] Paul Dourish, John Lamping, and Tom Rodden. 1999. Building Bridges: Customisation and Mutual Intelligibility in Shared Category Management. In *Proceedings of the International ACM SIGGROUP Conference on Supporting Group Work* (Phoenix, Arizona, USA) *(GROUP '99)*. ACM, New York, NY, USA, 11–20. doi:10.1145/320297.320299

[27] Peter Dunn-Rankin, Gerald A Knezek, Susan R Wallace, and Shuqiang Zhang. 1983. *Scaling methods.* Psychology Press.

[28] Debprotim Dutta and Indranil Bose. 2015. Managing a Big Data Project: The Case of Ramco Cements Limited. *International Journal of Production Economics* 165 (2015), 293–306. doi:10.1016/j.ijpe.2014.12.032

[29] Alex Endert, William Ribarsky, Cagatay Turkay, BL William Wong, Ian Nabney, I Díaz Blanco, and Fabrice Rossi. 2017. The State of the Art in Integrating Machine Learning into Visual Analytics. In *Computer Graphics Forum*, Vol. 36. Wiley Online Library, 458–486. doi:10.1111/cgf.13092

[30] Linda Flower and John R. Hayes. 1981. A Cognitive Process Theory of Writing. *College Composition and Communication* 32, 4 (1981), 365–387. http://www.jstor.org/stable/356600

[31] Ali Ghodsi, Ted Tomlinson, Patrick Wendell, Prem Prakash, and Cassie Miao. 2025. Databricks + Einblick | Databricks Blog. https://www.databricks.com/blog/welcome-data-intelligence-platform-databricks-einblick

[32] Google. 2025. Welcome to Colaboratory - Colaboratory. https://colab.research.google.com/

[33] Frank E. Grubbs. 1950. Sample Criteria for Testing Outlying Observations. *The Annals of Mathematical Statistics* 21, 1 (1950), 27 – 58. doi:10.1214/aoms/1177729885

[34] Philip Gia Guo. 2012. *Software Tools to Facilitate Research Programming.* PhD Thesis. Stanford University, Palo Alto, CA. https://purl.stanford.edu/mb510fs4943.

[35] Jesse Harden. 2023. Exploring and Evaluating the Potential of 2D Computational Notebooks. In *Companion Proceedings of the 2023 Conference on Interactive Surfaces and Spaces* (Pittsburgh, PA, USA) *(ISS Companion '23)*. Association for Computing Machinery, New York, NY, USA, 97–99. doi:10.1145/3626485.3626554

[36] Jesse Harden, Elizabeth Christman, Nurit Kirshenbaum, Mahdi Belcaid, Jason Leigh, and Chris North. 2023. "There is no reason anybody should be using 1D anymore": Design and Evaluation of 2D Jupyter Notebooks. In *Graphics Interface 2023* (Victoria, British Columbia, Canada). OpenReview.net. https://par.nsf.gov/biblio/10502316

[37] Jesse Harden, Elizabeth Christman, Nurit Kirshenbaum, John Wenskovitch, Jason Leigh, and Chris North. 2022. Exploring Organization of Computational Notebook Cells in 2D Space. In *2022 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*. IEEE, Rome, Italy, 1–6. doi:10.1109/VL/HCC53370.2022.9833128

[38] Andrew Head, Fred Hohman, Titus Barik, Steven M. Drucker, and Robert DeLine. 2019. Managing Messes in Computational Notebooks. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–12. doi:10.1145/3290605.3300500

[39] Nicholas J Hotz and Jeffrey Saltz. 2018. Domino Data Science Lifecycle–Data Science Project Management. *The Practical Guide to Managing Data Science at Scale. Domino Data Lab* (2018). https://domino.ai/resources/managing-data-science.

[40] Haowei Hsieh and Frank M. Shipman. 2002. Manipulating Structured Information in a Visual Workspace. In *Proceedings of the 15th Annual ACM Symposium on User Interface Software and Technology* (Paris, France) *(UIST '02)*. Association for Computing Machinery, New York, NY, USA, 217–226. doi:10.1145/571985.572018

[41] Sungwon In, Eric Krokos, Kirsten Whitley, Chris North, and Yalong Yang. 2024. Evaluating Navigation and Comparison Performance of Computational Notebooks on Desktop and in Virtual Reality. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 606, 15 pages. doi:10.1145/3613904.3642932

[42] David H. Jonassen and Philip Henning. 1999. Mental Models: Knowledge in the Head and Knowledge in the World. *Educational Technology* 39, 3 (1999), 37–42. http://www.jstor.org/stable/44428530

[43] Project Jupyter. 2021. Project Jupyter | Home. https://jupyter.org/

[44] Russell Jurney. 2017. *Agile Data Science 2.0: Building Full-Stack Data Analytics Applications with Spark*. O'Reilly Media, Inc.

[45] 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 (Dec 2012), 2917–2926. doi:10.1109/TVCG.2012.219

[46] Mary Beth Kery, Amber Horvath, and Brad Myers. 2017. Variolite: Supporting Exploratory Programming by Data Scientists. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '17)*. Association for Computing Machinery, New York, NY, USA, 1265–1276. doi:10.1145/3025453.3025626

[47] Mary Beth Kery and Brad A. Myers. 2018. Interactions for Untangling Messy History in a Computational Notebook. In *2018 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*. 147–155. doi:10.1109/VLHCC.2018.8506576

[48] Mary Beth Kery, Marissa Radensky, Mahima Arya, Bonnie E. John, and Brad A. Myers. 2018. The Story in the Notebook: Exploratory Data Science using a Literate Programming Tool. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–11. doi:10.1145/3173574.3173748

[49] Yea-Seul Kim, Nathalie Henry Riche, Bongshin Lee, Matthew Brehmer, Michel Pahud, Ken Hinckley, and Jessica Hullman. 2019. Inking Your Insights: Investigating Digital Externalization Behaviors During Data Analysis. In *Proceedings of the 2019 ACM International Conference on Interactive Surfaces and Spaces* (Daejeon, Republic of Korea) *(ISS '19)*. Association for Computing Machinery, New York, NY, USA, 255–267. doi:10.1145/3343055.3359714

[50] David Kirsh. 2010. Thinking with External Representations. In *AI & Society*, Vol. 25. Springer, 441–454. doi:10.1007/s00146-010-0272-8

[51] Nurit Kirshenbaum, Kylie Davidson, Jesse Harden, Chris North, Dylan Kobayashi, Ryan Theriot, Roderick S. Tabalba, Michael L. Rogers, Mahdi Belcaid, Andrew T. Burks, Krishna N. Bharadwaj, Luc Renambot, Andrew E. Johnson, Lance Long, and Jason Leigh. 2021. Traces of Time through Space: Advantages of Creating Complex Canvases in Collaborative Meetings. *Proc. ACM Hum.-Comput. Interact.* 5, ISS, Article 502 (Nov. 2021), 20 pages. doi:10.1145/3488552

[52] G. Klein, B. Moon, and R. R. Hoffman. 2006. Making Sense of Sensemaking 2: A Macrocognitive Model. *IEEE Intelligent Systems* 21, 5 (2006), 88–92. doi:10.1109/MIS.2006.100

[53] Thomas Kluyver, Benjamin Ragan-Kelley, Fernando Pérez, Brian E Granger, Matthias Bussonnier, Jonathan Frederic, Kyle Kelley, Jessica B Hamrick, Jason Grout, Sylvain Corlay, Paul Ivanov, Damián Avila, Sofia Abdalla, Carol Willing, et al. 2016. *Jupyter Notebooks - a publishing format for reproducible computational workflows*. IOS Press. 87–90 pages. doi:10.3233/978-1-61499-649-1-87

[54] Lisa Krizan. 1999. *Intelligence Essentials for Everyone*. Number 6 in Occasional Paper Series. Joint Military Intelligence College.

[55] Hanseung Lee, Jaeyeon Kihm, Jaegul Choo, John Stasko, and Haesun Park. 2012. iVisClustering: An Interactive Visual Document Clustering via Topic Modeling. *Computer Graphics Forum* 31, 3pt3 (2012), 1155–1164. doi:10.1111/j.1467-8659.2012.03108.x

[56] Eric S. Liu, Dylan A. Lukes, and William G. Griswold. 2023. Refactoring in Computational Notebooks. *ACM Trans. Softw. Eng. Methodol.* 32, 3, Article 77 (April 2023), 24 pages. doi:10.1145/3576036

[57] Jiali Liu, Nadia Boukhelifa, and James R. Eagan. 2020. Understanding the Role of Alternatives in Data Analysis Practices. *IEEE Transactions on Visualization and Computer Graphics* 26, 1 (Jan 2020), 66–76. doi:10.1109/TVCG.2019.2934593

[58] John D. Lowrance, Ian W. Harrison, and Andres C. Rodriguez. 2001. Capturing Analytic Thought. In *Proceedings of the 1st International Conference on Knowledge Capture* (Victoria, British Columbia, Canada) *(K-CAP '01)*. Association for Computing Machinery, New York, NY, USA, 84–91. doi:10.1145/500737.500753

[59] Richard Mander, Gitta Salomon, and Yin Yin Wong. 1992. A "Pile" Metaphor for Supporting Casual Organization of Information. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Monterey, California, USA) *(CHI '92)*. ACM, New York, NY, USA, 627–634. doi:10.1145/142750.143055

[60] Andrew Y Ng, Michael I Jordan, Yair Weiss, et al. 2001. On Spectral Clustering: Analysis and an Algorithm. In *Proceedings of the 15th International Conference on Neural Information Processing Systems: Natural and Synthetic* (Vancouver, British Columbia, Canada) *(NIPS'01, Vol. 14)*. MIT Press, Cambridge, MA, USA, 849–856.

[61] lkujiro Nonaka, Hirotaka Takeuchi, and Katsuhiro Umemoto. 1996. A Theory of Organizational Knowledge Creation. *International Journal of Technology Management* 11, 7-8 (1996), 833–845. doi:10.1287/orsc.5.1.14

[62] Sinan Ozdemir. 2016. *Principles of Data Science*. Packt Publishing Ltd.

[63] Fernando Perez and Brian E Granger. 2015. Project Jupyter: Computational narratives as the engine of collaborative data science. *Retrieved September* 11, 207 (2015), 108. https://blog.jupyter.org/project-jupyter-computational-narratives-as-the-engine-of-collaborative-data-science-2b5fb94c3c58.

[64] Jeffrey M Perkel. 2018. Why Jupyter is data scientists' computational notebook of choice. *Nature* 563, 7732 (2018), 145–147. doi:10.1038/d41586-018-07196-1

[65] João Felipe Pimentel, Leonardo Murta, Vanessa Braganholo, and Juliana Freire. 2019. A Large-Scale Study About Quality and Reproducibility of Jupyter Notebooks. In *2019 IEEE/ACM 16th International Conference on Mining Software Repositories (MSR)*. 507–517. doi:10.1109/MSR.2019.00077

[66] Peter Pirolli and Stuart Card. 2005. The sensemaking process and leverage points for analyst technology as identified through cognitive task analysis. In *Proceedings of the International Conference on Intelligence Analysis*, Vol. 5. McLean, VA, USA, 2–4.

[67] Denis Pombriant. 2013. Data, Information and Knowledge: Transformation of data is key. *Computer Law Review International* 14, 4 (2013), 97–102. doi:10.9785/ovs-cri-2013-97

[68] Yan Qu and George W. Furnas. 2008. Model-Driven Formative Evaluation of Exploratory Search: A Study under a Sensemaking Framework. *Information Processing & Management* 44, 2 (2008), 534 – 555. doi:10.1016/j.ipm.2007.09.006 Evaluating Exploratory Search Systems Digital Libraries in the Context of Users' Broader Activities.

[69] Deepthi Raghunandan, Niklas Elmqvist, and Leilani Battle. 2023. Measuring How Data Science Notebooks Evolve Over Time. *Interactions* 30, 1 (Jan. 2023), 17–18. doi:10.1145/3572863

[70] Einar Risvik, Jean A McEwan, Janet S Colwill, Regina Rogers, and David H Lyon. 1994. Projective mapping: A tool for sensory analysis and consumer research. *Food quality and preference* 5, 4 (1994), 263–269. doi:10.1016/0950-3293(94)90051-5

[71] Daniel E. Rose, Richard Mander, Tim Oren, Dulce B. Ponceléon, Gitta Salomon, and Yin Yin Wong. 1993. Content Awareness in a File System Interface: Implementing the "Pile" Metaphor for Organizing Information. In *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Pittsburgh, Pennsylvania, USA) *(SIGIR '93)*. ACM, New York, NY, USA, 260–269. doi:10.1145/160688.160735

[72] Adam Rule, Aurélien Tabard, and James D. Hollan. 2018. Exploration and Explanation in Computational Notebooks. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–12. doi:10.1145/3173574.3173606

[73] David E Rumelhart and Donald A Norman. 1976. *Accretion, Tuning and Restructuring: Three Modes of Learning*. Technical Report 7602. Office of Naval Research, Arlington, VA.

[74] David E Rumelhart and Donald A Norman. 1981. Analogical Processes in Learning. *Cognitive skills and their acquisition* (1981), 335–359.

[75] B. Saket, P. Simonetto, S. Kobourov, and K. Börner. 2014. Node, Node-Link, and Node-Link-Group Diagrams: An Evaluation. *IEEE Transactions on Visualization and Computer Graphics* 20, 12 (Dec 2014), 2231–2240. doi:10.1109/TVCG.2014.2346422

[76] Samuel Sanford Shapiro and Martin B Wilk. 1965. An analysis of variance test for normality (complete samples). *Biometrika* 52, 3-4 (1965), 591–611. doi:10.2307/2333709

[77] Colin Shearer. 2000. The CRISP-DM Model: The New Blueprint for Data Mining. *Journal of Data Warehousing* 5, 4 (2000), 13–22.

[78] Jeremy Singer. 2020. Notes on notebooks: is Jupyter the bringer of jollity?. In *Proceedings of the 2020 ACM SIGPLAN International Symposium on New Ideas, New Paradigms, and Reflections on Programming and Software* (Virtual, USA) *(Onward! 2020)*. Association for Computing Machinery, New York, NY, USA, 180–186. doi:10.1145/3426428.3426924

[79] George F Spiegel Jr and James P Barufaldi. 1994. The Effects of a Combination of Text Structure Awareness and Graphic Postorganizers on Recall and Retention of Science Knowledge. *Journal of Research in Science Teaching* 31, 9 (1994), 913–932. doi:10.1002/tea.3660310907

[80] Roderick Tabalba, Nurit Kirshenbaum, Jesse Harden, Michael Rogers, Arthur Nishimoto, Elizabeth Christman, Andy Yu, Ryan Theriot, Lance Long, Luc Renambot, Mahdi Belcaid, Chris North, Andrew Johnson, and Jason Leigh. 2023. SAGE3 - the Smart Amplified Group Environment. In *Science Gateways 2023 (SG23)*. Zenodo. doi:10.5281/zenodo.10034793

[81] Einblick Team. 2023. Einblick. *https://www.einblick.ai*. https://www.einblick.ai

[82] SAGE3 Team. 2024. SAGE3 - Collaborate Smarter. https://sage3.sagecommons.org/

[83] Sally Thorne. 2000. Data analysis in qualitative research. *Evidence-Based Nursing* 3, 3 (2000), 68–70. arXiv:https://ebn.bmj.com/content/3/3/68.full.pdf doi:10.1136/ebn.3.3.68

[84] James H Wandersee. 1990. Concept Mapping and the Cartography of Cognition. *Journal of Research in Science Teaching* 27, 10 (1990), 923–936. doi:10.1007/978-94-015-9454-7

[85] Weigang Wang and Jörg Haake. 1997. Supporting User-Defined Activity Spaces. In *Proceedings of the Eighth ACM Conference on Hypertext* (Southampton, United Kingdom) *(HYPERTEXT '97)*. Association for Computing Machinery, New York, NY, USA, 112–123. doi:10.1145/267437.267450

[86] Zijie J. Wang, Katie Dai, and W. Keith Edwards. 2022. StickyLand: Breaking the Linear Presentation of Computational Notebooks. In *Extended Abstracts of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI EA '22)*. Association for Computing Machinery, New York, NY, USA,

Article 269, 7 pages. doi:10.1145/3491101.3519653

[87] Nathaniel Weinman, Steven M. Drucker, Titus Barik, and Robert DeLine. 2021. Fork It: Supporting Stateful Alternatives in Computational Notebooks. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 307, 12 pages. doi:10.1145/3411764.3445527

[88] John Wenskovitch, Ian Crandell, Naren Ramakrishnan, Leanna House, Scotland Leman, and Chris North. 2018. Towards a Systematic Combination of Dimension Reduction and Clustering in Visual Analytics. *IEEE Transactions on Visualization and Computer Graphics* 24, 1 (Jan 2018), 131–141. doi:10.1109/TVCG.2017.2745258

[89] John Wenskovitch, Jian Zhao, Scott Carter, Matthew Cooper, and Chris North. 2019. Albireo: An Interactive Tool for Visually Summarizing Computational Notebook Structure. In *2019 IEEE Visualization in Data Science (VDS)*. 1–10. doi:10.1109/VDS48975.2019.8973385

[90] John Wenskovitch, Michelle Zhou, Christopher Collins, Remco Chang, Michelle Dowling, Alex Endert, and Kai Xu. 2020. Putting the "I" in Interaction: Interactive Interfaces Personalized to Individuals. *IEEE Computer Graphics and Applications* 40, 3 (2020), 73–82. doi:10.1109/MCG.2020.2982465

[91] Frank Wilcoxon. 1945. Individual comparisons by ranking methods. *Biometrics bulletin* 1, 6 (1945), 80–83. doi:10.2307/3001968

[92] Margaret Wilson. 2002. Six Views of Embodied Cognition. *Psychonomic bulletin & review* 9, 4 (2002), 625–636. doi:10.3758/BF03196322

[93] B.L. William Wong and Neesha Kodagoda. 2016. How Analysts Think: Anchoring, Laddering and Associations. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* 60, 1 (2016), 178–182. doi:10.1177/1541931213601040

[94] Leanne S Woolhouse and Rowan Bayne. 2000. Personality and the Use of Intuition: Individual Differences in Strategy and Performance on an Implicit Learning Task. *European Journal of Personality* 14, 2 (2000), 157–169. doi:10.1002/(SICI)1099-0984(200003/04)14

[95] Charles R. Wright, Michael R. Manning, Bruce Farmer, and Brad Gilbreath. 2000. Resourceful Sensemaking in Product Development Teams. *Organization Studies* 21, 4 (2000), 807–825. doi:10.1177/0170840600214006

[96] William Wright, David Schroh, Pascale Proulx, Alex Skaburskis, and Brian Cort. 2006. The Sandbox for Analysis: Concepts and Methods. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Montréal, Québec, Canada) *(CHI '06)*. Association for Computing Machinery, New York, NY, USA, 801–810. doi:10.1145/1124772.1124890

[97] Hongyuan Zha, Xiaofeng He, Chris Ding, Ming Gu, and Horst D Simon. 2002. Spectral Relaxation for k-Means Clustering. In *Proceedings of the 15th International Conference on Neural Information Processing Systems: Natural and Synthetic* (Vancouver, British Columbia, Canada) *(NIPS'01)*. MIT Press, Cambridge, MA, USA, 1057–1064.

[98] Pengyi Zhang. 2010. *Sensemaking: Conceptual Changes, Cognitive Mechanisms, and Structural Representations. A Qualitative User Study*. PhD Thesis. University of Maryland, College Park, MD. http://hdl.handle.net/1903/10371.

[99] Zheguang Zhao, Lorenzo De Stefani, Emanuel Zgraggen, Carsten Binnig, Eli Upfal, and Tim Kraska. 2017. Controlling False Discoveries During Interactive Data Exploration. In *Proceedings of the 2017 ACM International Conference on Management of Data* (Chicago, Illinois, USA) *(SIGMOD '17)*. Association for Computing Machinery, New York, NY, USA, 527–540. doi:10.1145/3035918.3064019

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009