EMIT\_Enrollment\_Summary.R

jbueno

Thu Feb 14 11:20:49 2019

# Title: EMIT\_Enrollment\_Summary  
# Author: Jacob Bueno de Mesquita cleaned up the script originally by Jing Yan and Don Milton  
# Date: January 7; February 2019  
  
# Summary:  
# This script does some basic data summarizing for the study and it appears to have been useful during the ongoing participant enrollment process and for periodic checks of the data and reports. I have pulled together and cleaned up code from 2 scripts: The "Snippets analysis\_1.r", and the "field database with redcap culture.R".  
# Of importance, the Clinical Database and the G2 Log, which are used to produce an enrollment summary which is written out as "/Users/jbueno/Box Sync/EMIT/EMIT\_Data\_Analysis\_Jake/EMIT\_UMD\_Natural\_Infection/Curated Data/Cleaned Data/Enrollment\_Summary.csv"  
# This enrollment summary give s a list of the subject IDs, whether they were enrolled in the study (1 if yes, 0 if no) and a "perform\_date", which is mostly likely the date of first G-II visit with each particular subject ID.   
# A set of roommate pairs is also identified in this script and printed to the console, although not saved as an output dataset. This list of roommates is critical for sequence analysis to test for transmission between roommate pairs.   
# Finally, a bit of data cleaning and summarizing, with comments is done with the Field Sample Database and the REDCap Database.   
  
#### \*\*\*\* Using Script: Jing Yan and Dr. Milton's "Snippets analysis\_1.r" \*\*\*\* ####  
  
# Perhaps the earlier lines of script in this program address what "Snippets analysis\_1.r" was mostly getting at.  
# However, "Snippets analysis\_1.r" provides some, perhaps useful, summary information and looks at roommates.   
  
###  
# Original file information:  
  
# By Jing Yan & Don Milton  
# December 14, 2015 - December 21, 2015  
# Purpose: Combine clinical redcap data with GII data from redcap and check that all subject enrolled  
# according to redcap clinical data have the appropriate number of GII records and that  
# persons not enrolled only have screening visits (up to 3) and no GII records.  
# Will also generate a list of subjects showing whether they were enrolled and how many GII  
# sessions they completed.   
# Will identify roommate screenings and roommates enrolled after screening as roommates.   
  
###  
  
#### Setting up the environment ####  
  
library(dplyr)  
library(tidyr)  
  
sessionInfo()

## R version 3.5.1 (2018-07-02)  
## Platform: x86\_64-apple-darwin15.6.0 (64-bit)  
## Running under: macOS 10.14.3  
##   
## Matrix products: default  
## BLAS: /System/Library/Frameworks/Accelerate.framework/Versions/A/Frameworks/vecLib.framework/Versions/A/libBLAS.dylib  
## LAPACK: /Library/Frameworks/R.framework/Versions/3.5/Resources/lib/libRlapack.dylib  
##   
## locale:  
## [1] en\_US.UTF-8/en\_US.UTF-8/en\_US.UTF-8/C/en\_US.UTF-8/en\_US.UTF-8  
##   
## attached base packages:  
## [1] stats graphics grDevices utils datasets methods base   
##   
## other attached packages:  
## [1] bindrcpp\_0.2.2 lme4\_1.1-19 Matrix\_1.2-14   
## [4] arsenal\_1.5.0 lubridate\_1.7.4 data.table\_1.11.8  
## [7] knitr\_1.20 readxl\_1.1.0 RcppRoll\_0.3.0   
## [10] forcats\_0.3.0 stringr\_1.3.1 dplyr\_0.7.7   
## [13] purrr\_0.2.5 readr\_1.1.1 tidyr\_0.8.2   
## [16] tibble\_1.4.2 ggplot2\_3.1.0 tidyverse\_1.2.1   
## [19] htmlTable\_1.12 rmarkdown\_1.10 markdown\_0.8   
##   
## loaded via a namespace (and not attached):  
## [1] Rcpp\_0.12.19 lattice\_0.20-38 utf8\_1.1.4 assertthat\_0.2.0  
## [5] rprojroot\_1.3-2 digest\_0.6.18 R6\_2.3.0 cellranger\_1.1.0  
## [9] plyr\_1.8.4 backports\_1.1.2 evaluate\_0.12 highr\_0.7   
## [13] httr\_1.3.1 pillar\_1.3.0 rlang\_0.3.0.1 lazyeval\_0.2.1   
## [17] rstudioapi\_0.8 minqa\_1.2.4 nloptr\_1.2.1 checkmate\_1.8.5   
## [21] splines\_3.5.1 htmlwidgets\_1.3 munsell\_0.5.0 broom\_0.5.0   
## [25] compiler\_3.5.1 modelr\_0.1.2 pkgconfig\_2.0.2 htmltools\_0.3.6   
## [29] tidyselect\_0.2.5 fansi\_0.4.0 crayon\_1.3.4 withr\_2.1.2   
## [33] MASS\_7.3-51 grid\_3.5.1 nlme\_3.1-137 jsonlite\_1.5   
## [37] gtable\_0.2.0 magrittr\_1.5 scales\_1.0.0 cli\_1.0.1   
## [41] stringi\_1.2.4 testthat\_2.0.1 xml2\_1.2.0 tools\_3.5.1   
## [45] glue\_1.3.0 hms\_0.4.2 yaml\_2.2.0 colorspace\_1.3-2  
## [49] rvest\_0.3.2 bindr\_0.1.1 haven\_1.1.2

#### READ in and work with Clinical Database ####  
  
clinical\_in\_file <- "/Users/jbueno/Box Sync/EMIT/EMIT\_Data\_Analysis\_Jake/EMIT\_UMD\_Natural\_Infection/UMD\_Raw\_Data/REDCAP/EMITClinicalUMD2013.csv"  
clinical\_umd <- read.csv(clinical\_in\_file)  
  
## Check whether there was anyone with no visit 1 who has a record for having had a g2 run ##  
  
g2\_id <- unique(clinical\_umd$field\_subj\_id[grep('^g2', clinical\_umd$redcap\_event\_name)])  
  
visit1\_id <- unique(clinical\_umd$field\_subj\_id[grep('^visit', clinical\_umd$redcap\_event\_name)])  
  
# Number of subjects with at least 1 g2 visit who also have a visit 1  
print(sum(g2\_id %in% visit1\_id))

## [1] 178

# Number of subjects with at least one g2 visit regardless of having a visit 1  
print(length(g2\_id))

## [1] 178

# If the above two prints are the same, then everyone who gave a g2 sample had a visit 1 screening visit.  
  
## Check to see if there are persons with visits 2 or 3 and no visit 1 ##  
  
# Note that screening visits were only identified as screening in the redcap names for visits 2 and 3 but visit 1 was also screening  
  
visit2\_3gp <- clinical\_umd %>%   
 group\_by(field\_subj\_id, redcap\_event\_name) %>%   
 summarise(n = n())  
  
visit2\_3ID <- visit2\_3gp$field\_subj\_id[grepl('^screen', visit2\_3gp$redcap\_event\_name)]  
  
no\_visit2\_3ID <- unique(visit2\_3gp$field\_subj\_id[!grepl('^screen', visit2\_3gp$redcap\_event\_name)])  
  
# Subjects with Visit 2 or 3 who also have a Visit 1  
print(sum(visit2\_3ID %in% visit1\_id))

## [1] 21

# Number of subjects with at least one visit 2 or 3 arm regardless of having a visit 1  
print(length(visit2\_3ID))

## [1] 21

## Subjects receiving second and third screening visits ##  
  
visit2\_3DT <- clinical\_umd %>%   
 filter(grepl('^screen', redcap\_event\_name)) # all screening visit 2 and 3 records i.e. screened more than once  
  
visit2\_3DT2 <- clinical\_umd %>%   
 filter(field\_subj\_id %in% visit2\_3DT$field\_subj\_id) # all records for subj with a visit 2 or 3  
  
visit2\_3DT21 <- visit2\_3DT2 %>%   
 filter(grepl(0, is\_rmmate)) # select non roommate referrals  
  
# nonrr\_enroll\_visit2\_3 contains the records for people who were not roommate referrals and had visits 2 or 3.  
nonrr\_enroll\_visit2\_3 <- visit2\_3DT21 %>%   
 left\_join(visit2\_3DT2, by = "field\_subj\_id") %>%   
 select(field\_subj\_id, redcap\_event\_name.y)  
  
# Subjects 114, 127, and 250 have more than one screen visits but not referred by roomate.   
# All of them got enrolled. 114 has 2 screen visits, 127 has 3 screen visits, and 250 has 2 screen visits  
  
# The following subjects had more than one screening visit but were not roommates.  
unique(visit2\_3DT21$field\_subj\_id)

## [1] 114 127 250

# List of subjects and visits for repeatedly screened persons who were not roommate referrals  
print(nonrr\_enroll\_visit2\_3)

## field\_subj\_id redcap\_event\_name.y  
## 1 114 visit\_1\_part\_a\_arm\_1  
## 2 114 screen\_visit\_2\_arm\_1  
## 3 114 g2\_run\_1\_arm\_1  
## 4 127 visit\_1\_part\_a\_arm\_1  
## 5 127 screen\_visit\_2\_arm\_1  
## 6 127 screen\_visit\_3\_arm\_1  
## 7 127 g2\_run\_1\_arm\_1  
## 8 250 visit\_1\_part\_a\_arm\_1  
## 9 250 screen\_visit\_2\_arm\_1  
## 10 250 g2\_run\_1\_arm\_1

visit2\_3DT3 <- visit2\_3DT2 %>%   
 filter(!grepl('^screen', redcap\_event\_name)) # all records except visit 2 or 3 records for subj with a visit 2 or 3  
  
# check number of records including screening visits  
# Number of subjects with a visit 2 or 3 and a visit 1 or were also renrolled  
print(sum(unique(visit2\_3DT2$field\_subj\_id) %in% unique(visit2\_3DT3$field\_subj\_id)))

## [1] 20

# Number of all subject with a visit 2 or 3  
print(length(unique(visit2\_3DT2$field\_subj\_id)))

## [1] 20

#### Roommates ####  
  
# identify the roommate referal subject   
rr <- clinical\_umd %>%   
 select(field\_subj\_id, redcap\_event\_name, is\_rmmate, indx\_id)  
  
# rr1 gives the the list of all the subjects referred by roomate with only the visit1\_arm  
rr1 <- rr %>%   
 filter(rr$is\_rmmate == 1)  
  
# Total number of roommate referrals screened  
print(length(rr1$field\_subj\_id))

## [1] 46

# rr2 shows all the arms and all fields  
rr2 <- rr1 %>%   
 left\_join(rr, by = c('field\_subj\_id'))  
rr3 <- rr2 %>%   
 select(field\_subj\_id, redcap\_event\_name.y, is\_rmmate.y, indx\_id.y) # limits fields from rr2  
  
# The following roommate referred subjects were screened  
print(unique(rr3$field\_subj\_id))

## [1] 133 134 135 144 147 152 160 178 180 190 199 205 206 209 217 218 219  
## [18] 220 221 224 237 239 246 254 259 263 265 268 269 275 278 279 280 283  
## [35] 286 295 307 312 324 325 330 332 334 335 354 362

rr4 <- rr3 %>%   
 filter(grepl('^g2', redcap\_event\_name.y))  
  
# The number of roommate referrals enrolled was  
print(length(rr4$field\_subj\_id))

## [1] 4

# The following roommate referred subjects were enrolled  
print(unique(rr4$field\_subj\_id))

## [1] 178 237 265 335

rr5 <- rr4 %>%   
 left\_join(rr3, by = c('field\_subj\_id')) %>%   
 select(field\_subj\_id, redcap\_event\_name.y.y, is\_rmmate.y.y, indx\_id.y.y)  
  
# Roommate referred subjects who were enrolled and their index ID  
print(rr5)

## field\_subj\_id redcap\_event\_name.y.y is\_rmmate.y.y indx\_id.y.y  
## 1 178 visit\_1\_part\_a\_arm\_1 1 143  
## 2 178 g2\_run\_1\_arm\_1 NA NA  
## 3 237 visit\_1\_part\_a\_arm\_1 1 223  
## 4 237 g2\_run\_1\_arm\_1 NA NA  
## 5 265 visit\_1\_part\_a\_arm\_1 1 255  
## 6 265 g2\_run\_1\_arm\_1 NA NA  
## 7 335 visit\_1\_part\_a\_arm\_1 1 328  
## 8 335 screen\_visit\_2\_arm\_1 NA NA  
## 9 335 g2\_run\_1\_arm\_1 NA NA

## Method 1, find the enrolled subjects and how many times they were enrolled ##  
  
enroll <- clinical\_umd %>%   
 filter(grepl('^g2\_run\_1', redcap\_event\_name)) %>%   
 group\_by(field\_subj\_id, redcap\_event\_name)  
  
# The total number of enrolled subjects was  
print(length(enroll$field\_subj\_id))

## [1] 178

enroll3 <- clinical\_umd %>%   
 filter(grepl('^g2\_run\_3', redcap\_event\_name)) %>%   
 group\_by(field\_subj\_id, redcap\_event\_name)  
  
# The total number of subjects that have three GII tests was  
print(length(enroll3$field\_subj\_id))

## [1] 27

enroll2 <- clinical\_umd %>%   
 filter(grepl('^g2\_run\_2', redcap\_event\_name)) %>%   
 group\_by(field\_subj\_id, redcap\_event\_name)  
  
# The total number of subjects that have two GII tests was  
print(length(enroll2$field\_subj\_id) - length(enroll3$field\_subj\_id))

## [1] 45

# The total number of subjects that have only one GII tests was  
print(length(enroll$field\_subj\_id) - length(enroll2$field\_subj\_id))

## [1] 106

# Subject 69 has 2 GII tests from redcap clinical record, but only have one GII record in the GII log on redcap, need further investigation.   
  
#### More data exploration and manipulation ####  
  
# remove the screen arm from the data, leave the field\_subj\_id and redcap\_event\_name and mark n = 1 for each redcap\_event\_name  
tab0 <- clinical\_umd %>%   
 filter(!grepl('^screen', redcap\_event\_name)) %>%   
 group\_by(field\_subj\_id, redcap\_event\_name) %>%   
 summarise(n = n())  
  
# tab1: remove screen\_arm and left with visit\_1 and g2\_arm, and count the number of visit\_1 and g2\_arm  
tab1 <- clinical\_umd %>%   
 filter(!grepl('^screen', redcap\_event\_name)) %>%   
 group\_by(field\_subj\_id, date\_enroll) %>%   
 summarise(n = n())  
  
# tab1test, remove screen\_arm and only picked the field\_sub\_id and count the number of appearance(each subject should have at least 1 )  
tab1test <- clinical\_umd %>%   
 filter(!grepl('^screen', redcap\_event\_name)) %>%   
 group\_by(field\_subj\_id) %>%   
 summarise(n = n())  
  
# tab1new, add a colunm called enroll, and for n = 2 or > 2, means the subject need to have at least one time g2, so it is enrolled(1)  
# otherwise it is not enrolled(0). also add a new colunm called GII\_time, if if n = 2, means one g2\_arm, so GII\_time = n-1 = 1, apply   
# the same method for 2 and 3 GII\_times  
tab1new <- tab1test %>%   
 mutate(enroll = ifelse(n >= 2, 1, 0),   
 GII\_time = ifelse(n >= 2, n-1, 0))  
  
# tab1new1 sorted data for both enrolled and unenrolled subjects  
tab1new1 <- tab1new %>%   
 select(field\_subj\_id, enroll, GII\_time)  
names(tab1new1)[1] <- "subject\_id"  
  
# get the enrolldate correspond with the field\_subj\_id  
enrollDate <- clinical\_umd %>%   
 filter(grepl('^visit', redcap\_event\_name) & date\_enroll != '') %>%   
 select(field\_subj\_id, date\_enroll)  
  
# Number of duplicated enrolldate for the same subject id  
sum(duplicated(enrollDate$field\_subj\_id))

## [1] 0

# merge the enrolldate file with the ta1new which contained enroll, GII\_time, by subject\_id  
tab1link <- tab1new %>%   
 left\_join(enrollDate, by = 'field\_subj\_id') %>%   
 select(field\_subj\_id, enroll, GII\_time, date\_enroll)  
  
# check if tab1link has the same number of rows as tab1new,since the row numbers are the same, and there is no missing data for date\_enroll in tab1link  
# so the enrolldate match with the enrollcheck and GII\_times  
nrow(tab1link)

## [1] 355

nrow(tab1new)

## [1] 355

sum(tab1link$date\_enroll == '')

## [1] 0

names(tab1link)[1] <- "subject\_id"  
  
# subject which are enrolled  
tab2link <- tab1link %>%   
 filter(tab1link$enroll == 1)  
  
# subject came in only for screening visits  
tab3link <- tab1link %>%   
 filter(tab1link$enroll == 0)  
names(tab3link)[4] <- 'first\_visit\_date'  
  
#### READ in and work with the G2 Log Data ####  
  
g2\_in\_file <- "/Users/jbueno/Box Sync/EMIT/EMIT\_Data\_Analysis\_Jake/EMIT\_UMD\_Natural\_Infection/UMD\_Raw\_Data/GII/EMITGIILogUMD2013.csv"  
g1 <- read.csv(g2\_in\_file)  
  
# Sujbect 81 in GII file appear to have a collection\_2\_arm 1 but it actually is a one time GII subject  
sum((g1$start\_dt) == "")

## [1] 1

# Remove the second arm of subject 81 (REVISE TO ID ROWS WITHOUT START DATE)  
g2 <- filter(g1, !(g1$start\_dt) == "")  
  
# Pick the subject id and start date and order them  
g3 <- g2 %>%   
 select(subject\_id, start\_dt) %>%   
 arrange(subject\_id, start\_dt)  
names(g3)[2] <- "perform\_date"  
  
# Remove the duplicated and just save the first come in date  
g4 <- g3[!duplicated(g2$subject\_id), ]  
  
# Group the original subject and number them  
g5 <- g3 %>%   
 group\_by(subject\_id) %>%   
 summarise(n = n())  
  
# g6 tells us all many times the subject enrolled for gii study  
g6 <- g5 %>%   
 mutate(GII\_time = ifelse(n >= 1, n, 0))  
  
# Merge g4 (first come in date) wit the GII\_time  
g7 <- g4 %>%   
 left\_join(g6, by = 'subject\_id') %>%   
 select(subject\_id, GII\_time, perform\_date)  
  
# Change the start\_dt as date\_enroll  
names(g7)[3] <- "date\_enroll"  
  
#### Producing Enrollment Summary from Clinical Database and G2 Log Data ####  
  
# Merge the subject id with enroll time date\_enroll, sample perform date, only the enrolled the subject  
g8 <- g3 %>%   
 left\_join(g7, by = 'subject\_id') %>%   
 select(subject\_id, GII\_time, date\_enroll, perform\_date)  
  
# Check the gii subject list with the enrolled list from culture study  
m <- g7 %>%   
 left\_join(tab1link, by = c('subject\_id', 'GII\_time', 'date\_enroll'))

## Warning: Column `date\_enroll` joining factors with different levels,  
## coercing to character vector

sum(is.na(m$GII\_time))

## [1] 0

m2 <- tab1link %>%   
 left\_join(g8, by = 'subject\_id', 'GII\_time') %>%   
 select(subject\_id, enroll, perform\_date)  
  
## Write out the enrollment summary ##  
write.csv(m2, "/Users/jbueno/Box Sync/EMIT/EMIT\_Data\_Analysis\_Jake/EMIT\_UMD\_Natural\_Infection/Curated Data/Cleaned Data/Enrollment\_Summary.csv")  
  
#### \*\*\*\* Using Script: Jing Yan's "field database with redcap culture.R" \*\*\*\* ####   
  
###  
## Original file information:  
  
# "field data base check with redcap culture.R"  
# by Jing Yan  
# December 16, 2015  
# Purpose:check if all the redcap culture sample id match with the field id   
  
###  
  
#### READ in and work with FIELD SAMPLE DATABASE ####  
  
field\_db\_in\_file <- "/Users/jbueno/Box Sync/EMIT/EMIT\_Data\_Analysis\_Jake/EMIT\_UMD\_Natural\_Infection/UMD\_Raw\_Data/EMIT UMD Field\_db/field\_db.csv"  
a <- read.csv(field\_db\_in\_file, as.is = T)  
  
names(a)

## [1] "SUBJECT\_IDENTIFIER" "SAMPLE\_ID" "COLLECTION\_DT"   
## [4] "ID" "TYPE\_NAME" "UNIT\_NAME"   
## [7] "RAPID\_TEST"

a1 <- a %>%   
 select(SUBJECT\_IDENTIFIER, SAMPLE\_ID, COLLECTION\_DT, TYPE\_NAME)  
names(a1)[2] = "sample\_id"  
names(a1)[3] = "Dates\_a"  
names(a1)[4] = "Sample.Type"  
  
# Add a new column as newdate which the same as collection\_dt but with format m/d/y  
a2 <- a1 %>%   
 mutate(newdate = as.Date(Dates\_a, format = '%m/%d/%Y'))  
  
#### READ in and work with UMD SAMPLES DATABASE (REDCAP DATA) ####  
  
# Read in redcap\_culture data  
  
sample\_in\_file <- "/Users/jbueno/Box Sync/EMIT/EMIT\_Data\_Analysis\_Jake/EMIT\_UMD\_Natural\_Infection/UMD\_Raw\_Data/REDCAP/EMITUMDSamples2013\_DATA.csv"  
b <- read.csv(sample\_in\_file, as.is = T)  
  
# Add a new column named as new date which is the same with date of sample collection  
b2 <- b %>%   
 mutate(newdate = as.Date(dt\_visit, format = '%m/%d/%Y'))  
  
#b3 proves that there is no missing date of sample collection in the redcap culture file  
b3 <- b2 %>%   
 filter(dt\_visit != '') %>%  
 rename(Sample.Type = sample\_type)  
  
m <- b3 %>%   
 left\_join(a2, by = c('sample\_id', 'newdate', 'Sample.Type'))  
  
m1 <- m %>%   
 select(sample\_id, newdate, Sample.Type, Dates\_a)  
  
m11 <- m1 %>%   
 select(sample\_id, newdate, Sample.Type)  
  
sum(is.na(m1$Dates\_a))

## [1] 1

m2 <- m1 %>%   
 filter(!grepl('.', Dates\_a))  
  
# This result suggests that all the redcap culture sample IDs and sample types and enrolled data match with the field ID data.   
# Only 237\_6 was included in the field data base but not the redcap culture data.   
# 237 is an enrolled subject for 1 time, it should not have a \_6 sample  
# From the above code we know all the culture redcap sample ID can be found in field ID, not sure if all the field ID can be found in culture redcap  
  
m3 <- a2 %>%   
 left\_join(b3, by = c('sample\_id', 'newdate', 'Sample.Type'))  
  
m4 <- m3 %>%   
 select(sample\_id, newdate, Sample.Type)  
  
sum(is.na(m4$Date.of.sample.collection))

## [1] 0

m5 <- m4 %>%   
 filter(!grepl('.', newdate))  
# m5 are the samples that were included in the field\_id but not included in the redcap culture  
  
# There is no output for this section - rather this is part of the data checking and exploratory analysis.   
# It also contains some interesting roommate information.   
# A report can be generated from the objects in this piece of the script if desired.