Leaf2Tableau: from Clinical Data to Clinical Knowledge Discovery

Xiyao Yang

A thesis

submitted in partial fulfillment of the

requirement for the degree of

Master of Science

University of Washington

2017

Committee:

Sean D Mooney

Mark M. Wurfel

Adam Wilcox

John H Gennari

Program Authorized to Offer Degree:

Biomedical and Health Informatics

©Copyright 2017

Xiyao Yang

University of Washington

Abstract

Leaf2Tableau: from Clinical Data to Knowledge Discovery

Xiyao Yang

Chair of the Supervisory Committee:

Sean D Mooney, PhD, Professor

Biomedical Informatics and Medical Education

Leaf2Tableau, a self-service and real-time clinical data visualization pipeline, is designed and developed to handle data visualization requests for queries developed in Leaf, a clinical data explorer developed by University of Washington Medicine Information Technology Services. It can extract and visualize any Leaf datasets into a portable format that researchers can easily explore without needing a highly technical or statistical background, providing a quick visual summary of the target population. This completes a clinical data warehouse (CDW) self-service model with a researcher constructing a query to identify a specific patient cohort in Leaf and subsequently developing custom visualizations for exploration or publication, as well as receiving data files for analysis.

CONTENTS

1 INTRODUCTION AND SIGNIFICANCE 6

2 BACKGROUND 15

3 LITERATURE REVIEW 8

4 RESEARCH DESIGN AND METHODS 19

4.1 OVERVIEW 19

4.2 TABLEAU DATA EXTRACT API 20

4.3 PUBLISH SINGLE DATASOURCES 21

4.4 PUBLISH PRE-CONSTRUCTED WORKBOOK 21

4.5 ADD USER PERMISSION 22

4.6 REAL CLINICAL USE CASE-- ACUTE KIDNEY DEFINITION (AKD) 23

5 RESULTS AND DISCUSSION 26

6 CONCLUSION 33

ACKNOWLEDGEMENTS

The author would like to express her heartfelt gratitude to the member of her committee, Dr. Sean D Mooney, Dr. Mark M Wurfel, Dr. Adam Wilcox, and Dr. John Gennari, for their invaluable contributions to this work. She would also like to thank Tony Black, Nic Dobbins, Bob Meizlik, Pavan Bhatraju, Eve Wilkerson and each of the members of Leaf team for their support, patience, insights, and general willingness to listen to her thoughts on this project at various times.

1 Introduction and Significance

As a result of technology advancements in clinical data warehouse (CDW) performance and storage capacity, the amount and the complexity of clinical data around clinical staff and researchers are constantly increasing. However, due to cognitive attributes of human beings, information processing by the human mind is not suited to analyzing large volumes of detailed data. This is especially true in the design of informatics tools in the clinical environment, which should prioritize enabling members of the clinical staff (physicians, technicians, nurses, students, managers) to take advantage of effective presentation and interaction with the data.

Information visualization can enable users to reveal deep details of clinical data by exploiting human's visual recognition abilities.[1] Information Visualization is well-studied in medicine[2], public healthcare[3], electronic medical data[4], and medical imaging[5]; but is relatively new to clinical research informatics. Current clinical informatics research includes plentiful examples of visualizing problem- and domain-specific clinical data, such as distributed time-oriented clinical records and their analysis;[6] few tools help researchers extract retrospective data obtained through query tools and delve into it using data visualization for a more general purpose.

Leaf is a next-generation self-service clinical data explorer sponsored by the University of Washington Medicine IT Services and the Institute for Translational Health Services. It supplies aggregate counts, information tables, basic visualizations and exporting functions of patient populations from the Caradigm clinical data repository. Leaf is effective at estimating patient cohort sizes and exporting cohort information for the purposes of quality improvement and research. Its exporting destination, REDCap, is a mature web application for building and managing online surveys and databases; and has an extendable architecture where plugins that support additional features can be developed. Currently, REDCap can export patient data extracted from Leaf to data analysis tools such as R, SPSS, SAS, and Stata. However, the distance between cohort identification and cohort information visualization is time-consuming and far from intuitive for Leaf users.

Visualization systems such as HARVEST[7] focus primarily on visualizing individual patients' longitudinal medical history rather than an entire cohort. SMART apps built a plugin inside i2b2, providing an EMR-like view and a natural-feeling medical review process for each patient.[8] Gnaeus[9] is an example of a cohort visualization tool; but it does not assist in finding the cohort. Important research has also been undertaken on visualization of patient histories, such as the LifeLine and KNAVE projects.[7] However, none of them match our attempts to bridge the gap between real-time clinical data from the whole patient population in the University of Washington Medical Center and its affiliated medical institutes, and clinical knowledge discovery. Meanwhile, an intuitive and user-friendly application in clinical settings will be welcomed by potential users. Therefore, Tableau, as a successful visualization software focused on business intelligence, is a handy choice for us to fill in the gap.

2 Literature Review

As electronic health record (EHR) use continues to rise across the nation, data from the EHR becomes a valuable resource for clinical quality improvement and translational research. This data, combined with other information such as genomic data, lays the foundation for personalized healthcare. The US National Institutes of Health has developed a repository that brings together clinical research data and provides researchers with access to EHR data: The Biomedical Translational Research Information System (BTRIS).[14] The i2b2 system at Partners HealthCare allows direct user access to de-identified data based on the role and training of the user, as well as the technical security of the client machine.[14] The Stanford Translational Research Integrated Database Environment (STRIDE) also creates a standards-based informatics platform providing summary statistics about patient research cohorts.[14] While each of the systems described above provides an interface for end users, when researchers find that they are unable to obtain the desired data directly, they need a human intermediary to obtain their data.[14] James J. Cimino et al. designed a de-identified self-service tool based on BTRIS that successfully extracted 4 queries out of 30 in its version 1.0. Although Dr.Cimino explained that the tool is expandable in data domains and attributes, it may not support cohort exploring, event comparing or longitudinal displaying.[14]

In early 2010, Harvard Medical School and Boston Children's Hospital began an interoperability project called Substitutable Medical Applications and Reusable Technologies (SMART) with the distinctive goal of developing a platform to enable medical applications to be written once and run unmodified across different healthcare IT systems.[15] Joshua C Mandel et al. adopted SMART on a new, openly licensed Health Level Seven draft standard called Fast Health Interoperability Resources (FHIR), making SMART powerful enough to support development of various clinical applications; for example, Bilirubin Chart aims to highlight the overlay of bilirubin [Mass/volume] results over a time-based risk chart. Aggregated Patient Data is designed to pull patient data from external health systems into one place while Growth Chart is developed to present a child's growth overtime.[15] As a technology platform, SMART on FHIR requires secondary development to build applications and is naturally far from individual end-clients. Meanwhile, it does not convert large-scale data or demonstrate real time query translations on top of large data sets.[15] Without deliberate modification, SMART prefers to deliver visual analytics on individual medical records.[15]

To help digest such large EHR data sets, the emerging field of visual analytics employs the human eye as a statistical tool for quickly recognizing patterns and dissimilarities.[16] In point of care, HARVEST is developed by New York Presbyterian Hospital and Department of Biomedical Informatics, Columbia University to act as an interactive, problem-oriented patient record summarization system. It differs from previous work that it has natural language parser of the patient notes and aggregates and presents information from multiple care settings. However, because HARVEST tracks single patients' timelines and documents all the problems, clinical questions hiding in cohort or relying on clinical note frequency will be missing. And

HARVEST is constrained to one medical center, which impairs its feasibility to a more diverse research-oriented environment.

On the other hand, John D. Manning et al. examined Lifelines2, a visualization program based on BTRIS from the University of Maryland's Human-Computer Interaction Laboratory, and found Lifelines2 has incredible potential to speed up analysis in a translational research setting.[16] But the learning curve of Lifelines2 is steep; it requires several hours of training and it is limited to temporal categorical events, meaning no duration qualifiers or numerical data may be imported.[16]

Megan Monroe et al. also introduced a visualization tool, EventFlow, that transforms an entire dataset of temporal event records into an aggregated display, allowing researchers to analyze population-level patterns and trends.[17] EventFlow is developed by the University of Maryland and designed to perform an algorithm consisting of a series of targeted simplifications that allow users to precisely and iteratively pare down complex temporal event datasets to the key visual elements that reveal meaningful patterns. Dr. Monroe's work draws on techniques from both temporal event query and data mining, as well as understanding how temporal relationships can be accessed and transformed within complex datasets.[17]

From the perspective of technology acceptance model (TAM), the three pre-determined high-level themes--Perceived Usefulness (PU), Perceived Ease of Use (PEOU), Actual Use (AU) can be used to compare EventFlow and Leaf2Tableau. Regarding PU, EventFlow serves as a useful visual tool and has clear potential to facilitate temporal patterns reveal in clinical research. For PEOU, EventFlow users are able to understand its visualizations with analysts' assistance in data input. EventFlow has already been applied to a lot of research while Leaf2Tableau is still in its pilot stage. Besides EventFlow, clinical researchers still need a self-service tool for preparing the data before they simplify and find patterns in it. The AU theme reveals that Leaf2Tableau serves better for real-time clinical data retrieval and intuitive target population identification.

Therefore, compliance with diverse clinical and research settings as well as a goal of easy use becomes the first concerns for the design of clinical data informatics tools. Leaf, the clinical data explorer developed in UWM ITS, is built upon flexible SQL builders and can be migrated to other clinical data warehouses and healthcare systems. It is a self-service web application with intuitive and friendly user-interface, compatible to most popular browsers and operational systems. Moreover, Leaf provides various logic and time/encounter combinations among query criteria, e.g. OR, AND, AND NOT, IN THE SAME ENCOUNTER, to help construct complex and multi-dimension clinical questions. It also allows users to see and modify SQL commands in its user interface, to customize filters, and to create their own concepts, granting them a huge room to fully explore their interested data. The most important is, Leaf is a tool fully covered by our design and under our control.

Besides, to determine the other end of the clinical data visualization pipeline, a comparison among popular visual analytics tools in the market has also been conducted. Figure 1 reflects the level of functionality of known data visualization tools and their derived score of ease of use in 2009, among which QlikView, Tableau and SpotFire rank relatively high and are still popular in both BI and healthcare fields. [20] Andrew Pandre, the author of Figure 1, also summarized his subjective quantitative comparison among QlikView, Tableau, SpotFire and Microsoft in aspects of pricing, time to implement, scalability, data interactivity and so on, where Tableau beats the other three products.[20]

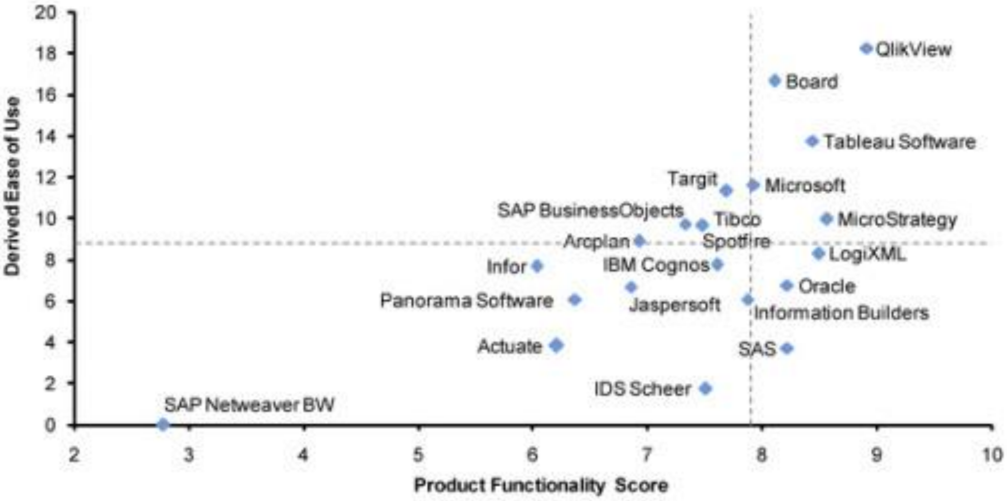
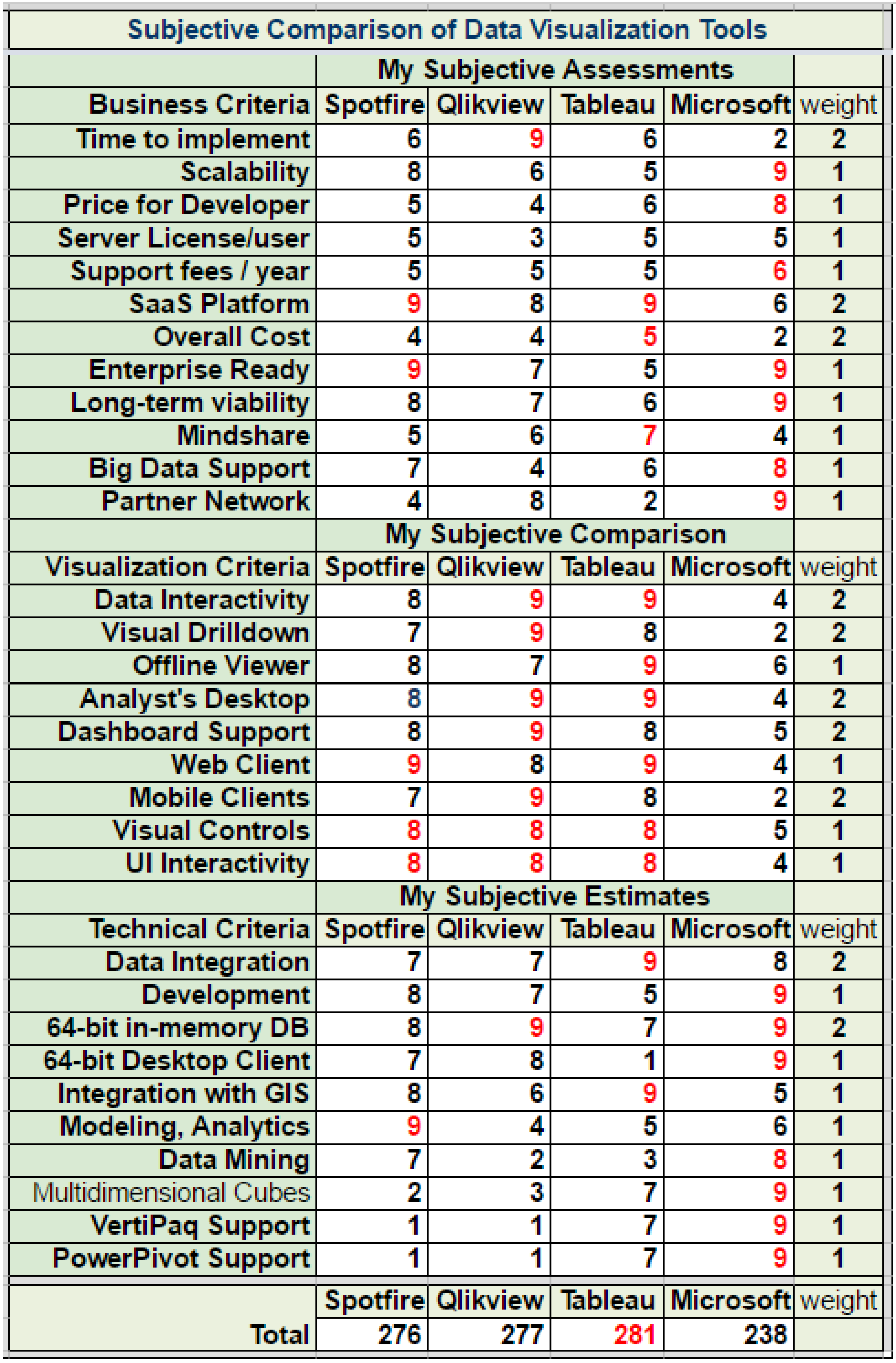


Figure 1: The level of Functionality of known BI and Data Visualization Tools (Andrew Pandre, 2016)



Subjective Comparison of Data Visualization Tools					
	My Subjective Assessments				
Business Criteria	Spotfire	Qlikview	Tableau	Microsoft	weight
Time to implement	6	9	6	2	2
Scalability	8	6	5	9	1
Price for Developer	5	4	6	8	1
Server License/user	5	3	5	5	1
Support fees / year	5	5	5	6	1
SaaS Platform	9	8	9	6	2
Overall Cost	4	4	5	2	2
Enterprise Ready	9	7	5	9	1
Long-term viability	8	7	6	9	1
Mindshare	5	6	7	4	1
Big Data Support	7	4	6	8	1
Partner Network	4	8	2	9	1
	My Subjective Comparison				
Visualization Criteria	Spotfire	Qlikview	Tableau	Microsoft	weight
Data Interactivity	8	9	9	4	2
Visual Drilldown	7	9	8	2	2
Offline Viewer	8	7	9	6	1
Analyst's Desktop	8	9	9	4	2
Dashboard Support	8	9	8	5	2
Web Client	9	8	9	4	1
Mobile Clients	7	9	8	2	2
Visual Controls	8	8	8	5	1
UI Interactivity	8	8	8	4	1
	My Subjective Estimates				
Technical Criteria	Spotfire	Qlikview	Tableau	Microsoft	weight
Data Integration	7	7	9	8	2
Development	8	7	5	9	1
64-bit in-memory DB	8	9	7	9	2
64-bit Desktop Client	7	8	1	9	1
Integration with GIS	8	6	9	5	1
Modeling, Analytics	9	4	5	6	1
Data Mining	7	2	3	8	1
Multidimensional Cubes	2	3	7	9	1
VertiPaq Support	1	1	7	9	1
PowerPivot Support	1	1	7	9	1
	Spotfire	Qlikview	Tableau	Microsoft	weight
Total	276	277	281	238	

Figure 2: Subjective Data Visualization Comparison - end of 2011 (Andrew Pandre, 2011)

While there is no single best visualization product, Tableau, compared to its competitors, is good at connecting multiple datasets to generate a lot of views in one workbook and share the data filters and markers. It is fast to implement and does not need software experts to develop scalable SaaS. And Tableau is excellent in interaction with Online Analytical Processing (OLAP) cubes, which perform multi-dimensional data analysis and provides the capability for complex calculations, trend analysis, and sophisticated data modeling. Using Tableau allows users without a solid background in statistics to move quickly on visualizing and understanding the structure of data sets and gleaning top level insights. Thus, I believe that building a data visualization pipeline integrating Leaf with Tableau's powerful visualizing features will benefit lots of clinical researchers and staff.

3 Background

Leaf2Tableau is built based on the new generation clinical data query tool Leaf, which is developed by University of Washington Medicine (UWM) IT services (ITS), Nicholas Dobbins, and his team, including this author.

The data flow (Figure 3) behind Leaf is straightforward. First, raw data coming from ORCA, Epic, Cerner and other systems are parsed and ingested by UWM ITS Amalga team in real time. Then, the Amalga team creates and updates entities to store these data. Leaf utilizes these entities as the basis for its concepts to form query criteria. End users interpret their questions into these concepts to find their desired patient population.

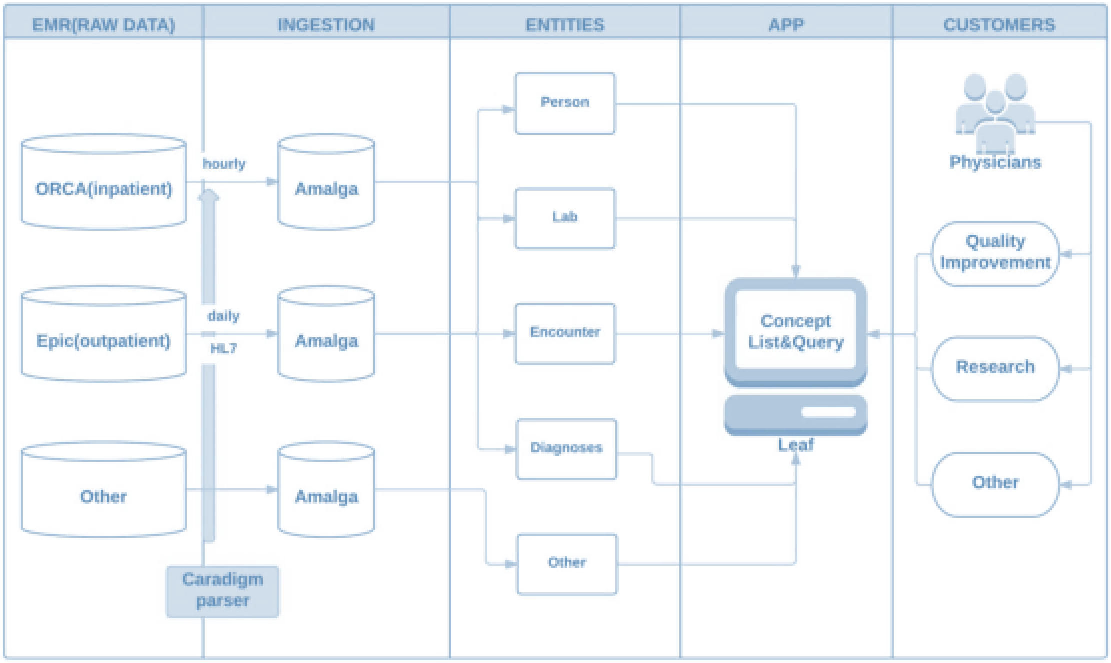
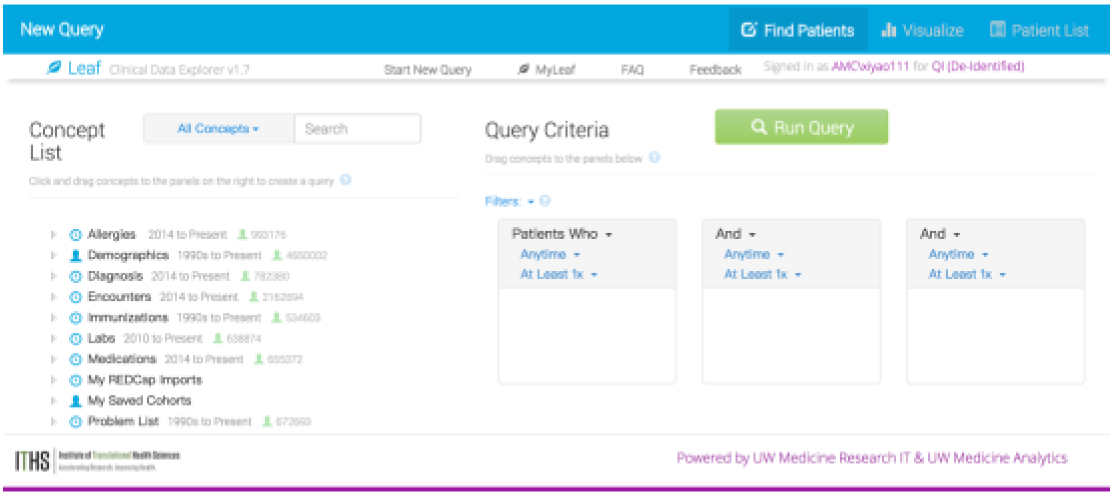


Figure 3: Data flow behind Leaf.

Leaf has three major tabs--*Find Patients, Visualiz*e and *Patient List*. The *Find*

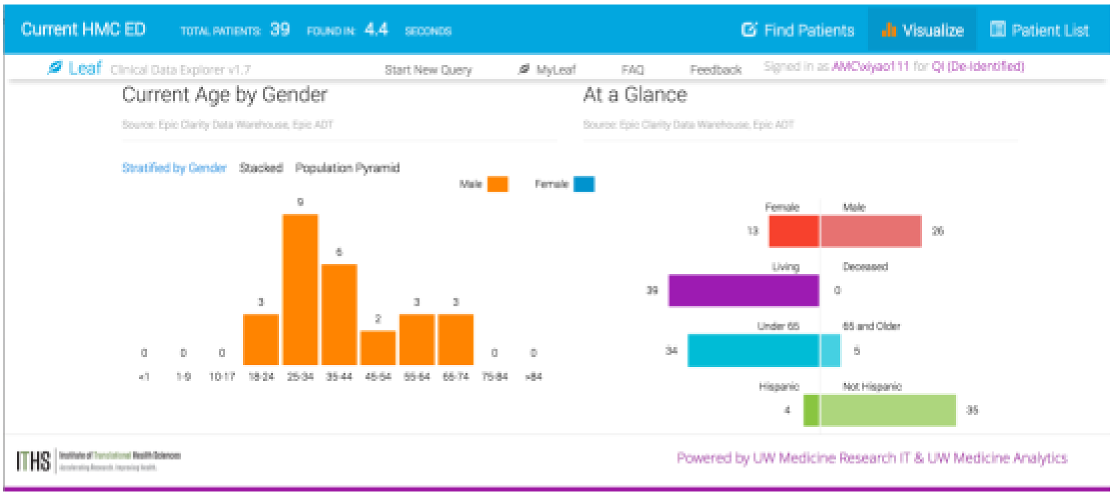
Patients tab contains the cohort search features of Leaf. Users can search medical concepts and drag them over into three query panels and combine various logic for their query. By clicking the *Run Query* button, researchers obtain the total count of their target cohort. They can also add concepts as custom filters and save the existing query. Leaf will also show the source and retrieval time of its query as well as display the SQL syntax behind the query for highly motivated researchers to validate the data.



New Query
Find Patients
Visualize
Patient List
Leaf Clinical Data Explorer v1.7
Start New Query
MyLeaf
FAQ
Feedback
Signed in as AMCxiyao111 for QI (De-identified)
Concept List
All Concepts
Search
Click and drag concepts to the panels on the right to create a query
Allergies 2014 to Present 993176
Demographics 1990s to Present 4650002
Diagnosis 2014 to Present 782380
Encounters 2014 to Present 2162694
Immunizations 1990s to Present 534603
Labs 2010 to Present 638874
Medications 2014 to Present 655372
My REDCap Imports
My Saved Cohorts
Problem List 1990s to Present 672693
Query Criteria
Run Query
Drag concepts to the panels below
Filters:
Patients Who
Anytime
At Least 1x
And
Anytime
At Least 1x
And
Anytime
At Least 1x
ITHS Institute of Translational Health Sciences
Powered by UW Medicine Research IT & UW Medicine Analytics

Figure 4: Screenshot of Leaf; Find Patient Page.

The *Visualize* tab provides visualizations on population stratified by age, gender and vital status, visits in past 12 months and clinics and services in past 12 months. These graphics are built with the D3 Javascript package and aim at demonstrating a general view of the queried cohort. This tab also has a breakdown view to record the initial query criteria.



Current HMC ED TOTAL PATIENTS: 39 FOUND IN: 4.4 SECONDS

Leaf Clinical Data Explorer v1.7 Start New Query MyLeaf FAQ Feedback Signed in as AMC\xiyao111 for QI (De-Identified)

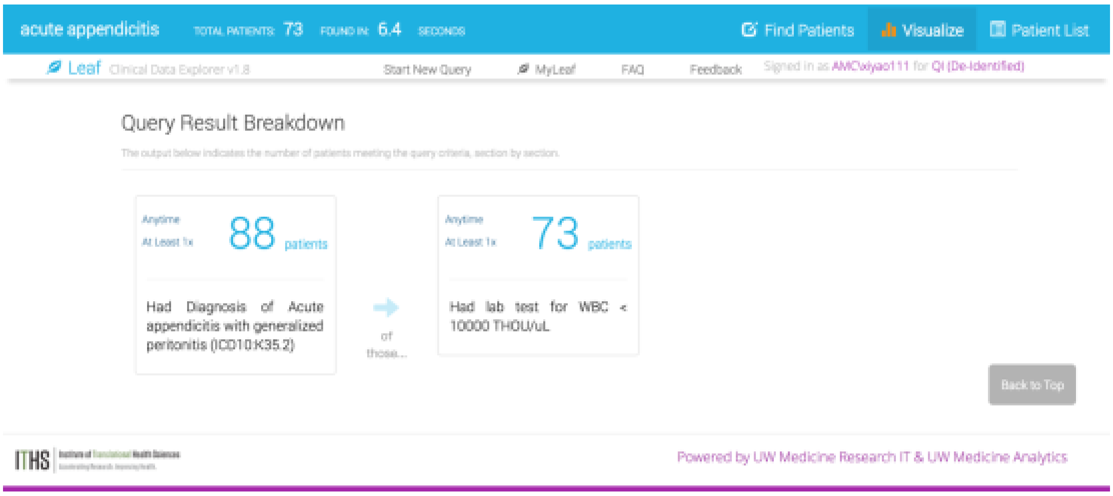
Current Age by Gender

Source: Epic Clarity Data Warehouse, Epic ADT

At a Glance

Source: Epic Clarity Data Warehouse, Epic ADT

Figure 5: Screenshot of Leaf; Visualize Page.



acute appendicitis TOTAL PATIENTS: 73 FOUND IN: 6.4 SECONDS

Leaf Clinical Data Explorer v1.8 Start New Query MyLeaf FAQ Feedback Signed in as AMC\xiyao111 for QI (De-Identified)

Query Result Breakdown

The output below indicates the number of patients meeting the query criteria, section by section.

Anytime At Least 1x 88 patients

Had Diagnosis of Acute appendicitis with generalized peritonitis (ICD10:K35.2)

of those...

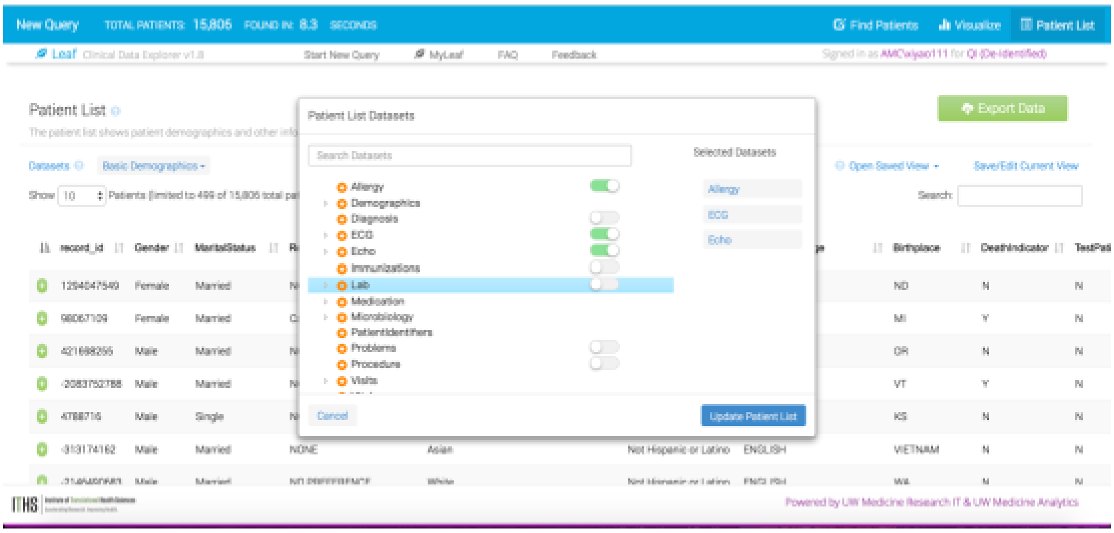
Anytime At Least 1x 73 patients

Had lab test for WBC < 10000 THOU/uL

Find Patients
Visualize
Patient List
Back to Top
ITHS Institute of Translational Health Sciences
Powered by UW Medicine Research IT & UW Medicine Analytics

Figure 6: Screenshot of Leaf; Visualize Page—Query Result BreakDown.

After Leaf users get their query results, they can switch to *Patient List* tab, where they can configure the detailed information of the target population to be displayed as a table. For example, when I get a total number of 77 patients who are currently in the emergency department, I can configure the column of their longitudinal A1c observational values as a spark line in *Patient List* by selecting the dataset *A1c* as well as its configuration *A1c Trend.*



record_id	Gender	MaritalStatus	R			ge	Birthplace	DeathIndicator	TestPat
1294047549	Female	Married	N				ND	N	N
98067109	Female	Married	C				MI	Y	N
421698255	Male	Married	N				OR	N	N
-2083752788	Male	Married	N				VT	Y	N
4788716	Male	Single	N				KS	N	N
-313174162	Male	Married	NONE	Asian	Not Hispanic or Latino	ENGLISH	VIETNAM	N	N
-2146400683	Male	Married	NO PREFERENCE	White	Not Hispanic or Latino	ENGLISH	WA	N	N

Figure 7: Screenshot of Leaf; Patient List Page.

4 Research Design and Methods

4.1 Overview

Leaf uses the Caradigm Clinical Data Warehouse (CDW) as a source for clinical data, which it extracts and distributes for research and quality control. In digesting the raw data from tons of data sources such as Epic and Cerner, the Caradigm CDW parses them into entities like Person, Lab, Medicine, etc., which Leaf displays as Concepts on its user interface. One of the goals of the Leaf2Tableau is to facilitate the process of delivering and visualizing extracted data to a researcher once they have found a desired patient population in Leaf.

Tableau is a commercial software package for authoring visualizations and it has independent servers and sites to restrict user access, which can be used to prevent the widespread distribution of personal health information. Clinical data can be delivered in the form of Tableau Data Extract (.tde) and packaged workbook (.twbx) files, which can then be published on the Tableau server pre-established by the University of Washington IT Services.

The initial workflow was designed as two separate pathways and depicted in Figure 8. When a user requests an extract of a patient cohort with their information for a given Leaf query, the queries are logged in a request table along with the user's identity. Leaf2Tableau uses the Tableau Data Extract API to create Tableau Data Extract (TDE) files. Leaf uses the DataTables Javascript library to deliver query results in its *PatientList* tab. After the data transformation has been done, the TDE files are published as a Tableau Data Source to the Tableau Server. In the other pathway, the TDE extracted directly from upstream CDW is transformed as Visualization Template Repository grouped by entities and delivered to Tableau Workbook as custom datasets that can be manipulated by users.

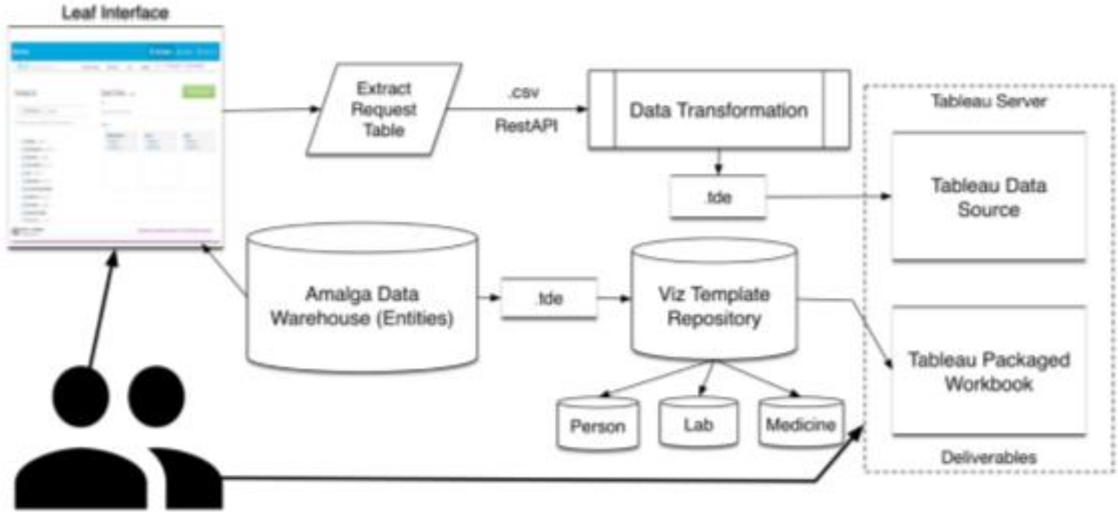


Figure 8: The pipeline of Leaf2Tableau begins with a Leaf query and ends with a deliverable containing TDE files for constructing additional visualizations and pre-constructed workbooks with interactive visualizations copied from a template.

4.2 Tableau Data Extract API

To prepare the TDE file, preliminary staging of extracted data is necessary. Tableau replaces the patient and encounter identifiers with randomly generated keys specific to that request in its data extract. This reduces security concerns that additional information could be leaked or deduced from a "connect the dots" attack with multiple data extracts.[10] Also, with the goal of protecting personal health information, Leaf date shifts each patient's age and each record's date time in its de-identified mode, which is used in this project, faithfully preserving the time range between events within a patient's record.[11]

Because Tableau does not provide Data Extract SDK or API that can be fit in Leaf development environment, I used a third-party API written in C# that contains basic classes such as Row.cs, Table.cs and Collation.cs to define a TDE table. Modification of the API is mainly focused on changing .dll files loading directory, debugging the compatibility among Leaf, Caradigm Intelligence Platform and Tableau Data Extract API, and writing in TDE file from Leaf datasets through DataTable.

4.3 Publish Single Datasources

Leaf datasets can be published to Tableau server as single datasources solely. Once staging is complete, queries that extract selected dimensions, such as diagnoses, medications, procedures, and so on, are logged and executed. The result set of these queries are written to TDE files by adding DataTable columns and rows. The TDE files are published as the data source in Tableau Server when users click the Export button in Leaf. After getting the response from Tableau REST, Leaf returns an URL to the published datasource. By clicking the URL, users enter the Tableau Server user interface, create new workbook from the datasource and generate their own visualization with high freedom.

4.4 Publish Pre-Constructed Workbook

To construct the workbook, a template must be copied from a local file repository and its data sources have to be set to the newly extracted TDE files. The workbook templates should contain visualizations we have developed and found useful for a general-purpose data extract. Then users are directed to Tableau Server user interfaces to view the workbook or to form new views by leveraging the published TDE files. However, to configure and develop the template repository is time-consuming and out of the author's capability and beyond her accessibility while manipulating Leaf entities. In this project, there are no packaged workbooks or pre-designed templates in the final design.

As a last step, Leaf2Tableau compresses all data files, workbooks, and visualizations into a single compressed file that is deliverable to the researcher.

4.5 Add User Permission

Even though the scope of the project is to explore the potential of self-service clinical data visualization via de-identified mode in Leaf, however, the security of personal health information will always be the priority and first concern. Based on that, I used the Add Default Permission and the Add Datasource Permission in the Tableau server REST API while publishing data. The default user permission is set as inaccessible to any datasource, while the user is granted the right to see his or her own datasource every time they request exporting to tableau in Leaf. The goal is to prevent users from glancing at others' workbooks or datasources, causing possible information leaks.

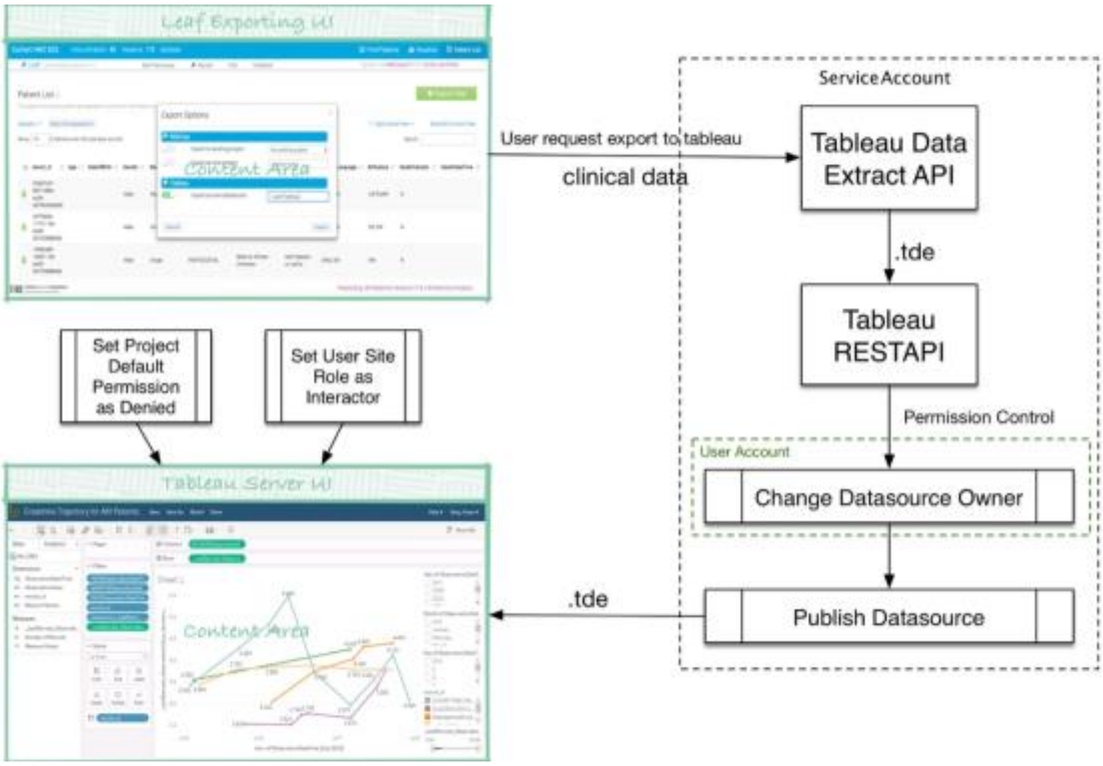


Figure 9: Permission Control Workflow. Firstly, by manually adding leaf users to tableau server, their project default permissions are denied and their site role are interactors initially. That means they can see nothing because they have no access to any datasource or views. When a user request export clinical data from leaf to Tableau, service account will generate TDE files through Tableau data extract API. Then it will set the user as the owner of the datasource and publish the TDE files to Tableau server. After above process, users can only view and edit their own published datasource, reducing the security concern of patient health information disclosure.

4.6 Real Clinical Use Case-- Acute Kidney Injury Definition (AKD)

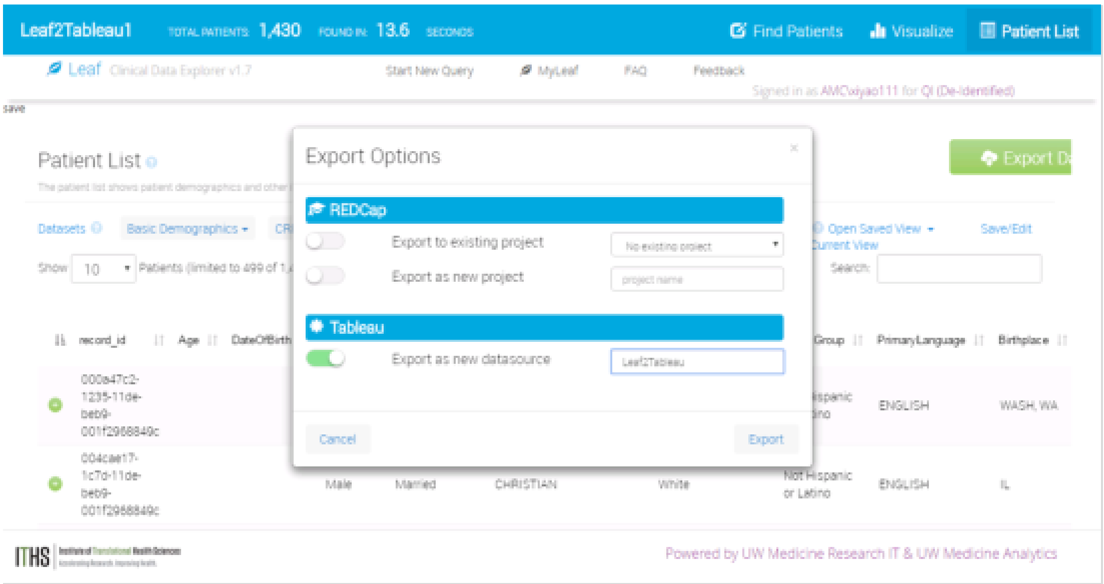
As an engineering project, Leaf2Tableau is created to pull out cohort longitudinal health information and demonstrate key observational values via Tableau visualization, assisting researchers to locate target populations as well as acquire a quick visual understanding. Clinical research examples in the real world are essential for shaping the tool, among which the knowledge gap still exists in examining how the trajectory of kidney function over the course of a hospital admission is related to clinical outcomes. AKD relies a lot on related lab value trajectory in 7 days to 90 days after acute kidney injury has been diagnosed.[12]

As we know, acute kidney injury (AKI) is common among intensive care unit (ICU) and trauma unit patients. It is highly heterogeneous and has variable comorbidities with poor outcomes. Traditional methods to stage AKI severity and distinguish patients at most noteworthy hazard for poor results concentrate on the greatest change in serum creatinine (SCr) values, which, however, are hampered by the need for a reliable baseline SCr value and the absence of a component differentiating transient from persistent rises in SCr.[13] Pavan K. Bhatraju et al. performed a secondary analysis and tested definitions for resolving and non-resolving AKI subphenotypes and selected the definitions resulting in subphenotypes with the greatest separation in risk of death relative to non-AKI controls.[13] They found that a resolving AKI subphenotype (defined as a decrease in SCr of 0.3 mg/dl or 25% from maximum in the first 72 h of study enrollment) was associated with a low risk of death and a non-resolving AKI subphenotype (defined as all AKI cases not meeting the “resolving” definition) was associated with a high risk of death.[13]

Based on Pavan’s study, visualizing patients’ early creatinine trajectory in the ICU and categorizing them by resolving and non-resolving subphenotypes are essential to better differentiate patients at risk of AKI-associated mortality. In order to retrieve useful data to assist the AKD study, the target population query criteria should consist of patients who have been diagnosed as having an acute kidney injury in the ICU or trauma unit. The datasets extracted should include age, sex, BMI, race, comorbidity, ICU events and admission status.

5 Results and Discussion

Leaf users query a cohort of patients matching their criteria and configure the datasets they want to export. In the AKD example, I queried for *Patient Who Had Diagnosis of Acute Kidney Failure and Had Lab Test for Creatinine* and chose Creatinine and Basic Demographics as my exporting datasets for these specific 4,333 patients. After that, an exporting request was processed and Leaf2Tableau returned an access URL to the published datasources.



Export Options

REDCap

○ Export to existing project: No existing project

○ Export as new project: project name

Tableau

● Export as new datasource: Leaf2Tableau

Cancel | Export

Show 10 Patients (limited to 499 of 1,

record_id	Age	DateOfBirth					Group	PrimaryLanguage	Birthplace
000e47c2-1235-11de-beb9-001f2968849c							ispanic ino	ENGLISH	WASH, WA
004cae17-1c7d-11de-beb9-001f2968849c			Male	Married	CHRISTIAN	White	Not Hispanic or Latino	ENGLISH	IL

Figure 10: Leaf2Tableau Exporting Panel.

Tableau published datasources can be linked with a shared unique identifier and can create as many views as desired in a workbook. Data and images can be downloaded from Tableau Server once the work has been done. To modify a view, users can drag and drop field names as well as writing simple SQL syntax in columns and rows in the Tableau Server editing panel. To prevent information overload, multi-perspective summaries are provided, such as sum, average, maximum, minimum, etc.

Leaf2Tableau attempts to fulfill two common visualization needs: discovering trends and identifying possible patterns. Figure 11 shows how numbers of creatinine records a patient have per day could help identify clinical research interest. Figure 12 shows how a high-volume set of lab values can be concisely summarized with a traditional box-and-whiskers plot. Figure 13 shows a single patient's creatinine trajectory in 2016. Because current AKI classification systems have defined AKI by an increase in serum creatinine of 0.3 mg/dl or 50% over a 48-72 hr period[13], Figure 14 aims at answering the question: how many patients have over 0.3 mg/dl increase in serum creatinine within 72 hours?

But these visualizations are not enough for distinguishing the resolving group from the non-resolving group of AKI patients. The resolving group is defined by 0.3 or 25% decrease in serum creatinine.[13] The non-resolving group is defined by no decrease. [13] Researchers may want to mark the two groups' trajectory with rate of mortality or events like renal replacement therapy and then compare the two groups.[19] Figure 15 stratified the cohort in Figure 14 by death indicator. However, a likely request from users would be to visualize inflection points. In the case of the AKD study, Tableau limits itself to a single label, and Leaf does not provide functions to implement algorithms on its datasets.

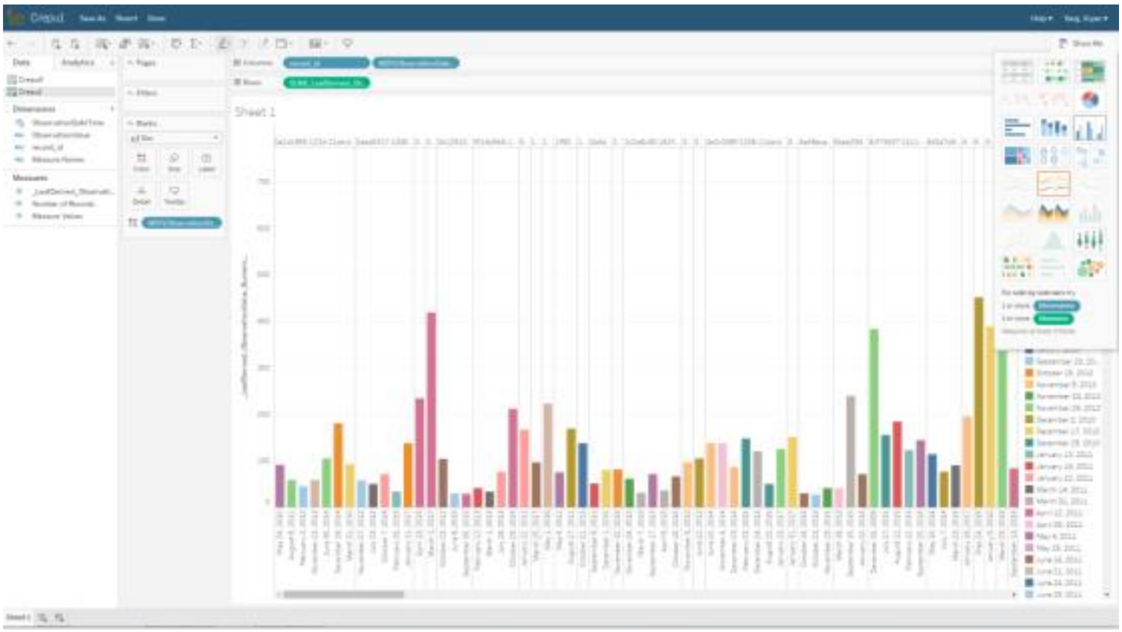


Figure 11: Numbers of creatinine records a patient have per day.

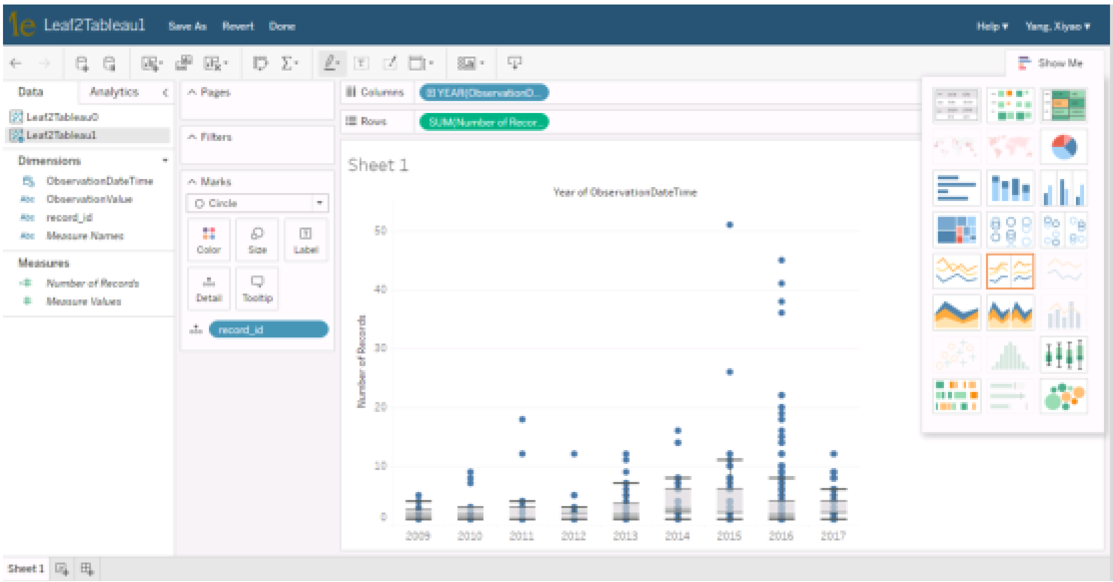


Figure 12: A high-volume set of lab values summarized with a traditional box-and-whiskers plot.

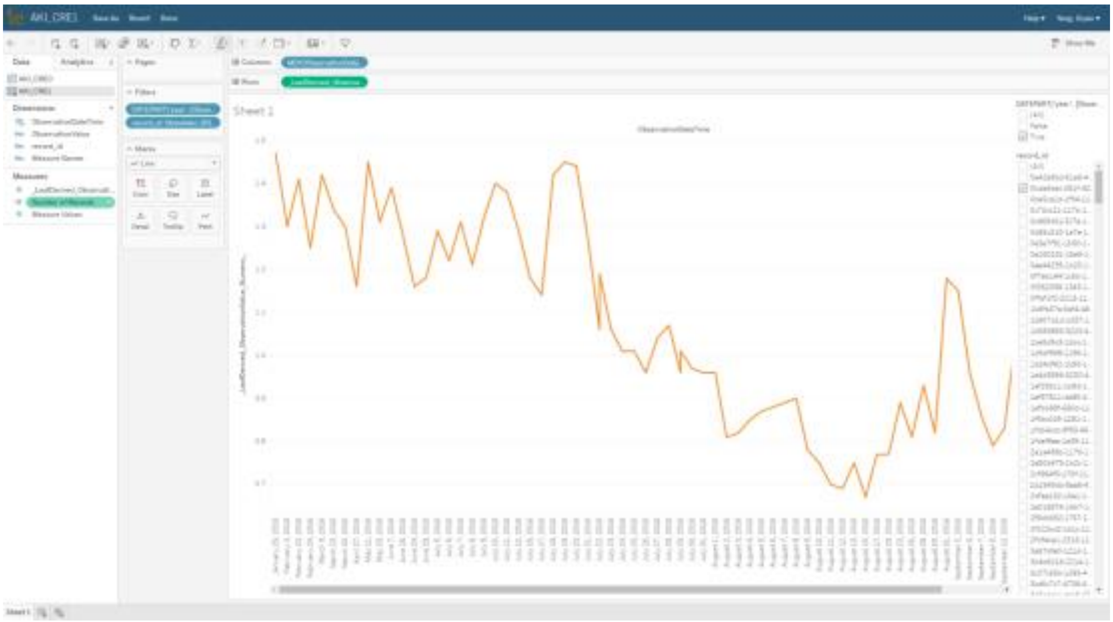


Figure 13: A single patient's creatinine trajectory in 2016.

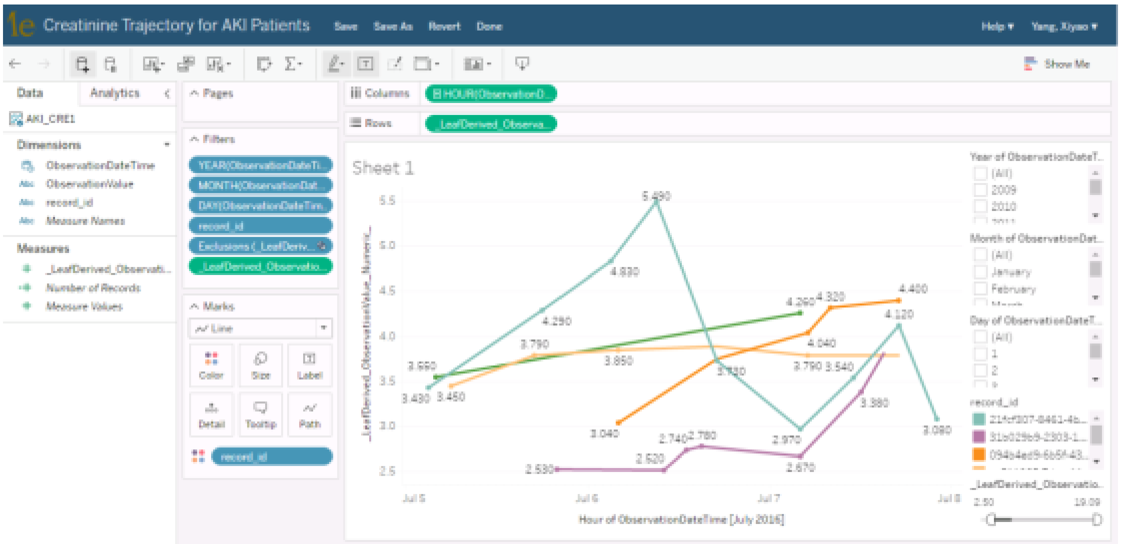


Figure 14: Between July 5th to 8th, how many patients had an increase of over 0.3 mg/dl in serum creatinine within 72 hours?

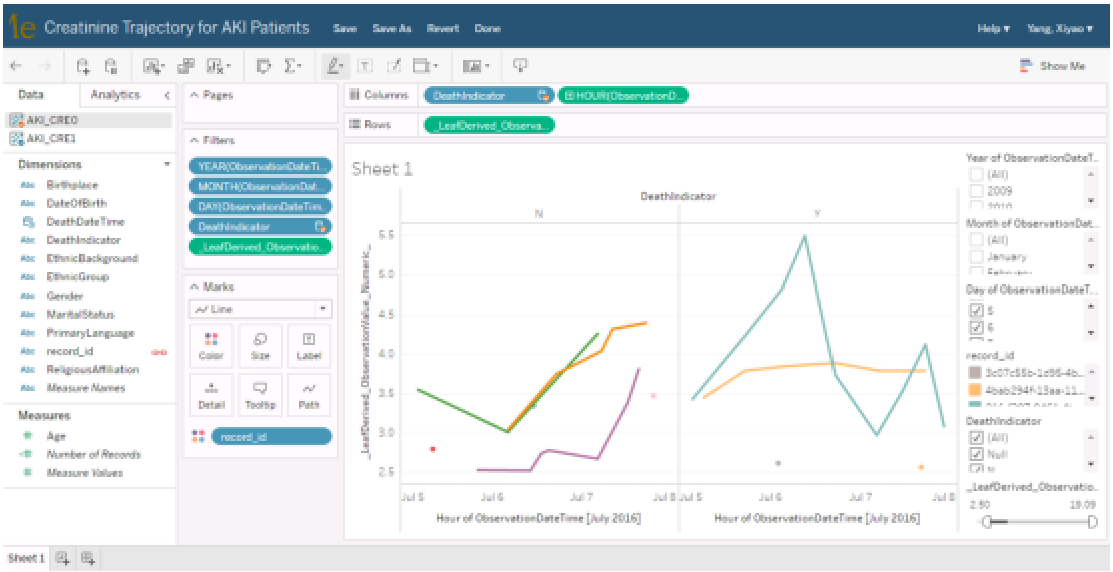


Figure 15: Stratification of the cohort in Figure 11 with death indicator.

For security and practical reasons, all queries are logged so that we know exactly what data has been included in a given data extract. This is necessary in both security and regulatory audits of clinical data releases. However, due to Tableau server license limitation and secure concern, we cannot store users' workbooks for a long time. So users can't re-access to their datasources as well as visualizations 48 hours after they publish and modify them. They are welcome to download the data as text files and images to save and share their work, or have a deeper exploration in them via Tableau Desktop. By default, all extracts are released with identifiers marked as *record_id* that can be connected and inner-joined within datasources on Tableau server.

There is more than one strategy for visualizing data in Leaf. Before creating the entire pipeline, Leaf had already implemented D3.js to visualize basic demographics, religions, insurance, residence and ethnicity of queried patients. Initial feedback indicates an intuitive glance at key observational value was greatly welcomed. The

Leaf team piloted the idea of providing visualizations to researchers as part of their data extract requests, so we added sparklines in *PatientList* numeric datasets to satisfy this demand. However, sparklines have no common or labeled axes, which makes them not comparable and hard to convey clinical knowledge. Then a trial of embedding Tableau pre-processed views in Leaf to provide more information was attempted but not favored. Software development is an inarguably expensive process and often requires maintenance in perpetuity. We avoid the need for programming visualization plug-ins using Tableau SDK by extracting data in Leaf and connecting to the Tableau ecosystem through RESTful APIs, which improve the user's experience by providing a consistent and professional visualization environment.

In our early findings, a clinical researcher was pleased that he did not have to completely rely on data analytic collaborators to explore the data set and construct the visualizations. Being able to engage a very large dataset without the need for an advanced statistical package eliminated a barrier for clinical research.[1] The pitfall of this approach is that not every visualization need can possibly be met. This is largely due to the data manipulation schema of Caradigm Entities and the functional limits of Tableau server. A highly-motivated researcher can use the TDE files to construct his or her own visualizations for research or publications without needing template workbooks. There is a natural learning curve to Tableau and there have been studies of known barriers and challenges with novices creating visualizations.[18]

The approach for having visualization as a service relies upon the idea that existing information visualization authoring tools are highly effective in creating visualizations. Once the Tableau datasources for visualization are authored, instances of the visualizations can operate on specific datasets and be released to the researchers to aid them in surveying the data.[1] Tableau has been chosen as a visualization framework because of its wide-spread free licenses for University of Washington students and faculty and its free PDF-like reader tool that researchers can easily download. As the process already yields TDE files, additional formats and powerful extension of visual analytics can be supported in the future.

The entire pipeline is automated which drastically increases the ability of UWM ITS CDW to release data to researchers quickly. UWM ITS has extracted data as a data analysis service for several years and the potential to create a general-purpose data extracts greatly unburdens its Research IT team. UWM ITS Amalga Application Team uses and maintains Caradigm CDW that feeds Leaf. We could have interfaced Tableau with Caradigm Entities directly, but by choosing to layer Tableau with Leaf, we made it entirely self-service. The Leaf2Tableau efforts are reusable and inter-operable in the biomedical community for other CDWs because its SQL builders are easy to migrate. As illustrated in Figure 1, Leaf acts as a customer-facing portal that enables cohort discovery under a self-service model where data extracts and visualizations are additional deliverables of the process.

As future work, we are experimenting with Leaf2Tableau on effective ways to tell a story with visualizations for a given patient population. Intuitively, this requires the most relevant templates to be chosen based upon the population selected and to order visualizations in a way that tell a meaningful story.

6 Conclusion

Leaf2Tableau, a clinical data visualization pipeline, is designed and developed to handle data visualization requests for queries developed in Leaf. The resulting data extracts contain raw data files and intuitive data visualizations in the Tableau Server user interface, which assist researchers in exploring and understanding their data effectively. This completes a CDW self-service model with a researcher constructing a query to identify a specific patient cohort in Leaf and subsequently developing custom visualizations for exploration or publication, as well as receiving data files for analysis.

BIBILOGRAPHY

1. Harris DR, Henderson DW, Sciences T. i2b2t2 : Unlocking Visualization for Clinical Research. :98-104.

2. Chittaro L. Information visualization and its application to medicine. Artificial intelligence in medicine.2001;22(2):81-88.

3. Shneiderman B, Plaisant C, Hesse BW. Improving healthcare with interactive visualization. Computer.2013;46(5):58-66.103

4. West VL, Borland D, Hammond WE. Innovative information visualization of electronic health record data: a systematic review. Journal of the American Medical Informatics Association. 2015;22(2):330-339.

5. Bui AA, Hsu W. Medical Data Visualization: Toward Integrated Clinical Workstations. In: Medical Imaging Informatics. Springer; 2010. p. 139-193.

6. Shahar Y, Goren-Bar D, Boaz D, Tahan G. Distributed, intelligent, interactive visualization and exploration of time-oriented clinical data and their abstractions. Artificial intelligence in medicine. 2006;38(2):115-135.

7. Hirsch JS, Tanenbaum JS, Gorman SL, Liu C, Schmitz E, Hashorva D, et al. HARVEST, a longitudinal patient record summarizer. Journal of the American Medical Informatics Association. 2015;22(2):263-274.

8. Wattanasin N, Porter A, Ubaha S, Mendis M, Phillips L, Mandel J, et al. Apps to display patient data, making SMART available in the i2b2 platform. In: AMIA Annual

Symposium Proceedings. vol. 2012. American Medical Informatics Association; 2012. p. 960.

9. Federico P, Unger J, Amor-Amoros A, Sacchi L, Klimov D, Miksch S. Gnaeus: utilizing clinical guidelines for knowledge-assisted visualization of EHR cohorts. In: EuroVis Workshop on Visual Analytics (EuroVA) vol. 2015. The Eurographics Association; 2015.

10. Whang SE, Garcia-Molina H. Secure Data Management. Springer; 2012. A model for quantifying information leakage; pp. 25–44.

11. El Emam K, Arbuckle L. O'Reilly Media, Inc.; 2013. Anonymizing health data: case studies and methods to get you started.

12. Chawla LS, Bellomo R, Bihorac A, et al. Acute kidney disease and renal recovery: consensus report of the Acute Disease Quality Initiative (ADQI) 16 Workgroup. Nat Rev Nephrol. 2017;13(4):241-257. http://dx.doi.org/10.1038/nrneph.2017.2.

13. Bhatraju PK, Mukherjee P, Robinson-Cohen C, et al. Acute kidney injury subphenotypes based on creatinine trajectory identifies patients at increased risk of death. Crit Care. 2016;20(1):372. doi:10.1186/s13054-016-1546-4.

14. Cimino JJ, Ayres EJ, Beri A, Freedman R, Oberholtzer E, Rath S. Developing a Self-Service Query Interface for Re-Using De-Identified Electronic Health Record Data. Studies in health technology and informatics. 2013;192:632-636.

15. Joshua C Mandel, David A Kreda, Kenneth D Mandl, Isaac S Kohane, Rachel B

Ramoni; SMART on FHIR: a standards-based, interoperable apps platform for electronic health records. J Am Med Inform Assoc 2016; 23 (5): 899-908. doi: 10.1093/jamia/ocv189

16. Shneiderman B. The eyes have it: A task by data type taxonomy for information visualizations. VL '96 Proceedings of the 1996 IEEE Symposium on Visual Languages. 1996 Sep;3:336-43

17. Monroe M, Lan R, Lee H, Plaisant C, Shneiderman B. Temporal Event Sequence Simplification. 2013;(March).

18. Grammel L, Tory M, Storey MA. How information visualization novices construct visualizations. Visualization and Computer Graphics, IEEE Transactions on. 2010; 16(6):943-952.

19. Bhatraju PK, Mukherjee P, Robinson-Cohen C, et al. Acute kidney injury subphenotypes based on creatinine trajectory identifies patients at increased risk of death. Crit Care. 2016;20(1):372. doi:10.1186/s13054-016-1546-4.

20. Pandre, A. (2016, February 02). Tools. Retrieved May 22, 2017, from https://apandre.wordpress.com/tools/