R Programming Report – Draft

Elle Holbrook, Katie Key, Brian Kim, Lizette Van Zyl

### Introduction

For the final project, our group was tasked with the input and output of the data via the Shiny app. This involved appropriately "tidying" the different data tables and outputting the other groups' graphs and plots in the Shiny app. Since there were four different data tables, each group member was challenged to tidy one of the data sets. The data sets were labeled: efficacy, plasma, tissue laser, and tissue standard pk. Each member of our group describes the work they did individually to those data sets. One group also describes the work she did to an original template of the data we received. These descriptions are followed by challenges faced and lesson learned by the group as a whole.

### Description of creating the tidy efficacy summary NOT used in Shiny app (Elle)

The tidy efficacy summary file originated as an Excel file with three tables in one sheet. The first contained data on efficacy measured at maximum blood concentration summarized by drug; the second contained similar data on efficacy measured at trough blood concentration; and the third was a metadata table containing brief descriptions of the abbreviated variable names in the columns of all the different data files. Loading in the lung and spleen efficacy tables wasn't difficult, but the metadata were stored in two different columns each containing a different number of rows so it took some time to find a way to combine these pieces of information into one column within a vector or dataframe.

Cleaning the two concentration efficacy tables required 5 basic steps. First, because the first column of the dataset, which contains drug names, had no header in the original file, when the data were loaded in it was named X\_\_1 and needed to be renamed to drug. Next a series of mutate steps was needed. Because the first table contained data obtained at the Cmax and the second at the trough, a column called level was created for each in which all cells indicated either Cmax or trough, respectively. Then the Dose column needed to be split into two variables. Using regular expressions, numerical values from the Dose column were pulled into a new dosage column and other character values were incorporated into a dose\_int column. Next the ULU through MacUptake columns needed to be recoded as numerical values, followed by a select step to remove the Dose column and other columns no longer needed.

The tidy efficacy summary table was created by using bind\_rows to put the Cmax and trough data into one table, converting to a dataframe and writing to a csv file.

### Description of creating the tidy efficacy template used in Shiny app (Katie)

The data set entitled "efficacy" that was used in the Shiny app contained a lot of excess information that was not needed for a "tidy" version. As an example, the data set contained a column called StudyID that repeated the same information about the name of the data set. Similarly, the columns Drug\_Units and Elung\_Units repeated the same units of the measurement for each row of the respective columns. While this information is important, it is unnecessary as columns in a data table.

The main outcomes being examined in this data set were lung efficacy and spleen efficacy based on different combinations of drug, dose, and days of treatment. Therefore, this data set was "tidyed" with this in mind, and the clean version of the data set has these columns: Protocol\_Animal, drug, Group, dosage, days\_treatment, dose\_interval, lung\_efficacy\_log, and speen\_efficacy\_log. To get lung\_efficacy\_log and spleen\_efficacy\_log, we utilized the log() function.

One tricky aspect about cleaning this data set was changing the facotrs in the days\_treatment and dose\_interval (which was originally treatment\_interval). The factors were changed so that there could be comparision between the number of days of treatment, which was either 20 or 40, and have this comparison include controls. In the original data set, controls had 0 listed as days\_treatment. However, we knew which of the controls were for the 20 day period and 40 day period because the treatment\_interval column listed "4 wk" or "8 wk". Since the drug was given five days a week, 4 wk is the 20 day period and 8 wk is the 40 day period. The factor names were changed, combined, and then separated, again, to ensure the correct label lined up with the correct variable in the other column. The days\_treatment column in the "tidy" data set now has each row (that had information) with 0 (for baseline), 20, or 40. The dose\_interval column has factors of: baseline, QD, or control.

### Description of creating the tidy plasma template used in Shiny app (Brian)

The dataset containing information on plasma concentrations was already relatively tidy. Each variable in the dataset was its own column, each observation forms a row, and each type of observational unit can form a table. However, work still had to be done on the dataset containing information on plasma concentrations so it would be in a format that is easier to interpret. The plasma dataset contained information about plasma concentrations in mice for different drugs, drug dosing, and timepoints. The unmodified dataset contained eleven columns. Two of the columns contained no data and were not selected for inclusion in the tidy version of the plasma dataset. Additionally, two columns were removed from the tidy version of the plasma dataset because they contained information that was not relevant to interpret exposure or outcome data. The tidy version of the plasma dataset contained seven columns. The columns were mouse\_number, drug, Group, Protocol\_Animal, Dosing, Timepoint, and plasma\_concentration. The outcome variable is plasma\_concentration and displays the mouse’s plasma concentration in ng/ml.

### Description of creating the tidy tissue laser template used in Shiny app (Lizette)

In the first step of cleaning, unnecesary columns were removed to simplify the data. These included StudyID, Metabolite, Units, Collection, Sample ID since the information in these columns never changed. The units were rather included in the column title for Parent, where the results of the four regions sampled are entered.

To clean the tissue laser capture data further, some assumptions were made about what would be included in the actual data from the research team. For instance, no mice ids were listed in the original data file, but the data was grouped in fours since four samples were taken for each mouse: inner caseum, outer caseum, rim, and uninvolved lung. Each combination of drug type, protocol animal, and dosing was for a single mouse, and each mouse had samples taken from each region of the lung. Using this information, mice ids were assigned by creating a list of numbers where each number was repeated four times and the highest number was equal to the total number of sample entries divided by four. The original data had to be altered slightly so that the numbers lined up, as a few of the rows for uninvolved lung were not in the data with NA values, but rather had no row included at all. These rows were manually input into the orignial Excel file. In the future, the research group will need to ensure that all four lung samples are entered for each mouse, and that missing data has NA listed instead of not including that row of data in the file.

After this, the data was spread so that the each mouse id had four columns, each relating to a different lung sample area (ULU, RIM, ICS, OCS), instead of repeating the information for each mouse in four different rows. These entries were then switched to a numeric class, and the dataframe was saved as a csv file.

### Description of creating the tidy tissue standard PK template used in Shiny app (Elle)

The standard PK data template contained information on the pharmacokinetic properties of the drugs tested on a mouse-by-mouse level (not averaged for each drug). The pharmacokinetic properties of the drugs were measured at different timepoints (typically Cmax and trough) and in different compartments (within the lesion and within the general lung tissue) for each mouse.

The original template did not contain mouse IDs of any kind, so once the data were loaded in a vector of IDs was created to add as a new column in the dataset. First, however, the data needed to be examined to determine how many unique mice the file contained; it was determined that for each distinct combination of Compound, Group and Protocol\_Animal, two measurements were taken (lung and lesion compartments) and it was assumed that each of these pairs were taken from one mouse. Therefore, the numerical ID vector was created to match the length of the dataframe and have each ID repeated two times. select was used to limit the dataset to only the necessary columns, the Compound and Parent columns were renamed to be more descriptive and then spread was used to convert the Comparment column into columns for each compartment, containing the respective Parent values. These new columns were then renamed to match the SLE and SLU variable names in the tidy data templates and recoded as numerical values.

### Challenges

One of the biggest challenges, which we still don't quite have an answer to, is that the dosage and dose interval data are missing from the mouse-by-mouse level spreadsheets. Another efficacy template contains this information but each drug appears to have more than one dose and/or dose interval within that spreadsheet so it can't be assumed that every mouse received the same dose/dose interval for a given compound. Additionally, we realized after creating the tidy data templates that it was much more challenging to relate information between the different tables, so it couldn't be assumed that a mouse within the standard PK template was also listed within the laser capture template, for example; therefore, these mouse-by-mouse level templates could not be combined into one file. Additionally, assumptions were made about the units of observation for these files (described above).

Therefore, one of the greatest challenges we faced when trying to tidy the different data frames was determining what information the principal investigators wanted in the tidy version of the dataset. We also had challenges understanding what all the variables and columns meant due to minimal background and knowledge in microbiology.

In regards to the output of the data tables and graphical representations in the Shiny app, the biggest challenged we faced was learning the coding that is unique to Shiny apps. The codes needed to effectively run at Shiny app are written a bit differently than those needed to run functions in a .R file, for example. Further, the Shiny app is very particular in how the ui and server aspects of the app work together. We had a difficult time getting the Shiny app to upload and recognize the Excel file inputs before Dr. Anderson helped us write the code for it.

### Do Differently

If we had the opportunity to go back, we feel it would have been beneficial to contact the principal investigators earlier. Earlier contact with the principal investigators would ensure a mutual understanding of what variables were of greatest importance and a better understanding of the background knowledge in microbiology necessary to understand the data. Further, communicating with the research group sooner could have allowed us to work more efficienty. We ended up "tidying" two sets of templates while only needing one for the Shiny app and for the other groups. Unfortunately, we believe groups may have been using the original templates to write their functions and develop their graphs and not the ones that would actually be used by the research group utilizing the Shiny app we created.

Our group also feel that it would have been beneficial to communicate with the other two groups sooner, not only to ensure that they understand the files we created but also to help us understand what graphs, images, and relationships they were looking at and would like to see within the final Shiny app.

### Interesting Things in Data

It was interesting to think about the challenges surrounding identifying individual mice and how their data could be related to that of other mice in the studies. We hadn't previously considered the issues inherent in the fact that one mouse, as small as it is, can only be used for so many measurements before another mouse is needed to round out the data for a given drug.

Overall, we learned a lot about the importance of "tidy" data using real data. It was an important lesson to learn that not all data we may receive will be clean. However, it was rewarding to take the research group's data collected from a lot of time and hard work and be able to give them a Shiny app that allows them to visually and analytically explore their data all in one go.