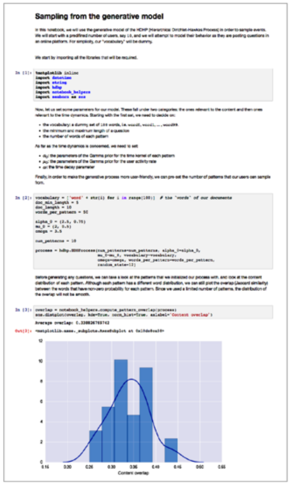
Adam Rule **Research Statement**

Individuals and organizations increasingly rely on data analysis to generate insights and make decisions in domains as diverse as healthcare, engineering, journalism, public policy, and scientific research. Yet, analyses routinely span multiple files of diverse types which can be difficult to piece together into the exact sequence of steps used to produce a result. Analysts also rarely annotate their work with decisions or reasoning, even as small changes to how data are collected, cleaned, and modeled can lead to vastly different results. If insights derived from data are to be reviewed, reused, and trusted, the process used to generate them must be tracked and communicated with greater clarity. To meet this need, ***I study how analysts use interactive documents to track, perform, and share their work.***

I am a human-computer interaction researcher (HCI) with expertise in cognitive science and medical informatics. My research aims to improve the *intelligibility* and *reproducibility* of data analyses by *developing interactive systems that encourage analytical and collaborative best practices*. I develop techniques to observe how thousands of analysts use millions of documents to perform, track, and share their work and build prototypes to test theories about how interactive documents might better support collaborative analysis. To date, my research has focused on two domains: use of computational notebooks (i.e., Jupyter Notebook) in academic research and use of electronic heath records (EHRs) in healthcare. *My work has resulted in eight major papers, won two paper awards at top HCI conferences, influenced the product roadmaps of industry leading data analysis software, and provided a corpus of over 1 million data analysis documents which has been downloaded over 200 times.*

SCAFFOLDING EXPLANATION IN COMPUTATIONAL NOTEBOOKS

One fundamental challenge of data analysis is tracking analyses in ways that both humans and computers understand. While computers are good at producing and consuming data, humans understand the world through narrative. This introduces a tension between tracking analyses through mediums which computers understand (e.g., raw data files, analysis scripts) and those that humans more easily work with (e.g., textual accounts, visualizations). Moreover, as analysts try different versions of an analysis it can be difficult, without tedious record-keeping, to track which script produced which result, and why that analytical variant was even tried in the first place. In the last decade, data analysts have started using computational notebooks, which enable analysts to mix executable code and visualizations with explanatory text in a single document, to address these issues. Notebooks aim to help analysts write *computational narratives* supporting collaborative, lucid, and reproducible analysis [1]. They are used by millions of people and have been applied to a variety of domains. *But are notebooks being used to share compelling narratives, or simply to explore data? Do analysts find it easier to collaborate and explain their work in notebooks?*



Sampling from the generative model

In this notebook, we will use the generative model of the HDHP (Hierarchical Dirichlet-Hawkes Process) in order to sample events. We will start with a predifined number of users, say 10, and we will attempt to model their behavior as they are posting questions in an online platform. For simplicity, our "vocabulary" will be dummy.

We start by importing all the libraries that will be required.

```
In [1]: %matplotlib inline
        import datetime
        import string
        import hdhp
        import notebook_helpers
        import seaborn as sns
```

Now, let us set some parameters for our model. These fall under two categories: the ones relevant to the content and then ones relevant to the time dynamics. Starting with the first set, we need to decide on:

- the vocabulary: a dummy set of 100 words, i.e. word0, word1, ... , word99.
- the minimum and maximum length of a question
- the number of words of each pattern

As far as the time dynamics is concerned, we need to set:

- α_0: the parameters of the Gamma prior for the time kernel of each pattern
- μ_0: the parameters of the Gamma prior for the user activity rate
- ω: the time decay parameter

Finally, in order to make the generative process more user-friendly, we can pre-set the number of patterns that our users can sample from.

```
In [2]: vocabulary = ['word' + str(i) for i in range(100)]  # the "words" of our documents
        doc_min_length = 5
        doc_length = 10
        words_per_pattern = 50

        alpha_0 = (2.5, 0.75)
        mu_0 = (2, 0.5)
        omega = 3.5

        num_patterns = 10

        process = hdhp.HDHProcess(num_patterns=num_patterns, alpha_0=alpha_0,
                                  mu_0=mu_0, vocabulary=vocabulary,
                                  omega=omega, words_per_pattern=words_per_pattern,
                                  random_state=12)
```

Before generating any questions, we can take a look at the patterns that we initialized our process with, and look at the content distribution of each pattern. Although each pattern has a different word distribution, we can still plot the overlap (Jaccard similarity) between the words that have non-zero probability for each pattern. Since we used a limited number of patterns, the distribution of the overlap will not be smooth.

```
In [3]: overlap = notebook_helpers.compute_pattern_overlap(process)
        sns.distplot(overlap, kde=True, norm_hist=True, axlabel='Content overlap')

        Average overlap: 0.330826769742

Out[3]: <matplotlib.axes._subplots.AxesSubplot at [illegible]>
```

Fig 1. My research has explored how analysts use computational notebooks like Jupyter Notebook to track and share data analyses.

In my PhD dissertation I analyzed use of computational notebooks at three different scales in work that won a best paper honorable mention at CHI, the top conference in HCI [2]. First, I developed a method to scrape and analyze all 1.25 million Jupyter Notebooks hosted publicly on GitHub. Second, I coded features of ~150 notebooks shared as supplementary material to academic publications. Finally, I interviewed 15 academic data analysts who used Jupyter Notebook. Together these three methods revealed a lack of explanation in computational notebooks (1 in 4 had no explanatory text) driven by data explorations which tended to produce "messy" notebooks which analysts were hesitant to clean and share. Notebooks were being used more for the way they supported iterative analysis than how they supported clear tracking and communication of process and results. Consequently, most notebooks were loose and unannotated collections of scripts that even their original authors had difficulty understanding and re-running.

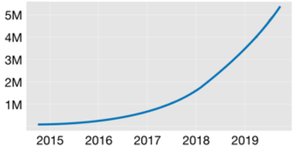


Fig 2. Publicly shared Jupyter Notebooks on GitHub over time. In 2017 I scraped and curated a dataset of all 1.25M notebooks on GitHub. This corpus has supported multiple studies by other researchers and my original analysis was recently replicated and extended by researchers at Amazon with 4M notebooks. https://github.com/jupyter-resources/notebook-research

To help reduce the *tension between data exploration and process explanation*, I developed and tested an extension to Jupyter Notebook in work that was published at CSCW, the leading HCI conference on collaborative work. [3] Through both a controlled lab study with undergraduate data science students and field deployment with academic researchers I demonstrated how simply enabling analysts to label and fold sections of their notebook aided both replication of the analysis by others and presentation of results in lab meetings. This work yielded the insight that *interfaces enabling active reading* (e.g., flipping, folding, marking up of analyses) might better support the collaborative process of data analysis.

This work also highlighted the need for *best practices for use of computational notebooks*. One interviewee noted that she received formal training during biology and chemistry labs on how to track experiments in paper notebooks, but lacked similar standards and training for computational notebooks. Working with leading educators and researchers, including one of the co-founders of Jupyter, I helped consolidate a set of best practices for conducting and sharing analyses in Jupyter Notebooks which was published in one of the top bioinformatics journals [4]. These practices encompassed both analytical best practices (e.g., how to modularize code and perform version control) and communicative ones (e.g., annotating process and not just results). Aimed at practitioners, this article was viewed more than 25,000 times in the first month after its release.

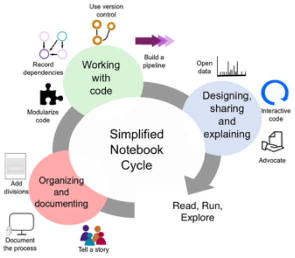
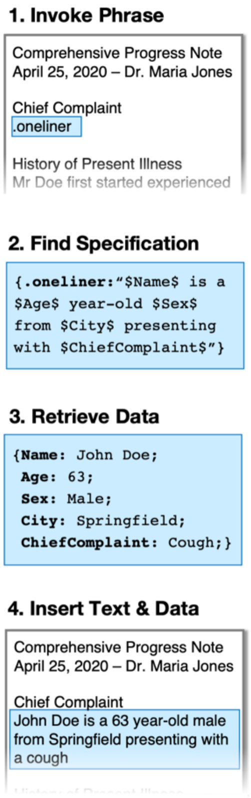


Fig 3. I helped develop best practices for the use of Jupyter Notebooks in collaboration with leading researchers and educators, including Project Jupyter co-founder Fernando Perez.

MIXING DATA, NARRATIVE, AND ACTION IN ELECTRONIC HEALTH RECORDS

While millions of analysts use computational notebooks, tens of millions of "end-user analysts" work with data without writing a line of code using systems such as spreadsheets. *How might interactive systems help these analysts work with, interpret, and communicate about data? In particular, how might notebook-like systems mixing analysis and commentary help them work more effectively?* One domain where I have begun to explore these questions is healthcare. Healthcare routinely involves the collection, analysis, and interpretation of large amounts of patient data. Moreover, team members with diverse skills and expertise need to communicate clearly about this data and its interpretation to provide and coordinate care. One of the primary tools used to do so is electronic health records (EHRs), complex software systems that support a range of healthcare activities including ordering, billing, documentation, and coordination of care. I have focused on how providers document and communicate about care using two EHR-supported formats. The first is *structured data* *fields* capturing patient information such as vital signs, problem lists, and family histories. The second is *textual notes* which providers write to summarize and interpret data, justify care plans, and coordinate care. As in other domains, healthcare workers have struggled to track and communicate how they work with data, in large part due to the recurring challenge of mixing mediums that are human and machine readable. My work has provided evidence about how providers mix structured data and narrative and how tying them more closely together might better support healthcare workflows.

My recent work has explored how providers and their staff use templates to write clinical notes. These templates enable providers to create specifications for full notes or parts of notes that mix boilerplate text and dynamically retrieved patient data. Providers can invoke these templates by typing keywords into their notes. For example, typing *.oneliner* into a note might insert the phrase "John Doe is a 63 year-old male from Springfield presenting with a cough", using a specification with embedded data-links to pull data from the patient record (e.g., "$NAME$ is a AGE year-old SEX from $CITY$ presenting with $CHIEFCOMPLAINT$"). While providers have used templates for decades, few studies have examined how they do so, especially modern templates which are highly customizable and composable. My work has begun to quantify how often providers use templates and their impact on clinical documentation. For example, in one study currently under review, [5] I found providers use templates to document the vast majority of patient visits (95%), that most of the information imported by these templates was large data tables (such as medication lists) rather than explanatory text, and that providers primarily used personal templates rather than sharing them with other providers. In a second study, I found that one consequence of this reliance on templates is highly redundant notes, with 75% of text in notes for subsequent visits by the same patient with the same provider being exactly the same [6]. This redundancy can make it difficult for providers to quickly scan a note to see what has changed in a patient's care.


1. Invoke Phrase

Comprehensive Progress Note
April 25, 2020 – Dr. Maria Jones

Chief Complaint
.oneliner

History of Present Illness
Mr Doe first started experienced

2. Find Specification

```
{.oneliner:"$Name$ is a
$Age$ year-old $Sex$
from $City$ presenting
with $ChiefComplaint$"}
```

3. Retrieve Data

```
{Name: John Doe;
 Age: 63;
 Sex: Male;
 City: Springfield;
 ChiefComplaint: Cough;}
```

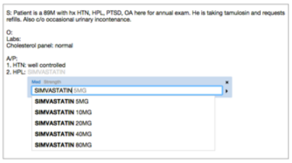
4. Insert Text & Data

Comprehensive Progress Note
April 25, 2020 – Dr. Maria Jones

Chief Complaint
John Doe is a 63 year-old male from Springfield presenting with a cough

Fig 4. In other research, I study how physicians write clinical notes using custom templates mixing boilerplate text and dynamically retrieved data. This mixing of text and data in documents by invoking commands represents one form of end-user analysis.

Providers struggle to mix text and data in effectively in their notes but have shown a willingness to invoke textual commands to construct documentation. This observation raises the question of how notebook-like interfaces might better support clinical workflows by more tightly integrating text and data. In one study with the Veterans Medical Research Foundation I explored how providers might simultaneously place medication orders and document having placed those orders by parsing note-text in real time [7]. This system enabled providers to write free text in their note, and intersperse that text with medication orders constructed using an inline search interface. Rather than have a separate page to place orders, and then need to reference those structured orders when describing care plans in their note, providers could just place the order from their note while writing their note. In a usability study, providers expressed how this paradigm could save documentation time, improve patient safety by reducing discrepancies between their notes and structured orders, and how they would like to be able to both retrieve information (e.g., find most recent colonoscopy) and place other orders (e.g., x-ray of left leg) by simply typing into their notes. These results suggest untapped potential for notebooks in healthcare.


S: Patient is a 89M with hx HTN, HPL, PTSD, OA here for annual exam. He is taking tamulosin and requests refills. Also c/o occasional urinary incontenance.

O:
Labs:
Cholesterol panel: normal

A/P:
1. HTN: well controlled
2. HPL: SIMVASTATIN

Fig 5. The ActiveNotes system enabled physicians to place medication orders inline with other note text using a domain-specific search interface, bringing the notebook paradigm to clinical notes.

FUTURE RESEARCH AGENDA

In the coming years I aim to better equip researchers to study data analysis and help analysts robustly conduct and communicate analyses in diverse domains. I see three avenues for this future work: 1) developing techniques to scale observation of data-driven activities, 2) establishing analytical and communicative best practices, and 3) developing interactive systems to scaffold learning and applying best practices.

First, I want to develop tools to better track data analysis activities both in detail and at scale. This will include developing technologies that track interactions with specific pieces of software and those tracking activity across applications and even outside computing environments. Early in my PhD I developed software that enabled our team to track cross-application computer use, including screen recording, for weeks at a time without researcher intervention [8]. I have also been working with a national network of clinicians and researchers to develop standards for use of EHR audit logs to study clinical activities at scale. Better understanding the shape of data analysis in diverse domains will help us develop better tools to support it. Second, I aim to better articulate analytical and communicative best practice. While some programming best practices apply to data analysis, data analysis typically has different goals and outputs than programming, necessitating a different workflow. Building on prior work, [4] I would aim to collaboratively identify these best practices, particularly those that involve annotation and explanation. Finally, I aim to develop interactive systems that better support data analysis. With computational notebooks, this will involve developing extensions that encourage best practices and scaffold learning of them in educational contexts. With EHRs this will involve exploring how the notebook paradigm of mixing text, data, and orders in a single document might enhance clinical documentation and the practice of healthcare. My ultimate aim is to help analysts, especially end-user analysts in diverse domains, conduct analyses that are lucid, reproducible, and sound.

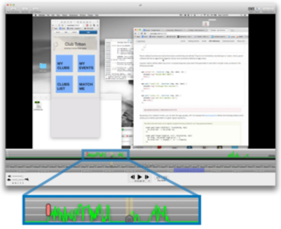


Fig 6. I developed the Traces program to enable long-term naturalistic studies of computer mediated work. Critically, Traces supports experience sampling and enables screen recording of participant's computers for weeks at a time without researcher intervention. These recordings can be used to guide follow-up interviews when visualized in a tool such as ChronoViz.

REFERENCES

[1] Perez F, Granger BE. Project Jupyter: Computational narratives as the engine of collaborative data science. 2015.

[2] **Rule A**, Tabard A, Hollan JD. Exploration and explanation in computational notebooks. *In Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, 2018. (Best Paper Honorable Mention)

[3] **Rule A**, Drosos I, Tabard A, Hollan JD. Aiding collaborative reuse of computational notebooks with annotated cell folding. *Proceedings of the ACM on Human-Computer Interaction, CSCW*, 2018

[4] **Rule A**, Birmingham A, Zuniga C, Altintas I, Huang SC, Knight R, Moshiri N, Nguyen MH, Rosenthal SB, Pérez F, Rose PW. Ten simple rules for writing and sharing computational analyses in Jupyter Notebooks. *PLoS Computational Biology*. 2019.

[5] **Rule A**, Hribar MR, Chiang MF. Clinical Documentation as End-User Programming (In Submission)

[6] Hribar MR, **Rule A**, Huang AE, Dusek H, Goldstein IH, Henriksen B, Lin WC, Igelman A, Chiang MF. Redundancy of Progress Notes for Serial Office Visits. *Ophthalmology*, 2019.

[7] **Rule A**, Rick S, Chiu M, Rios P, Ashfaq S, Calvitti A, Chan W, Weibel N, Agha Z. Validating free-text order entry for a note-centric EHR. *In American Medical Informatics Association annual symposium proceedings*, 2015

[8] **Rule A**, Tabard A, and Hollan J. Traces: A Flexible, Open-Source Activity Tracker for Workplace Studies. *CSCW Workshop on the Quantified Workplace*. 2016.