Paper PP09

A Data Visualization App to Reduce Event-Based Requests

Diana Tai, Bristol Myers Squibb, Berkeley Heights, NJ, USA
Lixiao Chen, Bristol Myers Squibb, Berkeley Heights, NJ, USA
Juanyao Huang, Bristol Myers Squibb, Berkeley Heights, NJ, USA
Po-Chun “Kelvin” Lin, Bristol Myers Squibb, Berkeley Heights, NJ, USA

ABSTRACT

R Shiny applications have proven to be an effective framework for data exploration and analytics due to the unique features offered.

In medical affairs and market access (MMA), programming activities mainly focus on exploratory analysis, supporting posters, abstracts, manuscripts, publications, and conference presentations. Large quantities of event-based requests (EBR) have become a burden for the MMA biostatistics and programming groups. A comprehensive R Shiny app has been built to help statisticians and clinical teams explore data/perform data analytics to help decide whether a formal EBR is required. This app has proven to be an effective tool that has significantly reduced the number of EBRs needed and greatly impacted the budget that is allocated to support EBRs.

INTRODUCTION

The R Shiny application we developed is a flexible, dynamic, and powerful data review and analysis tool suitable for a variety of data, including post-processed data, SDTM, ADaM, and real-world data (RWD). Statistical programmers are able to use the application to analyze data, create tables/graphs, and more. The ease of use means that statisticians, clinicians, clinical scientists, worldwide patient safety (WWPS), and the health economics and outcomes research (HEOR) team can use the app as well. With all the flexibility and accessibility provided to a broad audience, this leads to a significant reduction in the number of EBRs needed.

OVERVIEW

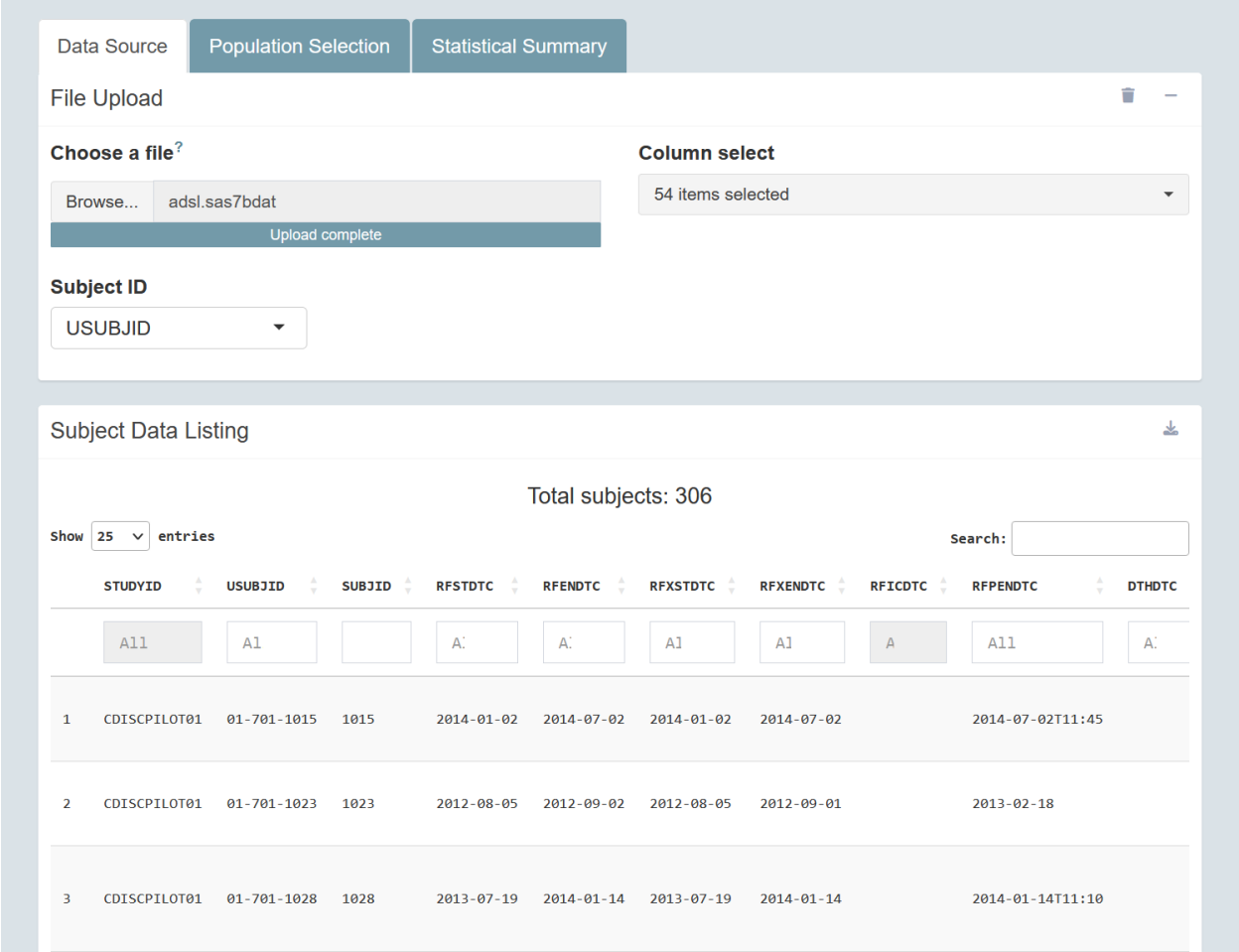
The application was created using a variety of R packages, with *shiny* serving as the basic framework. This was enhanced with additional packages such as *shinydashboard*, *shinyWidgets*, and *shinyjs*. For data manipulation and transformation, we made use of packages from the *tidyverse* collection, including *dplyr*, *tidyr*, and *purrr*. Most of the data used in creating the sample outputs originates from *pharmaverseadam*, which contains test data for the *pharmaverse* family of packages. The additional packages used to create and display particular outputs will be mentioned later as they come up in discussion.

There are three main sections within the app: Data Exploration, the Table Builder, and the Graph Builder. Data Exploration contains many options to explore both BDS and subject-level data. The Table Builder and Graph Builder offer common templates for users to choose from. They allow for output generation with high efficiency, where one template can create numerous different tables and graphs depending on the input file uploaded and the options chosen.

DATA EXPLORATION

Inside Data Exploration, there are two modules to deal with BDS and subject-level data separately. The first step for either is to upload a data file. A variety of file formats are accepted, including SAS® data, Microsoft Excel®, and CSV. After a file has been uploaded, the contents will be displayed within the app. The *DT* package was used to render the subject data listing inside an interactive table powered by the *DataTables* library. This allows users to check and filter the data in a multitude of ways. Most of the other tables in this section also make use of this package.

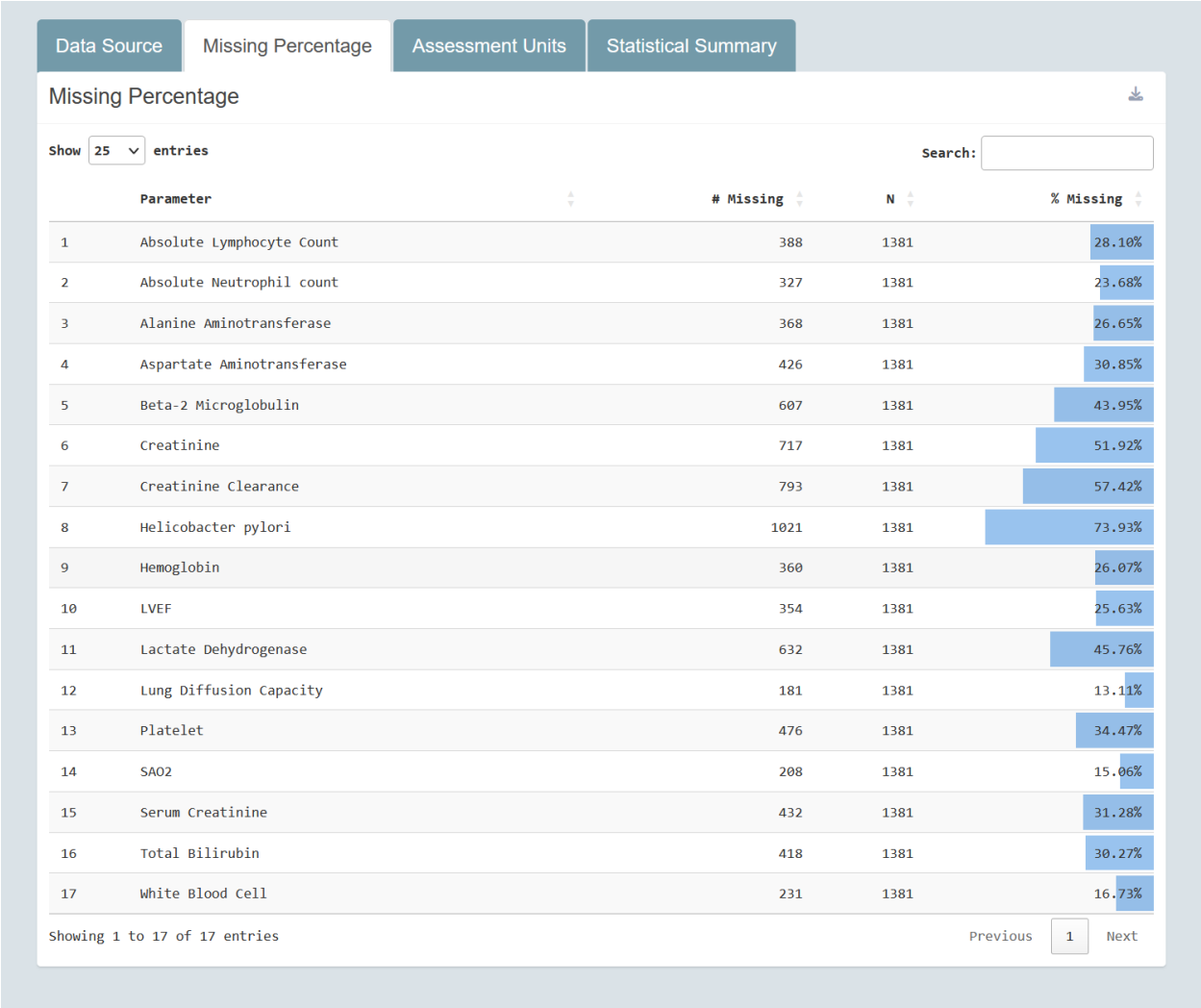
Figure 1. *Data Source page for subject-level data with an ADSL file uploaded.*



Now, a few of the additional tabs will be discussed in further detail.

To monitor data quality, the missing percentage tab shows a breakdown of missing values in the data, grouped by parameter. Based on the number missing and the total number of subjects, the missingness is displayed in the rightmost column, with a graphical depiction of the percentage shown for ease of viewing.

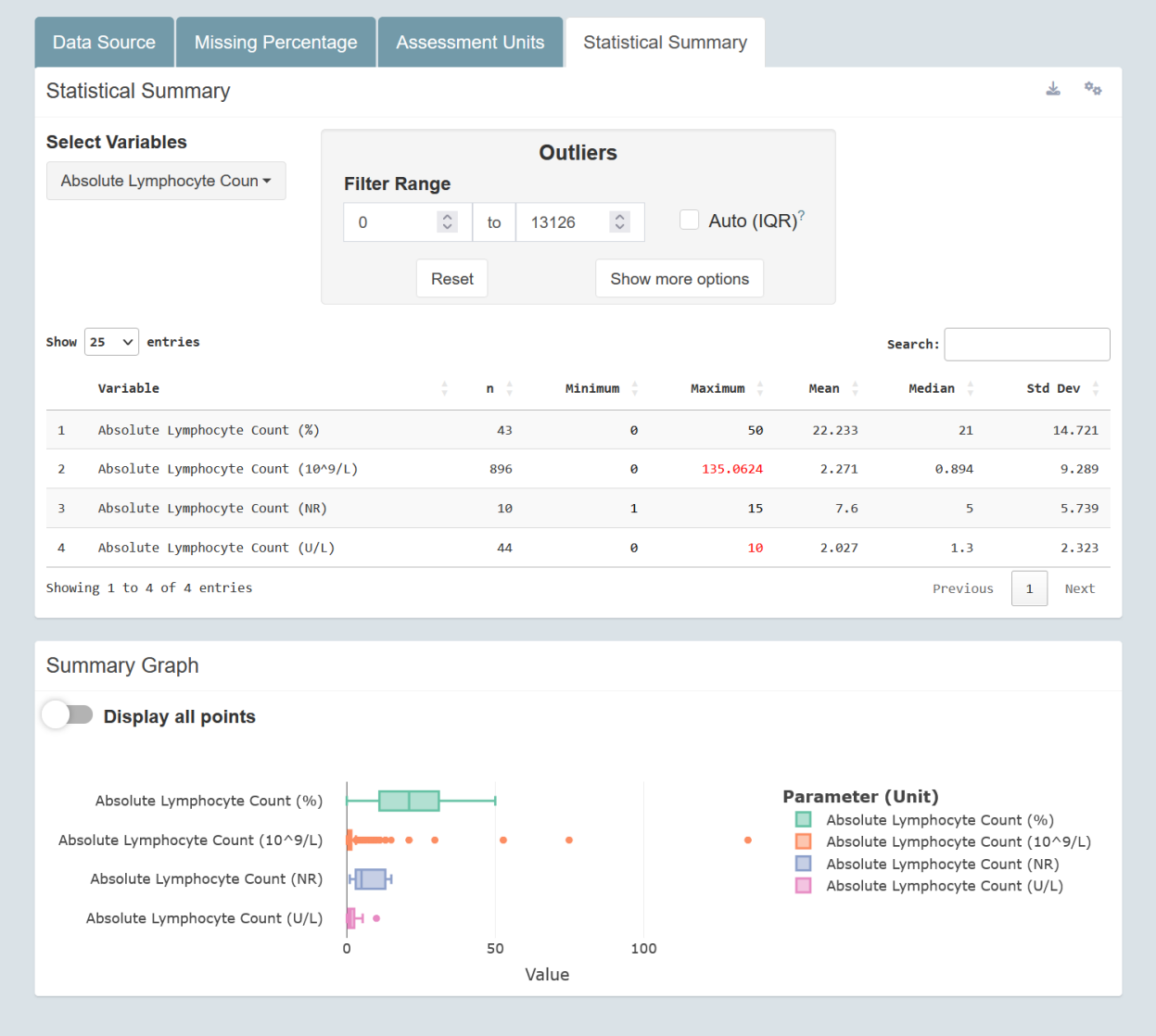
Figure 2. *Missing Percentage page with table showing missingness in data.*



	Parameter	# Missing	N	% Missing
1	Absolute Lymphocyte Count	388	1381	28.10%
2	Absolute Neutrophil count	327	1381	23.68%
3	Alanine Aminotransferase	368	1381	26.65%
4	Aspartate Aminotransferase	426	1381	30.85%
5	Beta-2 Microglobulin	607	1381	43.95%
6	Creatinine	717	1381	51.92%
7	Creatinine Clearance	793	1381	57.42%
8	Helicobacter pylori	1021	1381	73.93%
9	Hemoglobin	360	1381	26.07%
10	LVEF	354	1381	25.63%
11	Lactate Dehydrogenase	632	1381	45.76%
12	Lung Diffusion Capacity	181	1381	13.11%
13	Platelet	476	1381	34.47%
14	SAO2	208	1381	15.06%
15	Serum Creatinine	432	1381	31.28%
16	Total Bilirubin	418	1381	30.27%
17	White Blood Cell	231	1381	16.73%

The summary tab calculates and displays a table containing a statistical summary of the data. Outliers in the minimum and maximum values are emphasized with red text. A subset of variables/parameters can be chosen. The numeric input inside the “Outliers” box can be used to manually change the range of displayed values, while the automatic option will filter out outliers based on interquartile range (IQR). The corresponding boxplot of the distribution is shown underneath the table. Users can zoom and pan through this interactive graph created using the *plotly* package. By default, the individual plot points are not shown, but this can be toggled with the option displayed above the boxplot.

Figure 3. *Statistical Summary page with summary table and graph.*



When subject-level data is chosen, the population selection tab will be available. Inclusion/exclusion criteria can be applied to create a custom attrition table, thanks to the *visR* package. The choice of filter options comes directly from the columns and values of the data. Users can choose as many different filters as there are variables. They can then select the desired values to keep, either the distinct choices or a numeric/time range. The available selection input is automatically chosen depending on the detected variable type. The resulting attrition table is able to show the size of different subgroups and populations.

Figure 4. *Population Selection page with three filters chosen and attrition table displayed.*

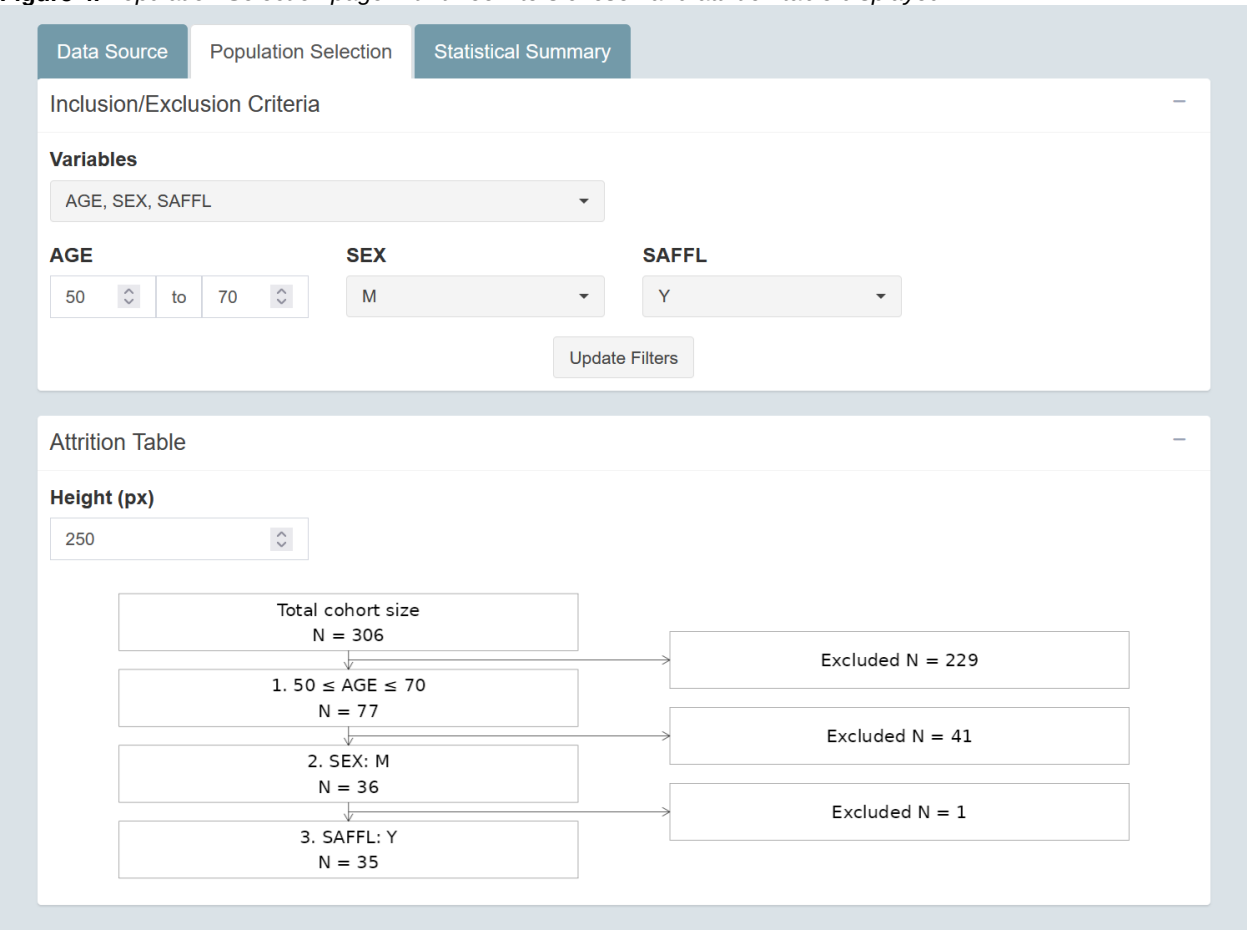
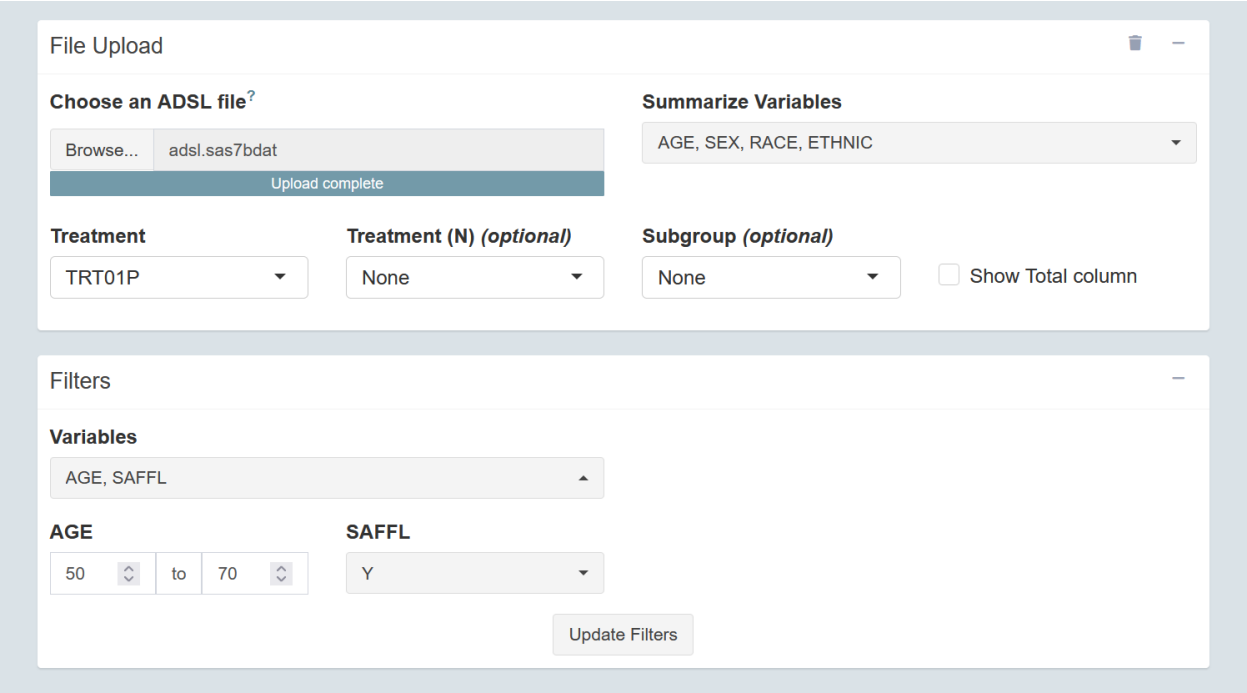


TABLE BUILDER

The Table Builder provides common templates for flexible table generation. The templates currently available include Demographic, Safety (AE), and Efficacy (Summary by Visit, Survival, Response) tables. Like in the previous section, users can upload data files with several different file formats accepted. They can also choose the variables that will be used in table creation. For example, what will the treatment variable be out of the columns available from the data? *TRT01P*, *TRT01A*, or something else? Other options include applying filters based on variables from the input data and adding titles and footnotes to the table. Individual tables will have additional unique options, such as subgroup analysis, selection of statistical methodology and analysis periods, etc. The resulting table is viewable in the application as a *gt* table. This can then be downloaded in RTF and PDF formats, which was made possible with functions from the *reporter* package.

Figure 5. *File Upload and Filters sections of a demographic table.*



Several table examples are shown below.

Table 1. *Demographic table generated by the application.*

Demographic Table			
Variable	Placebo (N=86)	Xanomeline High Dose (N=84)	Xanomeline Low Dose (N=84)
Age			
n	86	84	84
Mean (SD)	75.2 (8.59)	74.4 (7.89)	75.7 (8.29)
Median	76.0	76.0	77.5
Q1, Q3	69.0, 82.0	70.5, 80.0	71.0, 82.0
Min, Max	52, 89	56, 88	51, 88
Sex			
F	53 (61.6)	40 (47.6)	50 (59.5)
M	33 (38.4)	44 (52.4)	34 (40.5)

Table 2. *Adverse Events by SOC/PT table generated by the application (snippet).*

Adverse Events by System Organ Class/Preferred Term			
System Organ Class Preferred Term	Placebo (N=86)	Xanomeline High Dose (N=72)	Xanomeline Low Dose (N=96)
GENERAL DISORDERS AND ADMINISTRATION SITE CONDITIONS	21 (24.4)	36 (50.0)	51 (53.1)
Overall total number of GENERAL DISORDERS AND ADMINISTRATION SITE CONDITIONS	48	118	126
APPLICATION SITE PRURITUS	6 (7.0)	21 (29.2)	23 (24.0)
APPLICATION SITE ERYTHEMA	3 (3.5)	14 (19.4)	13 (13.5)
APPLICATION SITE DERMATITIS	5 (5.8)	7 (9.7)	9 (9.4)
APPLICATION SITE IRRITATION	3 (3.5)	9 (12.5)	9 (9.4)
APPLICATION SITE VESICLES	1 (1.2)	5 (6.9)	5 (5.2)
FATIGUE	1 (1.2)	5 (6.9)	5 (5.2)
APPLICATION SITE URTICARIA	NA	1 (1.4)	2 (2.1)
OEDEMA	NA	NA	2 (2.1)

Table 3. *Survival table generated by the application.*

Survival Table		
Survival	Placebo (N=86)	Xanomeline High Dose (N=84)
Status		
Event (%)	3 (3.5)	2 (2.4)
Censored (%)	83 (96.5)	82 (97.6)
Time in Months		
25th Percentile (95% CI)	64.0 (12.0, NA)	43.0 (43.0, NA)
Median (95% CI)	175.0 (12.0, NA)	64.0 (43.0, NA)
75th Percentile (95% CI)	175.0 (12.0, NA)	64.0 (43.0, NA)
HR (95% CI)		2.79 (0.35, 22.09)
P-value		0.3299
Probability of PFS % (95% CI)		
At 18 Months	80.0 (20.4, 96.9)	
At 24 Months	80.0 (20.4, 96.9)	
Follow-Up (months)		
Median (95% CI)	1.00 (1.00, 1.00)	1.00 (1.00, 1.00)
Min, Max	1.00, 175.00	1.00, 64.00

GRAPH BUILDER

The Graph Builder provides common templates for flexible graph generation. The templates currently available include Survival, Bar, Correlation, Sankey, Tepee, Swimmer, and Forest plots. Like in the Table Builder, users are able to upload data and have customization options available, including filters and titles. The *ggplot2* package was used to create these various graphs. The results are shown in the app and are again downloadable as RTF and PDF files. The user is also able to simply save a PNG of the plot image with a right click.

Several graph examples are shown below.

Figure 6. *Sankey diagram generated by the application.*

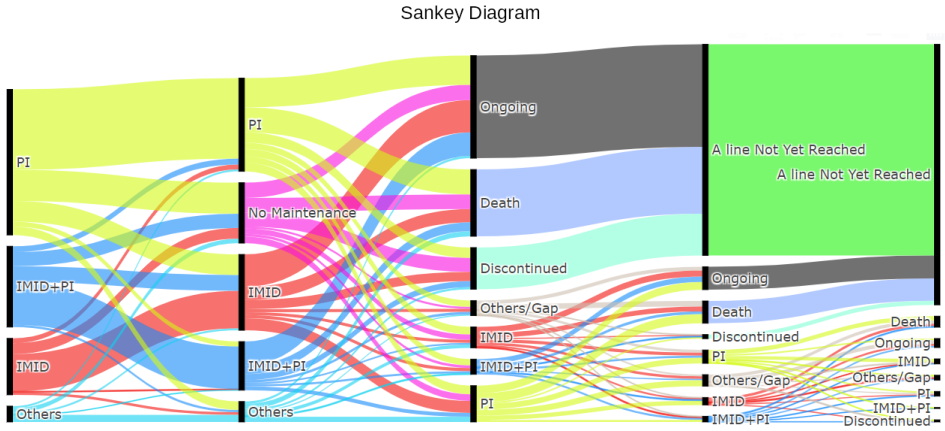


Figure 7. *Forest plot generated by the application.*

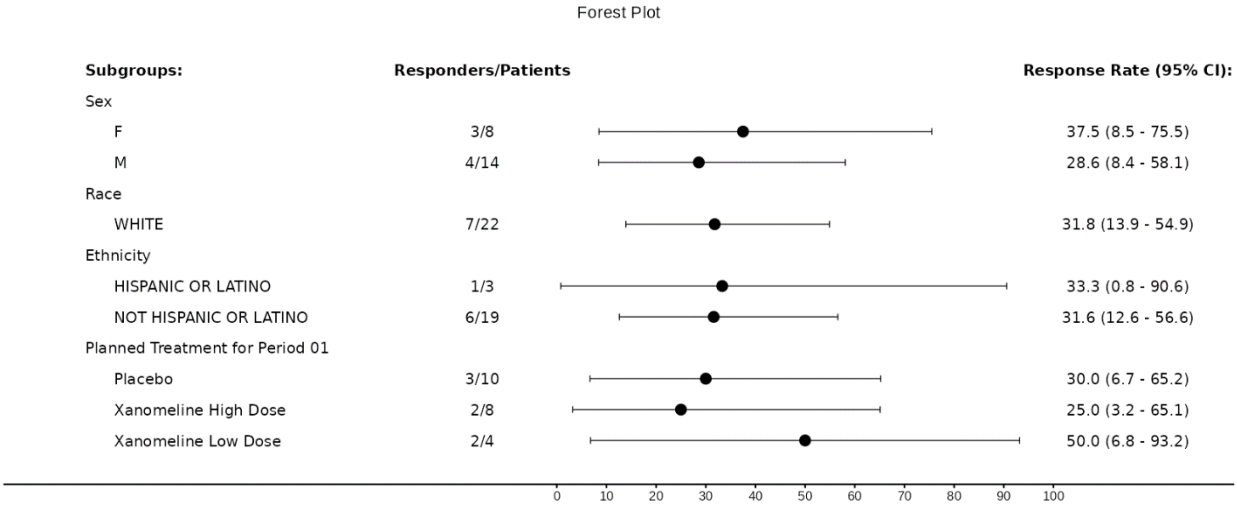
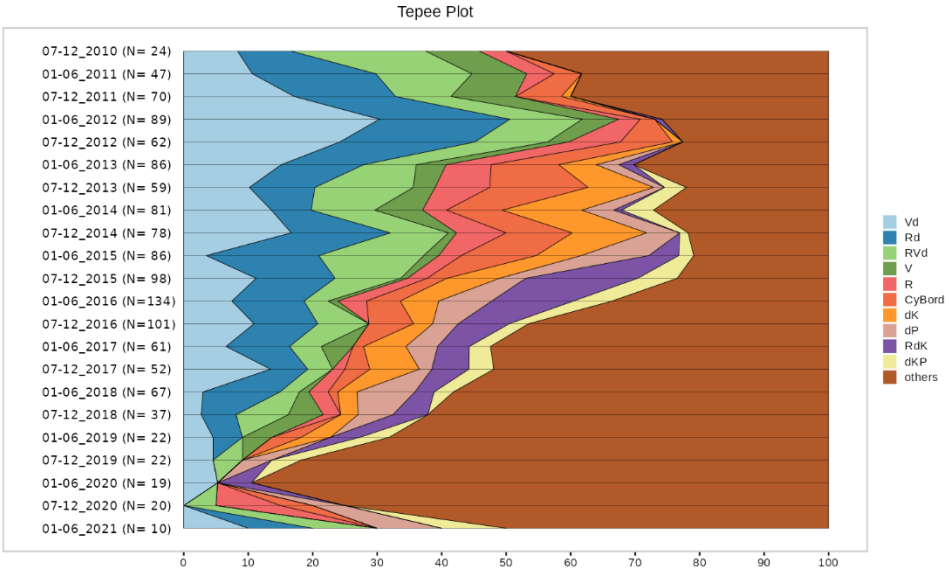


Figure 8. *Tepee plot generated by the application.*



CONCLUSION

This R Shiny application offers users the ability to explore data and create tables and graphs. The flexibility of the Shiny web application framework and other R packages allowed us to create an interface that is simple to navigate and offers many options. Users are able to upload their own data and customize their outputs. This allows for informal preliminary exploration without the need to request formal EBRs. All of this positively impacts the overall budget that is needed to support EBRs.

ACKNOWLEDGMENTS

We would like to thank Joy Wang, Lorraine Fang, and Yanhui “Kevin” Yang for their guidance and contributions on this project.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the authors at:

Diana Tai
Bristol Myers Squibb
Email: diana.tai@bms.com

Lixiao Chen
Bristol Myers Squibb
Email: lixiao.chen@bms.com