Resubmission of InfoVis 2019 submission "Extending Spreadsheets to Support Seamless Navigation at Scale" to TVCG

A previous version of this paper was submitted to InfoVis 2019 (submission ID: 1043). This version was rejected as the paper co-chairs deemed that revising the paper would require more time than allocated to meet reviewer expectations. The paper co-chairs mentioned the following: "*We particularly encourage such revisions where submissions were positively received by reviewers, but the revisions required were deemed to be beyond the scope of the conference review cycle. If you address the issues raised and subsequently submit to TVCG, please make reference to this InfoVis submission and include a description of how you addressed the InfoVis reviewers' comments.*" The entire set of reviews and the old INFOVIS submission are attached at the end of this document. The reviewers asked for the following changes for a revision:

C1: More clearly articulate the merits over alternative tools, such as Keshif.
C2: Convincingly explain the design choice around the binning mechanism or revise that approach.
C3: Justify the choice of comparing with Excel and note the limitations of that approach.
C4: Better justify the choice of tasks in the study.
C5: Explain the choice of the four research questions.
C6: Provide more details about the study, including reporting intra-participant differences.

We thank the reviewers and meta-reviewer for their detailed, constructive feedback. Taking this feedback into account, we have spent the last four months preparing a revised version of this paper. We believe the paper is substantially stronger as a result. Every section in the paper has been revised. We have attached the new version of the paper with this "summary of changes" document, where the changes are highlighted in blue for reviewers' convenience. We briefly describe the changes to each section before returning to the changes required in the revision (**C1**–**C6**).

Changes Organized by Section

- In Section 1, we now cite work that highlights the widespread adoption of spreadsheets even among users of advanced enterprise solutions, further justifying why we focused on addressing navigation challenges within spreadsheets. In fact, recent debate within the visualization community following VIS 2019 also echoes this view[1]—instead of designing a new sophisticated tool from scratch that may cater to a small population, we aim to enhance the user experience for existing spreadsheet tools with nearly a billion users. We further augment the definitions of spreadsheet interactions like scrolling and steering in Section 1, and now clearly articulate the challenges of designing a general-purpose plug-in for spreadsheets.
- In Section 2, our usage scenario has been substantially revised using Brehmer and Munzner's typology [1] to clarify the scope of tasks supported by spreadsheets, as well as those that are enhanced when using a spreadsheet with a NOAH plug-in; our new Table 1 provides use-cases supported within NOAH for each category.
- In Section 3, we now clearly articulate the differences in goals of spreadsheets and tabular data analysis tools (TDA), further justifying our choice of Excel, the most widely used spreadsheet tool, as our baseline in the evaluation study.
- In Section 4, we now explain the third design consideration for NOAH more clearly (**DC3**)—focusing on motivating the binning mechanism.
- In Section 5, we now justify why we are developing a multi-granularity overview, and explain why we opted for histograms as an overview representation, by contextualizing our approach using prior work on multi-scale aggregation [2].
- In Section 6, we further clarified the goals of our study and reframed our research questions to better reflect those goals. Our research questions haven't been substantially altered; rather, they have been grouped together and made more precise. In this version, we now explain our choice for the quiz tasks in detail.
- Section 7 has been completely revamped to better reflect our research questions, while adding more qualitative observations regarding the user experience with NOAH. We now include additional analysis results on intra-participant differences.
- In Section 8, we have added a summary of our takeaways from the evaluation study, while discussing limitations of our evaluation study and NOAH's design while highlighting future enhancement opportunities.

Changes Organized by Reviewer Concerns (C1–C6)

1) *Merits over other tools* (**C1**): Throughout the paper (and especially in Sections 1, 3, and 6), we have emphasized that the main contribution of NOAH is its *design as a general-purpose navigation plug-in* to any existing spreadsheet tool. As spreadsheets have a massive user-base, enhancing exploration and formula computation on large datasets while maintaining the spreadsheet look-and-feel as much as possible, has the potential to impact hundreds of millions of spreadsheet users. Most of these users employ spreadsheets as their primary data management and analysis tool while shunning enterprise solutions with more advanced features. Therefore, our goal is to develop a solution to improve navigation within spreadsheets. Indeed, we could have tried to enhance navigation in other tools, such as Keshif, or Tableau, like the reviewers suggest—but we would be forcing spreadsheet users to adopt an entirely new tool, something that they are clearly loath to do.
2) *Explanation of the binning mechanism* (**C2**): We now explain our choice of the binning mechanism in the context of the third design consideration (*DC3*), as detailed in Section 4. We further expand on this in the blurb titled "Why a Multi-granularity Binned Overview" in Section 5. In brief, we opted for binning to provide a clear and concise representation of the overall data distribution while minimizing user's back and forth movement across multiple screens during navigation.

[1] https://twitter.com/FILWD/status/1187411664611749888

At some level, we do need to limit the number of values displayed so that the overview fits on the screen: binning is a natural solution for that issue. We cite related work on multi-scale aggregation [2] to motivate the choice of a multi-granularity overview and show that binned aggregation via histograms [3] is a suitable representation of such an overview. However, we acknowledge limitations for this choice for binning categorical data and discuss possible solutions in Section 8 in the context of our user study findings.

3) *Justification of Excel as a Baseline* (**C3**): Since our goal is to improve spreadsheet user experience—and there are very good reasons for doing so, as outlined in item 1) above—the natural comparison point is Excel. That said, beyond popularity, we have provided a thorough justification for what spreadsheet tools like Excel offer relative to tabular data analysis tools (TDA), making it a more appropriate point of comparison. Our justification for using Excel over tabular data analysis (TDA) tools involves a) highlighting the appeal of spreadsheets among users who shun more advanced enterprise tools (Section 1), b) identifying the scope of tasks supported by spreadsheets and TDA tools (Section 2 and 3), c) explaining that the goal of spreadsheets is to present the raw data as is, amenable to editing, formula computation, and comparison of derived data, all done in-situ, unlike TDA tools that hide the data while providing summarized statistics (Section 3), and d) highlighting the importance of addressing the shortcomings of a tool, the user base of which far exceeds that of TDA tools (Section 6).

4) *Enhancing the user study design* (**C4, C5**): We have added additional explanations regarding the goals of the study while better articulating the research questions, provided a better justification for the choice of the tasks for the study (see Section 6.1) and added a detailed discussion on the limitations of the study. Specifically, we merged our previous *RQ1* (evaluating the quiz task performance) with *RQ4* (summarizing user feedback on performance via a survey) into a unified research question (new *RQ1*) on the participants' navigation performance. Moreover, we merged previous *RQ2* and *RQ3* that explored how specific features of NOAH affected participants' navigation experience into one broad question (new *RQ2*) that explores their overall experience with NOAH and its components. Moreover, we have justified the choice of the quiz tasks by a) relating their goals with the scope of tasks supported by spreadsheets (see Section 2), and b) explaining how these tasks can reveal how people may use NOAH for spreadsheet navigation. Finally, we have identified three limitations of the study, resulting from the sample size and the study design, which we justify based on the scope of the study.

5) *Reporting Intra-participant Differences* (**C6**): We performed additional analysis of the study data to report intra-participant submission time differences for the quiz phase tasks. While we summarize the results of the analysis in Section 7.1.1, the detailed results can be found in Appendix. We find that despite a few exceptions, across all tasks, participants' task submission times were faster using NOAH—all but one participants completed at least four tasks in less time using NOAH compared to Excel. However, we already obtained similar observations from Figure 5 in Section 7.1. Therefore, we decided to include the results of the intra-participant analysis in the Appendix as opposed to the main paper.

We realize that these extensive changes throughout the paper have increased the paper length. As the TVCG submission guidelines indicate "*The regular paper page length limit is defined at 12 formatted pages, including references and author biographies. Any pages or fraction thereof exceeding this limit are charged $220 per page. Regular papers may not exceed 18 formatted pages*", so we have decided to let the paper length stand since our paper is at 16 pages currently and we would be happy to pay for the extra pages. We would also be happy to reduce length by moving appropriate content from the main body to the supplementary material as suggested by reviewers. The following are attached with the "Summary of changes" document:

- The resubmitted TVCG paper with changes highlighted in blue.
- The entire set of reviews of the INFOVIS submission.
- The previous INFOVIS submission.

Extending Spreadsheets to Support Seamless Navigation at Scale

Sajjadur Rahman, Mangesh Bendre, Yuyang Liu, Shichu Zhu, Zhaoyuan Su,
Karrie Karahalios, Aditya Parameswaran

Abstract—Spreadsheets are one of the most popular tools for ad-hoc exploration and analysis of data. Despite that, exploring and analyzing spreadsheet datasets that span more than a few screens via operations such as scrolling or issuing formulae, is often overwhelming for end-users. Users easily lose context as they explore the data via scrolling and suffer from cognitive and mechanical burdens while issuing formulae on data spanning multiple screens. We propose integrating a navigation plug-in with spreadsheets to support the seamless exploration of large datasets that are increasingly the norm. Our interface, NOAH, developed using lessons from classical overview+detail interfaces, embeds a multi-granularity zoomable overview alongside the spreadsheet. Users can employ the overview to explore the data at various granularities. Furthermore, they can issue formulae over subsets of data without performing cumbersome scrolling or range selection operations, enabling users to gain a high or low-level perspective of the spreadsheet data. NOAH preserves spreadsheet semantics and look and feel, while introducing such enhancements. Our user study demonstrates that NOAH makes it more intuitive, easier, and faster to navigate spreadsheet data compared to traditional spreadsheets like Microsoft Excel, for a variety of navigational tasks; participants made $2.5\times$ fewer mistakes in NOAH than in Excel while being twice as fast in completing the tasks.

Index Terms—Spreadsheet navigation, data exploration, overview+detail, zooming.

1 Introduction

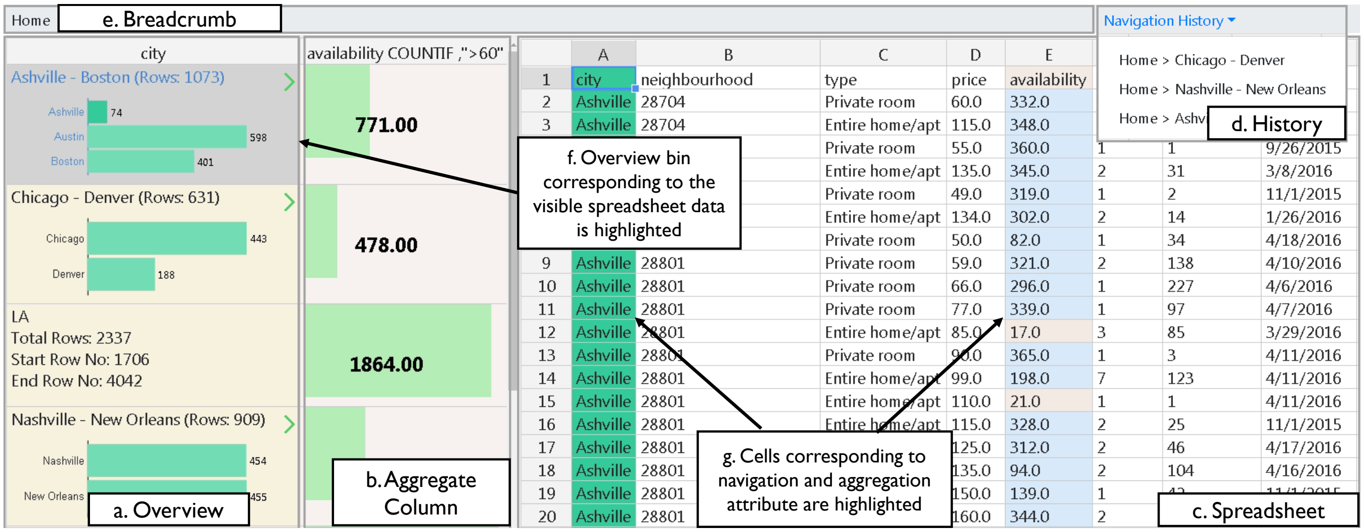
With a user base of more than one-tenth of the world's population, ***spreadsheets are by far the most popular medium for ad-hoc exploration and analysis of data*** [4]. Studies show that information workers prefer to operate on their data within spreadsheets while shunning enterprise solutions with more advanced analytical features [5], [6]. One popular joke among those developing business intelligence applications is that the "export to excel" button is the third-most commonly used button from the menu bar, after OK and Cancel [7]. Spreadsheets enable users to view, structure, and present data in an intuitive tabular layout, wherein users can map their data and tasks; this tabular layout is essential to the popularity of spreadsheets [8].

Using this tabular layout effectively involves navigation, *i.e.*, *"the process of viewing and manipulating the computer display to show another portion of the information space"* [9]. Navigation is supported via two unit operations, scrolling and steering. Scrolling is the action of moving displayed text or graphics up, down, or across a computer screen, in order to view different parts of the spreadsheet. For example, when analyzing data, users may scroll to compare data across different screens, or to get a high-level view of the overall spreadsheet. Steering, on the other hand, involves clicking the left mouse button and then dragging the mouse pointer through the spreadsheet to select a specific region. For example, to issue a formula, users may steer to select the subset of the data to be operated on as an argument within the formula. Most frequently used spreadsheet formulae require users to perform steering actions [10], [11]. Overall, both scrolling and steering are crucial as users navigate spreadsheets to identify, compare, and summarize data.

However, navigating spreadsheets using scrolling or steering is challenging, since spreadsheet data span multiple screens, making it hard to synthesize, analyze, makes sense of, or operate on it [8], [12]. With the ease of data generation, and with spreadsheets now supporting increasingly larger datasets, *e.g.*, Google Sheets now supports five million cells [13], a 12.5× increase from the previous limit of 400K cells, *navigating data within spreadsheets is only becoming even harder*, thanks to multiple inter-related reasons:

- *Loss of overview and context.* When navigating spreadsheets, users can easily lose the context of where they are and where they should go next [12]. The only navigational context provided by spreadsheets is the built-in scrollbar that acts as a one-dimensional overview and indicates the user's current location on the sheet. However, since this overview does not capture the layout and structure of the data, users are forced to mentally assimilate the layout and recall it on-demand, as they navigate a spreadsheet.
- *Cognitive and mechanical burdens.* The lack of contextual cues leads to severe cognitive and mechanical burdens for users [14]. Users often end up taking their own drastic measures to avoid getting lost; for example, some users create personalized overviews extrinsic to

- *Sajjadur Rahman, Yuyang Liu, Shichu Zhu and Karrie Karahalios are with the Department of Computer Science at University of Illinois, Urbana-Champaign. Email: {srahman7, yuyangl2, szhu28, kkarahal}@illinois.edu.*
- *Mangesh Bendre is with VISA Research. Email: mbendre@visa.com.*
- *Zhaoyuan Su with the School of Information and Computer Sciences at the University of California Irvine. Email: nick.su@uci.edu.*
- *Aditya Parameswaran is with the I School and EECS at the University of California, Berkeley. Email: adityagp@berkeley.edu.*


Home
e. Breadcrumb
Navigation History ▾
Home > Chicago - Denver
Home > Nashville - New Orleans
Home > Ashv
d. History
city
Ashville - Boston (Rows: 1073)
Ashville
74
Austin
598
Boston
401
Chicago - Denver (Rows: 631)
Chicago
443
Denver
188
LA
Total Rows: 2337
Start Row No: 1706
End Row No: 4042
Nashville - New Orleans (Rows: 909)
Nashville
454
New Orleans
455
a. Overview
availability COUNTIF ,">60"
771.00
478.00
1864.00
b. Aggregate Column
f. Overview bin corresponding to the visible spreadsheet data is highlighted
g. Cells corresponding to navigation and aggregation attribute are highlighted
c. Spreadsheet

	A	B	C	D	E			
1	city	neighbourhood	type	price	availability			
2	Ashville	28704	Private room	60.0	332.0			
3	Ashville	28704	Entire home/apt	115.0	348.0			
			Private room	55.0	360.0	1	1	9/26/2015
			Entire home/apt	135.0	345.0	2	31	3/8/2016
			Private room	49.0	319.0	1	2	11/1/2015
			Entire home/apt	134.0	302.0	2	14	1/26/2016
			Private room	50.0	82.0	1	34	4/18/2016
9	Ashville	28801	Private room	59.0	321.0	2	138	4/10/2016
10	Ashville	28801	Private room	66.0	296.0	1	227	4/6/2016
11	Ashville	28801	Private room	77.0	339.0	1	97	4/7/2016
12	Ashville	28801	Entire home/apt	85.0	17.0	3	85	3/29/2016
13	Ashville	28801	Private room	90.0	365.0	1	3	4/11/2016
14	Ashville	28801	Entire home/apt	99.0	198.0	7	123	4/11/2016
15	Ashville	28801	Entire home/apt	110.0	21.0	1	1	4/11/2016
16	Ashville	28801	Entire home/apt	115.0	328.0	2	25	11/1/2015
17	Ashville	28801		125.0	312.0	2	46	4/17/2016
18	Ashville	28801		135.0	94.0	2	104	4/16/2016
19	Ashville	28801		150.0	139.0	1		
20	Ashville	28801		160.0	344.0	2		

Fig. 1. NOAH: navigation interface consisting of (a) a zoomable overview and (b) an aggregate column integrated with (c) a spreadsheet. A context bar consisting of (d) a navigation history displaying locations visited so far using the overview, and (e) a breadcrumb showing the current navigation path (*e.g.*, Home). (f) The users current focus in the spreadsheet is highlighted on the overview. (g) Columns corresponding to the navigation attribute (city) and aggregate column (availability) are highlighted on the spreadsheet.

the spreadsheet, by sketching maps of spreadsheets on paper [12]. Other users add their own landmarks such as headers or colored cells, as a visual affordance to assist in navigation [12]. Steering via dragging the mouse pointer across multiple screens to select a subset of data as input to a formula can often be challenging as well: the only remedy is for users to abandon steering entirely and instead remember the range of the subset of data of interest, and then correctly enter this range as the argument to the formula, often giving rise to errors that are increasingly prevalent in spreadsheets [15].

- *Visual discontinuities.* The limited viewport afforded to the user introduces a visual discontinuity between the information being displayed. For example, comparing spatially separated subsets of data within the spreadsheet requires moving back and forth between multiple viewports, which can be overwhelming [8], [12]. As an alternative, users tend to copy subsets of data side by side to reduce the visual discontinuity [8], [12], which is cumbersome.

Overall, while navigating present-day spreadsheets, ***users often lose context, get overwhelmed, and experience visual discontinuities.*** Addressing these challenges requires considerable manual effort. As we will argue in Section 3, existing spreadsheet features such as pivot tables, named ranges, and subtotals, partially alleviate some of the aforementioned challenges but do not eliminate them entirely. For example, pivot tables generate a summary while losing the correspondence between the raw data and the summary, while named ranges require users to manually associate names with ranges of data.

So, how do we support more effective navigation of data within spreadsheets? One approach would be to try to integrate an overview of the overall structure of the data along with the spreadsheet [16] resulting in a classical overview+detail interface where the spreadsheet is the detailed view. Overview+detail interfaces are used to facilitate navigation in various domains such as text editors and maps [14]. Users can manipulate the overview or detailed view, to perform high-level or low-level operations, respectively. Overview+detail interfaces have been shown to be effective in these domains, reducing cognitive load for users by providing them the big picture first, helping them quickly assimilate the information space [14]. Our goal is to *integrate an overview plug-in with spreadsheets* that captures the overall structure of the data, while supporting interactions that address the difficulties in typical navigational operations like scrolling and steering. ***It is essential that our interface is a plug-in that enhances the capabilities of spreadsheets that so many users are used to and reliant on, as opposed to a potentially jarring or confusing replacement for spreadsheets.***

However, while an overview plug-in for spreadsheets does seem appealing and natural, developing it leads to several challenges.

- *Overview modality.* One could simply add a zoomed out version of the entire sheet as a pane on the side to create a spreadsheet overview, as in popular presentation software like Microsoft PowerPoint, or text editors like Sublime Text. The zoomed out overview would display the data at a lower magnification. Unfortunately, this approach would not suffice for a spreadsheet. An overview should provide a comprehensible big picture view; for a spreadsheet of numbers, text, or formulae, when zooming out beyond a point, an overview displayed at such low magnification would be unreadable. Another approach, adopted by map tools like the early versions of Google maps, is to use the overview to provide a global context of the user's current location currently displayed in the zoomed-in detailed view [14]. While the overview remains static, users can perform semantic zooming operations [17] on the detailed view which allows objects to be represented differently at different scales. Since spreadsheets already display the raw data, zooming into and out of a detailed view consisting of this raw data is not meaningful. How do we design an overview to dynamically change as users seek a more fine-grained or coarse-grained view of the overall structure of the data?

- *Construction of the overview.* Given a spreadsheet with many rows, one approach to constructing a dynamic overview is mapping rows of data to high level groups, similar to online maps. In online maps, cities are grouped into states and states are grouped into countries, forming a multi-granularity hierarchy. How do we automatically group spreadsheet rows together in a similar "meaningful" way such that this grouping applies to all data types, including strings and numbers? If the automatically generated grouping is not semantically meaningful, how do we allow the users to customize the grouping modality? How do we facilitate interactions that enable users to view the overview at multiple granularities?
- *Operations on the overview.* Following the construction of a dynamic overview, the next challenge is to design simple interactions that achieve similar outcomes as scrolling and steering. For example, an alternative to scrolling can be to leverage the groups of the overview to access the rows mapped to that group. As the granularity of the dynamic overview changes, how do we efficiently update the mapping from spreadsheet rows to the finer or coarser groups so that scrolling remains seamless? Similarly, how do we leverage the overview to steer spreadsheet data (i.e., select a range of data) for formula computation? How do we present the results of the formula within the overview? One approach can be to adopt the pivot table-like presentation of results. Within their summary view, pivot tables display aggregate formula results (*e.g.*, `SUM`, `COUNT`) alongside each group. However, unlike pivot tables, users can explore the overview at multiple granularities. As the granularity changes, the grouping of rows also changes; making the previous formula results inconsistent with the new groups. How do we recompute the results of a formula in a convenient manner as the granularity changes without requiring the users to reissue the formula from scratch?
- *Seamless integration as a plug-in.* Finally, how can we design an overview with a generic set of features, that can be integrated with any existing spreadsheet tool, operating on any dataset, without impacting existing functionalities or look-and-feel? How do we ensure that the interactions supported by the overview are consistent with traditional spreadsheet semantics, and complement existing spreadsheet interactions? How do we enable coordinated interactions across both views, *i.e.*, the overview and the raw spreadsheet, such that they remain consistent at any given time?

NOAH: a navigation plug-in for spreadsheets. We address the aforementioned challenges in NOAH[1], an in-situ **n**avigation interface for **o**verviewing and **a**nalyzing spreadsheet data **h**olistically. NOAH is constructed as a plugin to an existing spreadsheet tool, DATASPREAD [18], an open-source scalable web-based spreadsheet. While NOAH's design is not tied to DATASPREAD, we opted not to use other popular spreadsheet tools like Google Sheets and Microsoft Excel because they are closed source. Figure 1 shows a snapshot of NOAH. When the user chooses to explore the data by a specific attribute, a multi-granularity overview is constructed and displayed within NOAH, next to the raw spreadsheet data (Figure 1a). Users can zoom into or out of the overview to obtain a fine or coarse-grained perspective of the data distribution. The distribution at each granularity is captured by a histogram, enabling users to assimilate the data via summary statistics. Each bin (group) of the histogram is mapped to a collection of rows in the spreadsheet. Cumbersome scrolling operations are eliminated in favor of a few clicks on the overview interface. Instead of steering to analyze the data, users can issue formulae on the overview with interactions similar to pivot table construction, and view results on a separate *aggregate column*, alongside the overview (Figure 1b). In this manner, users can issue formulae on different subsets of the data while remaining on the same screen, reducing visual discontinuity. NOAH ensures that there is coordination between the overview and the spreadsheet: for example, panning and zooming on the overview are reflected on the spreadsheet by displaying the spreadsheet data corresponding to the bin currently in focus in the overview. Finally, NOAH automatically creates contextual and historical information (Figure 1d and 1e) while displaying visual cues (Figure 1f and 1g) so that users don't lose context during navigation.

The primary contribution of our work is twofold:

- We formalize the design of a general navigation (overview+detail) interface for exploration and analysis of large spreadsheets. We realize this design in the form of NOAH, a plugin to a spreadsheet tool, ensuring that interactions supported by NOAH complement existing spreadsheet operations.
- We conduct a user study to evaluate the benefits and limitations of this plugin. The study required users to perform tasks that were representative of popular spreadsheet operations, using both Microsoft Excel and NOAH. The study shows that compared to Microsoft Excel, participants were able to complete spreadsheet navigation tasks correctly and quickly in NOAH. Participants made 2.5× **fewer** mistakes while being 2× **faster** with a NOAH-integrated spreadsheet than with Excel.

2 NOAH USE CASES

Users prefer spreadsheets over enterprise solutions to view, explore, and analyze data [5], [6]. To understand the scope of typical user tasks on spreadsheets, we make use of the typology of abstract data exploration tasks [1]—see Table 1. This typology characterizes the range of domain-independent tasks performed on visual representations of data, developed after analyzing task classification systems in over two dozen papers, and has been applied to a variety of scenarios, including developing models for visualization system design [19], designing task taxonomies

[1] A video demo of NOAH can be found at https://www.youtube.com/watch?v=iZsboe3x680

for cartograms [20], and defining the scope of tasks in a number of domains, *e.g.*, interactive task authoring [21], document mining [22], multivariate network analysis [23], mass cytometry [24], among others. While all the tasks in Table 1 can be performed using spreadsheets, NOAH enhances the experience for many of these tasks, indicated by a checkmark (✓). We describe these tasks in the context of a real usage scenario for NOAH below.

TABLE 1
Example use cases where NOAH provides benefits beyond spreadsheets (labeled by ✓if improved; × if it remains the same), employing Brehmer and Munzner's typology [1].

Purpose	Use Cases
Consume	`discover` (✓: *generation of hypotheses*, e.g., Rebecca finds a trend in larger cities and wants to check if it is present in smaller cities), `present` (✓: *communication of information*, e.g., Rebecca sees the overall availability trends in the context of raw listings, and can present this view to her co-workers), `enjoy` (✓: *casual encounters with visualization*, e.g., Rebecca uses the overview "at a glance" to understand which cities are present in the dataset, and how many listings are present per city)
Search	`explore/browse` (✓: *searching based on characteristics where location is unknown/known*, e.g., Rebecca tries to find Chicago listings with availability greater than 60 days), `locate/lookup` (✓: *searching based on entities where location is unknown/known*, e.g., Rebecca wants to find all entries corresponding to a given city like Chicago)
Query	`identify` (✓: *returning the characteristics of entity found during search*, e.g., Rebecca wants to examine Chicago listings to assess typical availabilities of listings in Chicago), `compare` (✓: *returning characteristics of multiple entities*, e.g., Rebecca wants to compare listing patterns in Boston to that of Chicago), `summarize` (✓: *returning characteristics of several entities*, e.g., Rebecca wants to gain an understanding of overall rental patterns across cities)
Produce	`export/save` (×), `generate/record` (✓: *generation or recording of new information*, e.g., Rebecca issues an aggregate formula to generate summary availability statistics across cities)

We now describe a usage scenario that illustrates the benefits of integrating NOAH into typical spreadsheets. Let's assume that Rebecca, a journalist, is exploring the *Inside Airbnb* dataset [25], a dataset of all the Airbnb listings across different US cities. This dataset was created to investigate the long-standing accusation that many listings in Airbnb are illegally run as hotel businesses, while avoiding taxes; any listing available for rent for more than 60 days a year is considered to be operated as a hotel [26].

Given that this is the first time she's examining this dataset, Rebecca wants to first gain a bird's eye view of the data. Without NOAH, Rebecca would have had to use a pivot table (discussed in Section 3) to construct a summary—however, since this summary is disconnected from the underlying data, it is hard for Rebecca to map the summary statistics to the raw data to obtain further details about listings from any given city. If she wanted to examine listings from a specific city, Rebecca would have to either use search capabilities or perform an explicit filter for this information, and would have to switch back and forth between the pivot table results and the raw listings, present at disparate locations on the spreadsheet. Even at the first step of exploration, Rebecca would experience *substantial cognitive burdens, loss of context, and visual discontinuities*, with subsequent steps becoming progressively more challenging.

Using NOAH, she organizes the overview by city and starts casually exploring the dataset, understanding which cities are present, and roughly how many listings does each city have—with NOAH providing a high-level overview of cities (Figure 1a) (`enjoy`). The overview consists of sorted non-overlapping bins containing one or more cities. She can click on any bin and the corresponding data will be displayed at the top of her screen. For example, clicking on the *Ashville-Boston* bin displays the Ashville listings (Figure 1c); she can similarly find and examine properties of the Chicago listings by clicking on the *Chicago-Denver* bin (`locate` followed by `identify`). She can also zoom into bins using the "⟩" arrows, zoom out of bins using the "⟨" arrows, and pan by clicking on various bins at the same level. We discuss the construction of the overview and associated interactions in Section 5.

Next, say Rebecca wants to analyze one of the larger cities to understand the overall renting pattern (`summarize`). She studies a few cities at a time, examining and comparing the number of listings for each city, as displayed on the overview (`compare`). She decides to focus on Boston, her hometown, and wants to find out how many listings in Boston violate the "rent availability > 60 days" condition (`identify`). In a typical spreadsheet, Rebecca needs to manually steer and then select the Boston listings as input to a `COUNTIF` formula that counts the number of rows that satisfy the above mentioned condition. Using NOAH, she can zoom into the *Ashville-Boston* bin (Figure 2a and 2b) and then issues a `COUNTIF` operation on the overview (`generate`). The result is displayed as an *aggregate column* alongside the overview (Figure 1b). Rebecca learns that more than half of the listings in Boston are effectively operating as hotels (`discover`)—a large number!

Based on this insight, Rebecca then wants to understand availability statistics for an even larger city, Chicago (`compare`). As she uses the overview to navigate to Chicago, NOAH automatically updates the aggregate column to the `COUNTIF` formula results for Chicago (`identify`), without Rebecca needing to reissue it by performing a cumbersome steering operation as in traditional spreadsheets. Rebecca learns that Chicago exhibits a similar renting pattern as Boston, with more than half the listings operating as hotels. She can then hypothesizes that this trend may hold for all large cities, and can check whether the smaller cities have a different pattern (`discover`). Note that, the rows that satisfy the "rent availability > 60 days" condition, are listed in the spreadsheet adjacent to the overview in sky blue (Figure 1g) (`explore`). With the raw data presented side-by-side, she can also dive into other attributes of the listings operating as hotels to see if there are any other identifying characteristics, *e.g.*, if they are all managed by a small number of agencies acting as individual renters (`identify`).

Finally, as Rebecca navigates the data, her navigation history (Figure 1d), *i.e.*, recently visited cities, and current navigation path (Figure 1e) are kept up-to-date, allowing her to maintain context during navigation (`record`). She can revisit any previously visited cities (`lookup`) by simply clicking on the relevant path in the navigation history.

Overall, with NOAH, users can quickly comprehend the data via the overview, access any region within the data without having to scroll endlessly, and request additional details on demand without having to steer across multiple screens. As users navigate and analyze the data, they can revisit previously accessed data via the navigation history, not losing context of what they have explored.

3 Related Work

We now discuss tools and techniques that partially address the limitations of navigating data, both inside and outside spreadsheets.

3.1 Spreadsheet Tools and Prototypes

Both commercial spreadsheet tools as well as academic prototypes provide partial solutions to navigational challenges.

Microsoft Excel. Excel enables users to manually create references to a spreadsheet region using the named ranges [27] feature, accessible from the menu bar. Users can click on a named range to navigate to the referred region. However, the onus is on the user to create named ranges for each region of interest. The pivot table [28] feature allows users to create a summary view to compare subsets of data without having to provides a summary view, enabling users to compare subsets of data without having to navigate to various locations within the sheet. This summary is placed in a separate region of the spreadsheet, preventing users from accessing the data underlying the summary, impeding navigation. A similar overview feature, SUBTOTAL [29], adds a new row at the end of each distinct subset of data with summary information. Users can expand the summary to view the actual spreadsheet data. However, for datasets with many subsets (*e.g.*, for numeric data), the number of new lines inserted (i.e., the summary) can itself become very large, spanning multiple screens, and can cause increased visual discontinuity during navigation. Finally, NodeXL [30] is a plug-in that provides a spreadsheet network overview and supports navigational operations, *e.g.*, zooming in/out, dynamic filtering, on the overview; this plug-in only supports network datasets, such as biological or social networks.

Google Sheets Explore. Google Sheets Explore [31] provides an overview of the data by auto-generating charts of data statistics. Users can specify queries to the tool (similar to a web search) asking for different summary statistics. While Explore is a convenient means to understand high-level data characteristics, it doesn't address the navigational challenges related to scrolling and steering.

Scalable Spreadsheet Summarization and Exploration. Smart-drill-down [32] generates an interactive summary of a large spreadsheet table as a collection of rules; users can drill-down to a specific rule to view more fine-grained rules. Hillview [33] displays the approximate results of group-by queries on large spreadsheet tables. While these tools support summarization at scale, providing an overview of the spreadsheet, they don't preserve spreadsheet semantics, nor do they make it easy to scroll or steer through large spreadsheets. ABC [34] and DataSpread [35] support interactive exploration of very large spreadsheet datasets, beyond main-memory limits, maintaining spreadsheet look-and-feel, but do not provide any new spreadsheet capabilities to assist with navigation. We build NOAH as a plugin to DataSpread, since it is open-source.

Interactive Tables. TableLens [36] is a focus+context view for browsing numerical information in tables, looking much like a spreadsheet with embedded bar charts. Cells out of focus display graphical bars proportional in length to the underlying values, providing a visual overview of the data, while cells within the user's current focus are magnified and display the graphical bars and the raw data. Ideas similar to TableLens have been adopted by DataLens [37] for visualizing digital calendars, and by FOCUS [38] and InfoZoom [39] for exploring database query results. Like TableLens, NOAH embeds graphical bars, but within the overview to depict the underlying data distribution. NOAH captures the user's current focus by highlighting the corresponding bin in the overview. While TableLens provides an easy mechanism to get a high-level view of the data and spot outliers, it suffers from the same disadvantages that focus+context views have relative to overview+detail ones. Unlike NOAH, which supports multiple granularities via binning, TableLens only supports one granularity (zoomed in or zoomed out): beyond a certain size, navigating (scrolling or steering) the zoomed out data is still cumbersome for users. Moreover, TableLens does not maintain the spreadsheet look-and-feel or capabilities.

Visual Interactive Spreadsheets. VisSh [40], SI [41], SSR [42], ASP [43], and PhotoSpread [44] extend the input/output capabilities of cells within spreadsheets, to display charts, animation, photos, or geometric objects, or accept input via direct manipulation dialogs, among others. While these tools allow users to represent and manipulate data in a more flexible manner, which in turn could help users getting a high-level sense of the data, they do not necessarily help users navigate data more effectively.

3.2 Spreadsheet Alternatives

We draw from work on navigation interfaces in non-spreadsheet interfaces as well.

Overview+Detail Interfaces. Cockburn et al. [14] provides a detailed survey of overview+detail and zooming interfaces. To improve navigation within large documents, overview+detail interfaces [45], [46] allow users to interact with an overview as they explore the document. Zooming interfaces [47], [48] provide a multi-granularity overview of the data and support interactions like zoom in/out to navigate across various granularities. We follow the same analogy of providing an overview of the spreadsheet first, allowing users to drill-down further.

Multiple Coordinated Views. Multiple coordinated views [49], *e.g.*, *Snap* [50], *Elastic Documents* [51] connect multiple views, for example, an overview and a detailed data view while enabling coordination between these views through brushing and linking. Similarly, NOAH connects spreadsheets with an overview and updates the spreadsheet as users interact with the overview and vice-versa.

Tabular Data Analysis (TDA). Visualization tools such as Tableau [52], Power BI [53], Keshif [54], Voyager [55] and analytical tools such as SPSS [56], SAS [57], can all provide summaries of tabular data in various forms (visualizations, aggregate statistics). These summaries are static overviews of the data—much like pivot tables, these summaries are not dynamically linked to nor are co-located with the underlying raw data. For example, Keshif [54] can display all the unique values corresponding to an attribute of interest, *e.g.*, cities of the Airbnb data [25]. However, users cannot view or inspect the raw data corresponding to each city in a spreadsheet-like tabular setting, while being able to edit this raw data at will. With TDA tools, the spreadsheet look-and-feel is lost, and as a result, users lose the ability to directly manipulate raw data, derive new data, and issue formulae for free-form analysis. Therefore, the goals of spreadsheets differ from TDA tools in two ways: a) facilitating direct manipulation of raw data in-situ and b) enabling arbitrary derivation of new data and summaries using various operations involving navigation, *e.g.*, issuing formulae. NOAH being a plug-in to spreadsheets, provides a unified interface that upholds both these goals while enhancing navigational capabilities for spreadsheet users.

4 NOAH: Design Considerations

In this section, we outline our design considerations for a spreadsheet navigation interface. Our design considerations were informed by prior work on information visualization [1], [58], overview+detail interfaces [14], multiple-coordinated views [59], and refined through our experiences across multiple design iterations.

DC1. Construct the overview in-situ. An overview helps users get a high-level picture of the data. However, maintaining the overview in a separate location from the data can lead to loss of context; instead, having it co-located with the data can help users make rapid glances to explore information between a bird's-eye view and a close-up detail [16].

DC2. Ensure reduced visual discontinuity while providing details on demand. Users often need to access subsets of data, and study their properties in detail, *e.g.*, via steering. Navigating back and forth between different subsets of data can lead to increased visual discontinuity. The interface should allow users to compute such details for various data subsets on demand [58]. The interface should maintain visual continuity as users navigate to a different subset, recomputing the details for the new subset.

DC3. Balance the screen space afforded to the overview. As the overview has limited screen-space available, we need to consider the trade-off between visual discontinuity (*DC*2) and clarity. Displaying a fine-grained overview improves visual clarity while increasing visual discontinuity—users need to scroll through the overview to access distant subsets of data. Displaying a coarse-grained overview decreases visual discontinuity at cost of reduced visual clarity—the overview may span too many data subsets and appear visually cluttered. The interface should further allow users to control the screen-space allocated to the overview.

DC4. Enable coordination between the spreadsheet and overview. Since users can view the overview and the spreadsheet simultaneously, interactions on both need to be linked [49], *i.e.*, an interaction on one should be reflected on the other [59]. For example, as a user scrolls through the spreadsheet, the user's current focus should be highlighted on the overview. However, not all interactions need to be interlinked, *e.g.*, changing the font size of a spreadsheet cell need not lead to a change in the overview.

DC5. Facilitate customization of the overview. As the overview is automatically generated, it may not reflect domain-specific context known only to the user [34]. For example, an overview constructed on a grading spreadsheet by binning nearby scores may not match the letter grade ranges that the instructors have in mind. Allowing users to customize the overview is therefore essential.

DC6. Display contextual and historical navigation information. The interface should record navigation history, allowing users to revisit previously visited locations [58], while also displaying their current navigation path for context.

5 User Interface

We now explain the design of NOAH's components and implementation details.

5.1 In-situ Overview

NOAH constructs the overview in-situ (**DC1**) next to the spreadsheet on an attribute of the spreadsheet dataset called the *navigation attribute*, selected by the user. Any attribute type that can be ordered can be a navigation attribute, *e.g.*, text, numbers. The overview is constructed at multiple granularities. Each granularity is divided into non-overlapping groups of data called *bins*. As shown in Figure 2d, an overview of the Airbnb data on the navigation attribute "city" has granularity levels. The highest (coarsest) granularity level consists of four bins. Figure 2a depicts the first four bins, the first of which is *Ashville-Boston*. Each bin contains summary information regarding the data subset/region it spans, *e.g.*, starting row and ending row number, and the total number of rows the region spans. Each bin displays an overview of the next (finer) granularity (if any) with embedded bar charts. For example, in Figure 2d, the topmost bin (*Ash-Bos*) spans three cities (*Ashville*, *Austin*, *Boston*), each of which is a bin in the next (finer) granularity. Correspondingly, Figure 2(a) shows three horizontal bar charts for the first *Ash-Bos* bin, one for each bin in the next granularity. Since the third bin from the top (*LA*) spans only one city, no bar chart is embedded. Users can perform different operations on the bins, *e.g.*, clicking to pan and semantic zooming in/out [17]. NOAH supports other interactions atop this multi-granularity overview, *e.g.*, customization and aggregation. We discuss these interactions in the context of our design considerations in Section 5.1.1.

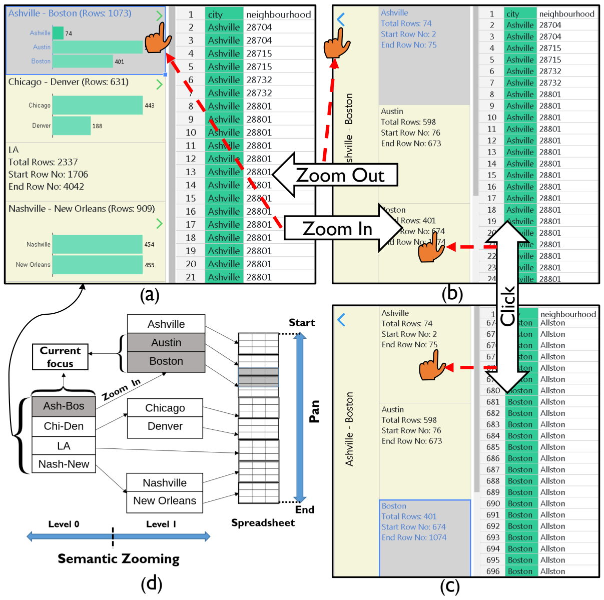
Why a Multi-granularity Binned Overview? A conventional design for overviews within popular interfaces is as a spatially partitioned collection of thumbnails on the left of the standard detailed view, similar to Microsoft Power Point or Adobe Reader. However, displaying too many thumbnails results in increased scrolling to access distant thumbnails, increasing visual discontinuity. On the other hand, displaying too few thumbnails reduces visual discontinuity, but at the cost of visual clarity—the thumbnails appear cluttered and fail to represent the underlying data clearly [14]. To strike a balance between these two objectives (**DC3**) we designed a multi-granularity overview that abstracts the data at varying levels of detail. Multi-granularity representations have been shown to scale better to larger datasets—-presenting information at multiple granularities makes visual representations more perceptually scalable and less cluttered [2]. Thus, the multi-granularity overview of NOAH provides an alternative to the aforementioned conventional spatially partitioned single-granularity representation of the data space, *e.g.*, in Power Point, by allowing users to control the scale at which the overview should be displayed [14]. Users can resize the overview to control the amount of spreadsheet data that remains visible. Users can also hide the overview if required.

The data structure underlying the overview is a histogram constructed on the values in the navigation attribute column. Histograms result from *binned aggregation*—consecutive data points are grouped into bins (or groups), where each bin represents a collection (group) of data points and is associated with a count aggregate, capturing the number of data points that fall in that group. In addition to providing high level (*e.g.*, densities) and low level (e.g., outliers) details, binned aggregation techniques enable a multi-granularity visual representation of data by varying the bin size and have therefore been deployed in interactive visualization of large scale datasets, *e.g.*, in *imMens* [3]. An additional benefit of a binned overview for spreadsheets is a decrease in visual discontinuity during navigation. As users are able to view an overview that fits in the computer screen, they can quickly navigate the data—the bins act as landmarks in the overview, enabling users to skip irrelevant bins and quickly navigate to the desired subset of data. We now discuss how the overview is constructed.

Overview Construction. To generate the overview, we construct an equi-histogram. Equi-depth histograms are commonly used for summarizing statistical properties of data, with applications in database systems for query optimization and approximate query processing [60] and in data mining applications for distribution fitting in data streams [61], among others. The equi-depth histogram can be constructed on any data types that can be ordered, *e.g.*, text, numbers, dates. For example, in the usage scenario explained in Section 2, the journalist grouped the data into cities for ease of navigation when exploring the larger cities in the Airbnb dataset. Each bin in the equi-depth histogram contains the same number of items, where each item is a value. For example, when constructing the overview on city, each value in the city column is assigned to a bin. The bins are constructed top-down (see Figure 2d). NOAH divides each of the bins at level k into new bins to construct the next lower level $k+1$, again, by applying the same concept of equi-depth histograms. If each value of the navigation attribute column was unique, *e.g.*, if it was a numerical ID, then construction of the histogram would be easy: each bin of the equi-depth histogram would contain almost the same number of items, where each item corresponds to one unique value of the attribute. Unfortunately, in practice, for many attributes, the same value is often repeated. For example, there are multiple listings per city. Therefore, an equi-depth histogram on the attribute city will result in consecutive bins sharing items of the same unique city value, resulting in undesirable overlap. Instead, we construct a best effort equi-depth histogram that is as close to an equi-depth histogram as possible, while ensuring that the ranges represented by each bin have no overlap.

5.1.1 Operations and Interactions

We now discuss the operations and interactions that can be performed on the overview.



1	city	neighbourhood
2	Ashville	28704
3	Ashville	28704
4	Ashville	28715
5	Ashville	28715
6	Ashville	28732
7	Ashville	28732
8	Ashville	28801
9	Ashville	28801
10	Ashville	28801
11	Ashville	28801
12	Ashville	28801
13	Ashville	28801
14	Ashville	28801
15	Ashville	28801
16	Ashville	28801
17	Ashville	28801
18	Ashville	28801
19	Ashville	28801
20	Ashville	28801
21	Ashville	28801

1	city	neighbourhood
2	Ashville	28704
3	Ashville	28704
4	Ashville	28715
5	Ashville	28715
6	Ashville	28732
7	Ashville	28732
8	Ashville	28801
9	Ashville	28801
10	Ashville	28801
11	Ashville	28801
12	Ashville	28801
13	Ashville	28801
14	Ashville	28801
15	Ashville	28801
16	Ashville	28801
17	Ashville	28801
18	Ashville	28801
19	Ashville	28801
20	ville	28801
21	ville	28801
22	ville	28801
23	ville	28801
24	ville	28801

1		neighbourhood
674	ston	Allston
67	ston	Allston
67	ston	Allston
67	ston	Allston
67	ston	Allston
67	ton	Allston
680	oston	Allston
681	Boston	Allston
682	Boston	Allston
683	Boston	Allston
684	Boston	Allston
685	Boston	Allston
686	Boston	Allston
687	Boston	Allston
688	Boston	Allston
689	Boston	Allston
690	Boston	Allston
691	Boston	Allston
692	Boston	Allston
693	Boston	Allston
694	Boston	Allston
695	Boston	Allston
696	Boston	Allston

Fig. 2. Navigational operations. (a) The overview at the highest level of granularity. (b) A zoomed in view of the *Ashville-Boston* bin. (c) As the user clicks on the *Boston* bin, the Boston listings are displayed on the sheet. The *Boston* bin is highlighted in gray to indicate users current focus. (d) Conceptualizing the multi-granularity overview.

Navigational Operation: Clicking. When a user clicks on a specific bin, NOAH displays the corresponding spreadsheet data; users can use this to jump to a specific spreadsheet location without having to scroll endlessly. For example, in Figure 2b, as the user clicks on the *Boston* bin, the data corresponding to Boston is displayed (Figure 2c). Note that the click operation is different from the traditional spreadsheet `Filter` operation. `Filter` hides spreadsheet data that do not satisfy the filtering condition while clicking brings the desired subset of data in view without hiding the rest. Users are free to navigate to other portions through scrolling even after clicking a bin, unlike filtering, where users need to issue another `Filter` to bring other data into view.

Navigational Operation: Semantic Zooming. Users can zoom into a specific bin to view more fine-grained information or zoom out to view more coarse-grained information, via semantic zooming [17]. For example, in Figure 2a, from the bin *Ashville-Boston* when the user zooms in to the next level, NOAH displays the bins *Ashville*, *Austin*, and *Boston* (Figure 2b). If the user zooms out of the current granularity, again NOAH displays the bins *Ashville-Boston*, *Chicago-Denver*, and others. Users can only zoom into any bin that contains multiple unique values. For example, in Figure 2d, at level 2, each bin corresponds to one city. Therefore, users can only click on those bins to bring that data into view, and cannot zoom in further. One issue with zooming interactions is discoverability of the zoom operation [14]. We circumvent this (see Figure 2c) by providing the root of the bin under selection for zoom out, and arrows for clicking to zoom in ("〉") and out ("〈").

Customizing the Overview. As NOAH constructs the overview automatically, the overview binning or organization may not capture domain-specific context or user needs. NOAH enables users to customize this organization (**DC5**). At any granularity, users can merge multiple consecutive bins into a single bin, or split a bin into multiple bins. Say the user wants to compare summary statistics of Boston and Chicago. In the current organization these two cities are in two different bins (see Figure 3a). Using the bin customization feature, the user can merge the two bins *Ashville-Boston* and *Chicago-Denver* to create a new bin *Ashville-Denver*. Users can now zoom into this bin and compare summary statistics of the cities in the same view. The interactions for splitting a bin depend on the data type. If the navigation attribute is textual, any bin can be split into as many bins as the number of unique values that bin contains. If the navigation attribute is numeric, users can split the bin into any arbitrary number of bins. Note that NOAH does not allow users to rearrange the order of the bins. Since the overview represents a histogram, the bins are ordered—reshuffling the bins violates that order.

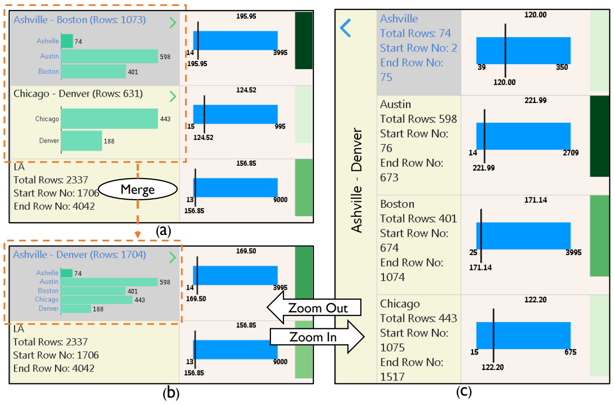


Fig. 3. (a) Chart view of the aggregate column. (b) A new bin is created by merging the top two bins. (c) Zooming into the newly created bin.

5.1.2 Coordination Between Overview and Spreadsheet

NOAH supports coordination between the overview and the corresponding spreadsheet data (**DC4**), *i.e.*, interactions on the overview may be reflected on the spreadsheet and vice-versa. One example of this coordination is indicating the navigation attribute on the spreadsheet using color (see the lime green column in Figure 1c) as user constructs the overview. However, not all overview interactions are coupled with the spreadsheet and vice versa. The coupling depends on the user's current focus—*to ensure consistency between the overview and the spreadsheet, any interaction on either interface that changes the current focus must be reflected on the other interface*. We now provide examples of both coupled and decoupled interactions.

Coupled interactions. Clicking a bin is an example of a coupled interaction as the user actively changes the focus to another bin on the overview. To reflect the change, NOAH populates the corresponding spreadsheet data on the screen. As the user scrolls on the spreadsheet, again the current focus changes and the corresponding bin on the overview is highlighted. For example, in Figure 2c, as the user clicks on the Boston bin, the spreadsheet displays the Boston listings. Conversely, as the user scrolls up, both Austin and Boston listings appear in the current window of the spreadsheet. Therefore, both the Austin and Boston overview bins are highlighted (see Figure 2d).

Decoupled interactions. When a user zooms into a bin that is already in the user's current focus, the spreadsheet view does not change. For example, in Figure 2a, the user zooms into the *Ashville-Boston* bin; here, the spreadsheet view stays the same (see Figure 2b). Similarly, the zoom out operation is decoupled. When a user zooms out, the overview displays a coarser granularity view of the user's current focus. Since the focus stays the same, there's no need to update the spreadsheet view. Similarly, operations like panning on the overview without clicking, and customizing the overview do not change user's current focus and are therefore decoupled. Online maps also adopt similar decoupling of the overview and detail [14]. However, their goal is to reduce network and computational overload, whereas in our case, the decoupling is based on the user's current focus.

5.2 Aggregate Columns

Users can issue spreadsheet formulae on the overview to compute aggregates for the data in each bin. The results are displayed as an *aggregate column* (see Figure 1b). Each entry in the aggregate column corresponds to the adjacent bin in the current granularity of the overview. For example, in Figure 3c, the aggregate column displays four aggregate statistics, one per bin. Users can issue several formulae simultaneously, each giving rise to a new aggregate column. However, adding an aggregate column takes up screen space, shrinking the spreadsheet view. As a workaround, users can resize or remove aggregate columns if required (**DC3**). When the user issues a formula on the overview, the spreadsheet column corresponding to the aggregate column is highlighted in grayish orange (see Figure 1c)—another example of coupled interaction. For conditional formulae like `COUNTIF`, cells that satisfy the condition are highlighted, *e.g.*, in Figure 1c, the cells with availability $\geq$ 60 are colored in sky blue. In this manner, users can quickly determine which cells are relevant to the aggregation operation.

Creating an aggregate column on the overview mimics how users create pivot tables. Users are not required to explicitly type formulae; rather they simply select the formula from a drop-down menu, and provide the necessary formula parameters to a form. The aggregate column can employ any statistical or mathematical formulae that operate over a range of data. Therefore, creating an aggregate column is equivalent to selecting subsets of data on the sheet, *i.e.*, steering, and then executing a formula on this subset, helping users avoid cumbersome steering operations. We have classified the formulae supported into five categories: a) summary (*e.g.*, min, max, average), b) frequency (*e.g.*, mode, large, small), c) conditional (*e.g.*, countif, sumif), d) spread (*e.g.*, var,stdev), and e) others (*e.g.*, sum, count).

Users can view the results either as raw values or as charts, and can toggle between the two. Raw values are displayed along with a colored bar, the *value bar*, whose length is proportional to the corresponding aggregate (see Figure 1b). Users can use the lengths to visually compare across bins. The chart representation varies depending on the formula type. All other categories except for the *others* category can be represented by charts. Figure 4 shows the chart representation for these categories along with different visual cues that highlight formula results as well as other information. We discuss these representations in detail in the Appendix.

Finally, we note that the aggregate column is kept in sync with the bins as users zoom in and out, eliminating

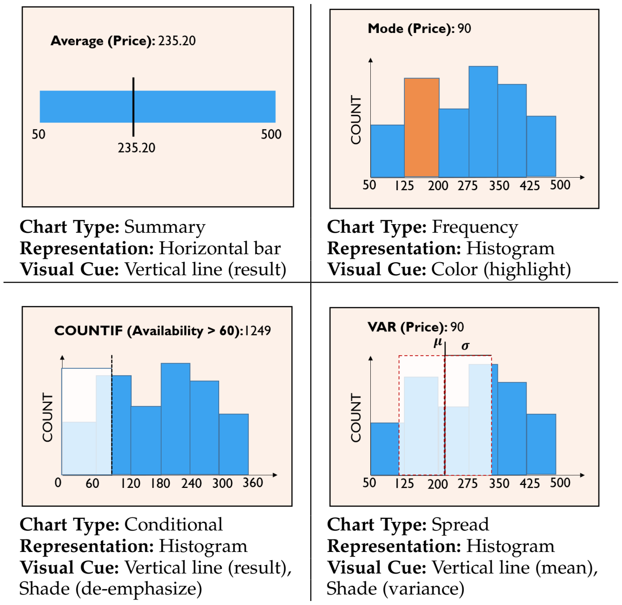


Chart Type: Summary
Representation: Horizontal bar
Visual Cue: Vertical line (result)

Chart Type: Frequency
Representation: Histogram
Visual Cue: Color (highlight)

Chart Type: Conditional
Representation: Histogram
Visual Cue: Vertical line (result), Shade (de-emphasize)

Chart Type: Spread
Representation: Histogram
Visual Cue: Vertical line (mean), Shade (variance)

Fig. 4. Formula types and their chart representations.

repeated steering operations. NOAH does not maintain any additional data structure for the aggregate column. The histogram underlying the overview records the result of the aggregate column entries corresponding to the bins. Next, we discuss how NOAH maintains user's navigational context.

5.3 Context Bar

The context bar consists of two components: a) a breadcrumb, and b) a navigation history. The breadcrumb [62] displays the current navigation path (see Figure 1e), thus maintaining the users' navigation context (**DC6**). Each component of the breadcrumb corresponds to a bin in the user's current navigation path. Therefore, users can visit any bins within the current navigation path by clicking on an appropriate component of the breadcrumb, without having to zoom in or zoom out. NOAH also maintains a list of recently visited bins (**DC6**) (see Figure 1d).

5.4 Implementation

We have integrated NOAH with DATASPREAD [35], a web-based spreadsheet. The DATASPREAD back-end maintains the histogram data structure and supports the aggregate column computation via its built-in formula engine. The NOAH front-end is built with HTML/CSS/JS technologies along with the D3 framework [63] for generating charts. We explain the NOAH architecture in detail in the Appendix.

6 EVALUATION STUDY DESIGN

In this section, we present the design of a user study to evaluate whether NOAH helps address spreadsheet navigational challenges.

6.1 Study Design and Participants

Our goal is to study the impact of an in-situ navigation plugin for spreadsheets, NOAH, on navigation and exploration of data. Therefore, we decided to compare a NOAH-integrated spreadsheet system with a typical, popular one, Excel, across various tasks. Similar domain specific-evaluations have been performed for evaluating various overview+detail interfaces, *e.g.*, database browsing [50] or tree navigation [64]. As explained in Section 3, the goals and user populations of spreadsheets and TDA tools are quite different. Therefore, we did not consider TDA tools for the comparative study. Our study was designed to answer the following questions:

- **RQ1.** How does the integration of an overview plugin like NOAH impact the efficiency of navigation within and the usability of spreadsheet systems?
- **RQ2.** How do the various components of NOAH impact users' navigational experiences?

Study Design. We conducted a 2 × 2 (2 datasets, 2 tools) mixed design within-subject study. The two tools used in the study were: Microsoft Excel, and NOAH integrated within DATASPREAD [18] (henceforth, referred to as NOAH for succinctness). As mentioned previously, we chose Excel for our comparative study because it is the most popular spreadsheet in use today. The study consisted of three phases: (a) an introductory phase explaining the essential features of NOAH via a video tutorial, followed by a warm-up session where participants explored a flight dataset [65] in NOAH to familiarize themselves with its features, (b) a quiz phase where the participants first used both the tools to perform targeted tasks on two different datasets (described later) followed by a survey to provide feedback on their impressions about Excel and NOAH, and (c) a semi-structured interview to collect qualitative feedback regarding the quiz phase.

Datasets. We used two datasets—the birdstrikes (used for evaluating visual data exploration tools like Keshif [54] and Voyager [55]), and the Airbnb [25] datasets. These datasets were chosen for their understandability to a general audience. The birdstrikes dataset records instances of birds hitting aeroplanes in different US states. The dataset has 10,868 records and 14 attributes (eight categorical, one spatial region, one temporal, four numeric). The Airbnb dataset was larger than the birdstrikes dataset. To ensure a fair comparison across tools, we created a sampled version of the original Airbnb dataset with 10,925 records, by uniformly sampling 10% of the records from each US city. This dataset contained 15 attributes (six categorical, two spatial region, one temporal, six numeric).

Participants. We recruited 20 participants (11 female, 9 male) via flyers across the university and via a university email newsletter. The average age of the participants was 31.06 years ($\sigma = 12.44$). The participants came from different backgrounds, *e.g.*, engineering (seven), business (five), administration (five), and natural science (three). During recruitment, prospective participants filled out an interest form where they answered questions about their spreadsheet expertise, their typical goals when using spreadsheets, and the spreadsheet operations they typically use. Participants were asked to rate their expertise with different spreadsheet systems, *e.g.*, Excel and Google Sheets, and their frequency of using various spreadsheet tasks *e.g.*, data management, data analysis, statistical modeling, and what-if analysis. We also asked participants about their familiarity with basic mathematical and statistical spreadsheet functions, as well as advanced operations, *e.g.*, pivot table, `SUBTOTAL`, and conditional formatting. To ensure that prior experience with spreadsheets didn't affect the performance of participants during the quiz phase, we only recruited participants who rated their experience with Excel to be greater than four on a scale of one (no expertise at all) to five (very experienced). The selected participants were familiar with performing various tasks on spreadsheets, *e.g.*, maintaining, tracking, and analyzing data, making predictions, and performing comparisons. All of the participants were familiar with the basic mathematical and statistical functions supported by Excel. Each participant received $10 per hour at the end of their session.

6.2 Study Procedure

We now explain each of the phases of our study in more detail.

Phase 1: Introduction to NOAH. We began the study by showing a six-minute video tutorial explaining the features of NOAH on a dataset of all the flights across the US for January 2018 [65]. The participants then explored the same dataset using NOAH to familiarize themselves with the tool for about 10 minutes. The quiz phase began as soon as the participants finished their exploration.

TABLE 2
Quiz tasks for the birdstrikes dataset. The task purposes and use cases correspond to the task typology discussed in Section 2.

Category	Question (Q), Purpose (P), Use case (U)
steer	**Q**: Organize the data by State. How many flights that had damages (damage = 1) originated from Florida?, **P**: *Search→Query*, **U**: `lookup→identify`
find	**Q**: How many flights in the currently visible spreadsheet window have damages?, **P**: *Search*, **U**: `browse`
steer	**Q**: Organize the data by State. How many flights that had damages (damage = 1) originated from California?, **P**: *Search→Query*, **U**: `lookup→identify`
compare (2)	**Q**: Which state between Florida and California has a higher number of flights with damages?, **P**: *Query*, **U**: `compare`
compare (N)	**Q**: Find the state with the most birdstrike occurrences, **P**: *Query→Search*, **U**: `summarize→locate`
customize	**Q**: Organize the data by *altitude*. What is the average cost of damages for altitude bin 0-450?, **P**: *Query→Search→Produce*, **U**: `generate→summarize→lookup`

Phase 2: The Quiz Phase. The purpose of the quiz phase was to evaluate the effectiveness of NOAH in addressing spreadsheet navigation limitations. During the quiz phase, each participant performed specific tasks on the two datasets in two sessions, using Excel for one and NOAH for the other. Each session was followed by a survey, described later. We alternated the order of the datasets between consecutive participants. The order of the tools was alternated between every two participants. We developed an online JavaScript-based quiz system that recorded user responses and submission times. We also recorded the participants' interactions with both tools using screen capture software. Participants were informed that they can refer to the Internet for help as many times as they wanted. However, due to their familiarity with Excel, none of the participants required external help. For reference, we also provided a printed handout to the participants that contained screenshots with the features of NOAH.

Quiz Tasks. We designed six tasks across five categories: steer (two tasks), find (one task), compare (2) (one task), compare (N) (one task), and customize (one task), encompassing six of the seven task typology use cases underlying the *Search*, *Query*, and *Produce* purposes: `lookup/locate`, `identify`, `browse`, `compare`, `summarize`, and `generate` (see Table 1)[2]. These tasks were selected to mimic a typical spreadsheet analysis workflow and are representative of navigation interactions required for the most frequently issued spreadsheet operations [10], [11]. The tasks were presented in the same order as shown in Table 2 for the birdstrikes dataset. The tasks for the Airbnb dataset mimic a scenario similar to the example in Section 2. We explain the scenario in the context of the birdstrikes dataset next.

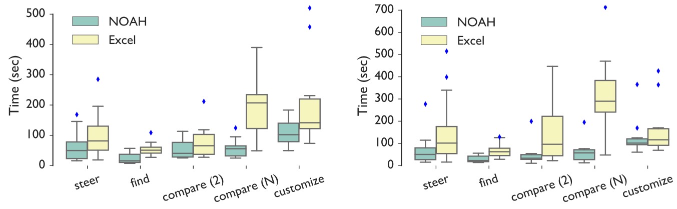
Say a user is interested in analyzing bird-strike statistics across US states. As the user is from Florida, she starts by computing bird-strike occurrences for that state (steer) and finds the number of occurrences to be quite high. After looking at the aggregate statistics, she decides to examine specific instances of bird-strike occurrences to inspect other attributes, *e.g.*, the altitude where the strike happened or the species of the bird (find). She notices a bird-strike occurrence at 50ft, which is surprising. She decides to investigate this issue later. For now, she focuses her attention on analyzing the state-wise statistics. She computes the same occurrence statistics for another large state, say California (steer) and then compares the statistics between the two large states (compare (2)). At this point, she becomes interested in learning the occurrence statistics across all states and in finding the state with the highest bird-strike occurrences (compare (N)). With the state-wise comparison completed, she decides to revisit low altitude bird-strikes. So she organizes the data by *altitude* and computes the occurrence statistics at low altitudes, along with the associated damages (customize). For this final task, the bins generated by NOAH did not correspond to the given altitude range and would require bin customization to compute the statistics.

Survey. After each session, participants rated the corresponding tool used on six metrics: confidence, comprehensibility, level of satisfaction, ease and speed of use, and ease of learning for spreadsheet navigation, on a Likert scale from one (*e.g.*, strongly disagree) to seven (*e.g.*, strongly agree). The survey asked multiple questions related to these metrics, 15 in total, to ensure reliability. Participants were also asked to describe the positive and negative aspects of both tools.

Evaluation. We evaluated the accuracy and completion time for each of the six tasks. We combined this analysis with qualitative survey, interview, and screen/audio recording data to provide insights that can be corroborated across multiple sources. Moreover, we analyzed the survey responses to quantify the usability of both Excel and the NOAH-integrated spreadsheet.

Phase 3: Interview Phase. Following the survey, we conducted a semi-structured interview to identify participants preferred tools for different tasks and to understand the

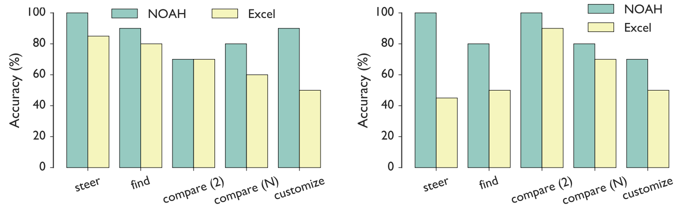
[2] We omitted the `export/save` use-case under the *Produce* purpose since NOAH is not targeted at improving that use case. We also did not study the *Consume* purpose so as to focus our evaluation on completion of tasks as opposed to an open-ended exploration setting, which is beyond the scope of our study.



(a) birdstrikes

(b) airbnb

Fig. 5. Submission times per category for each dataset. Median submission times are much smaller for NOAH compared to Excel.



(a) birdstrikes

(b) airbnb

Fig. 6. Per category accuracy for each dataset. Participants attained higher accuracy while completing tasks in NOAH compared to Excel.

reasoning behind their choices. We also asked participants to comment on the usefulness of different features provided by NOAH and Excel.

7 Results

In this section, we analyze the quantitative and qualitative data collected during the quiz and interview phases to address our research questions.

7.1 RQ1. Impact of overview-spreadsheet integration on navigation performance and spreadsheet usability

To answer RQ1, we first compare task completion times and accuracies in NOAH and Excel and then analyze the survey responses that evaluate the usability of the tools.

7.1.1 *Faster navigation without sacrificing accuracy*

In Figure 5a and 5b, we show the distribution of submission times of participants for the five task categories, for birdstrikes and Airbnb respectively. For most categories, *participants' median submission times using* NOAH *were less than the fastest submission times using Excel*. This observation suggests that the new capabilities offered by NOAH made spreadsheet navigation faster for these tasks. We analyzed the intra-participant differences in submission times, which also supported these observations: the majority of submission times using NOAH were faster than Excel—19 out of the 20 participants completed at least four tasks in less time using NOAH. The submission time differences were more prominent for the steer, find, and compare (N) tasks. In Section 7.2, we explain these outcomes in detail. For example, the aggregate column feature provides a faster alternative to steering, the overview-spreadsheet coordination accelerates raw data inspection, and the binned overview coupled with the context bar enables faster comparison. The differences in submission times were statistically significant for all of the tasks except customize. Both the intra-participant differences and the statistical significance test results are discussed in detail in the Appendix.

In Figure 6a and 6b, we show the percentage of correct submissions for the four quiz task categories, for the birdstrikes and Airbnb datasets, respectively. For all the tasks except for the fourth task, compare (2), for which the accuracy was the same for both tools, participants attained slightly higher accuracy with NOAH compared to Excel. However, the differences in accuracies were statistically significant for the steer tasks only (see Appendix). We evaluate the usability of Excel and NOAH next.

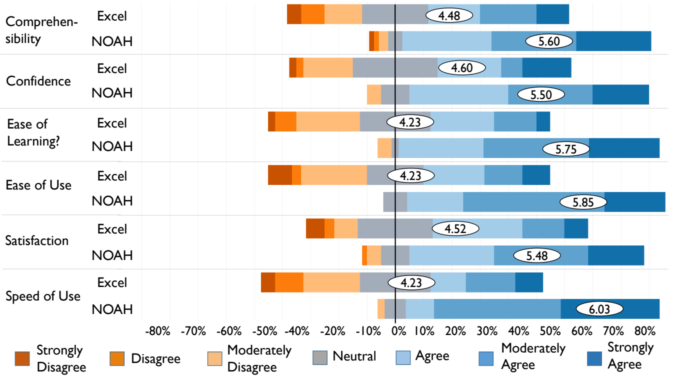


Fig. 7. Participants found NOAH to be easier to use compared to Excel while being faster in completing tasks involving navigation.

7.1.2 *Participants preferred* NOAH *to Excel*

Figure 7 shows a diverging stacked bar chart representation of the survey results in which participants rated their experience with Excel and NOAH. For each metric mentioned in Section 6, there are two stacked bar charts, one for Excel and one for NOAH. Each component within a stacked bar represents the percentage of responses for the corresponding rating, where the ratings are on a scale of one one (strong disagreement) to seven (strong agreement). The average rating for each metric is represented with a white ellipse. Notably, NOAH had a higher average rating than Excel for all the metrics. The aforementioned observation was further validated by a statistical significance test—the *Wilcoxon Signed-rank Test* (see Appendix). In particular, participants felt that using NOAH was faster and easier compared to Excel.

7.2 RQ2. Impact of NOAH and its components on spreadsheet navigation experience

To answer RQ2, we assess how NOAH's components, *i.e.*, the binned overview, aggregate column, and context bar, impacted participants' navigation. For each observation, we present participant feedback from the interview phase.

7.2.1 *Binned Overview: Customizable Hierarchical Organization*

Overall, the binned overview prevented participants from being overwhelmed during navigation, especially at scale. Personalizing the overview enabled participants to define their own grouping of the data, resulting in a more meaningful overview presentation. However, the newer interactions at times deviated from spreadsheet semantics, contributing to a steeper learning curve.

Overviews aid navigation at scale. Participants found it difficult to perform various navigation tasks in Excel, especially at scale; NOAH, on the other hand, helped participants avoid endless scrolling via clicking and semantically zooming on the overview, and provided cues for what to explore next via the bins of the overview. One participant ($P11$) commented—"*Excel can get overwhelming if you have a lot of data in it and sometimes with that data finding things can be difficult*". Participants ($N = 6$) mentioned that they would prefer NOAH when the dataset is large: "*If I just had a large amount of data then I would prefer to use* NOAH *because then you would be able to see all of it (bins) at once*" ($P2$). NOAH's binned overview helped participants comprehend the overall structure of the data better and prioritize the bin they want to visit next. One participant ($P5$) commented: "*I think it was just a little bit easier to navigate and find where things were because you could already see what bins had what.*" Another participant ($P1$) said: "*I like* NOAH *a lot better. It was a lot easier to look up different data and it was a lot quicker too*".

Overview customization enables related data to be analyzed together in task-specific ways. Bin customization enabled participants to personalize the overview based on their specific needs. One participant ($P16$) commented: "*I did like the fact that it lets you take a data sheet and, in some way, containerize the stuff you care and the stuff you don't care about.*" Participants (14 out of 20) preferred the feature to Excel's filtering feature when working with numeric data—"*That was so much easier in* NOAH *than it was in Excel to be able to specify the range that you wanted it to go in*" ($P17$). Our analysis of the video recordings revealed that for the birdstrikes dataset in Excel, the customize task involved filtering out certain values from a total of 451 unique values. This manual filtering led to a significant delays in task completion, compared to NOAH where they were able to use the bin customization feature. However, the time taken for this task was higher than other tasks in NOAH, as it required participants to restructure the overview before any calculation could be performed.

Overview customization interactions have a steeper learning curve. Unfamiliarity with the interactions required during customization in NOAH also contributed to higher task completion times for the customize task compared to other tasks. The unfamiliarity led to some participants ($N = 5$ out of 20) preferring Excel over NOAH for this task. One participant ($P11$) commented: "*Since I'm not used to spreadsheet data being presented that way, it took a little bit of getting used to.*" Participants found some of the terminology used in the interface—*e.g.*, explore, bin—quite unfamiliar ($N = 14$). Moreover, two participants didn't understand how the bins were constructed and requested implementation details during the interview.

Tradeoffs between hierarchical and flat overviews. While participants generally appreciated the binned representation of the overview for numeric data, a number of participants ($N = 6$) stated that they would have preferred a non-hierarchical overview for categorical data, where each bin corresponds to one item. One participant ($P13$) commented: "*I would prefer it start with all the bins split, and then I can merge them as I want.*" Another participant ($P4$) said—"*When I started, it* (NOAH) *had already grouped them, I think, alphabetically. So, that creates an extra step in that I then have to go split them and then re-merge them.*"

7.2.2 *Aggregate Column: In-situ Steering-free Computation*

The aggregate column feature enabled participants to avoid cumbersome steering interactions, resulting in faster and more accurate analysis compared to Excel. However, comparing the analysis results of more than two data subsets resulted in increased visual discontinuity and consequent errors.

Cumbersome steering replaced by a few button clicks with aggregate columns. The steer tasks required participants to issue a `COUNTIF` formula on a data subset. Participants found scrolling and steering in Excel to be cumbersome while issuing formulae—"*The one thing with Excel is I always try to go to the bottom of the data and type in the formula, and with something really long like this, the scrolling is a little bit cumbersome*" ($P4$). With NOAH, participants avoided steering by using aggregate column feature on the menu-bar and selecting the appropriate formula. Multiple participants ($N = 13$) found it easier to issue formulae using this feature. One participant ($P3$) commented: "*And that creates convenience sort of because then you don't have to memorize anything and using the system becomes easier.*" Another participant ($P13$) commented: "*There were some formulas to calculate, that were definitely easier in* NOAH *because the aggregate column did all the work and showed me the results.*" However, two participants found the aggregation operations applied on the bins to be opaque compared to Excel where a user can directly manipulate the formula.

Issuing formulae is faster and more accurate with aggregate columns. While the accuracies and submission times for the steer tasks in Excel varied significantly across datasets, using NOAH, participants exhibited higher accuracies and faster submission times irrespective of the dataset (see Figure 5 and 6). The automated and steering-free aggregate column feature of NOAH contributed to high accuracies (100%) for the steer tasks. One participant ($P12$) commented: "*With* NOAH, *you don't have to highlight every number versus Excel where you actually have to select everything.*" All of the 14 inaccurate submissions with Excel involved steering an incorrect spreadsheet region; 11 of the inaccurate submissions were with the Airbnb dataset. In NOAH, participants were able to avoid steering by using the aggregate column feature. Analysis of screen recordings revealed that, for Excel, for the birdstrikes dataset, several participants used the *autosum* feature to quickly count the number of 1's in the binary-valued column involved in the steering task. Summing up binary values is equal to the number of 1's in the collection. Other participants used the status bar at the bottom of the spreadsheet that displayed the sum of the cells in the selected column. In both cases, participants avoided steering the data resulting in fewer errors. On the other hand, for the Airbnb dataset, participants could not use these shortcuts as the column involved in the steering task was non-binary (it had 365 different values). Failure to avoid steering often led participants to select an incorrect range of data ($N = 14$ cases), resulting in incorrect responses. Therefore, the participants' ability to avoid steering depended on the data type.

Visual discontinuity during comparison while reduced, was not completely eliminated. For compare (N) tasks, participants had to perform N comparisons in NOAH while issuing the aggregate column operation once. However, the comparison among N bins resulted in increased visual discontinuity. This lead to some ($N = 4$ out of 20) incorrect submissions. In Excel, the experience was worse, as the participants had to perform N separate steering tasks. As a result, in Excel, the compare (N) task submission times were very high compared to compare (2) tasks (see Figure 5). In addition, the accuracies of the compare (N) task in Excel were lower ($N = 7$ out of 20 submissions were inaccurate).

7.2.3 History, Context, and Coordination

The context bar enabled participants to revisit previously explored bins. The aggregation results corresponding to that bin were automatically updated in the aggregate column, due to automatic syncing between the binned overview and aggregate column (see Section 5.2). The coordination between the overview and the raw spreadsheet data further helped participants relate the aggregate column results with the raw data.

History helps avoid repeated interactions. For the compare (2) task in NOAH, all of the participants used the context bar to navigate to a bin previously visited for the first steer task. As the bin currently being displayed was changed, the aggregate column was automatically updated to display values corresponding to that bin, enabling participants to view the aggregate column values instantly without having to reissue the operation. On the other hand, as Excel did not preserve any navigation history, participants had to re-execute the first steering operation. As a result, the submission times for compare (2) tasks were faster in NOAH compared to Excel (see Figure 5). One participant ($P16$) commented—"*Once I got familiar with the interface, it was easy to just say, I want to see this state, and I like that fact that like automatically it goes into the bins on NOAH, gave me summary information.*" Another participant ($P9$) said—"*Noah was easy to find and compare and toggle in between.*"

Overview-spreadsheet coordination helps relate interactions on the overview with the raw data. The coordination between the overview and spreadsheet in NOAH enabled users to quickly relate their interactions on the overview with the raw spreadsheet data. For example, for the find task, participants had to find all the cells within the spreadsheet that satisfy a condition corresponding to the preceding steer task. To do so, they had to skim through all the cells in the current window in Excel, resulting in higher completion times. Even though Excel supports a conditional formatting feature[3], issuing the feature added one additional step when performing the find task. In NOAH, participants benefited from having visual cues in the form of automatically colored cells, helping them relate the aggregate column with the raw data—"*You didn't have to do any additional steps and it was a visual cue right there, made it very quick to count it up ($P17$).*" Another participant ($P9$) commented—"*In Excel, you would have to add your own condition for formatting. But you have to build that (conditional formatting) every time you need to ask a question. This one* (NOAH) *at least something is pre-built in, and you can easily count.*" However, one participant ($P3$) pointed out that, when the data corresponding to the bin does not fit in the screen, they had to scroll through to identify relevant information.

8 Discussion and Conclusion

NOAH represents our first step towards a general purpose spreadsheet navigation plug-in to make spreadsheets more effective when exploring datasets that are increasingly the norm. Using NOAH, users can get a bird's eye view of the data, with the ability to scroll or seek additional details on demand via a multi-granularity overview, as well as employ aggregattion in-situ, which eliminates cumbersome steering operations. Quantitatively, we find that NOAH speeds up navigation without compromising accuracy. Qualitatively, study participants identify it as positively impacting their experience while overviewing and navigating large datasets, and issuing formulae. However, the user study revealed some limitations of NOAH. Moreover, the study design itself had some limitations. In this section, we discuss these limitations and propose extensions to address these limitations as well as enhance the capabilities of NOAH.

8.1 User Study Limitations

Our study has a few limitations that can be strengthened by future larger-scale and finer-grained studies.

Insufficient coverage of participant demographics. Our participant pool demographics only partially represents the demographics of the general audience intended for NOAH. A larger sample with more participants with a range of skillsets and backgrounds that better represents the spreadsheet user population would have provided more ecological validity to generalize our findings.

Lack of targeted comparisons with advanced spreadsheet features. We only compared the performance of a NOAH-integrated spreadsheet with a traditional spreadsheet. We did not evaluate specific spreadsheet features like pivot table and `SUBTOTAL` as they violate most of the design considerations proposed in Section 4; we discussed their limitations in Section 3. We instead allowed the participants to freely use any spreadsheet operations that they were comfortable with (including the advanced ones), enabling us to observe how introducing a plug-in like NOAH affected their navigation experience. However, a future study targeted at evaluating the pros and cons of these features for spreadsheet navigation would be valuable.

Isolated evaluation of NOAH components. While we did present the impact of various components of NOAH in Section 7.2, we did not isolate the effects of individual features during our study. For example, we did not study the effects of the binned overview (visual clarity versus visual continuity), display layout (screen space trade-off), and contextual presentation of data (raw text versus chart representation of aggregate columns) in isolation. A more

[3] The conditional formatting feature in Excel enables users to request cells that satisfy certain conditions to be colored.

fine-grained study that teases apart the contribution of individual components of NOAH is warranted.

8.2 Limitations of NOAH and Possible Enhancements

While we alluded to some limitations of NOAH in Section 7, we now discuss these in more detail and how we can possibly overcome them.

Lack of transparency and documentation of new interactions. Several operations in NOAH are quite different from typical spreadsheet interactions, *e.g.*, zooming or bin customization. Moreover, participants found some of the terminologies, *e.g.*, explore or bin, to be rather different from typical spreadsheet terminologies, which took some getting used to (see Section 7.2). In addition, some participants complained about the lack of explanation surrounding the overview construction and aggregate column computation. In the future, these issues can be addressed by using more relatable terminologies and improved documentation.

Deviation from the spreadsheet look and feel. A couple of participants ($P2$ and $P11$) mentioned that the aggregate column results hides the actual spreadsheet formula and they would prefer some visual cues that highlight the corresponding formula underlying each bin. Subsequent versions of NOAH can display the appropriate formula for each bin as users hover over the corresponding cell on the aggregate column. The aggregate column feature can be further enhanced by enabling users to issue user defined formulae, a feature requested by a number of participants ($N = 5$). Another feature that can be made more similar to spreadsheet interaction semantics is the bin customization operation. Currently, this operation in NOAH is performed from a menu bar, adding an additional step. The bin splitting and merging operations can be made more similar to how spreadsheet cells are split or merged—in Excel, these operations are direct and only require a single click. Similar direct adjustment of data grouping strategies have been explored for visualization tools [66] and can be adapted to this setting.

Absence of bespoke overview representations for various data types. The experience surrounding the construction of the overview can be further improved, especially for categorical data. Currently, the bins of the overview can be customized only after the overview is constructed. Providing the users the ability to select the representation (similar to bin customization) of the overview at the outset could have possibly addressed this issue. Understanding the impact of these representation choices for the overview is an interesting open question.

8.3 Additional enhancements

We now discuss other functionalities that can be introduced to further enhance the capabilities of NOAH.

Broadening the scope of overview-spreadsheet coordination. Spreadsheet users often perform various edit operations, *e.g.*, updating values, adding/deleting rows/columns. However, our current NOAH implementation assumes the data to be read-only. In our next version, we can add support for propagating spreadsheet updates to the overview. Moreover, the charts displayed in an *aggregate column* are non-interactive, *i.e.*, users cannot interact with the charts to visually look up relevant or interesting data points within the spreadsheet. In the future, we plan to extend NOAH to support visual querying through the charts in an aggregate column, similar to multi-modal linked selections in Keshif [54].

Enabling more flexible overview binning. NOAH currently constructs the overview on a single attribute. We can add support for multi-attribute navigation (*e.g.*, explore the Airbnb data by city and neighborhood), and multi-level navigation (*e.g.*, explore the neighborhoods after zooming into a specific city in the Airbnb data). Furthermore, bin customization currently supports changing the bin boundaries only, while maintaining the current order. Supporting user-defined ordering to allow arbitrary reshuffling of the bins can be another enhancement.

Supporting other spreadsheet operations. Spreadsheet operations that involve working with subsets of the data, *e.g.*, sorting, filtering, copy-pasting, can be supported by NOAH, but our current implementation does not support them. Future versions would support such operations as well. Other enhancements include adding annotations, *e.g.*, visual cues, text, to the overview and then exporting the customized overview for presentation or reporting—a required feature for information seeking tools [58].

Supporting navigation for semi-structured data. NOAH operates only on tabular data. However, spreadsheets can be semi-structured—formulae and text can be interspersed with tabular data. In such cases, NOAH can be used to support exploration and zooming for each such tabular region independently, supported by an overall overview (akin to a map-based panning tool) for users to select which tabular region they want to explore in detail. We can leverage existing work on spreadsheet table detection [67], and property identification [68] to support such extensions.

Achieving interactive scalability beyond traditional spreadsheet limits. The current version of NOAH addresses the perceptual scalability challenges while navigating Excel-scale (one million rows) data. As modern spreadsheets continue to support increasingly larger datasets—DATASPREAD [35] supports one billion rows—the interactions proposed in this paper may violate the interactive response time bound of 500 ms [69]. This opens the door to a new set of research challenges that may range from approximate query processing to progressive data analytics.

References

[1] M. Brehmer and T. Munzner, "A multi-level typology of abstract visualization tasks," *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 12, pp. 2376–2385, 2013.

[2] N. Elmqvist and J.-D. Fekete, "Hierarchical aggregation for information visualization: Overview, techniques, and design guidelines," *IEEE Transactions on Visualization and Computer Graphics*, vol. 16, no. 3, pp. 439–454, 2009.

[3] Z. Liu, B. Jiang, and J. Heer, "immens: Real-time visual querying of big data," in *Computer Graphics Forum*, vol. 32, no. 3pt4. Wiley Online Library, 2013, pp. 421–430.

[4] "How finance leaders can drive performance," https://enterprise.microsoft.com/en-gb/articles/roles/finance-leader/how-finance-leaders-can-drive-performance/.

[5] Y. E. Chan and V. C. Storey, "The use of spreadsheets in organizations: Determinants and consequences," *Information & Management*, vol. 31, no. 3, pp. 119–134, 1996.

[6] N. Raden, "Shedding light on shadow it: Is excel running your business," *DSSResources.com*, vol. 26, 2005.

[7] E. Bakke and D. R. Karger, "Expressive query construction through direct manipulation of nested relational results," in *Proceedings of the 2016 International Conference on Management of Data*. ACM, 2016, pp. 1377–1392.

[8] B. A. Nardi and J. R. Miller, *The spreadsheet interface: A basis for end user programming*. Hewlett-Packard Laboratories, 1990.

[9] D. V. Beard and J. Q. W. II, "Navigational techniques to improve the display of large two-dimensional spaces," *Behaviour & Information Technology*, vol. 9, no. 6, pp. 451–466, 1990.

[10] D. A. Bradbard, C. Alvis, and R. Morris, "Spreadsheet usage by management accountants: An exploratory study," *Journal of Accounting Education*, vol. 32, no. 4, pp. 24–30, 2014.

[11] B. R. Lawson, K. R. Baker, S. G. Powell, and L. Foster-Johnson, "A comparison of spreadsheet users with different levels of experience," *Omega*, vol. 37, no. 3, pp. 579–590, 2009.

[12] J. Watts-Perotti and D. D. Woods, "How experienced users avoid getting lost in large display networks," *International Journal of Human-Computer Interaction*, vol. 11, no. 4, pp. 269–299, 1999.

[13] "Google sheets limit," https://support.google.com/drive/answer/37603.

[14] A. Cockburn, A. Karlson, and B. B. Bederson, "A review of overview+ detail, zooming, and focus+ context interfaces," *ACM Computing Surveys (CSUR)*, vol. 41, no. 1, p. 2, 2009.

[15] R. R. Panko, "What we know about spreadsheet errors," *Journal of Organizational and End User Computing (JOEUC)*, vol. 10, no. 2, pp. 15–21, 1998.

[16] J. Grudin, "Partitioning digital worlds: focal and peripheral awareness in multiple monitor use," in *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, 2001, pp. 458–465.

[17] K. Perlin and D. Fox, "Pad: an alternative approach to the computer interface," in *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*. ACM, 1993, pp. 57–64.

[18] "DataSpread," https://dataspread.github.io/.

[19] M. Meyer, M. Sedlmair, P. S. Quinan, and T. Munzner, "The nested blocks and guidelines model," *Information Visualization*, vol. 14, no. 3, pp. 234–249, 2015.

[20] L. McNabb and R. S. Laramee, "Survey of surveys (sos)-mapping the landscape of survey papers in information visualization," in *Computer Graphics Forum*, vol. 36, no. 3. Wiley Online Library, 2017, pp. 589–617.

[21] J. Fulda, M. Brehmel, and T. Munzner, "Timelinecurator: Interactive authoring of visual timelines from unstructured text," *IEEE transactions on visualization and computer graphics*, vol. 22, no. 1, pp. 300–309, 2015.

[22] M. Brehmer, S. Ingram, J. Stray, and T. Munzner, "Overview: The design, adoption, and analysis of a visual document mining tool for investigative journalists," *IEEE transactions on visualization and computer graphics*, vol. 20, no. 12, pp. 2271–2280, 2014.

[23] J. Pretorius, H. C. Purchase, and J. T. Stasko, "Tasks for multivariate network analysis," in *Multivariate Network Visualization*. Springer, 2014, pp. 77–95.

[24] T. Höllt, N. Pezzotti, V. van Unen, F. Koning, E. Eisemann, B. Lelieveldt, and A. Vilanova, "Cytosplore: Interactive immune cell phenotyping for large single-cell datasets," in *Computer Graphics Forum*, vol. 35, no. 3. Wiley Online Library, 2016, pp. 171–180.

[25] "The inside airbnb dataset." http://insideairbnb.com/get-the-data.html.

[26] "A Lone Data Whiz Is Fighting Airbnb—and Winning," https://www.wired.com/2017/02/a-lone-data-whiz-is-fighting-airbnb-and-winning/.

[27] "Define and use names in formulas," https://www.excel-easy.com/examples/names-in-formulas.html.

[28] "Pivot Table," https://en.wikipedia.org/wiki/Pivot_table.

[29] "SUBTOTAL," https://exceljet.net/excel-functions/excel-subtotal-function.

[30] D. L. Hansen, B. Shneiderman, and M. A. Smith, *Analyzing social media networks with NodeXL: Insights from a connected world*. Morgan Kaufmann, 2010.

[31] "Google explore." https://www.techrepublic.com/article/how-to-use-the-new-google-explore-feature/.

[32] M. Joglekar, H. Garcia-Molina, and A. Parameswaran, "Smart drill-down: A new data exploration operator," *Proceedings of the VLDB Endowment*, vol. 8, no. 12, pp. 1928–1931, 2015.

[33] "Hillview." https://github.com/vmware/hillview.

[34] V. Raman *et al.*, "Scalable spreadsheets for interactive data analysis," in *ACM SIGMOD Workshop on DMKD*, 1999.

[35] M. Bendre *et al.*, "Towards a holistic integration of spreadsheets with databases: A scalable storage engine for presentational data management," in *ICDE*, April 2018.

[36] R. Rao and S. K. Card, "The table lens: merging graphical and symbolic representations in an interactive focus+ context visualization for tabular information," in *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, 1994, pp. 318–322.

[37] B. B. Bederson, A. Clamage, M. P. Czerwinski, and G. G. Robertson, "Datelens: A fisheye calendar interface for pdas," *ACM Transactions on Computer-Human Interaction (TOCHI)*, vol. 11, no. 1, pp. 90–119, 2004.

[38] M. Spenke, C. Beilken, and T. Berlage, "Focus: the interactive table for product comparison and selection," in *Proceedings of the 9th annual ACM symposium on User interface software and technology*. ACM, 1996, pp. 41–50.

[39] M. Spenke and C. Beilken, "Visualization and analysis of formula one racing results with infozoom-the demo." in *INTERACT*, 2003.

[40] F. Nunez and E. Blake, "Vissh: A data visualisation spreadsheet," in *Data Visualization 2000*. Springer, 2000, pp. 209–218.

[41] M. Levoy, "Spreadsheets for images," in *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*. ACM, 1994, pp. 139–146.

[42] E.-H. Chi, P. Barry, J. Riedl, and J. Konstan, "A spreadsheet approach to information visualization," in *Information Visualization, 1997. Proceedings., IEEE Symposium on*. IEEE, 1997, pp. 17–24.

[43] K. W. Piersol, "Object-oriented spreadsheets: the analytic spreadsheet package," in *ACM Sigplan Notices*, vol. 21, no. 11. ACM, 1986, pp. 385–390.

[44] S. Kandel, A. Paepcke, M. Theobald, H. Garcia-Molina, and E. Abelson, "Photospread: a spreadsheet for managing photos," in *Proceedings of the SIGCHI conference on human factors in computing systems*. ACM, 2008, pp. 1749–1758.

[45] A. Cockburn, C. Gutwin, and J. Alexander, "Faster document navigation with space-filling thumbnails," in *Proceedings of the SIGCHI conference on Human Factors in computing systems*. ACM, 2006, pp. 1–10.

[46] S. Kratz, I. Brodien, and M. Rohs, "Semi-automatic zooming for mobile map navigation," in *Proceedings of the 12th international conference on Human computer interaction with mobile devices and services*. ACM, 2010, pp. 63–72.

[47] D. Nekrasovski, A. Bodnar, J. McGrenere, F. Guimbretière, and T. Munzner, "An evaluation of pan & zoom and rubber sheet navigation with and without an overview," in *Proceedings of the SIGCHI conference on Human Factors in computing systems*. ACM, 2006, pp. 11–20.

[48] C. Plaisant, J. Grosjean, and B. B. Bederson, "Spacetree: Supporting exploration in large node link tree, design evolution and empirical evaluation," in *IEEE Symposium on Information Visualization, 2002. INFOVIS 2002.* IEEE, 2002, pp. 57–64.

[49] J. C. Roberts, "State of the art: Coordinated & multiple views in exploratory visualization," in *Coordinated and Multiple Views in Exploratory Visualization, 2007. CMV'07. Fifth International Conference on*. IEEE, 2007, pp. 61–71.

[50] C. North and B. Shneiderman, "Snap-together visualization: can users construct and operate coordinated visualizations?" *International Journal of Human-Computer Studies*, vol. 53, no. 5, pp. 715–739, 2000.

[51] S. K. Badam, Z. Liu, and N. Elmqvist, "Elastic documents: Coupling text and tables through contextual visualizations for enhanced document reading," *IEEE transactions on visualization and computer graphics*, 2018.

[52] "Tableau software." https://www.tableau.com/.

[53] "Microsoft power bi." https://powerbi.microsoft.com/en-us/.

[54] M. A. Yalçın, N. Elmqvist, and B. B. Bederson, "Keshif: Rapid and expressive tabular data exploration for novices," *IEEE transactions on visualization and computer graphics*, vol. 24, no. 8, pp. 2339–2352, 2018.

[55] K. Wongsuphasawat, D. Moritz, A. Anand, J. Mackinlay, B. Howe, and J. Heer, "Voyager: Exploratory analysis via faceted browsing of visualization recommendations," *IEEE Transactions on Visualization & Computer Graphics*, no. 1, pp. 1–1.

[56] M. J. Norušis, *The SPSS guide to data analysis*. Spss, 1986.

[57] I. SAS, "Sas user's guide: statistics," *SAS Institute, Cary, NC*, 1985.

[58] B. Shneiderman, "The eyes have it: A task by data type taxonomy for information visualizations," in *The Craft of Information Visualization*. Elsevier, 2003, pp. 364–371.

[59] M. Q. Wang Baldonado, A. Woodruff, and A. Kuchinsky, "Guidelines for using multiple views in information visualization," in *Proceedings of the working conference on Advanced visual interfaces*. ACM, 2000, pp. 110–119.

[60] Y. Ioannidis, "The history of histograms (abridged)," in *Proceedings 2003 VLDB Conference*. Elsevier, 2003, pp. 19–30.

[61] H. Mousavi and C. Zaniolo, "Fast and accurate computation of equi-depth histograms over data streams," in *Proceedings of the 14th International Conference on Extending Database Technology*. ACM, 2011, pp. 69–80.

[62] "A Lone Data Whiz Is Fighting Airbnb—and Winning," https://en.wikipedia.org/wiki/Breadcrumb_(navigation).

[63] M. Bostock, V. Ogievetsky, and J. Heer, "D^3 data-driven documents," *IEEE transactions on visualization and computer graphics*, vol. 17, no. 12, pp. 2301–2309, 2011.

[64] D. V. Beard and J. Q. W. II, "Navigational techniques to improve the display of large two-dimensional spaces," *Behaviour & Information Technology*, vol. 9, no. 6, pp. 451–466, 1990.

[65] "The us flight dataset." https://www.transtats.bts.gov/.

[66] A. Sarvghad, B. Saket, A. Endert, and N. Weibel, "Embedded merge & split: Visual adjustment of data grouping," *IEEE transactions on visualization and computer graphics*, vol. 25, no. 1, pp. 800–809, 2018.

[67] H. Dong, S. Liu, S. Han, Z. Fu, and D. Zhang, "Tablesense: Spreadsheet table detection with convolutional neural networks," in *Thirty-Third AAAI Conference on Artificial Intelligence*, 2019.

[68] Z. Chen, S. Dadiomov, R. Wesley, G. Xiao, D. Cory, M. Cafarella, and J. Mackinlay, "Spreadsheet property detection with rule-assisted active learning," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. ACM, 2017, pp. 999–1008.

[69] Z. Liu *et al.*, "The effects of interactive latency on exploratory visual analysis," *IEEE TVCG*, vol. 20, no. 12, pp. 2122–2131, 2014.

Sajjadur Rahman is a PhD student at the Department of Computer Science at University of Illinois, Urbana-Champaign, IL, USA. He received the M.Sc. degree in 2014 and the B.Sc. degree in 2012 from Bangladesh University of Engineering and Technology, Dhaka, Bangladesh.

Mangesh Bendre received the Ph.D. degree in 2018 from University of Illinois, Urbana-Champaign. He is a Research Scientist at VISA Research, Palo Alto, CA, USA.

Yuyang Liu is an undergraduate student at the Department of Computer Science at University of Illinois, Urbana-Champaign.

Shichu Zhu is a graduate student (MCS) at the Department of Computer Science at University of Illinois, Urbana-Champaign. He received the B.S. in Atmospheric and Oceanic Science from Peking University, Beijing, China.

Zhaoyuan Su is a PhD student at the Donald Bren School of Information and Computer Sciences at the University of California Irvine, CA, USA. He received the B.S. in Architecture from University of Illinois, Urbana-Champaign.

Karrie Karahalios received the Ph.D. degree in 2004 from Massachusetts Institute of Technology, Cambridge, MA, USA. She is a Professor in Computer Science at the University of Illinois, Urbana-Champaign.

Aditya Parameswaran received the Ph.D. degree in 2013 from Stanford University, CA, USA. He is an Assistant Professor in the School of Information and Electrical Engineering and Computer Sciences at the University of California, Berkeley, CA, USA. He was previously an assistant professor in Computer Science at the University of Illinois, Urbana-Champaign.

INFOVIS Review

Primary Reviewer review
score 3/5

Paper type

Algorithm / Technique

Expertise

Expert

Overall Rating

<b>3 - Possible Accept</b><br/> The paper is not acceptable in its current state, but might be made acceptable with significant revisions within the conference review cycle.<br/>If the specified revisions are addressed fully and effectively I may be able to return a score of '4 - Accept'.

Supplemental Materials

Acceptable

Justification

The paper describes a plugin for a spreadsheet tool that aids in navigating large (row-wise) spreadsheet datasets. It seems useful in practice, and I haven't seen this level of navigational aid in an actual spreadsheet tool. The study shows that the tool improves on a vanilla spreadsheet tool. I'm slightly concerned about novelty, as similar operations could be achieved with tools such as Keshif.

The Review

This paper discusses a tool (NOAH) for an in-situ interface for exploring large spreadsheets. The tool is implemented as a plugin for a web-based open source spreadsheet tool.

NOAH provides an overview of the dataset in panels on the left side of the spreadsheet. This overview takes the form of a histogram; the user has to click on the bins to drill down as a form of semantic zooming.

NOAH shows an aggregate column which has aggregate data in the form of raw numbers or visualizations for each bin. There is a context bar on the top that helps the user maintain his navigation context and also maintains a list of recently visited bins.

The paper is well written and easy to follow. The related work is comprehensive, although dismisses related tools with brief sentences. In particular, the treatment of Tabular Data Analysis tools is to short, and the argument that they are "not co-located with the underlying data, introducing visual discontinuities and lack of context" is IMO not true.

This is also the main weakness of the paper: A tool like Keshif supports most things that NOAH supports, it certainly supports all the things that the study tests. A strength of NOAH compared to TDA tools is that it's in an actual spreadsheet, where it's easy to derive new data based on formulas or expressions, but that's not well articulated in the paper and not really part of the study.

A design decision that I think is not optimal is the binning of multiple unrelated items into larger bins, supposedly to ensure scalability (e.g., Ash-Bos). It seems like an odd choice to group semantically unrelated items (lexicographic ordering isn't super helpful). It also doesn't seem to help much in terms of scalability, as each individual city is represented as a fairly large bar chart. It would probably be more effective to just bin separately, which also some participants of the study noted.

The paper describes a within-subject study with 20 participants who were selected based on their skills with spreadsheet software. The results of the study are positive with NOAH conditions consistently being completed faster and more accurately than Excel.

While I generally trust the results of the study, I have two comments: first, by using Excel vs another tool, it's very obvious to the users which of the tool the experimenter wants to do well, which is a potential source for bias. That could have potentially been mitigated by having them use the open source spreadsheet tool in both cases (which they are unlikely to be familiar with), which would also remove some additional confounders.

The selection of participants for the study indicates that all were expert spreadsheet users. I wonder how NOAH can help non-expert spreadsheet users navigate large datasets easily. Another study with users rating their spreadsheet skills 2 and above will help answer this question.

Overall I like the concept of a plugin to provide an in-situ overview of large datasets. The paper claims that this reduces scrolling and steering actions allowing a user to maintain context while reducing the sense of getting lost. The results of the experiments are indicators that the implementation of the concept has an impact on user experience.

Minor Comments:

* In Fig 1. the label for breadcrumb covers the actual breadcrumb view. Moving the label to the right will make it easy to get a sense of how breadcrumbs look in the final tool.

* Is it users that rate themselves as greater than 4, or at least 4 in terms of skills with spreadsheets?

Summary Rating

<b>Possible Accept</b><br/> The paper is not acceptable in its current state, but might be made acceptable with significant revisions within the conference review cycle.<br/>If the specified revisions are addressed fully and effectively I may be able to return a score of '4 - Accept'.

The Summary Review

All reviewers agree that NOAH is an interesting extension to navigate large spreadsheets and that the technique is simple and useful. The paper is also well written.

However, there are two major concerns:

* The novelty is limited, similar designs exist in tabular data visualization tools outside of spreadsheets. The benefits of integrating this kind of navigation in a real spreadsheet application isn't clearly articulated.
* The study design is biased as it does not compare to the state of the art. The study also doesn't test tasks related to viewing details. Other concerns with the study are raised in the individual reviews.

The reviews also bring up various other issues related to the design.

In light of the individual scores for the paper, we are giving a summary score of 3. However, we recognize that addressing the review comments, which are about substantial aspects (novelty / study), is likely beyond the scope of a minor conference revision.

Nevertheless, if accepted, we require the following changes for a revision:

* More clearly articulate the merits over alternative tools, such as Keshif.
* Convincingly explain the design choice around the binning mechanism or revise

that approach.
*Justify the choice of comparing to Excel and note the limitations of that approach
* Better justify the choice of tasks in the study.
* Explain the choice of the four research questions.
* Provide more details about the study, including reporting intra-participant differences (see reviews for details.)

Secondary Reviewer review
score 2.5/5

Paper type

Algorithm / Technique

Expertise

Knowledgeable

Overall Rating

<b>2.5 - Between Reject and Possible Accept</b><br/>

Supplemental Materials

Acceptable with minor revisions (specify revisions in The Review section)

Justification

This paper presents a neat idea (NOAH) to provide an overview alongside the table and use the visual elements in the overview as a shortcut to the raw data in the table. I agree that this would fix the navigation issue with Excel. However, because this overview is just a histogram for only one data column (i.e., one dimension), the significance of the research contribution is limited. In addition, the controlled experiment demonstrates if NOAH helps address spreadsheet “navigational challenges” instead of demonstrating NOAH’s value in facilitating holistic data exploration.

The Review

This paper presents a simple overview plug-in (called NOAH) designed to help people navigate a spreadsheet for a large-scale dataset. It also reports a controlled lab study conducted to compare NOAH with Excel.

In general, I agree with the motivation and overarching goal of this work: it could be useful to enable people to “explore” large tabular data “holistically” by showing the data rows and visualizations on the same page with tight coupling (i.e., without visual discontinuities in the authors’ term). However, the focus of this paper is too narrow and as a result the significance of research contribution is too small for a VIS publication.

As Shneiderman’s mantra—Overview first, zoom and filter, then detail on demand—indicates, seeing an overview is an important and useful step for data exploration. It is a neat idea to provide an overview alongside the table and use the visual elements in the overview as a shortcut to the raw data in the table. However, because this overview is just a histogram for only one data column, even with some additional features, NOAH’s exploration capability seems limited.

I note that the comparative study is designed to evaluate whether NOAH helps address spreadsheet “navigational challenges” instead of demonstrating NOAH’s value in facilitating holistic data exploration. It seems that the six quiz tasks are somewhat arbitrary or even biased to emphasize the navigational issue of Excel: I do agree that Excel’s navigation is not ideal. For the Steer and Customize tasks, why do the quiz tasks “instruct” participants to organize the data by some attributes? For the Identify task, why is it asking participants to answer the question from “the currently visible spreadsheet window?” More importantly, people with a basic level of expertise can easily answer these tasks using existing BI tools such as Tableau, Pivot Table, and Power BI. I know that, as the authors pointed out, these tabular data analysis tools do now show the tabular data and visualizations together on the same sheet.

However, Excel is not designed to best support navigation and instead it provides additional flexible features that the authors acknowledge as limitations. To me, it is unfair to choose the tool with a limitation as a baseline (excluding other more power existing tools) along with some questionable tasks, and show that NOAH is better than the baseline for the tasks demonstrate the limitation. Therefore, the study results are neither interesting nor informative.

What do the authors mean by the “mixed design within-subject study?” I am not sure why the authors wanted to treat the dataset as a main factor when the four main research questions do not involve dataset characteristics? I expect the explanation of the comparison of dataset characteristics and the discussion of interaction effect, which would make it worth introducing it as a main factor instead of running the study as a simple A/B comparison.

While I appreciate the authors’ effort on conducting a comparative study, it would have been a lot more interesting and informative if the authors conducted a qualitative study with high-level questions (which can be combined with a few meaningful low-level ones) that would encourage participants to utilize NOAH’s most novel features to see *IF* and *HOW* people use NOAH in their data exploration process.

Here are some additional comments.
- Even though the authors described their design considerations, I have some doubts about NOAH’s design. For example, despite the authors’ justification on the overview construction process using the common summarization technique, the number of items in each group seems small to me. Also, it is weird to see the different bar height between different groups when they have different number of cities and to see the completely different representation when a group has only one city (Fig. 2).
- Section 7.1.2 and 7.1.3 read like a mix of some results and discussion.
- Space is missing for “var,stdev” (p6) and for “responses.The” (p8).
- Additional period for “plugin..” (p9).
- [16] and [17] are the same paper.

External Reviewer 1 review
score 3/5

Paper type

System

Expertise

Knowledgeable

Overall Rating

<b>3 - Possible Accept</b><br/> The paper is not acceptable in its current state, but might be made acceptable with significant revisions within the conference review cycle.<br/>If the specified revisions are addressed fully and effectively I may be able to return a score of '4 - Accept'.

Supplemental Materials

Acceptable

Justification

This is a well-structured, well-written paper that tackles an interesting challenge: augmenting spreadsheets with visualizations to fasten some operations, reduce the need for steering and for navigating, and reducing cognitive load. The system presented does not present any novel visualization aspect, however the problem at hand is worth addressing and this is some good evidence that visualization in an overview+detail setting can help. I do have some reservations about the study; still I think that these points could be addressed by adding some details here and there and completing the discussion.

The Review

The structure of the paper is very good, with a coherent and solid flow from related work to design considerations (well connected to previous work) to implementation (well connected to design considerations). This emphasizes a sound and thoughtful design process. The authors took great care in preparing the accompanying video and the detailed supplemental material.

The implemented system, NOAH, is convincing and its design is well justified. Overall, the implementation makes sense and aligns with the goal of facilitating navigating and steering in spreadsheets.

I have little comments on the first part of the paper, which I find very well executed. The comments I have are:
- What about DC2 in the implementation section?
- I wonder whether the authors considered more direct interactions for some of the operations. For example, one could use Embedded Merge & Split technique to facilitate merging and splitting bins in histograms [A].
- the authors use words like "natural", "intuitive" and "cumbersome" quite a bit - removing these would make writing more scientific at places (as these are kind of assumed common sense from the authors but not necessarily backed up by previous research).

The rationale for the study seems good. The inclusion criteria of being close to expert in Excel is also compelling. I think that the choice the authors make to focus on qualitative explanations (even in the analysis of the quantitative results) of why the differences exist rather than the quantitative comparisons is good. The explanation of the results leads to some interesting findings regarding how people use a tool such as NOAH.
I have some concerns with the study - both minor concerns and more major concerns.

The minor concerns are as follows:
- I would have liked to see more explanations regarding how the four research questions were selected. Why these ones? Are there others that were considered?
- The input method for participants to specify their answers is not described. Not that it is a big problem, because task completion time is several dozens of seconds, but still it should be specified.
- I think the fact that participants subjectively indicated preferring NOAH over Excel should be tone-downed a bit. Because it was clear to them what the experimenter's solution is (NOAH), they were largely sensitive to bias in assessing the two different conditions.

The (more) major concerns are as follows:
- Given the within-subject design, and given that the authors assume that the two datasets lead to comparable results, I was expecting to see intra-participant differences being reported. This means subtracting the mean completion time for participant P for task T with Excel from the mean completion time for participant P for task T with NOAH. Plotting this information would reveal differences that are less affected by between-participant differences. Still, this is not a deal-breaker as the authors report the results separately for each dataset.
- I find the choice of datasets a bit limiting. If the real challenge is the difficulty to navigate in very large datasets (the authors mention several millions of cells in the paper), then why selecting datasets with only 14 or 15 columns? This easily fits on a single screen and only requires scrolling vertically. This should at least be discussed in the discussion (the experiment targets scalability in terms of rows, but not in terms of columns).

Other notes:
- The authors cheated a bit too obviously with the template (for example, removed a lot of spacing between the Figures).

[A] A. Sarvghad, B. Saket, A. Endert and N. Weibel, "Embedded Merge & Split: Visual Adjustment of Data Grouping," in IEEE Transactions on Visualization and Computer Graphics, vol. 25, no. 1, pp. 800-809, Jan. 2019. doi: 10.1109/TVCG.2018.2865075

External Reviewer 2 review
score 3.5/5

Paper type

Algorithm / Technique

Expertise

Expert

Overall Rating

<b>3.5 - Between Possible Accept and Accept</b><br/>

Supplemental Materials

Acceptable

Justification

++ tight coupling between side by side overview and details for spreadsheets (which appears to be novel in the case of spreadsheets)
++ important topic because so many people use spreadsheet
+ fairly simple and useful
+ well explained related work (both for industry and research), requirements and rationale
+ User study shows benefits of NOAH over Excel spreadsheet
+ clear writing in general, with useful video demo

- - User study does not compare with state of the art of overview + details tool in spreadsheet (e.g. Subtotal does that according to the related work)
- - the actual steering function(s) used to focus the detail view is not described precisely enough
- drawbacks of the systematic binning are not discussed
- - User study had no task that actually used the details part of overview+details (while this seems to be a main contribution, so this was surprising).

The Review

Additional comments:

The title says “At Scale” but there is no discussion of scalability. Does it work with the 2M cells mentioned in the intro? No test with different sizes of large datasets. Nothing is done or said to convince me that the testing warrants saying “at scale” in the title of the paper.
REQUIRED: remove or justify.

Abstract is fine.

We usually talk about scrolling and zooming for navigation. Is steering the same as zooming? Or is steering Sorting and/or Filtering? Something else? The paper needs to clarify what operations are actually included in the “steering”.

I see that the implementation is done only in Dataspread but that is fine, and completely fair for a research tool.

I found the “Ashville-Boston bin” confusing…. Why were several cities binned together by default?
Isn’t it a problem to spot outliers? (e.g. when in a bin one city is the high value a outlier and another a very low outlier).   If all the bars are visible anyway why bin them by default i.e. at the start?
REQUIRED: Clarify the binning by default of categorical variables.

The contextual and historical information are not particularly novel but are appropriately included (and authors should be commended for doing so).   Can the paper clarify how the history is kept? E.g. keeping the complete state info or recompute from start? Any challenges worth reporting?

I am not sure I really understand the difference with SUBTOTAL.   Not clear…
The paper says: “the size of the summary itself can become very large and span multiple screens, e.g., for numeric data. “ is it because there is no binning? Or because too many attributes are summarized in extra columns with a visual representation (and it add too many columns)?   Not clear.   Can the paper also clarify what level of coordination there is in Subtotal? In short: is the difference a fundamental difference or only a superficial layout one?
REQUIRED: Clarify difference with Subtotal.

In 5.1.1
The paper says that clicking is not filtering but what is it? A novel sort?
Describe the algorithm used E.g. When it says “as the user clicks on the bin, the data corresponding to Boston is displayed”: what really is done on the table side? We do not know what the original sort order was. Then what was done? I imagine that it is probably is a series of actions: Select all Boston rows, Move selection at the top, Keep the rest below in original order. Or it may be something else…
For example if there is a “filter by > 60 days” created then sorting the detail view by the # of days may be useful and increase the sense of coordination between overview and details.
REQUIRED: clarify operations done in the detail view

In general:
I would like to see at least one example of the use of a continuous attribute e.g. numerical or date .
Aren't they some drawbacks in allowing only a single overview attribute? E..g doesn't it make it harder to see correlations between attributes?

User study:
It looks like the user study had no task using the details part of the overview+details (while coordination is a main contribution). Strange.

While I understand that it is easier to compare a new tool to Excel I wondered why it was not compared with Subtotal, which seems to be a close state of the art. In the current study we have so many things changing between the 2 interfaces that we will never know what really helped the most: it is the presence of an overview? or the bin mechanism which reduces overview scrolling? or fact that only a single overview can be seen at once?, or the tight coupling between details and overview? I cannot tell…

If all the tasks "only" involve counts (and no need to see the details) then a better candidate for comparison may be Keshif, where (if I remember correctly) all count distributions are shown by default for all attributes making easy to find counts, min and max. Comparisons are possible between 2 selections.

Reconsidering the study is not feasible in time I assume, but may be choices could be better justified?

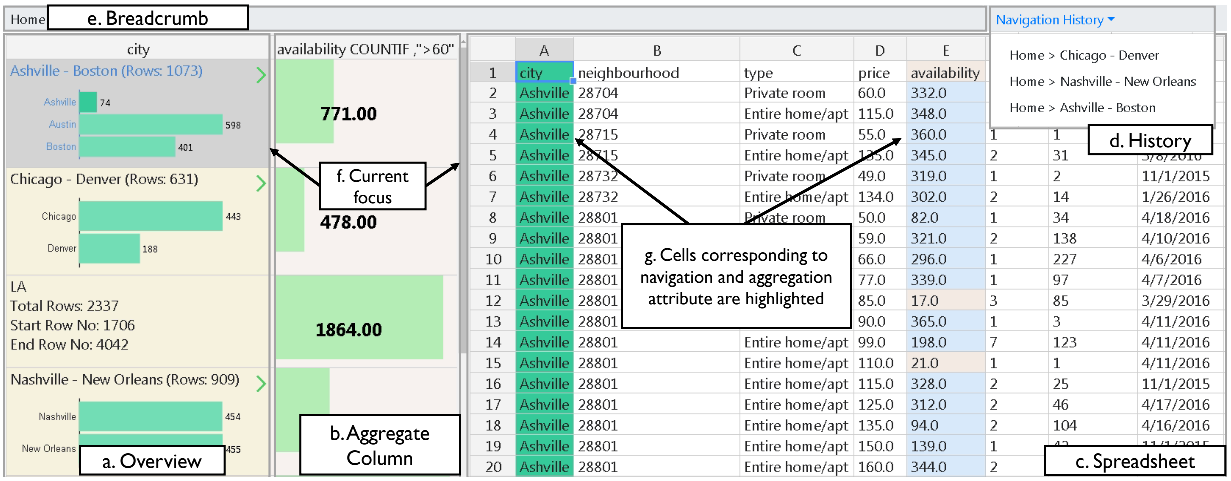
- smaller things:
In the text be careful to clarify when things happen in the overview or the details, especially since the overview also have columns… E.g. when the paper say that users can view results on a separate aggregate Column" I could not tell what part of the interface the column was added to (until I saw the video).

I really do not understand the benefit of binning cities i.e. categorical variables… The individual bars for each city are shown anyway so adding binning in the overview only increases the length of the overview (and the need for scrolling) so why not just show a bar chart with one bar for each city first, and then allow binning only as needed?

INFOVIS Paper

Extending Spreadsheets to Support Seamless Navigation at Scale

Sajjadur Rahman, Mangesh Bendre, Pingjing Yang, Yuyang Liu, Shichu Zhu, Zhaoyuan Su,
Kevin Chen-Chuan Chang, Karrie Karahalios, and Aditya Parameswaran



	A	B	C	D	E			
1	city	neighbourhood	type	price	availability			
2	Ashville	28704	Private room	60.0	332.0			
3	Ashville	28704	Entire home/apt	115.0	348.0			
4	Ashville	28715	Private room	55.0	360.0	1	1	
5	Ashville	28715	Entire home/apt	135.0	345.0	2	31	[illegible]
6	Ashville	28732	Private room	49.0	319.0	1	2	11/1/2015
7	Ashville	28732	Entire home/apt	134.0	302.0	2	14	1/26/2016
8	Ashville	28801	Private room	50.0	82.0	1	34	4/18/2016
9	Ashville	28801		59.0	321.0	2	138	4/10/2016
10	Ashville	28801		66.0	296.0	1	227	4/6/2016
11	Ashville	28801		77.0	339.0	1	97	4/7/2016
12	Ashville	28801		85.0	17.0	3	85	3/29/2016
13	Ashville	28801		90.0	365.0	1	3	4/11/2016
14	Ashville	28801	Entire home/apt	99.0	198.0	7	123	4/11/2016
15	Ashville	28801	Entire home/apt	110.0	21.0	1	1	4/11/2016
16	Ashville	28801	Entire home/apt	115.0	328.0	2	25	11/1/2015
17	Ashville	28801	Entire home/apt	125.0	312.0	2	46	4/17/2016
18	Ashville	28801	Entire home/apt	135.0	94.0	2	104	4/16/2016
19	Ashville	28801	Entire home/apt	150.0	139.0	1	[illegible]	[illegible]
20	Ashville	28801	Entire home/apt	160.0	344.0	2		

Fig. 1. NOAH: navigation interface consisting of (a) a zoomable overview and (b) an aggregate column integrated with (c) a spreadsheet. A conext bar consisting of (d) navigation history displaying locations visited so far using the overview, and (e) breadcrumb showing the current navigation path. (f) User's current focus in the spreadsheet is highlighted on the overview. (g) Columns corresponding to the navigation attribute (city) and aggregate column (availability) are highlighted on the spreadsheet.

Abstract— Spreadsheets are one of the most popular tools for ad-hoc exploration and analysis of data. Despite that, exploring and analyzing spreadsheet datasets that span more than a few screens via traditional operations, *e.g.*, scrolling, issuing formulae, can be overwhelming for end-users. Users easily lose context as they explore the data via scrolling and suffer from cognitive and mechanical burdens while issuing formulae on data spanning multiple screens. We propose integrating a navigation plug-in with spreadsheets to support the seamless exploration of large datasets that are increasingly the norm. Our interface, NOAH, developed using lessons from classical overview+detail interfaces, embeds a multi-granularity zoomable overview alongside the spreadsheet. Users can employ the overview to explore the data at various granularities and issue formulae over subsets of data without performing cumbersome scrolling or range selection operations, enabling users to gain a high-level or low-level perspective of the spreadsheet data. NOAH preserves the spreadsheet semantics, and look and feel while introducing such enhancements. Our user study demonstrates that NOAH makes it more intuitive, easier, and faster to navigate spreadsheet data compared to traditional spreadsheets like Microsoft Excel, for a variety of navigational tasks; participants made 2.5× fewer mistakes in NOAH than in Excel while being twice as fast in completing the tasks.

Index Terms—Spreadsheet navigation, overview+detail, zooming, comprehension

1 Introduction

With a user base of more than one-tenth of the world's population, spreadsheets are by far the most popular medium for ad-hoc exploration and analysis of data by end-users [7]. Spreadsheets enable users to view, structure, and present data in an intuitive tabular layout, wherein users can map their data and tasks; this tabular layout is essential to the popularity of spreadsheets [39].

Using this tabular layout effectively involves navigation, *i.e.*, *"the process of viewing and manipulating the computer display to show another portion of the information space"* [16]. Navigation is supported via two unit operations, scrolling and steering. Scrolling involves panning the spreadsheet to view other portions of the sheet on the computer screen. For example, when analyzing data, users may scroll to compare data across different screens, or to get a high-level view of the overall spreadsheet. Steering, on the other hand, involves dragging the mouse pointer through the spreadsheet spanning multiple screens. For example, to issue a formula, users may steer to select the subset of the data within the spreadsheet to be operated on as an argument within the formula. Most frequently used spreadsheet formulae require users to perform steering actions [21, 36]. Overall, both scrolling and steering are crucial as users navigate spreadsheets to identify, compare, and summarize data.

However, navigating spreadsheet data using scrolling or steering is often challenging, even for moderately sized spreadsheets that only span a few screens [39, 57], for multiple inter-related reasons:

- *Loss of overview and context.* When navigating spreadsheets, users can easily lose the context of where they are and where they should go next [57]. The only navigational context provided by spreadsheets is the built-in scrollbar that acts as a one-dimensional overview and indicates the user's current location on the sheet. However, since this overview does not capture the layout and structure of the data, users are forced to mentally assimilate the layout as they navigate a spreadsheet.
- *Cognitive and mechanical burdens.* The lack of contextual cues leads to severe cognitive and mechanical burdens for users [28]. Users often end up taking their own drastic measures to avoid getting lost; for example, they sometimes create their own overview

- *All the authors are with the Department of Computer Science at University of Illinois, Urbana-Champaign. Email: {srahman7, bendre1, py2, yuyangl2, szhu28, zsu7, kcchang, kkarahal, adityagp}@illinois.edu.*

by sketching maps of spreadsheets on paper [57]. Other users add their own landmarks, *e.g.*, headers or colored cells, as a visual affordance to assist in navigation [57]. Steering across multiple screenfuls to select a subset of data as input to a formula can often be challenging as well: the only remedy is for users to abandon steering entirely and instead remember the range of the subset of data of interest, and then correctly enter this range as the argument to the formula, often giving rise to errors that are increasingly prevalent in spreadsheets [44].

- *Visual discontinuities.* The limited viewport afforded to the user introduces a visual discontinuity between the information being displayed. For example, comparing spatially separated subsets of data within the spreadsheet requires moving back and forth between multiple viewports, which can be overwhelming. As an alternative, users tend to copy subsets of data side by side to reduce the visual discontinuity [39, 57], which is cumbersome.

Overall, while navigating present-day spreadsheets, ***users often lose context, get overwhelmed, and experience visual discontinuities, addressing all of which requires considerable manual effort***. As we argue in Section 3, existing spreadsheet functionalities such as pivot tables, named ranges, and subtotals, while providing partial support for some of these challenges (*e.g.*, pivot table generates a summary while losing the mapping between the raw data and the summary; named ranges still require users to manually associate names with ranges of data), do not eliminate them entirely. With the ease of generation of data, and with spreadsheets now supporting data of increasingly larger scales—*e.g.*, Google Sheets now supports 2M cells, a five-fold increase from the previous limit of 400K cells—navigating data within spreadsheets is only going to get harder, not easier.

So, how do we support more effective navigation of data within spreadsheets? One approach would be to try to integrate an overview of the overall structure of the data along with the spreadsheet [31]. The resulting integration would then be a classical overview+detail interface where the spreadsheet is the detailed view. Overview+detail interfaces are used to facilitate navigation in various domains such as text editors and maps [28], with users being able to manipulate the overview or detailed view, to perform high-level or low-level operations respectively. Overview+detail interfaces have been shown to be effective in these domains, reducing cognitive load for users by providing the big picture first, helping them quickly assimilate the information space [28]. Thus, in this paper, our goal is to extend existing spreadsheets with an overview plug-in, giving an overview+detail interface.

While an overview plug-in for spreadsheets does seem appealing and natural, developing it leads to a host of challenges.

- *Overview modality.* One approach for an overview is to simply add a zoomed out version of the entire sheet as a pane on the side, as in popular presentation software like Microsoft PowerPoint or text editors like Sublime Text. This approach would not suffice for a spreadsheet because while it would help users make more informed scrolling decisions, it would otherwise be unreadable. Another approach, adopted by map tools like the early versions of Google maps, is to use the overview to provide context for the user's current location in the detailed view [28]. Users can perform zooming operations on the detailed view as the overview remains static. Since spreadsheets already display the raw data, zooming into and out of a detailed view consisting of this raw data is not meaningful. Instead, the goal of our spreadsheet overview is to provide a big picture view of the spreadsheet, capturing the data distribution and summarizing it, while enabling meaningful interactions for navigating the data. This necessitates *semantic zooming* [45] instead of geometric zooming (unlike text editors or presentation software), on the overview view instead of the detailed view (unlike map tools). Several domains, *e.g.*, online maps and program visualization, adopt semantic zooming to ensure improved readability and comprehension of the information displayed [55]. However, to the best of our knowledge, no prior work has ever developed semantic zooming for spreadsheet data.
- *Construction of the overview.* Given a spreadsheet with many rows, it is not clear how to employ semantic zooming to construct the overview. How do we group rows together in a "meaningful" way such that this grouping applies to all data types, including strings and numbers? Since we want to allow users to control how the overview is constructed, we can let them select the grouping attribute(s). Following this, we need mechanisms to automatically group nearby rows into meaningful bins. Further, the bins need to be constructed at various granularities, allowing users to zoom in and zoom out of the overview. We also need to be able to allow the user to customize the bins to construct meaningful categories, while providing interpretable and intelligent defaults. Customizing these bins will also eliminate visual discontinuities: users can simply arrange the bins the way they want and keep them near each other for easy comparison.
- *Operations on the overview.* While the scrolling issues are partially addressed by having the overview, the steering issues remain unaddressed. How do we leverage the overview to allow users to select regions of interest for issuing formulae without having to steer across multiple screens? As the granularity of the overview changes (during zoom in/out) how do we ensure that users don't have to reissue the formula from scratch? For this, we can take inspiration from aggregation in relational databases: in relational databases, each "group" is associated with one or more aggregates, so as long as the groups are appropriately constructed (using the mechanisms alluded to above), formulae can be issued once without requiring any steering, and automatically maintained for all groups during zoom in and zoom out.
- *Seamless integration as a plug-in.* Finally, the design of the overview should be generic so that it can be integrated with any existing spreadsheet tool on any spreadsheet dataset without impacting existing functionalities or look-and-feel. Any new interactions introduced should be consistent with traditional spreadsheet semantics, and complement existing spreadsheet interactions. Interactions between the overview and the raw spreadsheet should also be coordinated to the extent possible.

We address the aforementioned challenges in our tool NOAH, an in-situ **n**avigation interface for **o**verviewing and **a**nalyzing spreadsheet data **h**olistically. NOAH is constructed as a plugin to an existing spreadsheet tool, DATASPREAD [3], an open-source scalable web-based spreadsheet. (While NOAH's design is not tied to DATASPREAD, we opted not to use other popular spreadsheet tools like Google Sheets and Microsoft Excel because they are closed source.) Figure 1 shows a snapshot of NOAH. When the user chooses to explore the data by a specific attribute, an overview is constructed and displayed within NOAH, next to the raw spreadsheet data (Figure 1a). The overview is essentially a histogram and depicts the overall data distribution for the chosen attribute, enabling users to quickly assimilate the data. NOAH ensures that there is coordination between the overview and the spreadsheet: for example, panning and zooming on the overview are reflected on the spreadsheet by displaying the spreadsheet data corresponding to the bin currently in focus in the overview. Cumbersome scrolling operations are eliminated in favor of a few clicks of the overview interface. Instead of steering to analyze the data, users can issue aggregation operations on the overview and view results on a separate *aggregate column*, alongside the overview (Figure 1b). In this manner, users can issue formulae on different subsets of the data while remaining on the same screen, reducing visual discontinuity. Finally, NOAH automatically creates contextual and historical information (Figure 1d and 1e) while displaying visual cues (Figure 1f and 1g) so that users don't lose context during navigation.

The primary contribution of our work is twofold:

- We formalize the design of a general navigation (overview+detail) interface for exploration and analysis of large spreadsheets. We realize this design in the form of NOAH, a plugin to an existing web-based spreadsheet tool, ensuring that interactions supported by NOAH complements existing spreadsheet operations.
- We conduct a user study to evaluate the benefits and limitations of this plugin. The study shows that spreadsheet tasks involving

navigation that are hard to do correctly or complete in a timely manner in Microsoft Excel were easy to complete correctly and quickly in NOAH with participants making 2.5× **fewer** mistakes in **NOAH** while being 2× **faster.**

2 Case Study: Investigative Reporting of Airbnb Listings

To illustrate the benefits of NOAH, consider a scenario where Rebecca, an investigative journalist is exploring the Inside Airbnb dataset [8], a dataset of all the Airbnb listings across different US cities. This dataset was created to investigate the long-standing accusation that many listings in Airbnb are illegally run as hotel businesses, while avoiding taxes; any listing available for rent for more than 60 days a year is considered to be operated as a hotel [1].

Without any prior knowledge of the dataset, Rebecca starts by finding out information about cities in the dataset and their corresponding listings; for example, which cities are present, and roughly how many listings does each city have. Without NOAH, Rebecca may have used a pivot table (discussed in Section 3) to construct a summary—however, since this summary is disconnected from the underlying data, it is hard for Rebecca to map the summary statistics to the raw data to obtain further details about listings from any given city. If she wanted to identify listings from a specific city, Rebecca would have to either use search capabilities or perform an explicit filter for this information, and would have to switch back and forth between the pivot table results and the raw listings, present at disparate locations on the spreadsheet. Even at the first step of this exploration, Rebecca would experience substantial cognitive burden, loss of context, and visual discontinuities, with subsequent steps becoming progressively more challenging.

Using NOAH, she simply explores the data by city, with NOAH providing a high-level overview of cities (Figure 1a). The overview consists of sorted non-overlapping bins containing one or more cities. She can click on any bin and the corresponding data will be displayed at the top of her screen. For example, clicking on the *Ashville-Boston* bin displays the Ashville listings (Figure 1c). She can also zoom into bins using the "⟩" arrows, zoom out of bins using the "⟨" arrows, and pan by clicking on various bins at the same level (Figure 2). We discuss the construction of the overview and associated interactions in Section 5.

Next, say Rebecca wants to analyze one of the larger cities. She identifies these cities by comparing the number of listings for each city displayed on the overview. She decides to focus on Boston, her hometown, and wants to find out how many listings in Boston violate the "rent availability > 60 days" condition. To do so, she zooms into the *Ashville-Boston* bin (Figure 2a and 2b) and then issues an aggregation operation on the overview, *i.e.*, a COUNTIF formula[1]—note that the equivalent steering operation in a traditional spreadsheet would be cumbersome as Rebecca would have to manually select all the Boston listings as input to the COUNTIF formula. The result is displayed as an *aggregate column* alongside the overview (Figure 1b). The rows that satisfied the condition in the COUNTIF formula, *i.e.*, the listings that serve as hotels, are listed in the spreadsheet adjacent to the overview in "sky blue" (Figure 1g), enabling easy comparison on other attributes. Rebecca learns that more than half of the listings in Boston are effectively operating as hotels—a large number!

Based on this insight, Rebecca then wants to understand availability statistics for an even larger city, Chicago. As she uses the overview to navigate to Chicago, NOAH automatically updates the aggregate column to the COUNTIF formula results for Chicago, without having Rebecca reissue it by performing a cumbersome steering operation as in traditional spreadsheets. Rebecca learns that Chicago also exhibits a similar renting pattern as Boston, with more than half the listings operating as hotels. She can then check if this trend holds for all large cities, and whether the smaller cities have a different pattern. With the raw data presented side-by-side, she can also dive into other attributes of the listings operating as hotels to see if there are any other identifying characteristics, *e.g.*, if they are all managed by a small number of agencies acting as individual renters.

Finally, as Rebecca navigates the data, her navigation history (Figure 1d), *i.e.*, recently visited cities, and current navigation path (Figure 1e) are kept up-to-date, allowing her to not lose context during navigation. She can revisit any previously visited cities by simply clicking on the relevant path in the navigation history.

Overall, with NOAH, users can quickly comprehend the data via the overview, access any region within the data without having to scroll endlessly, and request additional details on demand without having to steer across multiple screens. As users navigate and analyze the data, they can revisit previously accessed data via the navigation history, not losing context of what they have explored.

3 Related Work

In this section, we present tools and techniques that partially address the limitations of navigating spreadsheets as discussed in the introduction.

3.1 Existing Spreadsheet Tools

We first discuss features supported by commercial spreadsheet tools and other academic prototypes.

3.1.1 Commercial Spreadsheet Tools

Commercial spreadsheet tools include Microsoft Excel, Google Sheets, and Airtable—of these, Excel and Google Sheets support features targeted at improving users' navigation experience.

Microsoft Excel. Excel allows users to maintain context during navigation by manually creating named ranges [4], that are essentially references to a spreadsheet region, and appear in a *name box* within the Excel menu bar. Users can click on a named range to navigate to the referred region. However, the onus is on the user to create named ranges for each region of interest.

Excel and other spreadsheet tools support pivot tables [10], helping summarize data via aggregation. The summarized view enables users to compare subsets of data without having to navigate to various locations within the spreadsheet, partially addressing the visual discontinuity problem. This summary is not presented in-situ with the raw spreadsheet data and is placed in a separate region, preventing users from accessing the data underlying the summary, impeding navigation.

Excel provides an alternative to pivot tables—SUBTOTAL [11], which adds a new row at the end of the corresponding subset of data with summary information. Initially, SUBTOTAL only displays the summary while hiding the data. Users can expand the summary to view the actual spreadsheet data. However, as the number of subsets within the data increases, the size of the summary itself can become very large and span multiple screens, *e.g.*, for numeric data. Exploring the summary therefore requires manual effort—the increased visual discontinuity places cognitive load on users as they compare the data.

Finally, NodeXL [32] is an Excel plug-in that provides an overview of a network and allows users to perform navigational operations, *e.g.*, zooming in/out, dynamic filtering, on the overview; this plugin only supports network datasets, such as biological or social networks.

Google Sheets. Google Sheets Explore [5] provides an overview of the data by auto-generating charts of data statistics. Users can specify queries to the system (similar to a web search) asking for different summary statistics. While Explore is a convenient means to understand high-level data characteristics, it doesn't address the navigational challenges related to scrolling and steering.

3.1.2 Academic Prototypes

Existing academic work focuses on adding a richer set of operations to spreadsheets, or increasing their scalability. We discuss work specifically targeting navigation; other enhancements such as managing hierarchical data [23] or supporting joins [15], are omitted.

Interactive Tables. TableLens [49] is a focus+context view for browsing numerical information in tables, looking much like a spreadsheet with embedded bar charts. Cells out of focus display graphical bars proportional in length to the underlying values, providing a visual overview of the data, while cells within the user's current focus are magnified and display the graphical bars and the raw data. Ideas similar to TableLens

[1] COUNTIF is a type of formula in spreadsheets that counts the number of rows that satisfy some condition.

have been adopted by DataLens [18] for visualizing digital calendars, and by FOCUS [54] and InfoZoom [53] for exploring database query results. Like TableLens, NOAH embeds graphical bars, but within the overview to depict the underlying data distribution. NOAH captures the user's current focus by highlighting the corresponding bin in the overview. While TableLens provides an easy mechanism to get a high-level view of the data and spot outliers, it suffers from the same disadvantages that focus+context views have relative to overview+detail ones. Unlike NOAH, which supports multiple granularities via binning, TableLens only supports one granularity (zoomed in or zoomed out): beyond a certain size, navigating (scrolling or steering) the zoomed out data is still cumbersome for users. TableLens does not support non-numeric data, nor does it maintain the spreadsheet look-and-feel or capabilities (maintained in NOAH due to the detailed view).

Visual Interactive Spreadsheets. VisSh [43], SI [37], SSR [26], ASP [46], and PhotoSpread [34] extend the input/output capabilities of cells within spreadsheets, to display charts, animation, photos, or geometric objects, or accept input via direct manipulation dialogs, among others. While these tools allow users to represent and manipulate data in a more flexible manner, which in turn could help users getting a high-level sense of the data, they do not necessarily help users navigate data more effectively.

Scalable Spreadsheet Summarization and Exploration. Smart-drill-down [33] generates an interactive summary of a large spreadsheet table as a collection of rules; users can drill-down to a specific rule to view more fine-grained rules. Hillview [6] displays the results of group-by queries on large spreadsheet tables. While these tools support summarization at scale, providing an overview of the spreadsheet, they don't preserve spreadsheet semantics, nor do they make it easy to scroll or steer through large spreadsheets. ABC [48] and DATASPREAD [19] support interactive exploration of very large spreadsheet datasets, beyond main-memory limits, maintaining spreadsheet look-and-feel, but do not provide any new spreadsheet capabilities to assist with navigation. We build NOAH as a plugin to DATASPREAD, since it is open-source.

3.2 Spreadsheet Alternatives

Finally, we discuss solutions to navigational and summarization challenges in other domains.

Overview+Detail Interfaces. Cockburn et al. [28] provides a detailed survey of overview+detail and zooming interfaces. To improve navigation within large documents, overview+detail interfaces [27, 35] allow users to interact with an overview as they explore the document. Zooming interfaces [40, 47] provide a multi-granularity overview of the data and support interactions like zoom in/out to navigate across various granularities. We follow the same analogy of providing an overview of the spreadsheet first, allowing users to drill-down further.

Multiple Coordinated Views. Multiple coordinated views [50], *e.g.*, *Snap* [41], *Elastic Documents* [14] connect multiple views, for example, an overview and a detailed data view while enabling coordination between these views through brushing and linking. Similarly, NOAH connects tabular spreadsheets with an overview and updates the state of the spreadsheet as users interact with the overview and vice-versa.

Tabular Data Analysis. Tabular data visualization tools [9, 12, 30, 58, 59], business intelligence tools [9, 42, 51], and relational databases can all provide summaries of data in various forms (visualizations, aggregate statistics, SQL query results) that can serve as overviews. Much like pivot tables, these summaries are not dynamically linked to nor are co-located with the underlying data, introducing visual discontinuities and lack of context.

4 NOAH: Design Considerations

In this section, we outline our design considerations for a spreadsheet navigation interface. In our supplementary material, we additionally characterize the use-cases that such a navigation interface is good for and not good for. Our design considerations were informed by prior work on information visualization [22, 52], overview+detail interfaces [28], and multiple-coordinated views [56], and refined through our experiences across multiple design iterations.

DC1. Construct the overview in-situ. An overview helps users get a high-level picture of the data. However, maintaining the overview in a separate location from the data can be cumbersome; instead, having it be co-located with the data can help users make rapid glances to consult information between a bird's-eye view and a close-up detail [31].

DC2. Ensure increased visual continuity while providing details on demand. Users often need to access subsets of data, and study their properties in detail (*e.g.*, via steering); navigating back and forth can lead to decreased visual continuity. The interface should allow users to compute such details for various data subsets on demand [52]. The interface should maintain visual continuity as users navigate to a different subset, recomputing the details for the new subset.

DC3. Balance the screen space afforded to the overview. As the overview has limited screen-space available, the interface needs to take the trade-off between visual continuity (*DC2*) and clarity into account: Displaying a fine-grained overview reduces visual continuity as users need to scroll through the overview to access distant subsets of data. Displaying a coarse-grained overview reduces visual clarity as it may span too many data subsets, increasing clutter. The interface should also allow users to control the screen-space allocated to the overview.

DC4. Enable coordination between the spreadsheet and overview. Since users can view the overview and the spreadsheet simultaneously, interactions on both need to be coordinated [50], *i.e.*, an interaction on one should be reflected on the other interface [56]. For example, as a user scrolls through the spreadsheet, the user's current focus should be highlighted on the overview. However, not all interactions need to be interlinked, *e.g.*, changing the font size of a spreadsheet cell need not lead to a change in the overview.

DC5. Facilitate customization of the overview. As the overview is automatically generated, it may not reflect domain-specific context known only to the user [48]. For example, an overview constructed on a grading spreadsheet by binning nearby scores may not match the letter grade ranges that the instructors have in mind. Allowing users to customize the overview is therefore essential.

DC6. Display contextual and historical information regarding users' navigation. The interface should record navigation history, allowing users to revisit previously visited locations [52], while also displaying their current navigation path for context.

5 User Interface

We now present NOAH, a navigation interface for spreadsheet data; we explain the design of NOAH's components and its implementation.

5.1 In-situ Overview

NOAH constructs the overview in-situ (**DC1**) next to the spreadsheet on an attribute of the spreadsheet dataset called the *navigation attribute*, selected by the user. Any attribute type that can be ordered can be the navigation attribute: this includes text, numbers, or dates. The overview is constructed at multiple granularities. Each granularity is divided into non-overlapping groups of data called *bins*. As shown in Figure 2d, an overview of the Airbnb data on the navigation attribute "city" has two levels of granularities. The highest level of the granularity consists of four bins. Figure 2a depicts NOAH showing the first four bins, the first of which is *Ash-Bos*. At each granularity, the bins act as landmarks on the overview and allow users to quickly navigate to the desired subset of data by skipping over irrelevant bins. Each bin contains summary information regarding the data subset/region it spans, *e.g.*, starting row and ending row number, the total number of rows the region spans. Each bin displays an overview of the next (finer) granularity (if any) with embedded bar charts. For example, in Figure 2d, the topmost bin (*Ash-Bos*) spans three cities (*Ashville*, *Austin*, *Boston*), each of which is a bin in the next (finer) granularity. Correspondingly, Figure 2(a) shows three horizontal bar charts for the first *Ash-Bos* bin, one for each bin in the next granularity. On the other hand, since the third bin from the top (*LA*) spans only one city, no bar chart is embedded. Users can perform different operations on the bins, *e.g.*, clicking to pan and semantic zooming in/out [45]. NOAH supports other interactions, *e.g.*, customization and aggregation. We discuss these interactions in the context of our design considerations later. We now explain the design of the overview.

Why a Multi-granularity Overview? A standard approach for an overview is to show a spatially partitioned collection of thumbnails, similar to Microsoft PowerPoint or Adobe Reader. However, displaying too many thumbnails results in increased scrolling to access distant thumbnails, increasing visual discontinuity. On the other hand, displaying too few thumbnails reduces visual discontinuity, but at the cost of visual clarity—the thumbnails appear cluttered and fail to represent the underlying data clearly [28]. To strike a balance between these two objectives (**DC3**) we designed a multi-granularity overview that abstracts the data at varying levels of detail. Multi-granularity interfaces provide an alternative to a single-granularity spatial partition of the data space, by varying the scale at which the data is displayed, enabling multiple views at varying granularities [28]. As explained earlier, we divide each granularity of the overview into bins. The highest (coarsest) level of the granularity provides a global view with less visual clarity and reduced visual discontinuity (with the former increasing and latter decreasing as the bins become more fine-grained). Users can resize the overview to control the amount of spreadsheet data that remains visible. Moreover, users can also hide the overview if required.

Overview Construction. The data structure underlying the overview is an equi-depth histogram constructed on the values in the navigation attribute column. Equi-depth histograms are commonly used for summarizing the statistical properties of data, with applications such as database query optimizers [24]. Each bin in the equi-depth histogram contains the same number of items, where each item is a value. For example, when constructing the overview on city, each value in the city column is assigned to a bin. The bins are constructed top-down (see Figure 2d)—NOAH divides each of the bins at level k into new bins to construct the next lower level $k+1$, again, by applying the same concept of equi-depth histograms. If each value of the navigation attribute column was unique, *e.g.*, if it was a numerical ID, then construction of the histogram is easy: each bin of the equi-depth histogram would contain almost the same number of items, where each item corresponds to one unique value of the attribute. Unfortunately, in practice, for many attributes, the same value is often repeated. For example, there are multiple listings per city. Therefore, an equi-depth histogram on the attribute city will result in consecutive bins sharing items of the same unique city value, resulting in undesirable overlap. Instead, we construct a best effort equi-depth histogram that is as close to an equi-depth histogram as possible, while ensuring that the ranges represented by each bin have no overlap.

5.1.1 Operations and Interactions

Users can navigate the spreadsheet by interacting with the bins of the overview, or by adding aggregate columns.

Navigational Operation: Clicking. When a user clicks on a specific bin, NOAH displays the corresponding spreadsheet data; users can use this to jump to a specific spreadsheet location without having to scroll endlessly. For example, in Figure 2b, as the user clicks on the *Boston* bin, the data corresponding to Boston is displayed (Figure 2c). Note that the click operation is different from the traditional spreadsheet Filter operation. Filter hides spreadsheet data that do not satisfy the filtering condition while clicking brings the desired subset of data in view without hiding the rest. Users are free to navigate to other portions through scrolling even after clicking a bin, unlike filtering, where users need to issue another Filter to bring other data into view.

Navigational Operation: Semantic Zooming. Users can zoom into a specific bin to view more fine-grained information or zoom out to view a more coarse-grained information, via semantic zooming [45]. For example, in Figure 2a, from the bin *Ashville-Boston* when the user zooms in to the next level, NOAH displays the bins *Ashville*, *Austin*, and *Boston* (Figure 2b). If the user zooms out of the current granularity, again NOAH displays the bins *Ashville-Boston*, *Chicago-Denver*, and others. Users can only zoom into any bin that contains multiple unique values. For example, in Figure 2d, at level 2, each bin corresponds to one city. Therefore, users can only click on those bins to bring that

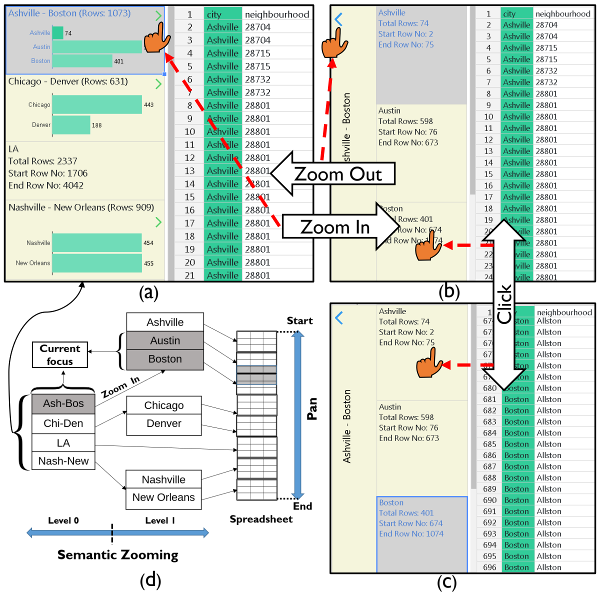


Fig. 2. Navigational operations. (a) The overview at the highest level of granularity. (b) A zoomed in view of the *Ashville-Boston* bin. (c) As the user clicks on the *Boston* bin, the Boston listings are displayed on the sheet. The *Boston* bin is highlighted in gray to indicate user's current focus. (d) Conceptualizing the multi-granularity overview.

data into view, and cannot zoom in further. One issue with zooming interactions is discoverability [28]. We circumvent this (see Figure 2c) by providing the root of the bin under selection for zoom out, and arrows for clicking to zoom in ("⟩") and out ("⟨").

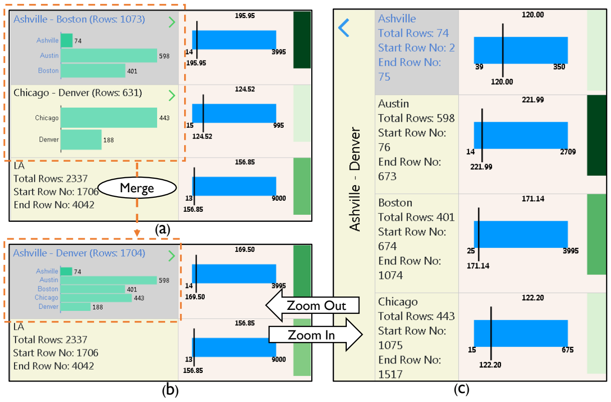


Fig. 3. (a) Chart view of the aggregate column. (b) A new bin is created by merging the top two bins. (c) Zooming into the newly created bin.

Customizing the Overview. As NOAH constructs the overview automatically, the overview binning or organization may not capture domain-specific context or user needs. NOAH enables users to customize this organization (**DC5**). At any granularity, users can merge multiple consecutive bins into a single bin, or split a bin into multiple bins. Say the user wants to compare summary statistics of Boston and Chicago. In the current organization these two cities are in two different bins (see Figure 3a). Using the bin customization feature, the user can merge the two bins *Ashville-Boston* and *Chicago-Denver* which creates a new bin *Ashville-Denver*. Users can now zoom into this bin and compare summary statistics of the cities in the same view. The interactions for splitting a bin depend on the data type. If the navigation attribute is textual, any bin can be split into as many bins as the number of unique values that bin contains. If the navigation attribute is numeric, users can split the bin into any arbitrary number of bins. Note that NOAH does not allow users to rearrange the order of the bins. Since the overview represents a histogram, the bins are ordered—reshuffling the bins violates that order.

5.1.2 Coordination Between Overview and Spreadsheet

NOAH supports coordination between the overview and the corresponding spreadsheet data (**DC4**), *i.e.*, an interaction on the overview may be reflected on the spreadsheet and vice-versa. Indicating the navigation attribute on the spreadsheet using lime green color (see Figure 1c) as user constructs the overview, is an example of such coordination. However, not all overview interactions are coupled with the spreadsheet and vice versa. The coupling depends on the user's current focus—*any interaction on either the overview or the spreadsheet that changes the focus must be reflected on the other interface*. We now provide examples of both coupled and decoupled interactions.

Coupled interactions. Clicking a bin is an example of a coupled interaction as the user actively changes the focus to another bin on the overview. To reflect the change, NOAH populates the corresponding spreadsheet data on the screen. As the user scrolls on the spreadsheet, again the current focus changes and the corresponding bin on the overview is highlighted. For example, in Figure 2c, as the user clicks on the Boston bin, the spreadsheet displays the Boston listings. Conversely, as the user scrolls up, both Austin and Boston listings appear in the current window of the spreadsheet. Therefore, both the Austin and Boston overview bins are highlighted (see Figure 2d). The zoom in operation, on the other hand, can be either coupled or decoupled based on the setting. Whenever a user zooms into a bin that is not the current focus, the spreadsheet is updated to show data for that bin.

Decoupled interactions. When a user zooms into a bin that is already in the user's current focus, the spreadsheet view does not change. For example, in Figure 2a the user zooms into the *Ashville-Boston* bin; here, the spreadsheet view stays the same (see Figure 2b). Similarly, the zoom out operation is decoupled. When a user zooms out, the overview displays a coarser granularity view of the user's current focus. Since the focus stays the same, there's no need to update the spreadsheet view. Similarly, operations like panning on the overview without clicking and customizing the overview do not change user's current focus and are decoupled. Online maps also adopt similar slight decoupling of the overview and detail [28]. However, their goal is to reduce network and computational overload whereas in our case, the decoupling is based on user's current focus.

5.2 Aggregate Columns

Users can issue spreadsheet formulae on the overview to compute aggregates for the data in each bin. The results are displayed as an *aggregate column* (see Figure 1b). Each entry in the aggregate column corresponds to the adjacent bin in the current granularity of the overview. For example, in Figure 3c, the aggregate column displays four aggregate statistics, one per bin. Users can issue several formulae simultaneously, each giving rise to a new aggregate column. However, adding an aggregate column takes up screen space thus shrinking the spreadsheet. As a workaround, users can resize or remove aggregate columns to make more screen space available to the spreadsheet (**DC3**). When user issues a formula on the overview, the spreadsheet column corresponding to the aggregate column is highlighted in grayish orange (see Figure 1c)—another example of coupled interaction. For conditional formulae like COUNTIF, cells that satisfy the condition are highlighted, *e.g.*, in Figure 1c), the cells with availability ≥ 60 are colored in sky blue. By viewing the colored cells, users can find out which cells are relevant to the aggregation operation.

Creating an aggregate column on the overview is done similarly to how users create pivot tables—users are not required to explicitly type formulae, rather simply select the formula from a drop-down menu, and provide the necessary formula parameters to a web form. This way, NOAH maintains the semantics of existing spreadsheet tools. Currently, NOAH supports statistical and mathematical formulae that operate over a range of data. Therefore, creating an aggregate column is equivalent to selecting a subset of data on the sheet, *i.e.*, steering and then executing a formula, helping users avoid cumbersome steering operations. We have classified the formulae supported into five categories: a) summary (*e.g.*, min, max, average), b) frequency (*e.g.*, mode, large, small), c) conditional (*e.g.*, countif, sumif), d) spread (*e.g.*, var,stdev), and e) others (*e.g.*, sum, count).

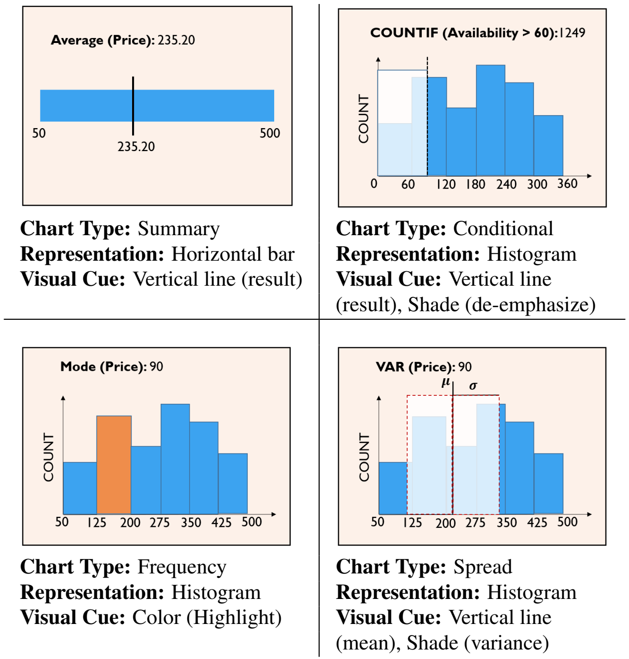


Chart Type: Summary
Representation: Horizontal bar
Visual Cue: Vertical line (result)

Chart Type: Conditional
Representation: Histogram
Visual Cue: Vertical line (result), Shade (de-emphasize)

Chart Type: Frequency
Representation: Histogram
Visual Cue: Color (Highlight)

Chart Type: Spread
Representation: Histogram
Visual Cue: Vertical line (mean), Shade (variance)

Fig. 4. Formula types and their chart representations.

Users can choose to view the result of formulae either as raw values or charts and can toggle between the two. Raw values are displayed along with a colored bar called *value bar* whose length is proportional to the corresponding aggregate (see Figure 1b); users can use the lengths to visually compare across bins. The chart representation varies depending on the formula type. Except for the *others* category all other categories can be represented by charts. Figure 4 shows the chart representation for these categories along with different visual cues that highlight formula results as well as other information. We discuss these representations in detail in the supplementary material.

Finally, we note that the aggregate column is kept in sync with the bins as users zoom in and out, eliminating repeated steering operations. NOAH does not maintain any additional data structure for the aggregate column. The histogram underlying the overview records the result of the aggregate column entries corresponding to the bins. Next, we discuss how NOAH maintains user's navigational context.

5.3 Context Bar

The context bar consists of two components: a) a breadcrumb, and b) a navigation history. The breadcrumb [2] displays the current navigation path (see Figure 1e), thus maintaining users' navigation context (**DC6**). Each component of the breadcrumb corresponds to a bin in the user's current navigation path. Therefore, users can visit any bins within the current navigation path by clicking on an appropriate component of the breadcrumb, without having to zoom in or zoom out. NOAH also maintains a list of recently visited bins (**DC6**) (see Figure 1d).

5.4 Implementation

We have integrated NOAH with DATASPREAD [19], a web-based spreadsheet. The DATASPREAD back-end maintains the histogram data structure and supports the aggregate column computation via its built-in formula engine. The NOAH front-end is built with HTML/CSS/JS technologies along with the D3 framework [20] for generating charts.

6 Evaluation Study Design

In this section, we present the design of a user study to evaluate whether NOAH helps address spreadsheet navigational challenges.

6.1 Study Design and Participants

Existing evaluation studies on overview+detail, zooming, and context+focus interfaces can be divided into two categories [28]: (a) low level and (b) high level evaluation. Low level evaluations focus on the performance of a system with or without the presence of enhancements, *e.g.*, overview or zooming. High level evaluations study the performance of a system for a specific task domain, *e.g.*, database browsing [41] or tree navigation [17]. Since we are interested in evaluating whether NOAH improves navigation for spreadsheets, we perform a similar high-level evaluation where we compare a NOAH-integrated spreadsheet with a traditional one, Excel, for various navigational tasks. Our study was designed to answer the following questions:

- **RQ1.** How does the addition of an overview help address spreadsheet navigational challenges, with respect to quantitative performance metrics such as speed and accuracy?
- **RQ2.** How do users make use of the aggregate column results in conjunction with the raw spreadsheet data to analyze data?
- **RQ3.** How does bin customization help users inject domain knowledge in addressing their specific exploration questions compared to the automated overview?
- **RQ4.** How is the user's subjective satisfaction with NOAH compared to traditional spreadsheets?

Study Design. We conducted a 2×2 (2 datasets, 2 tools) mixed design within-subject study. We explain the datasets next. The two tools used in the study were: Microsoft Excel, and NOAH integrated within DATASPREAD. The study consisted of three phases: (a) an introductory phase explaining the essential features of NOAH via a video tutorial, followed by a warm-up session where participants explored a flight dataset [13] in NOAH to familiarize themselves with its features, (b) a quiz phase where the participants first used both the tools to perform targeted tasks on two different datasets followed by completing one survey for each tool to provide feedback on their impressions about the tool, and (c) a semi-structured interview to collect qualitative data regarding the quiz phase.

Datasets. We used two datasets—birdstrikes (used in Keshif [59] and Voyager [58]), and airbnb [8]. These datasets were chosen for their real-world interest to a general audience. The birdstrikes dataset records instances of birds hitting aeroplanes in different US states. The dataset has 10,868 records and 14 attributes (8 categorical, 1 spatial region, 1 temporal, 4 numeric). To ensure a fair comparison across tools, we created a sampled version of the original airbnb dataset with 10,925 records, by uniformly sampling 10% of the records from each US city. This dataset contains 15 attributes (6 categorical, 2 spatial region, 1 temporal, 6 numeric).

Participants. We recruited 20 participants (11 female, 9 male) via flyers across the university and an university email newsletter. The average age of the participants was 31.06 years ($\sigma = 12.44$). The participants came from different backgrounds, *e.g.*, engineering (seven), business (five), administration (five). During recruitment, prospective participants filled out an interest form where we asked them questions about their spreadsheet expertise, types of spreadsheet tasks performed, and usage of spreadsheet operations. Participants were asked to rate their expertise with different spreadsheet software, *e.g.*, Excel and Google Sheets, and their frequency of using various spreadsheet tasks *e.g.*, data management, data analysis, statistical modeling, and what-if analysis. We also asked participants about their familiarity with basic mathematical and statistical spreadsheet functions as well as advanced operations, *e.g.*, pivot table, SUBTOTAL, and conditional formatting. To ensure that participants' experience with spreadsheets didn't affect their performance during the quiz phase, we only recruited participants who rated their experience with Excel to be greater than four (on a scale of one to five, with five being the highest). The selected participants were familiar with performing various tasks on spreadsheets, *e.g.*, maintaining, tracking, and analyzing, making predictions, and performing comparisons. All of the participants were familiar with the basic mathematical and statistical functions supported by Excel. Each participant received $10 per hour at the end of their session.

6.2 Study Procedure

We now explain each of the phases of our study in more detail.

Phase 1: Introduction to NOAH. We began the study by showing a six-minute video tutorial explaining the features of NOAH on a dataset of all the flights across the US for January 2018 [13]. The participants then explored the same dataset using NOAH to familiarize themselves with the tool for about 10 minutes. The quiz phase began as soon as the participants finished their exploration.

Table 1. Quiz tasks for the birdstrikes dataset.

Category	Question
Steer	Organize the data by State. How many flights that had damages (damage = 1) originated from Florida?
Identify	How many flights in the currently visible spreadsheet window have damages?
Steer	How many flights that had damages (damage = 1) originated from California?
Compare (2)	Which state between Florida and California has higher number of flights with damages?
Compare (*N*)	Find the state with the most birdstrike occurrences.
Customize	Organize the data by Altitude. What is the average cost of damages for altitude bin 0-450?

Phase 2: The Quiz Phase. During the quiz phase, each participant performed specific tasks on the two datasets in two sessions, using Excel for one and NOAH for the other. Each session was followed by a survey, described later. The order of the datasets was alternated between consecutive participants. The order of the tools was alternated between every two participants. We developed an online JavaScript-based quiz system that recorded users' responses as well as submission times. We also recorded the participants' interactions with both tools using screen capture software. Participants were also informed that they can refer to the Internet for help as many times as they wanted. However, due to their familiarity with Excel, none of the participants needed external help. We also provided a printed handout to the participants that contained screenshots with the features of NOAH.

Quiz Tasks. The quiz tasks were designed to answer the first three research questions. We designed six tasks grouped into four categories: steer (two tasks), compare (two tasks), identify (one task), and customize (one task). While steer and compare tasks aim to address *RQ1*, identify and customize tasks address *RQ2* and *RQ3*, respectively, in addition to *RQ1*. The questions were presented in an increasing order of difficulty. We included two types of compare tasks: Compare (2) and Compare (*N*). Compare (2) required participants to compare the statistics of two specific subsets of data, whereas Compare (*N*) involved comparing statistics of all the data subsets. Table 1 lists the questions for birdstrikes. The questions for airbnb were similar.

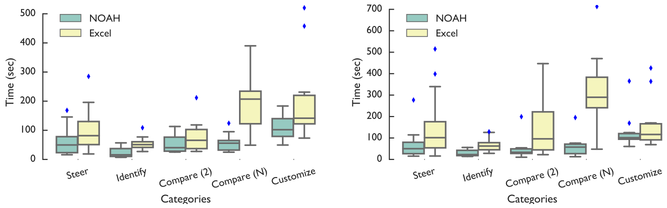
Survey. After each session, participants rated the corresponding tool used on six metrics: confidence, comprehensibility, level of satisfaction, ease and speed of use, and ease of learning for navigating spreadsheets, on a Likert scale from 1 (*e.g.*, strongly disagree) to 7 (*e.g.*, strongly agree). The survey asked multiple questions related to these metrics, 15 in total, to ensure reliability. Participants were also asked to mention the positive and negative aspects of both tools. The survey was designed to evaluate *RQ4*.

Evaluation. We evaluated the accuracy (either 0 or 1) and completion time for each of the six tasks. Moreover, for each quiz task category, we analyzed the video recordings to collect various information to better explain the results, *e.g.*, participants' task strategies (features used) and causes of task delay and failure. Moreover, we aggregated the survey responses for each of the aforementioned criteria to quantify the usability of both the tools.

Phase 3: Interview Phase. Following the survey, we conducted a semi-structured interview to identify participants' preferred tools for different tasks and understand the reasoning behind their choices. We also asked participants to comment on the usefulness of different features provided by NOAH and Excel.

7 Results

In this section, we analyze the quantitative and qualitative data collected during the quiz and interview phases to address the research questions in Section 6.1. We first discuss the results of the quiz phase tasks. We then present participants' experiences in using NOAH and Excel based on the interview and survey responses.



(a) birdstrikes **(b) airbnb**

Fig. 5. Submission times per category for each dataset. Median submission times are much smaller for NOAH compared to Excel.

7.1 Analysis of the Quiz Phase

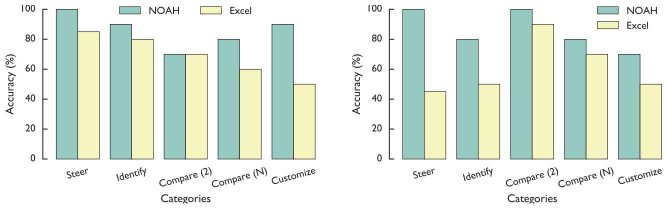
We first compare the participants' task completion time and accuracy for NOAH with Excel.

7.1.1 Comparison of Task Completion Time and Accuracy (*RQ1*)

Distribution of Completion Times. In Figure 5a and 5b, we show the distribution of submission times of participants for the four task categories, for birdstrikes and airbnb respectively. For most categories, *participants' median submission time for* NOAH *was less than the fastest submission time for Excel*, confirming that the new capabilities offered by NOAH makes spreadsheet navigation faster.

Accuracy Distribution. In Figure 6a and 6b, we show the percentage of correct submissions for the four task categories, for birdstrikes and airbnb respectively. For all the tasks except for Compare (2)—for which the accuracy was the same for both tools—participants attained slightly higher accuracy with NOAH compared to Excel.

Since we conducted the study on a small population (20), we further evaluated the statistical significance of the results—see our supplementary material. For all tasks except customize, the difference in completion times was statistically significant. However, the difference in accuracies was statistically significant for the steer tasks only.



(a) birdstrikes **(b) airbnb**

Fig. 6. Per category accuracy for each dataset. Participants attained higher accuracy while completing tasks in NOAH compared to Excel.

7.1.2 Steer Results (*RQ1*)

The steer tasks required participants to issue a COUNTIF formula on a data subset. However, in Excel, the completion time and accuracy for this differed significantly from one dataset to another (see Figure 5 and 6). We found that the data type of the columns on which steering was to be performed for the two datasets—binary (birdstrikes) as opposed to non-binary (airbnb)—contributed to the deviation.

Data type impacts task performance in Excel. Analysis of video recordings revealed that for birdstrikes, several participants used the *autosum* feature to quickly count the number of 1s in the column—summing up binary values is equal to the number of 1s in the collection. Other participants used the status bar at the bottom of the spreadsheet that displays the sum of the cells in a selected column. In both cases, participants avoided steering the data. On the other hand, for airbnb, participants could not use these shortcuts as the column was non-binary (it had 365 different values) and ended up steering the data, resulting in higher completion times. Therefore, the participants' ability to avoid steering depended on the data type. Failure to avoid steering led to participants often selecting an incorrect range of data ($N = 14$), resulting in incorrect responses.The errors mostly occurred for airbnb ($N = 11$) where participants were not able to avoid steering.

Data-independent navigation in NOAH. *Using* NOAH, *participants were able to avoid steering by issuing an aggregate formula on the overview, resulting in faster submission times as well as higher accuracy.* Irrespective of the data type, participants used the same strategy resulting in consistent performance—the submission time for both datasets with NOAH is similar (see Figure 5).

7.1.3 Compare Results (*RQ1*)

Each comparison task in Excel required users to steer and then compare the results. For Compare (N) tasks in Excel, participants had to perform N comparisons, each time repeating similar tasks, *e.g.*, steering, or autosum. As a result, Compare (N) had higher submission times than Compare (2).

Repetitive tasks avoided in NOAH. The completion times were faster in NOAH than in Excel, irrespective of the comparison tasks and datasets. NOAH *offers three additional benefits apart from avoidance of steering that may have contributed to such improvement*: a) participants used the navigation features to access and compare different subsets of data quickly, b) participants did not need to reissue the aggregation formula for any of the bins they navigated to, and c) participants used the value bars presented along with the results in the aggregate column to visually compare different subsets.

Visual discontinuity impacts comparisons. For Compare (N) tasks, the number of subsets to be compared was higher in birdstikes (50) compared to airbnb (16). As a result, participants exhibited lower accuracy for birdstrikes when using Excel. In NOAH, all the participants first split all the bins to create N bins each corresponding to one state. Then participants panned across the overview to find the desired bin as they compared the values. Even in NOAH, *comparisons between multiple values resulted in increased visual discontinuity leading to some ($N = 4$ out of 20) incorrect submissions.*

7.1.4 Identify Results (*RQ1, RQ2*)

Coloring cues help accelerate identification in NOAH. For the *identify* task, participants had to skim through all the cells in the current window in Excel, resulting in higher completion times. Again, participants had to skim over binary values when counting in birdstrikes versus non-binary values for airbnb, resulting in even higher completion times for the latter. In NOAH, participants benefited from having visual cues in the form of colored cells, helping them relate the aggregate column with the raw data: *they quickly identified the spreadsheet cells that did not satisfy the condition based on cell color*.

7.1.5 Customize Results (*RQ1, RQ3*)

Bin customization is powerful but time-consuming. For airbnb in Excel, the customize task involved filtering out 26 values from the filter menu compared to filtering 451 values for the birdstrikes dataset. As a result, participants had to manually filter a large number of values and took more time to submit their responses. Participants took much less time in NOAH, as they were able to use bin customization. However, the time taken for this task was higher than other tasks in NOAH, as it required participants to restructure the overview before any calculation could be performed. Unfamiliarity with customization operations also contributed to higher task completion times (see Section 8).

7.2 Qualitative Observations

We now present a qualitative analysis of the participants' perceptions of both tools based on their interview responses.

Data comprehension easier in NOAH. Without an overview, participants found it difficult to perform various tasks on Excel—*Excel can get overwhelming if you have a lot of data in it and sometimes with that data finding things can be difficult (P11).* The binned representation of the data in NOAH, on the other hand, helped users to comprehend the overall structure of the data better and prioritize the bin they want to visit next, enabling faster navigation. Out of 20 participants, 15 preferred NOAH for tasks involving navigation. One participant ($P5$) commented: *I think it was just a little bit easier to navigate and find where things were because you could already see what bins had what.* Another participant ($P1$) commented: *I like* NOAH *a lot better. It was a lot easier to look up different data and it was a lot quicker too*. Participants ($N = 6$) additionally mentioned that they would prefer

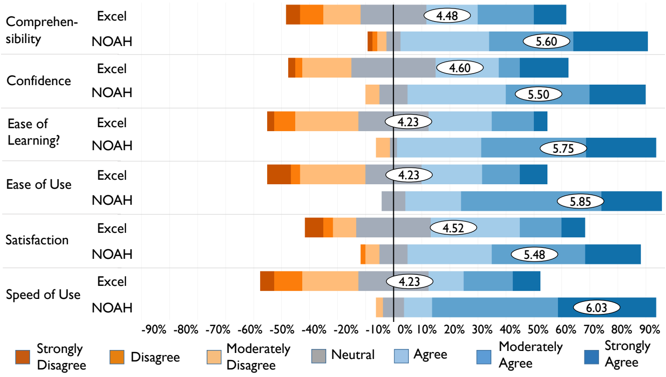


Fig. 7. Participants found NOAH to be easier to use compared to Excel while being faster in completing tasks involving navigation.

NOAH over Excel when the dataset is large. One participant (*P*2) commented: *I guess if I just had a large amount of data then I would prefer to use* NOAH *because then you would be able to see all of it ...at once.* Finally, participants preferred the automatic data highlighting feature while performing the *identify* task (16 out of 20). The colored cells focused their attention to relevant spreadsheet regions—*It was a visual cue right there, made it very quick to count it up (P*17*).* However, one participant (*P*5) mentioned the fact that some of the color choices would make NOAH unusable for the colorblind population.

Scrolling and steering is cumbersome in Excel. Participants found scrolling and steering in Excel to be cumbersome while issuing formulae and performing comparisons—*The one thing with Excel is I always try to go to the bottom of the data and type in the formula, and with something really long like this, the scrolling is a little bit cumbersome (P*4*).* With NOAH users can (a) avoid scrolling by using clicking or zooming operations, and (b) avoid steering by performing aggregate operations on the overview. One participant (*P*3) commented: *And that creates convenience sort of because then you don't have to memorize anything and using the system becomes easier.* Another participant commented: *With* NOAH*, you don't have to highlight every number ... versus Excel you actually have to select everything (P*12*).*

Fine-grained control is powerful but sometimes difficult to use. The bin customization feature enables users to personalize the overview based on their specific needs. One participant (*P*16) commented: *I did like the fact that it lets you take a data sheet and, in some way, containerize the stuff you care and the stuff you don't care about.* 14 out of 20 participants found the bin customization feature to be useful. Participants preferred the feature to Excel's filtering feature when working with numeric data—*That was so much easier in* NOAH *than it was in Excel to be able to specify the range that you wanted it to go in (P*17*).* However, six participants found the feature to be difficult to use when working with textual data and would have preferred a pivot table-like single level overview—*I would have expected them to completely be split, and then I can merge them if I want to (P*13*).*

Participants preferred NOAH to Excel on all subjective user satisfaction metrics (*RQ4*) Figure 7 displays the subjective participant ratings for various satisfaction metrics using a diverging stacked bar chart. For a given metric, each white ellipse denotes the average rating given by the participants for the corresponding tool. We can see from the figure that participants preferred NOAH to Excel for all metrics. Notably, participants found NOAH to be easier to use compared to Excel, while being faster in execution. Even though the features offered by NOAH were new to participants they found it fairly easy to learn.

8 Discussion

In this section, we discuss the shortcomings of our user study, the limitations of NOAH as a spreadsheet plug-in for navigation, and possible enhancements.

Limitations of Study Design. Our evaluation compared a NOAH-integrated spreadsheet with a traditional spreadsheet. However, it is certainly possible to evaluate intermediate versions of NOAH that isolate the effects of the zoomable overview (visual clarity vs. visual continuity), display layout (screen space trade-off), and contextual presentation of data (raw text vs. chart representation of aggregate columns). Moreover, our participant pool demographics may not match the demographics of the general audience intended for this tool.

Unfamiliarity with NOAH. Several interactions on NOAH are quite different from typical spreadsheet interactions, *e.g.*, zooming, bin customization. This unfamiliarity led to some participants ($N = 5$ out of 20) preferring Excel over NOAH. Participants found some of the terminologies used in the interface—*e.g.*, explore, bin—to be quite unfamiliar ($N = 14$) for an Excel audience. Moreover, two participants didn't understand how the bins were constructed. These issues can be addressed by using more relatable terminologies and proper documentation. Two participants found the aggregation operations applied on the bins to be opaque compared to Excel where a user can directly manipulate the formula. In subsequent versions of NOAH, we can further display the appropriate formula for each bin as users hover over the corresponding cell on the aggregate column. Moreover, participants ($N = 5$) noted the fact that NOAH currently does not support user defined formulae.

Beyond Tabular Data. NOAH operates only on tabular data. However, spreadsheets can be semi-structured—formulae and text can be interspersed with tabular data. We can extend NOAH to act as a map highlighting heterogeneous regions on such complex spreadsheets which users can utilize to navigate the spreadsheet. We can leverage existing work on spreadsheet table detection [29], and property identification [25] to construct the map.

Scope of Overview-Spreadsheet Coordination. Spreadsheet users may perform various edit operations, *e.g.*, updating values, adding/deleting rows/columns. However, NOAH currently assumes the data to be read-only. In our next version, we can add support for propagating the spreadsheet updates to the overview. Moreover, the charts displayed in an *aggregate column* are non-interactive, *i.e.*, users cannot interact with the charts to visually look up relevant or interesting data points within the spreadsheet. In the future, we plan to extend NOAH to support visual querying through the charts in an aggregate column.

From Perceptual to Interactive Scalability. The current version of NOAH addresses the perceptual scalability challenges while navigating Excel-scale (one million rows) data. As modern spreadsheets continue to support even more large scale data—DATASPREAD [19] supports one billion rows—the interactions proposed in this paper may violate the interactive response time bound of 500 ms [38]. This opens the door to a new set of research challenges that may range from approximate query processing to progressive data analytics.

Other Enhancements. NOAH currently constructs the overview on a single attribute. We can add support for multi-attribute navigation (*e.g.*, explore the Airbnb data by city and neighborhood), and multi-level navigation (*e.g.*, explore the neighborhoods after zooming into a specific city in the Airbnb data). Furthermore, bin customization currently supports changing the bin boundaries only while maintaining the current order. Supporting user defined ordering to allow arbitrary reshuffling of the bins can be another enhancement. Finally, a complete characterization of the spreadsheet operations that can be supported by NOAH is another open question, *e.g.*, other operations that involve working with subsets of spreadsheet data, *e.g.*, sorting, filtering, copy-pasting.

9 Conclusions

We presented NOAH, an in-situ navigation interface, designed as a spreadsheet plugin.. Using NOAH, users can get a bird's eye view of the data, with the ability to scroll or seek additional details on demand, using a multi-granularity overview, as well as aggregate columns that eliminate cumbersome steering operations. We conducted a user study that demonstrated that participants found NOAH to be much more easy to use compared to Excel for navigating large spreadsheet datasets. Across different navigation tasks, participants were able to attain higher accuracy when using NOAH compared to Excel while being twice as fast. NOAH represents our first step towards making spreadsheets more effective on large datasets that are increasingly the norm.

References

[1] A Lone Data Whiz Is Fighting Airbnb—and Winning. https://www.wired.com/2017/02/a-lone-data-whiz-is-fighting-airbnb-and-winning/.

[2] A Lone Data Whiz Is Fighting Airbnb—and Winning. https://en.wikipedia.org/wiki/Breadcrumb_(navigation).

[3] DataSpread. https://dataspread.github.io/.

[4] Define and use names in formulas. https://www.excel-easy.com/examples/names-in-formulas.html.

[5] Google explore. https://www.techrepublic.com/article/how-to-use-the-new-google-explore-feature/.

[6] Hillview. https://github.com/vmware/hillview.

[7] How finance leaders can drive performance. https://enterprise.microsoft.com/en-gb/articles/roles/finance-leader/how-finance-leaders-can-drive-performance/.

[8] The inside airbnb dataset. http://insideairbnb.com/get-the-data.html.

[9] Microsoft power bi. https://powerbi.microsoft.com/en-us/.

[10] Pivot Table. https://en.wikipedia.org/wiki/Pivot_table.

[11] SUBTOTAL. https://exceljet.net/excel-functions/excel-subtotal-function.

[12] Tableau software. https://www.tableau.com/.

[13] The us flight dataset. https://www.transtats.bts.gov/.

[14] S. K. Badam, Z. Liu, and N. Elmqvist. Elastic documents: Coupling text and tables through contextual visualizations for enhanced document reading. *IEEE transactions on visualization and computer graphics*, 2018.

[15] E. Bakke, D. Karger, and R. Miller. A spreadsheet-based user interface for managing plural relationships in structured data. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pp. 2541–2550. ACM, 2011.

[16] D. V. Beard and J. Q. W. II. Navigational techniques to improve the display of large two-dimensional spaces. *Behaviour & Information Technology*, 9(6):451–466, 1990.

[17] D. V. Beard and J. Q. W. II. Navigational techniques to improve the display of large two-dimensional spaces. *Behaviour & Information Technology*, 9(6):451–466, 1990.

[18] B. B. Bederson, A. Clamage, M. P. Czerwinski, and G. G. Robertson. Datelens: A fisheye calendar interface for pdas. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 11(1):90–119, 2004.

[19] M. Bendre et al. Towards a holistic integration of spreadsheets with databases: A scalable storage engine for presentational data management. In *ICDE*, April 2018.

[20] M. Bostock, V. Ogievetsky, and J. Heer. D^3 data-driven documents. *IEEE transactions on visualization and computer graphics*, 17(12):2301–2309, 2011.

[21] D. A. Bradbard, C. Alvis, and R. Morris. Spreadsheet usage by management accountants: An exploratory study. *Journal of Accounting Education*, 32(4):24–30, 2014.

[22] M. Brehmer and T. Munzner. A multi-level typology of abstract visualization tasks. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2376–2385, 2013.

[23] K. S.-P. Chang and B. A. Myers. Using and exploring hierarchical data in spreadsheets. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pp. 2497–2507. ACM, 2016.

[24] S. Chaudhuri, R. Motwani, and V. Narasayya. Random sampling for histogram construction: How much is enough? *ACM SIGMOD Record*, 27(2):436–447, 1998.

[25] Z. Chen, S. Dadiomov, R. Wesley, G. Xiao, D. Cory, M. Cafarella, and J. Mackinlay. Spreadsheet property detection with rule-assisted active learning. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pp. 999–1008. ACM, 2017.

[26] E.-H. Chi, P. Barry, J. Riedl, and J. Konstan. A spreadsheet approach to information visualization. In *Information Visualization, 1997. Proceedings., IEEE Symposium on*, pp. 17–24. IEEE, 1997.

[27] A. Cockburn, C. Gutwin, and J. Alexander. Faster document navigation with space-filling thumbnails. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, pp. 1–10. ACM, 2006.

[28] A. Cockburn, A. Karlson, and B. B. Bederson. A review of overview+ detail, zooming, and focus+ context interfaces. *ACM Computing Surveys (CSUR)*, 41(1):2, 2009.

[29] H. Dong, S. Liu, S. Han, Z. Fu, and D. Zhang. Tablesense: Spreadsheet table detection with convolutional neural networks. In *Thirty-Third AAAI Conference on Artificial Intelligence*, 2019.

[30] S. Gratzl, A. Lex, N. Gehlenborg, H. Pfister, and M. Streit. Lineup: Visual analysis of multi-attribute rankings. *IEEE transactions on visualization and computer graphics*, 19(12):2277–2286, 2013.

[31] J. Grudin. Partitioning digital worlds: focal and peripheral awareness in multiple monitor use. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 458–465. ACM, 2001.

[32] D. L. Hansen, B. Shneiderman, and M. A. Smith. *Analyzing social media networks with NodeXL: Insights from a connected world*. Morgan Kaufmann, 2010.

[33] M. Joglekar, H. Garcia-Molina, and A. Parameswaran. Smart drill-down: A new data exploration operator. *Proceedings of the VLDB Endowment*, 8(12):1928–1931, 2015.

[34] S. Kandel, A. Paepcke, M. Theobald, H. Garcia-Molina, and E. Abelson. Photospread: a spreadsheet for managing photos. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pp. 1749–1758. ACM, 2008.

[35] S. Kratz, I. Brodien, and M. Rohs. Semi-automatic zooming for mobile map navigation. In *Proceedings of the 12th international conference on Human computer interaction with mobile devices and services*, pp. 63–72. ACM, 2010.

[36] B. R. Lawson, K. R. Baker, S. G. Powell, and L. Foster-Johnson. A comparison of spreadsheet users with different levels of experience. *Omega*, 37(3):579–590, 2009.

[37] M. Levoy. Spreadsheets for images. In *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pp. 139–146. ACM, 1994.

[38] Z. Liu et al. The effects of interactive latency on exploratory visual analysis. *IEEE TVCG*, 20(12):2122–2131, 2014.

[39] B. A. Nardi and J. R. Miller. *The spreadsheet interface: A basis for end user programming*. Hewlett-Packard Laboratories, 1990.

[40] D. Nekrasovski, A. Bodnar, J. McGrenere, F. Guimbretière, and T. Munzner. An evaluation of pan & zoom and rubber sheet navigation with and without an overview. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, pp. 11–20. ACM, 2006.

[41] C. North and B. Shneiderman. Snap-together visualization: can users construct and operate coordinated visualizations? *International Journal of Human-Computer Studies*, 53(5):715–739, 2000.

[42] M. J. Norušis. *The SPSS guide to data analysis*. Spss, 1986.

[43] F. Nunez and E. Blake. Vissh: A data visualisation spreadsheet. In *Data Visualization 2000*, pp. 209–218. Springer, 2000.

[44] R. R. Panko. What we know about spreadsheet errors. *Journal of Organizational and End User Computing (JOEUC)*, 10(2):15–21, 1998.

[45] K. Perlin and D. Fox. Pad: an alternative approach to the computer interface. In *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, pp. 57–64. ACM, 1993.

[46] K. W. Piersol. Object-oriented spreadsheets: the analytic spreadsheet package. In *ACM Sigplan Notices*, vol. 21, pp. 385–390. ACM, 1986.

[47] C. Plaisant, J. Grosjean, and B. B. Bederson. Spacetree: Supporting exploration in large node link tree, design evolution and empirical evaluation. In *IEEE Symposium on Information Visualization, 2002. INFOVIS 2002.*, pp. 57–64. IEEE, 2002.

[48] V. Raman et al. Scalable spreadsheets for interactive data analysis. In *ACM SIGMOD Workshop on DMKD*, 1999.

[49] R. Rao and S. K. Card. The table lens: merging graphical and symbolic representations in an interactive focus+ context visualization for tabular information. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 318–322. ACM, 1994.

[50] J. C. Roberts. State of the art: Coordinated & multiple views in exploratory visualization. In *Coordinated and Multiple Views in Exploratory Visualization, 2007. CMV'07. Fifth International Conference on*, pp. 61–71. IEEE, 2007.

[51] I. SAS. Sas user's guide: statistics. *SAS Institute, Cary, NC*, 1985.

[52] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *The Craft of Information Visualization*, pp. 364–371. Elsevier, 2003.

[53] M. Spenke and C. Beilken. Visualization and analysis of formula one racing results with infozoom-the demo. In *INTERACT*, 2003.

[54] M. Spenke, C. Beilken, and T. Berlage. Focus: the interactive table for product comparison and selection. In *Proceedings of the 9th annual ACM symposium on User interface software and technology*, pp. 41–50. ACM, 1996.

[55] K. L. Summers, T. E. Goldsmith, S. Kubica, and T. P. Caudell. An experimental evaluation of continuous semantic zooming in program. In

Information Visualization, 2003. INFOVIS 2003. IEEE Symposium on, pp. 155–162. IEEE, 2003.

[56] M. Q. Wang Baldonado, A. Woodruff, and A. Kuchinsky. Guidelines for using multiple views in information visualization. In *Proceedings of the working conference on Advanced visual interfaces*, pp. 110–119. ACM, 2000.

[57] J. Watts-Perotti and D. D. Woods. How experienced users avoid getting lost in large display networks. *International Journal of Human-Computer Interaction*, 11(4):269–299, 1999.

[58] K. Wongsuphasawat, D. Moritz, A. Anand, J. Mackinlay, B. Howe, and J. Heer. Voyager: Exploratory analysis via faceted browsing of visualization recommendations. *IEEE Transactions on Visualization & Computer Graphics*, (1):1–1.

[59] M. A. Yalçın, N. Elmqvist, and B. B. Bederson. Keshif: Rapid and expressive tabular data exploration for novices. *IEEE transactions on visualization and computer graphics*, 24(8):2339–2352, 2018.